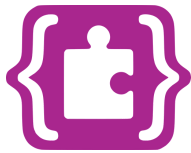


# MAAK EEN THEREMIN

MET DE AFSTANDSENSOR

#R1AS08



## Beschikbaar op



## Vereiste voorkennis

- R1AS04 - Simpele lichtsensor
- R1AS06 - Morsecode

## Materiaal

- 1 Programmeerbord "**STM32 IoT Node Board**"
- 1 piëzo zoemer of een luidspreker
- 1 Breadboard
- Jumper draden

## Wat is het?

Een theremin is een elektronisch muziekinstrument dat kan worden bespeeld zonder het aan te raken. Het oorspronkelijke concept is gebaseerd op het gebruik van twee antennes om de positie van de handen te detecteren. Eén antenne wordt gebruikt voor het volume, en de andere voor de toonhoogte.

## Duur

30 minuten

## Moeilijkheidsgraad

Gevorderd

## LEERDOELEN

- Gebruik een afstandssensor en begrijp hoe die werkt
- Muziek maken met een heel vreemd instrument
- Gebruik de map-functie om een getal van het ene bereik naar het andere te transformeren



# MAAK EEN THEREMIN MET DE AFSTANDSENSOR



Een theremin is een elektronisch muziekinstrument dat bespeeld wordt door de afstand tussen de handen en twee antennes te variëren. De speler raakt het instrument niet aan. De rechterhand beïnvloedt de toonhoogte en de linkerhand het geluidsvolume. Doordat minimale bewegingen al hoorbaar zijn, klinkt het instrument bijzonder expressief.

Onze versie zal eenvoudiger zijn: we zullen alleen de toonhoogte regelen met de afstandssensor. Het volume zal vooraf bepaald worden. **Laten we muziek maken!**

Bronnen: <https://en.wikipedia.org/wiki/Theremin>, <https://youtu.be/x0NVb25p1oU>



## STAP 1 - MAAK HET



### Bedrading zoemer/luidspreker

In theorie is een zoemer niet gepolariseerd (dat betekent dat er geen "+" of "-" is), maar vaak heb je een zwarte en een rode draad of tekens ("+" en/of "-") op het apparaat. Als dat het geval is sluit je de (rode) draad aan de "+"-zijde van de zoemer aan op pin **D3** en de andere op pin **GND**.

Als er geen kleur of indicatie is sluit je de ene draad aan op pin **D3** en de andere op pin **GND**.

### Sluit het bord aan op de computer

Sluit het bord met uw USB-kabel aan op je computer via de **micro-USB ST-LINK connector** (in de rechterhoek van het bord). Je zou een nieuwe schijf genaamd **DIS\_L4IOT** op je computer moeten zien verschijnen. Dit station wordt gebruikt om het bord te programmeren door een binair bestand te kopiëren.

### Open MakeCode

Ga naar de [Let's STEAM MakeCode editor](https://makecode.lets-steam.eu). Maak op de startpagina een nieuw project aan door op de knop "Nieuw Project" te klikken. Geef je project een naam (zodat je later terug kan keren naar deze opdracht) en start je editor.

Bron: [makecode.lets-steam.eu](https://makecode.lets-steam.eu)

### Programmeer je bord

Kopieer de code uit de sectie "**Codeer het**" hieronder en plak deze in de MakeCode Javascript Editor. Indien je dit nog niet gedaan hebt, geef je nu best naam aan je project en klik je op de "**Downloaden**" knop. Kopieer het binaire bestand vervolgens naar de schijf op je computer met de naam **DIS\_L4IOT** en wacht tot het lichtje op het bord stopt met knipperen. Je programma zal nu worden uitgevoerd.

### Uitvoeren, wijzigen, spelen

Het programma zal automatisch uitgevoerd worden telkens je het opslaat of het bord reset (druk daarvoor op de knop met het label RESET). Probeer het voorbeeld te begrijpen en te wijzigen.

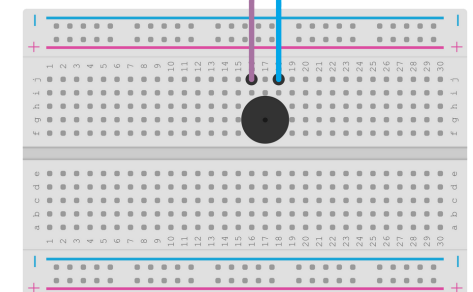
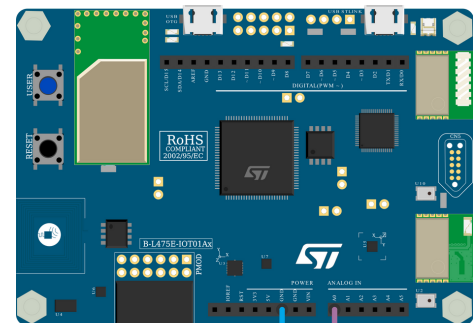
1

2

3

4

5



Bedrading zoemer/luidspreker



## STAP 2 – CODEER HET

```
let distance = 0
forever(function () {
  // Krijg afstand
  distance = input.distance(DistanceUnit.Millimeter)

  if (distance > 500) {
    // Zet de afstand om in frequentie
    let note = Math.map(distance, 0, 500, 440, 830)
    music.ringTone(note)
  } else {
    music.stopAllSounds()
  }
})
```

### Variabelen

In dit programma zijn er 2 variabelen. De variabele **distance** wordt gebruikt om de afstand te kunnen bewaren en zo een bepaalde toon te kunnen spelen. De tweede variabele is **note**, die technisch niet noodzakelijk/verplicht is, maar helpt om elke stap van het programma beter te begrijpen. Het bevat de omzetting van de afstand naar de noten.

### Afstand meten

Een variabele gebruiken om afstand te bewaren is geweldig, maar weten hoe je de afstand meet is beter! Hiervoor gebruiken we de functie **input.distance(DistanceUnit.Millimeter)**. De parameter **DistanceUnit.Millimeter** geeft aan de functie aan dat we het resultaat in millimeters willen (1 meter = 1.000 millimeter).

### Voorwaarde

De voorwaarde, **if (distance > 500) { ... }**, geeft de informatie dat we alleen een geluid afspelen als de gemeten afstand kleiner of gelijk is aan 500 millimeter.

### Zet de afstand om in frequentie

Het belangrijkste deel is de **conversie**. Om dit te doen, gebruiken we een wiskundige functie genaamd **map**. Deze functie zet een waarde om van één bereik naar een ander. In dit geval wordt de waarde omgerekend van **afstandsbereik** naar **frequentiebereik**. Zoals je in de code hierboven kan zien heeft deze functie vijf parameters, namelijk: **map(waarde, in\_min, in\_max, out\_min, out\_max)**. We bekijken ze één voor één:

- **waarde**: de waarde die we gaan omzetten (distance)
- **in\_min**: De minimumwaarde van het bereik van de input (afstand)
- **in\_max**: de maximumwaarde van het bereik van de input (afstand)
- **out\_min**: de minimumwaarde van het bereik van de output (frequentie)
- **out\_max**: de maximumwaarde van het bereik van de output (frequentie)

Nu is het duidelijker wat deze lijn code doet: de afstand die zit opgeslagen in de variabele **distance** (met een bereik van 0 mm tot 500 mm) omzetten in frequentie (met een bereik van 440 Hz tot 830 Hz).



**De gekozen frequenties zijn niet willekeurig, het frequentiebereik van 440Hz tot 830Hz vertegenwoordigt een octaaf. Dit betekent dat je alle noten kan vinden: LA, SI, DO, RE, MI, FA, SOL**

Nu we een frequentie hebben is het tijd om muziek temaken, gewoon met de **music.ringTone(note)**.



## STAP 3 - VERBETER HET



Verander de waarden van de **map** om **octaven** en/of afstand toe te voegen om je muziek te verbeteren.

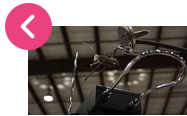
Probeer een **potentiometer** toe te voegen om het volume te regelen.



## VERDER GAAN



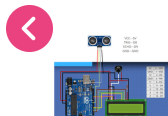
**Theremin** - Leer meer over de geschiedenis, de werkingsprincipes en het gebruik van de theremin.  
<https://en.wikipedia.org/wiki/Theremin>



**LED Ring Distance Sensor** - Ontdek een leuk project, dat zal eindigen in een alternatieve parkeersensor.  
<https://www.instructables.com/LED-Ring-Distance-Sensor/>



**Water Level Detector** - Ontdek ultrasone sensoren die elektrische energie omzetten in akoestische golven.  
<https://www.instructables.com/Water-Level-Detector-2/>



**Cat Feeder** - Gebruik een ultrasone sensor om een automatisch kattenvoedersysteem te maken.  
<https://www.instructables.com/Cat-Feeder/>



### Gekoppelde activiteitenbladen

**R1AS05 -  
Potentiometer**

