

Rapport séance 3

Martelli Gino G1

J'ai fait le code et le branchement pour le sonar et j'ai continué à coder le programme pour le projet.

Sonar :

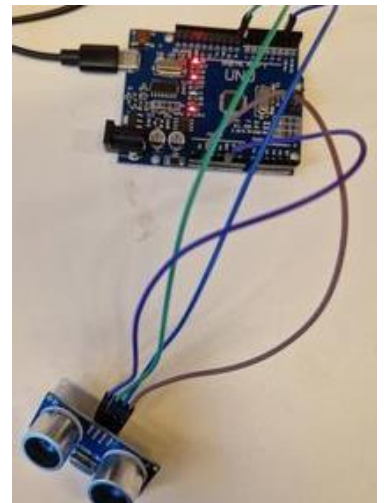
J'ai branché le sonar sur ma carte Arduino pour programmer et tester le code lié au sonar.

J'ai récupéré un programme d'exemple de la librairie NewPing qui donnait la distance `#include <NewPing.h>` mais

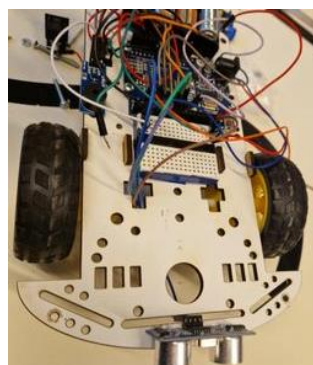
l'exemple ne fonctionnait plus lorsque je l'implémentais dans le programme principal.

J'ai donc récupéré du code d'un camarade qui avait un programme fait en cours qui donnait la distance sans avoir importé NewPing.

J'ai ensuite ajouté dans le programme une fonction qui teste la distance basée sur le code récupéré, et cette fois ci ça a fonctionné.



```
33 void loop() {
34     digitalWrite(trig, LOW);
35     delayMicroseconds(5);
36     digitalWrite(trig, HIGH);
37     delayMicroseconds(10);
38     digitalWrite(trig, LOW);
39     t = pulseIn(echo, HIGH);
40     d = (t/2)/29;
41     if (count==0) {
42         time=millis();
43     }
44     else {time=millis()+d*10;}
45     valeur(time);
46     /*Serial.println(d);*/
47     /*Serial.println(valeur(d));*/
48     if (n) {
49         digitalWrite(verte,HIGH);
50     }
51     else {
52         digitalWrite(verte,LOW);
```



J'ai ensuite branché le sonar sur le véhicule

Programmation :

J'ai continué le programme de la voiture : J'ai d'abord fait la structure de la boucle principale pour avoir un code mieux structuré.

```
49 void loop() {
50     if(turretPixy()){ //1
51         stopTurret(); //aju
52         while(!tryTase()){
53             moveToObject();
54         } // il se dirige v
55         moveAround(); // en
56     }
57 }
```

La fonction `turretPixy()` devra faire tourner la caméra Pixy sur une tourelle et renvoie true si elle détecte un objet rouge.

La fonction `stopTurret()` fait retourner la caméra Pixy à un angle de 0° avec la voiture puis fait tourner la voiture en direction de l'objet détecté.

La fonction `tryTase()` renvoie true si il a détecté un objet a moins de 5cm du sonar et qu'il l'a tase, false sinon.

La fonction `moveToObject()` fait avancer le véhicule jusqu'à l'objet. C'est la fonction `testPixy()` du rapport de la séance 2.

La fonction `moveAround()` fait déplacer la voiture autre part.

On a donc la tourelle qui, une fois un objet détecté, se redresse pour que la voiture se dirige vers l'objet. Après que l'objet à été tase, la voiture change d'endroit pour recommencer les recherches d'objets rouges avec la tourelle.

J'ai ensuite modifié le programme pour le simplifier mais j'ai gardé le même fonctionnement :

La fonction `turretPixy()` renvoie maintenant les coordonnées de l'objet détecté sur le champ de vision de la caméra Pixy, et ses coordonnées sont utilisées dans `stopTurret()` pour que la voiture se place en direction de l'objet. La fonction `TaseObject()` (`moveToObject()`) déplace la voiture jusqu'à l'objet et renvoie true si elle est à moins de 6 centimètres (après l'avoir tase), et false si elle perd l'objet ciblée de son champ de vision.

```
50 void loop() {
51     coord = turretPixy();
52     stopTurret(coord);
53     if(TaseObject()){ //
54         moveAround(); // ch
55 }
```

J'ai ensuite commencé à écrire les fonctions utilisées :

turretPixy() : La ligne 59 récupère les objets rouges. Tant qu'il n'y a pas d'objets détectés (ligne 60), la caméra continuera de tourner sur la tourelle (ligne 61). Après avoir fait un tour, le véhicule se déplacera autre part (ligne 62). Lorsqu'un objet est détecté, il retourne ses coordonnées horizontal (x) (ligne 64)

```
58 float turretPixy() { //la tourell
59     pixy.ccc.getBlocks(true, 1);
60     while(!pixy.ccc.numBlocks){
61         //tourne le servomoteur
62         //apres un tour entier la voi
63     }
64     return pixy.ccc.blocks[0].m_x;
65 }
```

stopTurret() : fera retourner la caméra Pixy à un angle de 0° avec la voiture puis fera tourner la voiture en direction de l'objet détecté.

```
67 void stopTurret(coord){
68     //ajuste la caméra au centre et tourne l
69 }
```

moveAround() : fera changer la voiture de place pour trouver d'autres objets

```
111 void moveAround(){
112     // change de place pour trouver un autre objet
113 }
```

tryTaser() : récupère la distance entre le sonar et l'objet en face de la voiture (ligne 72 à 77). Si l'objet est proche (moins de 6 centimètres, ligne 78), le véhicule taser l'objet (ligne 80) puis renvoie true (ligne 81), sinon renvoie false (ligne 83)

```
71 bool tryTaser(){
72     digitalWrite(trig,LOW);
73     delayMicroseconds(5);
74     digitalWrite(trig,HIGH);
75     delayMicroseconds(10);
76     digitalWrite(trig,LOW);
77     t = pulseIn(echo,HIGH);
78     if(t/58 < 6){
79         Serial.println("tase");
80         //tase();
81         return true;
82     }
83     return false;
84 }
```

TaseObject() : La ligne 89 récupère les objets rouges. Tant que la caméra à un objet rouge dans le champ de vision (ligne 90), le véhicule avance vers lui. Les lignes 91 à 103 ont été récupérées sur la fonction testPixy() du rapport de la séance 2.

S'il est proche de l'objet, il le tase puis renvoie true (ligne 104 et 105).

S'il perd l'objet de son champ de vision, il renvoie false (ligne 108) *.

```
88 bool TaseObject(){
89     pixy.ccc.getBlocks(true, 1);
90     while(pixy.ccc.numBlocks) {
91         pixy.setLamp(1,0); //Allume les 2 leds Pixy
92         if (pixy.ccc.blocks[0].m_x<150){
93             // Tourne le véhicule à gauche pour centrer l'objet
94             turnLeft();
95         }
96         else if (pixy.ccc.blocks[0].m_x>165){
97             // Tourne le véhicule à droite pour centrer l'objet
98             turnRight();
99         }
100         else {
101             //Le véhicule avance
102             forward();
103         }
104         if(tryTaser()){
105             return true;
106         }
107     }
108     return false;
109 }
```

*lorsqu'il renvoie false, le véhicule se remet en « mode turret » pour retrouver l'objet (dans la boucle principale loop())

La fonction tase() n'a pas été faite car il n'y a pas de taser pour l'instant.