

Ultrasonie parkeerhulp met Arduino Uno

Dogan Ibrahim (Groot-Brittannië)

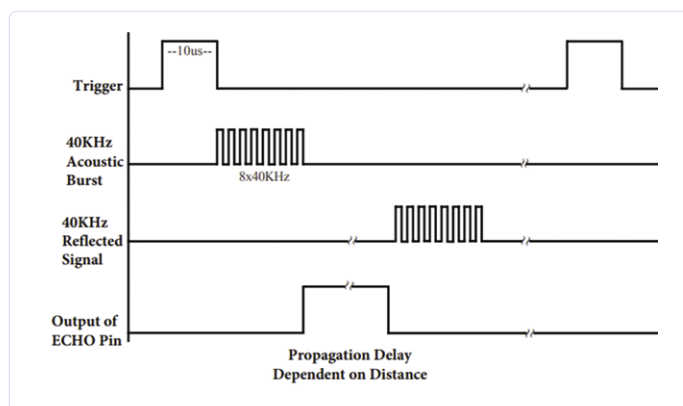
In dit project gebruiken we de ultrasonie sensormodule KY-050 in combinatie met een Arduino en een actieve zoemer KY-012 om te helpen bij het achteruit inparkeren of fileparkeren van onze auto. Naarmate de afstand tot andere auto's kleiner wordt, piept de zoemer sneller om de bestuurder te waarschuwen dat de achterkant van het voertuig dichterbij andere objecten komt. Het doel van dit kleine project is om te laten zien hoe de ultrasonie sensormodule kan worden gebruikt in een project om afstanden te meten.

De KY-050 is een 4-pins module die is afgebeeld in **figuur 1**. Deze module maakt gebruik van de ultrasonie zender/ontvanger-hardware type HC-SR04 en heeft de volgende eigenschappen:

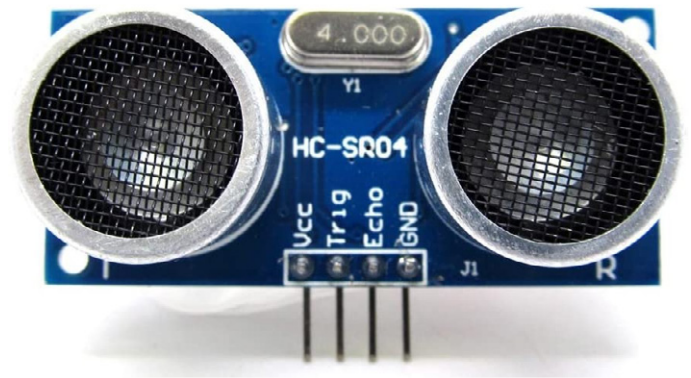
- > voedingsspanning: 5 V
- > stroomopname: 2 mA
- > detectie-afstand: 2...450 cm
- > ingangs-triggersignaal: 10 µs TTL
- > sensorhoek: <15 graden

De aansluitingen van de KY-050 hebben deze benamingen: **Vcc**: voedingsspanning; **Trig**: trigger-ingang; **Echo**: echo-uitgang; **Gnd**: massa. De (sequentiële) werking van de KY-050 ultrasonie sensor-module is als volgt (zie **figuur 2**):

- > een triggerpuls van 10 µs wordt naar de module gestuurd;
- > de module zendt vervolgens acht 40kHz-blokgolven naar het doel en maakt de Echo-pin hoog;
- > het programma start een timer;



Figuur 2. Werking van de ultrasonie sensormodule.



Figuur 1. De KY-050 ultrasonie module.

- > het signaal wordt door het doel terug naar de module weerkaatst;
- > wanneer het weerkaatste signaal door de module wordt ontvangen, wordt de Echo-pin laag;
- > de timer wordt gestopt;
- > de tijdsduur tot de ontvangst van het echosignaal wordt berekend; het resultaat is een maat voor de afstand tot het doel.

De afstand tot het object wordt als volgt berekend:

$$\text{afstand tot object (in m)} = (\text{tijdsduur tot echo in seconden} \times \text{geluidssnelheid}) / 2$$

De geluidssnelheid bedraagt ongeveer 340 m/s, of 0,034 cm/µs, en dus

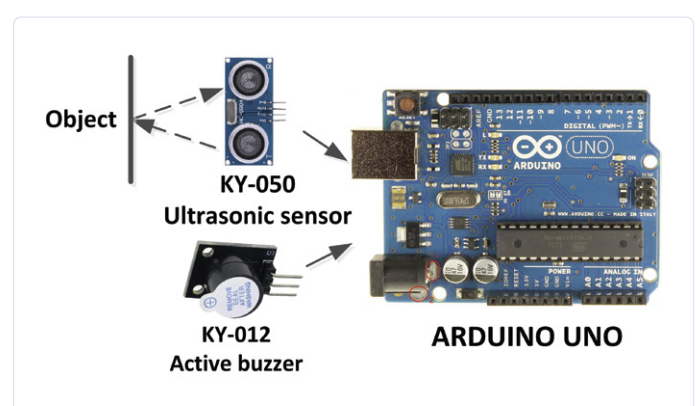
$$\text{afstand tot object (in cm)} = (\text{tijdsduur tot echo in } \mu\text{s}) \times 0,034 / 2$$

of

$$\text{afstand tot object (in cm)} = (\text{tijdsduur tot echo in } \mu\text{s}) \times 0,017$$

Als de tijdsduur tussen uitzenden van het signaal en het ontvangen van de echo bijvoorbeeld 294 microseconden bedraagt, bedraagt de afstand tot het object:

$$\text{afstand tot object (in cm)} = 294 \times 0,017 = 5 \text{ cm}$$



Figuur 3. blokschema van het project.

Listing 1. 'Ultrason' programma voor de Arduino.

```

/*****
 *   ULTRASONIC REVERSE PARKING WITH BUZZER
 *   =====
 *****/
int trig = 2;           // trig pin
int echo = 3;           // echo pin
int buzzer = 4;         // buzzer
int dely;

long tim;
float distance;

void setup ()
{
  Serial.begin(9600);
  pinMode(trig, OUTPUT);    // trig is output
  pinMode(echo, INPUT);     // echo is input
  pinMode(buzzer, OUTPUT);  // buzzer is output
  digitalWrite(buzzer, LOW); // buzzer OFF
}

void loop ()
{
  digitalWrite(trig, LOW);    // clear trig
  delayMicroseconds(5);      // 5us delay

  digitalWrite(trig, HIGH);   // set trigger HIGH
                             // for 10us
  delayMicroseconds(10);     // 10us delay
  digitalWrite(trig, LOW);    // remove trigger

  tim = pulseIn(echo, HIGH); // read the echo
  distance = tim * 0.034 / 2; // calculate distance
  Serial.println(distance);   // display distance

  //
  // Set the delay depending on the distance
  // to the object
  //
  if(distance > 100)
    dely = 0;
  else if(distance > 70 && distance < 90)
    dely = 600;
  else if(distance > 50 && distance < 70)
    dely = 400;
  else if(distance > 30 && distance < 50)
    dely = 300;
  else if(distance > 10 && distance < 30)
    dely = 200;
  else if(distance < 10)
    dely = 10;

  if(distance < 100)          // if less than 100cm
  {
    digitalWrite(buzzer, HIGH); // buzzer ON
    delay(dely);                // delay
    digitalWrite(buzzer, LOW);  // buzzer OFF
    delay(dely);                // delay
  }
}

```

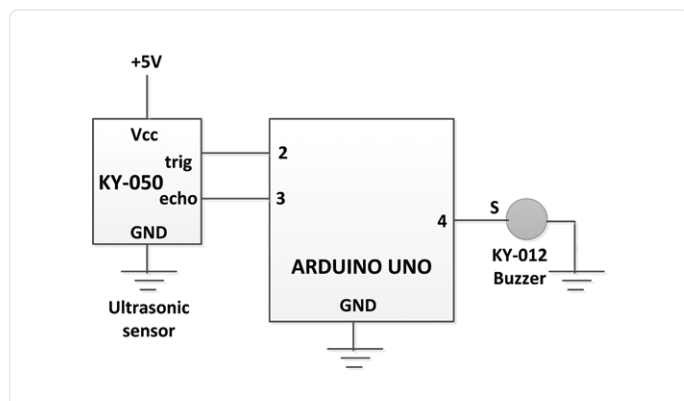
Figuur 3 toont het blokschema van het project (in 'embedded platform'-stijl) en **figuur 4** het aansluitschema. De volgende verbindingen moeten worden gemaakt tussen KY-050, KY-012 en Arduino Uno:

Arduino Uno-Pin	KY-050-pin	KY-012-pin
2	trig	
3	echo	
4		S
GND	GND	GND
+5V	Vcc	

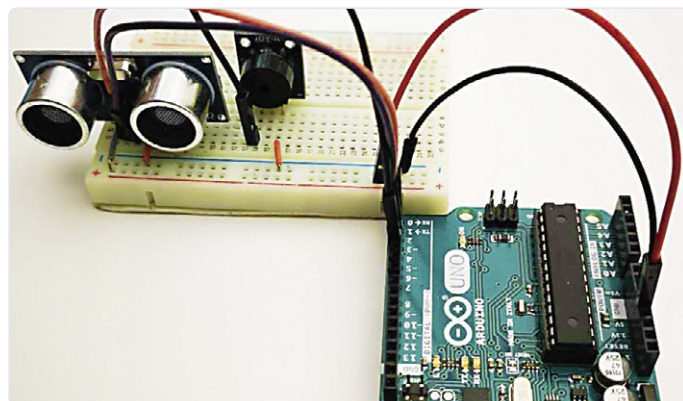
De KY-012 buzzermodule is direct verbonden met Arduino Uno poort-pin 4. Het verdient aanbeveling om een serieweerstand van 100 Ω te gebruiken om de uitgangsstroom van de poort te beperken, vooral als andere apparaten ook stroom van de poortpinnen betrekken. Verwijzend naar **listing 1** worden eerst aan **trig**, **echo** en **buzzer**

poortnummers 2, 3 en 4 toegewezen. Binnen de **setup**-routine wordt **buzzer** geconfigureerd als uitgang en wordt deze uitgeschakeld. Ook zijn **trig** en **echo** geconfigureerd als respectievelijk uitgang en ingang. Binnen de hoofd-programmalus wordt een triggerpuls verstuurd en wordt het statement **pulseIn** gebruikt om het echosignaal in te lezen. De afstand tot het object voor de ultrasone sensor wordt vervolgens berekend in centimeters en opgeslagen in een variabele die **distance** wordt genoemd. De variabele **dely** (sic) wordt ingesteld op verschillende waarden, afhankelijk van de berekende afstand. **dely** wordt kleiner naarmate de sensor dichterbij het object komt. Ten slotte wordt de zoemer geactiveerd met een duur ter waarde van de variabele **dely**. Het netto resultaat is dat de snelheid waarmee de pieptonen op elkaar volgen wordt verhoogd naarmate de sensor dichterbij het object komt. Merk op dat de afstand wordt weergegeven op de seriële monitor. Ter afsluiting toont **figuur 5** de schakeling zoals die op een breadboard is gebouwd. Niet slecht voor een zondagmiddag! 

200211-04



Figuur 4. Het schema van het project.



Figuur 5. Het prototype is op breadboard opgebouwd.