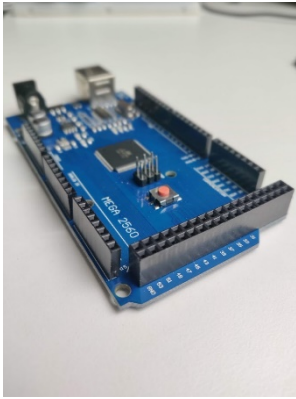


Arduino muziekspeler



Voor deze opdracht zal gebruik gemaakt worden van 'Arduino', dat is een kleine computer om elektronica-projecten mee aan te sturen.

Het is de bedoeling om, aan de hand van een code, een bestaand of verzonden liedje te laten afspelen door te programmeren met de Arduino.



Doelgroep

Deze opdracht is geschikt voor Tieners (10 tot 12-jarigen).



Benodigheden

Deze materialen en grondstoffen heb je nodig:

Materialen	Grondstoffen
<ul style="list-style-type: none">• Arduino (maakt niet uit welk model)• Laptop met Arduino IDE software• Buzzer• 2x male-female kabels• USB kabel	/

Je hebt geen Arduino, maar wenst er wel graag 1 aan te kopen?

Dit kan via bol.com, de site www.elektronicavoorjou.nl en tal van andere. De kostprijs hiervoor begint rond de €30,00 en kan afhankelijk van het type oplopen tot meer dan €100,00 . Er bestaan tal van pakketten.

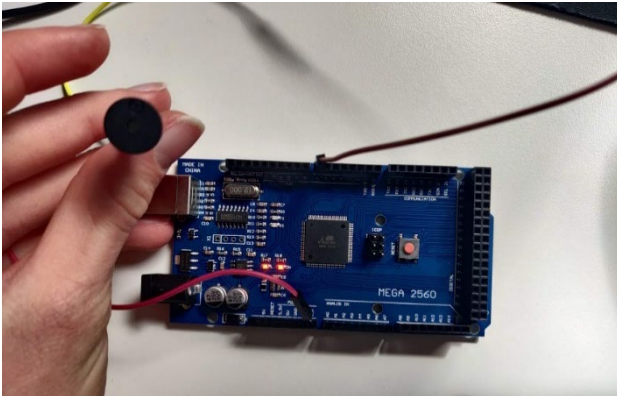
Je installeerde nog niet eerder de Arduino IDE software (gratis te downloaden)? Volg dan deze stappen:

1. Ga naar de officiële site: www.arduino.cc
2. Selecteer het tabblad "Software"

Maak de keuze "code online" (hier hoeft je niet voor te downloaden) of "getting started" (hier zal je het programma wel downloaden).

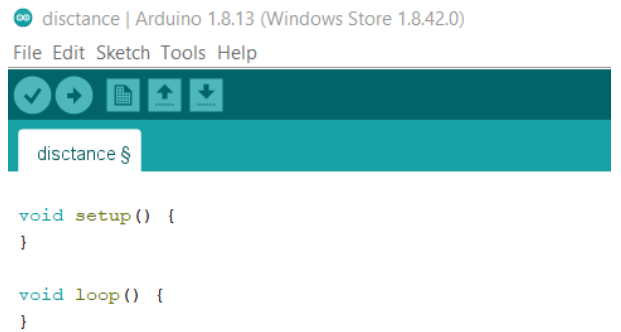


Stap 1



Verbind de buzzer met de GND pin en de andere aan kabel aan pin nummer 7

Stap 2



We starten de software Arduino IDE op

```
distance
int tempo = 108;

int buzzer = 7;

// Bijvoorbeeld noot C5 is een achtste noot, dan zet je in de array:
// NOTE_G5, 8
// NOTE_G5 slaat op de noot hoogte
// 8 slaat op de lengte van de noot.

int melody[] = {
  NOTE_A5,8, NOTE_A5,8, NOTE_A5,8, //1
  NOTE_F5,2, NOTE_C6,2,
  NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F6,2, NOTE_C6,4,
  NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F6,2, NOTE_C6,4,
  NOTE_A5,8, NOTE_A5,8, NOTE_A5,8, NOTE_G5,2, NOTE_C5,8, NOTE_C5,8, NOTE_C5,8,
  NOTE_F5,2, NOTE_C6,2,
  NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F6,2, NOTE_C6,4,

  NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F6,2, NOTE_C6,4, //8
  NOTE_A5,8, NOTE_A5,8, NOTE_A5,8, NOTE_G5,2, NOTE_C5,-8, NOTE_C5,16,
  NOTE_D5,-4, NOTE_D5,8, NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F5,8,
  NOTE_F5,8, NOTE_G5,8, NOTE_A5,8, NOTE_G5,4, NOTE_D5,8, NOTE_E5,4,NOTE_C5,-8, NOTE_C5,16,
  NOTE_D5,-4, NOTE_D5,8, NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F5,8,

  NOTE_C6,-8, NOTE_G5,16, NOTE_G5,2, REST,8, NOTE_C5,8, //13
  NOTE_D5,-4, NOTE_D5,8, NOTE_A5,8, NOTE_A5,8, NOTE_G5,8, NOTE_F5,8,
  NOTE_F5,8, NOTE_G5,8, NOTE_A5,8, NOTE_G5,4, NOTE_D5,8, NOTE_E5,4,NOTE_C6,-8, NOTE_C6,16,
  NOTE_F6,4, NOTE_DS6,8, NOTE_CS6,4, NOTE_C6,8, NOTE_AS5,4, NOTE_G5,8, NOTE_G5,4, NOTE_F5,8,
  NOTE_C6,1
};
// sizeof toont de de nummers van de bytes, iedere integer value bestaat uit 2 bytes.
// er zijn 2 value's per noot (toonhoogte en lengte zie vorige blokje commentaar in de code)
// Dus iedere noot bestaat uit 4 bytes.
int noot = sizeof(melody) / sizeof(melody[0]) / 2;

// Berekent de tijd van 1 noot in milli secondes
int tijdnoot = (60000 * 4) / tempo;

int splitser = 0, standaardnoottijd = 0;

void setup() {
  // De for lus zorgt ervoor dat je telkens de melodie zal herhalen als deze gedaan is
  // Zoals je nog weet zit in noot heel wat info over de noothoofte en de lengte van de noot
  for (int dezeNoot = 0; dezeNoot < noot * 2; dezeNoot = dezeNoot + 2) {

    // Berekent de tijd van elke noot dat hij speelt
    splitser = melody[dezeNoot + 1];
```



Stap 3

Zie grote foto hierboven

Stap 4

distance | Arduino 1.8.13 (Windows Store 1.8.42.0)

File Edit Sketch Tools Help



```
int noot = sizeof(melody) / sizeof(melody[0]) / 2;

// Berekent de tijd van 1 noot in milli secondes
int tijdnoot = (60000 * 4) / tempo;

int splitser = 0, standaardnoottijd = 0;

void setup() {
  // De for lus zorgt ervoor dat je telkens de melodie zal he
  // Zoals je nog weet zit in noot heel wat info over de noot
```

We programmeren de code voor de soundtrack van Star Wars te spelen.

```
#define NOTE_B0 31
#define NOTE_C1 33
#define NOTE_CS1 35
#define NOTE_D1 37
#define NOTE_DS1 39
#define NOTE_E1 41
#define NOTE_F1 44
#define NOTE_FS1 46
#define NOTE_G1 49
#define NOTE_GS1 52
#define NOTE_A1 55
#define NOTE_AS1 58
#define NOTE_B1 62
#define NOTE_C2 65
#define NOTE_CS2 69
#define NOTE_D2 73
#define NOTE_DS2 78
#define NOTE_E2 82
#define NOTE_F2 87
#define NOTE_FS2 93
#define NOTE_G2 98
#define NOTE_GS2 104
#define NOTE_A2 110
#define NOTE_AS2 117
#define NOTE_B2 123
#define NOTE_C3 131
#define NOTE_CS3 139
#define NOTE_D3 147
#define NOTE_DS3 156
#define NOTE_E3 165
#define NOTE_F3 175
#define NOTE_FS3 185
#define NOTE_G3 196
```

Om de code uit te voeren druk op de upload knop.

#define NOTE_GS3 208
#define NOTE_A3 220
#define NOTE_AS3 233
#define NOTE_B3 247
#define NOTE_C4 262
#define NOTE_CS4 277
#define NOTE_D4 294
#define NOTE_DS4 311
#define NOTE_E4 330
#define NOTE_F4 349
#define NOTE_FS4 370
#define NOTE_G4 392
#define NOTE_GS4 415
#define NOTE_A4 440
#define NOTE_AS4 466
#define NOTE_B4 494
#define NOTE_C5 523
#define NOTE_CS5 554
#define NOTE_D5 587
#define NOTE_DS5 622
#define NOTE_E5 659
#define NOTE_F5 698
#define NOTE_FS5 740
#define NOTE_G5 784
#define NOTE_GS5 831
#define NOTE_A5 880
#define NOTE_AS5 932
#define NOTE_B5 988
#define NOTE_C6 1047
#define NOTE_CS6 1109
#define NOTE_D6 1175
#define NOTE_DS6 1245
#define NOTE_E6 1319
#define NOTE_F6 1397
#define NOTE_FS6 1480
#define NOTE_G6 1568
#define NOTE_GS6 1661
#define NOTE_A6 1760
#define NOTE_AS6 1865
#define NOTE_B6 1976
#define NOTE_C7 2093
#define NOTE_CS7 2217
#define NOTE_D7 2349
#define NOTE_DS7 2489
#define NOTE_E7 2637
#define NOTE_F7 2794
#define NOTE_FS7 2960
#define NOTE_G7 3136
#define NOTE_GS7 3322
#define NOTE_A7 3520
#define NOTE_AS7 3729
#define NOTE_B7 3951
#define NOTE_C8 4186

```

#define NOTE_CS8 4435
#define NOTE_D8 4699
#define NOTE_DS8 4978
#define REST 0

int tempo = 108;

int buzzer = 7;

// Bijvoorbeeld noot C5 is een achtste noot, dan
// zet je in de array:
// NOTE_G5, 8
// NOTE_G5 slaat op de noot hoogte
// 8 slaat op de lengte van de noot.

int melody[] = {
  NOTE_AS4,8, NOTE_AS4,8, NOTE_AS4,8,//1
  NOTE_F5,2, NOTE_C6,2,
  NOTE_AS5,8, NOTE_A5,8, NOTE_G5,8,
  NOTE_F6,2, NOTE_C6,4,
  NOTE_AS5,8, NOTE_A5,8, NOTE_G5,8,
  NOTE_F6,2, NOTE_C6,4,
  NOTE_AS5,8, NOTE_A5,8, NOTE_AS5,8,
  NOTE_G5,2, NOTE_C5,8, NOTE_C5,8, NOTE_C5,8,
  NOTE_F5,2, NOTE_C6,2,
  NOTE_AS5,8, NOTE_A5,8, NOTE_G5,8,
  NOTE_F6,2, NOTE_C6,4,

  NOTE_AS5,8, NOTE_A5,8, NOTE_G5,8,
  NOTE_F6,2, NOTE_C6,4, //8
  NOTE_AS5,8, NOTE_A5,8, NOTE_AS5,8,
  NOTE_G5,2, NOTE_C5,-8, NOTE_C5,16,
  NOTE_D5,-4, NOTE_D5,8, NOTE_AS5,8,
  NOTE_A5,8, NOTE_G5,8, NOTE_F5,8,
  NOTE_F5,8, NOTE_G5,8, NOTE_A5,8, NOTE_G5,4,
  NOTE_D5,8, NOTE_E5,4,NOTE_C5,-8, NOTE_C5,16,
  NOTE_D5,-4, NOTE_D5,8, NOTE_AS5,8,
  NOTE_A5,8, NOTE_G5,8, NOTE_F5,8,

  NOTE_C6,-8, NOTE_G5,16, NOTE_G5,2, REST,8,
  NOTE_C5,8,//13
  NOTE_D5,-4, NOTE_D5,8, NOTE_AS5,8,
  NOTE_A5,8, NOTE_G5,8, NOTE_F5,8,
  NOTE_F5,8, NOTE_G5,8, NOTE_A5,8, NOTE_G5,4,
  NOTE_D5,8, NOTE_E5,4,NOTE_C6,-8, NOTE_C6,16,
  NOTE_F6,4, NOTE_DS6,8, NOTE_CS6,4,
  NOTE_C6,8, NOTE_AS5,4, NOTE_G5,8,
  NOTE_G5,4, NOTE_F5,8,
  NOTE_C6,1
};
// sizeof toont de de nummers van de bytes,
// iedere integer value bestaat uit 2 bytes.

```

```

// er zijn 2 value's per noot (toonhoogte en
// lengte zie vorige blokje commentaar in de code)
// Dus iedere noot bestaat uit 4 bytes.
int noot = sizeof(melody) / sizeof(melody[0]) / 2;

// Berekent de tijd van 1 noot in milli secondes
int tijdnoot = (60000 * 4) / tempo;

int splitser = 0, standaardnoottijd = 0;

void setup() {
  // De for lus zorgt ervoor dat je telkens de
  // melodie zal herhalen als deze gedaan is
  // Zoals je nog weet zit in noot heel wat info
  // over de noothoofte en de lengte van de noot
  for (int dezeNoot = 0; dezeNoot < noot * 2;
  dezeNoot = dezeNoot + 2) {

    // Berekent de tijd van elke noot dat hij speelt
    splitser = melody[dezeNoot + 1];
    if (splitser > 0) {
      // regular note, just proceed
      // Bij gewone noot hebben we deze regel
      code nodig
      standaardnoottijd = (tijdnoot) / splitser;
    } else if (splitser < 0) {
      // Gepunte noteh hebben negatieve splitser,
      // daarom hebben we een ander
      // stukje code nodig die de absolute waarde
      gebruikt.
      standaardnoottijd = (tijdnoot) / abs(splitser);
      standaardnoottijd *= 1.5; // Deze zorgt dat
      korte noten nets iets langer worden.
    }

    // We spelen 90 % van de noot en laten 10 %
    // pauze tussen de noten
    tone(buzzer, melody[dezeNoot],
    standaardnoottijd*0.9);

    // Deze wacht tot de noot uitgespeeld is om
    // verder te gaan met spelen.
    delay(standaardnoottijd);

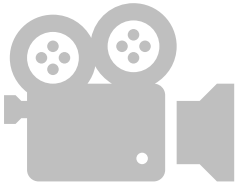
    // Deze stopt het geluidsignaal naar de
    // buzzer als de noot gedaan is met spelen.
    noTone(buzzer);
  }
}

void loop() {
}

```



Demonstratiefilmpje



In dit filmpje kunnen jullie de demonstratie vinden van het project.

Probeer eerst zelf de proef uit te voeren, maar wanneer het niet lukt, dan mag je zeker dit filmpje bekijken.

Het filmpje kan je vinden via deze link: <https://youtu.be/1PjPbua-u1Y>



Bronnen

Voor het uitwerken van dit project haalden wij de mosterd bij volgende bronnen:

- <https://create.arduino.cc/projecthub/GeneralSpud/passive-buzzer-song-take-on-me-by-a-ha-0f04a8>

Wil je nog meer weten over dit onderwerp, bekijk dan zeker deze links:

- <https://musescore.com/> *Via deze link kan je de muzieknoten opzoeken van bekende liedjes.*

Robin Deleu

Hogeschool VIVES – afstudeerrichting: Electronica-ICT