



UNIVERSIDAD DE GUANAJUATO

DIVISION DE INGENIERIAS

INFORMÁTICA
INDUSTRIAL

PROYECTO FINAL

***DETECCIÓN DE
OBJETOS A
DISTANCIA.***

JUAN ADOLFO GUTIERREZ GAYTAN

Introducción:

La ingeniería industrial (también ingeniería en gestión industrial o ingeniería en organización industrial) es una profesión de ingeniería que se ocupa de la optimización de procesos, sistemas u organizaciones complejos mediante el desarrollo, la mejora y la implementación de sistemas integrados de personas (recursos humanos), riqueza, conocimiento, información y equipamiento, energía, materiales y procesos.

¿Qué es un sensor HC-SR04?

El sensor HC-SR04 es un módulo que incorpora un par de transductores de ultrasonido que se utilizan de manera conjunta para determinar la distancia del sensor con un objeto colocado enfrente de este. Un transductor emite una “ráfaga” de ultrasonido y el otro capta el rebote de dicha onda. **El tiempo que tarda la onda sonora en ir y regresar a un objeto puede utilizarse para conocer la distancia que existe entre el origen del sonido y el objeto.**

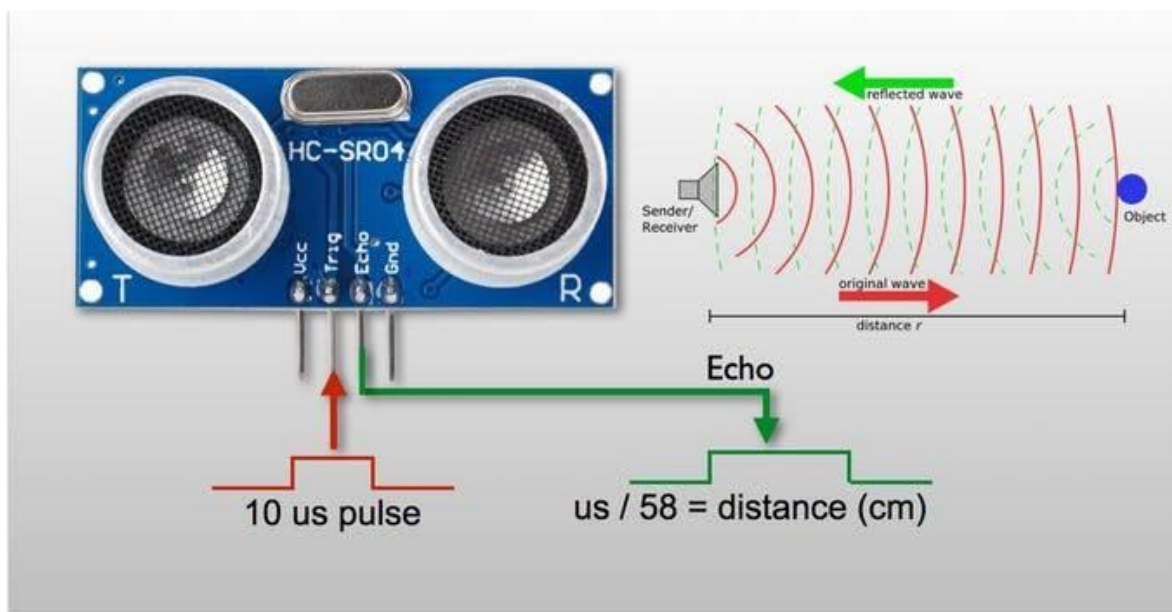


Imagen 1 – HC-SR04

En el mercado existen 2 tipos de módulos para medición de distancia por ultrasonido que se utilizan en robótica y aplicaciones similares:

- **Interfaz mediante pulso de eco:** Este tipo de sensores poseen un pin de disparo y otro de eco, algunos incluso combinan ambas funcionalidades en un solo pin. El

microcontrolador que desea realizar una medición debe medir el tiempo en alto de un pulso digital.

- **Interfaz serial (I2C o UART):** Este tipo de módulos entregan su medición en formato digital a través de una interfaz serial (I2C o UART). Habitualmente se acceden como si se tratara de una memoria serial I2C y podemos leer los registros donde se encuentran las mediciones de distancia.

¿Cómo se comunica un sensor HC-SR04?

La interfaz del sensor HC-SR04 y arduino se logra mediante 2 pines digitales: el pin de disparo (trigger) y eco (echo). La función de cada uno de estos pines es la siguiente:

- **El pin trigger** recibe un pulso de habilitación de parte del microcontrolador, mediante el cual se le indica al módulo que comience a realizar la medición de distancia.
- **En el pin echo** el sensor devuelve al microcontrolador un pulso cuyo ancho es proporcional al tiempo que tarda el sonido en viajar del transductor al obstáculo y luego de vuelta al módulo.

¿Cómo calcular la distancia con un sensor HC-SR04?

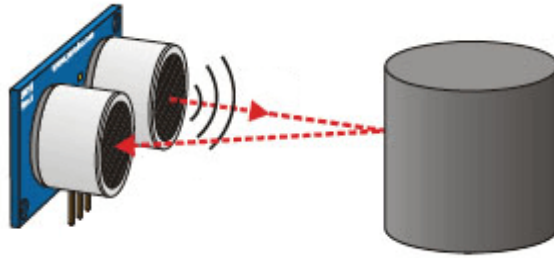
El sensor se basa simplemente en medir el tiempo entre el envío y la recepción de un pulso sonoro. Sabemos que la velocidad del sonido es 343 m/s en condiciones de temperatura 20°C, 50% de humedad, presión atmosférica a nivel del mar. Transformando unidades resulta:

$$343 \frac{m}{s} \cdot 100 \frac{cm}{m} \cdot \frac{1}{1000000} \frac{s}{\mu s} = \frac{1}{29.2} \frac{cm}{\mu s}$$

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.

$$Distancia(cm) = \frac{Tiempo(\mu s)}{29.2 \cdot 2}$$

El motivo de dividir por dos el tiempo (además de la velocidad 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.


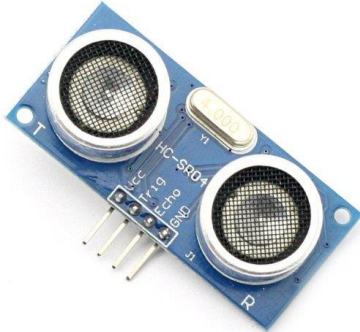







Ahora lo pasamos a código y que daría de la siguiente forma:

tiempo = (pulseIn(entrada, HIGH)/2);

*distancia = float(tiempo * 0.0343);*

Materiales

#	Nombre	Ilustración
1	Arduino Uno	
1	Sensor ultrasónico hc-sr04	

1	Piezo	
1	Pantalla LCD 16x2	
1	Potenciómetro 1k	
3	Diodos Led	
4	Resistencias 120 ohms	

Librerías utilizadas

Librería Liquid Crystal

Esta biblioteca permite a una placa Arduino controlar las pantallas LiquidCrystal (LCD) en base a la Hitachi HD44780 chipset (o compatible), que se encuentra en la mayoría de las pantallas LCD basadas en texto. La biblioteca trabaja tanto en el modo de 4 como en 8 bits.

Funciones de la librería

LiquidCrystal()

Descripción:

Crea una variable de tipo LiquidCrystal. La pantalla se puede controlar por medio de 4 u 8 líneas de datos. En el primer caso, omitir los números de pines D0 a D3 y dejar esas líneas sin conectar. El pin RW puede estar conectado a tierra en lugar de conectarse a un pin en el Arduino; si es así, omite de los parámetros de esta función.

begin()

Descripción:

Inicializa la interfaz de la pantalla LCD, y especifica las dimensiones (anchura y altura) de la pantalla. `begin ()` debe ser llamado antes de cualquier otro comando de la biblioteca LCD.

clear()

Descripción:

Borra la pantalla LCD y posiciona el cursor en la esquina superior izquierda.

setCursor()

Descripción:

Coloca el cursor del LCD; es decir, establece la ubicación en la que se mostrará el texto escrito a continuación en la LCD.

print()

Descripción:

Imprime un texto en la LCD

Funciones predefinidas utilizadas.

pulsein()

Descripción:

Lee un pulso (HIGH o LOW) en un pin. Por ejemplo, si el valor es alto, pulseIn () espera a que el pin pase a nivel HIGH, se inicia el tiempo, espera a que el pin pase a nivel LOW y para el cronómetro. Devuelve la longitud del impulso en microsegundos o 0 si no se recibe un pulso completo dentro del tiempo de espera.

La temporización de esta función se ha determinado de forma empírica y probablemente mostrará errores en los pulsos más cortos. Funciona en pulsos de 10 microsegundos a 3 minutos de duración. Tenga en cuenta también que si el pin ya está a nivel HIGH cuando la función es llamada, se espera que el pin pase a nivel LOW y luego a nivel HIGH antes de que empiece a contar. Esta rutina se puede utilizar sólo si se activan las interrupciones. Por otra parte, la resolución más alta se obtiene con intervalos cortos.

Esquema.

