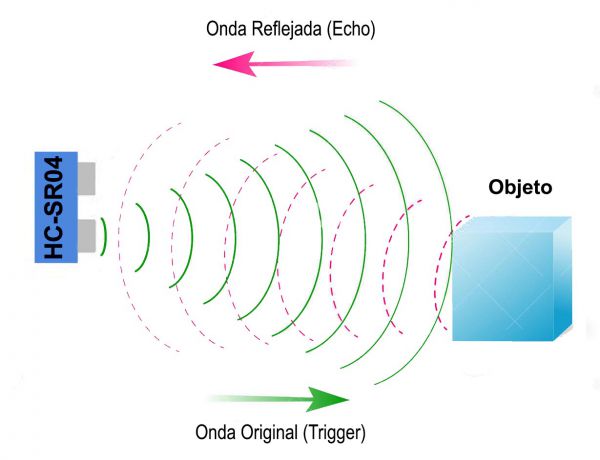
SENSOR DE ULTRASONIDOS HC-SR04

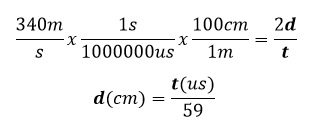
El funcionamiento consiste en emitir un sonido ultrasónico por uno de sus transductores, y esperar que el sonido rebote de algún objeto presente, donde el eco es captador por el segundo transductor. La distancia es proporcional al tiempo que demora en llegar el eco.



Midiendo el tiempo entre pulsos, conociendo la velocidad del sonido, podemos estimar la distancia del objeto contra cuya superficie impacto el impulso de ultrasonidos

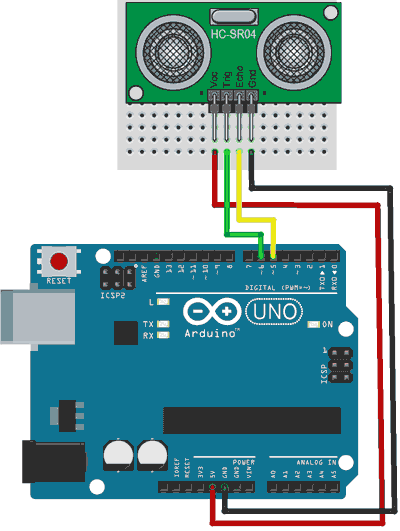
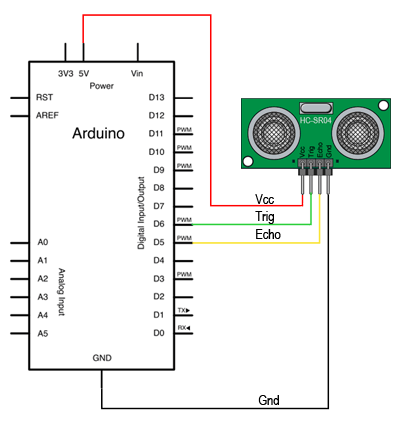
El rango de medición teórico del sensor HC-SR04 es de 2cm a 400 cm, con una resolución de 0.3cm.

El sensor se basa simplemente en medir el tiempo entre el envío y la recepción de un pulso sonoro. Sabemos que la velocidad del sonido es 343 m/s en condiciones de temperatura 20 ºC, 50% de humedad, presión atmosférica a nivel del mar. Transformando unidades resulta



Es decir, el sonido tarda 29,2 microsegundos en recorrer un centímetro. Por tanto, podemos obtener la distancia a partir del tiempo entre la emisión y recepción del pulso mediante la siguiente ecuación.

El motivo de divir por dos el tiempo (además de la velociad del sonido en las unidades apropiadas, que hemos calculado antes) es porque hemos medido el tiempo que tarda el pulso en ir y volver, por lo que la distancia recorrida por el pulso es el doble de la que queremos medir.



<https://es.aliexpress.com/wholesale?catId=0&initiative_id=SB_20210413122229&SearchText=SENSOR+DE+ULTRASONIDOS+HC-SR04>

const int Trigger = 2; //Pin digital 2 para el Trigger del sensor

const int Echo = 3; //Pin digital 3 para el Echo del sensor

void **setup**() {

**Serial**.begin(9600);//iniciailzamos la comunicación

  pinMode(Trigger, OUTPUT); //pin como salida

  pinMode(Echo, INPUT); //pin como entrada

  digitalWrite(Trigger, LOW);//Inicializamos el pin con 0

}

void **loop**()

{

  long t; //timepo que demora en llegar el eco

  long d; //distancia en centimetros

  digitalWrite(Trigger, HIGH);

  delayMicroseconds(10); //Enviamos un pulso de 10us

  digitalWrite(Trigger, LOW);

  t = pulseIn(Echo, HIGH); //obtenemos el ancho del pulso

  d = t/59;             //escalamos el tiempo a una distancia en cm

**Serial**.print("Distancia: ");

**Serial**.print(d); //Enviamos serialmente el valor de la distancia

**Serial**.print("cm");

**Serial**.println();

  delay(100); //Hacemos una pausa de 100ms

}

const int EchoPin = 5;

const int TriggerPin = 6;

void setup**()** **{**

Serial.begin**(**9600**)**;

pinMode**(**TriggerPin, OUTPUT**)**;

pinMode**(**EchoPin, INPUT**)**;

**}**

void loop**()** **{**

int cm = ping**(**TriggerPin, EchoPin**)**;

Serial.print**(**"Distancia: "**)**;

Serial.println**(**cm**)**;

delay**(**1000**)**;

**}**

int ping**(**int TriggerPin, int EchoPin**)** **{**

long duration, distanceCm;

digitalWrite**(**TriggerPin, LOW**)**; //para generar un pulso limpio ponemos a LOW 4us

delayMicroseconds**(**4**)**;

digitalWrite**(**TriggerPin, HIGH**)**; //generamos Trigger (disparo) de 10us

delayMicroseconds**(**10**)**;

digitalWrite**(**TriggerPin, LOW**)**;

duration = pulseIn**(**EchoPin, HIGH**)**; //medimos el tiempo entre pulsos, en microsegundos

distanceCm = duration \* 10 / 292/ 2; //convertimos a distancia, en cm

return distanceCm;

**}**