

CONCOURS Collège Lycée CGÉNIAL

LOIN DU CŒUR, PRÈS DES ONDES



PAR : ROCHE TITOUAN, LE REUN ELISABETH, GREINER BASTIAN



Sciences à l'**E**cole



<https://youtu.be/1MNoDaHLcdo>

PROFESSEUR : MR PICHOU

FONDATION
CGÉNIAL

PREMIERE PARTIE : LE SIMULATEUR	5
1) Pourquoi réaliser un simulateur ?	5
2) Utilisation d'un haut-parleur électrodynamique	5
3) Enregistrement de l'accélération avec son smartphone	6
4) Méthode analogique d'intégration	7
DEUXIEME PARTIE : LE DETECTEUR INTERFERENTIEL	12
1) L'interféromètre de Michelson	12
2) Notre interféromètre	12
3) La détection des variations d'amplitude	15
TROISIEME PARTIE : LE DETECTEUR MULTIPLIEUR	17
1) Principe	17
2) La multiplication des signaux	18
3) Récupération de la phase	19
4) Courbe de fonctionnement	20
QUATRIEME PARTIE : LA DETECTION DES BATTEMENTS	21
1) Détection des battements	21
2) La sensibilité	21

INTRODUCTION

Notre principale motivation pour réaliser ce projet innovant était de trouver une solution aux problèmes qui sont survenus avec la pandémie du Covid19, et plus particulièrement aux problèmes créés dans le cadre du secourisme.

En effet, nous avons passé une formation de premiers secours durant la crise du Covid19, où les formateurs insistaient sur la nécessité d'effectuer le moins de contact possible. Or le moyen le plus rapide de prendre le pouls est par contact physique. Ainsi, nous nous sommes demandé s'il était possible de le prélever à distance, et voici comment a germé l'idée de ce projet dans notre esprit !

PROBLEMATIQUE

Ainsi, tout au long de ce rapport, nous essaierons de répondre à la problématique qui a rythmé l'accomplissement de notre projet : comment mesurer un battement cardiaque à distance ?

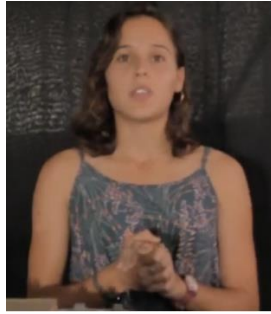
Nous avons remarqué que les battements de cœur provoquaient un mouvement de la poitrine observable à l'œil nu et détectable par l'accéléromètre d'un smartphone.

C'est ce mouvement que nous allons essayer de détecter à distance.

L'estimation de l'amplitude de ce mouvement étant de l'ordre du millimètre, c'est naturellement vers les ondes ultrasonores que nous nous sommes dirigés pour élaborer nos différents systèmes de détection.

Ce projet est innovant car il n'existe pas jusqu'ici pas d'appareil capable de détecter de manière fiable les battements de cœur à distance.

PRESENTATION DE L'EQUIPE



Elisabeth : élève de terminale spécialités maths/physique. Responsable de la réalisation d'un détecteur basé sur la mesure du déphasage entre deux ondes ultrasonores.



Titouan : élève de terminale spécialités maths/physique. Responsable de la réalisation d'un détecteur basé sur les interférences entre deux ondes ultrasonores.



Bastian : élève de terminale spécialités maths/physique. Responsable de la réalisation d'un simulateur de battements cardiaque.

PREMIERE PARTIE : LE SIMULATEUR (par Bastian)

1) Pourquoi réaliser un simulateur ?

Notre projet va consister à réaliser et tester différents systèmes de détection des mouvements générés par les battements de notre cœur.

Les différents systèmes de détection que nous allons réaliser devront être testés tout au long de notre projet. Comme il n'est pas toujours pratique de tester sur un humain nos différents prototypes, la décision de réaliser un simulateur de battements cardiaques s'est rapidement imposée. Passionné d'électronique et de codage, je décide de relever ce défi.

2) Utilisation d'un haut-parleur électrodynamique

a) Le haut-parleur électrodynamique

Nul besoin d'avoir suivi des TP de physique pour avoir observé le mouvement de la membrane d'un haut-parleur lorsque celui-ci est traversé par un signal électrique.



Le haut-parleur (HP) ne sera pas utilisé ici pour produire un son mais l'idée est d'utiliser le déplacement de sa membrane pour simuler les mouvements générés par les battements du cœur.

b) Utilisation d'un générateur basse fréquence (GBF)

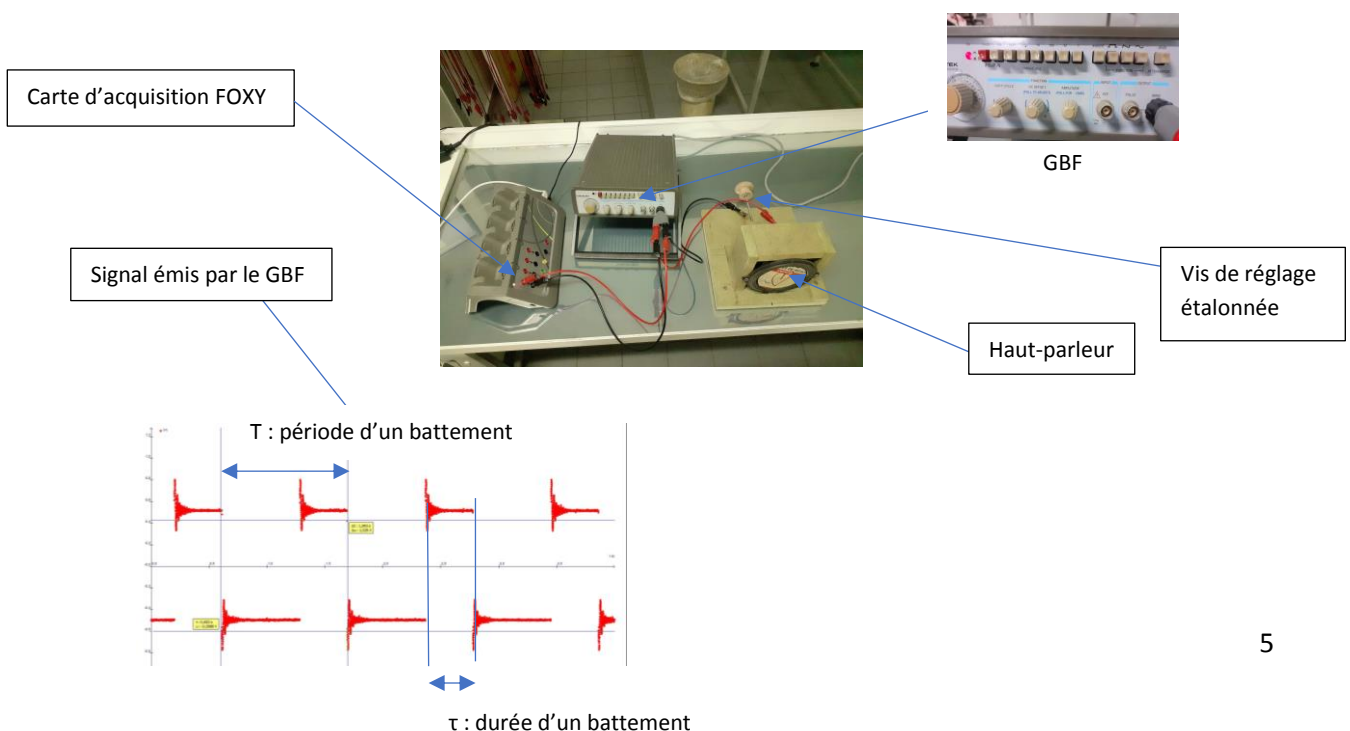
Afin que mes camarades puissent commencer à tester leurs différents prototypes, je leur ai proposé une première solution simple à mettre en œuvre.

Même si celle-ci n'est pas satisfaisante en termes de reproduction fidèle, elle nous permet d'avancer sur notre projet tout en continuant à mener parallèlement les recherches d'un meilleur dispositif.

D'après nos premières observations, le mouvement du thorax est périodique et peut-être assimilé en première approximation à un mouvement ascendant puis descendant, pour un sujet allongé. Je décide donc d'alimenter le haut-parleur avec un simple GBF en utilisant un signal « carré » de fréquence 1 Hz. En effet, au repos, la période des battements du cœur est de l'ordre de la seconde.

Pour améliorer ce premier dispositif, je décide de modifier la durée d'un battement, c'est-à-dire le rapport cyclique = τ/T du signal.

La forme de la membrane n'étant pas très ressemblante à la forme de la surface d'un thorax ou d'un abdomen, je décide de la planifier en collant un carton plan sur celle-ci.



Au-delà de sa simplicité, l'avantage de ce simulateur est de pouvoir modifier l'amplitude des déplacements via l'intensité du signal émis par le GBF ce qui va nous permettre de tester la sensibilité de nos prototypes.

Cependant, le mouvement n'était pas "naturel", et je me suis dit que, plutôt que d'envoyer au haut-parleur le signal d'un GBF, il était peut-être possible d'enregistrer le mouvement de notre propre thorax, de le convertir en signal électrique et de l'envoyer au haut-parleur.

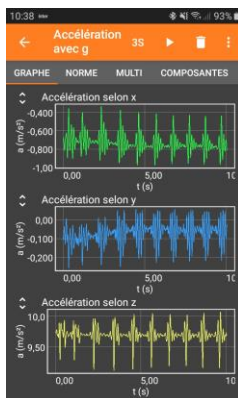
Pendant que le reste de l'équipe travaille sur différents prototypes, je poursuis mes recherches afin de me rapprocher plus fidèlement des déplacements générés par les battements de notre cœur.

3) Enregistrement de l'accélération avec un smartphone

a) L'application PHYPHOX



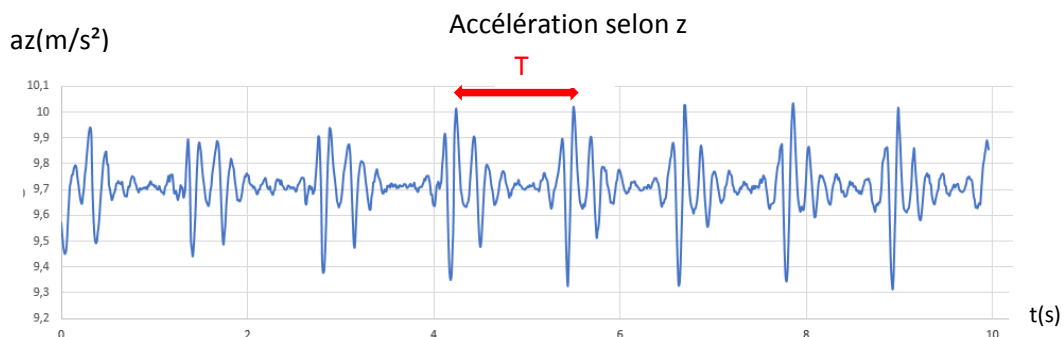
Lors d'une de nos séances de TP, notre professeur nous a présenté cette application qui utilise de nombreux capteurs présents dans nos smartphones, tel que l'accéléromètre. Sans rentrer dans les détails du principe de fonctionnement d'un accéléromètre, il faut savoir que celui-ci mesure les variations de l'accélération du téléphone par rapport à une référence qui peut être celle de l'accélération de la pesanteur terrestre. L'accélération étant vectorielle, il est bon de connaître l'orientation des axes du repère utilisé.



Vous pouvez facilement réaliser l'expérience chez vous : allongez-vous en vous munissant de votre smartphone sur lequel vous avez pris soin d'installer cette application.

Après avoir sélectionné l'accéléromètre sur l'écran d'accueil et paramétré le déclenchement et la durée d'acquisition, posez votre smartphone le plus horizontalement possible sur votre abdomen ou votre thorax. Veillez à retenir votre respiration durant l'acquisition. Voici ci-contre une copie d'écran de ce qu'on peut obtenir. Exportez ensuite vos mesures afin de pouvoir exploiter le fichier sur votre PC.

Après avoir téléchargé et ouvert le fichier sur votre PC, vous pouvez tracer à l'aide d'un tableur-grapheur l'accélération a_z selon z en fonction du temps t . En effet, nous nous intéressons ici aux seuls déplacements perpendiculaires à notre abdomen ou thorax.



Nous allons maintenant nous interroger sur la façon de transformer ces données d'accélération a_z selon z en déplacement z .

Nous avons vu en classe que l'accélération s'obtient en dérivant la vitesse v et que la vitesse s'obtient en dérivant la position.

L'idée est donc d'intégrer deux fois l'accélération afin d'obtenir la position z en fonction du temps.

4) Méthode analogique d'intégration

Notre professeur m'a indiqué l'existence de circuits électroniques capables d'intégrer des signaux électriques. Mais avant de les découvrir, il va me falloir convertir les données numériques sur l'accélération en signal électrique analogique.

a) Conversion des valeurs de l'accélération en signal électrique numérique

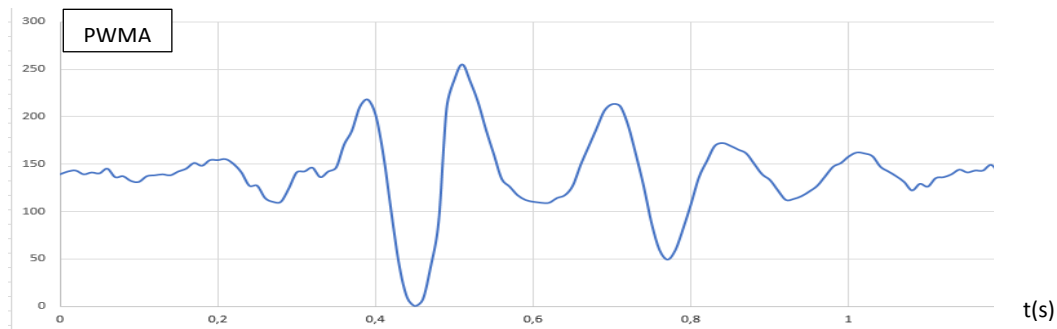
Pour pouvoir générer un signal électrique fidèle aux données de notre tableur, nous allons utiliser un microcontrôleur. L'idée est d'utiliser le port PWM d'un microcontrôleur. Ce port, utilisé en sortie, peut générer des tensions comprises entre 0V et 5V codées sur un octet. C'est-à-dire sur $2^8 = 256$ valeurs.

Une fois le fichier de données enregistré au format .csv (;) à côté d'un programme python.

Le programme arduino.py disponible en annexe ou via le lien

https://drive.google.com/drive/folders/1VdZw5jgdtSgqSnjyJpyzgrZKXF6Xb8bW?usp=share_link va :

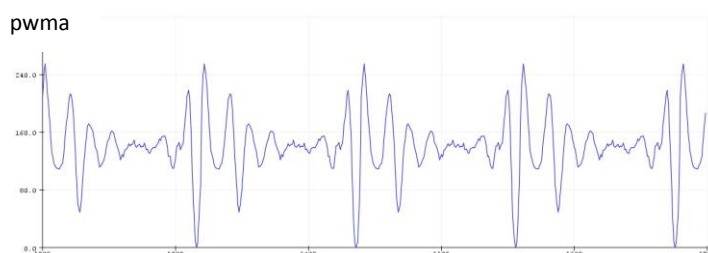
- Récupérer les valeurs de l'accélération verticale.
- Relever la plus petite valeur de l'accélération verticale et la retirer à l'ensemble des valeurs. Cela revient à rajouter un OFFSET négatif pour que la plus petite valeur soit désormais égale à 0 (toutes les valeurs sont positives).
- Relever la plus grande valeur de la nouvelle liste et créer une variable PWM = valeurs de la liste x 255 / valeur maximale. Les valeurs de PWM sont comprises entre 0 et 255.
- Créer une variable PWMA qui est l'arrondi de la variable PWM. Les valeurs sont désormais entières.



- Injecter ces valeurs PWMA dans une boucle du programme en C++ qui se répète avec un « delay » de 10 ms correspondant à la fréquence d'échantillonnage de l'application PHYPHOX de 100 Hz.

Cette boucle est placée dans la boucle principale pour reproduire la périodicité du signal : (voir programme complet pwm.ino en annexe ou via le lien fourni)

```
int tableau[] = {valeurs PWMA}; # déclaration du tableau contenant les valeurs PWMA envoyée par le programme python
int l = ; # nombre de valeurs fournies par le programme python de l'ordre de la centaine pour un patient au repos
void setup() {
  pinMode (3, OUTPUT); # mise en sortie du port PWM correspondant au PIN 3 de l'ARDUINO
}
void loop() {
  for (int i=0;i<l;i++) # boucle qui se répète autant de fois qu'il y a de valeurs.
  {
    analogWrite (3,c[i]); # on écrit une valeur du tableau dans le port 3
    delay(10); # on attend 10 ms avant d'écrire la valeur suivante
  }
}
```

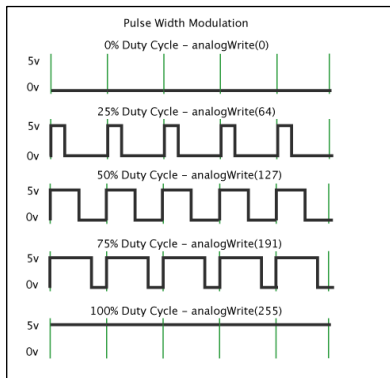
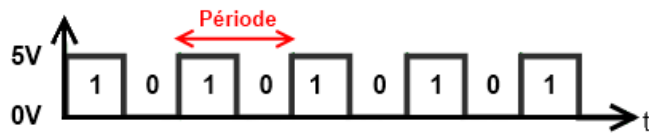


b) Notre convertisseur numérique-analogique (CNA)

Nous disposons désormais de données binaires que nous souhaitons convertir en signal analogique.

La PWM est en fait un signal numérique qui, à une **fréquence** donnée (490 Hz pour l'ARDUINO), a un **rapport cyclique** qui change.

Prenons l'exemple d'un signal logique qui émet un 1, puis un 0, puis un 1, puis un 0...



Le *rapport cyclique*, un mot bien particulier pour désigner le fait que le niveau logique 1 peut ne pas durer le même temps que le niveau logique 0.

Le rapport cyclique est mesuré en pourcentage (%). Plus le pourcentage est élevé, plus le niveau logique 1 est présent dans la période et moins le niveau logique 0 l'est. En somme, cette image extraite de la documentation officielle d'Arduino nous montre quelques exemples d'un signal avec des rapports cycliques différents.

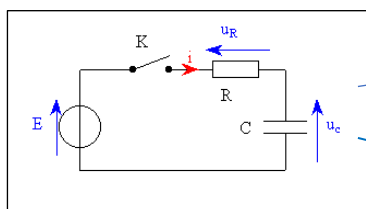
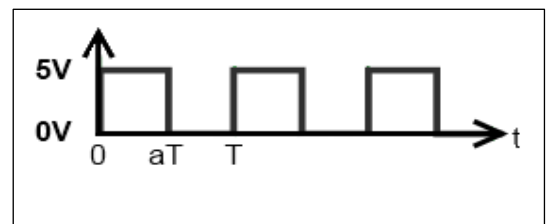
Après avoir généré ce signal, il va nous falloir le transformer en signal analogique. Pour l'instant ce signal est encore constitué d'états logiques, on va le transformer en extrayant sa *valeur moyenne*.

Reprenons notre signal carré : $0 \leq a \leq 1$ ($a=0$ pour un rapport cyclique de 0 % et $a=1$ pour un rapport cyclique de 100 %).

La valeur moyenne de ce signal est égale à : $5a$ en volt.

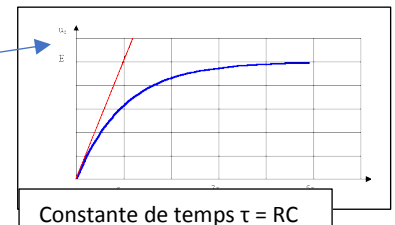
En extrayant la valeur moyenne pour chaque valeur de notre PWM, on va pouvoir créer un signal analogique compris entre 0 et 5V.

Pour réaliser cette extraction, nous allons utiliser la charge et la décharge d'un condensateur C à travers une résistance R .

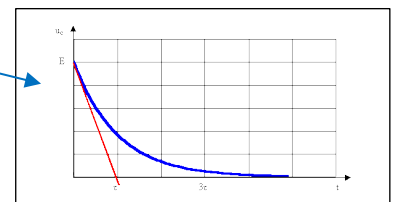


K fermé ($E=5V$) : le condensateur se charge

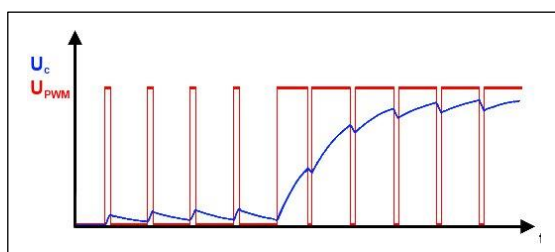
K ouvert ($E=0V$) : le condensateur se décharge



Constante de temps $\tau = RC$



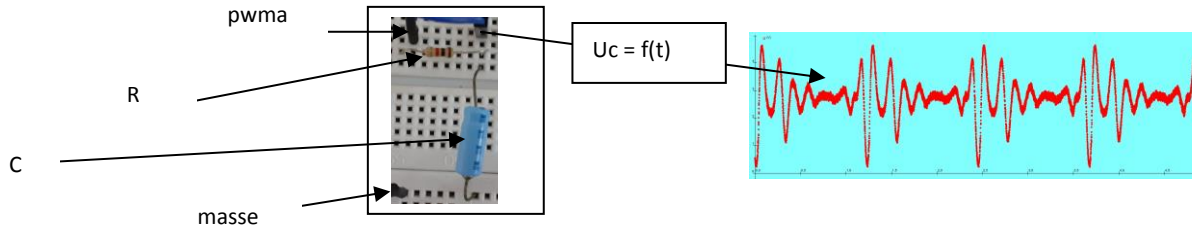
Si on impose notre signal PWM au circuit RC, à chaque fois que le signal de la PWM sera au NL 1 (Niveau Logique 1), le condensateur va se charger. Dès que le signal repasse au NL 0, le condensateur va se décharger :



Si on choisit une constante de temps RC suffisamment petite, le signal va se stabiliser autour d'une valeur comprise entre 0 (rapport cyclique nul) et 5V (rapport cyclique de 100 %). Cette valeur correspond à la valeur moyenne du PWM. Il faut que cette valeur soit atteinte avant que le rapport cyclique ne change (ici toutes les 10 ms). La constante RC ne doit pas être trop petite sous peine de voir le signal suivre le PWM. Elle doit donc être suffisamment grande par rapport à la période du PWM ($f = 490 \text{ Hz}$ soit une période d'environ 2 ms).

Pour résumer : $2 \text{ ms} < RC < 10 \text{ ms}$

Après différents essais, je choisis la combinaison suivante : $R = 550 \Omega$ et $C = 10 \mu\text{F}$ soit $RC = 5,5 \text{ ms}$

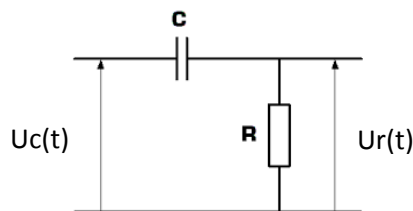


Nous disposons dorénavant d'un signal électrique analogique reproduisant l'allure de l'accélération de notre abdomen. Voyons maintenant comment intégrer ce signal.

c) Intégration électronique

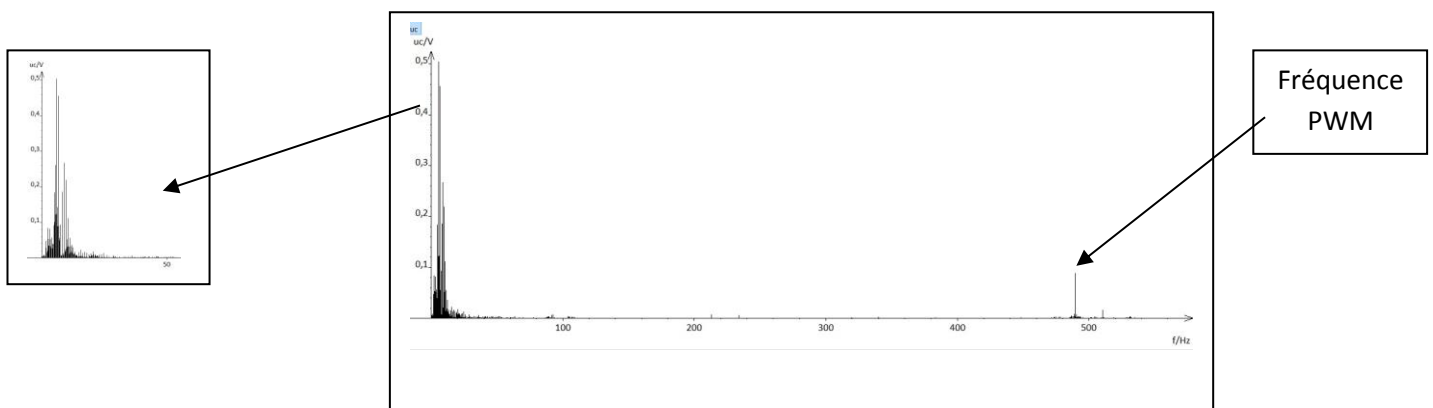
- Suppression de la composante continue

Comme la primitive d'une constante est une fonction linéaire, nous devons la supprimer pour ne pas avoir de dérive de notre signal. Nous utilisons un simple filtre passe-haut CR.

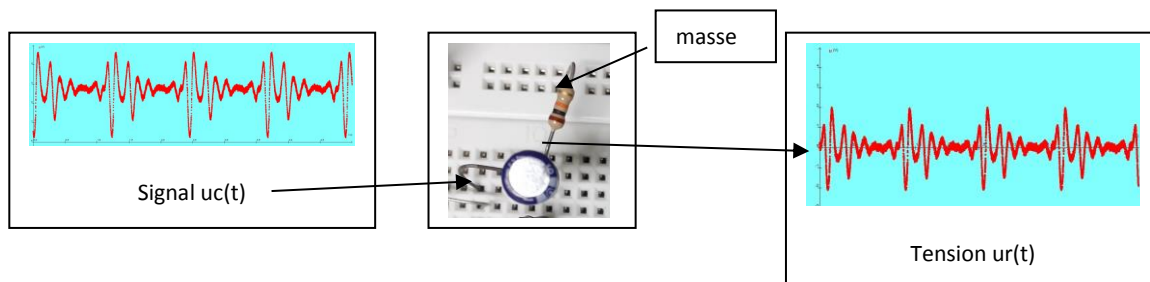


Un tel filtre possède une fréquence de coupure $f_c = \frac{1}{2\pi RC}$. C'est-à-dire que les seules les fréquences supérieures à f_c seront conservées.

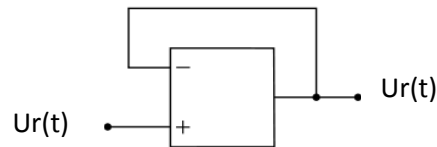
Pour déterminer les valeurs de R et C, nous avons effectué l'analyse spectrale du signal $u_c(t)$ précédent : nous remarquons un résidu de la fréquence du PWM à 490 Hz et une concentration de pics autour de la valeur 6 Hz.



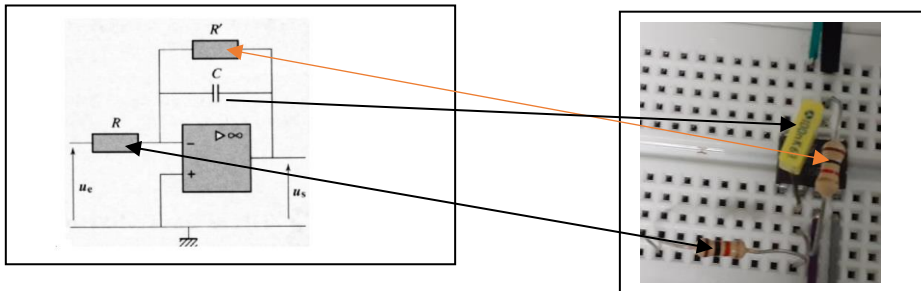
Nous choisissons $R = 10 \text{ k}\Omega$ et $C = 10 \mu\text{F}$ ce qui correspond à $f_c = 1,6 \text{ Hz}$



- **Le circuit suiveur** : sa sortie va se comporter comme une source idéale de tension de par la faible impédance de sortie de l'AOP. On ne modifiera pas le comportement de l'intégrateur.



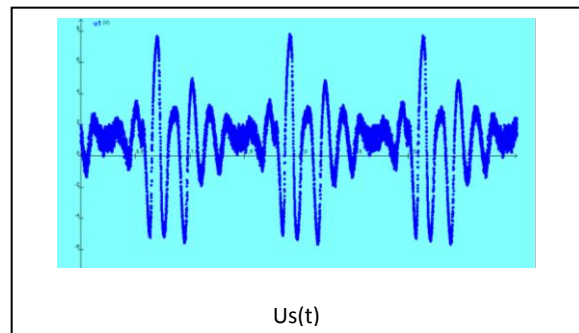
- **Le circuit intégrateur** : ce circuit permet d'intégrer le signal d'entrée si la période de celui-ci est très petite devant $R'C$



$$u_s = -\frac{1}{RC} \int U_e dt \text{ avec ici } u_e = u_r(t)$$

Je choisis $R' = 1 \text{ M}\Omega$ et $C = 10 \text{ }\mu\text{F}$ ce qui correspond à un produit $R'C = 10\text{s}$.

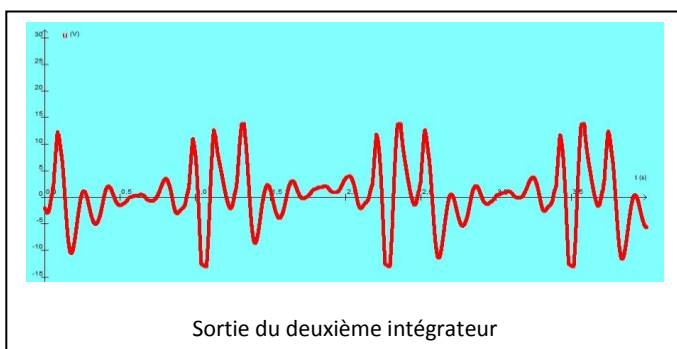
En effet, l'analyse spectrale a montré que les fréquences les plus basses qui composent le signal sont de l'ordre de quelques Hz ce qui correspond à des périodes de l'ordre de la seconde.



Nous obtenons un signal censé reproduire l'allure de la variation de la vitesse du thorax en fonction du temps.

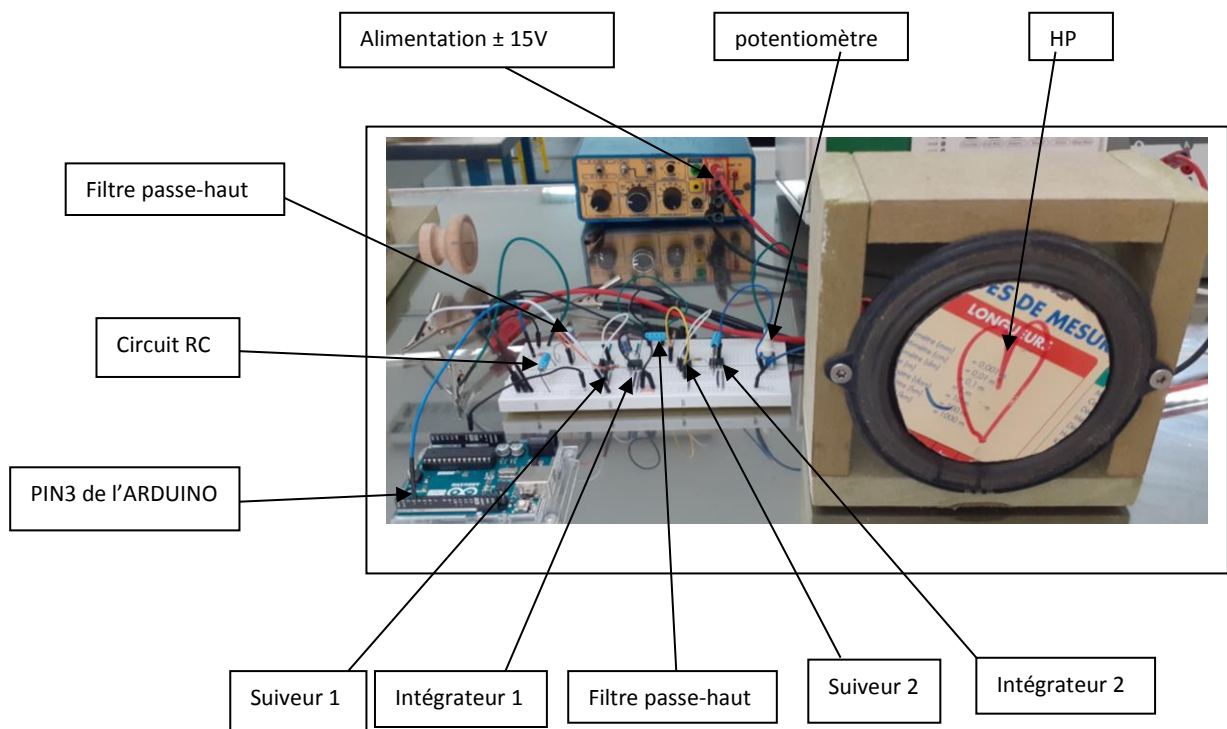
Nous constatons cependant la présence d'une composante continue que nous allons filtrer comme précédemment grâce à un filtre passe-haut de mêmes caractéristiques.

Nous répétons l'opération précédente grâce à un deuxième circuit intégrateur identique au précédent en intercalant un deuxième circuit suiveur entre les deux.



Nous obtenons un signal censé reproduire l'allure de la variation de la position du thorax en fonction du temps.

Je connecte le haut-parleur à la sortie du circuit en intercalant un potentiomètre pour pouvoir faire varier l'amplitude du mouvement du simulateur.



d) Protection du microcontrôleur

Les données du constructeur du microcontrôleur nous indiquent de ne pas dépasser une intensité de 20 mA dans les broches de la carte. Or la tension générée à la sortie du deuxième intégrateur présente une amplitude de l'ordre d'une dizaine de volts. La résistance de notre haut-parleur étant de 4Ω , le courant traversant le circuit serait de l'ordre de l'ampère !

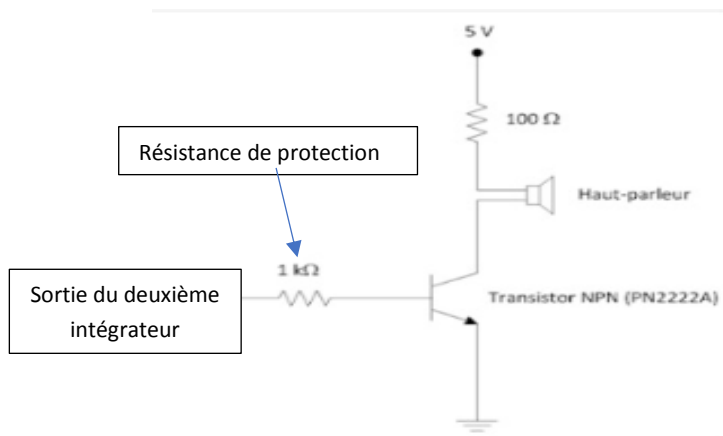
Même si ce courant n'est pas continu, nous prenons le risque de détériorer la carte.

La solution consistant à rajouter une résistance de protection en série avec le haut-parleur ne convient pas car le courant serait trop faible pour mettre la membrane en mouvement.

Le constructeur nous indique également que la broche 5V de la carte peut supporter jusqu'à 500 mA.

L'idée est donc d'utiliser cette broche pour alimenter notre haut-parleur et de piloter le courant qui le traverse avec notre circuit en le protégeant avec une résistance de $1\text{ k}\Omega$ afin de ne pas détériorer notre carte.

Après recherches, le composant électronique qui nous permet de réaliser ce pilotage est un transistor.



Calcul de la résistance à mettre en série avec le haut-parleur : $5/0,5 = 10\Omega$

Mais $P = RI^2 = 10 \times 0,5^2 = 2,5\text{W} > \text{limite constructeur} = 0,25\text{W}$

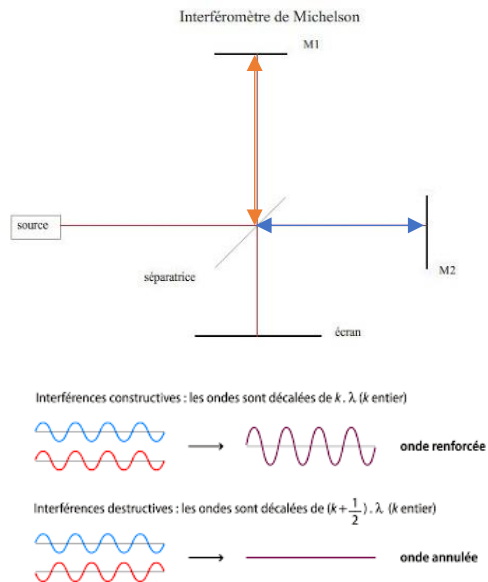
donc $R = 100\Omega$ car $I_{\text{max}} = 5/100 = 0,05\text{A} < 500\text{mA}$ et $P = 100 \times 0,05^2 = 0,25\text{W}$

Nous avons toujours la possibilité de rajouter un potentiomètre en série avec la résistance de 100Ω pour faire varier l'amplitude.

Pendant que je travaillais sur l'amélioration de mon simulateur, mes camarades se penchaient sur l'élaboration de deux détecteurs de battements cardiaques qu'ils vont vous présenter dans la suite de ce rapport.

1) L'interféromètre de Michelson

L'interféromètre de Michelson est un dispositif optique mis au point vers 1887 par Albert Abraham Michelson et Edward Morley. Ce dispositif produit des interférences par division d'amplitude. Il est constitué essentiellement de deux miroirs M1 et M2 et d'une lame semi-réfléchissante appelée séparatrice car elle sépare le signal émis par la source en deux ondes de même amplitude. L'une est réfléchi vers M1 et l'autre est transmise vers M2.



Supposons la source monochromatique.

Les deux ondes effectuent donc un aller-retour entre la séparatrice et les miroirs M1 et M2 avant de se superposer en direction de l'écran après respectivement transmission et réflexion sur la séparatrice.

Ces deux ondes étant cohérentes, elles vont interférer.

Si le trajet rouge est égal au trajet bleu, les deux ondes seront en phases et l'interférence sera constructive. L'amplitude sur l'écran sera maximale.

Si on recule l'un des miroirs d'un quart de la valeur de la longueur d'onde, le trajet va augmenter d'une demi longueur d'onde et les interférences seront destructives. L'amplitude sur l'écran sera nulle.

Ce phénomène se répète périodiquement. On observe une alternance d'interférences constructives et destructives à chaque déplacement d'un des miroirs d'une demi longueur d'onde.

Contrairement à celui de Michelson, notre interféromètre ne sera pas lumineux mais ultrasonore.

2) Notre interféromètre

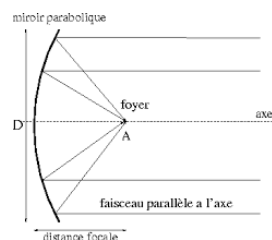
a) La source et le récepteur

La source, l'émetteur d'ultrasons, n'étant pas directive, il faut focaliser le faisceau d'ultrasons. Plus exactement, il faut que l'onde émise soit plane. C'est à dire que les fronts d'onde soient parallèles entre eux.

L'utilisation d'une lentille de paraffine n'est pas adaptée car les phénomènes d'absorption et de réflexion sont trop importants et le signal à la sortie est trop faible pour être détecté.

J'opte donc pour une focalisation par réflexion.

Pour obtenir une onde plane, il faut placer la source à la focale d'un miroir parabolique.



Après avoir fouillé avec le technicien dans les armoires du laboratoire, mon choix se porte sur ce dispositif qui à l'origine sert à chauffer les ballons en chimie. Ici, il servira de miroir pour les ultrasons.

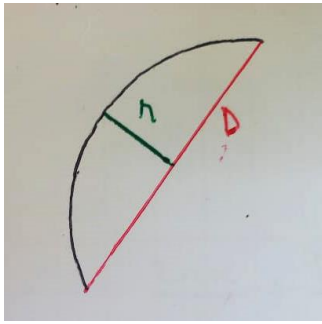


Sa forme n'est pas parabolique mais sphérique. En me documentant sur les effets d'un tel miroir, j'apprends qu'en plaçant une source sur son axe à une distance égale à la moitié de son rayon, l'onde émise sera plane à condition de ne pas trop s'éloigner de son axe. On appelle ça les conditions de Gauss en optique.

Il ne me reste plus qu'à trouver le rayon de ce miroir. Le chauffe ballon est prévu pour s'adapter à un ballon de 250 mL. Je mesure au pied à coulisse le diamètre d du ballon qui correspond au diamètre du « miroir ».



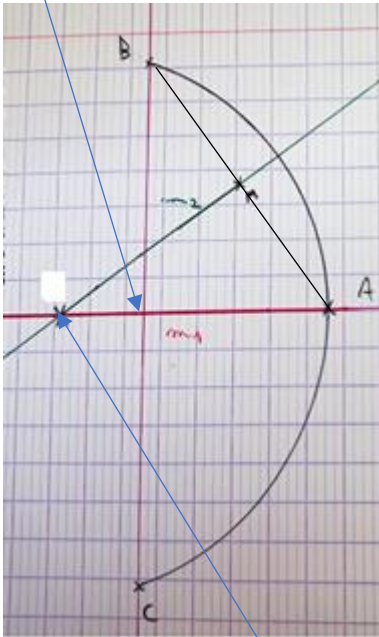
Cette méthode n'étant pas très satisfaisante, nous mettons à profit nos cours de mathématiques pour résoudre le problème suivant :



On peut mesurer facilement r et D , la largeur et la profondeur du chauffe-ballon et on cherche le rayon de la sphère qui vient s'y emboîter.

C'est Bastian qui a résolu le premier ce problème de géométrie !
Voici sa solution :

Origine du repère



Centre de la sphère

Les coordonnées des points A, B et C sont respectivement :

$(r ; 0) ; (0 ; D/2) ; (0 ; -D/2)$

On sait que le centre de la sphère est à l'intersection des médiatrices des segments [AB] et [BC].

L'équation de m_1 est facile à trouver : $y=0$

Pour m_2 , c'est un peu plus compliqué...

La pente de (AB) est $-D/2r$ donc celle de m_2 est $2r/D$ (l'opposé de l'inverse).

Les coordonnées du point M, milieu de [AB] sont les demi-sommes des coordonnées de A et B soient : $(r/2 ; D/4)$

On trouve l'ordonnée à l'origine b de la droite m_2 en injectant les coordonnées de M dans l'équation de m_2 : $2r/D \times r/2 + b = D/4$ soit $b = (D^2 - 4r^2)/4D$

On résout ensuite l'équation suivante pour trouver l'abscisse x du centre de la sphère :

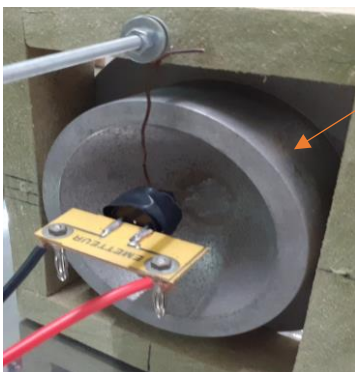
$2rx/D + (D^2 - 4r^2)/4D = 0$ soit $x = (4r^2 - D^2)/8r < 0$

Le rayon cherché est donc : $r - (4r^2 - D^2)/8r$

Avec les valeurs mesurées de r et D , on trouve $d = 2CA = 9,0$ cm

Il ne nous reste plus qu'à fixer notre émetteur à $d/4 = 2,25$ cm du fond du chauffe ballon. Nous considérons que l'onde émise est plane.

Nous fixons notre récepteur de la même façon. L'onde plane reçue sera focalisée sur le récepteur. Ce dispositif remplacera l'écran dans notre interféromètre.



Emetteur d'onde plane

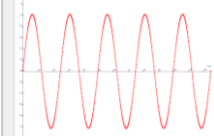
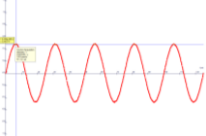
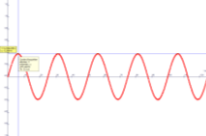
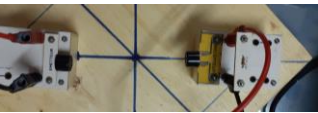

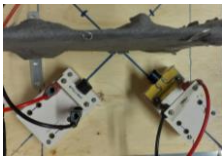
Récepteur




b) La séparatrice

Nous devons trouver une surface plane constituée d'une matière qui transmette 50% de l'onde ultrasonore et qui réfléchisse le reste.

Je teste différents matériaux en transmission et en réflexion que je fixe sur un support de ma fabrication.

Amplitude signal récepteur	1000 mV	500 mV	450 mV
%	100	50	45
Signal récepteur			
Montage			

Quelques matières testées					
Transmission %	33	25	72	70	85

Je retiens le tissu qui se rapproche le plus d'une transmission et d'une réflexion de 50 % chacune.

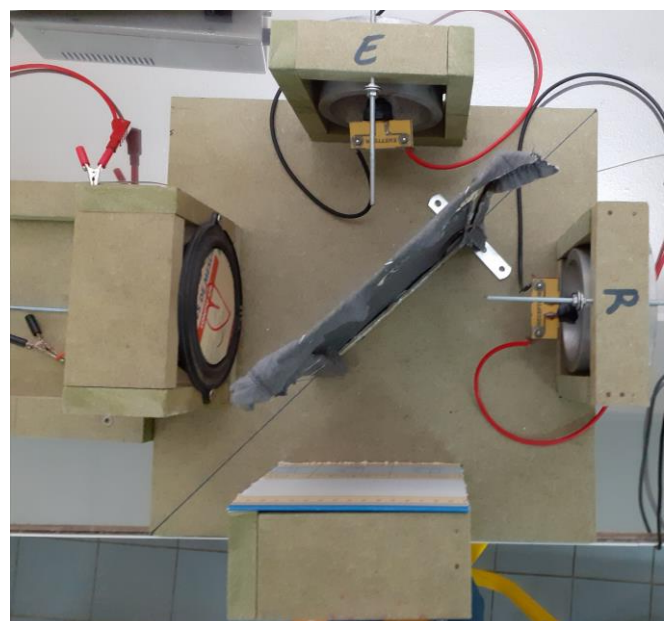
c) Les miroirs

Un miroir sera fixe et l'autre sera constitué du simulateur de battements.



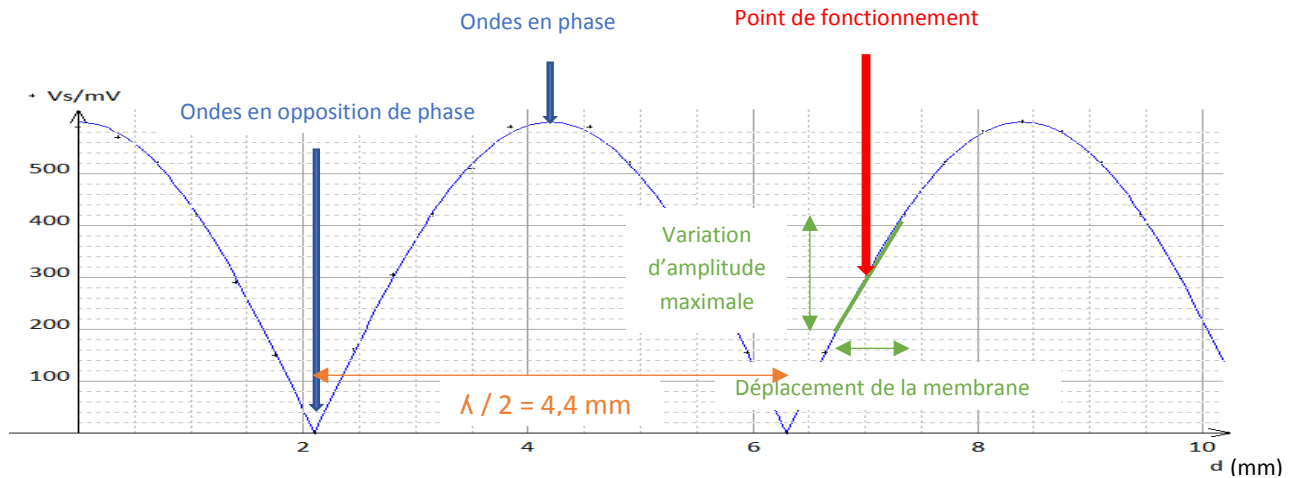
Miroir fixe

d) notre interféromètre



e) La courbe de fonctionnement

On alimente l'émetteur avec un GBF délivrant un signal sinusoïdal de fréquence 40 kHz. On observe le signal reçu grâce à une carte d'acquisition reliée aux bornes du récepteur. On affine le réglage de la fréquence afin d'avoir le signal reçu avec la plus grande amplitude possible. On relève la valeur de l'amplitude V_s en fonction du déplacement d de la membrane du haut-parleur. A chaque demi-tour de vis, le haut-parleur avance de 0,35 mm et la distance parcourue par les ultrasons entre l'émission et la réception augmente de 0,7 mm à cause de l'aller-retour. La période de la courbe de fonctionnement est donc égale à une demi longueur d'onde ultrasonore.



Le point de fonctionnement (position optimale du haut-parleur et dans la pratique, du thorax du patient) se trouve à mi-chemin entre une deux interférences constructives et destructives. On dit que les ondes sont en quadrature de phase. Concrètement, on se placera à la moitié du maximum d'amplitude pour avoir une détection optimale.

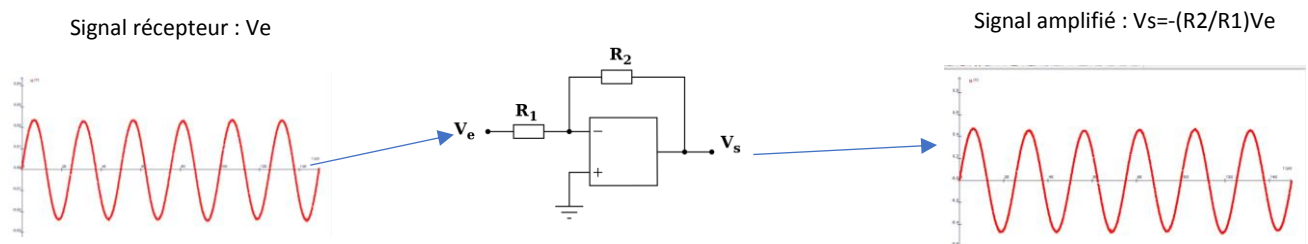
3) La détection des variations d'amplitude

a) Le détecteur

Lors des battements, l'amplitude du signal reçu par le récepteur varie. Pour détecter ces variations, il nous faut donc réaliser un dispositif électronique capable de détecter l'amplitude du signal reçu.

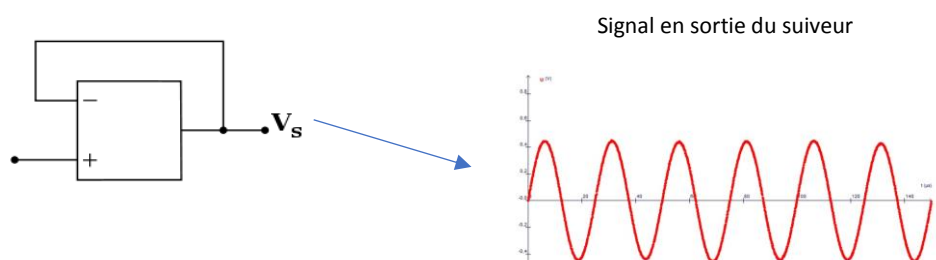
Mon dispositif se compose de trois parties :

Un circuit amplificateur : l'amplitude du signal reçu étant de l'ordre de quelques centaines de mV, il faut l'amplifier afin de pouvoir la détecter. On utilise un circuit amplificateur inverseur grâce à un AOP UA741.

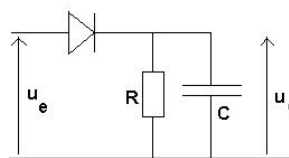


J'ai choisi $R_2 = 100 \text{ k}\Omega$ et $R_1 = 1 \text{ k}\Omega$ ce qui correspond théoriquement à une amplification égale à 100.

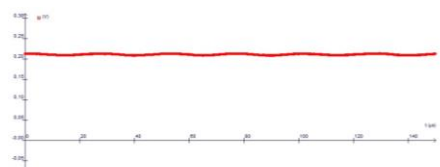
Un circuit suiveur : sa sortie va se comporter comme une source idéale de tension de par la faible impédance de sortie de l'AOP. On ne modifiera pas le comportement du détecteur de crête.



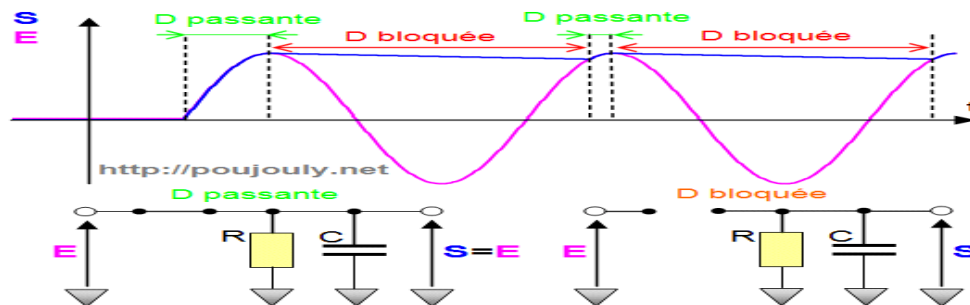
Un circuit détecteur de crête :



Signal en sortie du détecteur de crête



Pour simplifier l'étude du fonctionnement de ce montage on peut considérer une diode D idéale sans seuil comme le montre le chronogramme suivant :



On choisit une condition initiale où le condensateur C est totalement déchargé ce qui signifie que $S=0$. Dès que la tension d'entrée $E>0$ la diode D devient alors passante ce qui signifie que $S=E$. Si la tension E chute rapidement, le condensateur C maintient une certaine tension $S>E$ aux bornes de la résistance R. Dans ces conditions la diode devient bloquée et le condensateur se décharge (lentement) dans la résistance R. Si la constante de temps RC est choisie suffisamment grande devant la période du signal d'entrée E on récupère bien la tension crête en sortie de ce montage ce qui justifie le nom de ce montage.

En présence d'un signal modulé en amplitude, le choix de la constante de temps RC doit répondre à l'encadrement suivant pour permettre une bonne récupération du signal modulant :

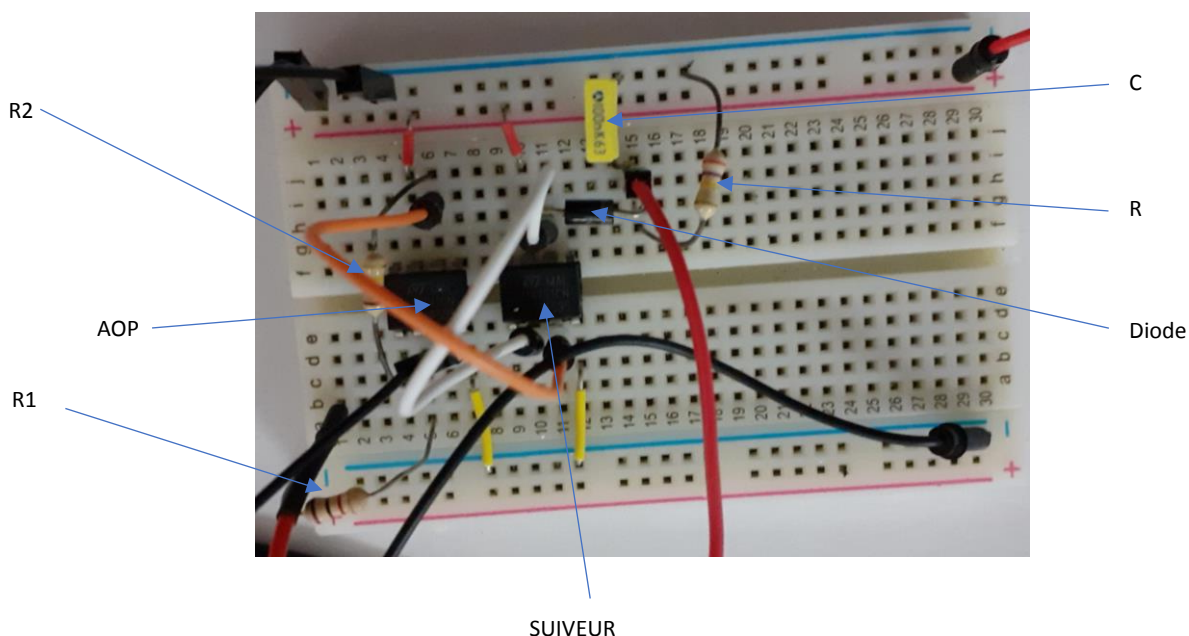
$$10T_{\text{porteuse}} < RC < T_{\text{modulation}}/10$$

Cet encadrement traduit le fait que le condensateur doit se charger suffisamment vite par rapport aux variations de la modulation, c'est-à-dire les variations de positions de notre simulateur et suffisamment lentement par rapport aux variations de la porteuse, c'est-à-dire le signal ultrasonore.

Ici, $T_{\text{modulation}}$ est de l'ordre de la seconde alors que T_{porteuse} est l'inverse de la fréquence ultrasonore soit environ $25 \mu\text{s}$.

$$\text{Ce qui donne : } 2,5 \times 10^{-4} \text{ s} < RC < 0,10 \text{ s}$$

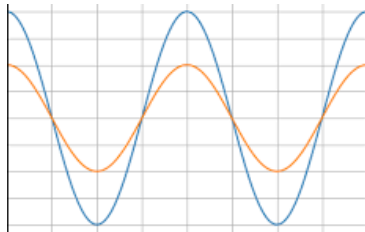
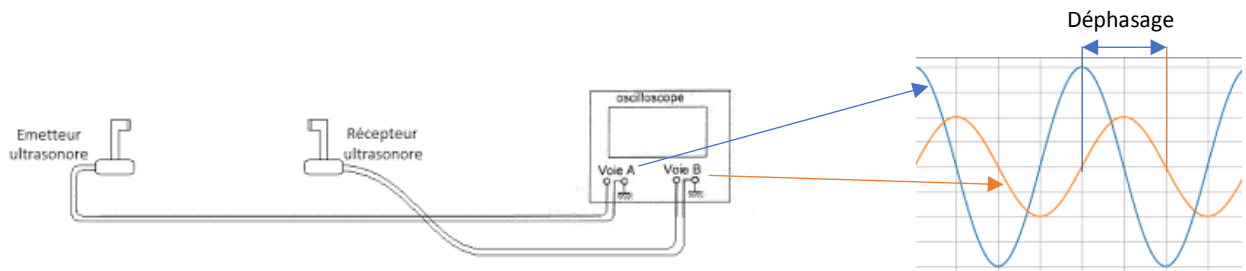
Après différents essais, je choisis $C = 100 \text{ nF}$ et $R = 270 \text{ k}\Omega$ soit $RC = 2,7 \times 10^{-2} \text{ s}$



1) Principe

Le dispositif que nous allons vous présenter a été inspiré par une séance de travaux pratiques lors de laquelle nous avons appris à mesurer la longueur d'onde d'une onde périodique progressive ultrasonore.

Lorsque l'on place un émetteur ultrasonore alimenté par une tension sinusoïdale de fréquence 40 kHz à une distance d quelconque d'un récepteur ultrasonore, les tensions électriques aux bornes de ces deux composants sont déphasées.



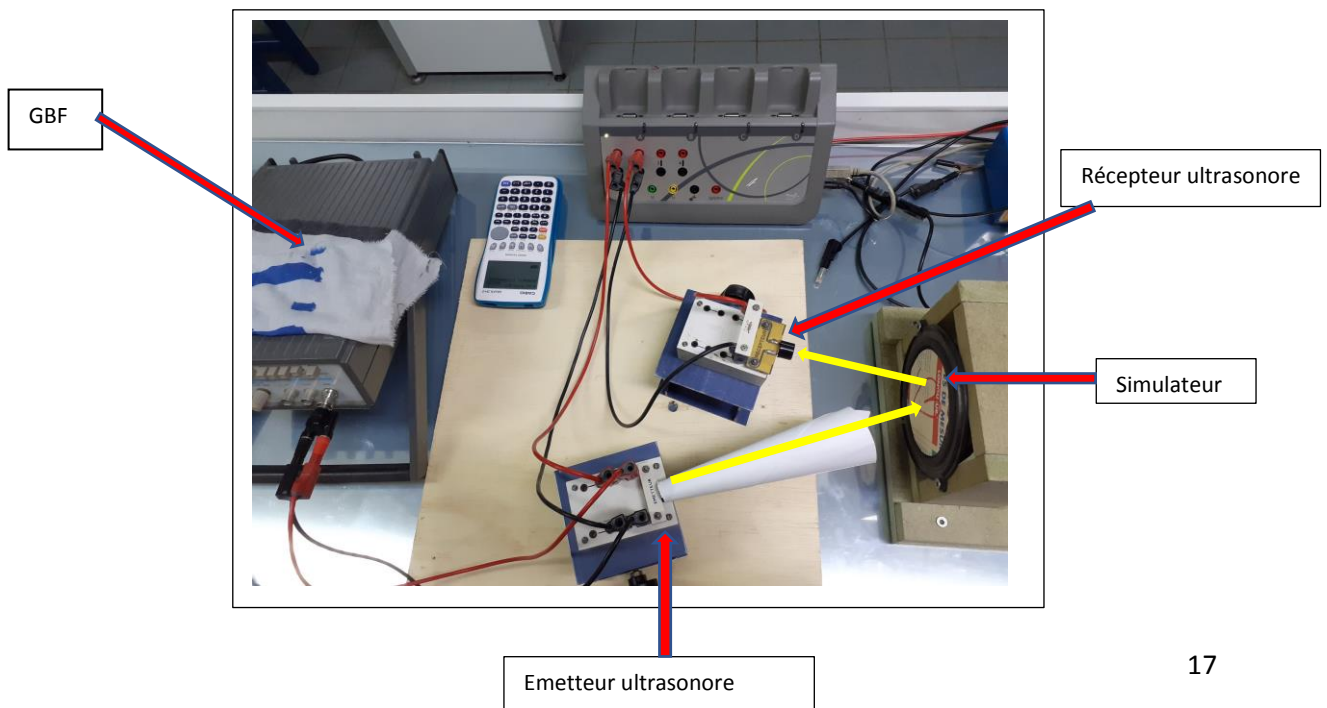
Si on recule lentement le récepteur, on observe que le déphasage varie. Pour une certaine position particulière du récepteur, ce déphasage est nul. On dit que les ondes sont en phase. En termes de mathématiques, on dit que les deux sinusoïdes sont déphasées d'un multiple entier de 2π .

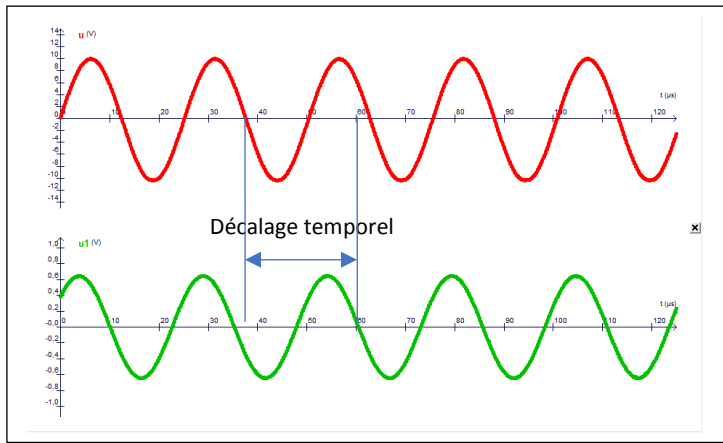
Si on continue de reculer le récepteur, les ondes se déphasent pour revenir en phase à nouveau au bout d'une distance proche de 9 mm. Si on recule encore le récepteur de 9 mm, les ondes se retrouvent à nouveau en phase. On constate une périodicité spatiale de l'onde que l'on nomme longueur d'onde notée λ . On peut améliorer la précision de la mesure en mesurant le déplacement correspondant à un grand nombre de fois la longueur d'onde.

On trouve $\lambda = (8,8 \pm 0,1) \text{ mm}$.

C'est la distance parcourue par l'onde pendant la période temporelle T du signal. La fréquence f de fonctionnement de l'émetteur est de 40,0 kHz. On peut en déduire la célérité v des ultrasons dans l'air grâce à la relation $v = \lambda \times f$ soit ici : $v = 352 \text{ m/s}$.

La longueur d'onde des ultrasons étant supérieure à l'amplitude des mouvements provoqués par les battements du cœur, l'idée est de placer l'émetteur et le récepteur côte à côte en face de notre simulateur de battements cardiaques. En effet, lorsque celui-ci va se mettre en mouvement, la distance d entre l'émetteur et le récepteur va varier au cours du temps et donc le déphasage aussi : sur la photo ci-dessous, les flèches jaunes représentent le trajet des ultrasons





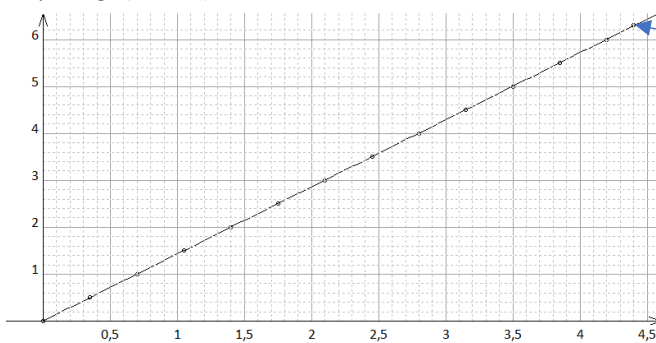
Signal émis

Signal reçu

Ce déphasage étant proportionnel au déplacement, il contient donc l'information qui nous intéresse. Nous avons tracé la courbe du déphasage (décalage temporel $\times 2\pi \times f$) en fonction du déplacement du simulateur en partant d'une position prise pour origine lorsque les courbes sont en phases.

On remarque sans surprise que pour un déplacement d'une demi longueur d'onde (4,4 mm), les deux courbes se retrouvent en phase. Le déphasage a varié de $2\pi = 6,28$. En effet, le trajet a alors augmenté d'une longueur d'onde, ce qui correspond à la distance parcourue pendant une période temporelle.

Déphasage (radians)



(4,4 mm ; 6,28 rad)

Nous allons voir dans la suite comment récupérer ce déphasage.

Position du simulateur (mm)

2) La multiplication des signaux

a) La théorie

Pour accéder au déphasage entre l'onde émise et l'onde reçue, nous allons multiplier les deux signaux.

Le signal S_E émis peut-être modélisé par une sinusoïde : $S_E = A_E \cos\left(\frac{2\pi t}{T} + \varphi_E\right)$

φ_E étant la phase du signal émis en sortie d'émetteur et A_E son amplitude.

Le signal S_R reçu peut-être modélisé par une sinusoïde : $S_R = A_R \cos\left(\frac{2\pi t}{T} + \varphi_R\right)$

φ_R étant la phase du signal reçu en entrée de récepteur et A_R son amplitude.

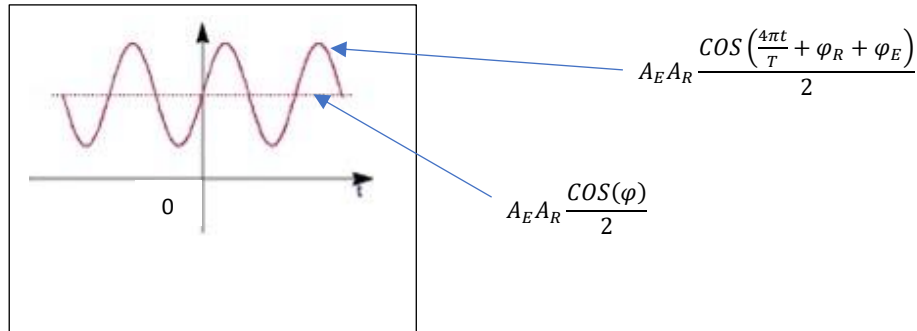
Le déphasage entre le signal reçu et le signal émis sera noté $\varphi = \varphi_R - \varphi_E$

Si on multiplie ces deux signaux, on obtient un signal $S = S_E \times S_R = A_E \cos\left(\frac{2\pi t}{T} + \varphi_E\right) \times A_R \cos\left(\frac{2\pi t}{T} + \varphi_R\right)$

On utilise ensuite la propriété suivante : $\cos(a) \times \cos(b) = \frac{\cos(a+b) + \cos(a-b)}{2}$

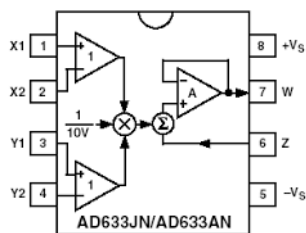
Donc : $S = A_E A_R \frac{\cos\left(\frac{4\pi t}{T} + \varphi_R + \varphi_E\right)}{2} + A_E A_R \frac{\cos(\varphi)}{2}$

Nous obtenons un signal de fréquence double soit 80 kHz auquel se rajoute une composante continue (OFFSET) contenant le déphasage φ .



b) La pratique

Afin de multiplier nos signaux électriques, nous allons utiliser un composant électronique nommé multiplieur AD-633 alimenté en $V_s = \pm 15V$



Le multiplieur donne en sortie : $W = \frac{(X1 - X2)(Y1 - Y2)}{10} + Z$

Il faut donc connecter l'émetteur à la borne X1 et le récepteur à la borne Y1.

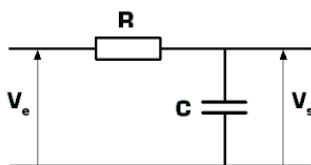
Les bornes X2, Y2 et Z sont reliées à la masse.



3) Récupération de la phase

a) Le filtre passe bas RC

Le signal W est composé d'une composante sinusoïdale de fréquence 80 kHz et d'une composante continue. En électronique, il existe un dispositif très simple composé d'une résistance R et d'un condensateur C permettant de supprimer les hautes fréquences d'un signal. On l'appelle le filtre passe bas :

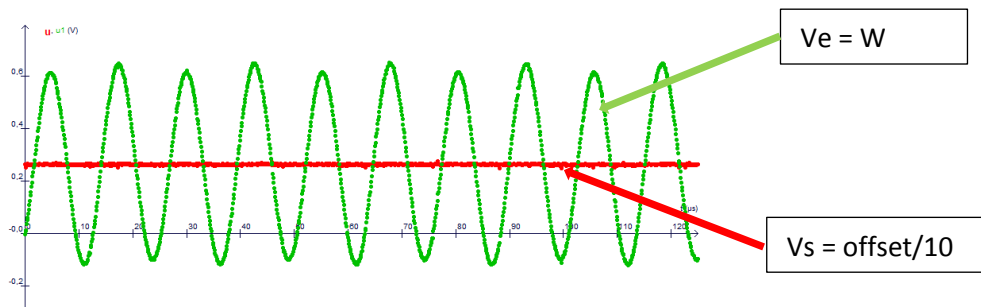
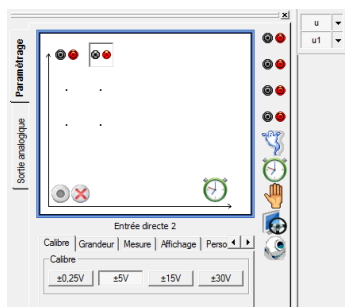


Si on applique la tension $W = V_e$ à l'entrée d'un filtre passe bas judicieusement choisi, la

tension de sortie sera : $V_s = \frac{A_E A_R \cos(\varphi)}{10 \cdot 2}$

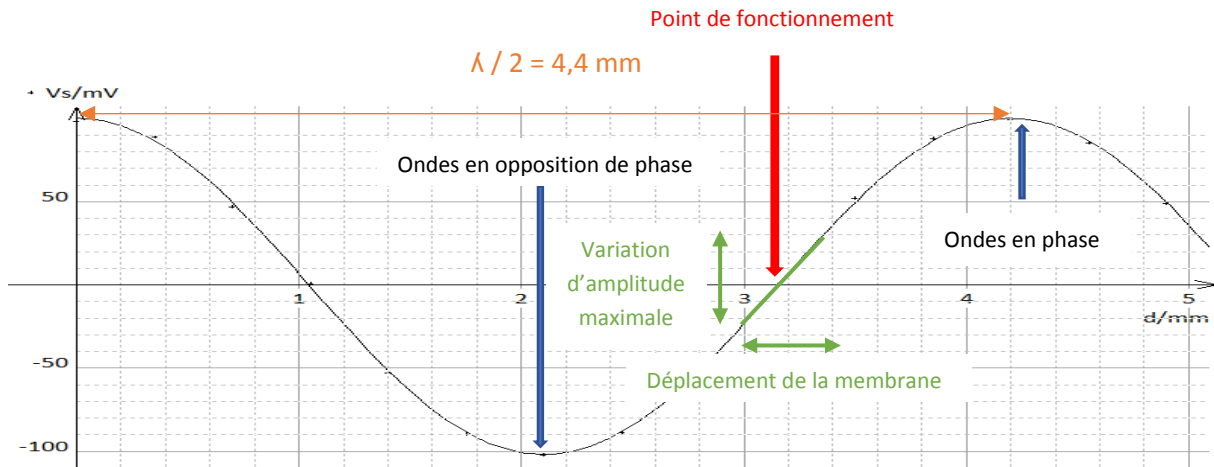
La fréquence f_c de coupure d'un circuit RC est égale à $\frac{1}{2\pi RC}$.

En choisissant $R = 1 \text{ k}\Omega$ et $C = 10 \text{ }\mu\text{F}$, on obtient $f_c = 16 \text{ Hz} \ll 80 \text{ kHz}$



4) Courbe de fonctionnement

On trace la valeur de la tension V_s en fonction du déplacement d de la membrane du haut-parleur. A chaque demi-tour de vis, le haut-parleur avance de 0,35 mm et la distance parcourue par les ultrasons entre l'émission et la réception augmente de 0,7 mm à cause de l'aller-retour. La période de la courbe de fonctionnement est donc égale à une demi longueur d'onde ultrasonore.

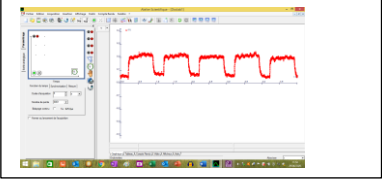
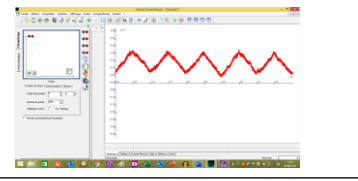
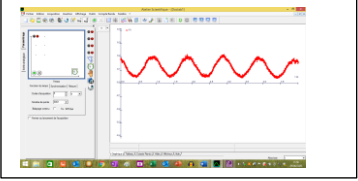


Le dispositif sera d'autant plus précis que nous détectons les battements autour de son point de fonctionnement correspondant à un signal émis en quadrature de phase avec le signal reçu ce qui correspond à un OFFSET nul ($V_s=0V$) et un déphasage de $\pi/2$.

QUATRIEME PARTIE : LA DETECTION DES BATTEMENTS

1) Détection des battements

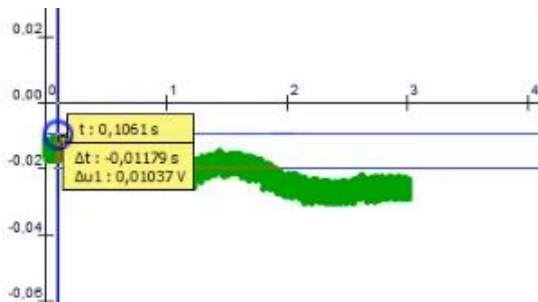
On place le simulateur devant nos détecteurs et on constate que non seulement les systèmes sont capables de détecter les battements du simulateur à distance mais ils permettent aussi de différencier les signaux ce qui nous laisse entrevoir la possibilité de reconnaissance par mesure de battements cardiaques à distance. En effet, les battements cardiaques sont une « signature » comme peuvent l'être nos empreintes digitales.

Signal en créneaux	Signal triangulaire	Signal sinusoïdal
		

De plus, la détection est toujours possible si on intercale un tissu entre le dispositif et le simulateur ce qui permettrait de détecter les battements à travers les vêtements.

2) La sensibilité

En l'absence de battements, on réalise une acquisition et on constate que le signal est bruité :



On mesure l'épaisseur du signal soit un bruit d'environ 10 mV.
Il faut que l'amplitude du signal soit supérieure au bruit pour pouvoir l'observer.
On sait qu'un tour de vis (0,7mm) correspond environ à une tension de 200 mV autour du point de fonctionnement.
On pourra donc détecter des battements d'amplitudes supérieurs à 1/20^{ème} de tour de vis soit 35 μ m.
Nos détecteurs peuvent donc détecter, en théorie, des battements de l'ordre de 35 μ m.

CONCLUSION

Nous sommes donc arrivés à la fin de notre objectif principal de ce projet, détecter un battement cardiaque à distance.

Nous avons en effet créé un simulateur ainsi qu'un détecteur principal, le déphaseur, qui a été testé sur nous-mêmes, à travers nos vêtements, et qui démontre donc que nos nombreuses heures de travail ne furent pas vaines.

A travers ce projet, nous avons découvert ce qu'était le monde de la recherche et les sentiments de réussite, mais aussi, il faut se l'avouer, de frustration qu'il procure.

Notre objectif principal atteint, nous pouvons dorénavant nous pencher sur les perspectives à venir de ce projet. Des perspectives qui font apparaître de nouvelles problématiques que nous prendrons plaisir à essayer de résoudre. Celles-ci tourneraient notamment autour des problèmes de miniaturisation, mais aussi de la résolution et de la sensibilité du signal afin de pouvoir en obtenir un plus fidèle aux mouvements du thorax. Cela passe entre autres par la capacité à ne pas être parasité par les mouvements et la respiration des personnes et ainsi être toujours au point de fonctionnement du dispositif. Nous pourrions également nous pencher sur un nouveau domaine d'utilisation car il faut savoir que chacun présente une empreinte cardiaque qui lui est propre et qui pourrait constituer une signature, du même ordre que l'empreinte digitale.

Enfin, nous tenons à remercier toutes les personnes qui nous ont aidé dans la résolution :

- *Notre professeur préféré de physique Mr Pichoud, qui a été disponible tout au long du projet et à toute heure du jour et de la nuit pour nous prodiguer des conseils sans lesquels nous n'aurions pas pu atteindre notre objectif.*
- *Mr Blot, notre super agent de laboratoire à l'enthousiasme et la bonne humeur communicative qui nous a permis de travailler dans les meilleures conditions.*
- *Mr Yoyotte, le proviseur de notre beau lycée, qui a mis une salle à notre disposition pour que nous puissions effectuer nos expériences en toute sérénité.*
- *Nous tenons aussi à adresser un très grand merci à Pablo Delamarche, notre ami et réalisateur de la vidéo qui a fourni un travail des plus rigoureux pour rendre celle-ci la plus qualitative possible.*
- *Mme Odier, notre professeure d'EPS qui nous a fourni les accessoires de secourisme.*

Programme arduino.py

```

from math import *
from tkinter import *

#trouve la moyenne d'un ensemble de valeurs
def moyenne(lst):return(sum(lst)/len(lst))

#trouve l'ecart type
def ecart_type(lst):
    m=moyenne(lst)
    ecart_carre=[(m-x)**2 for x in lst]
    return(sqrt(moyenne(ecart_carre)))

#convertis en valeurs PWM
def to_pwm(az):
    Min=min(az)
    az=[x-Min for x in az]
    Max=max(az)
    PWM=[x/Max*255 for x in az]
    PWMA=[round(x) for x in PWM]
    return(PWMA)

#tente de detecter la duree d'une periode
def find_p(lst):
    #on attribue a chaque valeur possible de la periode une note
    liste_entropique=[]
    for per in range(1,len(lst)):
        entropie=[]
        for start in range(per):
            actu=start
            points=[]
            while actu<len(lst):
                points.append(lst[actu])
                actu+=per
            entropie.append(ecart_type(points))
        entropie_moyenne=moyenne(entropie)
        liste_entropique.append(entropie_moyenne)

    #le signal trace par les notes en fonction de la periode presente une composante lineaire qu'on supprime
    derive=[liste_entropique[x+1]-liste_entropique[x] for x in range(len(liste_entropique)-1)]
    moyenne_derive=moyenne(derive)
    liste_bis=[liste_entropique[x]-x*1.5*moyenne_derive for x in range(len(liste_entropique))]

    #on retourne la periode qui a le meilleur score
    return(liste_bis.index(min(liste_bis)))

#fonction qui trace un signal
def draw(lst, d=100, z=1):
    last=[0, 0]
    for x in range(len(lst)):
        canvas.create_line(last, [x, lst[x]*z+d])
        last=[x, lst[x]*z+d]
    tk.update()

```

```

#preparation d'une fenetre de dessin
tk=Tk()
canvas=Canvas(tk, height=500, width=3000)
canvas.pack()

#recuperation du fichier CSV contenant les valeurs de l'acceleration en fonction du temps
fichier_acceleration=open("Acc.csv")
valeur_acceleration=fichier_acceleration.readlines()[1:]
fichier_acceleration.close()

#mise en forme et passage en PWM des valeurs de l'acceleration
lst_p=[]
for x in valeur_acceleration:lst_p.append(float(x.replace(",",".").split(";")[3]))
lst_p=to_pwm(lst_p)

#detection et affichage de la periode
print("patientez...")
a=find_p(lst_p)
print("periode detectée:", a)
draw(lst_p)
for x in range(len(lst_p)//a):
    canvas.create_line(x*a, 0, x*a, 2000, fill="red")
    tk.update()

#selection et recuperation de la periode choisie
select=int(input("veuillez selectionner une periode(voir fenetre tk), donner le numero de la periode :0,1,2..."))
resultat=lst_p[a*select:a*(select+1)]

#reecriture du fichier C++
reponse=input("reecrir le fichier c++? oui/non: ")
if reponse.lower()=="oui":
    path=__file__[:-10]+"pwm\\pwm.ino"
    arduino=open(path, "r")
    ancient_script=arduino.readlines()[2:]
    arduino.close()
    script="int tableau[]={"+str(resultat)[1:-1]+"};\nint l="+str(len(resultat))+";\n"+"".join(map(str, ancient_script))
    arduino=open(path, "w")
    arduino.write(script)
    arduino.close()

```

programme pwm.ino

```

int tableau[]={135, 128, 122, 128, 133, 143, 139, 139, 126, 130, 140, 169, 192, 206, 184, 145, 67, 55, 45, 58, 80, 109, 147,
176, 196, 201, 195, 185, 167, 154, 144, 133, 123, 119, 116, 114, 123, 136, 142, 159, 167, 177, 187, 194, 203, 201, 193, 170,
139, 105, 74, 61, 74, 89, 114, 132, 150, 162, 169, 179, 175, 169, 165, 152, 142, 133, 122, 123, 121, 121, 126, 133, 143, 148,
155, 154, 159, 156, 156, 146, 144, 142, 142, 139, 141, 138, 134, 130, 132, 137, 137, 143, 144, 145, 148, 150, 153, 147, 140,
138, 137, 141, 133, 132, 129, 134, 136, 145, 149, 151, 149, 150, 144, 149, 143, 142, 139, 137, 136, 137, 142};
int l=121;

void setup() {
    pinMode (3, OUTPUT);
}

void loop() {
    for (int i=0;i<l;i++)
        {analogWrite(3,tableau[i]);
        delay(10);
        }
}

```