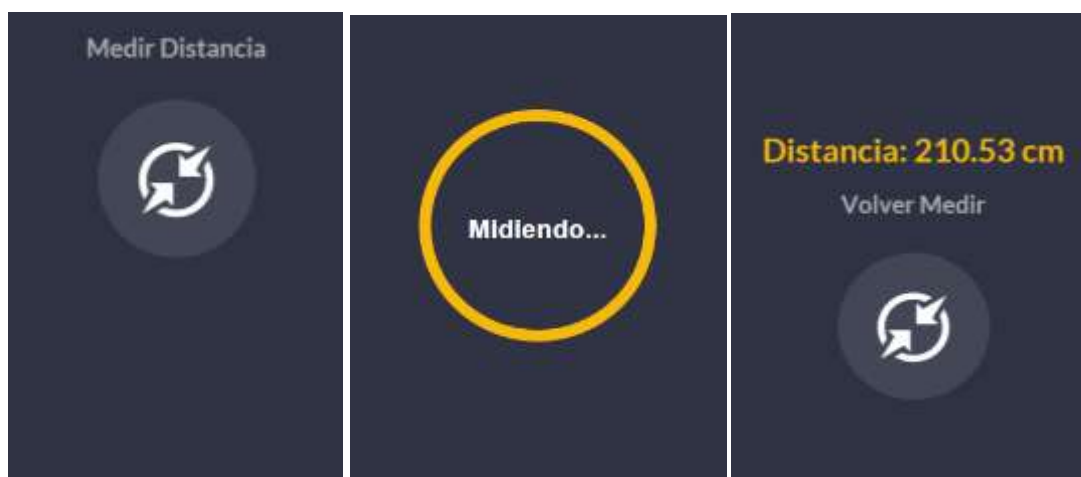
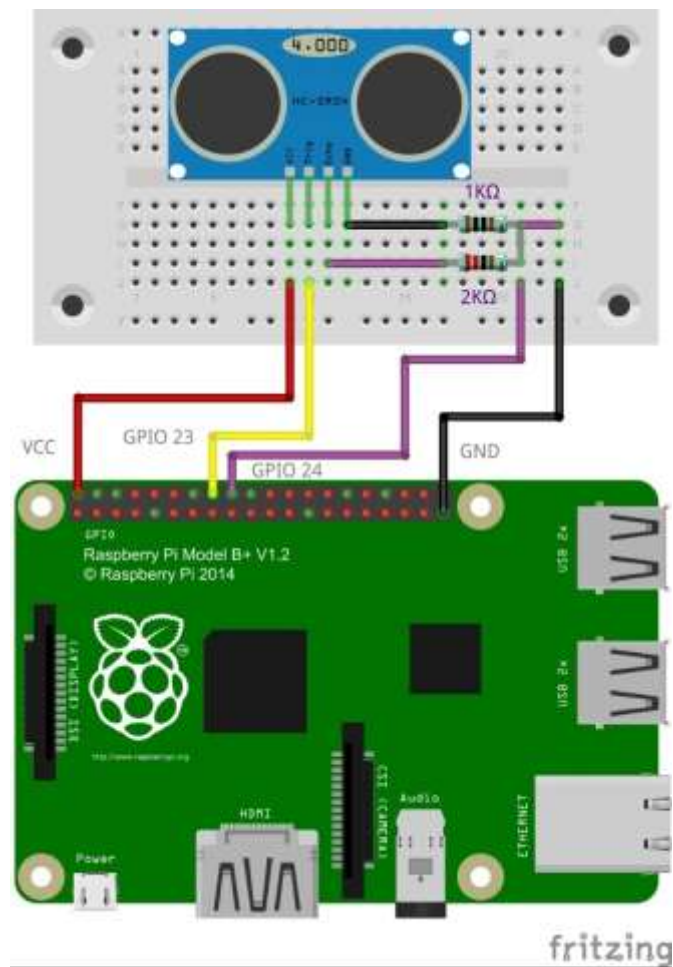


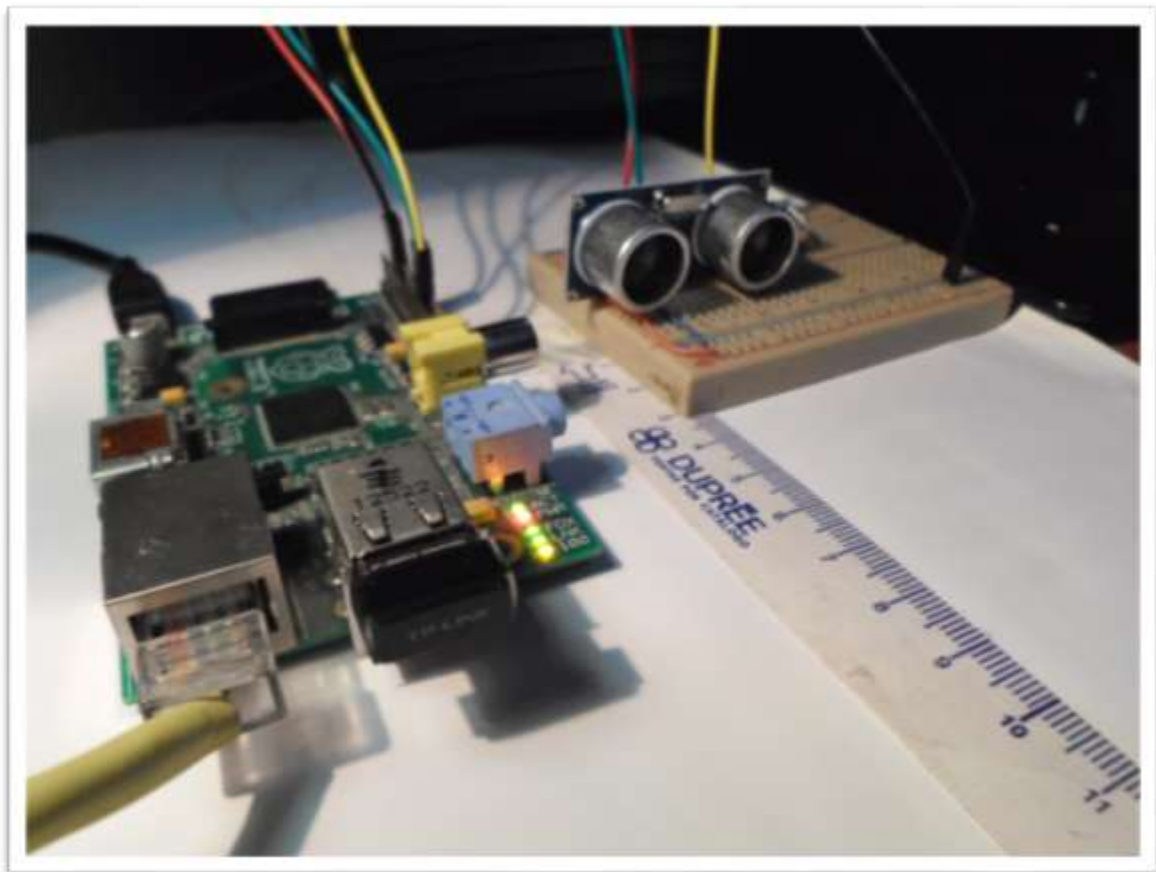
SENSOR ULTRASONIDO, CON RASPBERRY PI, VISUALIZACIÓN EN INTERFAZ WEB.



Autor: Jefferson Rivera Patiño
Email: riverajefer@gmail.com

Contents

Materiales.....	3
Teoría	3
Plano esquemático.....	4
Pasos para poner en marcha el proyecto	5
Código Python.....	7
Código Html - jQuery	8
Código PHP.....	9
Prueba desde interfaz Web.....	9
Enlace al repositorio.....	11
Referencias	11

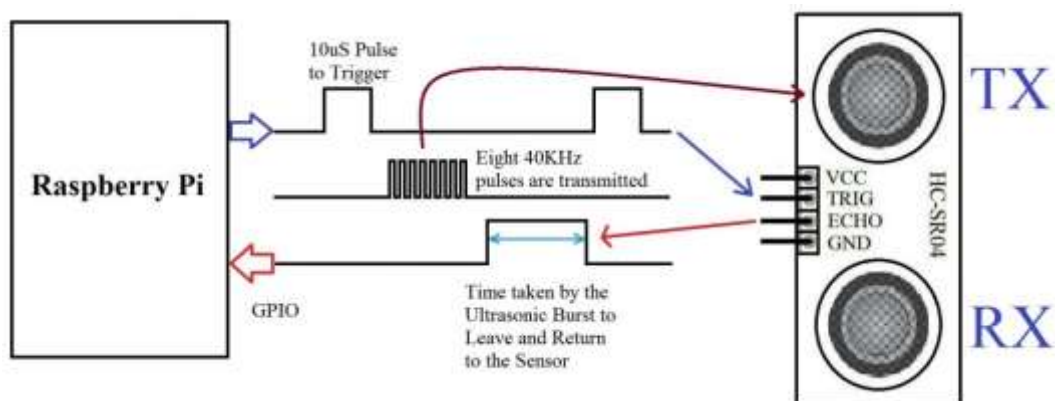
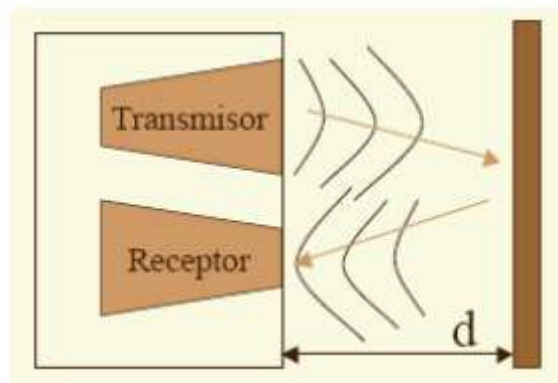


Materialles

- Raspberry Pi
- Sensor Ultrasonido HC-SR04
- Protoboard
- 1 Resistencia de 1K Ω
- 1 Resistencia de 2K Ω
- Cable Jumper Macho-Hembra

Teoría

El funcionamiento básico de los ultrasonidos como medidores de distancia se muestra de una manera muy clara en el siguiente esquema, donde se tiene un receptor que emite un pulso de ultrasonido que rebota sobre un determinado objeto y la reflexión de ese pulso es detectada por un receptor de ultrasonidos:

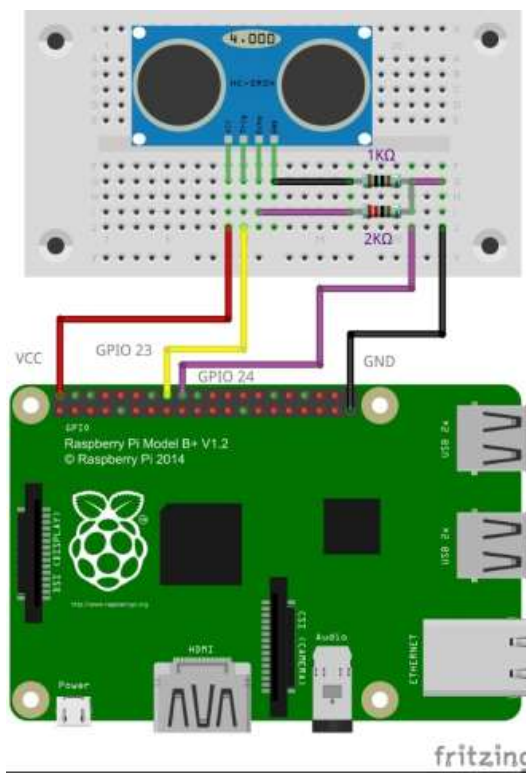
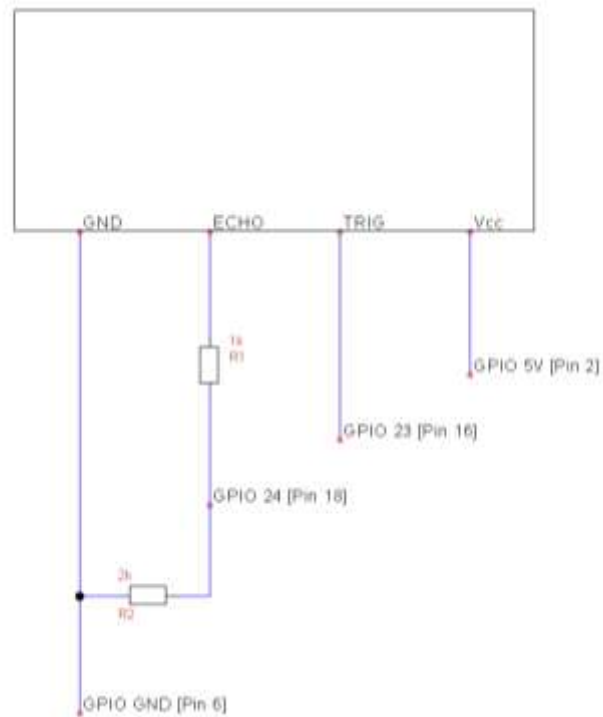


<http://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi>

<https://electrosome.com/hc-sr04-ultrasonic-sensor-raspberry-pi/>

<http://www.alcabot.com/alcabot/seminario2006/Trabajos/DiegoPerezDeDiego.pdf>

Plano esquemático



Pasos para poner en marcha el proyecto

1. Actualizar los repositorios

\$ sudo apt-get update

```
pi@raspberrypi ~ $ sudo apt-get update
```

2. Instalación de servidor Apache

\$ sudo apt-get install apache2 -y

```
pi@raspberrypi ~ $ sudo apt-get install apache2 -y
```

3. Instalación de PHP 5

\$ sudo apt-get install php5 libapache2-mod-php5 -y

```
pi@raspberrypi ~ $ sudo apt-get install php5 libapache2-mod-php5 -y
```

4. Restaurar el Servidor

\$ sudo /etc/init.d/apache2 restart

```
pi@raspberrypi ~ $ sudo /etc/init.d/apache2 restart
```

5. Permisos

\$ sudo chmod 775 /var/www/

```
pi@raspberrypi ~ $ sudo chmod 775 /var/www/
```

6. Añadimos el usuario pi al grupo www-data

\$ sudo usermod -a -G www-data pi

```
pi@raspberrypi ~ $ sudo usermod -a -G www-data pi
```

7. Dar permisos al usuario www-data de ejecutar comandos con sudo

\$ sudo visudo

```
pi@raspberrypi ~ $ sudo visudo
```

Agregamos el siguiente código al final, www-data ALL=(root) NOPASSWD:ALL

```
# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d
pi ALL=(ALL) NOPASSWD: ALL
www-data ALL=(root) NOPASSWD:ALL
```

8. Instalar librería GPIO Python

\$ sudo apt-get install python-rpi.gpio

```
pi@raspberrypi ~ $ sudo apt-get install python-rpi.gpio
```

9. Instalar Git

\$ sudo apt-get install git

```
pi@raspberrypi ~ $ sudo apt-get install git
```

10. Clonar repositorio

\$ cd /var/www # Entrar a la carpeta www

```
pi@raspberrypi ~ $ cd /var/www
```

```
pi@raspberrypi /var/www $ sudo git clone https://github.com/rpi-jefer/rpi-ultrasonido
```

\$ cd rpi-ultrasonido # Entrar a la carpeta del *rpi-ultrasonido*

11. Pruebas

\$ sudo python python/ultrasonido.py # probar el python

```
pi@raspberrypi /var/www/rpi-ultrasonido $ sudo python python/ultrasonido.py
```

```
pi@raspberrypi /var/www/rpi-ultrasonido $ sudo python python/ultrasonido.py
Medicion de la distancia en curso
Espere que el sensor se estabilice
Distancia: 219.67 cm
pi@raspberrypi /var/www/rpi-ultrasonido $
```

\$ sudo php php/medir.py # probar el php

Código Python

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)

TRIG = 23          #pin 23 como TRIG
ECHO = 24          #pin 24 como ECHO
V = 34300          # Velocidad del sonido 34300cm/s

print "Medicion de la distancia en curso"

GPIO.setup(TRIG,GPIO.OUT)          #TRIG como salida
GPIO.setup(ECHO,GPIO.IN)           #ECHO como entrada

GPIO.output(TRIG, False)           #TRIG en estado bajo
print "Espere que el sensor se estabilice"
time.sleep(2)                       #Esperar 2 segundos

GPIO.output(TRIG, True)             #TRIG en estado alto
time.sleep(0.00001)                #Delay de 0.00001 segundos
GPIO.output(TRIG, False)           #TRIG en estado bajo

while GPIO.input(ECHO)==0:          #Comprueba si ECHO está en estado bajo
    pulse_start = time.time()       #Guarda el tiempo transcurrido, mientras esta en estado bajo

while GPIO.input(ECHO)==1:          #Comprueba si ECHO está en estado alto
    pulse_end = time.time()         #Guarda el tiempo transcurrido, mientras esta en estado alto

t = pulse_end - pulse_start         #Se obtienen la duración del pulso, calculando la diferencia

distancia = t * (V/2)               #Se multiplica la duración del pulso, por 17150, para obtener la distancia
distancia = round(distancia, 2)     #Se redondea a dos decimales

if distancia > 2 and distancia < 400: #Comprueba si la distancia está dentro del rango
    print "Distancia: ",distancia,"cm" #Imprime la distancia
else:
    print "Fuera de Rango"           #Imprime fuera de rango
GPIO.cleanup()                      #Limpia los pines
```

Código Html - jQuery

```
<!DOCTYPE html>
<html lang="es" class="no-js">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Ultrasonido</title>
    <link rel="stylesheet" type="text/css" href="css/default.css" />
    <link rel="stylesheet" type="text/css" href="css/component.css" />
    <link rel="stylesheet" type="text/css" href="css/animate.css">
    <script src="js/modernizr.custom.js"></script>
    <script src="https://code.jquery.com/jquery-2.1.4.min.js"></script>
  </head>
  <body>

    <!-- Contendor, que muestra la salida (Distancia) -->
    <div id="salida" style="display:none">
      <h1 class="animated zoomInDown"></h1>
    </div>

    <!-- Contendor, que muestra el botón, para medir la distancia -->
    <div class="container animated fadeIn" id="medir">
      <section id="set-8">
        <h1>Medir Distancia</h1>
        <div class="hi-icon-wrap hi-icon-effect-8">
          <a href="#set-8" class="hi-icon hi-icon-contract" id="click-medir">Medir</a>
        </div>
      </section>
    </div>

    <!-- Contendor, que muestra el load (midiendo...) -->
    <div class="midiendo" id="load-midiendo" style="display:none">
      <div id="loader-container">
        <p id="loadingText">Midiendo...</p>
      </div>
    </div>

    <!-- Script, comunicación con el servidor -->
    <script>
      // Evento click sobre el boton medir
      $('#click-medir').click(function(){

        $('#medir').hide('slow'); // oculta el boton
        $('#load-midiendo').show('slow'); // muestra el load (midiendo)
        $('#salida').hide('slow'); // oculta la salida (si hay algo)

        // petición ajax al servidor
        $.ajax({
          url:'php/medir.php', // se encarga de ejectura el script python
          type:'POST', // metodo POST

          // Si todo sale bien
          success: function(response){

            $('#salida').show('slow'); // muestra la salida (Distancia)
            $('#salida h1').text(response); // muestra la salida (Distancia)
            $('#medir').show('slow'); // Muestra el boton medir
            $('#load-midiendo').hide('slow'); // oculta el load
            $('#set-8 h1').text('Volver Medir'); // cambia de texto a "Volver a Medir"
          }
        }); // fin ajax

      }); // fin click
    </script>

  </body>
</html>
```


Código PHP

```
<?php
// Ejecuta como root, el script python encargado de controlar el sensor ultrasonido

echo exec('sudo python /var/www/ultrasonido/python/ultrasonido.py');

// La salida la imprime con el echo
?>
```

Prueba desde interfaz Web

Obtener la IP local

```
pi@raspberrypi /var/www/rpi-ultrasonido $ ifconfig
eth0      Link encap:Ethernet  HWaddr b8:27:eb:4d:30:41
          inet addr:192.168.1.127  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:16633 errors:0 dropped:363 overruns:0 frame:0
          TX packets:11120 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:16246727 (15.4 MiB)  TX bytes:1234808 (1.1 MiB)

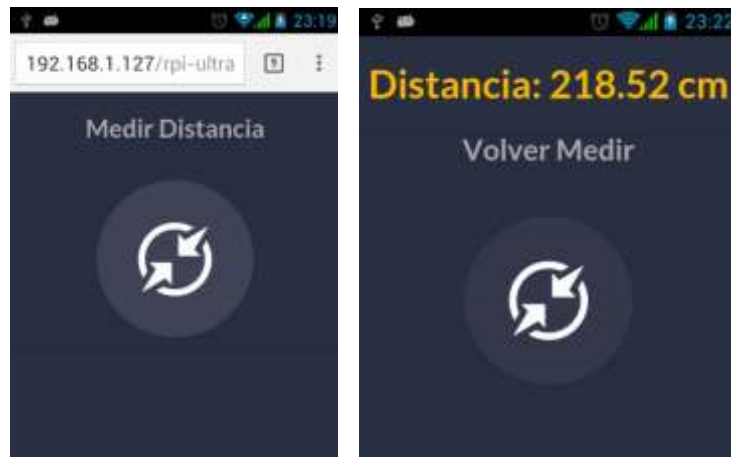
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:59 errors:0 dropped:0 overruns:0 frame:0
          TX packets:59 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:5064 (4.9 KiB)  TX bytes:5064 (4.9 KiB)
```

Desde un PC, conectado a la misma RED Local

http://192.168.1.127/rpi-ultrasonido



Desde el mobil.



Enlace al repositorio

<https://github.com/rpi-jefer/rpi-ultrasonido>

Referencias

<http://www.modmypi.com/blog/hc-sr04-ultrasonic-range-sensor-on-the-raspberry-pi>

<https://electrosome.com/hc-sr04-ultrasonic-sensor-raspberry-pi/>

<http://tympanus.net/codrops/2013/05/30/simple-icon-hover-effects/>

<http://www.micropik.com/PDF/HCSR04.pdf>