Interfacing HC-SR04 Ultrasonic Sensor with Raspberry Pi

30 Jan

Interfacing HC-SR04 Ultrasonic Sensor with Raspberry Pi

By Vivek Kartha (https://electrosome.com/author/vivek_kartha/)

Python Rpi (https://electrosome.com/category/tutorials/raspberry-pi/python-rpi/), Raspberry Pi (https://electrosome.com/category/tutorials/raspberry-pi/), Tutorials (https://electrosome.com/category/tutorials/)
HC-SR04 (https://electrosome.com/tag/hc-sr04/), Raspberry Pi (https://electrosome.com/tag/raspberry-pi/)
30 Comments (https://electrosome.com/hc-sr04-ultrasonic-sensor-raspberry-pi/#disqus_thread)

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.

Contents

(https://electrosome.com/wp-content/uploads/2014/10/HC-SR04-Ultrasonic-Sensor.jpg)

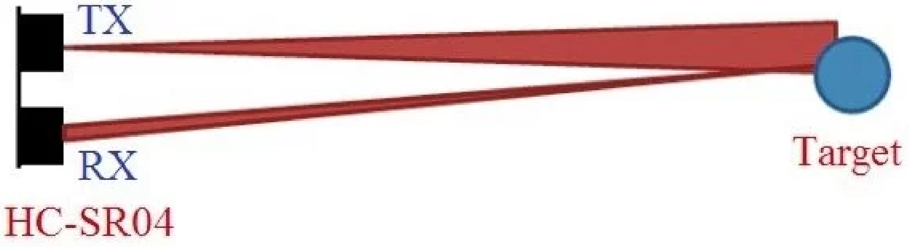
HC-SR04 Ultrasonic Sensor

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.

HC-SR04 ultrasonic distance sensor (https://electrosome.com/ultrasonic-sensors-distance-proximity/) module has four pins :

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

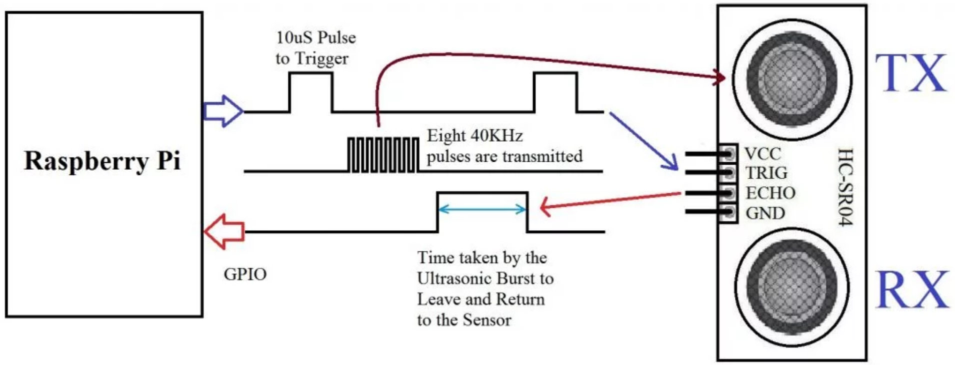
Working of HC-SR04



(https://electrosome.com/wp-content/uploads/2014/08/Ultrasonic-Module-Operation.jpg)

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 150μS to 25mS depending upon the distance of the obstacle from the sensor and it will be about 38ms if there is no obstacle.

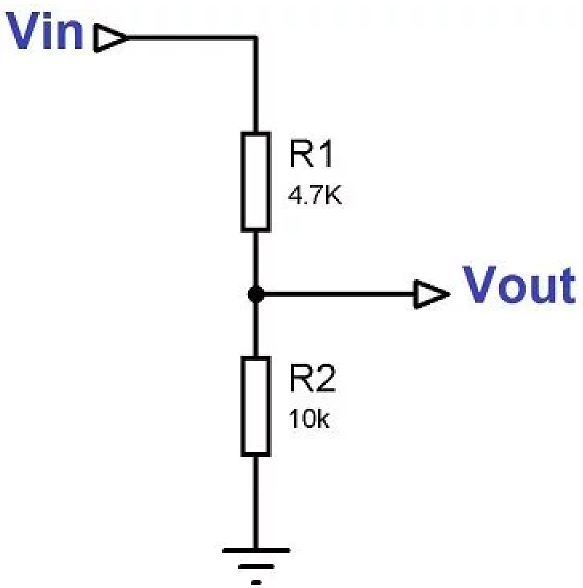

Raspberry Pi
10uS Pulse to Trigger
Eight 40KHz pulses are transmitted
GPIO
Time taken by the Ultrasonic Burst to Leave and Return to the Sensor
VCC
TRIG
ECHO
GND
HC-SR04
TX
RX

(https://electrosome.com/wp-content/uploads/2015/01/Interfacing-Raspberry-Pi-with-HC-SR04.jpg)

Interfacing Raspberry Pi (https://electrosome.com/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.



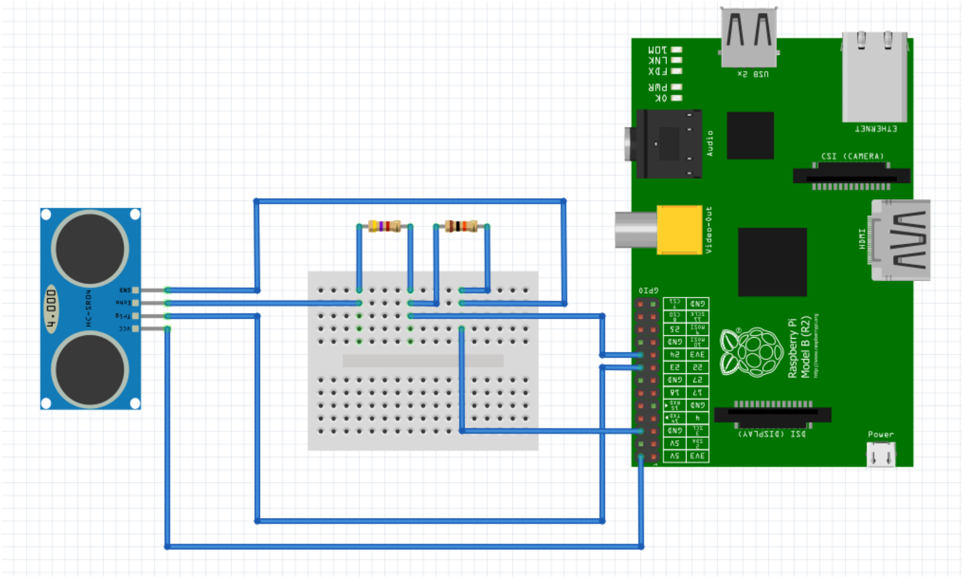
(https://electrosome.com/wp-content/uploads/2015/01/5V-to-3.3V-Voltage-Divider.jpg)

5V to 3.3V Voltage Divider

The following equation can be used for calculating resistor values,

“Vout = Vin x R2/(R1+R2)”

Connection Diagram



(https://electrosome.com/wp-content/uploads/2015/01/Interfacing-Raspberry-Pi-with-HC-SR04-Circuit-Diagram.png)

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

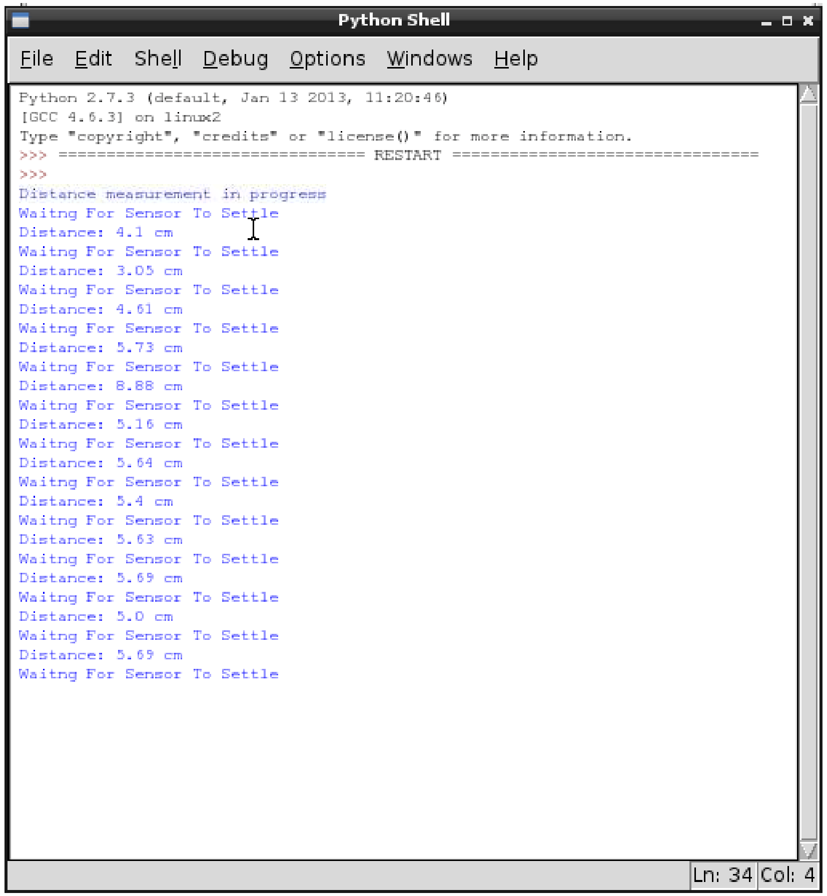
For accurate distance readings the output can be calibrated using a ruler. In the below program a calibration of 0.5 cm is added.

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
ECHO = 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 pu

  while GPIO.input(ECHO)==1:               #Check whether the ECHO is HIGH
    pulse_end = time.time()                #Saves the last known time of HIGH p

  pulse_duration = pulse_end - pulse_start #Get pulse duration to a variable


  distance = pulse_duration * 17150        #Multiply pulse duration by 17150 to
  distance = round(distance, 2)            #Round to two decimal points

  if distance > 2 and distance < 400:      #Check whether the distance is withi
    print "Distance:",distance - 0.5,"cm"  #Print distance with 0.5 cm calibrat
  else:
    print "Out Of Range"                   #display out of range
```

Run the above program.

Output



```
Python 2.7.3 (default, Jan 13 2013, 11:20:46)
[GCC 4.6.3] on linux2
Type "copyright", "credits" or "license()" for more information.
>>> ================================ RESTART ================================
>>>
Distance measurement in progress
Waitng For Sensor To Settle
Distance: 4.1 cm
Waitng For Sensor To Settle
Distance: 3.05 cm
Waitng For Sensor To Settle
Distance: 4.61 cm
Waitng For Sensor To Settle
Distance: 5.73 cm
Waitng For Sensor To Settle
Distance: 8.88 cm
Waitng For Sensor To Settle
Distance: 5.16 cm
Waitng For Sensor To Settle
Distance: 5.64 cm
Waitng For Sensor To Settle
Distance: 5.4 cm
Waitng For Sensor To Settle
Distance: 5.63 cm
Waitng For Sensor To Settle
Distance: 5.69 cm
Waitng For Sensor To Settle
Distance: 5.0 cm
Waitng For Sensor To Settle
Distance: 5.69 cm
Waitng For Sensor To Settle
```

(https://electrosome.com/wp-content/uploads/2015/01/Ultrasonic-distance-sensor-output.png)

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.

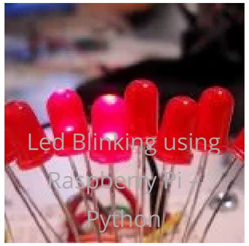
- The sensor has a wide angle of sensitivity. If there are objects near the line of sight it may give shorter readings.
- The ultrasonic sensor touching any surface can give wrong readings.

Any doubts? Comment below.

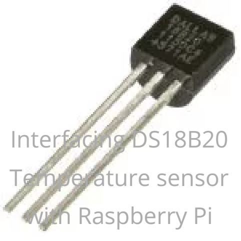
Related Posts:

Interfacing HC-SR501 PIR Motion Sensor with Raspberry Pi

Using Switch with Raspberry Pi – Python



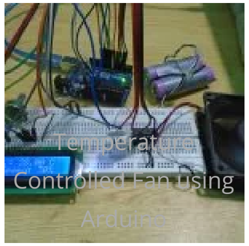
Led Blinking using Raspberry Pi – Python



Interfacing DS18B20 Temperature sensor with Raspberry Pi

Interfacing EM-18 RFID reader with Raspberry Pi

Using UART on Raspberry Pi – Python



Temperature Controlled Fan using Arduino

Monostable Multivibrator using Transistors

Share this post

Author

Vivek Kartha (https://electrosome.com/author/vivek_kartha/)

30 Comments **electroSome** 1 **Login**

♡ **Recommend** 1 **Share** **Sort by Best**

Join the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Deepthi • 6 months 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 • 7 months 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.

^ | ˅ • Reply • Share ›

Tiago Carvalho • a year ago

This program is not working for me..it's stuck on "Waitng 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 neededD?

^ | ˅ • Reply • Share ›

Ligo George Mod ➦ Tiago Carvalho • a year ago

Make sure that you have python gpio library.

^ | ˅ • Reply • Share ›

Subhajit Biswas • a year 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.

^ | ˅ • Reply • Share ›

Pagaltubbu ➦ Subhajit Biswas • a month 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!

∧ | ∨ • Reply • Share ›

Ligo George Mod ➦ Subhajit Biswas • a year ago

It seems like there is some connection mistake.

∧ | ∨ • Reply • Share ›

darian Brown • a year 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.

∧ | ∨ • Reply • Share ›

darian Brown ➦ darian Brown • a year ago

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

1 ∧ | ∨ • Reply • Share ›

Shaheen Makandar • 2 years ago

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

∧ | ∨ • Reply • Share ›

Ligo George Mod ➦ Shaheen Makandar • a year ago

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

∧ | ∨ • Reply • Share ›

Kartik Hadke ➦ Ligo George • a year ago

but that will require internet connection
so how to do that??
internet connection on raspberry pi will u explain.

∧ | ∨ • Reply • Share ›

Ligo George Mod ➦ Kartik Hadke • 10 months ago

Yes, for sending emails anyway you need internet connection.

∧ | ∨ • Reply • Share ›

Viraj • 2 years ago

Why 0.5cm calibration?

∧ | ∨ • Reply • Share ›

Ligo George Mod ➔ Viraj • 2 years ago

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

^ | ˅ • Reply • Share ›

Rifat Kasap • 2 years ago

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

^ | ˅ • Reply • Share ›

Chotu ➔ Rifat Kasap • 2 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.

1 ^ | ˅ • Reply • Share ›

Jacek “Yatsek” Radajewski • 2 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?

^ | ˅ • Reply • Share ›

Ligo George Mod ➔ Jacek “Yatsek” Radajewski • 2 years ago

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

^ | ˅ • Reply • Share ›

Ben • 2 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 :)

^ | ˅ • Reply • Share ›

Ligo George Mod ➔ Ben • 2 years ago

Ligo George Mod ➜ Ben • 2 years ago

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

⌃ | ⌄ • Reply • Share ›

eray • 2 years ago

Hello

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

⌃ | ⌄ • Reply • Share ›

Ligo George Mod ➜ eray • 2 years ago

It is clearly explained above.

⌃ | ⌄ • Reply • Share ›

alex • 2 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

⌃ | ⌄ • Reply • Share ›

Ligo George Mod ➜ alex • 2 years ago

time.time function ??

⌃ | ⌄ • Reply • Share ›

Stephen Loomis • 2 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

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.

^ | v • Reply • Share ›

Paul Krela • 3 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 ?

^ | v • Reply • Share ›

Vivek Kartha Mod ➦ Paul Krela • 3 years ago

Verify whether the echo pin of sensor is wired correctly.

1 ^ | v • Reply • Share ›

Avinash Gupta • 3 years ago

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

^ | v • Reply • Share ›

Vivek Kartha Mod ➦ Avinash Gupta • 3 years ago

Thanks for your suggestion..

^ | v • Reply • Share ›

ALSO ON **ELECTROSOME**

SPI Communication with PIC Microcontroller – MPLAB XC8

1 comment • 5 months ago•

AvatarAníbal Huerga del Canto — Hellol can not open the Proteus file. I have proteus 8. How can I open that file? Could you send

IoT Data Logger using Arduino and ESP8266

4 comments • 7 months ago•

AvatarMR SHITAL CHAWRE — hello friends,i m getting this is problem pls any suggestion how to remove this problem pls any

Custom Characters on LCD using PIC –

Using UART on Raspberry Pi – Python

RELATED **POSTS**

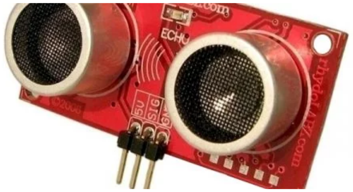
(https://electrosome.com/matrix-keypad-pic-mikroc-library/)

(https://electrosome.com/interfacing-ultrasonic-distance-sensor-ascii-pic-microcontroller/)

(https://electrosor
pic-microcontrolle

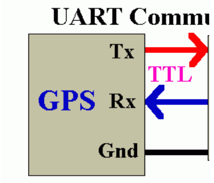
Interfacing Matrix Keypad with PIC Microcontroller using MikroC Library

(https://electrosome.com/matrix-keypad-pic-mikroc-library/)
June 20, 2012 | Ligo George
(https://electrosome.com/author/lijoppans/)
| MikroC
(https://electrosome.com/category/tutorials/pic-microcontroller/mikroc/), PIC Microcontroller
(https://electrosome.com/category/tutorials/pic-microcontroller/), Tutorials
(https://electrosome.com/category/tutorials/)
| Microcontroller
(https://electrosome.com/tag/microcontroller/), MikroC
(https://electrosome.com/tag/mikroc/), PIC
(https://electrosome.com/tag/pic/), Proteus
(https://electrosome.com/tag/proteus/), Tutorials
(https://electrosome.com/tag/tutorials/)



Interfacing Ultrasonic Distance Sensor : ASCII Output with PIC Microcontroller

(https://electrosome.com/interfacing-ultrasonic-distance-sensor-ascii-pic-microcontroller/)
March 7, 2013 | Ligo George
(https://electrosome.com/author/lijoppans/)
| Embedded
(https://electrosome.com/category/projects/embedded-projects/), MikroC
(https://electrosome.com/category/tutorials/pic-microcontroller/mikroc/), PIC Microcontroller
(https://electrosome.com/category/tutorials/pic-microcontroller/), PIC Microcontroller
(https://electrosome.com/category/projects/embedded-projects/pic-microcontroller-projects/), Projects
(https://electrosome.com/category/projects/), Tutorials
(https://electrosome.com/category/tutorials/)
| Microcontroller
(https://electrosome.com/tag/microcontroller/), MikroC



Using UA PIC Microcor – MPLAB

(https://electroso
pic-microcontrolle
October 3, 2014 |
(https://electroso
| MPLAB XC8
(https://electroso
microcontroller/m
Microcontroller
(https://electroso
microcontroller/),
(https://electroso
| Embedded
(https://electroso
Microcontroller
(https://electroso
MPLAB
(https://electroso
MPLAB XC8
(https://electroso
xc8/), PIC
(https://electroso
Proteus
(https://electroso
Tutorials
(https://electroso
XC8
(https://electroso