

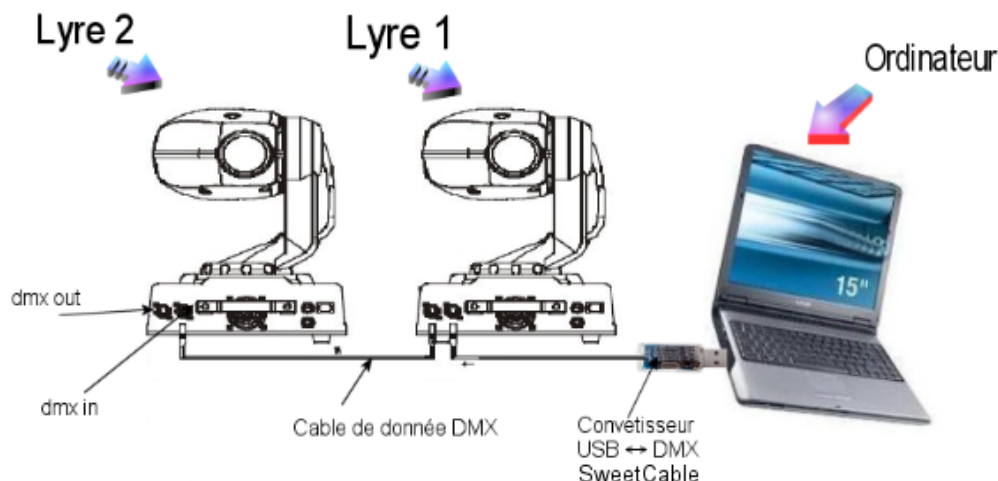
PROJET N°1

Problème technique 1:

On cherche à piloter deux projecteurs lyre ROBE 150CT pourvus de huit canaux, en utilisant le logiciel "SWEET LIGHT" pour appréhender les techniques de conception des scènes de lumière qui répondent à un cahier de charges de spectacle.

Système d'éclairage mettant en œuvre deux lyres

Illustration :



Présentation de la lyre:

- **Fonctionnalité :**

La lyre sert à créer une ambiance dans les soirées ou spectacles en émettant un faisceau de lumière paramétrable (couleur, motif, intensité, etc...) , elle est utilisée par les disc-jockeys et les animateurs de soirées.

- **Utilisation :**

La lyre est contrôlable par l'intermédiaire d'une console ou d'un ordinateur qui possède le logiciel prévu à cet effet (Sweetlight). De plus il est possible d'utiliser plusieurs lyres simultanément via l'interface DMX.

- **Caractéristique :**

La lyre est pivotable autour de l'axe X qui fait un angle de 530° et autour de l'axe Y qui fait un angle de 280°. La vitesse de rotation est réglable. On peut choisir la couleur parmi les couleurs proposées choisir les gobos, régler l'intensité lumineuse et la vitesse de défilement des gobos.

- **Description des fonctions :**

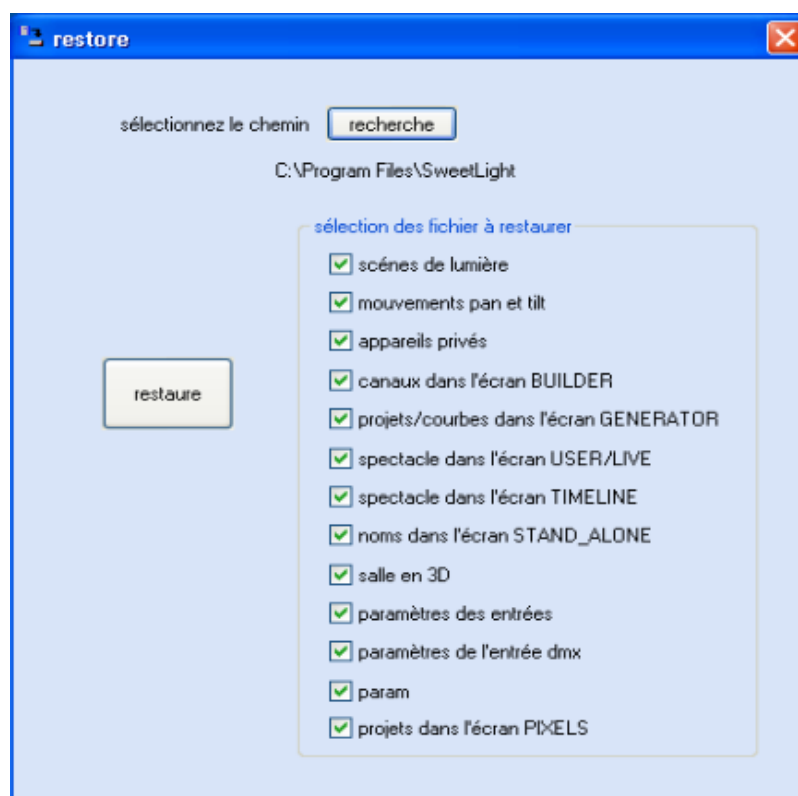
	Fonction rotation sur l'axe X.
	Fonction rotation sur l'axe Y.
	Fonction réglage de la vitesse de rotation sur les axes X et Y.
	Fonction choix des gobos.
	Fonction choix de la couleur.
	Fonction réglage de l'intensité lumineuse et fermeture/ouverture du volet.

Procédure d'ajout d'un projecteur Robe 150CT a huit canaux

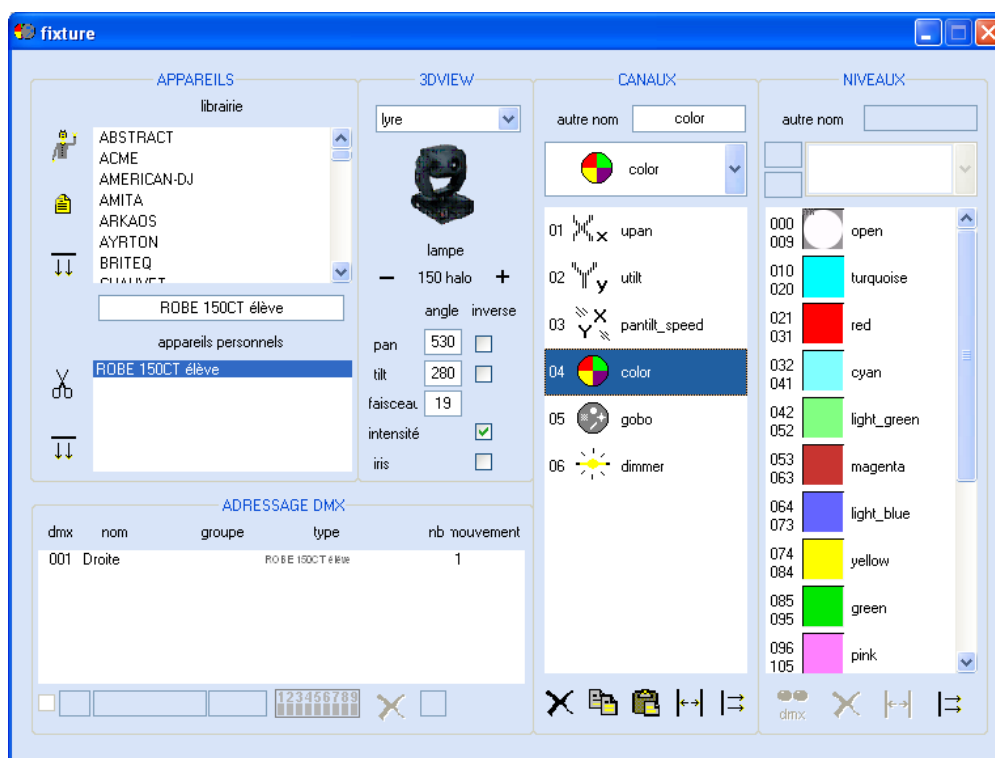
Il faut utilisé le logiciel SWEETLIGHT tout d'abord en allant sur 'control board' qui est le panneau principal du logiciel, qui permet de sélectionner toutes les autres fonctions.



Tout d'abord il faut chargé le fichier du Robe 150CT en utilisant la fenêtre 'Restore' :



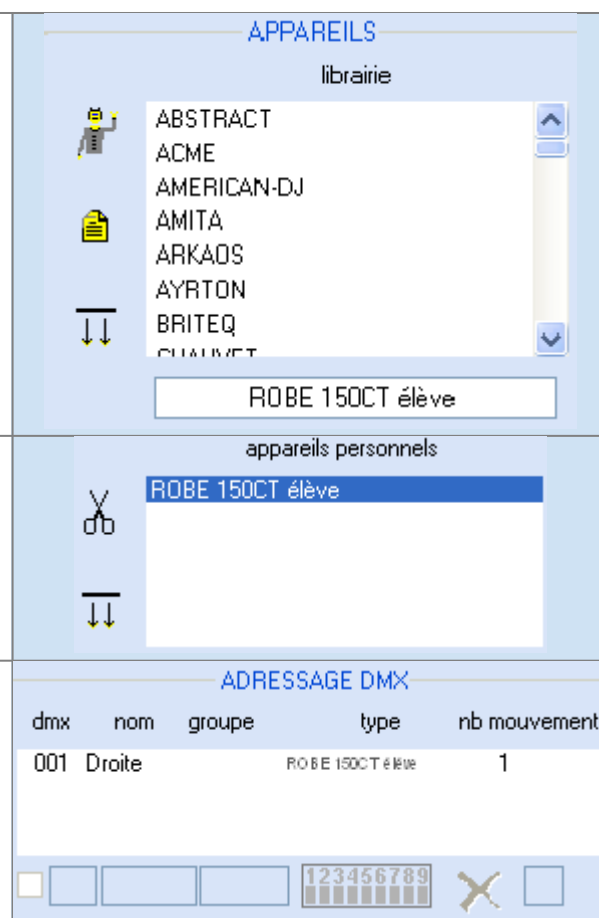
L'adresse DMX donnée par l'enseignant est A001, nous paramètrerons donc la lyre en conséquent. Pour cela nous utilisons la fonction **"Fixture"**. Celle-ci permet de choisir le(s) appareil(s) que l'on souhaite utiliser parmi une liste non exhaustive. Enfin nous attribuerons une adresse et choisirons les fonctions qui seront utilisées par la (ou les) lyre(s)..



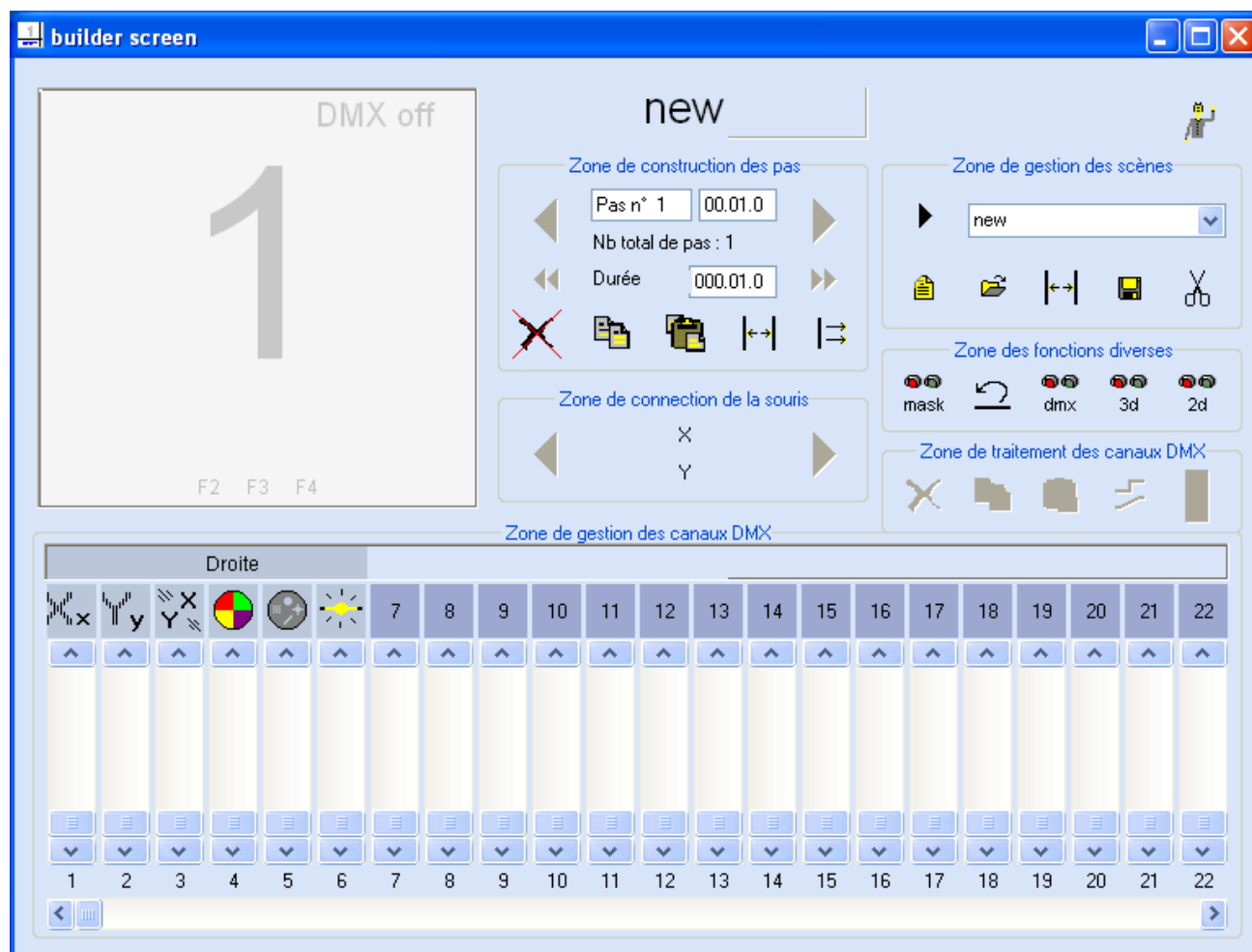
Liste de librairies permettant de sélectionner le ou les appareils que nous utiliserons dans notre spectacle.




Les appareils précédemment sélectionnés seront affichés dans cette liste. Celle-ci offre un aperçu global des différentes lyres qui pilotées afin de sélectionner celles que l'on veut paramétrer

Permet de choisir les adresse sur le bus DMX de chaque appareil



Ensuite on peut piloter la lyre précédemment régler avec la fonction **"Builder"**.
Les canaux précédemment régler dans Fixture sont affichés ici.



	Doit être activée pour piloter un appareil connecté a SWEETLIGHT.
	Activée cette fonction permet la simulation des appareils en 3D.
	Permet la simulation 2D.

Fonctionnement de deux lyre sous SWEET LIGHT

Avec deux lyres connectées en même temps le fonctionnement est pareil, il faut cependant faire attention sur certains points.

Dans l'écran FIXTURE partie ADRESSAGE DMX si on souhaite que les deux lyres fassent les mêmes choses en même temps il faut leur attribuer la même adresse. Sinon suivant le nombre de fonction il faut leurs attribuer une adresse différente.

Exemple: On sait que chaque fonction prend un canal or là il y a 6 fonctions si on veut contrôler les deux lyres séparément, il faut que les valeurs de leurs adresses est au moins une différence de 6.

The screenshot displays the SWEET LIGHT software interface with four main panels: APPAREILS, 3DVIEW, CANAUX, and NIVEAUX.

APPAREILS: A list of fixtures is shown, including ABSTRACT, ACME, AMERICAN-DJ, AMITA, ARKAOS, AYRTON, BRITEQ, and CHAUVET. The 'ROBE 150CT élève' fixture is selected and highlighted in blue.

3DVIEW: A 3D model of the selected fixture is shown. The 'lampe' (lamp) is set to '150 halo'. The 'angle' is set to 'inverse'. The 'pan' is set to '530' and the 'tilt' is set to '280'. The 'faisceau' (beam) is set to '19'. The 'intensité' (intensity) is checked, and the 'iris' is unchecked.

CANAUX: A list of functions is shown, including '01 upan', '02 utilt', '03 pantilt_speed', '04 color', '05 gobo', and '06 dimmer'. Each function is represented by a small icon and a label.

NIVEAUX: A list of levels is shown, including 'autre nom' and 'autre nom'.

ADRESSAGE DMX: A table showing the DMX addressing for the selected fixture. The table has columns for 'dmx', 'nom', 'groupe', 'type', and 'nb mouvement'.

dmx	nom	groupe	type	nb mouvement
001	Droite		ROBE 150CT élève	1
009	gauche		ROBE 150CT élève	1

At the bottom of the interface, there is a row of buttons for controlling the fixtures, including a 'dmx' button and several directional arrows.

Problème Technique n°2

On cherche à vérifier par l'expérimentation que la trame de la liaison filaire série DMX porte les informations de pilotage des projecteurs pour acquérir les connaissances et les compétences de mise en œuvre de plusieurs projecteurs dans un contexte de spectacle.

Généralités :

Le DMX (**d**igital **m**ultiplexing) est un protocole de communication série entre plusieurs appareils. Il est utilisé dans la liaison des appareils utilisés dans les spectacles (éclairage de scène, machine à fumée, etc...). Ce protocole peut gérer jusqu'à 512 canaux (*channels*) par trame. La norme prévoit l'utilisation de 16 canaux par appareil. L'intérêt de ce système est de permettre le branchement de plusieurs appareils sur le même câble, ainsi, n'avoir qu'un fil entre la commande et tous les appareils. Dans notre situation l'interface USB entre le PC et les lyres nous limitera à 100 canaux.

commande :

il existe différents moyens de générer un signal DMX, afin de commander les périphériques DMX :



via une **Table DMX**



via un **PC** équipé du logiciel "**SweetLight**" et d'un convertisseur USB↔DMX "**SweetCable**"

Connectivité :

La connectique du DMX 512 repose sur plusieurs standards de prise à 5 fils :

NC5FX & NC5MX

XLR5



Mais il existe aussi des prises non-conformes au standard à 3 fils comme les **XLR3** :



Cette connectique est utilisée par la lyre.

Caractéristique Électrique :



#	Nom	Vmin (V)	Vmax (V)
1	Masse	0	0
2	Data +	0,875	5,00
3	Data -	0,125	4,00
	(Data+) - (Data-)	-3.125	4.875

Nous remarquons que les signaux **Data -** et **Data +** sont inversés avec un offset différent. L'intérêt de ce système réside dans son immunité face aux parasites qui peuvent interférer avec le signal. Si un parasite survient il affectera les 2 fils Data+/- . La différence (Data+) - (Data-) donne donc un signal non affecté par ces parasites. Ces perturbations peuvent être due à l'allumage d'un moteur ou d'un en-fumeur ou autre machine de spectacle.

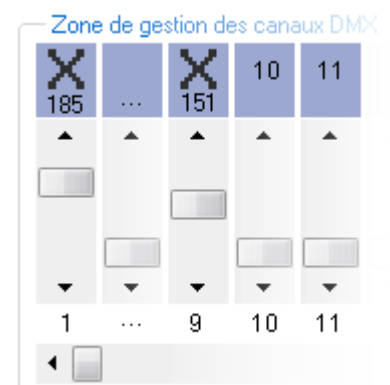
Protocole expérimental :

Nous allons relever le chronogramme d'un cycle sur le Data+ car Data+ est l'image de la tension différentiel **Data+ - Data-** du bus DMX. Pour cela nous utilisons l'oscilloscope en mode run/stop puis nous analyserons en détail les différentes parties et sous-parties d'un cycle entre le PC et 2 lyres.

Les 2 lyres sont pilotées par le logiciel sweetlight.

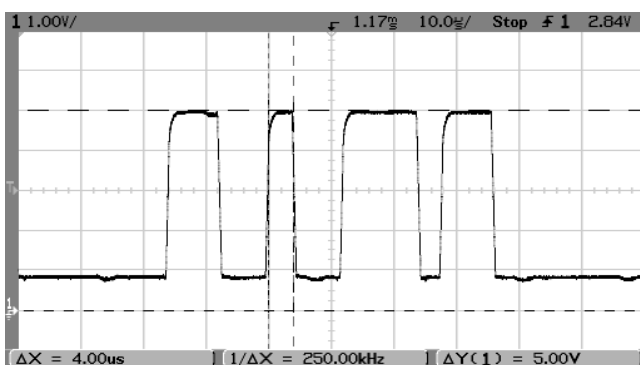
Elles sont configurées pour que :

- La première récupère la valeur de Xpan sur le channel 1
- La deuxième récupère la valeur de Xpan sur le channel 9



le builder est configuré comme ci dessus

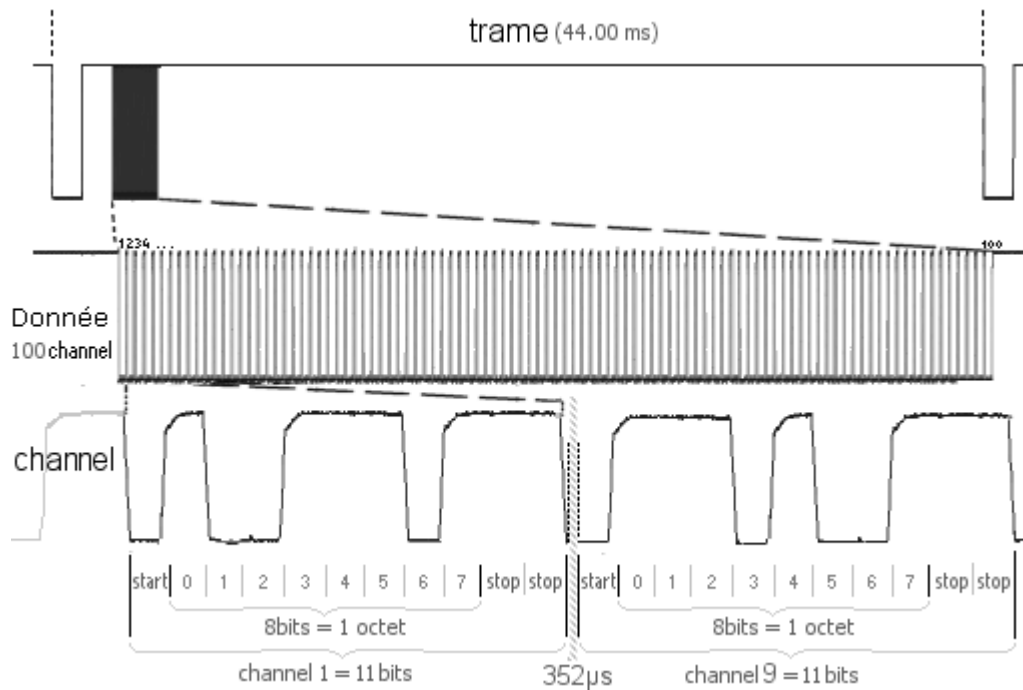
Chronogramme:



On remarque que le signal est de caractéristique :

- Vmin = 0.875 V
- Vmax = 5.00 V
- Période d'un bit : 4μs

Relevé :



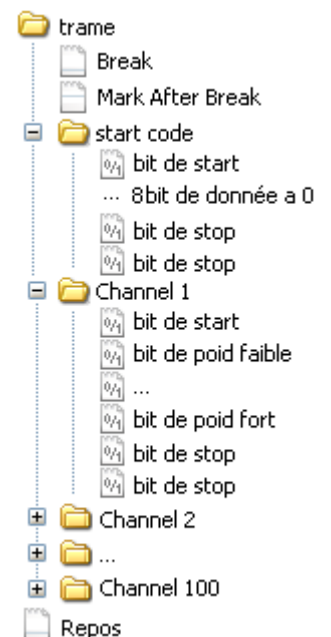
Dans ce chronogramme nous pouvons voir un cycle complet de 44.00 ms contenant une série de 100 canaux. (Nous n'utiliserons pas 512 canaux car l'adaptateur USB-DMX nous limite à seulement 100 canaux.)

Analyse :

Chaque canal est composé de 11 bits dont 8 bits de donnée.
Chaque bit a une durée de $4\mu s$ (voir chronogramme précédent).
Donc chaque canal occupe $11 \times 8\mu = 44\mu s$ de trame.
on en déduit le Baud Rate : $1/4\mu = 250kb/s$ ($31.25ko/s$)
et le Data Rate : $(8/11) \times 250k = 182kb/s$ ($22.72Ko/s$)

Dans notre chronogramme relevons les valeurs des 2 channels.
- La valeur binaire du channel 1 est 10111001 soit 185 en décimal.
- La valeur binaire du channel 2 est 10010111 soit 151 en décimal.
Ces valeurs correspondent aux signaux envoyés par sweetlight.

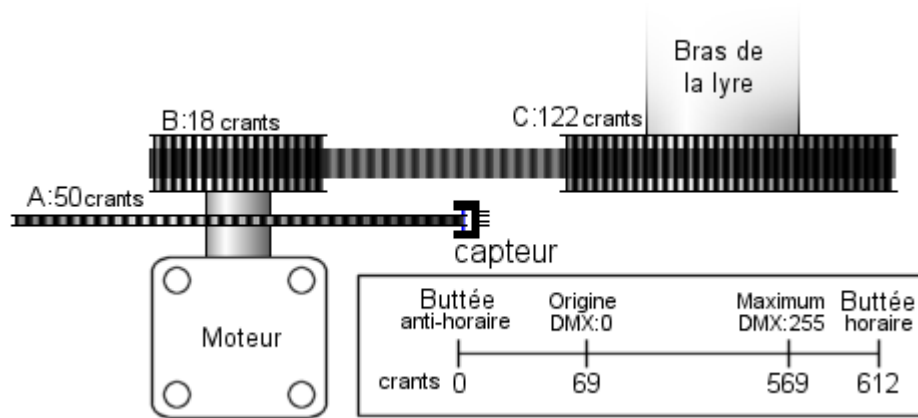
D'après nos analyses une trame est structurée de façon suivante \Rightarrow



Relevé des positions :

Pour effectuer les mesures nous mettons la lyre hors tensions. Nous enlevons le capot afin de voir les différentes roues. Puis nous faisons tourner manuellement le bras de la lyre et comptons les cran de la roue crantée.

schéma des roues :



Le rapport de rotation entre le moteur et le bras de la lyre est de $\frac{Z_b}{Z_a}$ soit $\frac{18}{122} = 0.147 = 14.7\%$

Un cran représente: $\frac{360}{50} = 7.2^\circ$ sur la roue **A** soit $7.2 \times 0.147 = 1,06^\circ$ sur le bras de la lyre.

Lorsque le bus DMX envoie la valeur 0 :

- la position de A est à 1 tour + 19 crans
- soit 69 crans.
- soit 497° sur la roue crantée A
- soit 73° sur le bras de la lyre

Lorsque le bus DMX envoie la valeur 255 :

- la position de A est à 12 tours + 19 crans.
- soit 569 crans.
- soit 4097° sur la roue crantée A
- soit 602° sur le bras de la lyre

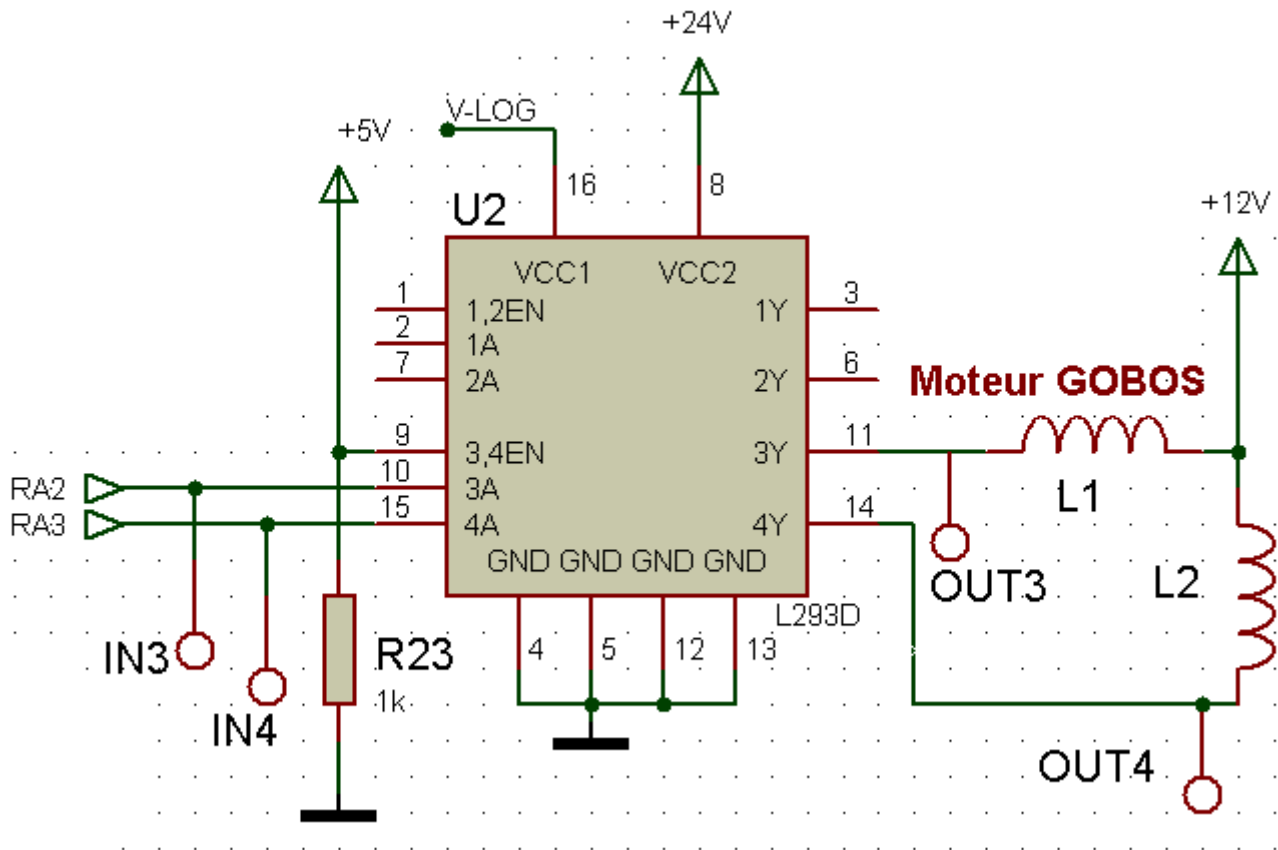
L'interface de la lyre offre donc une amplitude de **$569 - 69 = 500$ crans** ce qui représente exactement 10 tours de moteur ou $10 \times 0.147 = 1.47$ tours du bras de la lyre.

PROJET N°2

Problème technique 1:

On cherche à décomposer l'organisation du cycle des signaux de commande du moteur pas à pas GOBOS pour concevoir ce cycle, puis le transposer dans le sous-programme de commande du moteur PAN.

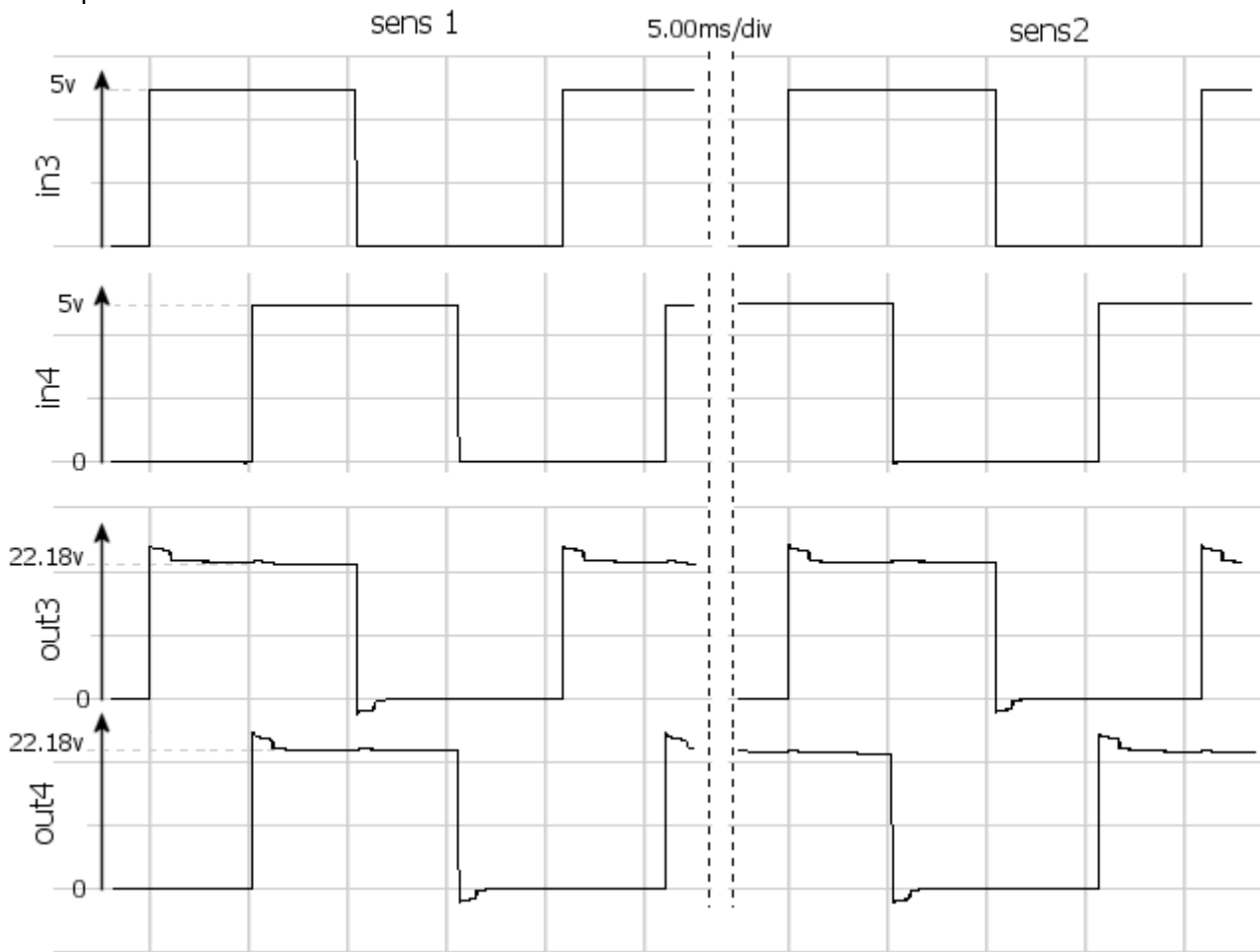
Schéma structurel reliant le programme GOBO et le moteur pas à pas



Le moteur pas à pas utilisé est le T8701-02 dont les spécifications générale sont :

Constructeur : Minebea Motor Manufacturing Corp
Numéro composant : T8701-02
Série : 17PM-K
Bobine de construction : unipolaire
Angle de pas : $1,8^\circ \pm 0,09^\circ$
Étape par tour : 200

Nous allons capturer les signaux IN₃ IN₄ OUT₃ et OUT₄ en correspondance de temps en utilisant l'oscilloscope



Le type d'alimentation des bobines du moteur est bipolaire car :

PAS₁ > lorsque in₃=5V et in₄=0V on a : 12V au borne de L1 et -12V au borne de L2

PAS₂ > lorsque in₃=5V et in₄=5V on a : 12V au borne de L1 et 12V au borne de L2

PAS₃ > lorsque in₃=0V et in₄=5V on a : -12V au borne de L1 et -12V au borne de L2

PAS₄ > lorsque in₃=0V et in₄=0V on a : -12V au borne de L1 et 12V au borne de L2

Les tensions aux bornes des bobines sont donc de $\pm 12V$ il n'est donc jamais a la masse
ce n'est donc pas une alimentation unipolaire, mais bipolaire.

On en déduit que le rôle du circuit L293D est de faire une adaptation en puissance des signaux d'entrée in₃ et in₄ d'amplitude 5V vers les signaux de sortie out₃ et out₄ d'amplitude 22.2V afin d'alimenter le moteur GOBO avec une puissance suffisante.

Calcul de la vitesse du moteur:

Le temps que met le moteur à faire 1 pas est de $\frac{t}{4}$ soit $\frac{20.90ms}{4} = 5.22ms$.

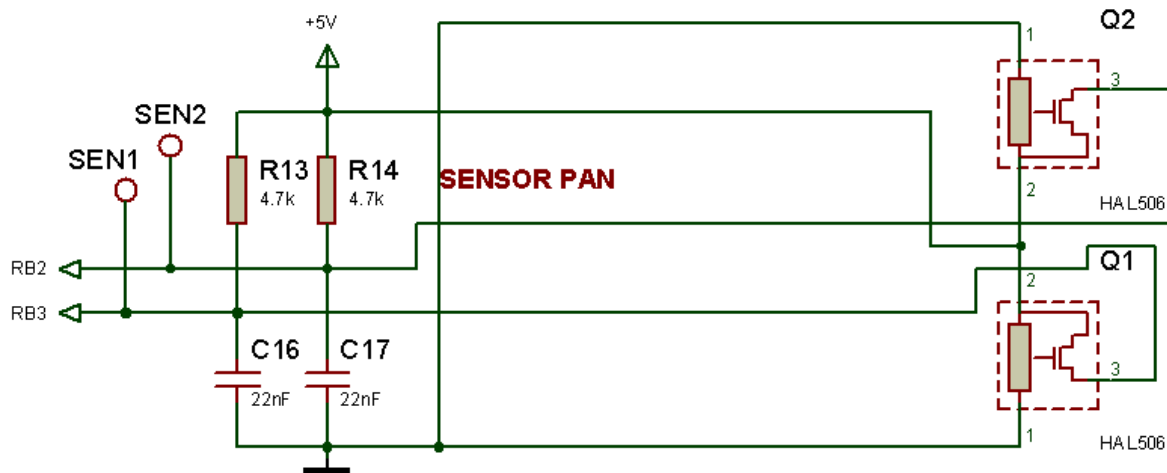
On sait que le moteur fait 200 pas pour accomplir 1 tour.

On en déduit donc la vitesse du moteur: $5.22 \cdot 10^{-3} \times 200 \times 60 = 62.64 \text{ tr/mn}$.

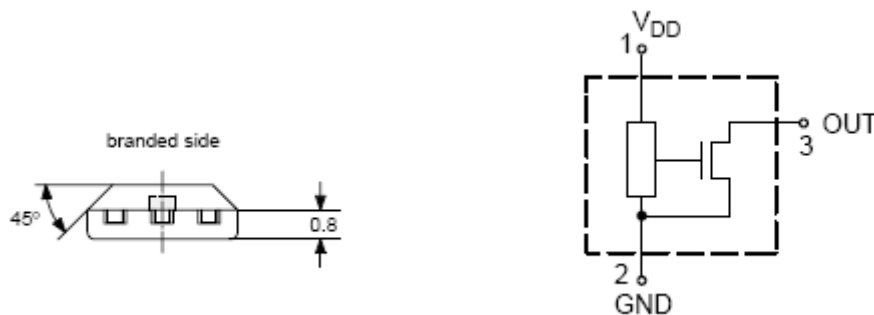
Problème technique 2

On cherche à mettre en œuvre deux capteurs à effet hall pour informer le sous-programme "ROTATION AXE PANORAMIQUE" de la position " RÉFÉRENCE" position qui est la butée anti horaire de l'axe panoramique.

Schéma structurel entre le microcontrôleur et les deux capteurs SENSOR PAN



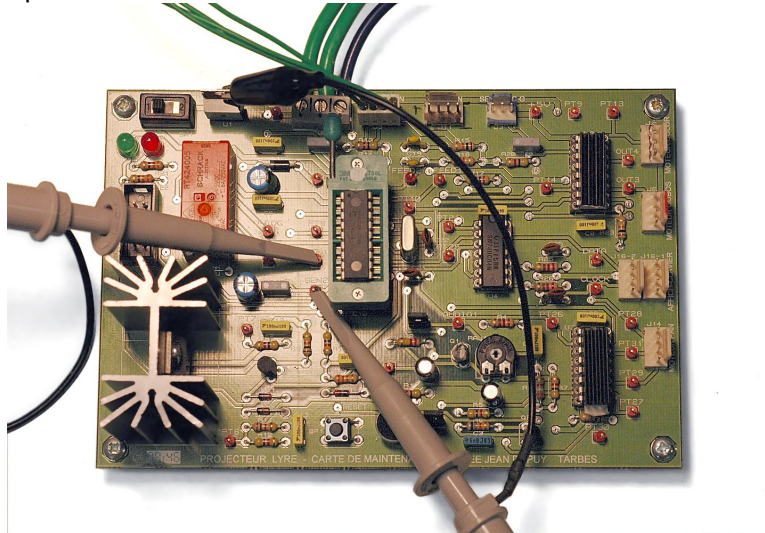
Principe de fonctionnement et illustration des capteur à effet hall



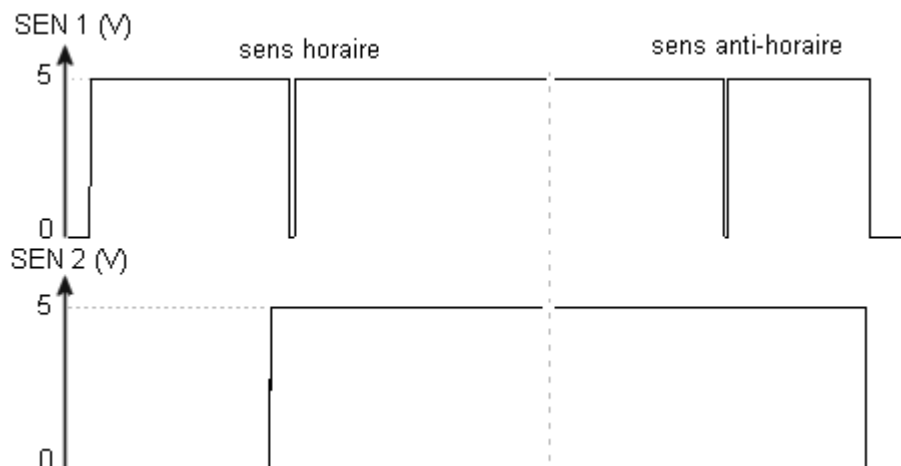
La sortie passe à l'état bas avec un champ magnétique de pôle sud sur le coté nommé "branded side". Lorsque il n'y a pas de champ magnétique la sortie est à l'état haut. Ce capteur ne réagit pas avec des champs magnétique du pôle nord.

Expérimentation permettant de mesurer sen1 et sen2

Les mesures se font à l'oscilloscope, des grips fils sont placés sur sen1 et sen 2 qui sont respectivement visionnés sur la voie1 et 2 de l'oscilloscope.



A ce moment-là il suffit de tourner l'axe pan manuellement dans le sens horaire puis anti-horaire pour en tirer 2 chronogrammes.



Remarque:

- sens horaire: la lyre, vue de haut, tourne dans le sens des l'aiguilles d'une montre
- sens anti-horaire: la lyre, vue de haut, tourne dans le sens inverse des l'aiguilles d'une montre

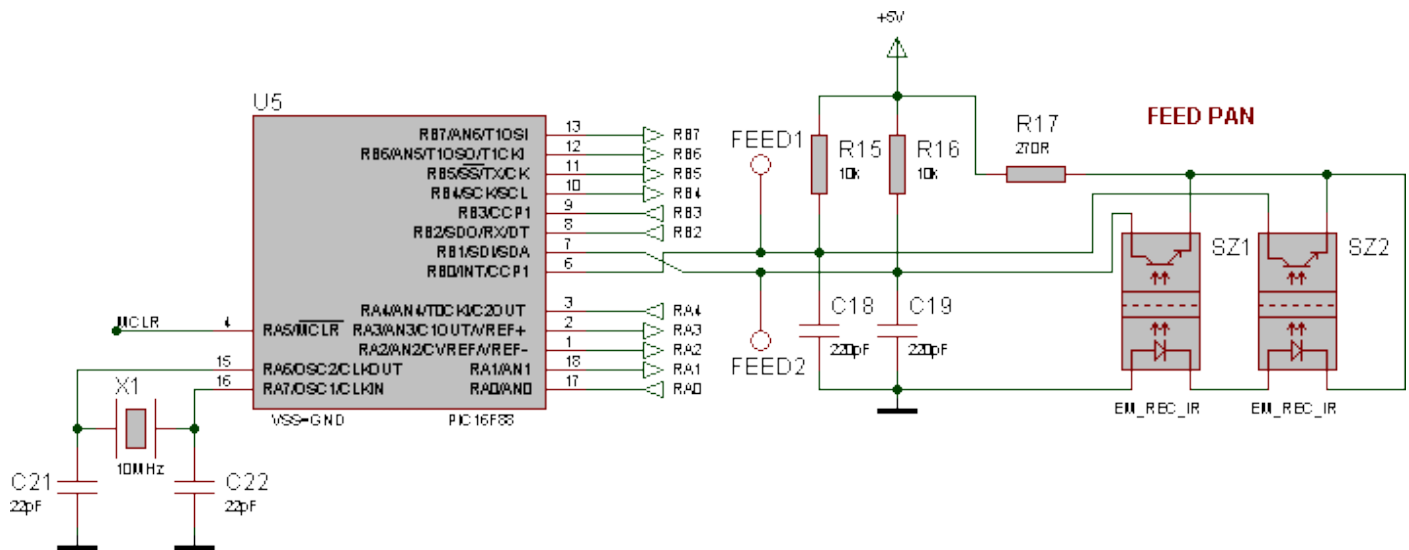
Synthèse

- La languette métallique est muni d'un aimant de pôle sud permettant de créer un champ magnétique qui sera détecté par les deux capteurs à effet hall. Car la languette est placé sous l'engrenage de l'axe pan, le champ magnétique est crée lors que la languette passe a côté de l'aimant fixe lors de la rotation de l'engrenage.
- Il y a deux capteurs pour savoir dans quel sens se trouve la lyre. Car un seul capteur ne suffit pas à connaitre le sens.

Problème technique 3

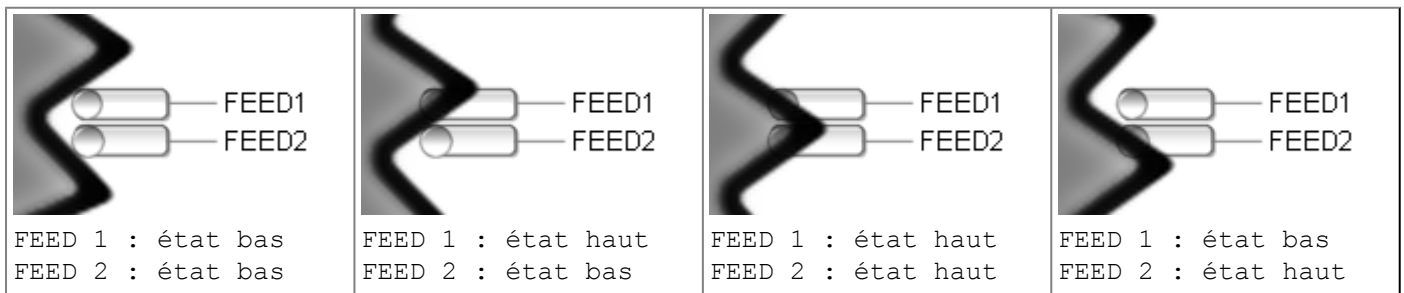
Schéma structurel partiel

faisant le lien entre le microcontrôleur et les deux capteurs FEED PAN repérés SZ1 et SZ2 avec les points de prélèvement FEED1, FEED2.

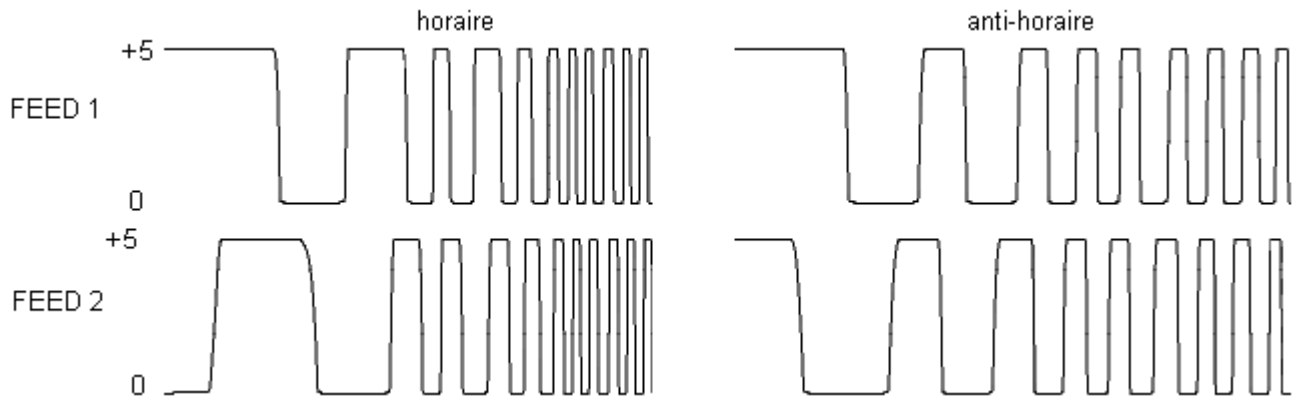


Les optocoupleurs étant composé de diodes, et de transistor, les résistances R15 R16 et R17 permettent de limiter la tention a leur bornes. Les deux capteurs infrarouges combinés au disque à créneaux permettent de mesurer la position angulaire de la lyre grâce au nombre de crant qui auront causé un front descendant successivement sur SZ1 puis SZ2 (sens horaire), ou inversement (front descendant sur SZ2 puis SZ1) si la lyre tourne dans le sens anti horaire). cette mesure permet a la lyre de savoir si le nombre de tours que doit ou a due effectuer la lyre c'est bien dérouler, ou si quelqu'un la manuellement fait pivoter alors qu'elle ne le devais pas, et ainsi corriger l'écart constater.

Illustration des deux capteurs infrarouges :



Signaux FEED 1 et FEED 2 en concordance de temps grâce à l'oscilloscope en mode "roll" dans les 2 sens de rotation et en accélérant graduellement :



Les signaux FEED₁ et FEED₂ renseignent le sous-programme ROTATION AXE PANORAMIQUE sur :

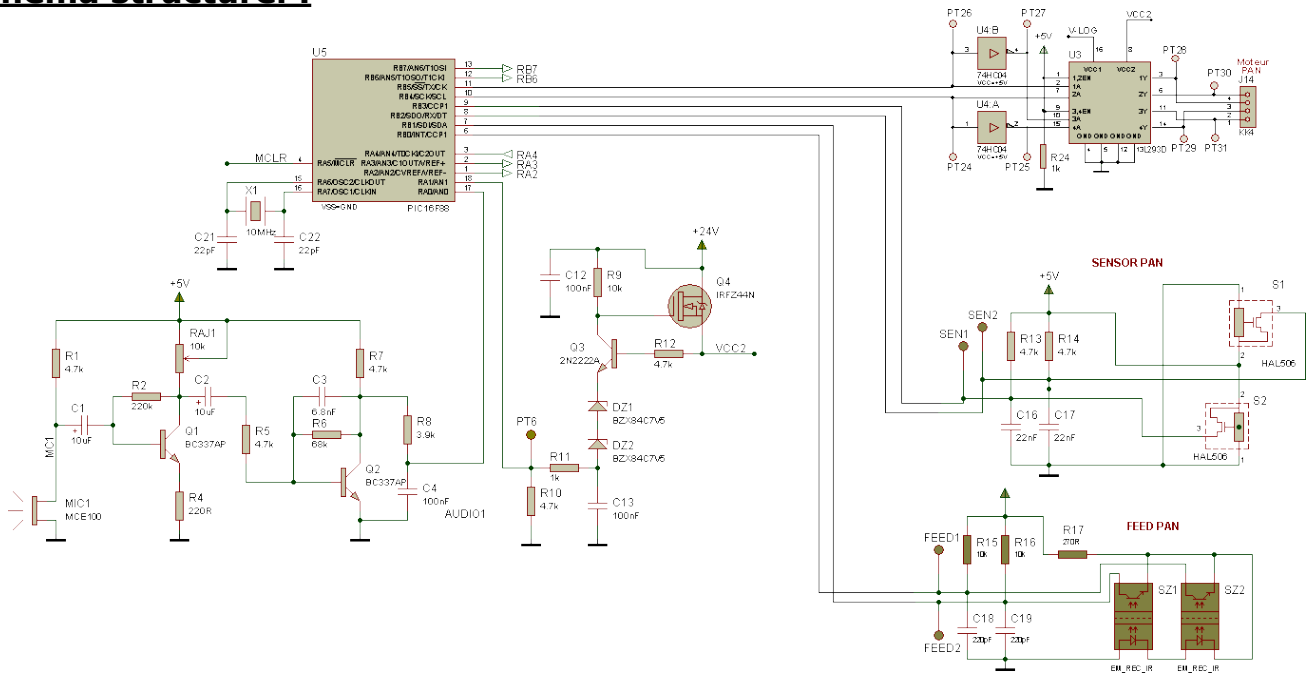
- La **fréquence** des impulsions permet de déterminer la **vitesse de rotation** de l'axe :
une augmentation de la fréquence signifie une augmentation de la vitesse de rotation.
une diminution de la fréquence signifie une diminution de la vitesse de rotation.
- **l'ordre des fronts** renseigne sur **le sens** de rotation de l'axe
si FEED2 est en avance sur FEED1 l'axe tourne en sens anti-horaire
si FEED1 est en avance sur FEED2 l'axe tourne en sens horaire
- grâce à cette association on a les 2 paramètres nécessaire au calcul de la position :
Le capteur à effet hall renseigne si la lyre a déjà effectué un tour complet
FEED1-2 renseigne sur le nombre de crans dépassés depuis que la lyre s'est initialisé ou depuis que l'aimant est passé devant le capteur effet hall.

Problème technique 4:

On cherche à concevoir un programme de "rotation axe panoramique" qui répond aux exigences suivantes :

- mise en position *RÉFÉRENCE* de l'axe panoramique
- mise en position *ORIGINE* de l'axe panoramique (valeur 0 pour le canal PAN sur le bus DMX)
- réaliser une séquence de mouvements libres :
 - pour une durée d'une minute environ
 - avec des vitesses de rotation différentes
 - sur la plage angulaire correspondant à la plage 0/255 pour le canal PAN sur le bus DMX.

Schéma structurel :



Passage de la commande du moteur PAN au moteur GOBOS:

Pour passer les commandes du moteur GOBOS au moteur PANORAMIQUE

Il nous faut changer les affectations des bit du port A et B sur la carte de maintenance :

- RB5 (PAN1) doit être brancher sur RA2 (GOBO1).
- RB4 (PAN2) doit être brancher sur RA3 (GOBO2).

On effectue donc les changements appropriés dans le flowcode.

Liste des ports entrées sorties et leur affectations :

A0	entrée Audio
A1	sortie ELEV_TEN
A2	sortie GOBO 1
A3	sortie GOBO 2
A4	entrée SENSOR (GOBO)

B0	entrée FEED 1
B1	entrée FEED 2
B2	entrée SEN 1
B3	entrée SEN 2
B4	sortie PAN 2
B5	sortie PAN 1
B6	sortie COLOR 1
B7	sortie COLOR

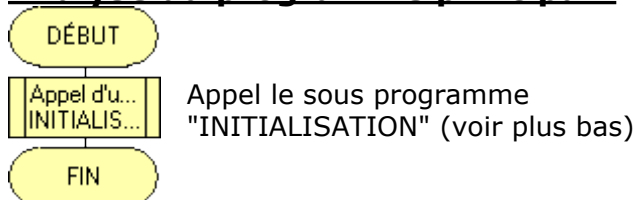
Rôle des variables du programme GOBO1 :

ROT_EN	Permet a la macro TMR0 de savoir si elle est autoriser a faire tourner le moteur. 0:non 1:oui
SENS	Définie le sens de rotation du moteur contrôlé par TMR0. (0=Anti-horaire;1=horaire)
DUREEPAS	Définie la durée d'un pas. es valeurs sont comprisent entre 30 et 70.
TEMP	Compteur incrémenté a chaque interruption jusqu'à atteinte DUREEPAS puis remis a 0.
PAS	Compteur incrémenté a chaque pas (0~3) permet de connaître les états a appliquer sur le moteur.
POSREF	Valeur du port A4 qui est le capteur SENSOR. Prend pour valeur 0 si la lyre est en butée.

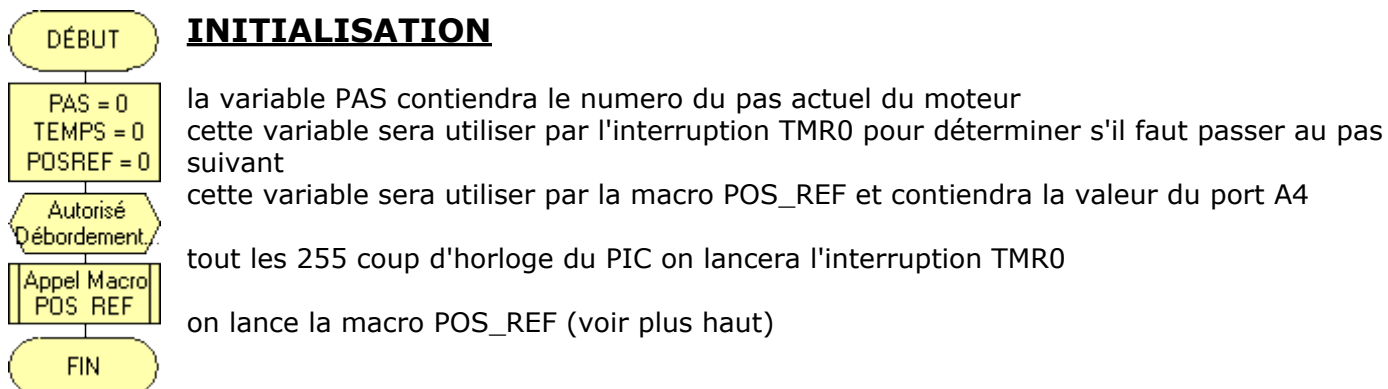
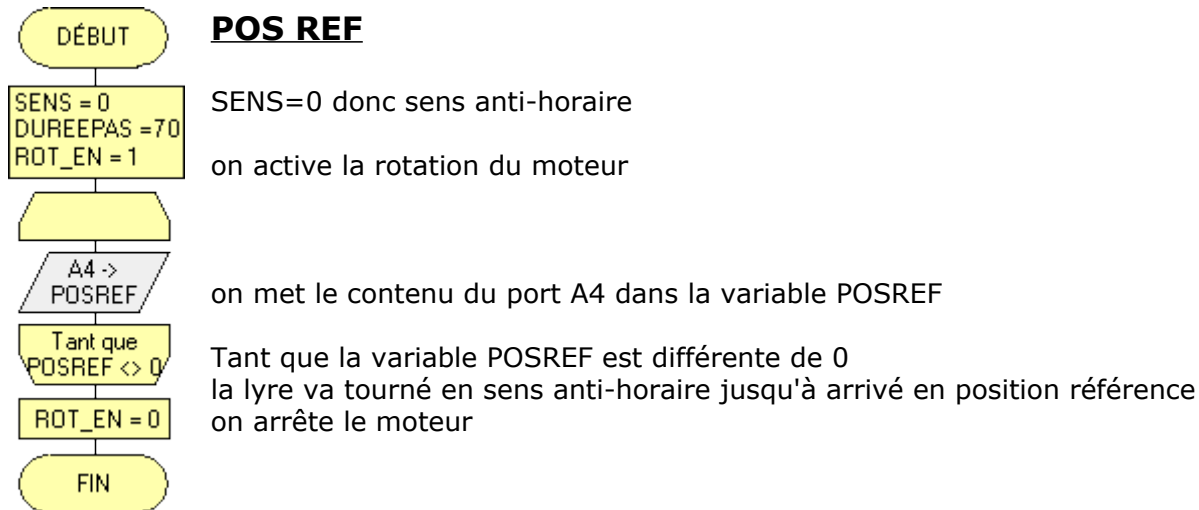
Rôle des variables du programme ROTPAN1:

RB_OUT	Valeur du port RB, utilisé lors des calculs relatif a la generation des pas
SENS	Définie le sens de rotation du moteur contrôlé par INT_MOT. (1=horaire 0=Immobile -1=Anti-horaire)
DUREE	Définie la durée d'un pas.
COUNT	Compteur incrémenté à chaque interruption jusqu'à atteinte DUREEPAS puis remis a 0.
PAS	Compteur incrémenté à chaque pas (0~3) permet de connaître les états à appliquer sur le moteur.
POSREF	Valeur du port A4 qui est le capteur SENSOR. Prend pour valeur 0 si la lyre est en butée.

Analyse du programme principal :



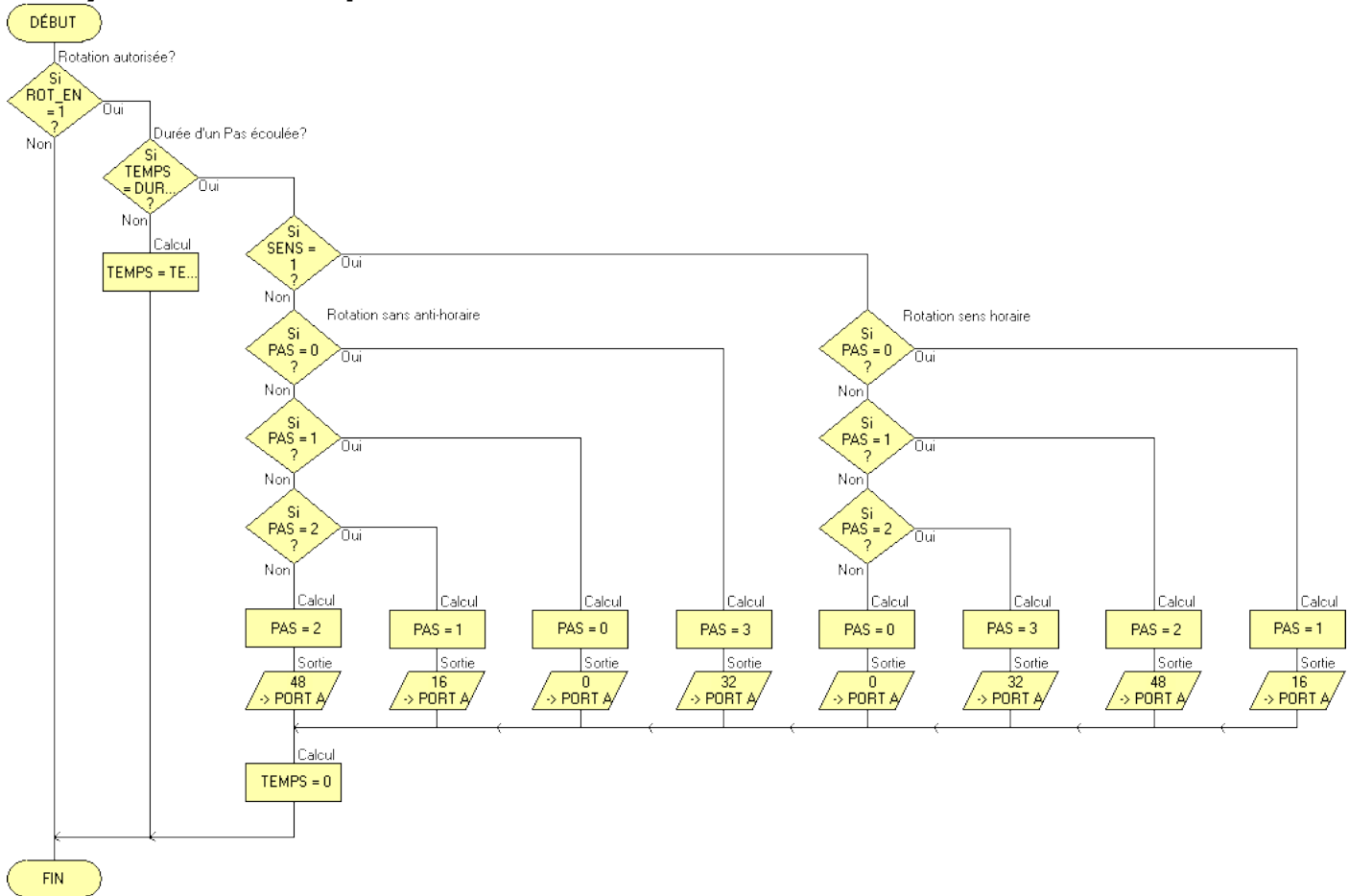
Analyse des sous programme :



Rôle des interruptions:

Les interruptions sont des sous programmes qui s'activent lorsque certaines actions surviennent. Par exemple lors d'un fronts montant sur une entrée ou lorsque le microprocesseur atteint son 256 eme coup d'horloge.

Analyse de l'interruption :TMR0



- Cette interruption génère les signaux qui commande le moteur pas à pas.
 - L'initialisation de cette interruption est faite dans le programme "initialisation"
- Son fonctionnement est le suivant :

- on verifie que ROT_EN autorise la rotation des moteurs
- on compare si la valeur definie par DUREEPAS correspond au compteur TEMP
 - Si ce n'est pas le cas on incremente TEMP
 - Si c'est le cas on passe au pas suivant : on verifie donc le SENS de rotation
 - en sens anti horaire :
 - PAS est décrémenté (et est ramener à 3 si inferieur a 0)
 - on applique en sortie, l'état correspondant au PAS
 - en sens horaire :
 - PAS est incrémenté (et est ramener à 0 si superieur à 3)
 - on applique en sortie, l'état correspondant au PAS

la frequence d'horloge est donnée par la relation : $F_{clock} = \frac{F_{quartz}}{4} = \frac{10 \times 10^6}{4} = 2.5 MHz$

l'interruption étant lancer tout les 256 coups d'horloge, on a : $F_{interrupt} = \frac{F_{clock}}{256} = 9765,625 \text{ Hz}$

et au final le pas est changé 1 fois toute les DUREEPAS : $\frac{F_{interrupt}}{DUREEPAS} = \frac{9765,625}{DUREEPAS}$

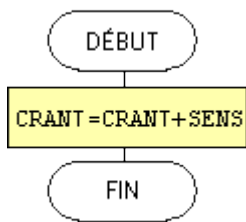
la période d'un pas en fonction de la variable DUREEPAS est donc : $\frac{DUREEPAS}{F_{interrupt}} = \frac{DUREEPAS}{9765,625}$

Énoncé du problème ROTPAN3:

On cherche à concevoir un programme qui :

- ramène l'axe panoramique du projecteur en position de REFERENCE
- positionne l'axe panoramique du projecteur sur ORIGINE (valeur 0 sur le bus DMX)
- répond à une séquence de mouvement libre d'une minute environ, et qui tienne compte :
 - des limites de la position angulaire :
 - position 0 sur le bus DMX
 - position 255 sur le bus DMX
 - des limites de la vitesse angulaire :
 - vitesse minimale pour la variable DUREEPAS=70
 - vitesse maximale pour la variable DUREEPAS=30
- une temporisation de l'ordre de 1sec sera insérée entre chaque ensemble de mouvements

Présentation des interruptions :

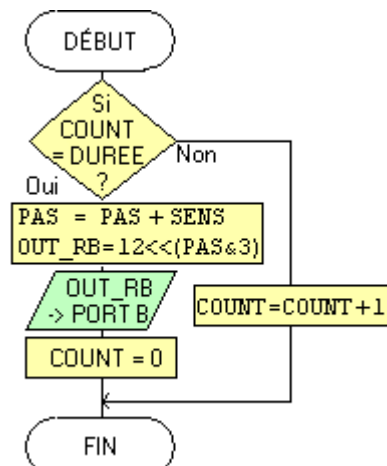


RB0/INT

cette interruption est lancée sur un front montant de RB4
donc lorsqu'une dent passe sur le capteur, on modifie la variable "CRANT"

- si le sens est horaire (SENS=1) : CRANT est incrémenté
- si le sens est anti-horaire (SENS=-1) : CRANT est décrémenté

SENS peut être négatif car CRANT et SENS sont des variables de type "entier"
donc coder sur 16bits ayant une plage de valeur de -32767 à 32768.



INT MOT (ancien TMR0)

on compare si COUNT = DUREE ?

si ce n'est pas le cas (non) :

on incrémente le compteur COUNT de 1

si c'est la cas (oui) :

> on ajoute la valeur de SENS à PAS donc :

- * si SENS vaut 1 PAS est incrémenté
- * si SENS vaut -1 PAS est décrémenté

> puis on calcul $12 \ll (PAS \& 3)$ ce calcul se décompose en 3 parties :

PAS & 3 : PAS est une valeur codée sur 8bits donc variant de 0 à 255.

l'opérateur "&" couple la valeur PAS à 3, sachant que $(3)_{10} = (11)_2$.

PAS & 3 nous donne donc les 2 derniers bits de PAS, soit un nombre entre 0 et 3.

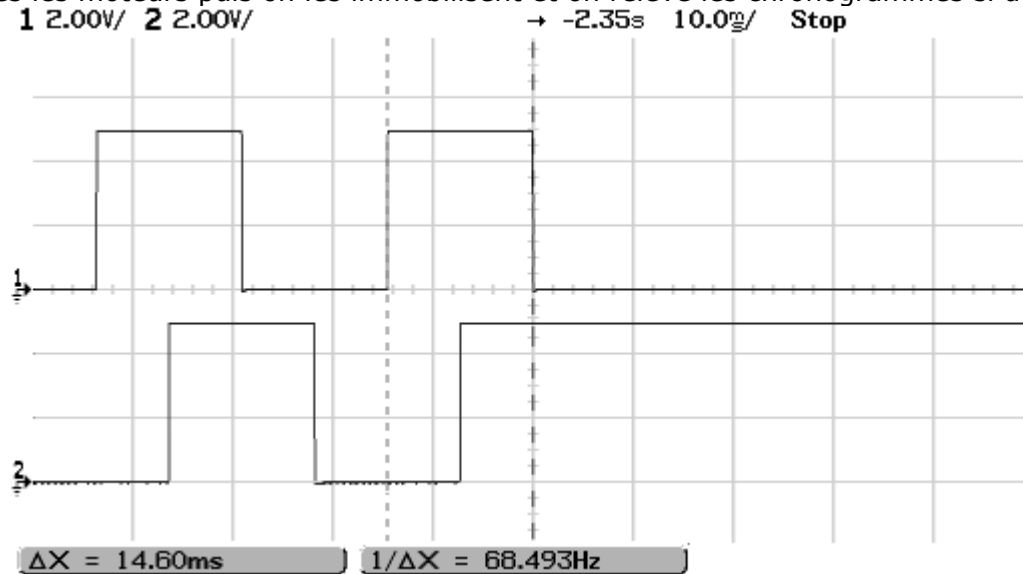
$12 \ll (n)$: on effectue un décalage de n bits vers la gauche sur le nombre 12 sachant que $(12)_{10} = (1100)_2$ et que n est le résultat de PAS & 3 (donc un entier compris entre 0 et 3)

OUT_RB n'est pas directement envoyé sur le port B. Seul les bits 4 et 5 seront envoyés, cela permet de ne pas envoyer Nous n'envoyons que les bits 4 et 5 sur le port B

PAS	calcul	sortie RB
		76543210
0	$(1100)_2 \ll 0$	00001100
1	$(1100)_2 \ll 1$	00011000
2	$(1100)_2 \ll 2$	00110000
3	$(1100)_2 \ll 3$	01100000

Vérification :

Nous vérifions grâce à un oscilloscope, si les moteurs sont bien commandés par la fonction INT_MOT. On branche la voie 1 de l'oscilloscope sur le port RB4 (PAN2), et la voie 2 sur le port RB5 (PAN1) on fait tourner les moteurs puis on les immobilise et on relève les chronogrammes ci-dessous :



on remarque que les formes des différents pas sont bien générés. Nous allons cependant vérifier par calcul si la durée de ces pas est conforme à nos attentes.

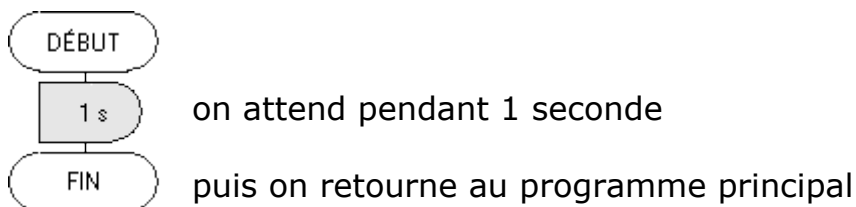
verification par calcul :

lorsqu'on met 70 dans la variable DUREEPAS on doit avoir une période de : $\frac{70}{9765,625} = 7.168ms$

dans le chronogramme la durée de 2 pas fait : 14.60ms, donc 1 pas = 7.3ms

$7.3ms \approx 7.168ms$ donc le programme remplit bien sa fonction.

Présentation du sous programme ATTEND :



Nous utilisons un sous-programme pour cette simple tâche car, plus tard, nous pourrions la modifier pour y intégrer la réaction avec le micro sans modifier le programme principal.

Présentation du programme principal :



ROTATION AXE PANORAMIQUE 3

détermine la durée d'un pas *plus elle augment plus la vitesse de rotation est lente*
 détermine le sens de rotation de la lyre (*horaire:1 anti-H:-1 immobile:0*)
 contient le numero du pas actuel *utilisé par la fonction RB0*
 compteur incrémentée par l'interruption INT_MOT
 variable contenant la valeur envoyer sur les bits de sortie B4 et B5 (*PAN1 PAN2*)
 variable contenant la valeur des bits B3 et B2 (*détecte la position butée*)

On active l'appelle de la fonction "INT_MOT"
 A chaque debordement du timer (>255)

on met la valeur des capteurs PAN (B3 + B2)
 dans la variable "BUTEE"

tant que la lyre n'est pas en position référence (BUTEE est differente de 0)
//on est maintenant en position REFERENCE (objectif de ROT_PAN1)

on immobilise la lyre

on attend pendant 1 seconde

permet de connaître la position de la lyre par rapport a la position REFERENCE

on part dans le sens horaire (sens = 1)

on autorise l'interruption RB0/INT

qui tiendra a jours la variable CRANT

tant que la position de la lyre

est inferieur au 69em crant

ne rien faire (donc continuer de tourner)

// on est en position DMX 0 (objectif de ROT_PAN2)

on immobilise la lyre

on appelle la

macro ATTEND

puis on repart dans le sens horaire (sens = 1)

tant que la position de la lyre

est inferieur au 569 em crant (DMX:255)

on fait varier la variable DUREE en fonction de CRANT :

CRANT va augmenter donc DUREE va augmenter : la lyre tourne en décélérant

// on est en position DMX 255 (objectif de ROT_PAN3)

on immobilise la lyre

on appelle la

macro ATTEND

puis on repart en sens anti horaire (sens = -1)

tant que la position de la lyre

est superieur au 69em crant (DMX:0)

on fait varier la variable DUREE en fonction de CRANT

CRANT va diminuer donc DUREE va diminuer : la lyre tourne en accélérant

//on est revenue en positon DMX:0 (on peut repeter ces actions pendant 1min ...)

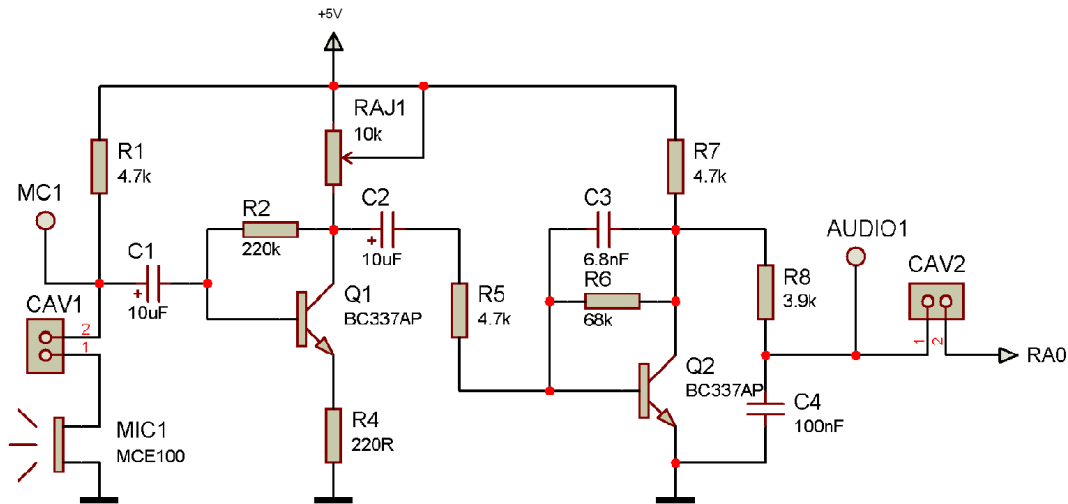
on immobilise la lyre

Problème technique 5:

Énoncé du problème ROTPAN4:

On ajoute une fonction de détection des pics sonores à notre programme ROT_PAN.
Cette fonction attend qu'un bruit suffisant survienne, avant de faire changer de sens les moteurs PAN.

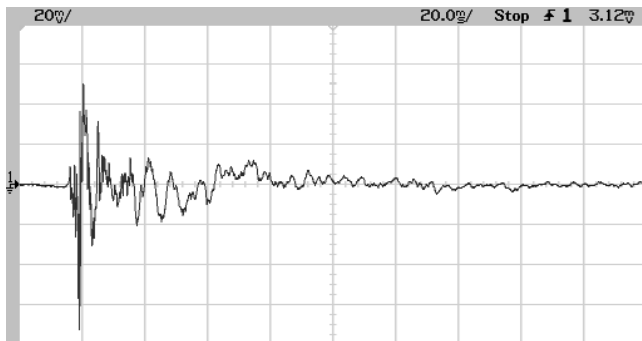
Schéma structurel partiel générant AUDIO1:



Le signal **AUDIO1** est associé à l'entrée **RA0** du PIC.

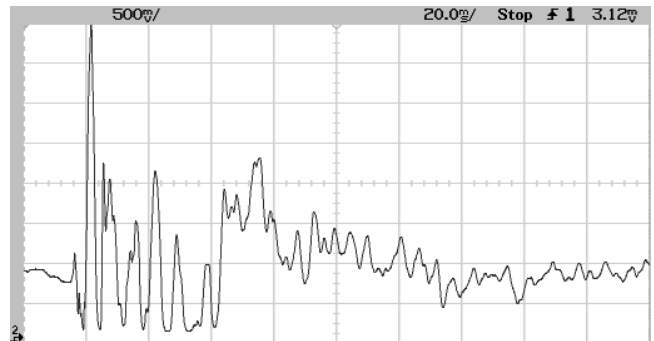
Mesure réalisées :

MC1 (micro) :



$\hat{U}_{max} : 76mV$

AUDIO1 (signal filtré) :

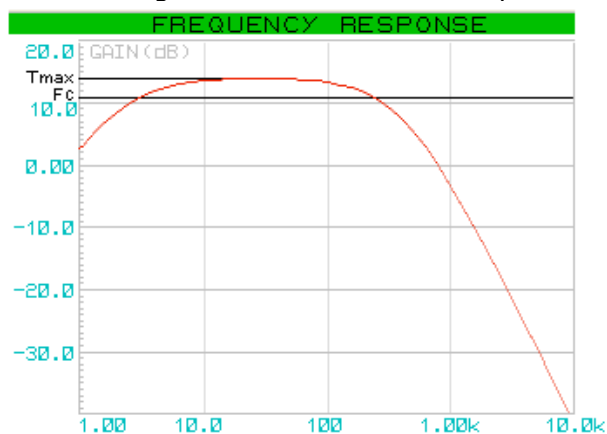


$\hat{U}_{max} = 3.8V$

composante continue = 0.9 V

Simulation:

Courbe de gain en fonction de la fréquence :



(Tmax augmente)

On remarque que le filtre est de type **actif** car le Gain en tension est supérieur à 0dB.

Ses caractéristiques sont les suivantes :

Tmax = 14dB lorsque $f = 30Hz$

la fréquence de coupure $F_c = 11dB$ à :

$f = 5Hz$ et $f = 500Hz$

Une augmentation de la valeur du potentiomètre RAJ1 entraîne une diminution de la courbe de Gain (Tmax diminue)

Une diminution de la valeur du potentiomètre RAJ1 entraîne une élévation de la courbe de Gain

Conception de ROT_PAN4:

Avant de nous lancer dans la conception du programme, nous allons déterminer les paramètres nécessaires à la bonne configuration de l'échantillonnage analogique.

Vref = 5V (d'après le schéma structurel)

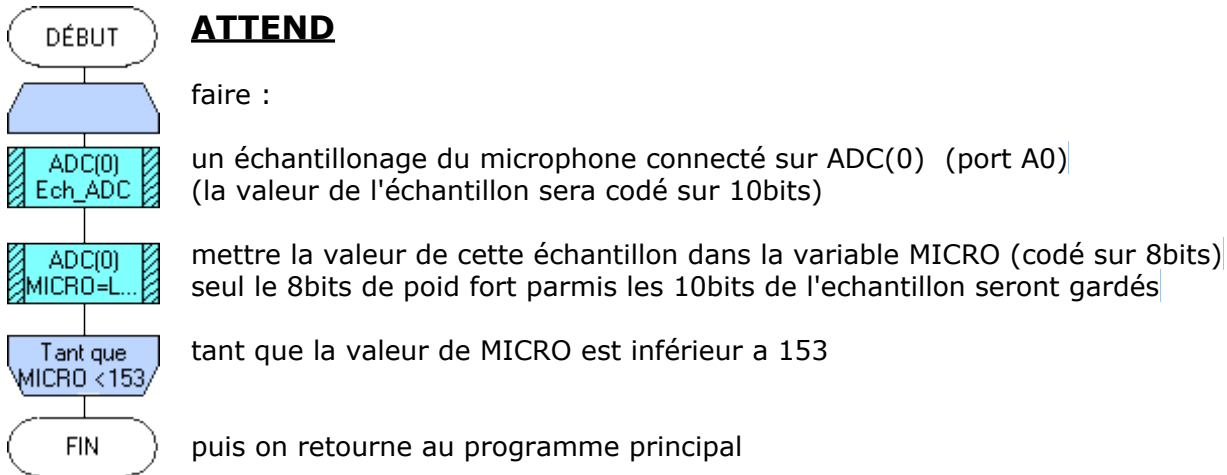
quantum (après stockage sur 8bits) $q = \frac{V_{ref}}{2^8 - 1} = \frac{5}{255} = 19,60mV$

Si nous voulons faire réagir la lyre lorsque l'entrée AUDIO1 dépasse 3V nous devons calculer :

$n = \frac{3}{0,0196} = 153$ Il faudra donc que la variable 8bits contenant l'échantillon du micro soit d'au moins

153, pour faire sortir la lyre de sa boucle d'attente

Nous pouvons maintenant commencer à modifier la macro ATTEND en utilisant les valeurs calculées :



Propriétés : Routine Composant

Nom à afficher : conv

Composant : ADC(0)

Macro : Ech_ADC
Lire_comme_Octet
Lire_comme_Entier

Paramètres :
Aucun paramètre n'est nécessaire pour appeler cette macro.

Valeur de retour :

Variables...

OK Annuler

Configuration de la routine d'échantillonnage

Nous choisissons dans la liste de gauche le composant que nous allons utiliser. Puis nous sélectionnons l'action à effectuer dans la liste de droite. Ici, nous lançons l'échantillonnage (sur 10bits) de la valeur sur ADC(0) qui est en fait RA0.

Propriétés : Routine Composant

Nom à afficher : conv

Composant : ADC(0)

Macro : Ech_ADC
Lire_comme_Octet
Lire_comme_Entier

Paramètres :
Aucun paramètre n'est nécessaire pour appeler cette macro.

Valeur de retour : (OCTET)

MICRO

Variables...

OK Annuler

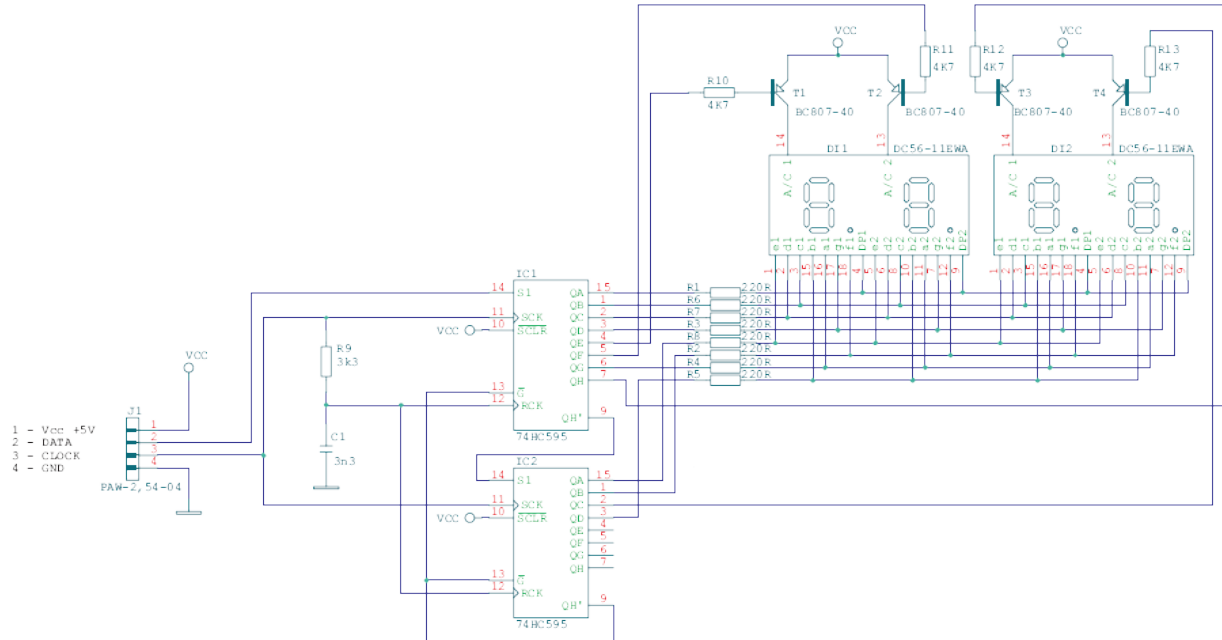
Configuration de la routine de lecture

Cette fenêtre est identique à la précédente, mais l'action choisie diffère : ici nous allons lire l'échantillon précédemment relevé et le stocker dans une variable, de type octet, définie dans le menu "valeur de retour" ici nous utiliserons la variable MICRO

PROJET N°3

Expérimentation CLOCK et DATA:

Dans cette expérimentation, nous allons combiner le projecteur à la carte de maintenance de façon à pouvoir brancher les sondes de l'oscilloscope et relever en correspondance temporelle : (CLOCK sur la source 1 et DATA sur la source 2 de l'oscilloscope).



Fonctionnement :

Les données arrivent sous forme série par le fil DATA sur l'entrée S1 du registre à décalage IC1. Les entrées SCK des circuits IC1 et IC2 reçoivent simultanément les fronts montants du signal d'horloge CLOCK

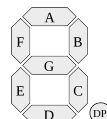
A chaque front montant du signal clock :

- Dans le circuit IC1 :
 - les registres internes sont décalés (QH reçoit QG, QG reçoit QF ... QB reçoit QA).
 - le registre interne de QA prend pour valeur l'état du signal SC1 (donc l'état de DATA)
 - le contenu du registre interne de QH est copié sur le fil QH'
- Dans le circuit IC2 :
 - les registres internes sont décalés.
 - le registre interne de QA prend pour valeur l'état du signal QH' sortant du circuit SC1
 - le contenu du registre interne de QH est envoyé sur le fil QH' donc sur les entrées G des deux circuits SC1 et SC2

A la fin de la série de 16 coups d'horloge :

- le signal DATA reste à l'état haut pendant une période de 18μs.
- donc le condensateur C1 se charge et devient équivalent à un interrupteur ouvert.
- cela provoque un état haut sur les entrées RCK.
- ce qui entraîne la copie des états des registres sur les sorties de IC1 et IC2.

Agencement des diodes dans l'afficheur 8 segments :

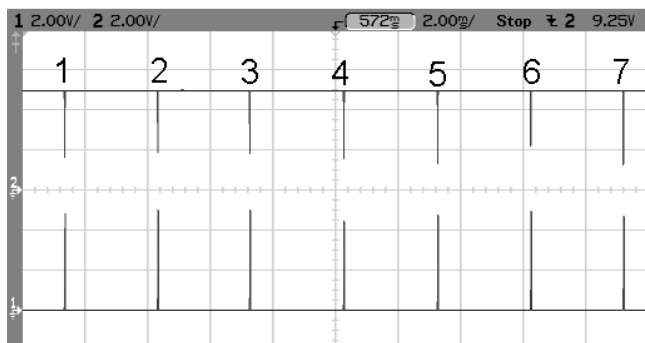


Relevés :

Dans nos relevés, la voie 1 (en bas) est branchée sur le fil CLOCK et la voie 2 sur le fil DATA.

L'afficheur digital de la lyre doit afficher la séquence :

8888 (dM.Ad.)



chronogramme global des 7 trames qui seront relevées



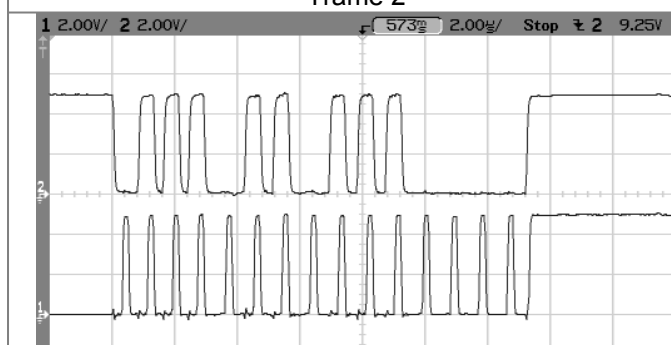
Trame 1



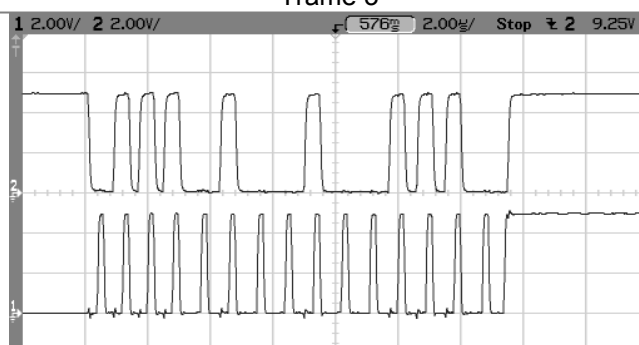
Trame 2



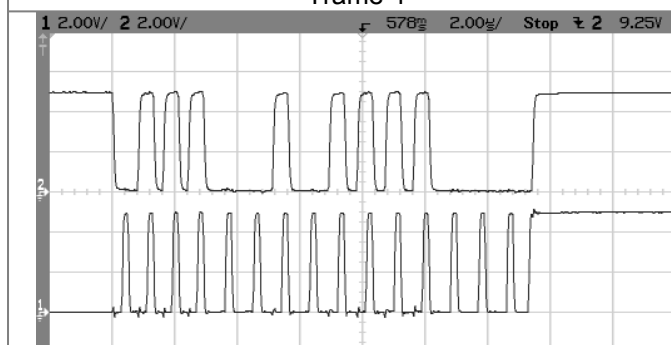
Trame 3



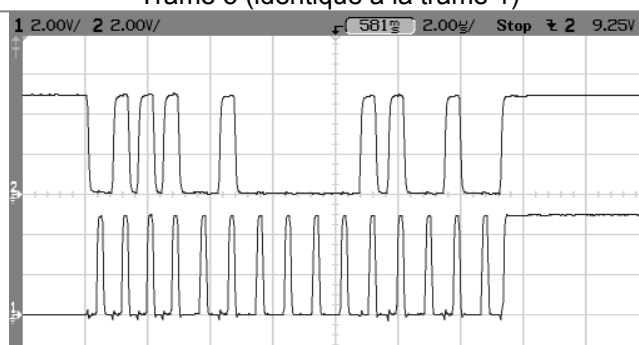
Trame 4



Trame 5 (identique à la trame 1)



Trame 6 (identique à la trame 2)



Trame 7 (identique à la trame 3)

Commentaires :

On remarque que l'ensemble des trames envoyées sont la répétition de 4 mêmes trames.

Caractéristique de CLOCK :

- le signal varie de 0 à +5V
- possède 16 front montant (donc provoque 16 décalages dans les registres)
- chaque front est espacé de $3\mu s$ donc d'une fréquence de 333.33Khz

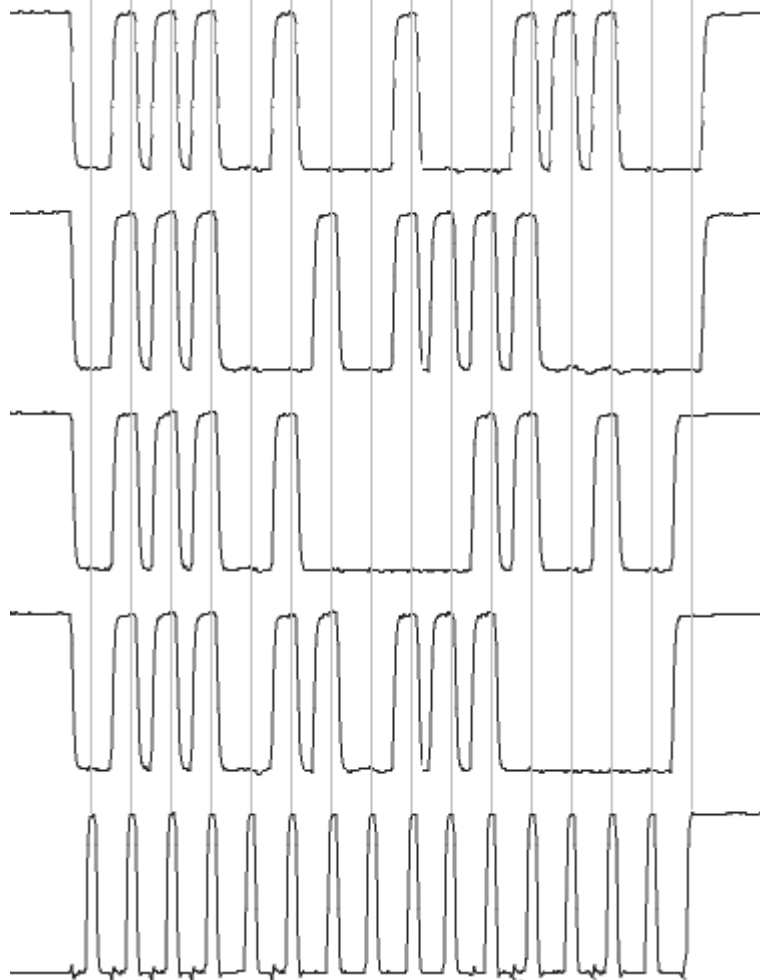
Caractéristique de DATA :

- le signal varie de 0 à +5V
- les états ne sont jamais constant : ils reviennent à 0V après le front montant de CLOCK
- reste à l'état haut jusqu'à la prochaine trame

Pour rafraîchir les 4 afficheurs, il nous faut 4 trames. chaque trame étant espacée de 3ms on en déduit

la fréquence d'affichage : $\frac{1}{4 \times 3ms} = 83.33Hz$

voici les 4 trames uniques en concordance de temps avec le signal clock (dernière ligne):



Correspondance des bits :

Selon le schéma structurel, sur les 16bits en sortie :

- 4 ne sont pas connectés.
- 4 servent à sélectionner l'afficheur sur lequel les 8 bits de donnée seront envoyés. Un état haut sur ces sorties provoque le blocage du transistor correspondant et donc de l'afficheur dont il dépend.
- 8 servent à allumer les diodes sur le(s) afficheurs sélectionnés. Les afficheurs étant à anodes communes, un état bas provoque une différence de potentiel au borne des diodes ce qui les allument. Les résistances R1 à R8 sont là pour limiter le courant les traversant.
- Le premier bit correspond à l'état des entrées G des 2 circuits. Cet état durera jusqu'au prochain envoi de DATA. Un état haut sur ces entrées met les sorties en haute impédance, ce qui correspond à un interrupteur ouvert. Les sorties ne sont donc plus reliées à la masse, donc les 4 afficheurs sont éteints. C'est pourquoi le premier bit est toujours à l'état bas : 0.
- Tableau de correspondance :

bit	Sortie	Circuit	description
1	QH'	IC2	appliqué sur les entrées G de IC1 et IC2
1	QH	IC2	Non connecté
2	QG	IC2	Non connecté
3	QF	IC2	Non connecté
4	QE	IC2	Non connecté
5	QD	IC2	Commande l'affichage du segment B
6	QC	IC2	Commande l'affichage sur le 4em afficheur
7	QB	IC2	Commande l'affichage du segment F
8	QA	IC2	Commande l'affichage du segment E
9	QH	IC1	Commande l'affichage sur le 3em afficheur
10	QG	IC1	Commande l'affichage du segment A
11	QF	IC1	Commande l'affichage sur le 2em afficheur
12	QE	IC1	Commande l'affichage sur le 1er afficheur
13	QD	IC1	Commande l'affichage du segment G
14	QC	IC1	Commande l'affichage du segment D
15	QB	IC1	Commande l'affichage du segment C
16	QA	IC1	Commande l'affichage du DP (Digital Point)

Remarque :

Contrairement à QH, QH' n'a pas besoin d'attendre de validation (copie des registres internes sur les sorties) pour influencer les entrées G des 2 circuits. Donc pendant le décalage de la nouvelle trame, les bits de la trame précédente qui sont éjectés, modifient les entrées G et font osciller les sorties entre le mode normal et le mode haute impédance. Cette période étant très courte (14µs) l'œil n'a pas le temps de percevoir de changement et ce "défaut" paraît inaperçu.

Vérification :

Selon les relevés sur le fil DATA, les 4 trames envoyées sur le fil DATA sont :

```
trame1: 0111010010011100
trame2: 0111001011110000
trame3: 0111010000110101
trame4: 0111011011100001
```

d'après le tableau exposé précédemment, nous analysons les 16 bits des 4 trames :

analyse de la trame n°1 :

- seul le 2ème afficheur est utilisé car on a **0** sur le 11ème bit.
- les diodes B,F,E,A,C et DP sont utilisées.

cela fait donc un **M.** sur le 2ème afficheur.

analyse de la trame n°2 :

- seul le 4ème afficheur est utilisé car on a **0** sur le 6ème bit.
- les diodes B,E,G,D,C et DP sont utilisées.

cela fait donc un **d.** sur le 4ème afficheur.

analyse de la trame n°3 :

- seul le 3ème afficheur est utilisé car on a **0** sur le 9ème bit.
- les diodes B,F,E,A,G et C sont utilisées.

cela fait donc un **a** sur le 3ème afficheur.

analyse de la trame n°4 :

- seul le 1er afficheur est utilisé car on a **0** sur le 12ème bit.
- les diodes B,E,G,D,C et DP sont utilisées.

cela fait donc un **d** sur le 1er afficheur.

Nous avons donc bien, au final, la chaîne **dM.Ad.** affichée sur les 4 afficheurs 8 segments :

