

## 3 Principe d'un sonar

**Fichiers Arduino**  
 Programme à compléter  
 Fiche d'accompagnement  
[hatier-clic.fr/pc1332](http://hatier-clic.fr/pc1332)

Cette activité consiste à étudier le principe d'un sonar construit avec un microcontrôleur, à le configurer et à vérifier son bon fonctionnement.

La version présentée ici utilise :

- un microcontrôleur Arduino Uno ou équivalent ;
- un émetteur-récepteur à ultrasons HC-SR05 ou équivalent ;
- des fils et une platine de montage adaptés.

Aucune connaissance spécifique sur le fonctionnement du microcontrôleur ou du module à ultrasons n'est nécessaire pour faire l'activité, non plus que pour le traitement du programme.

### A. Montage

Le module HC-SR05 comporte cinq broches. Pour une utilisation en sonar avec Arduino Uno, réaliser les connexions suivantes.

Broche du HC-SR05		Fiche de l'Arduino Uno
VCC	Connectée à	5 V
Echo	Connectée à	13
Trig	Connectée à	12
Out	Non connectée	
GND	Connectée à	GND

### B. Programme

Le module à ultrasons contient de l'électronique qui fait que son principe de fonctionnement est le suivant.

- Passer la broche Trig à l'état HIGH pendant plus de 10  $\mu$ s déclenche l'envoi d'une succession de huit périodes d'un signal TTL (signal créneau entre 0 et 5 V) à 40 kHz sur l'émetteur à ultrasons.
- Simultanément, la broche Echo passe à l'état HIGH et le restera tant que le récepteur ne détectera pas d'onde retour. Une fois l'écho détecté, cette broche repasse à l'état LOW.

Le programme Arduino à réaliser doit donc effectuer les opérations suivantes.

- Définir et typer les grandeurs utilisées.
- Initialiser les états des différents ports.
- Pour chaque mesure :
  - o commander l'émission de la salve d'ultrasons ;
  - o mesurer le délai de réception ;
  - o calculer la distance entre l'émetteur-récepteur et l'obstacle ;
  - o afficher la valeur obtenue.

L'objectif de l'activité est, une fois le programme complété pour être opérationnel, de tester la validité de la mesure effectuée pour étalonner l'appareil de mesure. Voilà pourquoi le programme ci-dessous ne comporte aucun test sur les limites d'utilisation du capteur, vu que c'est précisément ce qui peut être déterminé au cours de l'activité.

```
// Définition des fiches émetteur et récepteur
const int emetteur=12;
const int recepteur=13;
// Typage des grandeurs duree et distance
float duree, distance;
// Valeur de la célérité en mètres par seconde
const float vitesse=...;

void setup() {
  Serial.begin(9600);
  // Initialisation des fonctions d'émission et réception
  pinMode(emetteur, OUTPUT); digitalWrite(emetteur, LOW);
}

void loop() {
  // Déclenchement de l'envoi de l'onde : émetteur en état haut...
  digitalWrite(emetteur, HIGH);
  // ... pendant 10 µs
  delayMicroseconds(10);
  // Fin du signal d'envoi de l'onde : émetteur en état bas
  digitalWrite(emetteur, LOW);
  // Durée d'aller-retour du son en microsecondes
  duree = pulseIn(recepteur, HIGH);
  // Calcul de la distance en millimètres
  distance = ...;
  // Écriture série de la distance mesurée
  Serial.print("distance = ");
  Serial.print(distance,2);
  Serial.println(" mm");
  // Attente avant la mesure suivante, en millisecondes
  delay(1000);
}
```

### Célérité (question 1a)

La célérité du son dans l'air à 20 °C peut être prise égale à 344 m·s<sup>-1</sup>.

### Calcul de la distance

Cette expression, à écrire à la question 1b peut être modifiée à la question 3b en fonction du résultat de l'étalonnage.

## C. Résultats

Les mesures effectuées sur 400 mm avec une règle graduée au millimètre comme étalon donnent le graphique ci-contre.

On modélise par une droite le lien entre la distance mesurée à la règle  $d_m$  et la distance déterminée à l'aide du sonar  $d_s$  de la manière suivante :

$$d_s = a d_m + b$$

avec  $a = 0,88$  et  $b = 13$  mm.

