

```

/* Constantes pour les broches */
const byte TRIGGER_PIN = 2; // Broche TRIGGER
const byte ECHO_PIN = 3;    // Broche ECHO

/*servo*/
#include <Servo.h>
Servo monServomoteur;

/* Constantes pour le timeout */
const unsigned long MEASURE_TIMEOUT = 25000UL; // 25ms = ~8m Ã 340m/s

/* Vitesse du son dans l'air en mm/us */
const float SOUND_SPEED = 340.0 / 1000;

/*led*/
const int vertel = 7;
const int orangel = 8;
const int verte2 = 12;
const int orange2 = 13;

const int motor = 4;
/** Fonction setup() */
void setup() {

    /* Initialise le port sÃ©rie */
    Serial.begin(115200);

    monServomoteur.attach(4);

    /* Initialise les broches */
    pinMode(vertel, OUTPUT); //vertel est une broche de sortie
    pinMode(orangel, OUTPUT); //orangel est une broche de sortie
    pinMode(verte2, OUTPUT); //verte2 est une broche de sortie
    pinMode(orange2, OUTPUT); //orange2 est une broche de sortie
    pinMode(motor, OUTPUT); //motor est une broche de sortie
}

/** Fonction loop() */
void loop() {

    /* 1. Lance une mesure de distance en envoyant une impulsion HIGH de
    10Ãµs sur la broche TRIGGER */
    digitalWrite(TRIGGER_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIGGER_PIN, LOW);

    /* 2. Mesure le temps entre l'envoi de l'impulsion ultrasonique et
    son Ã©cho (si il existe) */
    long measure = pulseIn(ECHO_PIN, HIGH, MEASURE_TIMEOUT);

    /* 3. Calcul la distance Ã partir du temps mesurÃ© */

```

```

float distance_mm = measure / 2.0 * SOUND_SPEED;

/* Affiche les résultats en mm, cm et m */
Serial.print(F("Distance: "));
Serial.print(distance_mm);
Serial.print(F("mm "));
Serial.print(distance_mm / 10.0, 2);
Serial.print(F("cm, "));
Serial.print(distance_mm / 1000.0, 2);
Serial.println(F("m"));

if (distance_mm>50 && distance_mm<90){
    digitalWrite(verte1, HIGH); //allumer verte1
    digitalWrite(orange1, LOW); //eteindre orange1
    digitalWrite(verte2, LOW); //eteindre verte2
    digitalWrite(orange2, HIGH); //allumer orange2
    monServomoteur.write(180);
}

if ((distance_mm<50)|| (distance_mm>90 &&
distance_mm<250)|| (distance_mm>290)){
    digitalWrite(verte1, LOW); //eteindre verte1
    digitalWrite(orange1, HIGH); //allumer orange1
    digitalWrite(verte2, LOW); //eteindre verte2
    digitalWrite(orange2, HIGH); //allumer orange2
}

if (distance_mm>250 && distance_mm<290){
    digitalWrite(verte1, LOW); //eteindre verte1
    digitalWrite(orange1, HIGH); //allumer orange1
    digitalWrite(verte2, HIGH); //allumer verte2
    digitalWrite(orange2, LOW); //eteindre orange2
    monServomoteur.write(0);
}

/* Délai d'attente pour éviter d'afficher trop de résultats à la
seconde */
delay(100);
}

```