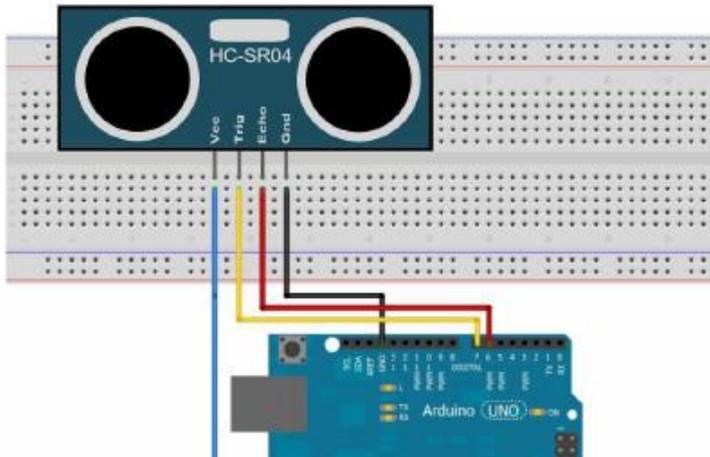


Beschreibung:

Der Pin „trigger“ bekommt vom Mikrocontroller-Board ein kurzes Signal (5V), wodurch eine Schallwelle vom Ultraschallsensor ausgelöst wird. Sobald die Schallwelle gegen eine Wand oder sonstigen Gegenstand stößt, wird sie reflektiert und kommt wieder zum Ultraschallsensor zurück. Sobald der Sensor diese zurückgekehrte Schallwelle erkennt, sendet der Sensor auf dem „echo“ Pin ein 5V Signal an das Mikrocontroller- Board. Dieser misst dann lediglich die Zeit zwischen dem Aussenden und der Rückkehr der Schallwelle und rechnet diese Zeit dann in eine Entfernung um. Der verwendete Befehl für die Zeitmessung lautet „PulsIn“, wobei der vorletzte Buchstabe ein großes „i“ ist.

Beschaltung:**Bild:****Sketch:**

```
int trigger=7; //Trigger-Pin des Ultraschallsensors an Pin7 des Arduino-Boards
int echo=6; //Echo-Pin des Ultraschallsensors an Pin6 des Arduino-Boards
long dauer=0; // Variable, unter der die Zeit gespeichert wird, die eine Schallwelle bis zur Reflektion und zurück
//braucht
long entfernung=0; //variable, unter der die berechnete Entfernung gespeichert wird.
void setup() //Beginn des Setups
{
  Serial.begin (9600); pinMode(trigger, OUTPUT); //Trigger-Pin ist ein Ausgang
  pinMode(echo, INPUT); //Echo-Pin ist ein Eingang
}
void loop()
{
  digitalWrite(trigger, LOW); //Hier nimmt man die Spannung für kurze Zeit vom Trigger-Pin, damit man später beim
  //senden desTrigger-Signals ein rauschfreies Signal hat.
  delay(5); //Dauer: 5 Millisekunden
  digitalWrite(trigger, HIGH); //Jetzt sendet man eine Ultraschallwelle los.
  delay(10); //Dieser „Ton“ erklingt für 10 Millisekunden.
  digitalWrite(trigger, LOW); //Dann wird der „Ton“ abgeschaltet.
  dauer = pulseIn(echo, HIGH); //Mit dem Befehl „pulseIn“ zählt der Mikrokontroller die Zeit in Mikrosekunden
  entfernung = (dauer/2) * 0.03432; // Man teilt zunächst die Zeit durch zwei (Weil man ja nur eine Strecke
  //berechnen. Den Wert multipliziert man mit der Schallgeschwindigkeit in der Einheit Zentimeter/Mikrosekunde und
  //erhält dann den Wert
  if (entfernung >= 500 || entfernung <= 0) //Wenn die gemessene Entfernung über 500 oder unter 0cm liegt,...
  {
    Serial.println("Kein Messwert"); //Ausgabe „KeinMesswert“, weil Messwerte evtl. falsch oder ungenau sind.
  }
  else //Ansonsten...
  {
    Serial.print(entfernung); //...soll der Wert der Entfernung an den serial monitor hier ausgegeben werden.
    Serial.println(" cm"); //Hinter dem Wert der Entfernung soll auch am Monitor die Einheit "cm" angegeben werden.
  }
  delay(1000); //Das delay von einer Sekunde sorgt in ca. jeder neuen Sekunde für einen neuen Messwert.
}
```