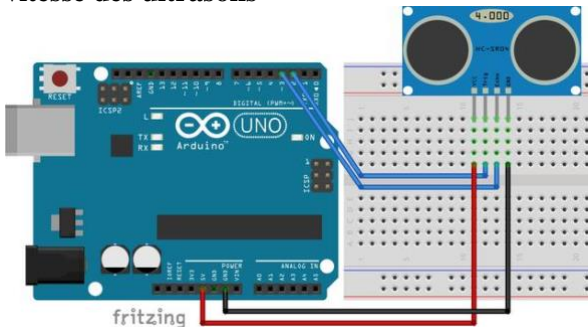
 Région académique GRAND EST	Usage des capteurs en physique chimie	
	Module	Acoustique
	Vitesse des ultrasons et mesure de distance	

Niveau	2 ^e professionnelle	CAP	1 ^{ère} ou T ^{le} professionnelle
Programme de formation	Acoustique : comment caractériser et exploiter un signal sonore ? Mettre en œuvre des émetteurs et des capteurs piézoélectriques.		Déterminer expérimentalement la vitesse de propagation d'un son dans l'air ou dans l'eau.

Objectif du TP	Etalonner une photorésistance pour réaliser un détecteur d'éclairement et un luxmètre.		
----------------	--	--	--

Données sur le composant	<p>Le capteur HC-SR04 est un capteur à ultrason. Il fonctionne avec une tension d'alimentation de 5 V, dispose d'un angle de mesure de 15° environ et permet de faire des mesures de distance entre 2 cm et 4 m avec une précision de 3 mm (en théorie, dans la pratique ce n'est pas tout à fait exact).</p> <p><i>N.B. Un capteur HC-SR04 revient à environ à 3-4 €. Il existe des capteurs à ultrason plus haut de gamme (donc plus précis) entre 20€ et 30€.</i></p> <p>Caractéristiques du composant : http://www.microsann.com/images/Atelier_Robotique/Documentation/Fiche_capteur_HC.pdf</p>		
--------------------------	---	--	--



Mise en œuvre du capteur	<p>Détermination de la vitesse des ultrasons</p>  <pre> VitesseUS // Déclare des variables de type "virgule flottante" float duree; float vitesse; // appelée duree et vitesse. int Pin_Trig=2; // Sélectionne la broche 2 pour être reliée à l'entrée Trigger du capteur ultrason HC-SR04. int Pin_Echo=3; // Sélectionne la broche 3 pour être reliée à la sortie Echo du HC-SR04. void setup() // Initialisation du programme { pinMode(Pin_Trig, OUTPUT); // Configure la broche Pin_Trig en sortie. pinMode(Pin_Echo, INPUT); // Configure la broche Pin_Echo en entrée. Serial.begin(9600); // Initialise la vitesse de communication entre la carte Plug'Uno et le PC 9600 bps. } void loop() // Boucle principale { digitalWrite(Pin_Trig, HIGH); delayMicroseconds(5); digitalWrite(Pin_Trig, LOW); // Une impulsion High (5 V) de 5 µs (le minimum est 2 µs) est envoyée sur la broche Trigger du capteur. // Le capteur génère alors une salve d'ultrasons à 40 kHz pendant 200µs (soit 8 périodes de 25 µs). duree = pulseIn(Pin_Echo, HIGH)*1e-6 ; // Le capteur attend le retour du signal réfléchi et lorsqu'il le détecte, la sortie echo envoie une impulsion de 5 V dont la durée // est égale à la durée d'un aller-retour de la salve. // L'impulsion mesure la durée (en µs) de l'impulsion (le chronométrage commence lorsque la broche Pin_Echo passe au niveau haut et s'arrête // lorsqu'elle repasse au niveau bas). vitesse = distance*2/duree ; // Calcule la vitesse qui correspond au rapport de la distance parcourue par l'onde (soit le double de la distance // entre le capteur et l'obstacle) par la durée d'un aller-retour. Serial.print("La vitesse est de "); Serial.print(vitesse); Serial.println(" m/s"); // Affiche la phrase entre guillemets. // Affiche la vitesse. // Affiche l'unité, et passe à la ligne. delay(1000); // Attend 1000 ms avant de recommencer. } </pre>		
	<p>Mesure d'une distance à l'aide des ultrasons</p> <pre> MesurededistanceavecUS // Valeur de la vitesse du son en m/s float vitesse = 330; float duree; float distance; int Pin_Trig=2; // Sélectionne la broche 2 pour être reliée à l'entrée Trigger du capteur ultrason HC-SR04. int Pin_Echo=3; // Sélectionne la broche 3 pour être reliée à la sortie Echo du HC-SR04. void setup() // Initialisation du programme { pinMode(Pin_Trig, OUTPUT); // Configure la broche Pin_Trig en sortie. pinMode(Pin_Echo, INPUT); // Configure la broche Pin_Echo en entrée. Serial.begin(9600); // Initialise la vitesse de communication entre la carte Plug'Uno et le PC 9600 bps. } void loop() // Boucle principale { digitalWrite(Pin_Trig, HIGH); delayMicroseconds(5); digitalWrite(Pin_Trig, LOW); // Une impulsion High (5 V) de 5 µs (le minimum est 2 µs) est envoyée sur la broche Trigger du capteur. // Le capteur génère alors une salve d'ultrasons à 40 kHz pendant 200µs (soit 8 périodes de 25 µs). duree = pulseIn(Pin_Echo, HIGH)*1e-6 ; // Calcule la distance. distance = vitesse*duree/2 ; Serial.print("La distance est de "); Serial.print(distance,3); Serial.println(" m"); // Attend 1000 ms avant de recommencer. delay(1000); } </pre>		

Commentaires			
Réinvestissement pédagogique			
Auteur(s)	E Ferry – JJ Kratz	Date de création	02 - 2020