### 1. Lösung

|  |  |
| --- | --- |
|  | Beschreibungen |
| **int trigger=7;** | Der Variablen trigger wird der Wert 7 zugewiesen. (Trigger-Pin des Ultraschallsensors an Pin 7 des Arduino-Boards) |
| int echo=6; | Der Variablen echo wird der Wert 6 zugewiesen. (Echo-Pin des Ultraschallsensors an Pin 6 des Arduino-Boards) |
| long dauer=0; | Unter der Variablen dauer wird die Zeit gespeichert wird, die eine Schallwelle bis zur Reflektion und zurück benötigt. Startwert ist 0. |
| long entfernung=0; | Unter der Variablen entfernung wird die berechnete Entfernung gespeichert . Anstelle von „int“ steht hier vor den beiden Variablen „long“. Das hat den Vorteil, dass eine größere Zahl gespeichert werden kann. Nachteil: Die Variable benötigt mehr Platz im Speicher. |
| void setup() { | Das Setup beginnt. |
| Serial.begin (9600); | Die serielle Kommunikation wird gestartet, damit man sich die Werte auf dem seriellen Monitor ansehen kann. |
| pinMode(trigger, OUTPUT); | Der Trigger-Pin (Pin 7) ist ein Ausgang. |
| pinMode(echo, INPUT); | Der Echo-Pin (Pin 6) ist ein Eingang. |
| } | Das Setup wird beendet. |
| void loop(){ | Das Hauptprogramm beginnt. |
| digitalWrite(trigger, LOW); | Die Spannung wird für kurze Zeit vom Trigger-Pin genommen, damit man beim Senden des Trigger-Signals ein rauschfreies Signal hat. |
| delay(5); |  |
| digitalWrite(trigger, HIGH); | Es wird eine Ultraschallwelle gesendet. |
| delay(10); | Das Programm macht eine Pause von 10 Millisekunden. |
| digitalWrite(trigger, LOW); | Der „Ton“ wird abgeschaltet. |
| dauer = pulseIn(echo, HIGH); | Der Mikrokontroller zählt die Zeit in Millisekunden, bis der Schall zum Ultraschallsensor zurückkehrt. |
| entfernung = (dauer/2) \* 0.03432; | Die ermittelte Zeit wird durch zwei geteilt, weil man nur eine Strecke berechnen möchte = „Hin“. Dann multipliziert man mit der Schallgeschwindigkeit in der Einheit Zentimeter/Mikrosekunde und erhält den Wert in Zentimetern. |
| Serial.print(entfernung); | Der Wert der Entfernung wird auf dem seriellen Monitor ausgegeben, der Cursor bleibt in der Zeile. |
| Serial.println(" cm"); | Hinter dem Wert der Entfernung wird am seriellen Monitor die Einheit "cm" angegeben. |
| delay(1000); } | Das Programm macht eine Pause von 1 Sekunde. Dadurch gibt es etwa in jeder Sekunde einen neuen Messwert. |