

Programme de vérification du schéma

```
int trig = 2; // déclare une variable de type 'int' appelée trig et valant 2
int echo = 3; // déclare une variable de type 'int' appelée echo et valant 3
long lecture_echo; // Les variables de type long sont des variables de taille élargie pour le stockage de nombre entiers, sur 4 octets
long cm; // Les variables de type long sont des variables de taille élargie pour le stockage de nombre entiers, sur 4 octets

void setup() /*La fonction setup() est appelée au démarrage du programme. Cette fonction est utilisée pour initialiser les variables, le sens des broches, les librairies utilisées.
La fonction setup n'est exécutée qu'une seule fois, après chaque mise sous tension ou reset (réinitialisation) de la carte Arduino.
*/
{
  pinMode(trig, OUTPUT); /* Configure la broche Trig pour qu'elle se comporte en sortie.
broche: le numéro de la broche de la carte Arduino dont le mode de fonctionnement (entrée ou sortie) doit être défini.
mode: soit INPUT (entrée) ou OUTPUT (sortie).
*/
  digitalWrite(trig, LOW); // Met un niveau logique , LOW (BAS) sur la broche ou est connecter Trig.
  pinMode(echo, INPUT); // Configure la broche Echo pour qu'elle se comporte en entree.
  Serial.begin(9600); // ouvre le port série et fixe le débit de communication à 9600 bauds.
}

void loop() /*Après avoir créé une fonction setup(), qui initialise et fixe les valeurs de démarrage du programme, la fonction loop () (boucle en anglais)
fait exactement ce que son nom suggère
et s'exécute en boucle sans fin, permettant à votre programme de s'exécuter et de répondre. Utiliser cette
fonction pour contrôler activement la carte Arduino.
*/
{
  digitalWrite(trig, HIGH); // Met un niveau logique , HIGH (HAUT) sur la broche ou est connecter Trig.
  delayMicroseconds(10); //Stoppe le programme pendant 10 microsecondes . Il y a 1000 microsecondes dans une milliseconde, et un million de microsecondes dans une seconde.
  digitalWrite(trig, LOW); // Met un niveau logique , LOW (BAS) sur la broche ou est connecter Trig.
  lecture_echo = pulseIn(echo, HIGH); /* Lit la durée de l'impulsion niveau HAUT appliquée sur la broche echo qui est configurée en ENTREE.
L'instruction pulseIn() attend que la broche passe à HAUT, commence alors le chronométrage, attend que
la broche repasse au niveau BAS et stoppe alors le chronométrage.
L'instruction renvoie la durée de l'impulsion en microsecondes ( 1 millions de microsecondes par secondes).
L'instruction s'arrête et renvoie 0 si aucune impulsion n'est survenue dans un temps spécifié.
*/

  cm = lecture_echo / 58; // Calcul la distance en cm
  Serial.print("Distance en cm : "); //Affiche les données de "Distance en cm" sur le port série.
  Serial.println(cm); // affiche un retour de chariot et un saut de ligne.
  delay(1000); /*Stoppe le programme pendant 1000 microsecondes . Il y a 1000 microsecondes dans une milliseconde,
et un million de microsecondes dans une seconde.*/
  //
  |
  //
  }
}
```