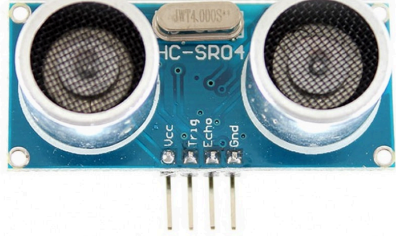


Afstandssensor

<p><u>Doel:</u></p> <ul style="list-style-type: none"> - gebruik van een ultrasone afstandssensor - meten van pulse-duur (pulseIn) 	<p><u>nodig:</u></p> <ul style="list-style-type: none"> • ultrasone afstandssensor • uitbreiding: LCD display
<p>Een ultrasone sensor meet de afstand tot een object, door een ultrasone puls uit te sturen, en de tijd te meten totdat deze weer ontvangen wordt (echolocatie). Deze techniek hebben we geleerd van vleermuizen. Zie voor meer uitleg bijv.: http://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/ http://www.micropik.com/PDF/HCSR04.pdf</p> <ul style="list-style-type: none"> • snelheid van het geluid: $v = 343 \text{ m/s} = 0,0343 \text{ cm/microsec}$ • afstand s tot object (t is tijd tot echo) $s = t * v / 2 = t * 0,0343/2$ 	 <p>De afstand tot het object is de helft van de afstand die de ultrasone puls aflegt: heen en terug.</p>
<p><u>Schakeling</u></p> <p>Verbind de ultrasone sensor:</p> <ul style="list-style-type: none"> • Trig met Arduino pin D9 • Echo met Arduino pin D10 • VCC met Arduino 5V • GND met Arduino GND 	
<pre>const int trigPin = 9; const int echoPin = 10; void setup() { pinMode(trigPin, OUTPUT); pinMode(echoPin, INPUT); digitalWrite(trigPin, LOW); Serial.begin(9600); } void loop() { digitalWrite(trigPin, HIGH); delayMicroseconds(10); // HIGH pulse digitalWrite(trigPin, LOW); long t_echo = pulseIn(echoPin, HIGH); int distance= t_echo * 0.0343/2; Serial.print("Distance (cm): "); Serial.println(distance); }</pre>	<p><u>Programma</u></p> <p>trig is de input van de sensor, en dus output van de Arduino. echo is de output van de sensor.</p> <p>De ultrasone sensor wordt gestart door een puls van 10 microseconde op de trig-ingang. De sensor stuurt daarop een ultrasone puls uit, en geeft een puls op de echo-uitgang totdat de ultrasone puls ontvangen wordt.</p> <p>Met behulp van de Arduino-functie pulseIn meten we de duur van deze echo-puls van de sensor. Uit deze duur en de snelheid van het geluid bepalen we de afstand (in cm).</p>

<p><u>Opdracht 1</u></p> <p>Bouw deze schakeling, en test deze. Bekijk de uitvoer met de Serial Monitor.</p> <p>Voeg eventueel een "delay" toe zodat je deze uitvoer beter kunt volgen.</p>	<ul style="list-style-type: none"> - de bundel van de sensor is vrij klein. Ga na waar de sensor op een afstand van 10 cm nog een object ziet, en waar niet meer. - hoe nauwkeurig is deze meting? op een afstand van 5cm? 25cm? 1m? 2m?
<p><u>Opdracht 2</u></p> <p>Gebruik een LCD-display in plaats van de Serial monitor, voor het weergeven van de afstand.</p>	
<p><u>Opdracht 3</u></p> <p>De NewPing-library geef nauwkeuriger resultaten dan het gebruik van pulseIn. Gebruik deze voor een aangepaste versie van het programma hierboven.</p> <p>http://playground.arduino.cc/Code/NewPing</p>	
<p><u>Opdracht 4</u></p> <p>De functie pulseIn is een blokkerende functie. Een alternatief is het gebruik van de aanpak met timers - zie het NewPing voorbeeld Event Timer Sketch.</p> <p>https://bitbucket.org/teckel12/arduino-new-ping/wiki/Home#!event-timer-sketch</p>	<p>Bestudeer het voorbeeld.</p> <p>Experimenteer hiermee met je eigen afstandssensor-opstelling.</p> <p>Bedenk een toepassing waar je dit nodig hebt - omdat je geen blokkerende pulseIn kunt gebruiken.</p>