

PROJET ARDUINO

Rapport de séance du 9 décembre 2019 (séance 1)

Afin de préparer au mieux cette première séance, nous avons au préalable réalisé notre cahier des charges ainsi que le diagramme de Gantt. Après un premier échange avec M. MASSON durant la séance du 2 décembre 2019, nous avons établi et défini notre projet assez précisément même si quelques doutes subsistent encore sur certains détails notamment sur les fonctions des 2 petits bras articulés (exemple : capacité de saisir ou non des objets). C'est grâce à cet échange que nous avons finaliser notre idée de conception du projet.

Dès le début de la séance, nous avons récupéré l'ensemble du matériel nécessaire à la bonne réalisation de notre projet. Puis, nous avons réparti le travail que chacun devait effectuer durant la séance selon le diagramme de Gantt (même si cela ne correspondait pas totalement car nous avons finalement commencé par coder et non par monter notre voiture).

En ce qui me concerne, je me suis concentré sur la partie anticollision de notre voiture, c'est à dire qu'elle devra sans cesse mesurer la distance devant elle (et derrière elle) lorsqu'elle se déplacera, et s'arrêter si la distance qui la sépare de l'objet est égale à une distance définie comme la distance minimale pour laquelle la voiture s'arrête.

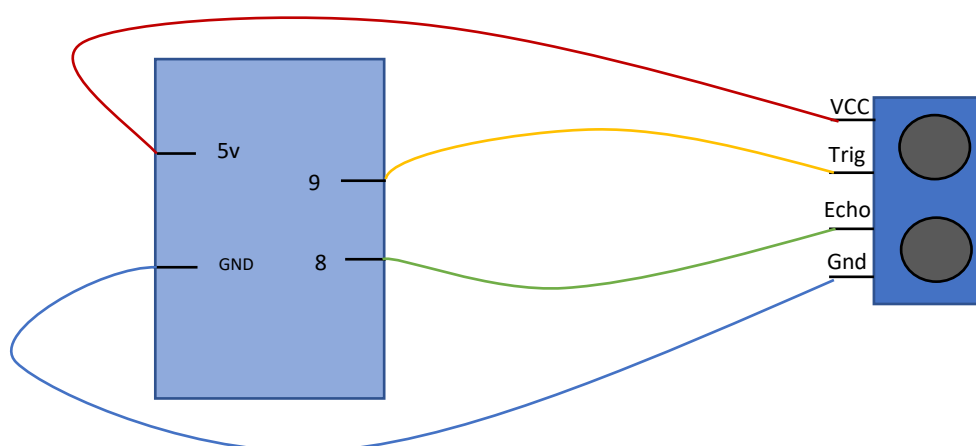
Mon but, pour cette séance, a donc été d'écrire le programme qui permet de calculer et afficher la distance qui va séparer la voiture d'un potentiel objet placé sur sa route.

Pour cela, il a fallu que j'utilise le module HC-SR04 qui est un capteur à ultrason permettant d'évaluer les distances entre un objet mobile et les obstacles rencontrés, d'où son utilisation. Il fonctionne de la manière suivante : il suffit d'envoyer une impulsion de 10µs en entrée et le capteur renvoie une largeur d'impulsion proportionnelle à la distance.

J'ai commencé par faire le montage correspondant (très simple) (je ne retrouve plus ma photo donc j'ai décidé de faire un schéma). Le matériel dont j'ai eu besoin est le suivant :

- Capteur HC-SR04
- Carte Arduino
- 4 fils (mâle vers femelle) pour câbler notre montage

Montage du module pour mesurer la distance voiture – potentiel objet



J'ai utilisé les fils (male vers femelle) afin d'éviter de faire montage sur la plaque que nous avons l'habitude d'utiliser en cours.

J'ai ensuite fait le code. Cela n'a pas été très difficile étant donné que nous avons vu en cours les mesures de distance.

Dans un premier temps, j'ai posé mes constantes pour la broche TRIGGER, pour la broche ECHO, ensuite, une variable *measure* dans laquelle je récupère le temps entre l'envoi de l'impulsion et son écho (que j'initialise à 0), de même pour *distance* qui récupère la distance à partir du temps dans *measure* ; puis, une constante *MEASURE_TIMEOUT* qui va servir de timeout pour la prise de mesure, ce timeout correspond au temps nécessaire avant de considérer qu'il n'y a pas d'obstacles (ici 25ms correspondant à 4m aller-retour pour 340m/s) et enfin, on définit la constante *SOUND_SPEED* pour définir la vitesse du son. Elle est en mm/μs car la mesure du temps se fait en microsecondes et car à la fin je veux une distance en mm.

```
const int trig =9; //broche TRIGGER
const int echo=8; //broche ECHO
long measure=0;
const long MEASURE_TIMEOUT=25000UL; //25ms = 8m à 340m/s
const float SOUND_SPEED = 340.0 / 1000; //Vitesse du son dans l'air en mm/us
float distance=0;
```

Puis, j'ai fait mon void setup (avec commentaires).

```
void setup() {
  //Initialisation des broches
  pinMode(trig,OUTPUT);
  digitalWrite(trig, LOW); //la broche TRIGGER doit être à LOW au repos
  pinMode(echo,INPUT);

  //Initialisation du port série
  Serial.begin(9600);
}
```

Et enfin, le void loop (avec commentaires).

```
void loop(){
  //Lance une mesure de distance en envoyant une impulsion HIGH de 10 μs sur la broche TRIGGER
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);

  //Mesure le temps entre l'envoi de l'impulsion ultrasonique et son echo (s'il existe)
  measure = pulseIn(echo, HIGH,MEASURE_TIMEOUT);

  //Calcul la distance à partir du temps mesuré
  distance = (measure/2.0)*SOUND_SPEED; // /2.0 car aller-retour

  //Délai d'attente pour éviter d'afficher trop de résultats à la seconde
  delay(500);

  //Afficher les résultats en mm
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println("mm");
}
```

CC : Je n'ai pas rencontré de difficultés particulières sur la tâche à effectuer, c'est pourquoi jusqu'à la fin du cours, j'ai tenté d'aider LU sur sa mission, celle de faire des recherches sur la mise en place d'une connexion Bluetooth entre la manette PS3 et la voiture (tâche beaucoup plus complexe avec laquelle nous rencontrons des problèmes). Me concernant, par la suite, je dois m'intéresser au laser.