

Home / Tutorials, Raspberry Pi, Python Rpi / Interfacing HC-SR04 Ultrasonic Sensor with Raspberry Pi Interfacing HC-SR04 Ultrasonic Sensor with Raspberry Pi



Start building apps to with 25+ free services

- Interfacing HC-SR04 Ultrasonic Sensor with Raspberry Pi
- 📤 By Vivek Kartha 🝃 Python Rpi, Raspberry Pi, Tutorials 🕒 HC-SR04, Raspberry Pi 💂 33 Comments

Ultrasonic distance sensors are designed to measure distance between the source and target using ultrasonic waves. We use ultrasonic waves because they are relatively accurate across short distances and don't cause disturbances as they are inaudible to human ear.



HC-SR04 Ultrasonic Sensor

Contents

- 1 Working of HC-SR04
- 2 Voltage Divider
- 3 Connection Diagram
- 4 Distance Calculation
 - 4.1 Calibration
- 5 Python Programming
- 6 Output
- 7 Reading accuracy

HC-SR04 is a commonly

used module for non contact distance measurement for distances from 2cm to 400cm. It uses sonar (like bats and dolphins) to measure distance with high accuracy and stable readings. It consist of an ultrasonic transmitter, receiver and control circuit. The transmitter transmits short bursts which gets reflected by target and are picked up by the receiver. The time difference between transmission and reception of ultrasonic signals is calculated. Using the speed of sound and 'Speed = Distance/Time' equation, the distance between the source and target can be easily calculated.



CleanMyMac X: Your Mac nuovo. Un Mac pulito, velc potente in un solo clic.

Ad MacPaw

Scopri di più

HC-SR04 ultrasonic distance sensor module has four pins:

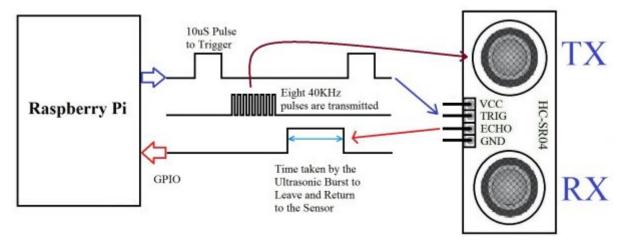
- VCC 5V, input power
- TRIG Trigger Input
- **ECHO** Echo Output
- GND Ground

Working of HC-SR04



Ultrasonic Module Operation

- 1. Provide trigger signal to TRIG input, it requires a HIGH signal of atleast 10µS duration.
- 2. This enables the module to transmit eight 40KHz ultrasonic burst.
- 3. If there is an obstacle in-front of the module, it will reflect those ultrasonic waves
- 4. If the signal comes back, the ECHO output of the module will be HIGH for a duration of time taken for sending and receiving ultrasonic signals. The pulse width ranges from $150\mu S$ to 25mS depending upon the distance of the obstacle from the sensor and it will be about 38mS if there is no obstacle.

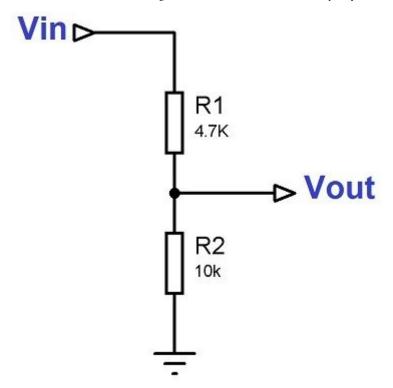


Interfacing Raspberry Pi with HC-SR04

Voltage Divider

The ECHO output is of 5v. The input pin of Raspberry Pi GPIO is rated at 3.3v. So 5v cannot be directly given to the unprotected 3.3v input pin. Therefore we use a voltage divider circuit using appropriate resistors to bring down the voltage to 3.3V.



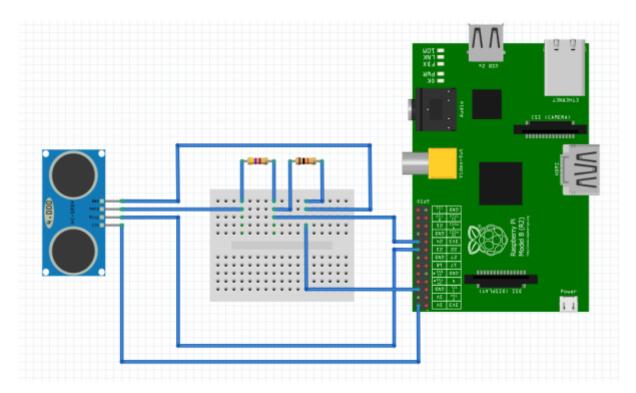


5V to 3.3V Voltage Divider

The following equation can be used for calculating resistor values,

"Vout = Vin x R2/(R1+R2)"

Connection Diagram



Interfacing Raspberry Pi with HC-SR04 Circuit Diagram

Distance Calculation

Time taken by pulse is actually for **to and fro** travel of ultrasonic signals, while we need only half of this. Therefore Time is taken as Time/2.

Distance = Speed * Time/2

Speed of sound at sea level = 343 m/s or 34300 cm/s

Thus, Distance = 17150 * Time (unit cm)

Calibration

For accurate distance readings the output can be calibrated using a ruler. In the below program a calibration of 0.5 cm is added.

Python Programming

```
import RPi.GPIO as GPIO
                                           #Import GPIO library
import time
                                           #Import time library
GPIO.setmode(GPIO.BCM)
                                           #Set GPIO pin numbering
TRIG = 23
                                           #Associate pin 23 to TRIG
ECH0 = 24
                                           #Associate pin 24 to ECHO
print "Distance measurement in progress"
GPIO.setup(TRIG,GPIO.OUT)
                                           #Set pin as GPIO out
GPIO.setup(ECHO,GPIO.IN)
                                           #Set pin as GPIO in
while True:
  GPIO.output(TRIG, False)
                                           #Set TRIG as LOW
  print "Waitng For Sensor To Settle"
  time.sleep(2)
                                           #Delay of 2 seconds
  GPIO.output(TRIG, True)
                                           #Set TRIG as HIGH
  time.sleep(0.00001)
                                           #Delay of 0.00001 seconds
  GPIO.output(TRIG, False)
                                           #Set TRIG as LOW
  while GPIO.input(ECHO)==0:
                                           #Check whether the ECHO is LOW
    pulse start = time.time()
                                           #Saves the last known time of LOW pulse
  while GPIO.input(ECHO)==1:
                                           #Check whether the ECHO is HIGH
    pulse_end = time.time()
                                           #Saves the last known time of HIGH pulse
  pulse_duration = pulse_end - pulse_start #Get pulse duration to a variable
  distance = pulse duration * 17150 #Multiply pulse duration by 17150 to get
distance
  distance = round(distance, 2)
                                           #Round to two decimal points
  if distance > 2 and distance < 400:
                                           #Check whether the distance is within range
    print "Distance:",distance - 0.5,"cm" #Print distance with 0.5 cm calibration
  else:
                                           #display out of range
    print "Out Of Range"
```

Run the above program.

Output



Ultrasonic distance sensor output

Distance is measured every two seconds and displayed.

Reading accuracy

- For greater accuracy use C over Python as Python on a Linux environment is not good for precise timing. As precise timing is required for accurate distance measurements.
- The sensor has a wide angle of sensitivity. If there are objects near the line of sight it may give shorter readings.
- The ultrasonic sensor touching any surface can give wrong readings.

Any doubts? Comment below.

Blinkenlights

Build a recreation of an Altair with CP/M and Altair DOS programs, or an historic Kenbak-1

Famous Davis

Like 4.8K people like this. Be the first of your friends.

Related Posts:



Interfacing HC-SR04
Ultrasonic Distance
Sensor with...



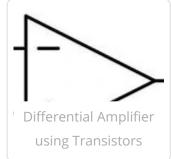
- Arduino



























Vivek Kartha



Christian Digaum Cantos • a year ago

how can i put the data on the QT desinger?

∧ V · Reply · Share ›



Pedro Azevedo • 2 years ago

when I run the code I receive this error:

Distance Measurement In Progess

test_ultra.py:10: RuntimeWarning: This channel is already in use, continuing anyway. Use GPIO.setwarnings(False) to disable warnings.

GPIO.setup(TRIG,GPIO.OUT)

Waiting For Sensor To Settle

Traceback (most recent call last):

File "test_ultra.py", line 27, in <module>

pulse_duration = pulse_end - pulse_start

NameError: name 'pulse_start' is not defined

Someone can help me?

∧ V • Reply • Share •



Deepthi • 2 years ago

when we get out of range can we send that message to our phone?? do we have any code for that??

∧ ∨ • Reply • Share •



David James Moore • 2 years ago

Thank you I have used your example to build a basic rig. I am planning on coding a C++ version



to store values in a database for historic reporting and alerting. You have really helped start me my project.



Tiago Carvalho • 3 years ago

This program is not working for me..it's stuck on "Waiting For Sensor To Settle"...

I've checked all the wiring and nothing seems to work.. any idea or pre requesit in raspberry configurations that are needeD?



Ligo George Mod → Tiago Carvalho • 3 years ago

Make sure that you have python gpio library.



Subhajit Biswas • 3 years ago

Hi,

I ran the code but it's not displaying the result. It stucked at a "Waiting for Sensor to settle". Checked circuit diagram also, everything seems to be correct.



Pagaltubbu → Subhajit Biswas • 2 years ago

I am having the same issue, despite checking my circuit diagram. Were you able to resolve your issue? If yes, please let me know how.

Thanks!



Ligo George Mod → Subhajit Biswas • 3 years ago

It seems like there is some connection mistake.



darian Brown • 3 years ago

When i run this it says that a in line 32 a float is required. as far as i can tell my code is exactly the same as yours.



darian Brown → darian Brown • 3 years ago

sorry i put a comma in the multiplier it works good.



Shaheen Makandar • 3 years ago

how to get email alert from this sensor? i mean do you have code for that



Ligo George Mod → Shaheen Makandar • 3 years ago

You can install some SMTP applications and send mail through it.



Kartik Hadke → Ligo George • 3 years ago

but that will require internet connection so how to do that??

internet connection on raspberry pi will u explain.

```
∧ ∨ • Reply • Share →
```

Show more replies



Viraj · 3 years ago

Why 0.5cm calibration?



Ligo George Mod → Viraj • 3 years ago

You can add a calibration to make the results more accurate.



Rifat Kasap • 3 years ago

anyone know how to connect 3 ultrasonic sensors to raspberry pi?



Chotu → Rifat Kasap • 3 years ago

Vcc and ground putted together to waste only 2 ports. for trigger and echo I suggest you to use 6 wires for 3 sensors and set a delay between triggering them (be aware u mustn't trigger them all at once, even if they are separed. Ultra sound can quickly come back to the sensor by reflecting from walls and corners giving you wrong calculations) the minimum delay needs to be 38 uS, when i was doing a project with them i putted a delay of 100ms and it was more than enough. i cant help you more, i wasnt using rasperry.



Jacek "Yatsek" Radajewski • 3 years ago

I used 470 ohm and 1k resistors in my setup (exactly 1/10th of what you have). It works, but should I anticipate problems?



Ligo George Mod → Jacek "Yatsek" Radajewski • 3 years ago

It will increase the current flow 10 times, make sure that it doesn't exceed the output current limit of SR04.



Ben • 4 years ago



Hi I really enjoyed your tutorial, thank you for posting. I bought one of these sensors and tried to run the code. However the sen

sor never receives the ultrasound pulse. I tried changing the pulse length, I checked the wiring multiple times, changed the wires and arrangement on the breadboard, and even bought a new sensor module. None of these fixed the problem. My only other thought is that I damaged the GPIO pins. When I first hooked it all up I didn't use resistors as I thought the new pi could receive 5V input. I checked the pins by trying to read input from an accelerometer, and that still worked but admittedly it used different pins. I was wondering if you had any thoughts on my problem? Thank you for the help:)

∧ V • Reply • Share •



Nishant Parmar → Ben • 10 months ago

If you are following the aforementioned circuit diagram, try using different pin numbers (worked for me with pin numbers 16 and 18, instead of 23 and 24 respectively).

∧ ∨ • Reply • Share ›



Ligo George Mod → Ben • 4 years ago

Try to find where program fails by printing comments at different parts of your program.



eray • 4 years ago

Hello

I cannot take 5V from echo pin. What is it reason? Any suggestions?

∧ V • Reply • Share •



Ligo George Mod → eray • 4 years ago

It is clearly explained above.

∧ V · Reply · Share ›



alex • 4 years ago

please can any one help me

i have been trying to use 2 ultra sonic sensors on the pi with one of my robotics projects but i cant seem to get the code to work properly. i cant figure out get a response from the second on i think the problem is with the time.time function

∧ V • Reply • Share •



Ligo George Mod → alex • 4 years ago

time.time function ??

∧ V • Reply • Share •



Stephen Loomis • 4 years ago

I replaced both while GPIO.input(ECHO)==0: and ==1;

with

GPIO.wait_for_edge(GPIO_ECHO, GPIO.BOTH)

start = time.time()

GPIO.wait_for_edge(GPIO_ECHO, GPIO.BOTH)

stop = time.time()

and initialized with

GPIO.add_event_detect(GPIO_ECHO, GPIO.BOTH)

Reducing CPU load significantly.

I tried using GPIO.RISING the GPIO.FALLING, but some times the device triggers all on its own and can be in the middle of an echo reply.



Paul Krela • 4 years ago

I have this letter all time and it wont go away and stays for very long time.

Waiting for sensor to settle

Any suggestions?



Vivek Kartha Mod → Paul Krela • 4 years ago

Verify whether the echo pin of sensor is wired correctly.



Avinash Gupta • 4 years ago

Good. But Voltage divider using resistors are not accurate for high speed signals. Better use a 3.3v Zener diodes.



Vivek Kartha Mod → Avinash Gupta • 4 years ago

Thanks for your suggestion..

ALSO ON ELECTROSOME

Controlling LED using ESP8266 and Telegram Bot – IoT Project

1 comment • 4 months ago

Darma Budi teja pranata — I will make arduino Avatarproject with gps . can bot send link google map?

Home Automation using Arduino and ESP8266 Module

37 comments • 2 years ago

somnath panja — https://uploads.disquscdn.c...
Avatarhttps://uploads.disquscdn.c...do not come ip address...why ??? arduino program is ok ????

Outling Otaliad with Olaway Davidaniana Mahila Dhana Datastay Oiyawii

Getting Started with Gloudx Development Board

1 comment • 6 months ago

Miss Issahasası

No link for Cloudy download

MODILE Phone Detector Circuit

1 comment • 2 years ago



RELATED **POSTS**



Getting Started with ESP8266 Programming – Arduino

February 1, 2018 | Basamma B | ESP8266, Tutorials | Arduino, ESP-01, ESP-12E, ESP8266, IoT, WiFi Module | 0 Comments

Contents1 ESP8266 WiFi Module2 Features of ESP82663 Prerequisite for programming3.1 Installing ESP8266 Board in Arduino IDE3.2 Bootloader Modes3.3 Circuit...

Read More →

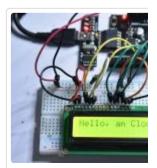


Interfacing MPU-6050 / GY-521 board with Arduino Uno

December 16, 2018 | Vivek Kartha | Arduino, Tutorials | Arduino, Arduino Uno, MPU-6050 | 1 Comment

Contents1 About MPU-60501.1 3-axis Gyroscope1.2 3-axis
Accelerometer1.3 On chip
temperature sensor1.4 Digital
motion processor (DMP)1.5 GY-521
breakout board1.5.1 Pinout2...

Read More →



Interfacing with Cloud Board

October 23, 2018 | A Olayiwola | CloudX, CloudX, LCD, Microc 0 Comments

Contents1 Componic CloudX LCD Library2 Setting2.2 LCD Cmd Write2.4 LCD Write0 WriteText2.6 LCD W Custom...

Read More →

RECENT POSTS



Sending E-mail from ESP8266 – IoT Project January 17, 2019

Controlling LED using ESP8266 and Telegram Bot – IoT Project



December 21, 2018



Interfacing MPU-6050 / GY-521 board with Arduino Uno December 16, 2018



Updating Sensor Data to Google Spreadsheet using ESP8266 – IoT Project November 28, 2018



Flashing Espressif and NodeMCU Firmware to ESP8266 October 28, 2018



Bluetooth Home Automation With CloudX October 27, 2018

FOLLOW US

f



G+

MOST VIEWED



Astable Multivibrator using Transistors by Ligo George



Getting Started with MPLAB XC8 Compiler – LED Blinking by Ligo George



LED Blinking using 8051 Microcontroller and Keil C – AT89C51 **by Ebin George**



5V Power Supply using 7805 Voltage Regulator with Design **by Manoj Shenoy**



What is a Microprocessor? **by Ligo George**



Transistor as a Switch **by Ligo George**

CATEGORIES

Electronics
555 Circuits
Basic Electronics
▶ Opamp
Electronics News
• IC
Power Supply
Sensors
What is Inside?
Featured
Mobiles
News
Processors
Projects
• Embedded
▶ Arduino
▶ AVR Microcontroller
▶ ESP8266
▶ PIC Microcontroller

▶ Hobby Circuits
▶ IoT
▶ Softwares
▶ Tablets
▶ Technology Transfer
▶ Tutorials
▶ 8051 Microcontroller
▶ 8085
▶ Arduino
► ATMEL AVR
▶ CloudX
▶ ESP8266
► MATLAB
▶ OpenWrt
▶ PCB Designing
▶ DipTrace
▶ OrCad
▶ PIC Microcontroller
▶ CCS C Compiler
▶ Hi-Tech C
▶ MikroC
▶ MPLAB XC8
▶ Python
▶ Raspberry Pi
▶ Python Rpi
▶ SCILAB

MOST COMMENTED



Digital Clock using PIC Microcontroller and DS1307 RTC by Ligo George



Voltmeter and Ammeter using PIC Microcontroller by Ligo George



Water Level Indicator and Controller using PIC Microcontroller by Ligo George



Interfacing HC-SR04 Ultrasonic Sensor with PIC Microcontroller **by Ligo George**

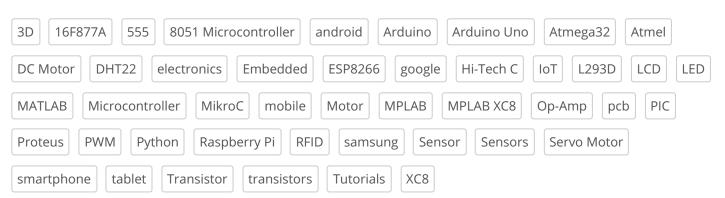


Automatic Power Factor Controller using Microcontroller by Divya Syamala



USB PIC Programmer: PICKit2 by Ligo George

TAGS



RECENT COMMENTS

▶ minhal shafiq on Interfacing EM-18 RFID reader with Raspberry Pi

- ▶ Theodore_Rimspoke on Interfacing LCD with PIC Microcontroller CCS C
- ▶ Lily on Simple Electronic Piano using 555 Timer
- ▶ azi alaei on Connecting ESP8266 to WiFi Network Beginners Guide
- ▶ sharad on Solar Tracker system Using LM358
- © Copyright 2018. All Rights Reserved.