



Étape 5 : Capteur à ultrasons

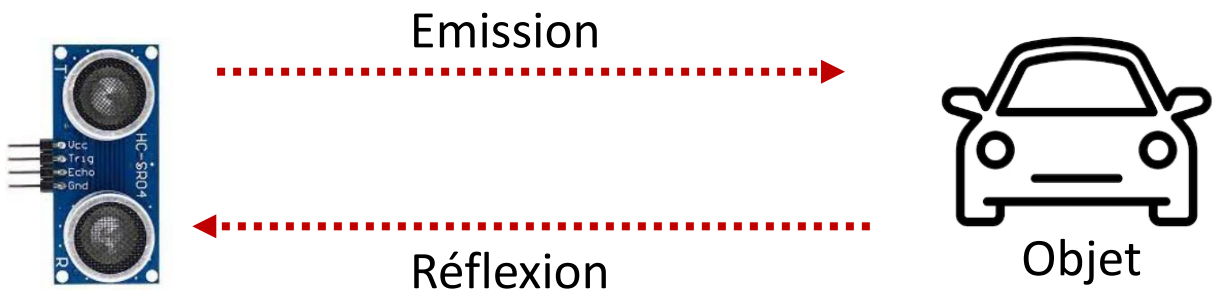


Objectifs:

- ❑ Comprendre le fonctionnement du capteur à ultrason ;
- ❑ Connecter un capteur à ultrasons à l'Arduino ;
- ❑ Créer un capteur de distance avec le capteur à ultrasons.



Comprendre le fonctionnement du capteur à ultrason



Un capteur à ultrasons est un appareil qui utilise des ondes ultrasonores (ondes inaudibles de fréquence supérieure à 20kHz) pour mesurer la distance entre lui-même et un objet. Il est composé d'un émetteur et d'un récepteur.

Voici comment cela fonctionne de manière très simple :

(1) Emission d'ondes : L'émetteur envoie un signal ondulatoire qui se propage dans l'air en direction d'un objet.

(2) Réflexion des ondes : Lorsque les ondes atteignent l'objet, elles se réfléchissent dessus, tout comme une balle rebondit sur un mur.

(3) Retour du signal : Le récepteur "écoute" le signal sonore rebondissant et mesure le temps entre l'émission et la réception. Connaissant la vitesse de l'onde dans l'air, cela nous permet d'accéder à la distance objet/capteur. Plus l'objet est proche, plus l'intervalle de temps mesuré est court.

Pour mesurer la distance, la formule est la suivante:

$$Distance = \frac{Vitesse \cdot temps}{2}$$

Dans l'air et dans des conditions usuelles, une bonne approximation de la vitesse du son est 343 m/s.

Le temps est mesuré par le capteur entre l'instant où il émet l'onde et l'instant où il reçoit la réflexion.

La distance peut alors être estimée.

? Question pratique : pourquoi y a-t-il un "2" dans la formule de la distance ?

Combien de temps va s'écouler entre l'émission et la réflexion d'un objet à 10 mètres de distance ?

À quelle distance est un objet pour lequel il y a 50 ms entre les instants d'émission et de réception ?

 Connecter un capteur à ultrasons avec l'Arduino
Créer un capteur de distance avec le capteur à ultrasons

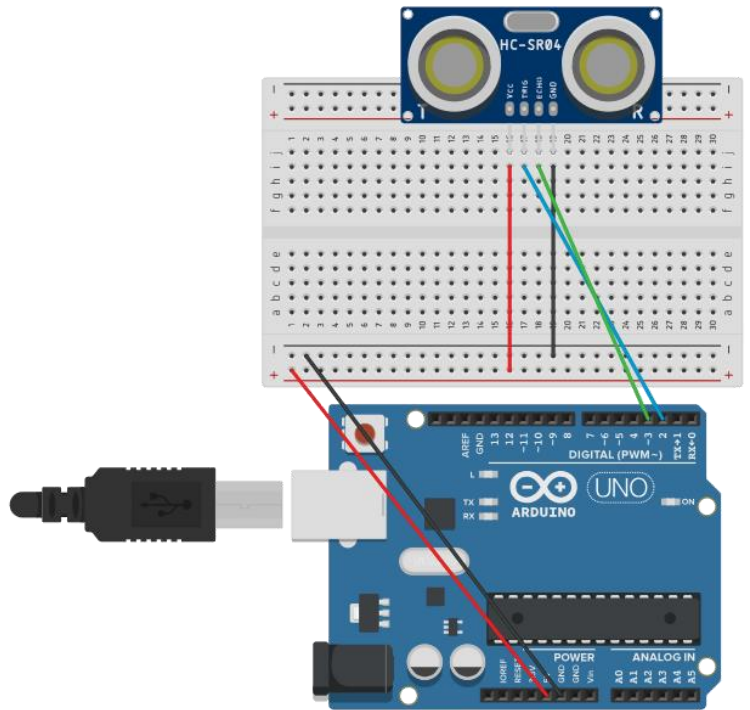


A – Entrée alimentation (5V)
B – TRIGGER (port numérique)
C – ECHO (port numérique)
D – Masse (0V)

Montage

On connecte les GND (pour Ground = Masse) à la même ligne (-).

On choisit aussi les ports de la carte reliés au trigger et à l'écho (ici, 3 et 2 respectivement), et on n'oublie pas d'alimenter le composant !



Pour utiliser un capteur à ultrasons, trois parties du code sont nécessaires :

(1) Déclaration: Le trigger et l'écho sont connectés à des ports, donc on va changer le nom des broches pour des noms plus intuitifs.

```
#define port_trigger 2  
#define port_echo 3
```

(2) Mode: Le trigger est une sortie et l'écho une entrée, donc on va les définir.

```
pinMode(port_trigger, OUTPUT)  
pinMode(port_echo, INPUT)
```

(3) Utilisation : Pour détecter la distance, il est nécessaire d'envoyer un signal au déclencheur (trigger) et de vérifier la réception d'un signal dans l'écho.

(3.1) Pour envoyer une onde, il faut changer l'état du déclencheur (trigger) de LOW à HIGH pendant 10 microsecondes.

(3.2) Pour vérifier le temps entre l'émission et la réception du signal, on peut utiliser la fonction **pulseIn(port_echo, HIGH)**.

? Exercice pratique: compléter le code ci-dessous avec les informations manquantes pour vérifier la distance d'un objet.

```
1. #define port_trigger 2
2. #define port_echo 3
3.
4. /* Vitesse du son en cm/us */
5. float vitesse_son = .....;
6.
7. void setup() {
8.     pinMode(port_trigger, OUTPUT); //l'émetteur est une sortie
9.     digitalWrite(port_trigger,LOW); // l'émetteur n'émet rien au début
10.    pinMode(port_echo,INPUT);      //le capteur est une entrée
11.    Serial.begin(9600);
12. }
13.
14. void loop() {
15.     digitalWrite(port_trigger,HIGH); //on allumé l'émetteur
16.     delayMicroseconds(10);          //pendant 10 microsecondes
17.     digitalWrite(port_trigger,LOW); //puis on l'éteint
18.     long measure = pulseIn(port_echo,HIGH); //on récupère le temps
19.     float distance_cm = .....;
20.     Serial.println(distance_cm);
21.     delay(500);
22. }
```

(1) Déclaration

(2) Modes

(3) Utilisation