

Ultrasonic Range Finder 4pins

<http://learning.grobotronics.com/arduino-ultrasonic-tutorial.html>

<http://arduinobasics.blogspot.gr/2012/11/arduinobasics-hc-sr04-ultrasonic-sensor.html>

Ultrasound is a high frequency sound (typically 40 KHz is used). A short burst of sound waves (often only 8 cycles) is sent out the "Transmit" transducer (left, above). Then the "Receive" transducer listens for an echo. Thus, the principle of ultrasonic distance measurement is the same as with Radio-based radar.

HARDWARE REQUIRED

Ultrasonic Ranging Detector (SR04)
Arduino Board
Breadboard
Breadboard Jumper Cables

CIRCUIT

- Works with many different ultrasonic sensor models: SR04, SRF05, SRF06, DYP-ME007 & Parallax PING)))™.
- Option to interface with all but the SRF06 sensor using only one Arduino pin.
- Doesn't lag for a full second if no ping echo is received like all other ultrasonic libraries.
- Ping sensors consistently and reliably at up to 30 times per second.

SCHEMATIC

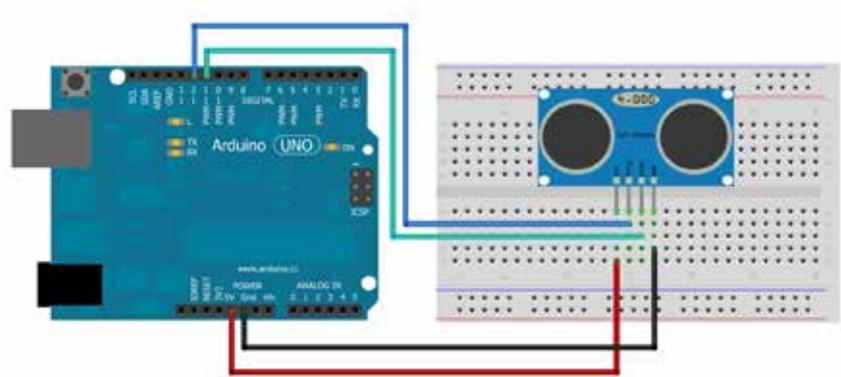
- Timer interrupt method for event-driven sketches.
- Built-in digital filter method ping_median() for easy error correction.
- Uses port registers when accessing pins for faster execution and smaller code size.
- Allows setting of a maximum distance where pings beyond that distance are read as no ping «clear».
- Ease of using multiple sensors (example sketch that pings 15 sensors).
- More accurate distance calculation (cm, inches & microseconds).
- Doesn't use pulseIn, which is slow and gives incorrect results with some ultrasonic sensor models.
- Actively developed with features being added and bugs/issues addressed.

NewPing Library Version 1.5

New in v1.5 - Released 8/15/2012:

Added ping_median() method which does a user specified number of pings (default=5) and returns the median ping in microseconds (out of range pings ignored). This is a very effective digital filter. Optimized for smaller compiled size (even smaller than sketches that don't use a library).

IMAGE



```
CODE #include <NewPing.h>

// Arduino pin tied to trigger pin on the ultrasonic sensor.
#define TRIGGER_PIN 12
// Arduino pin tied to echo pin on the ultrasonic sensor.
#define ECHO_PIN 11
// Maximum distance we want to ping for (in centimeters). Maximum sensor dis-
// tance is rated at 400-500cm.
#define MAX_DISTANCE 400

NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);
// NewPing setup of pins and maximum distance.

void setup() {
  Serial.begin(115200);
  // Open serial monitor at 115200 baud to see ping results.
}

void loop() {
  delay(50);
  // Wait 50ms between pings (about 20 pings/sec).
  // 29ms should be the shortest delay between pings.

  unsigned int uS = sonar.ping();
  // Send ping, get ping time in microseconds (uS).
  if (uS / US_ROUNDTRIP_CM==0) {
    // on spam data (=0) do nothing
  }
  else
  {
    Serial.print("Ping: ");
    Serial.print(uS / US_ROUNDTRIP_CM);
    // Convert ping time to distance in cm and print result
    // (0 = outside set distance range)
    Serial.println("cm");
  }
}
```