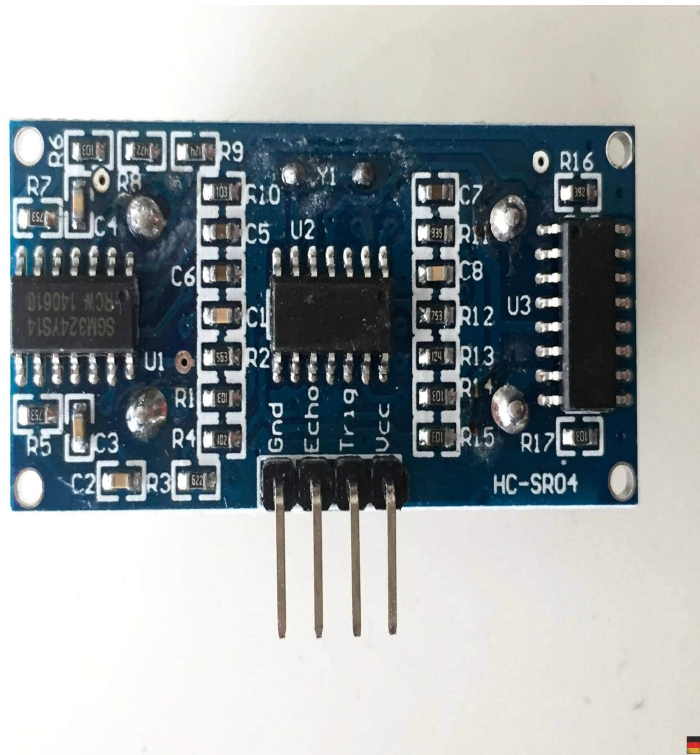


HC-SR04 mit Arduino auslesen

Schritt fuer Schritt von der Verdrahtung bis zum gefilterten Abstandswert.



MAKERSHOP ONLINE

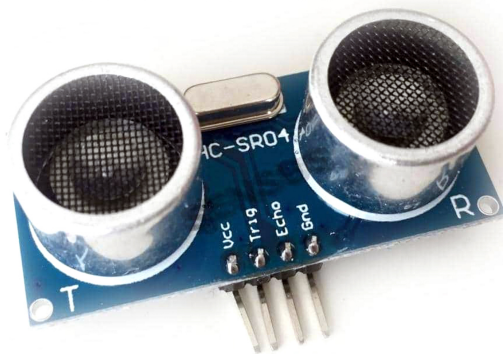
www.makershop.de

<https://www.makershop.de/sensoren/distanz/hc-sr04/>

Das lernst du in diesem Tutorial

Modul verdrahten, Echo-Puls messen, Distanz berechnen und typische Fehler erkennen.

HC-SR04 Ultraschall-Sensor



KLASSISCHER 5-V-ULTRASCHALLSENSOR

Mit einem Arduino Uno wird das 5-V-Modul direkt verbunden. Der Beispielcode nutzt `pulseIn()` mit einem festen Timeout.

- Typischer Familienwert: Messung von 2 bis 400 cm
- Getrennte Trigger- und Echo-Signale
- 40-kHz-Ultraschall mit zwei Schallwandlern
- Direkt an 5-V-Arduino-Boards nutzbar
- Vier bereits bestueckte Anschlusspins

Zeitbedarf: etwa 15 Minuten

Du benoetigst einen Arduino Uno, den HC-SR04 und vier Female-Female-Steckkabel.

1. HC-SR04 anschliessen



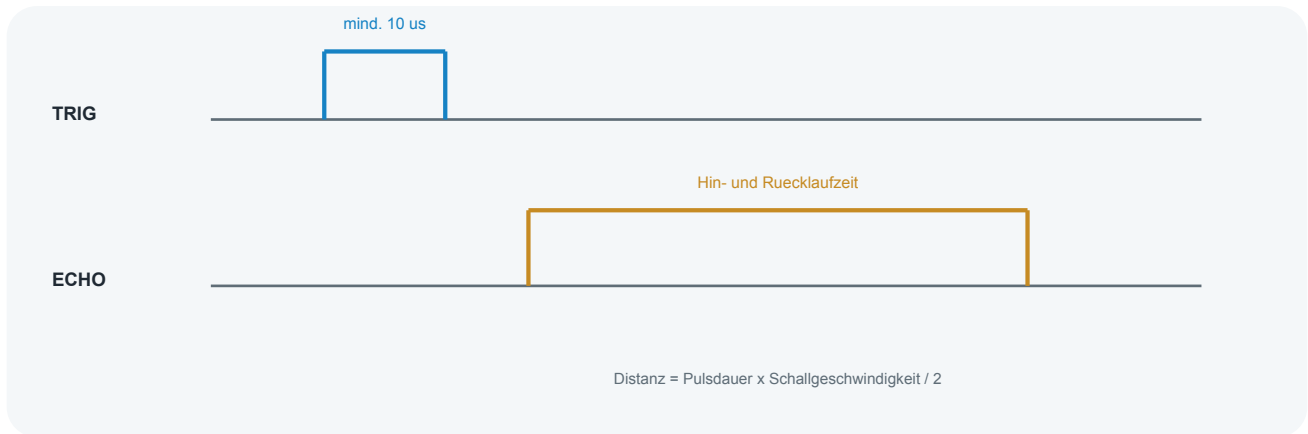
Nur bei 5-V-toleranten Eingängen direkt verbinden
ESP32 / Raspberry Pi: Echo ueber Spannungsteiler oder Pegelwandler

VCC	Arduino 5 V
GND	Arduino GND
TRIG	Arduino Digital-Pin 9
ECHO	Arduino Digital-Pin 10

Nur fuer 5-V-tolerante Eingänge direkt

Dieses Schaltbild gilt fuer den Arduino Uno. Bei 3,3-V-GPIOs muss Echo ueber einen Spannungsteiler oder Pegelwandler gehen.

2. Laufzeit messen



Minimaler Arduino-Test

```
const byte trigPin = 9;
const byte echoPin = 10;

void setup() {
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}

void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  unsigned long us = pulseIn(echoPin, HIGH, 30000);

  if (us == 0) {
    Serial.println("Kein Echo");
  } else {
    float cm = us * 0.0343 / 2.0;
    Serial.print(cm, 1);
    Serial.println(" cm");
  }
  delay(60);
}
```

3. Ergebnis einordnen

Echo-Zeit	pulseIn() liefert die High-Dauer in Mikrosekunden.
Schallgeschwindigkeit	Bei rund 20 Grad C etwa 0,0343 cm pro Mikrosekunde.
Hin- und Rueckweg	Das Ergebnis durch 2 teilen.
Timeout	30.000 us begrenzen die Wartezeit, falls kein Echo kommt.
Messpause	Mindestens etwa 60 ms bis zum naechsten Trigger warten.

Mehrere Werte beruhigen

Einfacher Mittelwert aus fuenf gueltigen Messungen

```
float sum = 0;
byte valid = 0;

for (byte i = 0; i < 5; i++) {
  unsigned long us = measureEcho();
  if (us > 0) {
    sum += us * 0.0343 / 2.0;
    valid++;
  }
  delay(60);
}

if (valid > 0) {
  float averageCm = sum / valid;
}
```

Ausreisser nicht blind mitteln

Fuer bewegte oder schwierige Ziele ist ein Medianfilter oft robuster als ein einfacher Mittelwert.

4. Typische Fehler schnell loesen

Immer 0 / Kein Echo	Ziel, Ausrichtung, Verdrahtung und Timeout pruefen.
Immer Maximalwert	Echo-Leitung oder fehlende Reflexion kontrollieren.
Starke Spruenge	Messpause vergroessern, Werte filtern und Zieloberflaeche pruefen.
Zu kurze Distanz	Blindbereich unter etwa 2 cm verlassen.
ESP32 reagiert instabil	Echo nicht direkt anschliessen; 5-V-Pegel reduzieren.
Mehrere Sensoren stoeren	Sensoren zeitlich nacheinander triggern.

Messaufbau verbessern

- Sensor senkrecht auf eine grosse, feste Flaeche ausrichten.
- Beide Schallwandler frei lassen und nicht hinter Stoff montieren.
- Fuer reproduzierbare Ergebnisse Temperatur und Einbauposition konstant halten.
- Bei Fuellstandsmessung Kondenswasser und direkten Fluessigkeitskontakt vermeiden.

[HC-SR04 bei MAKERSHOP.DE](#)

[HC-SR04 Moduldatenblatt](#)

[Arduino HC-SR04 Bibliothek](#)