

Arduino: sensor ultrasónico HC-SR04

(/tecno3eso/teoria/robotica/27-hcsr04)

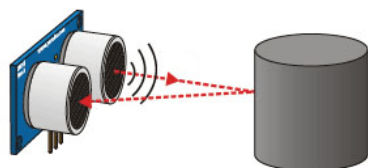
tecnología (/component/tags/tag/tecnologia)

programación (/component/tags/tag/programacion)

robótica (/component/tags/tag/robotica)

Introducción

En este tutorial explicaremos como utilizar un **sensor ultrasónico HC-SR04** para con ayuda de **Arduino** medir distancias y, de esta manera, conseguir que nuestros mecanismos sean capaces de detectar obstáculos. El sensor consta de un **terminal emisor** que emite un **ultrasonido** (inaudible para el oído humano por su alta frecuencia) y **espera un determinado tiempo con el terminal receptor la vuelta de la onda sonora**. Si no detecta nada es debido a que no hay ningún obstáculo en el rango de medidas del sensor (este sensor en concreto tiene un rango de distancias sensible entre 3cm y 3m con una precisión de 3mm) mientras que si recibe la respuesta usa el hecho de que el sonido lleva un movimiento rectilíneo uniforme para, a partir del tiempo que pasa desde que emite hasta que recibe la señal, realice de forma sencilla el cálculo de la distancia a la que se encuentra el objeto.



$$\text{Tiempo} = 2 \cdot (\text{Distancia} / \text{Velocidad})$$

$$\text{Distancia} = \text{Tiempo} \cdot \text{Velocidad} / 2$$



Explicación teorica

En este caso usaremos las siguientes características físicas de las ondas sonoras:

1. Se propagan en el aire siguiendo una **trayectoria rectilínea con velocidad constante (MRU)** con lo que podemos usar las ecuaciones del movimiento rectilíneo para hallar el **espacio que recorre la onda** (y así deducir la distancia a la que se encuentra el objeto) a partir de la medida del tiempo que tarda la onda desde que es emitida hasta que es recibida (ya que la velocidad es constante).
2. La **velocidad del sonido** en el aire podemos considerarla constante y de valor aproximado $v_{\text{SONIDO}} = 340 \text{ m/s}$. Como Arduino usa como sistema de medida del tiempo los **microsegundos (μs)** y teniendo en cuenta que **1 segundo = 1000000 μs = $10^6 \mu\text{s}$** , debemos cambiar de unidades la velocidad para ajustarnos a las necesidades de cálculo de Arduino:

$$v_{\text{SONIDO}} = 340 \text{ m/s} \rightarrow v_{\text{SONIDO}} = 340 \text{ m}/10^6 \mu\text{s} \rightarrow v_{\text{SONIDO}} = 0,00034 \text{ m}/\mu\text{s} \rightarrow v_{\text{SONIDO}} = 3,4 \cdot 10^{-4} \text{ m}/\mu\text{s}$$

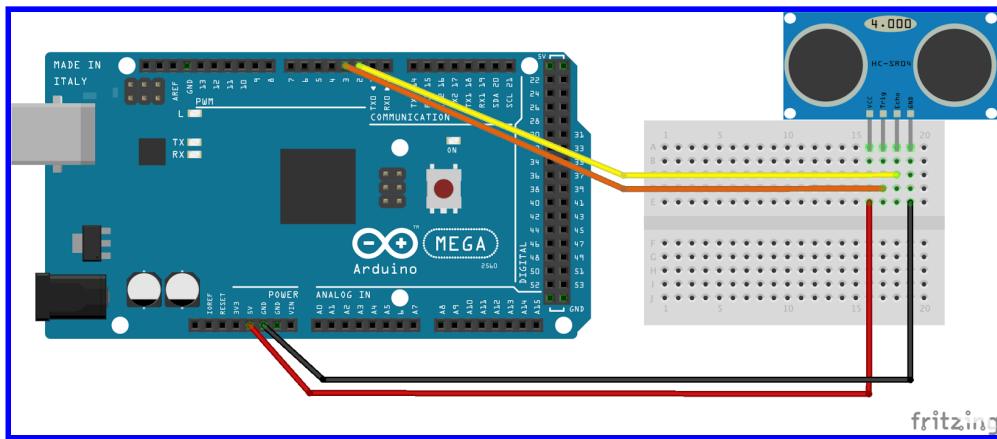
Pero si usamos esta expresión con Arduino nos daría una medida de la distancia del obstáculo en metros y no tiene demasiado sentido ya que el rango máximo al que puede trabajar este sensor es apenas de 3 m. Por tanto, los centímetros resultan más adecuados. Dado que **1 m = 100 cm = 10^2 cm** , con lo que obtendremos la expresión final que usamos en nuestro código:

$$v_{\text{SONIDO}} = 3,4 \cdot 10^{-4} \text{ m}/\mu\text{s} \rightarrow v_{\text{SONIDO}} = 3,4 \cdot 10^{-4} \cdot 10^2 \text{ cm}/\mu\text{s} \rightarrow v_{\text{SONIDO}} = 3,4 \cdot 10^{-2} \text{ cm}/\mu\text{s} \rightarrow v_{\text{SONIDO}} = 0,034 \text{ cm}/\mu\text{s}$$

3. Una vez que conocemos que se trata de un **MRU** podemos aplicar la expresión: **$e = v \cdot t$** (donde **e** es el **espacio recorrido** por la onda, **v** su **velocidad** y **t** el **tiempo** desde que sale del emisor hasta que se recibe en el receptor). Para concluir esta rápida explicación hay que tener en cuenta que si usamos esta expresión estamos calculando la distancia que recorre la onda sonora. Pero en este caso **esa distancia debe ser dividida entre 2** (para conocer la distancia a la que se encuentra el objeto) ya que el tiempo que se usa es el tiempo de ida más el tiempo de vuelta.

Circuito eléctrico (diagrama de conexiones)

En este caso realizaremos el sencillísimo diagrama del circuito usando el software para **diseño gráfico de circuitos Fritzing** [🔗](https://fritzing.org/home/) (<https://fritzing.org/home/>). En este caso sólo se necesita el sensor HC-SR04, la controladora Arduino y una serie de cables de conexión (aunque en el ejemplo que se muestra se usará la **placa de pruebas, protoboard o breadboard** [🔗](https://es.wikipedia.org/wiki/Placa_de_pruebas) (https://es.wikipedia.org/wiki/Placa_de_pruebas)). El diagrama eléctrico es muy sencillo ya que la primera patilla irá a **5 V (VCC)** mientras que la última irá a **tierra (GND)**, la patilla del **emisor o trigger** se conectará a un pin digital (en nuestro caso el pin 3) y el **receptor o echo** se se conectará a un pin digital (en nuestro caso el pin 2)



Código programa Arduino

En este primer ejemplo usaremos un código muy sencillo que **no usa librerías externas** y que solo usa las funciones nativas de la IDE de Arduino y dejaremos el uso de programas más complejos para siguientes ocasiones.