

MAKER

Maak het met de micro:bit

7 makerprojecten met de micro:bit



**Kaarten om dingen te
maken en aan te sluiten op
de micro:bit**

**Op iedere kaart een leuk
project waarmee je iets
maakt.**

Op iedere kaart een leuk project waarmee je iets maakt.

Kaart 1: Micro:beat

Je maakt een uniek muziekinstrument met de micro:bit. Door het te bewegen verandert de toon en met een binaire code verander je de beat.



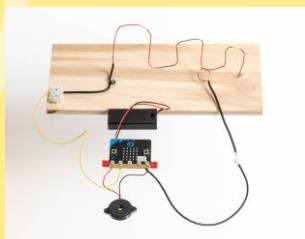
Kaart 2: Dobbelsteen

Je maakt eerst een eenvoudige dobbelsteen die reageert op schudden, daarna een dobbelsteen die reageert op de stand van de micro:bit.



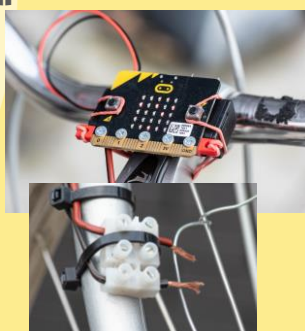
Kaart 3: Bibberspiraal

Het is de bedoeling dat je met een ijzeren ring een gebogen draad volgt. Telkens wanneer je de draad met de ring raakt gaat er een alarm af.



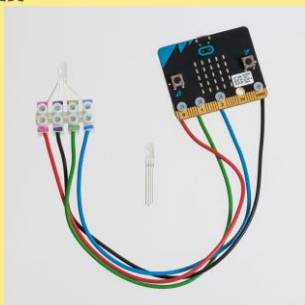
Kaart 4: Afstand meten

Maak je van de micro:bit een afstandsmeter voor je fiets. Met knop A start je de meting en met knop B laat je de gefietste afstand zien.



Kaart 5: Spelen met licht

Bij dit project sluit je een RGB-led aan en laat je verschillende kleuren zien.



Kaart 6: De quiz

Je gaat knoppen maken om een quiz te kunnen spelen. En we gaan hem zo programmeren met de micro:bits dat je weet wie er als eerste heeft gedrukt.



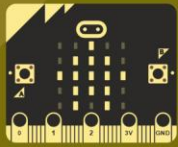
Kaart 7: Pratende plant

Je gaat een sensor maken die je in de grond naast je plant zet. Als de grond te droog is, geeft de sensor een seintje dat hij water nodig heeft.



Op www.micro-bit.nl staan alle programma's

Naast de micro:bit heb je ook andere onderdelen nodig. Deze zijn te koop bij Kiwi-Electronics



MAKER

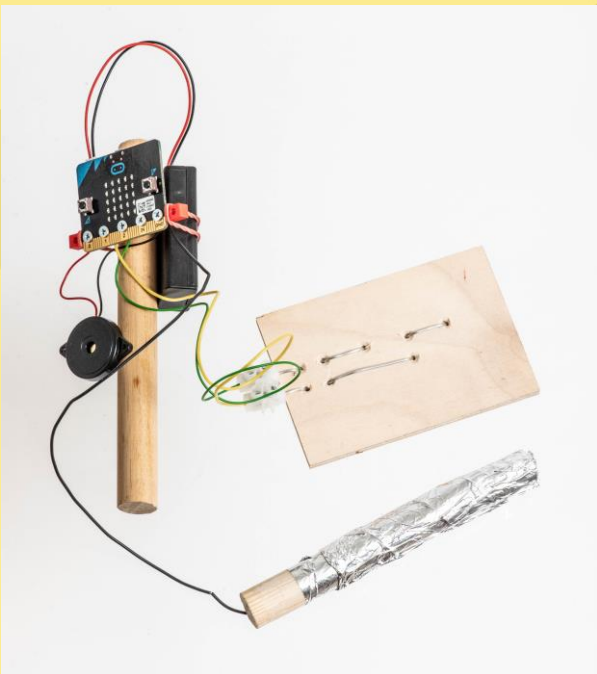
1

Micro:beat

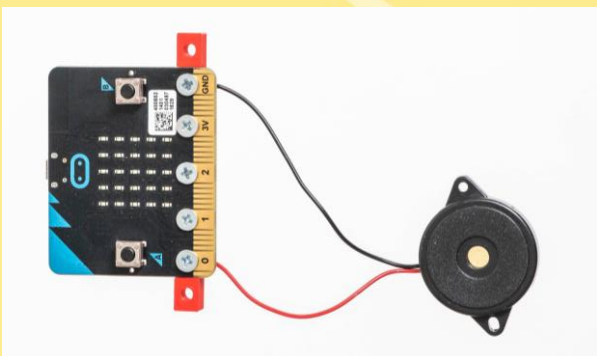
Micro:beat

Je maakt een uniek muziekinstrument met de micro:bit.
Door het te bewegen verandert de toon en met een
binaire code verander je de beat.

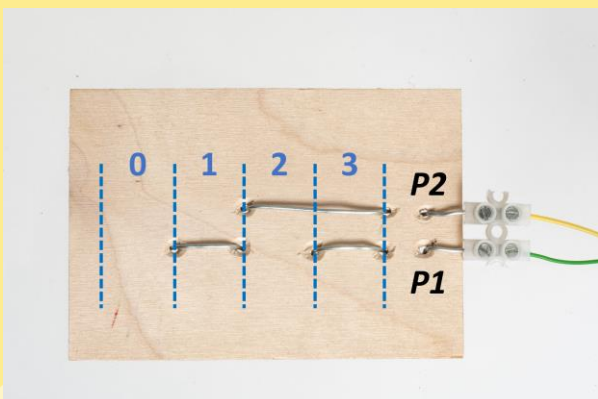
Bouw de micro:beat



Op de foto zie je hoe de micro:beat eruit ziet. De
luidspreker is aangesloten op de pinnen **GND** en **P0**:



De binaire code wordt gemaakt op een plankje waarop
je met een voeler over contacten gaat. De contacten
zijn aangesloten op de pinnen **P1** en **P2**. De voeler is
aangesloten op pin **GND**.

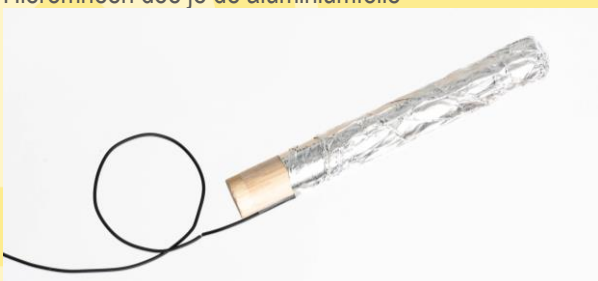


De voeler maak je van een rond houtje, een stuk draad en wat aluminiumfolie.

Verwijder eerst een flink eind van de isolatie van de draad en draai de draad om het stokje.



Hieromheen doe je de aluminiumfolie



Als je nu met de voeler over de contacten op het plankje gaat dan zijn er vier mogelijkheden. De voeler maakt:

- helemaal geen contact (0)
- contact met **P1** (1)
- contact met **P2** (2)
- contact met **P1** en **P2** (3)

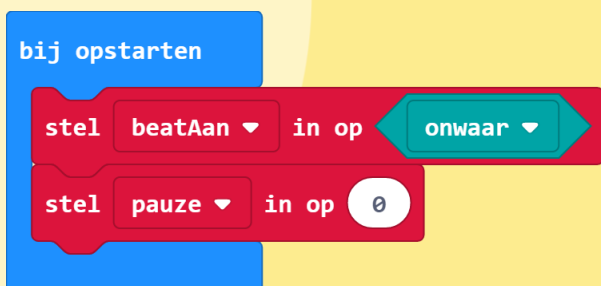


Bouw het programma

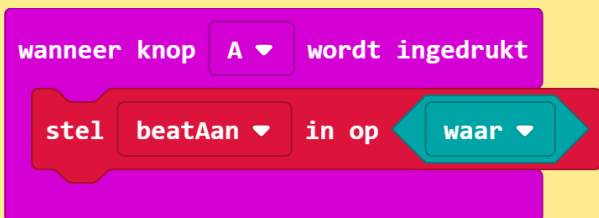
Definieer als eerste vijf variabelen:



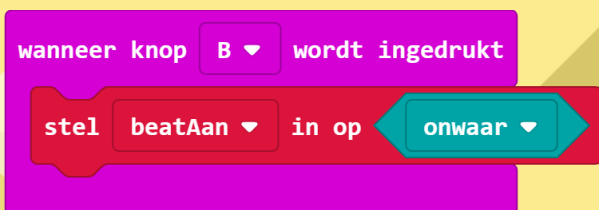
In het blok *bij opstarten* krijgen de variabelen *beatAan* en *pauze* hun startwaarde.



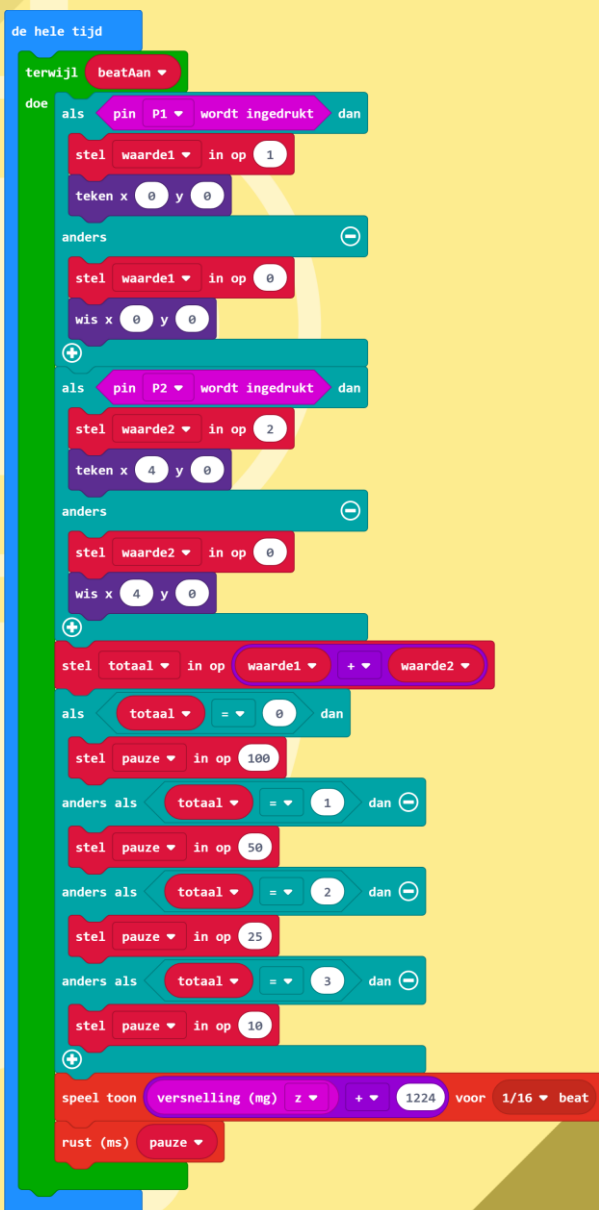
De micro:beat kan je aan- uitzetten met de knoppen A en B. Het programma voor het aanzetten van de met knop A ziet er zo uit:



En met knop B krijgt de variabele *beatAan* de waarde *onwaar*.



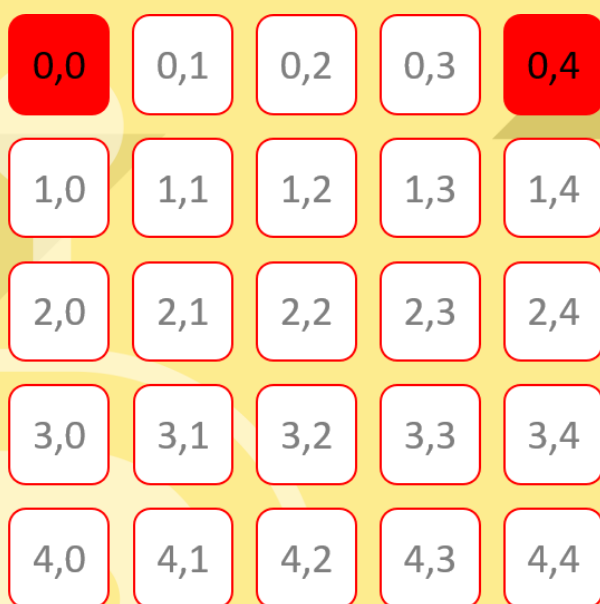
Tot slot bouw je het hoofdprogramma:



In dit programma zie je dat het alleen loopt als de variabele *beatAan* de waarde *waar* heeft. Dit wordt geregeld met de knoppen A en B.

Dan zijn er de variabelen *waarde1* en *waarde2* die worden ingesteld met de pinnen *P1* en *P2*.

Of de pinnen worden aangeraakt kan je zien met de leds op (0,0) en (0,4). Deze worden aan- en uitgezet met de blokjes *teken x y* en *wis x y*.



De inhoud van *waarde1* (0 of 1) en van *waarde2* (0 of 2) worden bij elkaar opgeteld naar de variabele *totaal*.

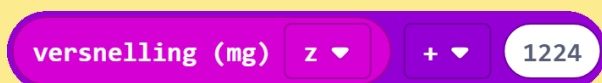
In een tabel ziet het er dan zo uit:

P1	P2	Binair	<i>waarde2</i> + <i>waarde1</i>	<i>totaal</i>	<i>pauze</i>
Nee	Nee	00	0+0	0	100
Nee	Ja	01	0+1	1	50
Ja	Nee	10	2+0	2	25
Ja	Ja	11	2+1	3	10

In een blok *als .. anders als* gebruiken we *totaal* om de lengte van de rust in te stellen.

Hoe meer contacten op het plankje worden aangeraakt, hoe korter de pauze. En hoe korter de pauze, hoe sneller de beat.

De toonhoogte wordt bepaald door het verdraaien van de micro:bit. Dit gaat met het blok *speel toon* met daarin:



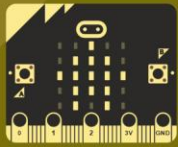
De laagste waarde die het blok *versnelling* kan hebben is -1024. Omdat het blok *speel toon* alleen werkt met positieve getallen, tellen we er 1224 bij op. Hiermee komt de laagste toon op 200 Hz.

TIP 1

De pinnen van de micro:bit zijn zo gevoelig dat ze al reageren op contact via de huid. Je kan de voeler dus in de hand houden en met je vingers contact maken met de contacten P1 en P2 op het plankje.

TIP 2

Je krijgt hele andere geluidseffecten door de contacten op het plankje niet te gebruiken om de beat in te stellen, maar juist om de toonhoogte in te stellen.



MAKER

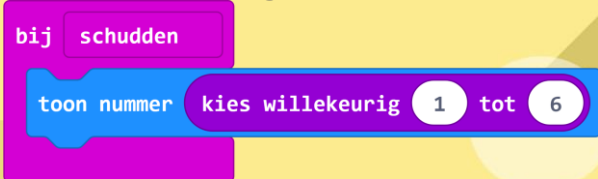
2

Dobbelsteen

Dobbelsteen

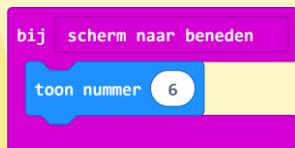
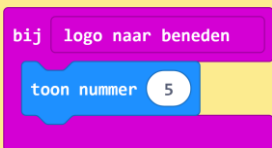
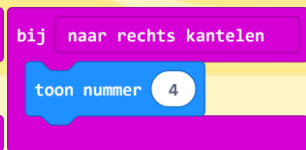
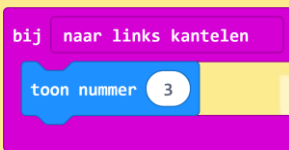
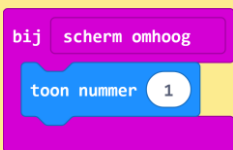
Je maakt eerst een eenvoudige dobbelsteen die reageert op schudden, daarna een dobbelsteen die reageert op de stand van de micro:bit.

Bouw het eerste programma



Bouw het tweede programma

We gaan de code nu aanpassen zodat je door het draaien van de micro:bit het juiste cijfer ziet. Hiervoor moet je telkens het blokje *bij schudden* aanpassen.

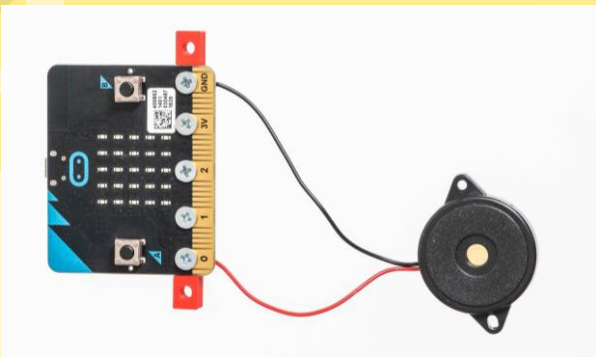


Geluid toevoegen

Hieronder zie je de code voor drie ogen, je kunt nu ook de vijf andere routines aanpassen.



Sluit nu de luidspreker aan. Één draadje gaat naar **P0** en het andere naar de **GND**. Draai je micro:bit, hoor je het juiste aantal toontjes?



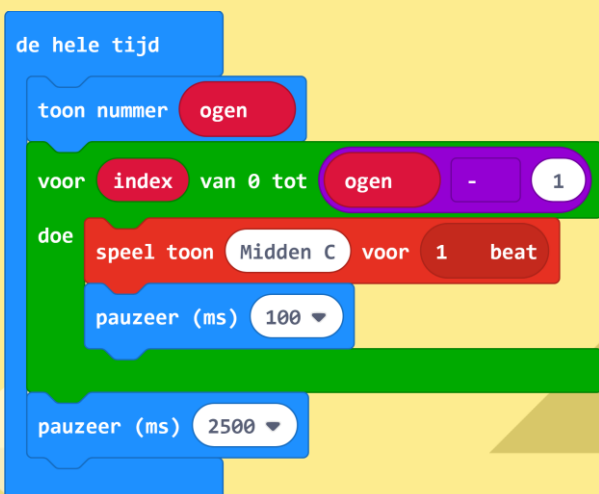
Handiger programmeren

Door gebruik te maken van een variabele (*ogen*) maak je het programma korter en is makkelijker om het aan te passen.

Maak eerst een variabele *ogen* en daarna deze zes routines:



En nu hoef je nog maar één keer deze code op te nemen:



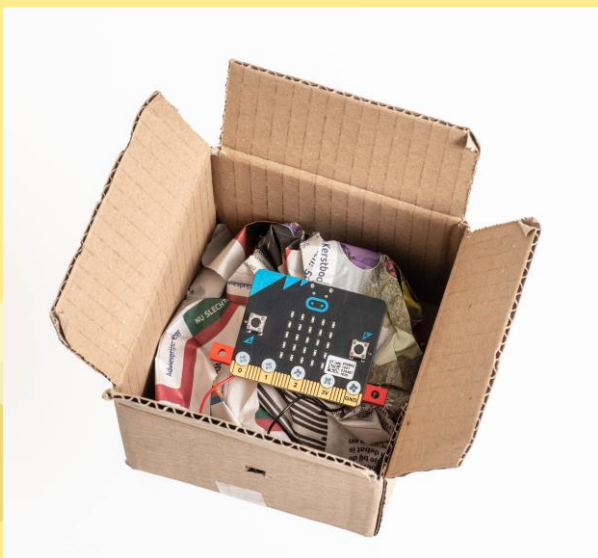
Dobbelsteen bouwen

Om een dobbelsteen te kunnen maken die je echt kunt gooien doe je de micro:bit in een vierkant doosje.

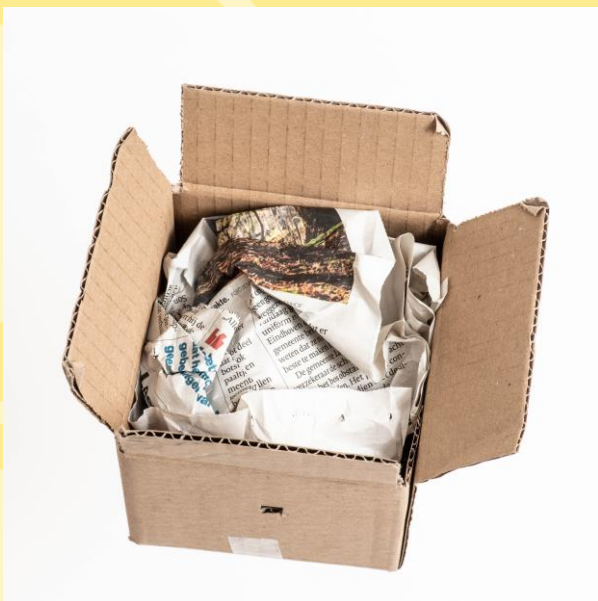
Maak een kleine opening in de doos en zet de luidspreker vast met plakband.



Vul de doos voor de helft met papier en papier en
plaats de micro:bit en de batterij houder in de doos

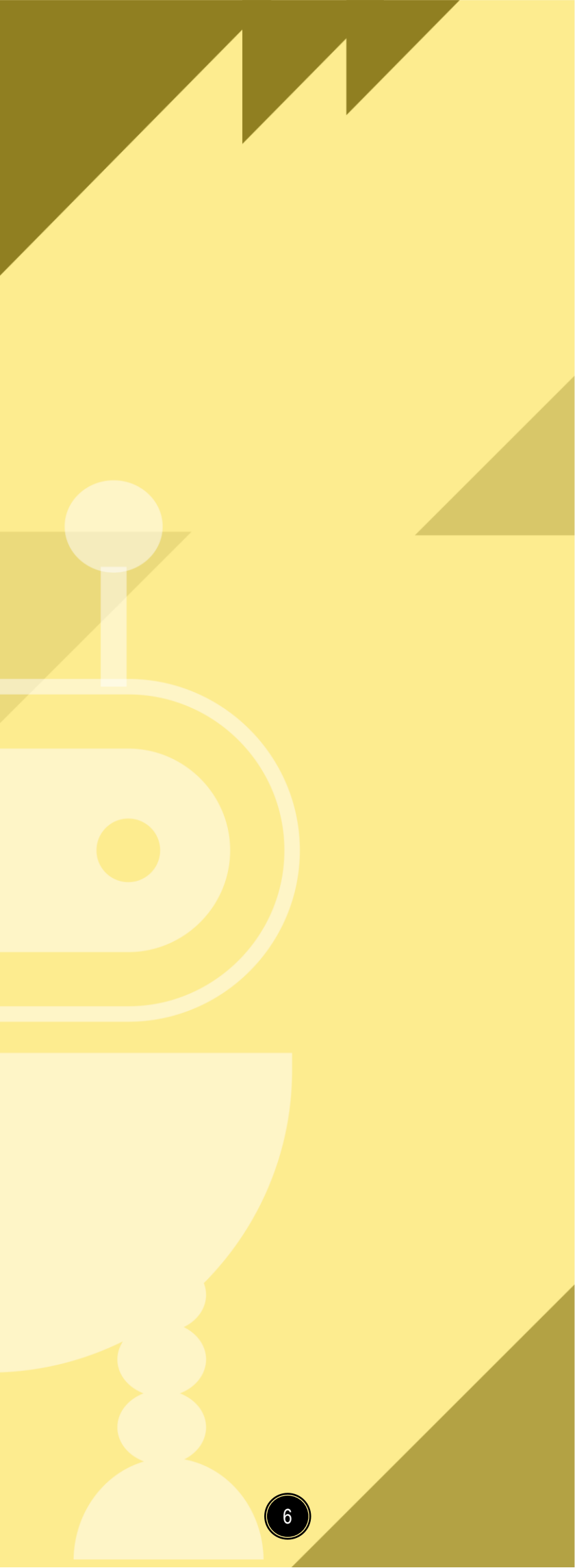


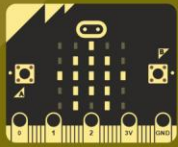
Vul de rest van de doos op met plastic of ander
materiaal



Maak de doos dicht met plakband en maak de dobbelsteen stippen op de doos met papier of stift.







MAKER

3

Bibberspiraal

Bibberspiraal

Het is de bedoeling dat je met een ijzeren ring een gebogen draad volgt. Telkens wanneer je de draad met de ring raakt gaat er een alarm af.

Bouw eerst de ring en de spiraal

Neem voor de ring een stuk ijzer- of koperdraad van 20 cm en buig dat in deze vorm.



Voor de spiraal neem je een plankje van ongeveer 50 x 20 x 2 cm. Hierop komt je de spiraal die je maakt van een stuk installatiedraad van ongeveer 150 cm. Haal de isolatie van de draad, maar laat op het einde ongeveer 4 cm zitten voor het eindcontact. Bevestig de draad met twee schroeven of twee kroonsteentjes op de plank. Zorg ervoor dat de draad door de ring gaat.

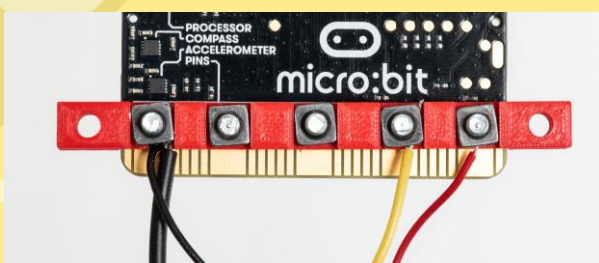
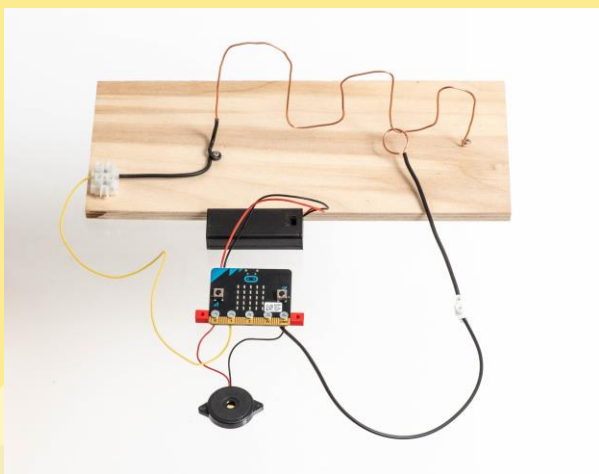
Buig nu de draad in een willekeurige vorm en bevestig deze op een plankje.



Sluit de micro:bit aan

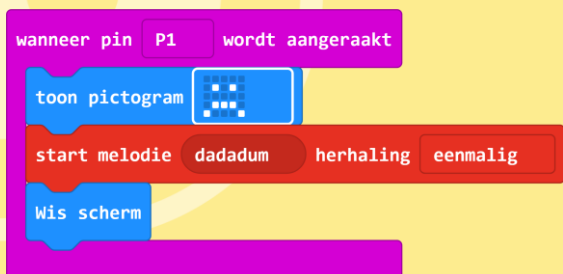
Je kan dit doen met de micro:bit aansluitstrip of met krokodilsnoertjes.

1. De ring sluit je aan op P0
2. De spiraal komt op de GND-aansluiting
3. Van de luidspreker sluit je de zwarte draad (-) ook aan op GND. De rode draad (+) komt op P0.



Het basisprogramma

Als de ring (op P1) via de spiraal contact maakt met GND verschijnt er een verdrietig gezichtje en hoor je een melodietje.



Eenvoudige puntentelling

Hierbij krijg je bij de start van het spel 10 punten en iedere keer als je de spiraal raakt, verlies je een punt.

Maak eerst een variabele met de naam *punten*.



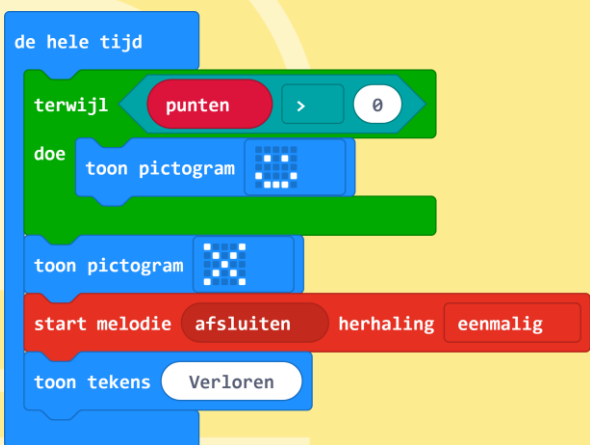


Bij het opstarten krijgt de variabele met de naam *punten* de waarde 10.

Iedere keer als je de spiraal aanraakt wordt er 1 keer een melodietje afgespeeld, wordt de waarde van de variabele *punten* met 1 punt verminderd en krijg je te zien hoeveel punten je nog over hebt.

Uitbreiding 1: Puntentelling

Door deze blokken toe te voegen krijg je een blij gezicht te zien zolang je nog punten hebt. Als alle punten op zijn verschijnt er een kruis, hoor je een melodietje en komt er 'Verloren' op het display.



Het blok *de hele tijd* is toegevoegd. Binnen dit blok wordt gecontroleerd of er nog genoeg punten over zijn om door te spelen.

Uitbreiding 2: Maximale speeltijd

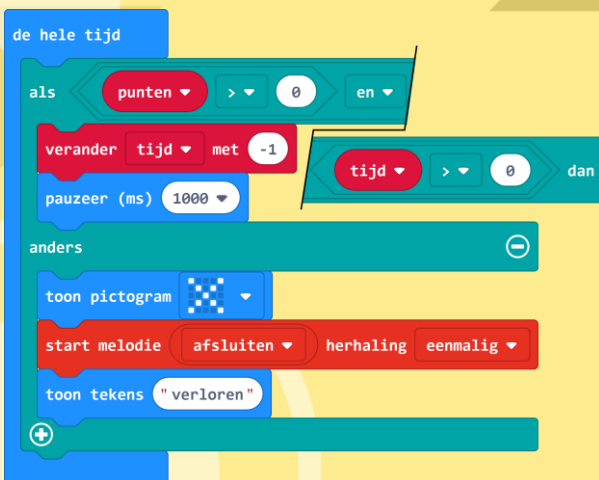
We gaan nu een maximale speeltijd (tijd) toevoegen.

Om de maximale speeltijd in te kunnen stellen, maak je eerst een nieuwe variabele *tijd*.

In het blok *bij opstarten* stel je *tijd* in op 20, dit betekent dat de speeltijd 20 seconde is.



Zolang de speeltijd niet is verstreken wordt de variabele *tijd* iedere seconde met 1 verminderd. Als de speeltijd voorbij is zien we een X en wordt de afsluitmelodie afgespeeld.

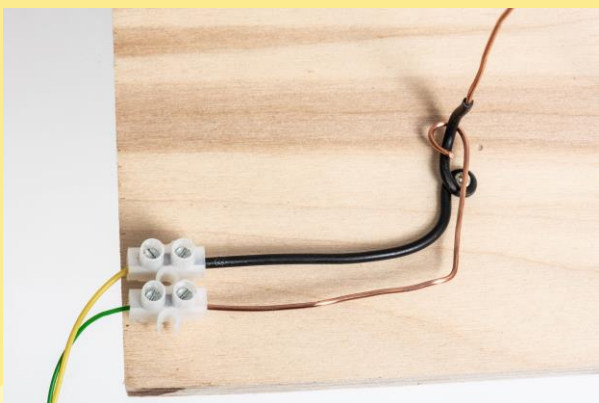


Als de speler contact maakt dan ziet deze een ☹ en hoort deze het wawawawaa-melodietje.



Finish gehaald !

Om te kunnen bepalen of je het eind van de spiraal hebt gehaald maken we aan het einde een eindcontact.



Deze sluit je aan op P2.

De speler start het spel door op de resetknop van de micro:bit te drukken. De variabele *tijdmeting* wordt dan waar. Dit geeft aan dat de tijdmeting is gestart.

```

bij opstarten
  stel punten ▼ in op 3
  stel tijd ▼ in op 20
  stel tijdmeting ▼ in op waar ▼
  toon pictogram [5x5 grid] ▼
  toon nummer punten ▼

```

De spiraal is aangesloten op pin P1:

```

wanneer pin P1 ▼ wordt aangeraakt
  toon pictogram [5x5 grid] ▼
  start melodie wawawawaa ▼ herhaling eenmalig ▼
  verander punten ▼ met -1
  toon nummer punten ▼

```

Het eindcontact is aangesloten op pin P2. Als er contact wordt gemaakt met het eindcontact dan wordt de tijdmeting gestopt door de variabele *tijdmeting* de waarde *onwaar* te geven.

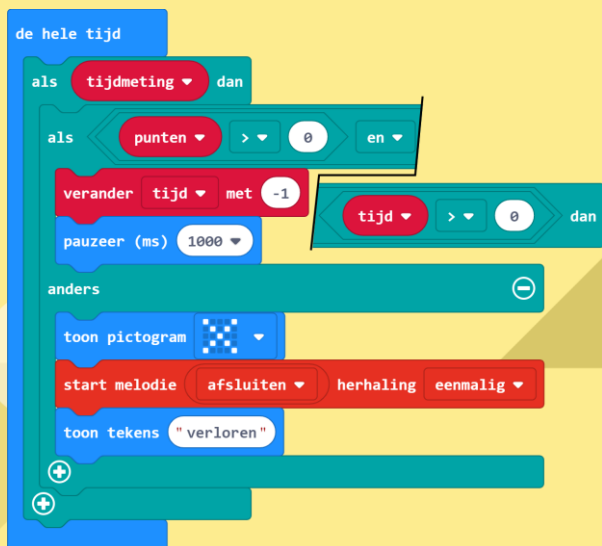
De speler hoort de opstart-melodie.

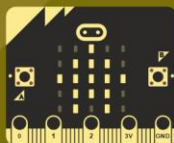
```

wanneer pin P2 ▼ wordt aangeraakt
  stel tijdmeting ▼ in op onwaar ▼
  toon pictogram [5x5 grid] ▼
  stel punten ▼ in op 3
  stel tijd ▼ in op 20
  start melodie opstarten ▼ herhaling eenmalig ▼

```

Zolang de variabele *tijdmeting* waar is, wordt de variabele *tijd* iedere seconde met 1 verlaagd.





MAKER

4

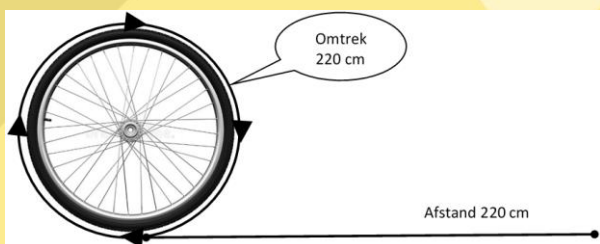
Afstand meten

Afstand meten

Maak je van de micro:bit een afstandsmeter voor je fiets. Met knop A start je de meting en met knop B laat je de gefietste afstand zien.

Hoe werkt het?

Met een elektrisch contact op je fiets meet de micro:bit hoe vaak het wiel is rondgegaan. Als je dit vermenigvuldigt met de omtrek van het wiel weet je de afstand die je hebt gefietst.



Je zet de teller op nul door tegelijk op de knoppen A en B te drukken. Vanaf dit moment wordt ook geteld hoe vaak het wiel rond gaat. Je ziet dit doordat het display telkens verandert:

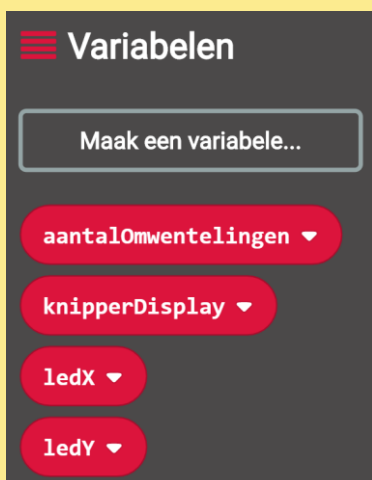


Als je op knop A drukt dan zie je hoeveel meter je hebt gefietst.

Bouw dit programma

Definieer de vier variabelen die je voor dit programma nodig hebt.

Bouw nu de onderdelen van het programma.



bij opstarten

stel **aantalOmwentelingen** in op 0
stel **knipperDisplay** in op **waar**

toon lichtjes



Dit deel van het programma zorgt er voor dat de variabelen

aantalOmwentelingen en *knipperDisplay* de juiste waarde krijgen. In *aantalOmwentelingen* wordt bijgehouden hoe vaak er contact is gemaakt en met *knipperDisplay* geven we aan of het display moet knipperen of dat de afstand moet worden getoond.

Hierna komt het deel waarin wordt geteld hoe vaak er contact wordt gemaakt tussen de pinnen 3V en P0.

de hele tijd

terwijl **waar**

doe
als **lees digitaal pin P0** = **1** dan
verander **aantalOmwentelingen** met **1**
als **knipperDisplay** = **waar** dan
voor **ledX** van 0 tot 4
doe
voor **ledY** van 0 tot 4
doe
wissel x **ledX** y **ledY**
pauzeer (ms) **200**

de hele tijd

terwijl **waar**
doe

Het lijkt misschien vreemd dat er in het blokje *de hele tijd* ook nog eens een blokje *terwijl...* staat. Dit is gedaan omdat het blokje *terwijl...* veel sneller werkt als het blokje *de hele tijd*.

Iedere keer als er contact wordt gemaakt verandert ook de afbeelding op het display. Dat doen we met het blokje *wissel x ledX y ledY* omdat dit sneller werkt als het blokje *toon lichtjes*.

Tot slot is er nog het blokje *pauzeer (ms) 200*. Dit is nodig omdat een elektrisch contact altijd een beetje stuitert. We noemen dit *denderen*. Om nu te voorkomen dat iedere dendering als één omwenteling wordt geteld

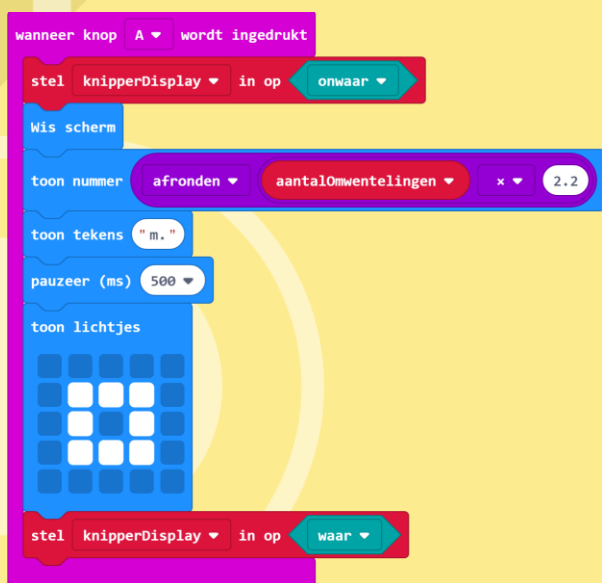
wachten we 200 milliseconden. De dender is dan zeker voorbij.



Ik snap het

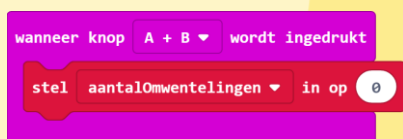
Het dendereffect van het contact lossen we op door telkens 200 ms te wachten. Als je dit niet zou doen, geeft de afstandsmeter dan meer of minder meters aan als je echt hebt gefietst?

Natuurlijk wil je ook zien hoeveel je hebt gefietst. Dat doen we met deze blokjes. De variabele *knipperDisplay* krijgt de waarde *onwaar* waardoor het knipperen van het display stopt en het scherm wordt gewist. Het programma hieronder is gemaakt voor een fietswiel met een omtrek van 2,2 meter. Je moet deze waarde aanpassen voor de omtrek van jouw fietswiel.



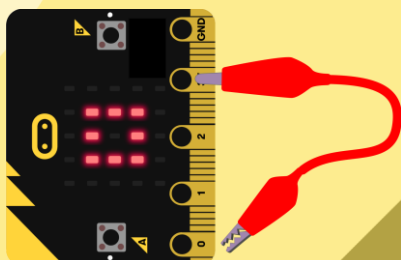
De afgeronde afstand wordt een halve seconde getoond waarna het display weer gaat knipperen.

Als laatste bouwen het deel van het programma waarmee we de teller op nul kunnen zetten:



Testen

Je kan dit programma testen door met een draadje telkens verbinding te maken tussen de 3V- en de P0-aansluiting van de micro:bit.



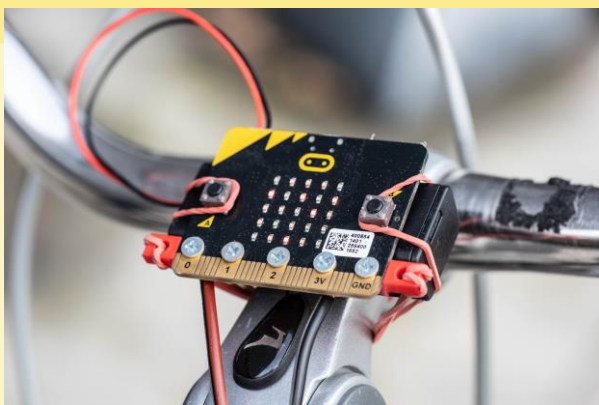
Maak het contact op de fiets

Op de foto zie je een voorbeeld van het contact. Dit bestaat uit een stuk ijzerdraad dat je om de spaken buigt en een kroonsteentje, draad en twee bindbandjes.

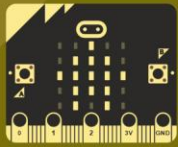


De draden sluit je aan op de 3V en de P0-aansluiting van de micro:bit.

Bevestig nu de micro:bit op het stuur, sluit hem aan op de batterij en ... rijden maar.



www.micro-bit.nl



MAKER

5

Spelen met licht

Spelen met licht

Bij dit project sluit je een RGB-led aan en laat je verschillende kleuren zien.

RGB-led

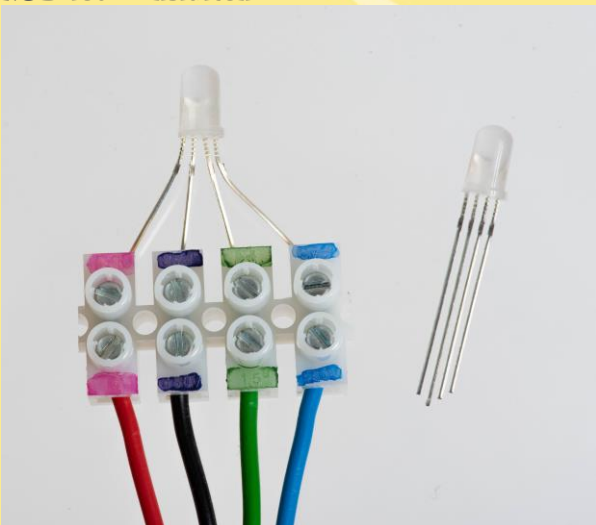
Een RGB-led is een speciale led met 4 aansluitdraden.

In een RGB-led zitten 3 gewone leds, een rode (R), een groene (G) en een blauwe (B). Die 3 leds kun je afzonderlijk laten branden, zodat de RGB-led rood, groen of blauw licht geeft. Je kunt de leds ook combineren.

Als je de rode en de groene led laat branden geeft de RGB-led geel licht. De rode en de blauwe led geven paars licht. En ... als je alle 3 de leds laat branden krijg je wit licht.



RGB-led aansluiten



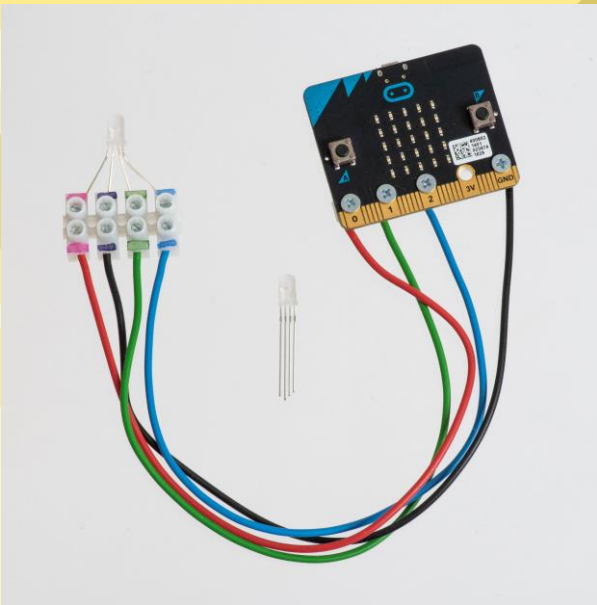
Draai de schroefjes van de 4 kroonsteentjes aan beide zijden een beetje los.

Steek de draden van de RGB-led in de kroonsteentjes en draai de schroefjes vast.

Onthoud in welk kroonsteentje de lange aansluitdraad is bevestigd. Deze lange draad moet je later namelijk verbinden met de GND van de micro:bit (de zwarte draad op de foto).

Neem de 4 draden en steek die met een van de gestripte uiteinden in de kroonsteentjes en draai de schroefjes vast.

Bevestig de andere uiteinden van de draden met de aansluitstrip op P0, P1, P2 en de GND van de micro:bit.



Het programma (digitaal)

Met dit programma krijgt de led iedere seconde een andere kleur als je op knop A drukt.



Kleuren mengen

Door kleuren tegelijk te laten branden kan je verschillende kleuren maken. Hieronder zie je de code om waarmee je geel licht maakt door de rode en de groene led tegelijk aan te zetten.



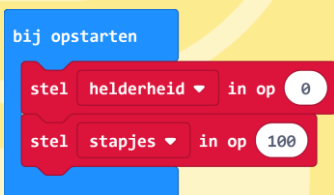
Wil je paars, lichtblauw of wit licht programmeren gebruik dan de kleurencirkel om te kijken welke kleuren je moet combineren.



Helderheid regelen (analoog programma)

Het blokje *Schrijf digitaal pin* kent maar twee standen: een 0 en een 1. De 1 gebruikte je als je het lichtje wilde aanzetten, de 0 als je het wilde uitschakelen. Het blokje *Schrijf analoog pin* kent maar liefst 1023 standen. Daarmee kun je de lichtsterkte regelen: hoe hoger het getal hoe helderder het licht.

Met de code hieronder kun je led die is aangesloten op P0 helderder en minder helder laten branden.



Bij het opstarten van de micro:bit wordt *helderheid* ingesteld op 0. Het programma begint dus met een ledje dat nog geen licht geeft.

Met *stappjes* ingesteld dat de helderheid van het lichtje straks in stapjes van 100 wordt opgevoerd.

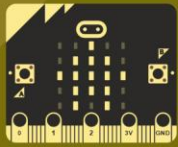


Met *schrijf analoog pin P0 naar helderheid* wordt de spanning ingesteld. De minimale waarde is 0 (led uit) en de maximale waarde is 1023 (led helemaal aan).

Steeds na 500 ms. (= ½ sec.) wordt de helderheid in stapjes van 100 opgevoerd. Als de helderheid groter wordt dan 1000 dan wordt de helderheid bij iedere stap met 100 verminderd.

Is de waarde kleiner dan 0 dan wordt de waarde weer verhoogd met stapjes van 100.

Dit proces herhaalt zich steeds.



MAKER

6

De quiz

De quiz

Je gaat knoppen maken om een quiz te kunnen spelen.
En we gaan hem zo programmeren met de micro:bits
dat je weet wie er als eerste heeft gedrukt.

Voor dit project heb je meerdere micro:bits nodig.

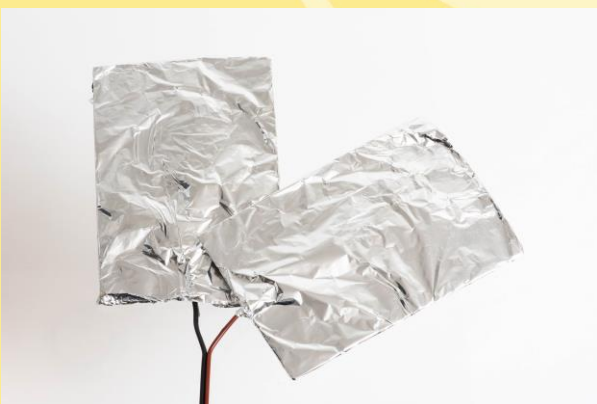
De knoppen

Voor iedere speler heb je één micro:bit en een knop nodig. Deze knop is gemaakt van twee stukken karton, aluminiumfolie en papier.

Verwijder een flink deel van de isolatie van de draden en plak deze op het karton.



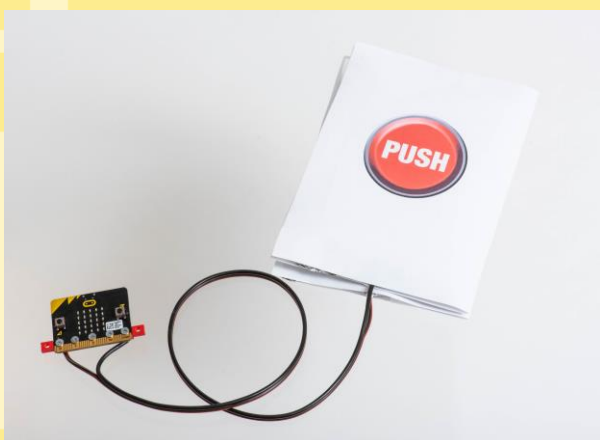
Omwikkel de twee stukken karton met aluminiumfolie.



Neem twee vellen papier en knip in ieder vel een gat en vouw dit om het karton. Zorg dat de folie zichtbaar wordt.



Leg de twee delen van de schakelaar op elkaar zodat de twee stukken aluminiumfolie elkaar kunnen raken als je er op drukt, maar elkaar niet raken als je er niet op drukt.



Bevestig de draden aan de GND en aan de P0 van de micro:bit.

Controleer of de knop werkt

Met dit programma kan je controleren of de knop het doet.



Programma bouwen

Maak eerst twee variabelen: *geblokkeerd* en *mijnPlaats*.



Bij het opstarten wordt de radio ingesteld op groep 1.
Alle micro:bits zitten in dezelfde radio groep.

Bij het opstarten ben je nog niet geblokkeerd.



Als je de knop indrukt EN als je nog niet geblokkeerd bent dan:

- wordt *mijnPlaats* verhoogd met 1 punt;
- zendt de radio de inhoud van de variabele *mijnPlaats* naar alle andere micro:bits.
- wordt jouw plaats op het scherm getoond;
- verandert de variabele *geblokkeerd* naar *waar*.
Je kunt zo maar één keer drukken.



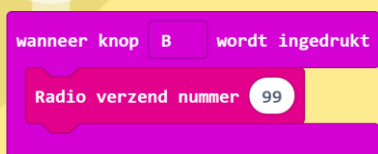
Tip

Je vindt het blokje *reset* in de groep *Besturen* (onder *Geavanceerd*)

Wanneer iemand anders op zijn/haar knop drukt en jouw micro:bit het plaatsnummer van die ander ontvangt EN je jezelf nog niet hebt gedrukt dan:

- wordt de variabele *mijnPlaats* ingesteld op het ontvangen nummer (*receivedNumber*). Op deze manier weten alle micro:bits wat de stand is.

Alleen de spelleider kan met knop B het resetsignaal versturen. Dit is het getal 99. Als dit getal wordt ontvangen wordt de micro:bit gereset.

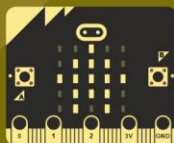


Tip

Gebruik knop A als je geen zin, tijd of spullen hebt om zelf een knop te maken.

Als je de quiz wilt spelen zonder grote zelfgemaakte knop dan kan je het blokje





MAKER

7

Pratende plant

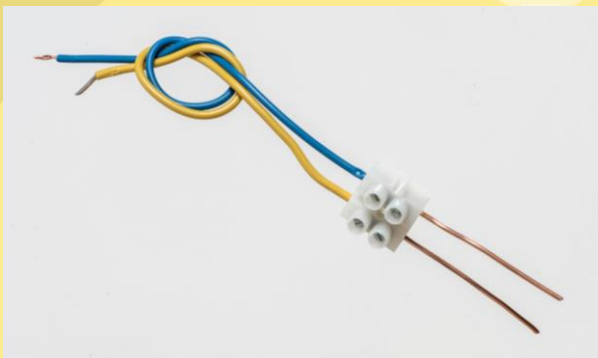
Pratende plant

Je gaat een sensor maken die je in de grond naast je plant zet. Als de grond te droog is, geeft de sensor een seintje dat hij water nodig heeft.

Je zou er zelfs ook nog geluid aan kunnen koppelen, dan heb je een zingende plant.

De sensor

Van installatiedraad, een kroonsteentje en soepel montagedraad maak je deze vochtsensor...

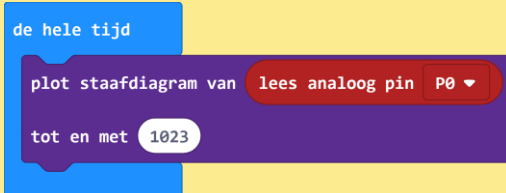


... of gebruik je iets anders:



Steek de sensor in de grond van de plant en verbind deze met de pinnen 3V en P0 van de micro:bit.

Het programma



Tip

Het blokje *Lees analoog van P0* vind je bij *Pinnen* onder *Geavanceerd*.

Je kan de sensor nu testen door de plant water te geven.

Energiezuinig programma

Er loopt nu de hele tijd stroom via de micro:bit naar de pinnen en de aarde. Dit is niet erg energiezuinig.

We gaan nu de code aanpassen zodat er iedere 5 seconden een meting gedaan wordt.

Hiervoor moeten we eerst de kabels op een iets andere manier verbinden met de micro:bit en passen we de code aan.

Haal de draad van de 3V vandaan en sluit deze aan op P1.



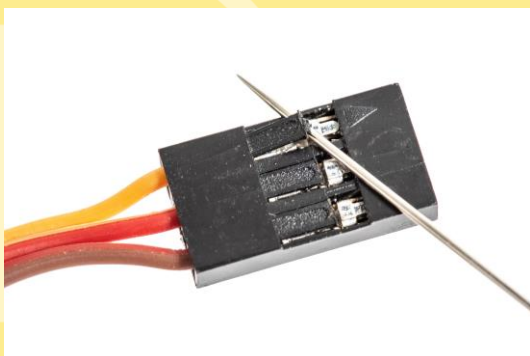
Maak een variabele *vochtgehalte*.



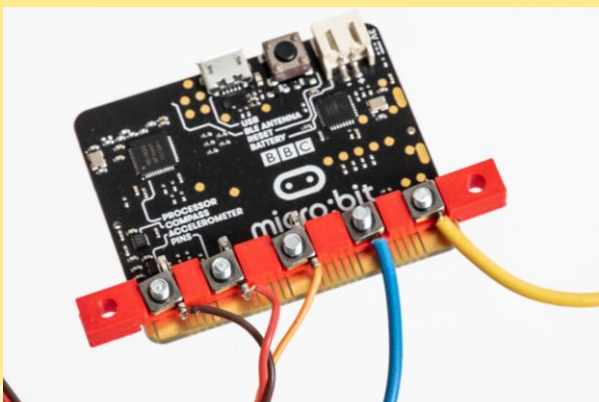
Toon bordje "WATER"

Als de plant het te droog krijgt kan hij dit aangeven door een bordje "WATER" te laten zien. Dit bordje zit op een servo die gaat draaien als de vochtigheid onder de 1.000 zakt.

Om de servo aan te sluiten verwijder je de zwarte connector van de servo. Je hoeft deze niet af te knippen, maar je kunt de busjes met een naald uit het zwarte deel halen.



Sluit de servo nu zo aan op de micro:bit



De draden op P0 en P1 komen van de sensor.

De draden van de servo sluit je zo aan:

- Oranje op P2
- Rood op 3V
- Bruin op GND

Maak nu een vlag aan de servo.



Bevestig de servo aan een stokje of een oude ballpen en steek deze in de grond.

Het hele systeem ziet er nu zo uit:



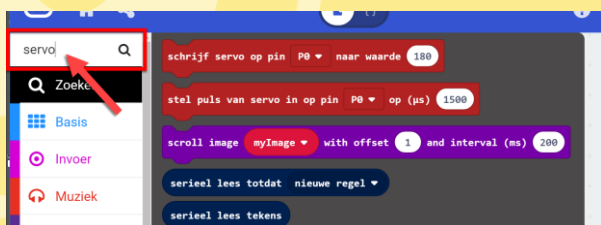
Het programma

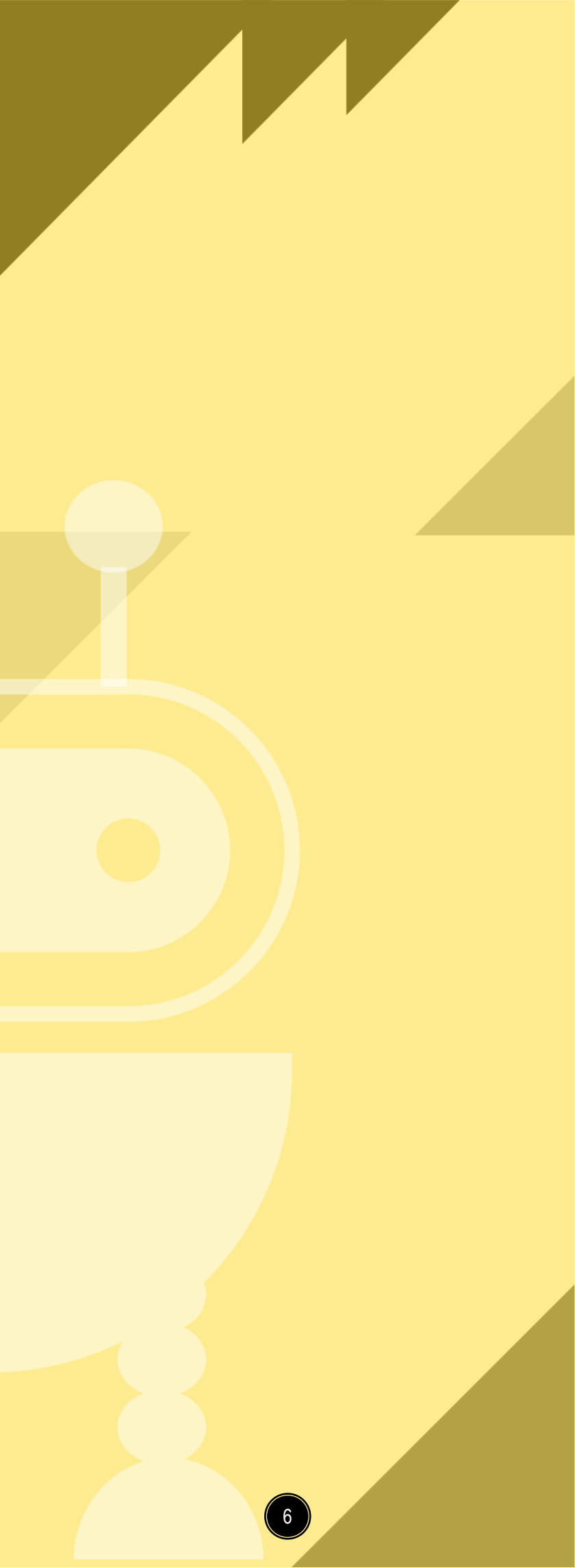
Dit programma test iedere 5 seconde of het vochtigheidsgehalte van de grond kleiner is dan 1000. De servo draait dan naar 180°, anders naar 0°.

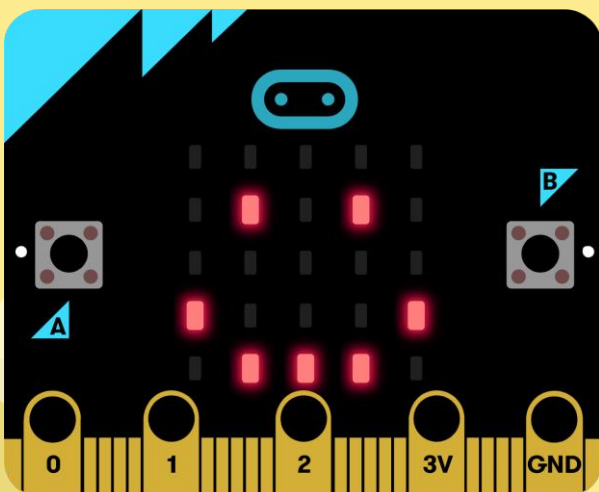


TIP

Gebruik het zoek zoekveld (links in de hoek bij het vergrootglas) om een blokje te vinden.







Deze micro:bit EXPLORER
kaartenserie is tot stand gekomen in
een samenwerking tussen Stichting
DevLab Academy en CodeKids.nl

DEVLAB | ACADEMY

CODEKIDS

Ga voor meer informatie over micro:bit naar de
website van *micro:bit community NL*

www.micro-bit.nl



© 2020