



LCD_RADAR : Afficher la mesure d'un radar à ultra-son

Ce croquis est un peu plus compliqué puisqu'il mêle l'utilisation du radar à ultra-son à celle de l'écran à cristaux liquides. En même temps c'est un bon moyen de voir le résultat de son programme que d'utiliser un écran ...

On n'utilise pas de nouvelles bibliothèques dans ce croquis, mais uniquement celles qui sont nécessaires pour l'utilisation de l'écran LCD sur le bus I2C.



1. L'idée du croquis :

Dans son principe, le programme, dans la fonction `loop()`, prend une mesure de distance sur l'objet le plus proche, puis affiche sur la première ligne de l'écran un message qui varie selon la valeur de la mesure, et affiche sur la deuxième ligne la valeur de la mesure.

Si la valeur de la mesure est en dehors de l'intervalle (0 à 200cm) du radar, on allume la LED de la carte qui est reliée à la broche n°13 qu'on aura préalablement configurée en sortie dans la fonction `setup()`.

2. La mesure de la distance :

On désactive pendant 2ms la broche de l'émetteur puis on émet une salve d'ondes sonores pendant 10ms en activant l'émetteur avant de l'éteindre à nouveau. Immédiatement après on lit la valeur de la durée de l'aller et retour entre le radar l'objet.

La distance en cm est obtenue en divisant la durée mesurée par 58,2 ; la méthode est fournie par le fabricant du radar.



3. Le croquis :

```

/* *****
* LCD_RADAR : Mesure de la distance d'un objet par ultra son
*             affichage de la valeur de la distance
* ***** */
#include <Wire.h>           // bibliothèque I2C
#include <LiquidCrystal_I2C.h> // Bibliothèque LCD I2C

#define I2C_ADDR 0x27      // Adresse de l'écran sur le bus I2C

#define echoPin 7         // N° broche émetteur ultra-son
#define trigPin 8        // N° Broche recpteur ultra-son
#define LEDPin 13        // N° Broche LED de la carte

// déclaration des variables
int maximumRange = 200; // Maximum range needed
int minimumRange = 0;  // Minimum range needed
int loin = 80;         // Au delà de 80 cm c'est loin
int pres = 30;        // En deçà de 30 cm c'est très près
long duration, distance; // Duration used to calculate distance

// déclaration de l'objet écran à cristaux liquides I2C "lcd"
LiquidCrystal_I2C lcd(I2C_ADDR);

void setup()
{ pinMode(trigPin, OUTPUT); // Configuration en sortie de la broche de l'émetteur à ultra-son
  pinMode(echoPin, INPUT); // Configuration en entrée de la broche du récepteur à ultra-son
  pinMode(LEDPin, OUTPUT); // Configuration en sortie de la broche de la LED sur la carte
  lcd.begin(16,2);         // Initialisation de l'objet "lcd" 16 caractères x 2 lignes
  lcd.home ();            // Positionner le curseur au premier caractère de la première ligne
                           // après l'avoir effacé
  lcd.print("Hello, ARDUINO "); // afficher la chaîne de caractères "Hello, ARDUINO "
  delay(1000);            // Attendre une seconde (1000 ms)
}

void loop()
{ // Le cycle suivant trigPin/echoPin permet de déterminer la distance
  // de l'objet le plus près en lui envoyant des ondes sonores

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);

  // On Calcule la distance en cm en utilisant la vitesse du son (données du constructeur).
  distance = duration/58.2;

  if (distance >= maximumRange || distance <= minimumRange)
  {
    digitalWrite(LEDPin, HIGH); // Si la distance mesurée n'est pas comprise entre les bornes
    // On allume la LED de la carte
  }
  else
  { // Sinon
    digitalWrite(LEDPin, LOW); // On éteint la LED de la carte
    lcd.setCursor(0, 1); // On positionne le curseur de l'écran à la première colonne
                          // de la deuxième ligne
    lcd.print("Distance : "); // On affiche la chaîne de caractères "Distance : "
    lcd.print(distance); // On affiche la valeur mesurée de la distance
    lcd.print(" "); // On nettoie la fin de la ligne en affichant des espaces
    lcd.setCursor(0,0);
    if (distance > loin) // Si la distance est supérieure au contenu de la variable
                        // "loin"
      lcd.print("      :-[      "); // On affiche la chaîne de caractères "      :-[      "
    else // Sinon
      if (distance < pres) // Si la distance est inférieure au contenu de la
                          // variable "pres"
        lcd.print(":-] :-] :-] "); // On affiche la chaîne de caractères ":-] :-] :-]"
      else // Sinon (la distance est comprise entre "pres" et
          // "loin")
        lcd.print("      :-]      "); // On affiche la chaîne de caractères "      :-]      "
  }
  delay (150); // On attend 150 ms avant de recommencer
}

```