

Rapport de séance :

Durant cette 5^{ème} séance j'ai continué d'avancer sur le projet :

-Tout d'abord nous avons refait les tests du Bluetooth pour transmettre les informations au moteur que l'on doit faire marcher en fonction de la corde jouée, et cela fonctionne !

-Ensuite comme je l'avais expliqué, pendant ces vacances j'ai été chercher et discuter sur des forums de discussions afin de trouver des solutions pour le fréquencesmètre :

-J'ai alors entendu parler de la mesure de la fréquence instantanée par FFT. On m'a d'ailleurs conseillé de regarder une vidéo ([voir lien vidéo](#)) expliquant le principe, et dans laquelle on y retrouve un montage très simple (utilisant le module : DEVMO High Sensitivity Sound Detection), ainsi qu'un code plus complexe construit à partir de la librairie « arduinoFFT.h ». D'après ce que j'ai compris le but est de garder uniquement la première valeur de la fréquence captée une fois le son émis (donc le fondamental sans les harmoniques). J'ai donc réalisé le montage ([voir photo 1](#)) et récupéré le code en le modifiant légèrement : en effet dans la vidéo l'expérience consiste à capter la fréquence de différents diapasons donc le code ne s'exécute qu'une fois et la personne reset la carte UNO à chaque fois. J'ai donc modifié le code afin de capter plusieurs fréquences et non une seule (en effet dans le cadre de l'accordeur nous avons besoin de capter plusieurs fréquences pour savoir laquelle est la bonne). Le code est visible sur le GitHub en cliquant sur ce lien : [GU06/FFT_01.ino at main · 00marco00/GU06 · GitHub](#)

-Concernant l'efficacité de cette nouvelle solution :

Tout d'abord, c'est quand même beaucoup mieux qu'avant : en effet maintenant sur certaines cordes j'arrive à capter des fréquences presque constantes mais qui ne correspondent absolument pas aux fréquences de référence. En effet c'est notamment le cas pour les cordes de La, Sol, Si, Mi. Je capte le La joué en harmonique à 53Hz (au lieu de 110Hz joué à vide) ([voir photo 2](#)), le Sol à 97Hz joué en harmonique (au lieu de 196Hz joué à vide), le Si joué à vide à 62Hz (au lieu de 246Hz joué à vide) ([voir photo 3](#)), parfois le Mi aigu en harmonique à 164Hz (au lieu de 329Hz joué à vide) ([voir photo 4](#)). En revanche pour le Mi grave et le Ré j'ai du mal à capter des fréquences constantes. Ce qui me perturbe est que non seulement pour les cordes de La, Sol, Si, Mi les fréquences captées ne correspondent pas aux Fr. de référence mais en plus elles n'ont pas beaucoup de sens : En effet, comme expliqué dans le [lien 1](#), les fréquences harmoniques sont des multiples du fondamental ($F_n = n * F_1$ avec F_1 la Fr. du fondamental). Or moi quand je joue la première harmonique (12^{ème} case de la guitare), je divise la longueur de la corde par deux, donc en théorie je devrais avoir la Fr. du fondamental multipliée par deux, et ce n'est absolument pas le cas ! Par exemple pour le La je devrais avoir le fondamental fois deux, donc $110\text{Hz} * 2 = 220\text{Hz}$, or moi j'ai 53Hz (donc pas multiplié mais divisé par deux). Bref, je capte certes des Fr. constantes mais je ne comprends absolument pas leurs origines.

-De plus j'ai toujours des problèmes avec le module micro :

En effet en classe je croyais que le module captait trop loin car quand la guitare ne jouait pas je captais quand même des fréquences (c'est pourquoi j'avais demandé la mousse...). Or, en refaisant les tests chez moi dans le silence le plus total, je me suis aperçu que le micro capte quand même une fréquence comprise entre 160 et 170Hz au lieu de normalement 0Hz (vu que pas de bruit). J'imagine que cela est dû à la mauvaise qualité du module mais ça risque d'être embêtant, notamment lors de la démo. De plus j'imagine que cela peut expliquer les fréquences incohérentes que je capte avec la guitare.

-En conclusion pour la FFT : Cette méthode est meilleure car je capte des Fr. constantes pour certaines cordes et plus du tout de fréquences aberrantes (genre 1500Hz, 3000Hz...) mais je ne comprends pas les fréquences captées. De plus le module micro capte des fréquences « imaginaires » (quand il n'y a pas de bruit).

-J'ai enfin effectué le reste des soudures des moteurs pour que Marco puisse continuer d'avancer sur les modules moteur.

Remarque :

Après avoir discuté avec Ilane, nous avons décidé de ne pas imprimer la pièce de l'accordeur avec l'imprimante 3D. En effet, comme nous ne savons pas encore tous les modules qu'il faudra implémenter et à quelle place nous les fixerons à l'intérieur de la pièce, nous allons plutôt essayer de la faire en bois à l'aide de la découpe laser.

Liens consultés durant la séance :

<https://www.youtube.com/watch?v=wbeV0J30LGQ&t=160s> (lien vidéo)

<https://www.instinctguitare.com/comment-jouer-des-harmoniques-naturelles-a-la-guitare/#:~:text=Par%20exemple%2C%20sur%20la%20guitare%2C%20la%20fr%C3%A9quence%20de%2C%20330Hz%2C%20440Hz%2C%20550Hz%20et%20ainsi%20de%20suite> (lien 1)

Travail à faire lors de la prochaine séance :

Lors de la prochaine séance il faudra s'occuper des modélisations 3D pour la deuxième partie du projet (le module qui accorde automatiquement la guitare). Ensuite il faudra regarder comment faire avec la pièce de l'accordeur en bois avec la découpe laser. De plus on pourra peut-être mettre en commun mon FM avec le travail sur les moteurs de Marco et donc faire les premiers tests pour les cordes où j'arrive à avoir une fréquence constante.

Photo 1 : Le montage :

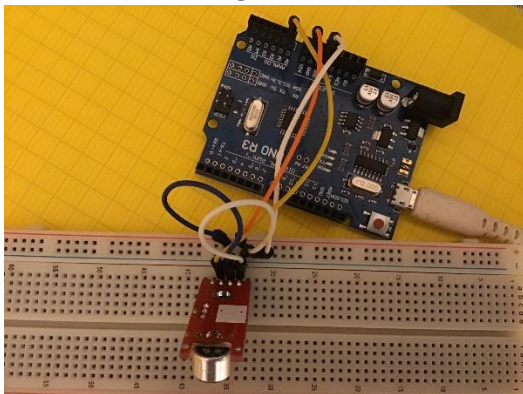


Photo 2 : Fr. captée pour le LA :

```
FFT_01
void setup()
{
  Serial.begin(115200);
  samplingPeriod = round(1000000*(1.0/SAMPLING_FREQUENCY));
}

void loop()
{
  for (int i = 0; i < SAMPLES; i++)
  {
    microseconds = micros();
    vReal[i] = analogRead(0);
    vImag[i] = 0;
  }
  /* Print the results of the simulated sampling according to the sampling period */
  FFT.Windowing(vReal, SAMPLES, FFT_WIN_TYP_HAMMING, FFT_FORWARD); /* Compute the FFT */
  FFT.Compute(vReal, vImag, SAMPLES, FFT_FORWARD); /* Compute the FFT */
  FFT.ComplexToMagnitude(vReal, vImag, SAMPLES); /* Compute the magnitude */
  double x = FFT.MajorPeak(vReal, SAMPLES, SAMPLING_FREQUENCY);
  Serial.println(x);
  //while(1); /* Run Once */
  delay(5000); /* Repeat after delay */
}
```

Photo 3 : Fr. captée pour le SI à vide :

FichierÉditionCroquisOutilsAide

Moniteur série

FFT_01

```
void setup()
{
  Serial.begin(115200);
  samplingPeriod = round(1000000*(1.0/SAMPLING_FREQUENCY));
}

void loop()
{
  for (int i = 0; i < SAMPLES; i++)
  {
    microseconds = micros();
    vReal[i] = analogRead(0);
    vImag[i] = 0;
  }
  /* Print the results of the simulated sampling according to the sampling period */
  FFT.Windowing(vReal, SAMPLES, FFT_WIN_TYP_HAMMING, FFT_FORWARD);
  FFT.Compute(vReal, vImag, SAMPLES, FFT_FORWARD); /* Compute the FFT */
  FFT.ComplexToMagnitude(vReal, vImag, SAMPLES); /* Compute the magnitude */
  double x = FFT.MajorPeak(vReal, SAMPLES, SAMPLING_FREQUENCY);
  Serial.println(x);
  //while(1); /* Run Once */
  delay(5000); /* Repeat after delay */
}
```

COM4

Envoyer

337.89
59.12
56.51
166.65
63.55
61.83
62.74
61.69
61.75
62.63
61.69
61.15
336.63
60.58
344.10
169.18

☐ Défilement automatique ☐ Afficher l'horodatage

Pas de fin de ligne 115200 baud Effacer la sortie

Photo 4 : Fr. captée pour le MI aigu :

FichierÉditionCroquisOutilsAide

Moniteur série

FFT_01

```
void setup()
{
  Serial.begin(115200);
  samplingPeriod = round(1000000*(1.0/SAMPLING_FREQUENCY));
}

void loop()
{
  for (int i = 0; i < SAMPLES; i++)
  {
    microseconds = micros();
    vReal[i] = analogRead(0);
    vImag[i] = 0;
  }
  /* Print the results of the simulated sampling according to the sampling period */
  FFT.Windowing(vReal, SAMPLES, FFT_WIN_TYP_HAMMING, FFT_FORWARD);
  FFT.Compute(vReal, vImag, SAMPLES, FFT_FORWARD); /* Compute the FFT */
  FFT.ComplexToMagnitude(vReal, vImag, SAMPLES); /* Compute the magnitude */
  double x = FFT.MajorPeak(vReal, SAMPLES, SAMPLING_FREQUENCY);
  Serial.println(x);
  //while(1); /* Run Once */
  delay(5000); /* Repeat after delay */
}
```

COM4

Envoyer

81.25
161.43
79.99
79.30
83.49
164.78
165.56
164.91
164.01
164.33
164.36
172.25
162.93
306.71

☒ Défilement automatique ☐ Afficher l'horodatage

Pas de fin de ligne 115200 baud Effacer la sortie