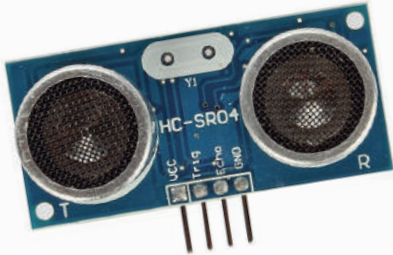


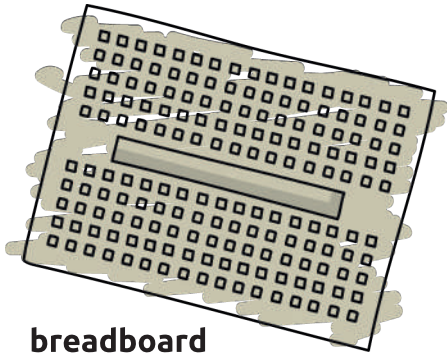
Deel 6

Rara, waar ben ik?

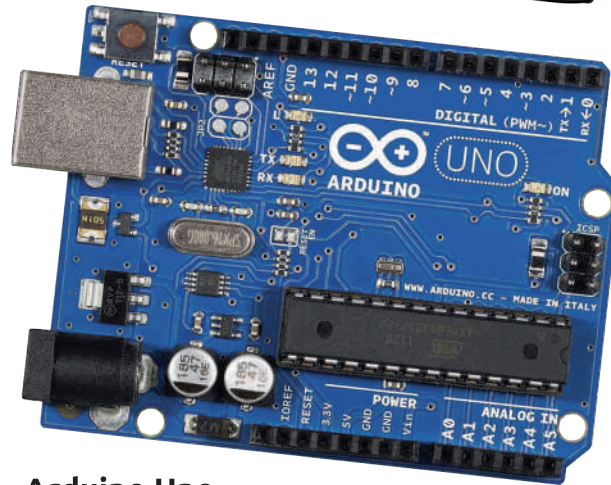
Wat heb je nodig?



Een **ultrasoon sensor** zendt ultrasonische golven uit om de afstand te meten.



breadboard



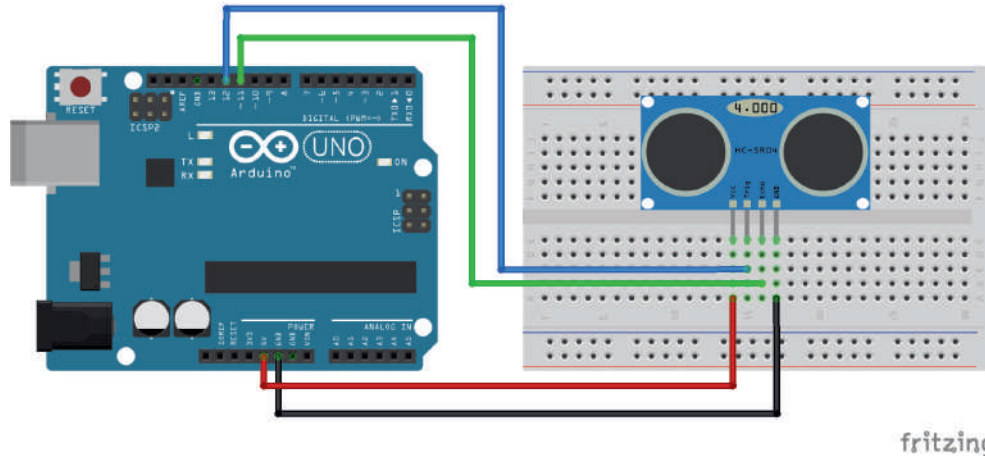
Arduino Uno



4 Jumperwires

OPDRACHT

Een LDR sensor kan licht zien, maar er bestaan nog meer sensoren! Een ultrasoon sensor kan de afstand meten. Schrijf een programma waarin je een ultrasoon sensor gebruikt.



fritzing

Verbind de vierde pin GND met GND.

Verbind de eerste pin Vcc met 5V.

Verbind de Trig pin met pin 12.

Verbind de Echopin met pin 11.

Wat ga je maken?

```
Platform Arduino Uno
digitale pin 11 Digital Input echo
digitale pin 12 Digitale output trigger

Start serial : zet snelheid op 9600 bps

Arduino doe eerst:
  Schrijf naar digital output trigger waarde HOOG
  wacht 100 microseconden

Arduino herhaal voor altijd:
  Schrijf naar digital output trigger waarde LAAG
  Lees een HOOG puls op pin # 11 wacht maximaal 2000 microseconden
  stel duurtijd in op duurtijd ÷ 29
  stel afstand in op afstand + 2
  serial print afstand naar nieuwe lijn
  serial print " cm " naar nieuwe lijn
  wacht 1000 milliseconden
```

STAPPENPLAN
op de volgende pagina's!

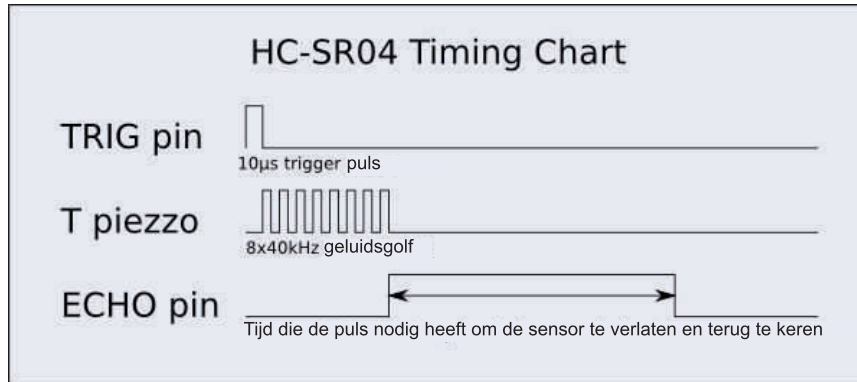
TIPS VOOR SLIMME PROGRAMMEURS

Wat is een ultrasoon afstandssensor?

De ultrasoon afstandssensor HC-SR04 werkt op basis van ultrasone geluidsgolven. De afstand tot het object kan berekend worden aan de hand van de tijd die nodig is om het weerkaatste geluid terug te ontvangen.

Met deze sensor kan je afstanden meten van 2 tot 450 centimeter met een hoek van niet meer dan 15 graden.

Hoe werkt het?



Op de trigger pin (TRIG pin) moet voor 10 µs (microseconden) een hoog signaal staan. Daarna gaat het piezo element een signaal uitzenden van 8x40 kHz. Nadat het signaal is uitgezonden, gaat het tweede piezo element "luisteren" naar het uitgezonden signaal (ECHO pin). met de tijd daartussen kan de afstand bepaald worden.

De snelheid van geluid door lucht is ca. 340 m/s of 29 µs per centimeter. De formule die we kunnen gebruiken, is dus **Aantal CM = TIJD : 29**

Maar omdat het signaal "heen" is gezonden en weer terug komt, moeten we de tijd delen door 2, de uiteindelijke formule is dan **Aantal CM = TIJD : 29 : 2**

1

- 1 Kies een Arduino Uno en voeg een digitale input en output toe.
- 2 Start de seriële monitor met het blokje uit het Comms-menu. Stel 9600 bits per seconde (bps of baud) in.

The image shows the Arduino IDE block editor interface. On the left is a category menu with the following items: In/Output, Functies, Herhalen, Logica, Wiskunde, Tekst, Variabelen, Tijd, Componenten, Licht&Geluid, Motoren, and Comms. The 'In/Output' category is highlighted with an orange box and a circled '1'. An orange arrow points from this box to a block in the workspace. The workspace contains a green 'Platform' block set to 'Arduino Uno'. Below it are two blocks: 'digitale pin 11' connected to a 'Digital Input' block with the label 'echo', and 'digitale pin 12' connected to a 'Digitale output' block with the label 'trigger'. Below these is a green 'Start serial' block with the text ': zet snelheid op 9600 bps'. The 'Comms' category in the menu is highlighted with an orange box and a circled '2'. An orange arrow points from this box to the 'Start serial' block.

2

! Kies een wachttijd in microseconden!

The screenshot shows a block-based programming environment for an Arduino Uno. On the left is a sidebar with categories: In/Output, Functies, Herhalen, Logica, Wiskunde, Tekst, Variabelen, Tijd, Componenten, and Licht&Geluid. The main workspace contains the following code blocks:

- Platform** Arduino Uno
- digitale pin** 11 → **Digital Input** echo
- digitale pin** 12 → **Digitale output** trigger
- Start serial** : zet snelheid op 9600 bps
- Arduino doe eerst:** (loop start)
- Schrijf naar digital output** trigger waarde HOOG
- wacht** 100 **microseconden** (highlighted with a red box and a red exclamation mark icon)
- Schrijf naar digital output** trigger waarde LAAG

3

Laat de afstand berekenen door de duurtijd te registreren van de echo pin op pin 11.

The screenshot shows the Arduino IDE interface with the following code blocks:

- Platform: Arduino Uno
- digitale pin 11: Digital Input echo
- digitale pin 12: Digitale output trigger
- Start serial: zet snelheid op 9600 bps
- Arduino doe eerst:
- Arduino herhaal voor altijd:
 - Schrijf naar digital output trigger waarde HOOG
 - wacht 100 microseconden
 - Schrijf naar digital output trigger waarde LAAG
 - stel duurtijd in op Lees een HOOG puls op pin # 11 wacht maximaal 2000 microseconden
 - stel afstand in op duurtijd ÷ 29
 - stel afstand in op afstand ÷ 2

Hoe bereken je de maximum duurtijd?

De snelheid van het geluid is $29 \mu\text{s}/\text{cm}$.
In $2000\mu\text{s}$ leg het dus

$2000 \mu\text{s} / (29 \mu\text{s}/\text{cm}) = 68,6 \text{ cm af.}$
Omdat het geluid heen en terug gaat, kan je dus tot $68,6 \text{ cm} / 2 = 34,3 \text{ cm}$ ver objecten zien. Alles wat verder is, daar wachten we niet op.

Hoe lang moet je wachten om een object op afstand A te herkennen?

De formule is $2 * A * 29 \mu\text{s}/\text{cm}$
Voorbeeld: Je wil een object tot op 100 cm kunnen waarnemen en niet verder.

$2 * 100 \text{ cm} * 29 \mu\text{s}/\text{cm} = 5800 \mu\text{s}$
De wachttijd is dus 5800 μs .

4 Voeg twee print-blokjes uit het Comms-menu toe.

! Vink bij het tweede print-blokje een nieuwe lijn aan!

The screenshot shows the Arduino IDE block editor interface. On the left is a category menu with 'Comms' highlighted in an orange box. The main workspace contains the following blocks:

- Arduino doe eerst:** (Purple block)
- Arduino herhaal voor altijd:** (Purple loop block containing:
 - Schrijf naar digital output** (Blue block): trigger, waarde: HOOG
 - wacht** (Green block): 100 microseconden
 - Schrijf naar digital output** (Blue block): trigger, waarde: LAAG
 - stel duurtijd in op** (Purple block): Lees een HOOG puls op pin # 11 wacht maximaal 2000 microseconden
 - stel afstand in op** (Purple block): duurtijd ÷ 29
 - stel afstand in op** (Purple block): afstand ÷ 2
 - serial print** (Green block): afstand naar nieuwe lijn (with a red exclamation mark icon)
 - serial print** (Green block): " cm " naar nieuwe lijn (checked)
 - wacht** (Green block): 1000 milliseconden

An orange arrow points from the 'Comms' menu to the second 'serial print' block.

KLAAR!

Nu moet je enkel nog de code op je Arduino zetten.

Platform **Arduino Uno**

digitale pin **11** Digital Input **echo**

digitale pin **12** Digitale output **trigger**

Start **serial** : zet snelheid op **9600** bps

Arduino doe eerst:

Arduino herhaal voor altijd:

Schrijf naar digital output **trigger** waarde **HOOG**

wacht **100** microseconden

Schrijf naar digital output **trigger** waarde **LAAG**

stel **duurtijd** in op **Lees een HOOG puls op pin # 11** wacht maximaal **2000** microseconden

stel **afstand** in op **duurtijd / 29**

stel **afstand** in op **afstand / 2**

serial print **afstand** naar nieuwe lijn

serial print **" cm "** naar nieuwe lijn

wacht **1000** milliseconden

```
Arduino Source Code

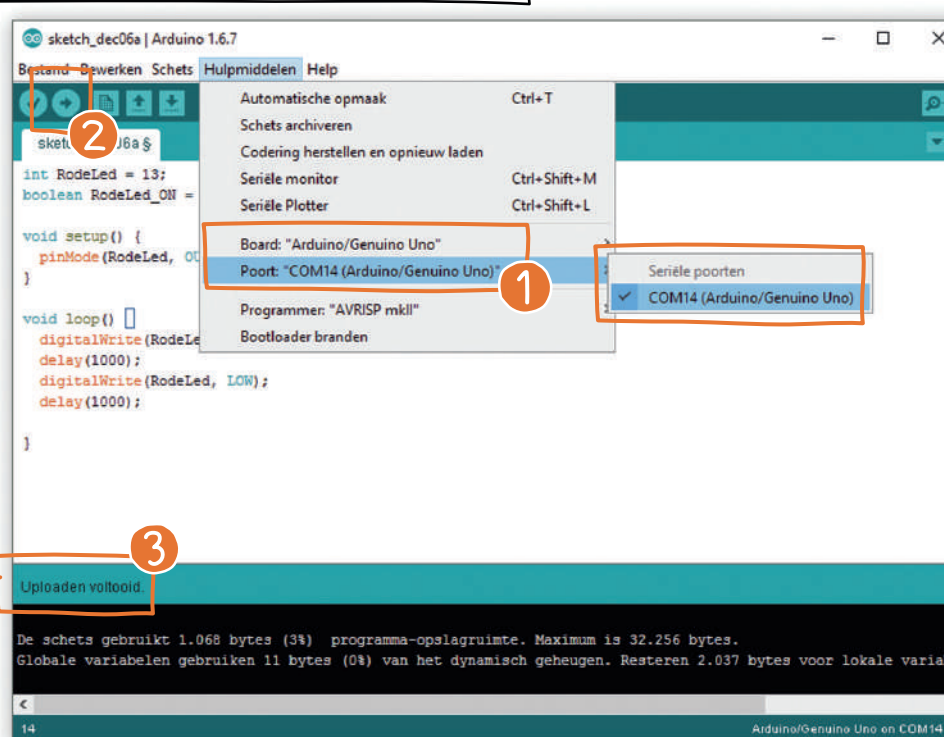
int duurtijd;
int afstand;
int echo = 11;
int trigger = 12;

void setup() {
  pinMode(echo, INPUT);
  pinMode(trigger, OUTPUT);
  Serial.begin(9600);
  pinMode(11, INPUT);
}

void loop() {
  digitalWrite(trigger, HIGH);
  delayMicroseconds(100);
  digitalWrite(trigger, LOW);
  duurtijd = pulseIn(11, HIGH, 2000);
  afstand = duurtijd / 29;
  afstand = afstand / 2;
  Serial.print(afstand);
  Serial.println("cm");
  delay(1000);
}
```

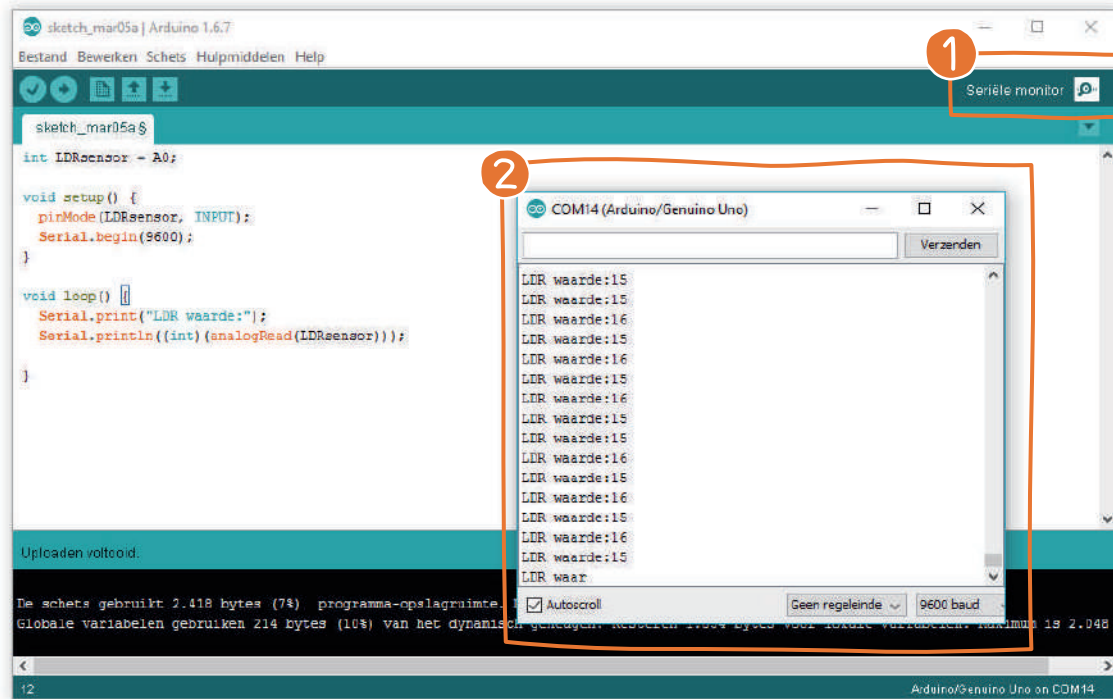
- 5
- 1 Verbind je Arduino met de computer. Selecteer 'Arduino Uno' en de juiste poort in de Arduino IDE.
 - 2 Klik op de pijl om de code op de Arduino te laden.
 - 3 Zie je 'Uploaden voltooid', dan zou de ultrasone sensor nu moeten werken.

Zie je een foutmelding?
Geen paniek, vraag hulp
aan de begeleider.



6

- 1 Open de seriële monitor.
- 2 Test de ultrasone sensor.



OEFENINGEN

Speel met de blokjes!

1

Gebruik de ultrasone sensor om de ogen van UGO of TeSS aan te sturen. Schrijf een programma waarmee de LEDs aangaan zodra er iemand dichterbij komt dan 80 cm voorbijloopt.

We lossen het samen op:

Eerst berekenen we de duurtijd van het signaal.

$$\text{Formule: } \text{duurtijd} = 2 * A * 29 \mu\text{s/cm}$$
$$A = 80 \text{ cm}$$

$$\text{Oplossing: } \text{duurtijd} = 2 * 80 \text{ cm} * 29 \mu\text{s/cm} = 4640 \mu\text{s}$$

Gebruik dat in de blokkencode!

```

Platform Arduino Uno
digitale pin 11 Digital Input echo
digitale pin 12 Digitale output trigger
digitale pin 8 LED Oog1 been polariteit plus
digitale pin 9 LED Oog2 been polariteit plus

Start serial : zet snelheid op 9600 bps

```

```

Arduino doe eerst:
Arduino herhaal voor altijd:
  Schrijf naar digital output trigger waarde HOOG
  wacht 100 microseconden
  Schrijf naar digital output trigger waarde LAAG
  stel duurtijd in op Lees een HOOG puls op pin # 11 wacht maximaal 4640 microseconden
  stel afstand in op duurtijd +- 29
  stel afstand in op afstand +- 2
  serial print afstand naar nieuwe lijn
  serial print " cm " naar nieuwe lijn
  als afstand < 80
    voer uit Zet LED Oog1 AAN
    Zet LED Oog2 AAN
  anders Zet LED Oog1 UIT
    Zet LED Oog2 UIT
  wacht 1000 milliseconden

```

IF-ELSE blok (Als - Voer uit - Anders)
 Vaak moet er enkel iets gebeuren **als** iets anders waar is. Daarvoor gebruiken we **if-else code**. Eerst geef je de **voorwaarde** in. Daarna schrijf je wat er uitgevoerd moet worden als de voorwaarde waar is. Je kan ook code opgeven voor het geval er niet aan de voorwaarde voldaan wordt. Je wil dus zeggen:
Als iets waar is, **dan** voer je dit uit, **anders** doe je dat.

In onze blokken code:
 Als de afstand kleiner is dan 80 cm, zet LED Oog1 en LED Oog2 aan, anders zet LED Oog1 en LED Oog2 uit.



De Creatieve **STEM**

