Ecole Polytechnique Thiès

Du cycle préparatoire aux métiers des sciences de l'ingénieur

Traitement du Signal

N.F Ngom Mme Diop

Conception de mini

R A D A R

Réalisé par

Mame Diarra Sow

Ibrahima Birane Faye

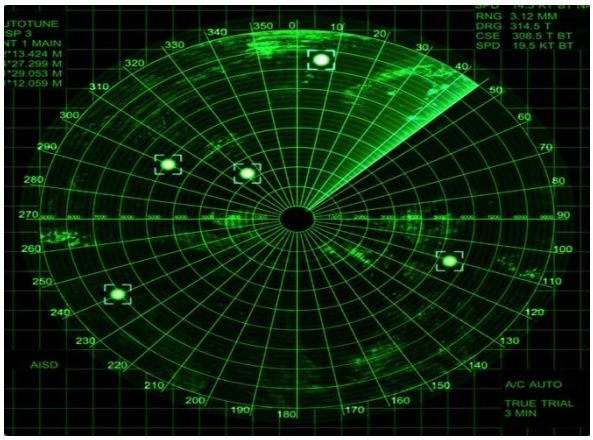
Pape Omar Diop

DESCRIPTION DU PROJET :

Le projet consiste à étudier un ultrason en faisant l'analyse de celui-ci, puis à concevoir un suiveur de position avec principalement un capteur ultrasonique et une carte Arduino, et enfin à concevoir un mini-radar avec presque le même matériel. On regroupe toutes ces fonctionnalités dans une interface Matlab.

QUELQUES NOTIONS :

Radar : Le Radar, acronyme anglais de (Radio Detection And Ranging) désigne un système qui diffuse une onde électromagnétique dans une portion de l'espace, et reçoit les ondes réfléchies par les objets qui s'y trouvent, permettant de détecter leur existence et déterminer certaines caractéristiques de ces objets tel que la position, l'altitude, la vitesse et parfois la forme de ces objets. Ces données permettent au Radar de renseigner l'utilisateur, mais aussi d'éliminer un grand nombre des objets indésirables pour ne conserver que les « cibles » intéressantes [2].



Ultra-son : L'ultra-son est une onde mécanique et élastique, qui se propage au travers de supports fluides, solides, gazeux ou liquides. La gamme de fréquences des ultrasons se situe entre 16 000 et 10 000 000 hertz, trop élevées pour être perçues par l'oreille humaine.

Senseur ultra-sonique :

Un capteur à ultrasons émet à intervalles réguliers de courtes impulsions sonores à haute fréquence. Ces impulsions se propagent dans l'air à la vitesse du son. Lorsqu'elles rencontrent un objet, elles se réfléchissent et reviennent sous forme d'écho au capteur. Celui-ci calcule alors la distance le séparant de la cible sur la base du temps écoulé entre l'émission du signal et la réception de l'écho.

Carte Arduino :

Une carte Arduino est une petite (5,33 x 6,85 cm) carte électronique équipée d'un micro-contrôleur. Le micro-contrôleur permet, à partir d'événements détectés par des capteurs, de programmer et commander des actionneurs ; la carte Arduino est donc une interface programmable.

Servo moteur : Un servomoteur est un système qui a pour but de produire un mouvement précis en réponse à une commande externe, C'est un actionneur (système produisant une action) qui mélange l'électronique, la mécanique et l'automatique.
Un servomoteur est composé :
- d'un moteur à courant continu
- d'un axe de rotation
- un capteur de position de l'angle d'orientation de
L'axe (très souvent un potentiomètre)
- une carte électronique pour le contrôle de la position de l'axe et le pilotage du moteur à courant continu

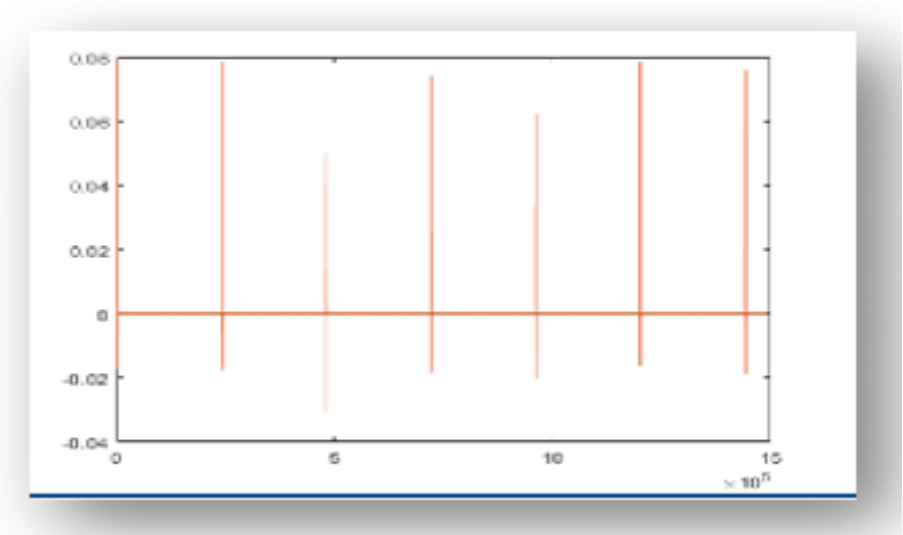
LES FONCTIONNALITES AVEC MATLAB :

L'analyse d'un ultrason :

Pour analyser l'ultrason nous avons d'abords représenté l'amplitude du signal en fonction du temps grâce au code qui suit :

```
[x,f]=audioread('C:\Users\ASUS\Desktop\ultrason.mp3
sound(x,f)
%la representation du signal
figure;plot(x);
```

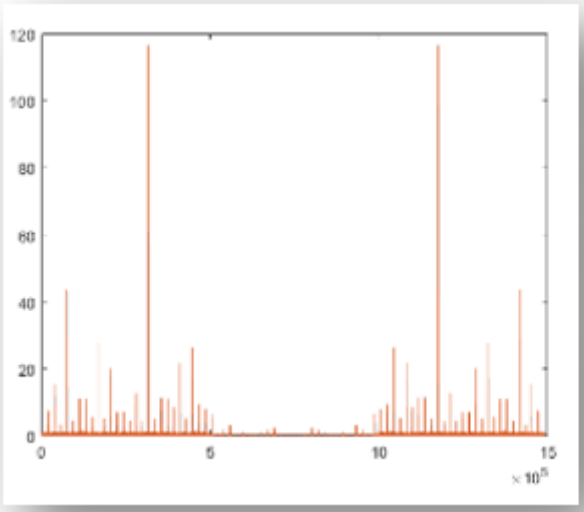
Et le rendu est illustré à la figure suivante :



Puis nous avons fait la représentation spectrale en utilisant la transformée de fourrier discrète FFT grâce au bout de code ci-dessous :

```
%la representation spectrale
figure;plot(abs(fft(x)));
```

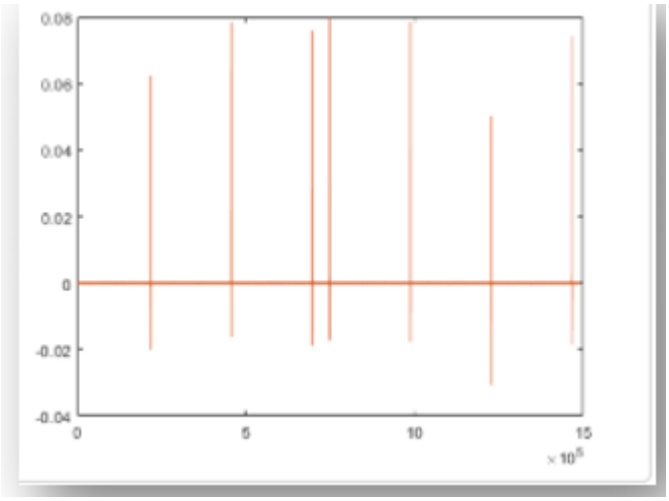
Et le rendu est illustré par la figure ci-dessous :



Nous avons également refait la représentation en centrant cette fois la fréquence nulle en utilisant FFTSHIFT grâce au bout de code qui suit :

```
figure;plot(fftshift(x));
```

Le rendu est illustré pal la figure suivante :



Comme les ultrasons ne sont perceptibles par l'oreille humaine on a fait une modulation de fréquence grâce au code qui suit :

```
%modulation de fréquence
fs = 1000;
fc = 200;
t = (0:1/fs)';
fDev = 50;
y = fmmod(x,fc,fs,fDev);
```

Fc=fréquence porteuse

Fs=fréquence d'échantillonnage du signal primaire

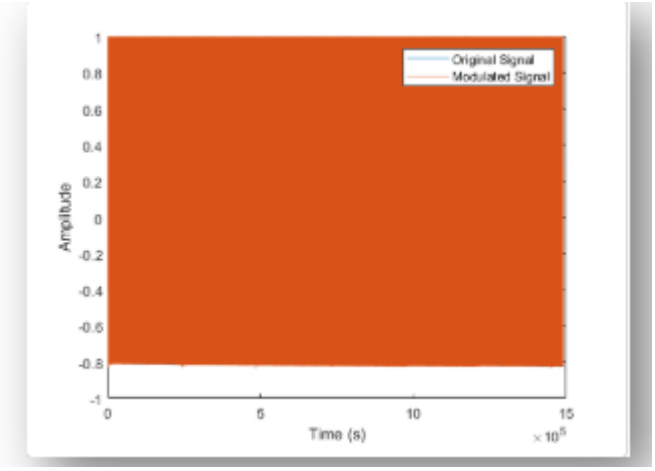
fDev=différence maximale entre la fréquence modulée et la fréquence nominale de la porteuse

y=fonction modulée

On a représenté la fréquence modulée grâce au bout de code qui suit :

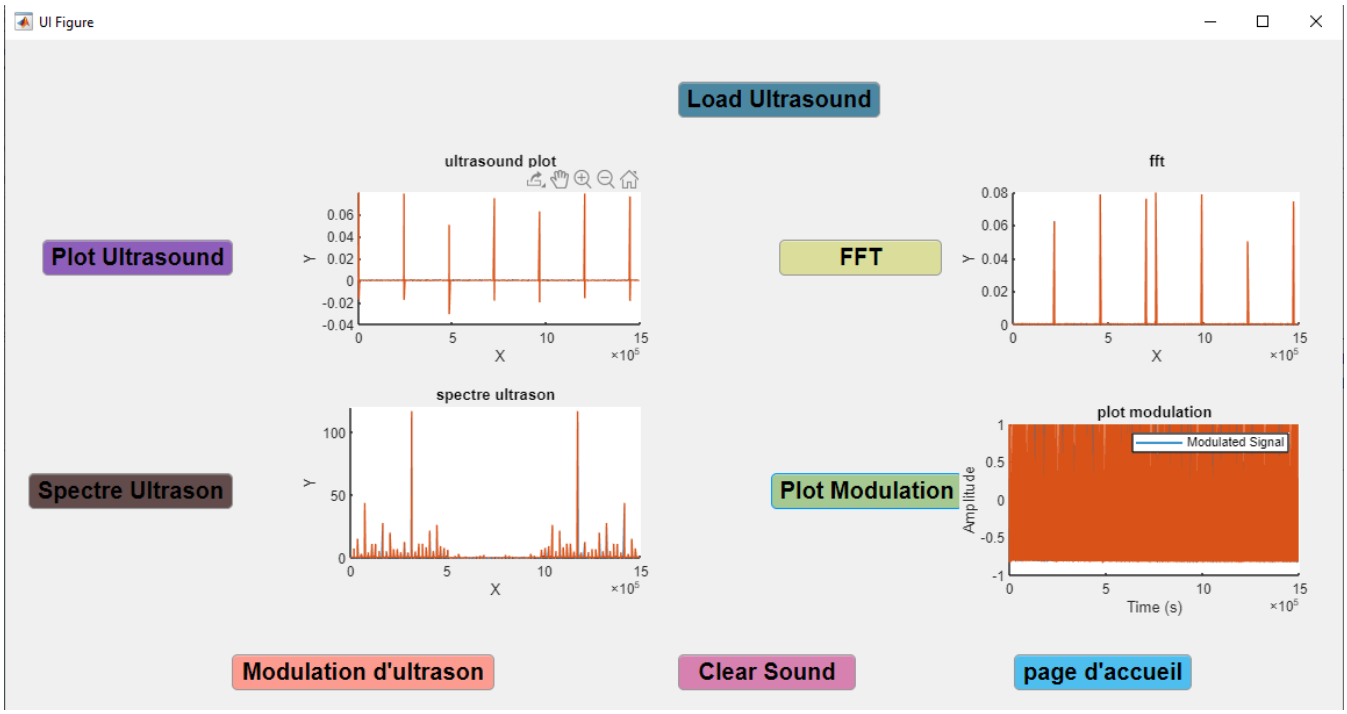
```
%tracer le  signal modulé
plot(y);
xlabel('Time (s)')
ylabel('Amplitude')
legend('Original Signal','Modulated Signal')
```

Après l'exécution nous avons le rendu suivant :



A la fin en écoutant le son de y avec le code sound (y, f) nous remarquons que le signal est devenu audible.

Ci-dessous nous avons un aperçu de l'interface d'analyse de l'ultrason :



Suiveur de position :

Montage : Pour faire le montage nous avons besoin :

-D'une carte Arduino

-D'un capteur ultrason hc-sr04

-D'un servo-moteur

-D'un ensemble de fils (jumpers)

-Une plaque d'essai (breadbord)

Le principe est simple, il consiste à affecter le câblage et a téléverser le code à l'aide du logiciel Arduino.

Comment ça fonctionne ? Le capteur émet un signal ultrasonore grâce au trigger puis récupère ce signal à partir de l'écho. La détection de l'obstacle se fait en calculant la distance grâce au temps de retour du signal

Après avoir détecté l'objet le servo moteur est déclenché et il commence à tourner suivant le sens de déplacement de l'objet détecté. Cette action est réalisée avec l'exécution du bout de code Arduino suivant :

```
if (distance <= 4) {
  // nothing detected
  detect = 0;
  // move left
  pos = pos + stepsize;
  myservo.write(pos);
  if (pos > 200){
    pos = 200;
  }
}
else {
  // something is detected
  detect = 1;
  // move right
  pos = pos - stepsize;
  myservo.write(pos);
  if (pos < 20) {
    pos = 20;
    }



  }
```

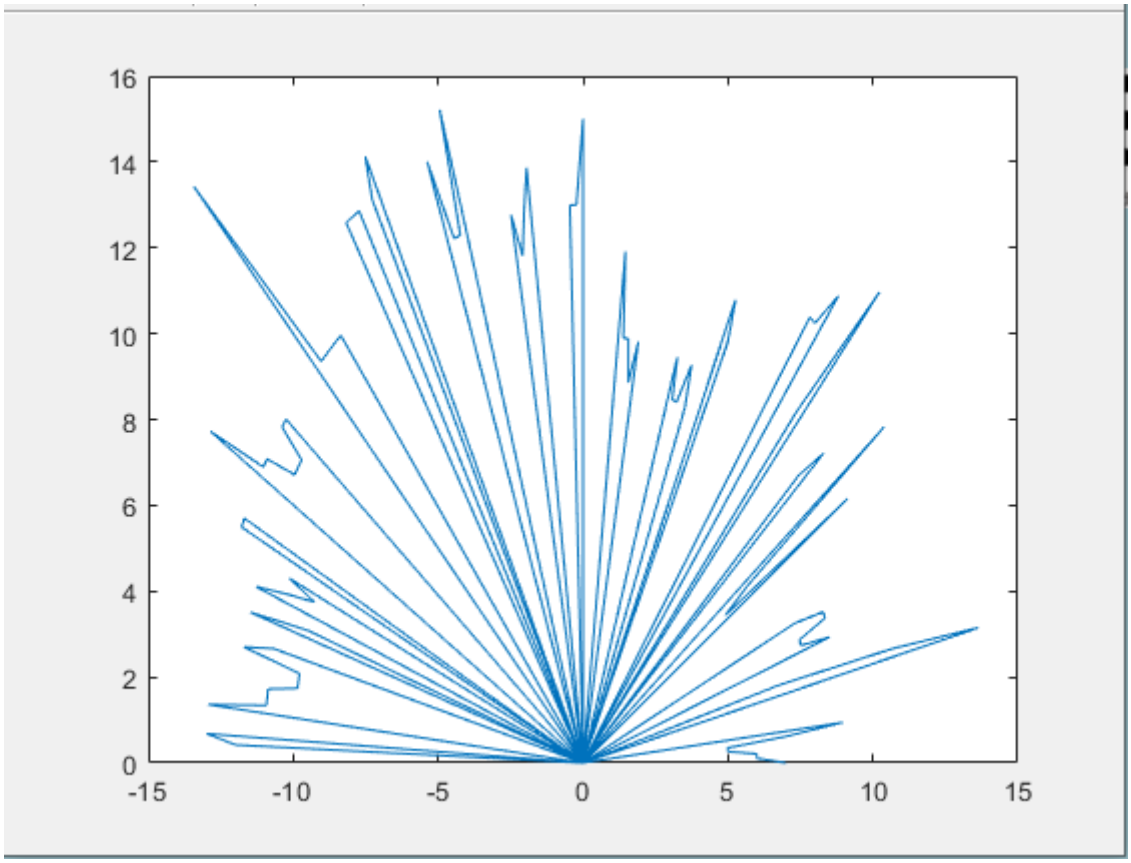
Et après grâce à Matlab nous avons réussi à ploter les différentes positions de l'objet en représentant les distances qui le sépare du suiveur à chaque instant grâce au bout de code qui suit

```
while i<180
    A = fscanf(s);
    num(i+1) = str2num(A);
    i = i+1;
end
fclose(s)

j = 1

while j<181
    tab(j,1) = (j-1)*inc
    tab(j,2) = num(j)
    tab(j,3) = num(j)*cosd((j-1)*inc)
    tab(j,4) = num(j)*sind((j-1)*inc)
    j = j+1
end
%figure
%polar(theta,num)

plot(tab(:,3),tab(:,4))
```

Grâce à la fonction println de Arduino on peut envoyer les variables notamment la distance au code Matlab qui le récupère grâce à la fonction fscanf. Mais il faut préalable lui donner le port du signal et le Bauderate qui renvoie au nombre de symbole par seconde comme illustré ci-dessous :

Le rendu est illustré dans la vidéo en pièce jointe.

Ci-dessous nous avons un aperçu de l'interface du suiveur de position

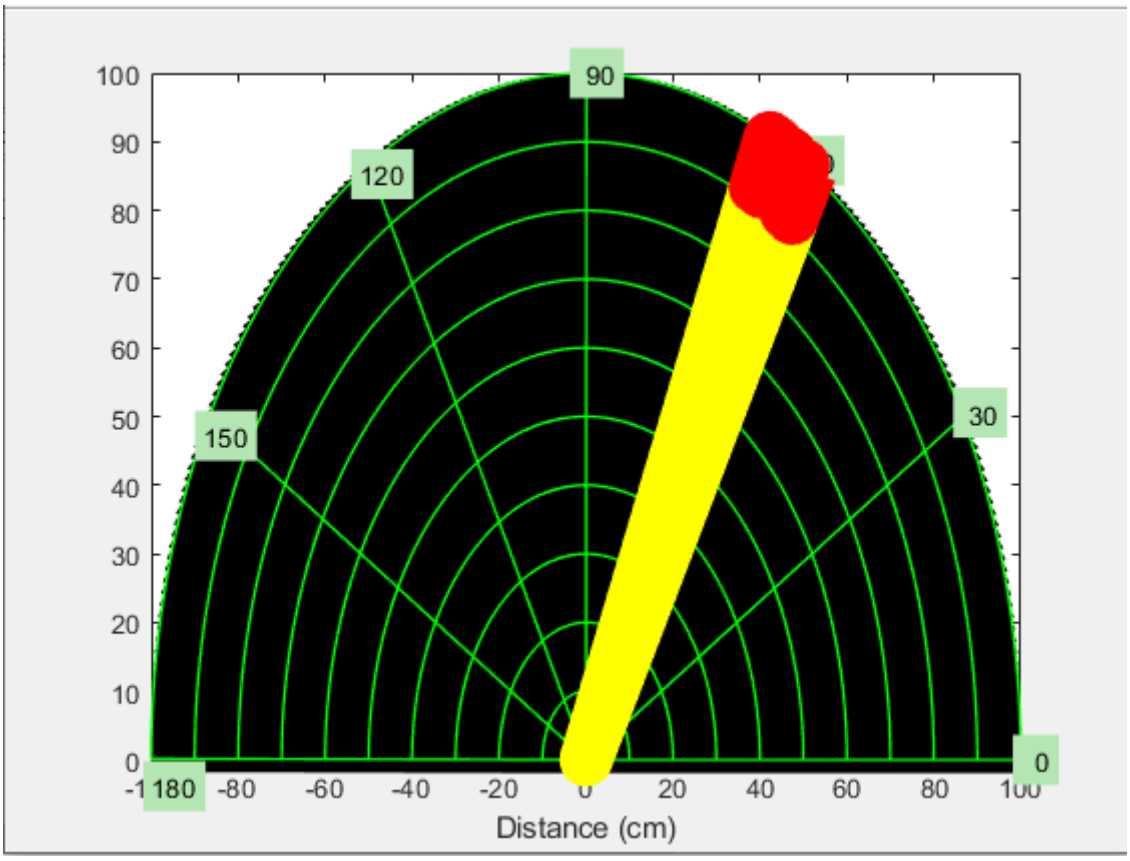


Mini-radar :

Le montage se fait avec les mêmes éléments que le suiveur de position et de la même manière sauf que le code à téléverser est différent. En effet nous avons téléverser un code qui fait tourner le servo-moteur même si rien n'est détecté et à chaque fois qu'un objet est détecté nous calculons la distance et renvoyer cette variable à Matlab grâce à println et la récupération se fait avec fscanf.

Une fois que la distance est récupérée nous allons la représenter sur le graphe de balayage du servo-moteur et l'effacer avec la fonction delete () avant de représenter une nouvelle distance.

Ci-dessous nous avons un aperçu de l'interface du radar :



Application python :

L'idée consiste à écrire un programme qui va coupler un capteur ultrason et la caméra du Raspberry Pi pour créer un système de détection de mouvement qui prendra une photo de la « cible » à chaque mouvement détecté.

Montage : Pour faire le montage nous avons besoin :

-Un Raspberry Pi

-La caméra Raspberry Pi

-Un capteur ultrason HC-SR04

-Une planche de prototypage (breadboard)

-Quelques fils M/F et M/M

- Des résistances

L’image ci-desssus correspond à l’apercu du code python une fois exécuter. Cet dernier va s’exécuter en boucle et calcule à chaque itération la distance et prend une photo s’il y a une différence majeure avec la distance prise antérieurement (last distance) . Ces images sont stockées dans le meme dossier source.