

# Lego Club

*De Pracht*

*Aalst-Waalre*

*Maarten Pennings*

*2014 feb 8 – 2021 Aug 28*

*Versie 16*



Playlist: in YouTube zoek naar de afspeellijst “Maarten Pennings - Lego Club”  
[http://www.youtube.com/playlist?list=PLrlJSwck1Q0iv\\_t6WtuNv7dbaEXJX42nd](http://www.youtube.com/playlist?list=PLrlJSwck1Q0iv_t6WtuNv7dbaEXJX42nd)

of naar het nieuwe kanaal “De Pracht Lego Club”

[https://www.youtube.com/channel/UCxKt3LKH9oVT\\_rLr5mPyQkg](https://www.youtube.com/channel/UCxKt3LKH9oVT_rLr5mPyQkg)

---

# Projecten

---

- 00 Begin
- 01 Molen
- 02 Clown
- 03 Deur
- 04 Flappy
- 05 Draaimolen
- 06 Krokoklap
- 07 Kaars
- 08 Kleurkenner
- 09 Vuilniswagen
- 10 Garage
- 11 Afstand bestuurbare auto
- 12 Go home
- 13 Landmeter
- 14 Adaptive Cruise Control
- 15 Voorleesbord
- 16 Dobbelsteen
- 17 Rondeteller
- 18 Parking
- 19 Blokrijder
- 20 Doosvolger
- 21 Koekjeskluis
- 22 Pancode
- 23 Memory
- 24 Radarwagen
- 25 Lift
- 26 Keukenwekker
- 27 Ballensorteerder
- 28 Colorcoaster
- 29 Line follower
- 30 Gripper
- 31 Don't Push
- 32 Rock, paper, scissors
- 33 Reactie tester
- 34 Morse
- 35 Radar

## Lego bouwwerk

Bij deze opdracht bouwen we eigenlijk niks met lego.

We leren hoe we de NXT – de computersteen – kunnen programmeren.

We doen maar één ding: een motor aansluiten aan de NXT – aan port B!



## Op de PC

Start op de PC “Lego Mindstorms NXT” op.



We gaan een nieuw programma maken voor de NXT.

Klik daarom op “Go” bij “Start een Nieuw Programma”.



Goed idee: eerst de naam van je project veranderen bijvoorbeeld van “Naamloos-1” in “Maarten-00-Begin” (als je *Maarten* heet, ten minste).

## Het NXT-programma

We maken een programma van één stap (één blok).

Je programma is de Technic staaf in het midden. Blokken staan links.

Sleep een (motor) blok van linksboven naar de Technic balk.

Dit blok zal de motor B (en C) even laten draaien.

Dat is voor ons eerste programma genoeg.



## Download

Verbind de NXT nu met een USB-kabel aan de PC, en zet de NXT aan.



Dan kunnen we het programma downloaden van de PC naar de NXT, en starten.

Als je dat wil, druk dan op de knop “Download en start” (middelste knop).

## Fout verbeteren

Eigenlijk is het programma niet goed. Het blok zet motor B én C allebei aan, en zorgt dan dat ze even ver draaien. Maar motor C is niet aangesloten. En die kan dus niet “even ver” draaien. En dan gaat het mis.

Klik even op het programma blok, zodat er een groene rand omheen komt.



**Pro Tip “instellingen”:** probeer ook eens ‘Richting’ te veranderen en kijk wat er gebeurt. Of ‘Vermogen’ of ‘Tijdsduur’. Bij ‘Tijdsduur’ kun je ook nog ‘graden’ kiezen – wat is dat?

Zet daarna links onder motor C uit, zodat alleen B draait.

En druk dan nog eens op “Download en start”.

## 01 Molen

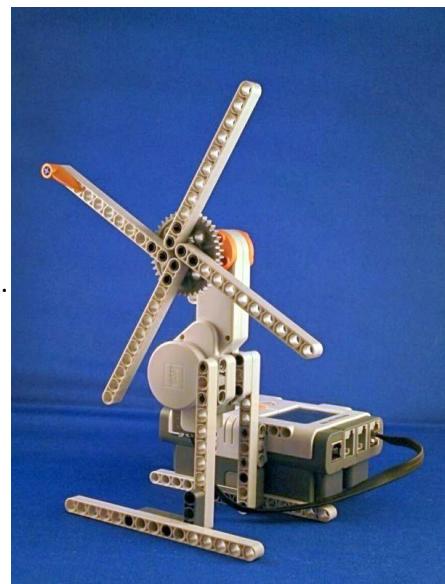
## Eén motor

### Bouw

Maak een windmolen.

Tips:

- Gebruik 1 motor.
- Maak de wieken vast aan een groot tandwiel.
- Geef één wiek een speciale steen (kun je goed 1 rondje aftellen).
- Maak de molen vast aan de NXT (voor stevigheid).
- Klik een snoer in de motor en in poort A van de NXT (poorten met een letter – A, B, of C – zijn voor motoren)



### Doel

Laat de wieken precies 1 rondje draaien – heel langzaam.

### Programma

Schrijf een programma voor de molen.

Dit programma bestaat uit 1 stap, een “motor” stap.

Deze stap laat de wieken 1 rondje maken.



Tips voor de “motor” stap:

- Laat maar één motor draaien (Poort A).
- Laat hem langzaam draaien (Vermogen 10).
- Kies voor Tijdsduur 1 Omwenteling.



### Extra

- Kun je de wieken de andere kant op laten draaien?
- Kun je de wieken ook maar een kwart rondje laten draaien?
- Kun je de wieken eerst 1 rondje linksom en daarna 1 rondje rechtsom laten draaien?

## 02 Clown

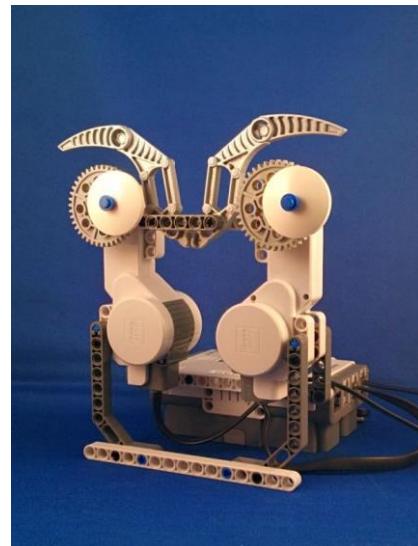
## Twee motoren

### Bouw

Maak een hoofd van een clown – beide ogen draaien.

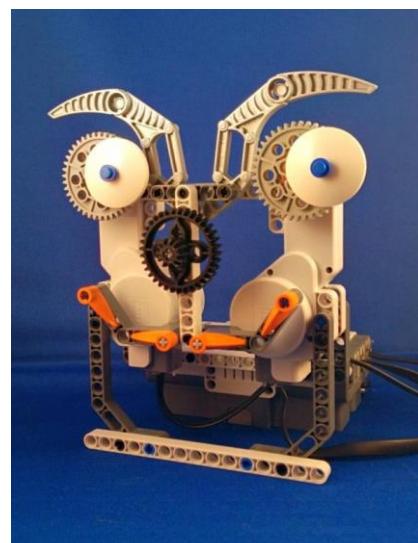
Tips:

- Gebruik twee motoren, één voor elk oog.
- Neem een groot tandwiel voor elk oog en zet het witte rondje uit het midden.
- Maak de ogen vast aan de NXT.
- Klik snoeren in de motoren en in poort A en B van de NXT
- Als je zin hebt kun je een mond en neus toevoegen.



### Doele

Laat de ogen 10 rondjes draaien.



### Programma

Schrijf een programma dat beide ogen 10 keer rond laat draaien.

Dit programma bestaat uit 1 stap, een “motor” stap.

Tips voor de “motor” stap:

- Laat twee motoren draaien (Poort A en B).
- Laat hem redelijk vlot draaien (Vermogen 33).
- Kies voor Tijdsduur 10 Omwentelingen.



### Extra

- Kun je de ogen ook een half rondje laten draaien?
- Kun je de ogen een half rondje rechtsom, half rondje linksom laten draaien (heel langzaam)?
- Moeilijk: kun je de ogen ook dezelfde kant op laten draaien?

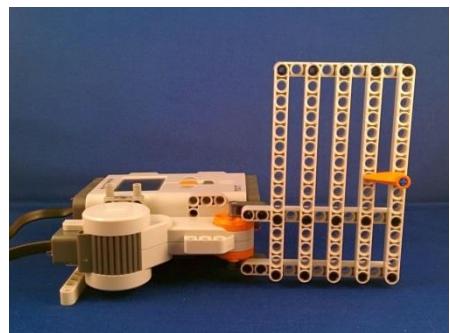
## 03 Deur

## Meer stappen en wacht-blok

### Bouw

Maak een deur die met een motor open en dicht kan.

- Gebruik één motor.
- Zorg dat de deur voldoende kan draaien.
- Maak de deurmotor vast aan de NXT (poort A).



### Doe!

De deur gaat open, blijft 5 seconde open, en gaat dan weer dicht.

### Programma

Dit programma bestaat uit 3 stappen: een motor stap (deur open), een wacht stap (5 seconde), en weer een motor stap (deur dicht).



Tips:

- Eerste blok: laat motor A de deur rustig (Vermogen 15) *open* draaien (Tijdsduur 80 graden).
- Tweede blok: Wacht (Controleer Tijd) tot 5 seconden voorbij zijn.
- Derde blok: laat motor A de deur rustig (Vermogen 15) *dicht* draaien (Tijdsduur 80 graden).



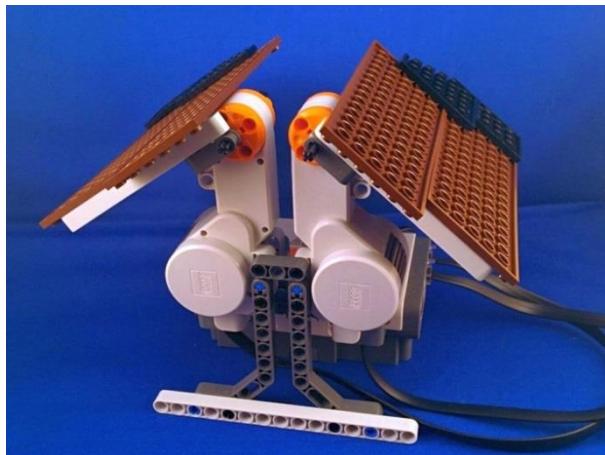
## 04 Flappy

Herhaal lus (10x)

### Bouw

Maak een vogel die met zijn vleugels klappert.

- Gebruik twee motoren, één voor elke vleugel, en verbind ze met poort A en B.
- Maak de vogel vast aan de NXT.
- Als je zin hebt kun je ogen en snavel toevoegen



### Doe!

De vogel klappert (op en neer) met zijn vleugels.

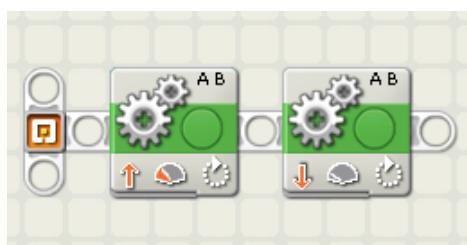
### Programma

Schrijf een programma dat de vleugels omhoog en daarna weer omlaag doet.

Dit programma bestaat uit 2 "motor" stappen, één voor omhoog, en één voor omlaag.

Tips:

- Laat twee motoren draaien (Poort A en B).
- Laat de eerste stap hem *omhoog* draaien (Vermogen 25)
- Laat de tweede stap hem *omlaag* draaien (Vermogen 15).
- Kies voor Tijdsduur bijvoorbeeld 100 Graden.



### Verbetering

Eén keer klapperen is weinig. Voeg een herhaal lus toe om deze twee stappen heen.

Controleer met Tellen zodat het flapperen 10 keer gebeurt.



**Pro Tip** Zet een extra motor blok, die de vleugels een klein beetje optilt, vóór de herhaal. Dat zorgt ervoor dat de vleugels niet tegen de grond klappen.

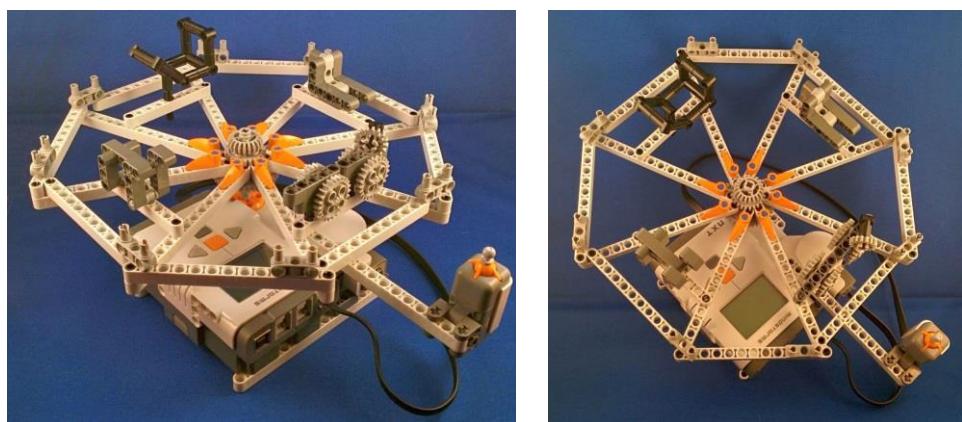
## 05 Draaimolen

## Druk-knop en herhaal eeuwig

### Bouw

Maak een draaimolen met een startknop.

- Gebruik één motor (poort A), en één drukknop (poort 1), maak beide vast aan de NXT.
- Op de foto zie je Technic staven aan een groot tandwiel, een plaat van gewone lego is makkelijker.
- Op de foto zie je de paarden en auto's ook van Technic, gewone lego is makkelijker.

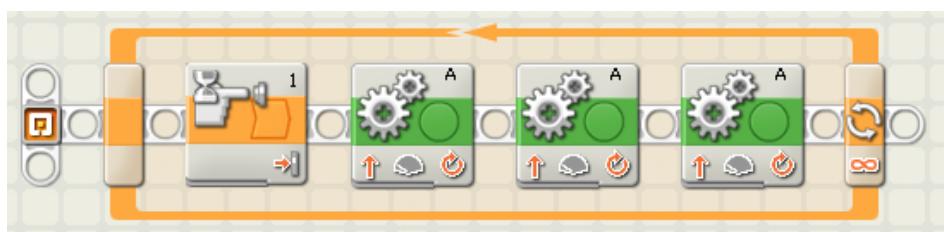


### Doeleind

Als je op de knop drukt begint de attractie: 1 langzaam rondje, 1 snel rondje en ten slotte 1 langzaam rondje. Daarna kan iedereen uitstappen. Druk je weer op de knop dan begint de attractie overnieuw.

### Programma

Ook hier passen we weer een herhaal lus toe.



- We hebben drie motor blokken, de eerste laat motor A rustig (Vermogen 10) draaien (Tijdsduur 1 Omwenteling), de tweede sneller (Vermogen 20, Tijdsduur 1 Omwenteling), en de derde weer rustig (Vermogen 10, Tijdsduur 1 Omwenteling).
- Om het geheel staat een herhaal lus (Controleer Eeuwig).
- Voor de motor begint is er een wacht blok: Controleer Sensor Tastsensor, Poort 1, Actie Ingedrukt.



**Pro Tip** Als je de schakelaar ingedrukt houdt, blijft de draaimolen draaien. Hoe komt dat? Kan je dat repareren?

## 06 Krokoklap

## Herhaal eeuwig met start en herstel

### Bouw

Maak een krokodillenbek die dichtklapt als je je vinger erin steekt.

Tips:

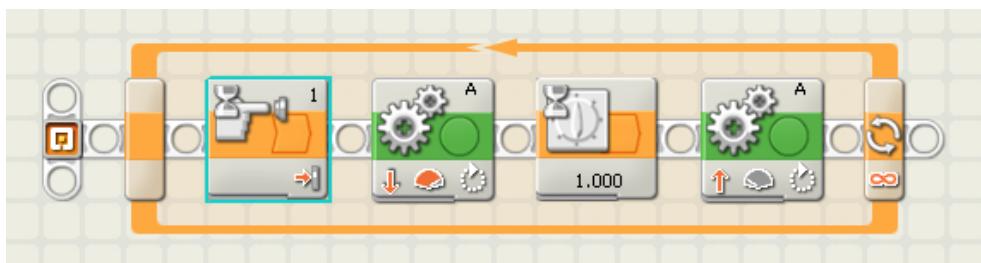
- Gebruik één motor (poort A), en één drukknop (poort 1).
- Gebruik tussen motor en bek twee wielen met een elastiek (voor “slip”).
- Omdat de drukknop wat hoog zit hebben we hem laten verspringen.
- Zorg dat de bek niet te ver naar achter kan (“aanslag”).



### Doeleind

Als je de tong in de krokodillenbek indrukt, klapt de bek dicht. Hij blijft heel even dicht, en gaat dan weer open. Daarna kun je het natuurlijk nog een keer (en nog een keer...) doen.

### Programma



Tips:

- We hebben een eeuwige herhaallus.
- Voor de motor begint is er een wacht blok: Controleer Sensor Tastsensor, Poort 1, Actie Ingedrukt.
- Daarna klapt de bek snel dicht: motor A *omlaag* draaien (Vermogen 100, Tijdsduur 20 graden).
- Daarna wachten we 1 seconde
- Daarna gaat de bek langzaam open: motor A *omhoog* draaien (Vermogen 15, Tijdsduur 20 graden).

**Let op:** als de bek dicht/open maar één keer lijkt te werken, zou het kunnen zijn dat de “open” stap niet afgemaakt kan worden. Dit gebeurt bijvoorbeeld als je het openen te ver wilt doen (Tijdsduur 90 graden), terwijl de bek niet verder kan. Als je goed luistert, hoor je dan de motor piepen.

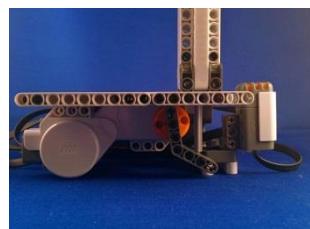
## 07 Kaars

## Microfoon, tonen en kalibreren

### Bouw

Maak een kaars met een vlam. De kaars gaat aan en uit door een motor die de vlam omhoog en omlaag beweegt. We steken de kaars aan met een aansteker (met dropknop). We doen de kaars uit door te blazen (bij de microfoon).

- Gebruik één motor (poort A).
- Gebruik een drukknop (poort 1) en een microfoon (poort 2).
- De kaars is een lege buis met een gat van 1x3.
- De kaars is gemonteerd op een plateau, precies boven een as-gat.
- De vlam is een as met een oranje bovenkant en een glij-blok van 1x3.
- De as van de vlam gaat door het as-gat in het plateau.
- Onder het as-gat in het plateau zit een motor met een “flap”.



### Doe!

Als je op de knop (“aansteker”) drukt, gaat de vlam omhoog (“aan”). Als je hard blaast gaat de vlam weer omlaag (“uit”). Natuurlijk kun je dit zo vaak doen als je wilt.

### Programma

- We hebben een eeuwige herhaallus met vier blokken
- Eerst een wacht blok: Controleer Sensor Tastsensor, Poort 1, Actie Ingedrukt.
- Daarna gaat de vlam aan: motor A *omhoog* draaien (Vermogen 20, Tijdsduur 60 graden).
- Het volgende blok wacht op geluid: Controleer Sensor Geluidssensor, Poort 2, Tot Geluid > 75.
- Daarna gaat de vlam uit: motor A *omlaag* draaien (Vermogen 20, Tijdsduur 60 graden).

**Pro Tip “kalibreren”:** Als je de geluidssensor geselecteerd hebt zie je het geluidsniveau “live” (hier 98). Bij ons hoorde niveau 75 bij flink blazen.



### Verbetering

Het zou natuurlijk leuk zijn om “happy birthday” te laten horen als de kaars is aangestoken. We voegen een geluidsblok toe met: Actie Toon en C gedurende 0.375 seconde. Zie hieronder voor de rest van de tonen.



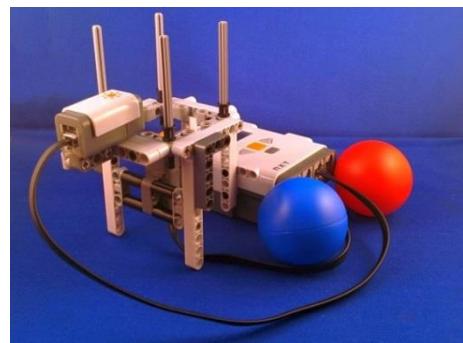
## 08 Kleurkenner

## Licht sensor, geluid en selectie

### Bouw

Maak een “tafel” waar je een bal in kan drukken en die dan vertelt (praten!) of de bal rood of blauw is.

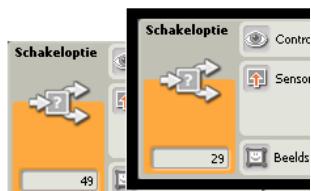
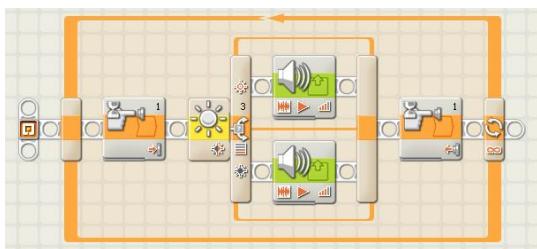
- Onder de tafel komt een drukknop (poort 1).  
De drukknop meet of er een bal op de tafel gedrukt wordt.
- Naast de tafel komt een licht sensor (poort 3).  
De licht sensor meet de kleur van de bal.
- Er is geen motor.
- De NXT zelf gaat “praten”.



### Doe!

Als de drukknop meet dat er een bal op de tafel gedrukt wordt, gaat de lichtsensor meten hoe licht de bal is. Is de bal licht (rood) dan roept de NXT “rood”, is de bal donker (blauw) dan roept de NXT “blauw”. Daarna kan je een nieuwe bal plaatsen.

### Programma



**Pro Tip “kalibreren”:** kijk welke licht-waarde de rode (hier 49) en de blauwe bal (hier 29) hebben. Neem een getal mooi in het midden (hier 39), en vul dat in bij Vergelijk van de schakeloptie.

- We hebben een eeuwige herhaallus.
- In het eerste blok wachten we tot de drukknop (poort 1) is ingedrukt.
- Het tweede blok is een samengesteld blok: schakeloptie. Dat kijkt naar de lichtsensor (poort 3) en kiest dan uit één van de twee paden, in dit geval aan de hand van de lichtsterkte, zet Vergelijk op 39.



- In beide paden komt een Geluid blok, Actie Geluidsbestand (afspelen), Bestand Red dan wel Blue.



- In het derde blok wachten we tot de drukknop (poort 1) weer is losgelaten.

### Verbetering

Zie je dat het licht in de lichtsensor aanblijft? Kun je het uitkrijgen?

## 09 Vuilniswagen

## Motor onbegrensd

### Bouw

Maak een vuilniswagen die vuilnis in een container kiept.

- Gebruik een motor voor het rechter en het linker voorwiel (poort B en C).
- Gebruik twee vrij draaiende wielen achter, of één los zwenkwiel
- Gebruik een motor om de laadbak te kunnen kiepen (poort A).
- Gebruik een drukknop (poort 1) om te weten of je bij de container bent.
- Maak een hefboom aan de aanraak sensor, anders duwt de vuilniswagen de container weg.



### Doeel

- De vuilniswagen heeft een soort laadbak voor het vuilnis.
- Wij gebruiken een bal als vuilnis.
- De vuilniswagen rijdt recht naar de container.
- Met een aanraak-sensor weet de vuilniswagen dat hij bij de container is.
- Dan stopt de vuilniswagen met rijden.
- Langzaam draait de laadbak en valt het vuilnis (de bal) in de container.
- Daarna draait de laadbak weer terug.
- Ten slotte rijdt de vuilniswagen weer een stukje achteruit.



### Programma

We zullen niet meer uitleggen hoe je de laadbak moet programmeren (rustig omlaag, even wachten, rustig omhoog). We zullen wel even uitleggen hoe je moet rijden tot een schakelaar wordt ingedrukt.

We beginnen met een beweeg blok dat motoren B en C aanzet (richting vooruit, Vermogen 20).

Wat hier speciaal is, is de tijdsduur: de motoren blijven altijd draaien – tot we ze stoppen in een ander blok.



Na dit “onbegrensde” beweeg blok komt een wacht blok: wacht tot de aanraak-sensor is ingedrukt.



Daarna komt weer een beweegblok. Maar nu zetten we de motor B en C op “Richting” stoppen!

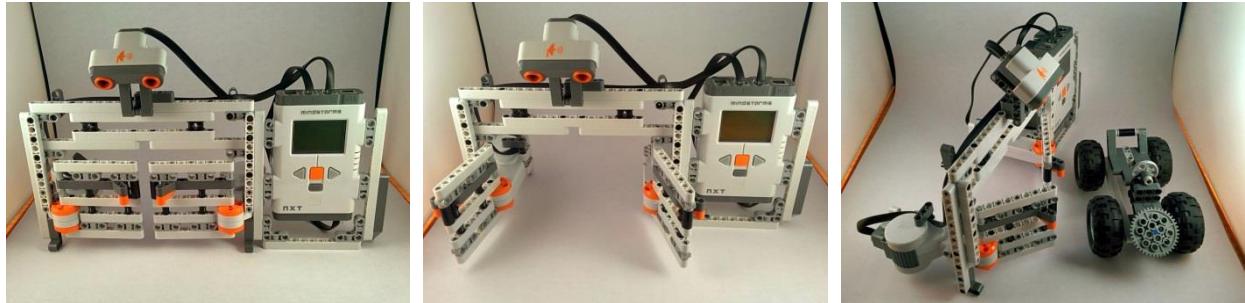


## 10 Garage

## Ultrasone sensor

### Bouw

Maak een garage met een dubbele deur die automatisch opengaat als er een auto aan komt.



- De linker deur gaat open met een motor op poort B.
- De rechter deur gaat open met een motor op poort C.
- De ultrasone sensor (poort 4) zit boven de deuren, schuin naar beneden.
- Maak ook een autootje dat groot genoeg is om gezien te worden door de ultrasone sensor.

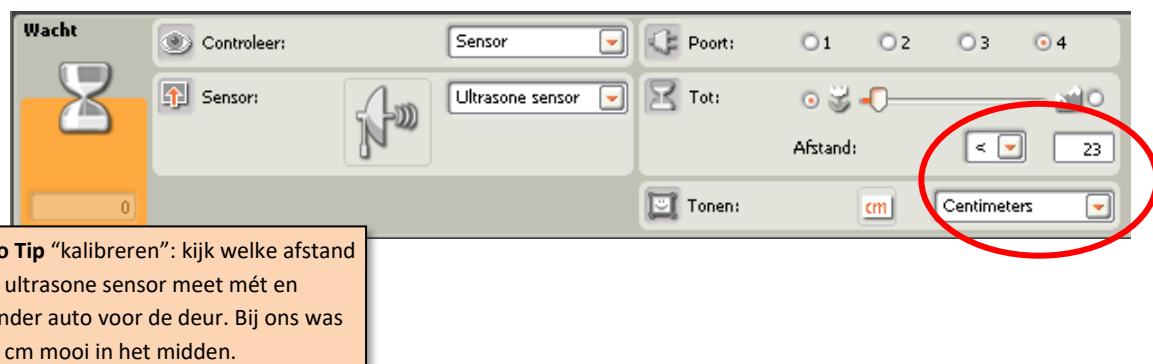
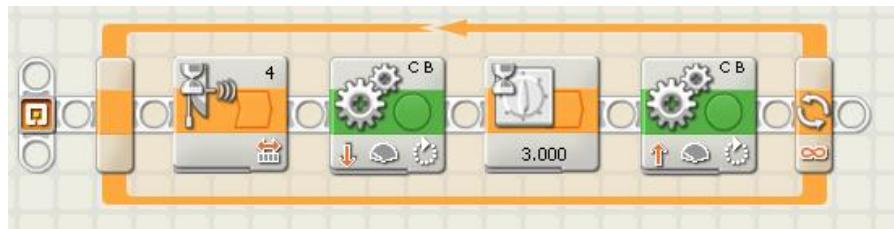
### Doe!

Als de auto aan komt rijden gaat de garagedeur vanzelf open. Na een tijdje gaat hij weer dicht.

### Programma

De ultrasoon sensor meet hoever hij kan kijken. Hij heeft een bereik van ongeveer 5 cm tot 250 cm.

- We hebben een eeuwige herhaal lus met vier blokken
- Eerst een wacht blok: Controleer Sensor Ultrasone, Poort 4, Tot afstand kleiner dan (<) 23cm.
- Daarna gaan de deuren open: motor B en C opendraaien (Vermogen 20, Tijdsduur 120 graden).
- Het volgende blok wacht enkele seconde.
- Daarna gaan de deuren dicht: motor B en C dicht draaien (Vermogen 20, Tijdsduur 120 graden).



## 11 Afstand bestuurbare auto

Schermscherm

### Bouw

Maak een auto met aparte motoren voor rechts en links en één schakelaar waarmee we hem vooruit, linksaf en rechtsaf kunnen laten gaan.

- De linker motor op poort B.
- De rechter motor op poort C.
- De druk knop op poort 1.
- Zet achterop een zwenkwiel.



### Doeleind

Het is de bedoeling dat deze auto drie modi heeft: rechtsgaand rijden, linksaf draaien en rechtsaf draaien. We laten steeds met het beeldscherm zien wat de huidige mode is. Voor rechtsgaand rijden bijvoorbeeld:

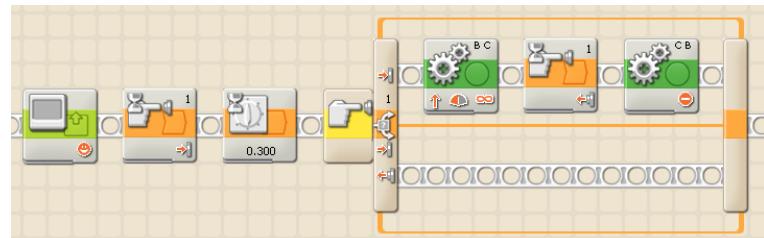


De truc die we gebruiken: een korte druk op de knop gaat naar de volgende mode. Een lange druk op de knop laat de auto echt de beweging maken die bij die mode hoort (rechtsgaand, linksaf of rechtsaf).

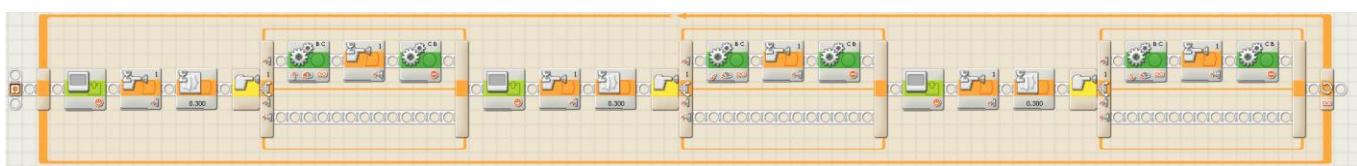
### Programma

Hiernaast zie je de afhandeling van één mode:

- Beeldscherm geeft de mode weer
- Wacht tot de knop wordt ingedrukt
- Wacht 0.3 seconde (lekker kort)
- Inspecteer de knop
- Is hij nog steeds ingedrukt, dan zet de motoren B en C “eeuwig” aan
- wacht tot de knop is losgelaten, en zet dan de motoren uit.



Het hele programma bestaat uit drie keer “hetzelfde” voor de drie modi, in een eeuwige herhaal lus:



### Verbetering

Het zou natuurlijk leuk zijn als de auto niet alleen op het scherm laat zien wat de mode is, maar als hij dat ook nog uit zou spreken (met een geluidsblok).

Heb je gezien dat je op het beeldscherm ook een hele grote pijl kan laten zien?



## 12 Go home

## Rotatie sensor en dataverbindingen

### Bouw

Maak een auto met vier wielen en een knop aan de voorkant. Maak ook een vlaggetje.

### Doe!

Het is de bedoeling dat deze auto vertrekt vanaf het vlaggetje, rechtdoor rijdt tot hij ergens tegenaan botst, even wacht, weer terug rijdt, en *precies stopt op de plek waar hij begon!* De knop “voelt” het botsen.



### Programma

Het programma zou vanzelf moeten spreken, behalve de nieuwe gele blokken en het touwtje:

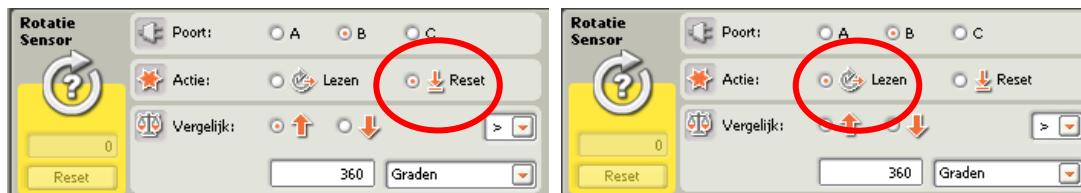


**Pro Tip:** Type uitleg bij je programma, net als hier.

### Gele blokken

Het eerste gele blok zet de rotatiesensor (“kilometerteller”) van motor B op 0 (Actie: Reset).

Het tweede gele blok vraagt de stand van de rotatie sensor op (Actie: Lezen).



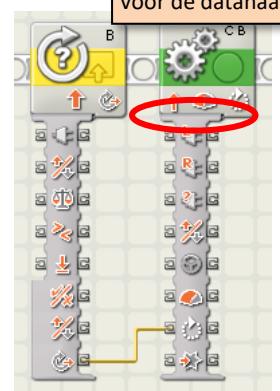
**Pro Tip:** klik hier voor de datanaaf.

### Touwtjes

Touwtjes geven waarden door van het ene blok naar het andere. Elk blok heeft daarvoor *dataverbindingen*. Deze worden zichtbaar als je de *datanaaf* links onder open klikt.

De gelezen stand van de rotatiesensor komt uit de onderste dataverbinding van het rotatiesensor blok (met de naam “Graden”). We trekken een touwtje naar de één na onderste dataverbinding van het motorblok (met de naam “Tijdsduur”).

Als je weer op de datanaaf klikt wordt hij zo klein mogelijk gemaakt.



## 13 Landmeter

## Datablokken

### Bouw

Maak een auto met vier wielen, een schakelaar aan de voorkant, en een bumper aan de achterkant.



### Doele

Het is de bedoeling dat deze auto met zijn bumper tegen de muur van een kamer wordt gezet, dat hij daar vertrekt en net zolang rijdt tot hij aan de overkant van de kamer is (schakelaar botst tegen muur). Op het display moet dan staan hoe lang de kamer is (in centimeters).

### Programma

We beginnen met een eenvoudige versie van het programma.

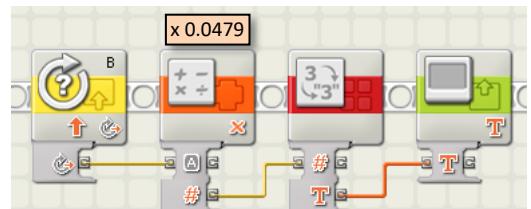


Het eerste gele blok zet de "kilometerteller" van motor B op 0, het tweede blok vraagt de eindstand op. Dit *getal* wordt eerst (rode blok) omgezet in een *tekst*, daarna wordt de tekst op het beeldscherm gezet:



### Rekenen

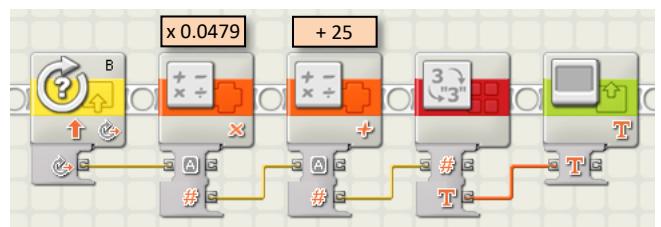
Laat de auto nu precies 100 centimeter rijden, en kijk op het scherm hoeveel graden dat is. Bij ons moesten de motoren 2088 graden draaien om 100 centimeter te rijden. Dus, voor elke graad draaien, rijden we 0.0479 centimeter vooruit. Zet dit oranje vermenigvuldigingsblok in je programma, en rijd nog een paar keer 100 centimeter om de vermenigvuldiging te controleren. Rijd ook eens 150 centimeter.



**Let op:** Als jouw auto een andere lengte heeft, of een andere maat wielen, moet je je eigen getallen bepalen!

Als we van muur tot muur willen meten, moeten we de lengte van de auto zelf er ook nog bij optellen. Onze auto bleek 25 centimeter te zijn. Zet dit optelblok ook nog in je programma.

Testen!



## 14 Adaptive Cruise Control

## Parallelle taken

### Bouw

Maak een auto met vier wielen en twee sensoren aan de voorkant: een schakelaar en de ultrasone sensor (afstandsmeter).

### Doe!

Het is de bedoeling dat deze auto op topsnelheid rijdt, tot de ultrasone sensor een muur aan ziet komen. Hoe dichter de auto bij de muur komt, hoe langzamer hij gaat rijden.

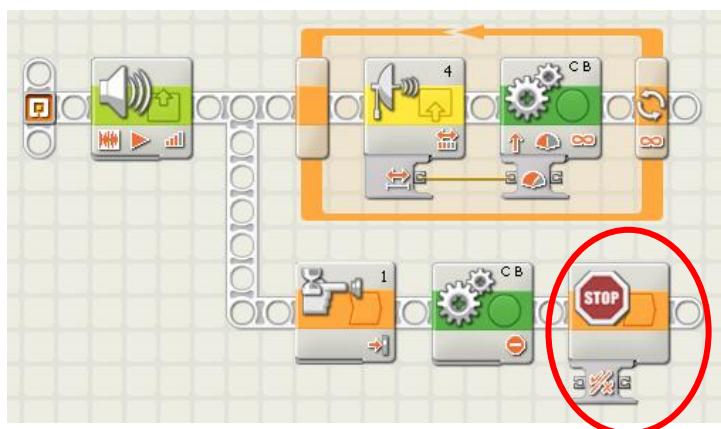
Als de schakelaar de muur raakt, dan stopt de auto.



### Programma

Dit programma bestaat uit drie delen.

- Het eerste is een biep om aan te geven dat de auto gaat rijden.
- Daarna volgt een splitsing: nu gaan twee taken naast elkaar lopen!
- De bovenste taak gebruikt de afstand uit de ultrasone sensor om de snelheid van de motoren in te stellen. Dat gebeurt continue in delus.
- In de onderste taak wordt gewacht tot de schakelaar wordt ingedrukt. Als dat gebeurt wordt de motor uitgezet, en het programma gestopt.



**Tip (parallelle taken):** maak eerst één taak op de gewone manier. Voor de tweede taak zet alvast het eerste blok los neer. Hoover met de muis boven het aftakpunt. Na even wachten zie je de Technic staaf iets zwaarter worden. Klik met de muis. Door met de muis te bewegen teken je een nieuwe Technic staaf. Met extra muis klikken kun je knikken maken. Klik ten slotte op het beginpunt van het nieuwe blok.



**Tip:** Het gele sensor blok zit op de tweede tab (bij de gele 'sensor' blokken).

**Tip:** Het oranje STOP blok zit ook op de tweede tab (bij de oranje 'verloop' blokken).

### Verbetering – Schokkend rijden

Een ultrasone sensor geeft af en toe een slechte waarde terug (veel te groot of veel te klein – probeer maar eens). De auto rijdt dan met schokken. Het is beter een paar uitlezingen te doen, en die te middelen.

**Pro Tip:** Als je het helemaal sjiek doet maak je een derde taak die een lopend gemiddelde bijhoudt!

### Verbetering – Juiste snelheid

De ultrasone sensor geeft een uitlezing van 0 tot 255 cm. Hierboven gebruiken we deze afstand om de snelheid van de motoren in te stellen. Dat is niet helemaal netjes: de grootste snelheid is niet 255 maar 100. Nog een probleem: als de afstand klein is dan wordt de ingestelde snelheid zó klein (~15) dat de motoren niet meer draaien; de auto staat stil.

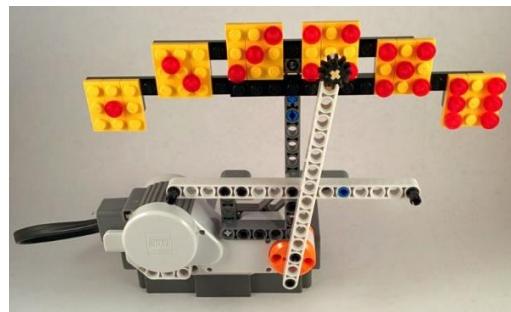
**Pro Tip:** Om de afstand om te zetten in een snelheid gebruik je óf een formule, óf meerder schakelopties.

## 15 Voorleesbord

## Meerweg schakeloptie en nulstand

## Doel

Een "voorleesbord" is een bord met plaatjes en een wijzer. Als je met de wijzer een van de plaatjes aanwijst, zal het bord voorlezen wat er op het plaatje staat.

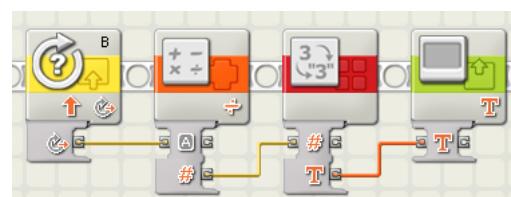


Bouw

Wij maken een voorleesbord met de getallen 1 tot en met 6. De wijzer zit aan een motor, die we als rotatie sensor gebruiken.

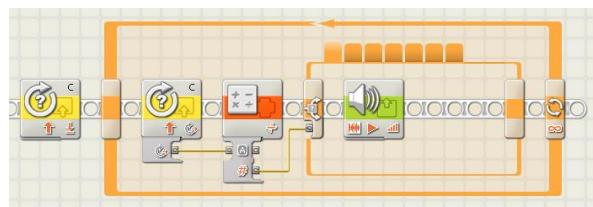
**Tip:** schrijf eerst dit programma, en zorg dat de gele bordjes met de getallen op vaste afstanden zitten (bij 15 graden).

**Tip:** Zorg dat er ook een positie “0” is.



## Programma

De kern van het programma zie je hieronder. Het reset eerst de rotatie sensor ("kilometerteller op 0") en gaat dan een oneindige lus in. In de lus lezen we steeds de hoek van de rotatie sensor, delen die door 15 en spelen dan het juiste geluidsfragment af.



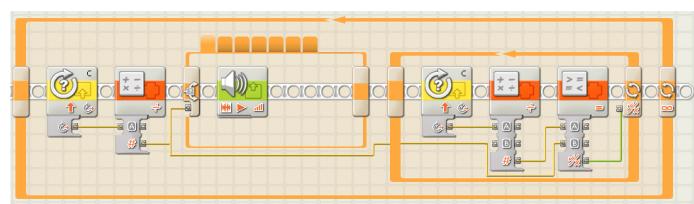
Uit de deling komt dan een getal van 1 tot en met 6. De schakeloptie leest bij 1 "één" voor, bij 2 "twee", etc. Als je de wijzer te veel naar recht of links beweegt komt er iets anders dan 1..6. Voeg nog één optie toe voor al deze gevallen (zet \*), hier is het bord stil.

The screenshot shows the Scratch script editor with a control block selected. The block has the following parameters:

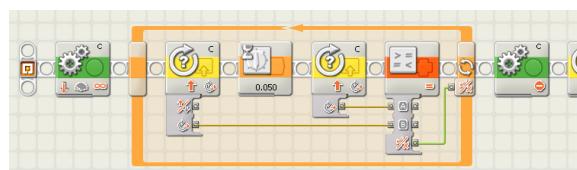
- Controleer:** Waarde
- Type:** Getal
- Voorwaarden:** A table with five rows:

1.	0
2.	1
3.	2
4.	3
5.	4
- Beeldscherm:** Vlak beeld

### Minder voorlezen:



Motor naar de nulstand:



**Tip:** We hebben gekozen dat onze wijzer stapjes van 15 graden per geel bordje maakt:

0..14	graden levert na deling door 15 op: 0
15..29	graden levert na deling door 15 op: 1
30..44	graden levert na deling door 15 op: 2
45..59	graden levert na deling door 15 op: 3
60..74	graden levert na deling door 15 op: 4
75..89	graden levert na deling door 15 op: 5
90..104	graden levert na deling door 15 op: 6

**Tip:** Instellingen schakeloptie: "Controleer" op "Waarde" (de uitkomst van de deling); het "Type" is een "Getal"; en "Beeldscherm" is *niet* "Vlak beeld". Dit laatste zorgt ervoor dat we meerdere "Voorwaardes" kunnen toevoegen.

Voeg als voorwaarde 1 getal 1 toe, als voorwaarde 2 getal 2, en zo voort tot voorwaarde 6 getal 6. Voeg ook voorwaarde 7 toe als default (zet \*).

Zet in de tab een geluidsblok dat het juiste getal voorleest  
(voorwaarde 1 leest "1" voor)

**Pro tip:** De oneindige lus blijft maar voorlezen. Wij hebben een lus toegevoegd die blijft wachten tot de wijzer een ander vak aanwijst dan wat net voorgelezen is.

**Super pro tip:** Als je de robot zomaar uit zet en dan weer aan, dan staat de wijzer niet altijd helemaal links. En dan heeft de robot verkeerde getallen voor

Wij laten daarom de motor zachtjes naar links draaien tot twee uitlezingen hetzelfde zijn. Dan stoppen we met lus-en en zetten de motor uit.

## 16 Dobbelsteen

## Random en nulstand

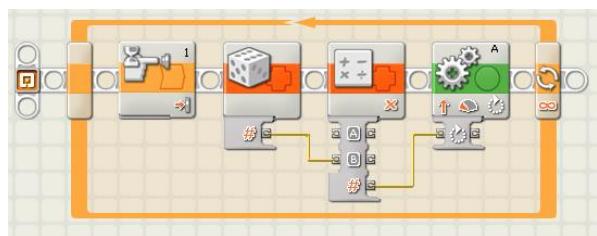
### Bouw en doel – 1

Maak een elektronische dobbelsteen. Maak hiervoor een wiel met de cijfers 1 tot en met 6. Maak het cijferwiel vast aan een motor. Zet ook een schakelaar op je robot. Het is de bedoeling dat na een druk op de schakelaar, de robot een willekeurig ("random") getal kiest. Hij laat het getal zien door met de motor het cijfer voor te draaien.

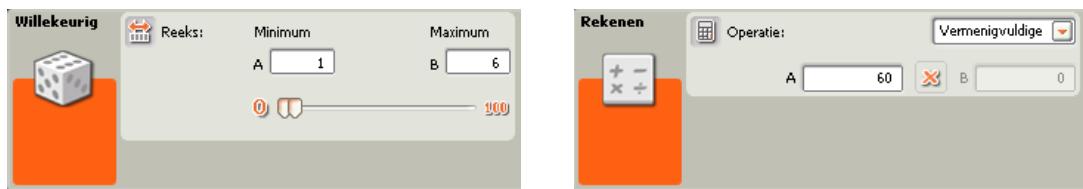


### Programma – 1

De eerste versie van het programma is redelijk standaard. We hebben een oneindige lus. Daarbinnen beginnen we met wachten op een druk op de knop. Dan komt er iets nieuws: we kiezen met het 'Willekeurig' blok een getal tussen 1 en 6 – dit is de worp met de dobbelsteen.



Hoeveel moeten we nu draaien? Een hele cirkel is 360 graden. Ons cijferwiel heeft 6 standen. Dus elke stand is 60 graden. We moeten de uitkomst van het 'Willekeurig' blok dus met 60 vermenigvuldigen.



### Bouw en doel – 2

We willen een uitbreiding. De dobbelsteen moet voorlezen wat we gegooid hebben. Dan moet de robot weten wat de "nulstand" is. We maken daarom aan het cijferwiel een wit blok. We plaatsen de lichtsensor zó dat hij het witte blok ziet, en dus de nulstand weet.

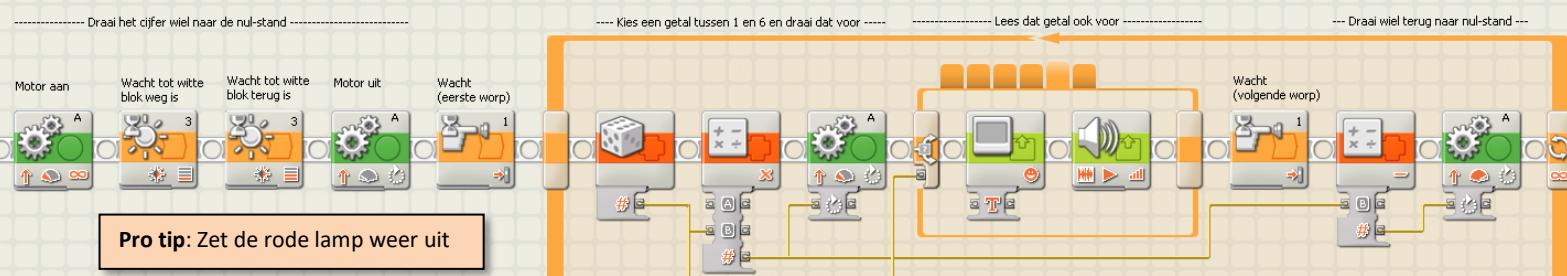


### Programma – 2

Het programma bestaat nu uit de volgende delen:

- Zoek de nulstand: draai het cijferwiel tot het witte blok weg is, en daarna tot het witte blok er is.
- Wacht tot de knop wordt ingedrukt voor de eerste worp.
- Kies een getal tussen 1 en 6, draai het cijferwiel, en lees het getal voor.
- Wacht tot de knop wordt ingedrukt voor de volgende worp.
- Draai eerst even het cijferwiel in de nulstand.

**Pro tip:** Het is extra stoer als het wiel eerst een paar keer (random?) hard draait.



## 17 Ronde teller

## Variabele

### Bouw en doel

Dit is meer een programmeeropdracht dan een bouwopdracht. We gaan een ronde teller maken. Dat is een apparaat dat telt bijvoorbeeld hoe vaak een raceauto langs de finish streep komt. Het bouwwerk is niks meer dan een ultrasoon sensor aan de NXT vastgemaakt. Deze opdracht gaat over programmeren (met variabelen).



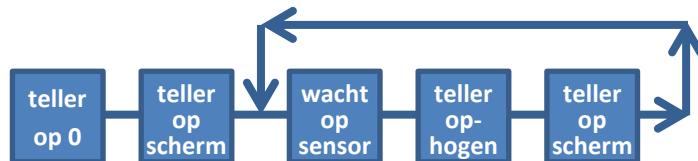
### Programma

De structuur van het programma zal duidelijk zijn:

- We hebben een oneindige lus
- Iedere keer als de ultrasoon sensor wat dichtbij ziet wordt de ronde teller opgehoogd.

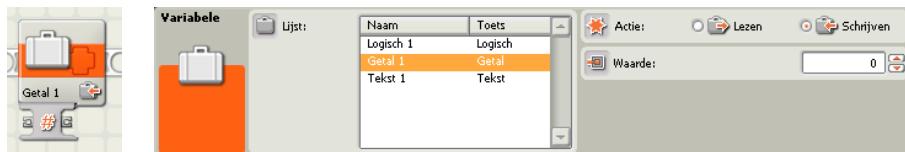
Laten we dit eens in meer detail beschrijven

1. De teller wordt op nul gezet
2. We zetten de teller op het scherm
3. Wacht tot de ultrasoon sensor wat ziet
4. De teller wordt opgehoogd
5. We zetten de nieuwe teller op het scherm
6. Oneindige lus gaat weer naar stap 3



Hoe kunnen we een teller bijhouden? Dat doen we in een *variabele*. Lego gebruikt daarvoor het plaatje van een koffer. Het idee van dat plaatje is: we onthouden de stand van de teller in een koffer. Let op:

- Een koffer kan 3 soorten dingen bevatten: *Logisch* (goed/fout), *Getal* (0,1,2,...) of *Tekst* ("Game over").
- Een blok heeft twee Acties: iets uit de koffer halen (*Lezen*), óf iets nieuws erin stoppen (*Schrijven*)



Hieronder kun je spieken hoe je het programma eruit zou kunnen zien. Let op:

- Voor 'teller op scherm' gebruiken we een 'koffer', en het 'Getal naar Tekst' blok.
- 'Wacht op sensor' wacht tot de afstand klein is en wacht daarna tot de afstand groot is. Anders zou het tellen maar door blijven gaan.
- Teller met één ophogen is drie stappen.
  - We hebben eerst een koffer blok om de (oude) waarde te lezen.
  - Dan hebben we een optel blok om er één bij op te tellen.
  - En tenslotte hebben we een koffer blok om de nieuwe waarde te schrijven.

**Pro tip:** Fragment "teller op scherm" komt twee keer voor in het programma. Kun je er een weg krijgen?



## 18 Parking

## Compleet project

### Bouw

Bouw een parkeerplaats met

- Drie parkeervakken
- Een in-rij slagboom (aan een motor) met een drukknop
- Een uit-rij slagboom (aan een motor) met een drukknop (als er geen drukknop meer is, neem dan een lichtsensor)
- Een scherm, bij de in-rij slagboom



### Doe!

Hoe moet de parkeerplaats gaan werken?

- Op het scherm staat altijd een *teller* die het aantal lege vakken geeft.
- Als je op de knop drukt bij de *in-rij* slagboom dan (1) gaat de boom open, (2) wordt er even gewacht, (3) gaat de boom weer dicht, en (4) de *teller* op het scherm wordt één lager (de nieuwe auto neemt een plek in).
- Als je op de knop drukt bij de *uit-rij* slagboom dan gaat de boom ook open/wacht/dicht, en de *teller* op het scherm wordt één hoger (een auto rijdt weg, dus er komt een plek vrij).

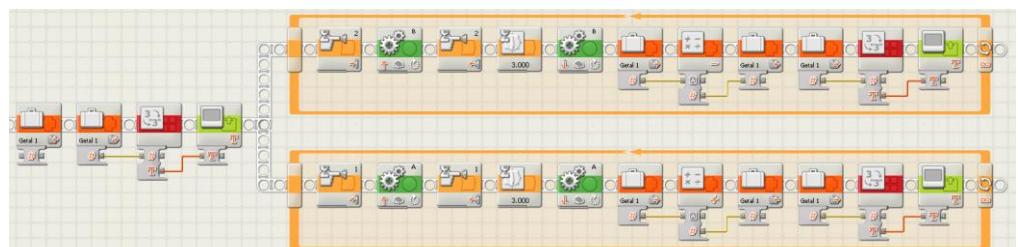
### Programma

Het programma lijkt heel veel op het vorige programma. We gebruiken weer een ‘koffer’, nu voor de *teller* die het aantal lege vakken bij houdt. Zie hieronder een schets van het programma van de *in-rij* slagboom.



Parallel hieraan moet er net zo'n programma komen dat de *uit-rij* slagboom bedient.

En bij het begin moet het scherm ook gevuld zijn. De hele structuur staat hiernaast.



**Pro tip:** Dit programma bevat een “bug”: als beide schakelaars ongeveer tegelijkertijd worden ingedrukt, dan klopt de teller niet meer. Probeer maar eens (voeg een ‘wacht’ toe).

### Verbetering

Als je op de knop drukt bij de *in-rij* slagboom en de teller staat op 0 (er zijn geen vrije plaatsen), dan mag je nog steeds naar binnen. Verbeter dat: laat een “boze biep” horen en zorg dat de slagboom niet opengaat.

## 19 Blokrijder

## Besturing en timing

### Bouw

Maak een auto die kan sturen.

Het makkelijkst is

- één motor voor het rechter voorwiel
- één motor voor het linker voorwiel
- een zwenkwiel achter



### Doe!

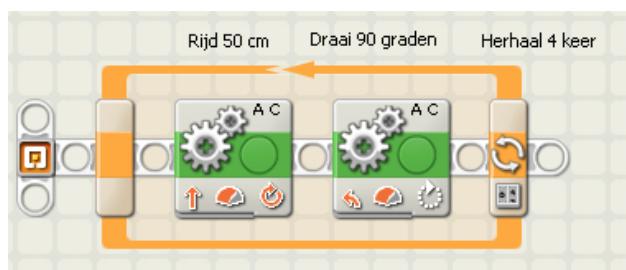
Het doel is dat de auto een blokje ("rondje") op een tafel rijdt.

Een rondje op de grond is veiliger in verband met vallen.



### Programma

Het programma is simpel. We gaan vier keer herhalen: rechtdoor rijden en draaien.



### Timing

De moeilijkheid zit hem in het 90 graden laten draaien van de *auto*. Het is eenvoudig om een *motor* 90 graden te laten draaien. Maar om de auto te laten draaien moeten we kiezen hoe lang en hard we de motoren moet laten draaien. Dat moeten we net zolang proberen tot het klopt.

Tip zet "Besturing" op "bochtje", en probeer "Tijdsduur" goed te krijgen.



### Let op

We regelen in deze opdracht het gedrag van een robot met timing. Dat is een *slecht* idee. Als de batterijen leger of voller zijn, als de banden harder of zachter zijn, als de vloer ruwer of gladder is, dan klopt de timing niet meer. Lego NXT heeft helaas geen sensor om oriëntatie te meten.

## 20 Doosvolger

## Regel lus

### Bouw

Bouw een auto die kan sturen, en die een afstand (ultrasoon) sensor heeft.

Het makkelijkst is:

- één motor voor het rechter voorwiel
- één motor voor het linker voorwiel
- een zwenkwiel achter

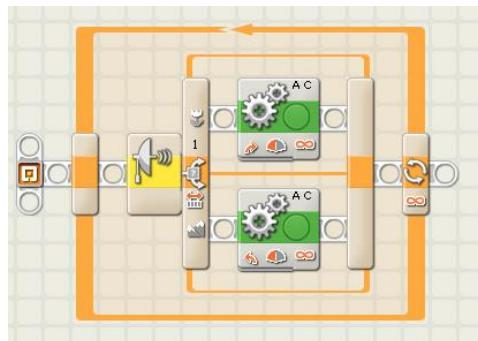


### Doe

De auto moet rondjes rond een doos kunnen rijden, zonder de doos te raken.

### Programma

We laten de hele tijd beide motoren draaien. Maar als we te dicht bij de doos zijn, sturen we van de doos af, en als we te ver van de doos zijn, sturen we naar de doos toe.



Hieronder zie je een voorbeeld van de motor aansturing. A (links) en C (rechts) beide Onbegrensd aan, maar wel met Besturing naar rechts.

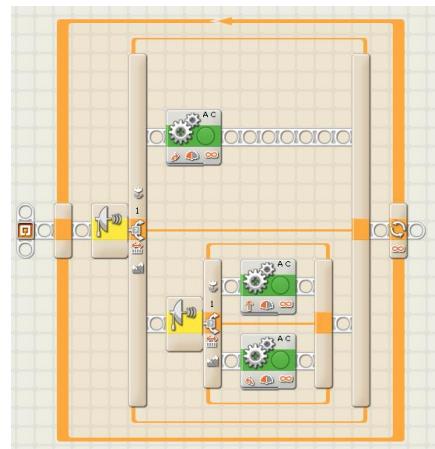


### Verbetering

De auto rijdt iets mooier als je niet twee gevallen (te dichtbij of te ver weg) maar drie gevallen onderscheidt:

- te dichtbij: stuur van de doos weg;
- mooie afstand: rijdt rechtdoor,
- of te ver weg: stuur naar de doos toe.

Je krijgt dan een schakeloptie in een schakeloptie, zie programma hiernaast.

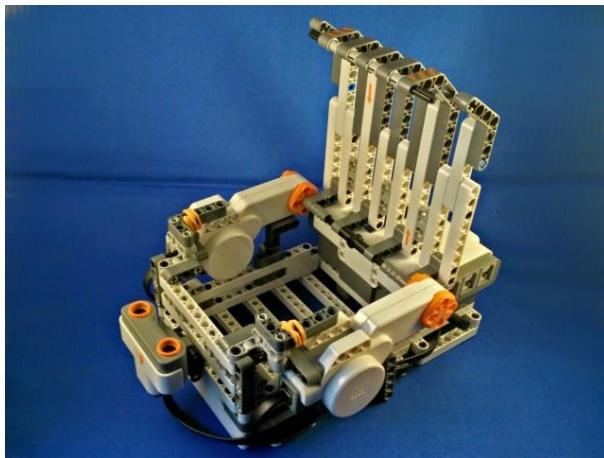


## 21 Koekjeskluis

## Compleet project

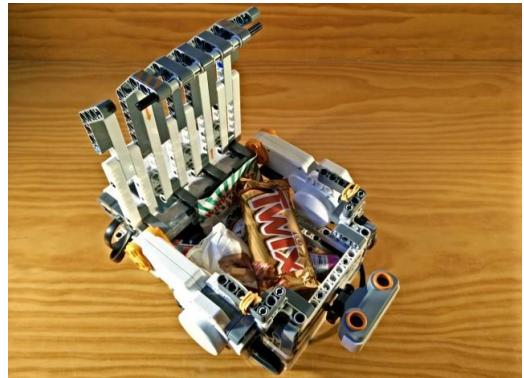
### Bouw

Bouw een stevige doos, met een stevige deksel. We gaan hierin onze koekjes bewaren! Gebruik twee motoren om de deksel te openen en te sluiten. Monteer een afstand (ultrasoon) sensor bij de deksel om te kijken of iemand onze koekjes probeert te stelen.



### Doeleind

Maak een veilige plaats voor koekjes. Laat zien hoeveel en hoe lekkere koekjes je hebt. Maar als iemand zijn hand dichtbij houdt, gaat de deksel snel dicht. En na een tijdje weer open.



### Programma

- We hebben een eeuwige herhaallus.
- Voor de motor begint is er een wacht blok: Controleer Sensor Ultrasoon Afstand kleiner dan 20 cm.
- Daarna klapt de deksel snel dicht: beide motoren snel *omlaag* draaien (90 graden).
- Daarna wachten we 5 seconde
- Daarna gaat de deksel met beide motoren *omhoog* (90 graden).



## 22 Pincode

## Compleet project

### Bouw

Bouw een stevige doos, met een stevige deksel, onze kluis. Gebruik twee motoren om de deksel te openen en te sluiten. Monteer de druk knop. Monteer ook de derde motor met wiel – we gaan die gebruiken als rotatie sensor om de pincode in te voeren.

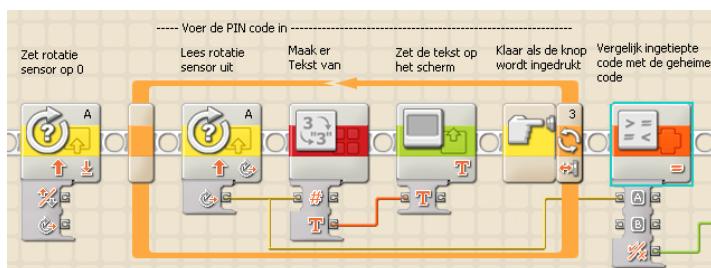


### Doel

Door aan het wiel te draaien verandert de pincode (die op het scherm staat). Heb je de pincode gekozen, druk dan op de knop. Had je de verkeerde pincode gekozen dan blijft de deur dicht. Was de code juist, dan zal de kluisdeur opengaan. Druk weer op de knop om de deur te sluiten.

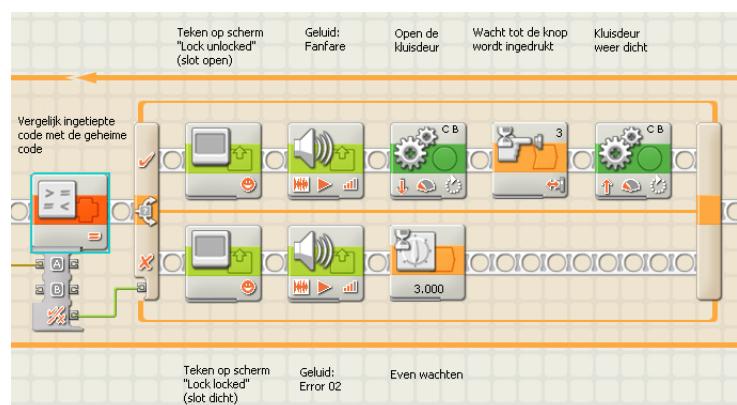
### Programma

Het programma bestaat uit twee belangrijke delen. Het eerste deel zorgt voor het invoeren van de pincode. Het eerste blok zet de rotatie sensor op 0. Daarna wordt de rotatie sensor uitgelezen, tekst van gemaakt en op het scherm gezet. Dit wordt herhaald tot de knop wordt ingedrukt. Dan wordt de ingevoerde code vergeleken met de geheime code

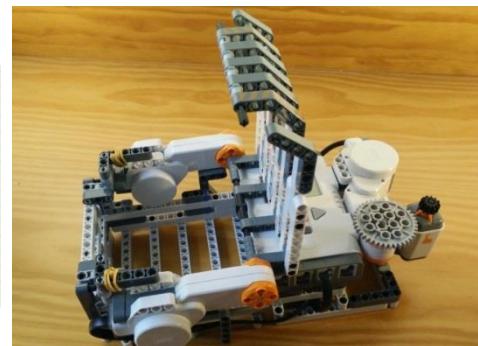
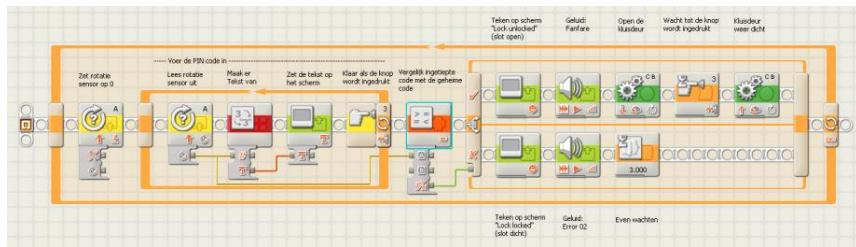


**Pro tip:** Gebruik gewoon een Vergelijk blok en zet B op de geheime pincode (hier 825).

Het tweede deel bedient de kluisdeur. Het begint met de pincode test. Als de pincode goed is wordt op het scherm een open slot getekend met Fanfare geluid. De kluisdeur wordt geopend. Als de knop wordt ingedrukt wordt de deur weer gesloten. Als de pincode fout is wordt op het scherm een gesloten slot getekend met een Error geluid.



Verder staat om deze twee delen nog een oneindige herhaal lus.

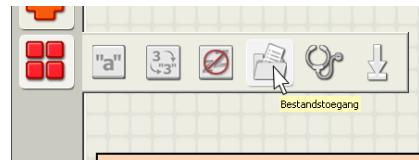


## 23 Memory

## Bestanden en NXT knoppen

### Doe!

Dit is een programmeeropdracht, we bouwen niks. We gaan een “memory machine” maken. Als je programma 1 opstart zal de tekst “Dit is geheim!” bewaard worden. Als je programma 2 opstart zal die tekst op het display gezet worden. We gebruiken hiervoor iets nieuws: bestanden.

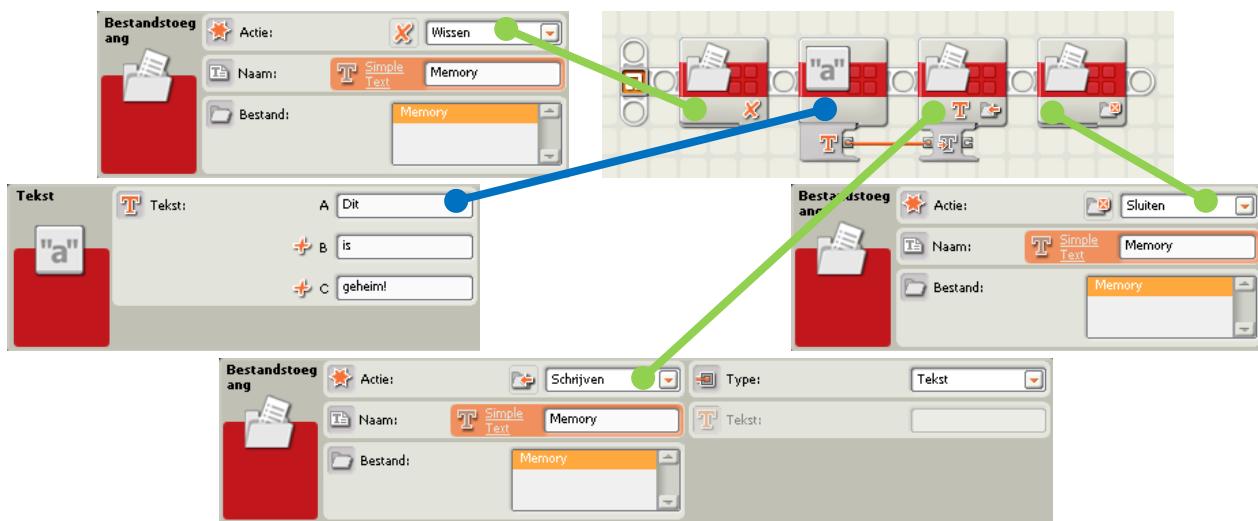


**Pro tip:** Bestandstoegang vind je in het tweede tab, bij de rode (geavanceerd) blokken.

### Programma 1 – schrijven

Als je getallen (of teksten) blijvend wilt bewaren heet dat in de computerwereld een bestand (*file* in het Engels). De NXT heeft bestanden, en er is één blok “bestandstoegang” om er mee te werken.

Hieronder staat een 4-staps programma om de tekst “Dit is geheim!” in het bestand “Memory” op te slaan.



Het eerste blok *wist* het bestand “Memory”. Dat is nodig, omdat de schrijf actie die we later doen *toevoegt*, en we willen *vervangen*. Het tweede blok is geen bestands actie, het produceert de tekst “Dit is geheim!”. Het derde blok *schrijft* de tekst die binnenkomt via de datanaaf naar bestand “Memory”. Merk op dat we het *Type* inderdaad op *Tekst* hebben gezet (een bestand kan ook getallen bewaren). De vierde stap is veilig: we *sluiten* bestand “Memory”. De stap is eigenlijk alleen nodig als het programma later gaat lezen.

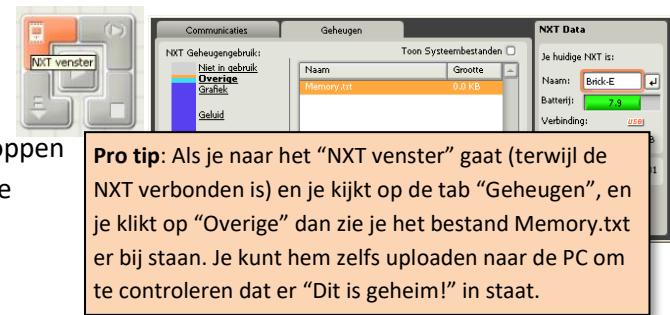
### Programma 2 – lezen

Hieronder zie je het leesprogramma. Het eerste blok leest bestand “Memory” (met *Type Tekst*).



### Verbetering

Schrijf een programma dat een *getal* onthoudt. Als je het opstart, komt het getal op het display. Met de pijltjes knoppen op de NXT kun je het getal veranderen. Als je op de oranje knop drukt, zal het nieuwe getal onthouden worden.



**Pro tip:** Als je naar het “NXT venster” gaat (terwijl de NXT verbonden is) en je kijkt op de tab “Geheugen”, en je klikt op “Overige” dan zie je het bestand Memory.txt er bij staan. Je kunt hem zelfs uploaden naar de PC om te controleren dat er “Dit is geheim!” in staat.

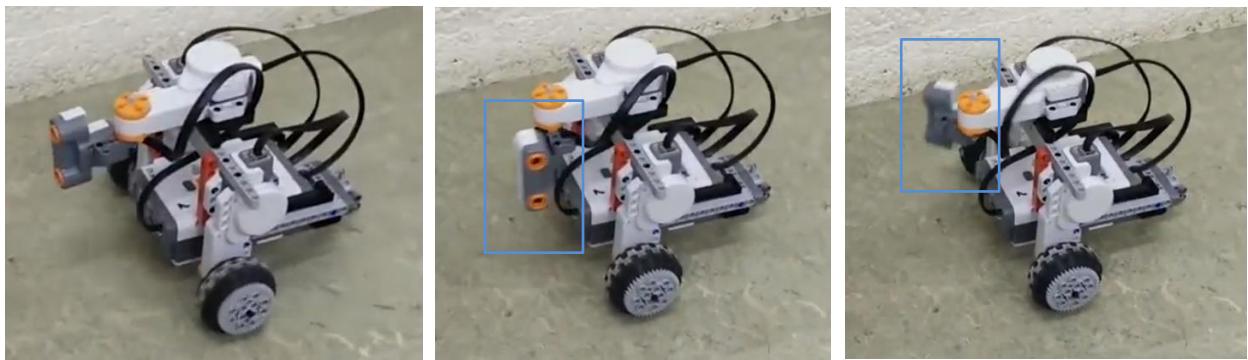
## 24 Radarwagen

## Compleet project

### Bouw

Bouw een auto die kan draaien; bijvoorbeeld één wiel links met een motor en één wiel rechts met een motor en een zwenkwielachter. Zet bovenop de auto een radar: een ultrasone afstand sensor die heen en weer kan zwenken (op een derde motor dus).

Hieronder zie je een voorbeeld van een radar-auto. Links staat de radar recht vooruit, op de ander foto's zie je de radar gedraaid.

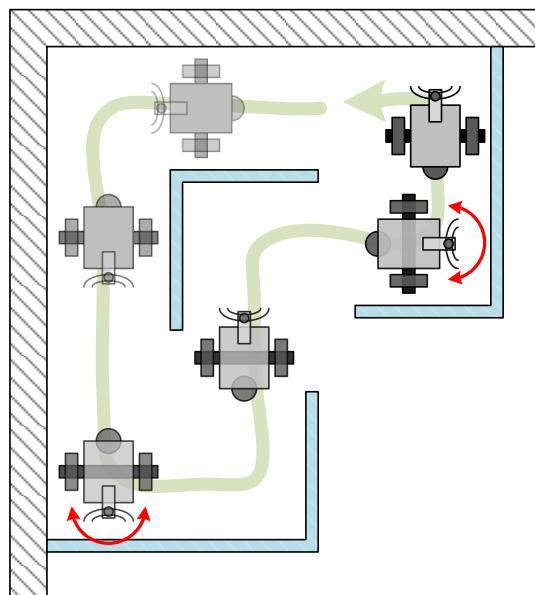


### Doe

Normaal staat de radar in de "recht vooruit stand". De auto rijdt ook recht vooruit. Dat blijft hij doen tot de radar een obstakel ziet op zeg 25cm. Dan stopt de auto.

De auto moet nu beslissen of hij naar rechts of naar links gaat sturen. Hiervoor draait de radar naar links en doet een afstandsmeeting. Daarna draait de radar naar rechts en doet een afstandsmeeting. Daarna bepaalt de auto of rechts of links meer ruimte is. Die kant stuurt hij vervolgens op.

De auto zou dan de groene route moeten kunnen rijden:



### Programma

Zelf te bepalen

## 25 Lift

## Compleet project

### Bouw

Bouw een lift van ten minste drie verdiepingen.

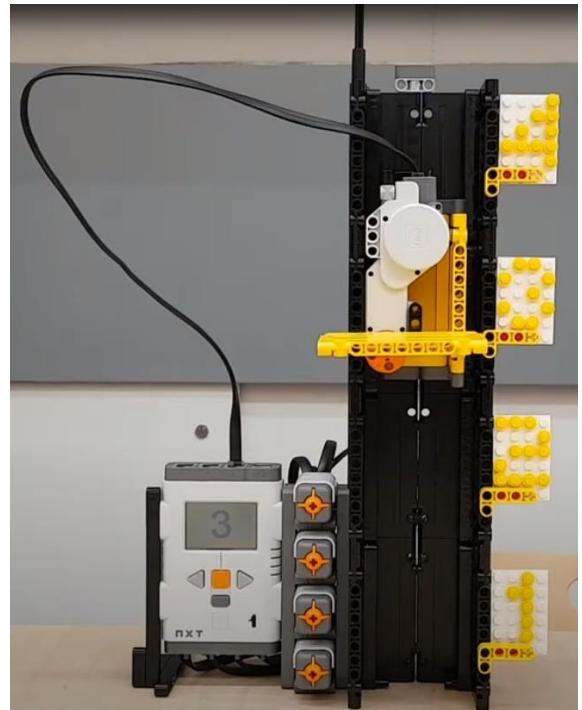
Het best werkt het als de cabine in een soort rails loopt, waarbij de motor een tandwiel aandrijft die in een tandheugel loopt.



Je kunt de cabine ook vrij ophangen aan een touwtje op een spool die door de motor wordt aangedreven. Lijkt makkelijker, maar werkt slechter.

De tweede foto laat zien hoe wij de liftschacht hebben gebouwd.

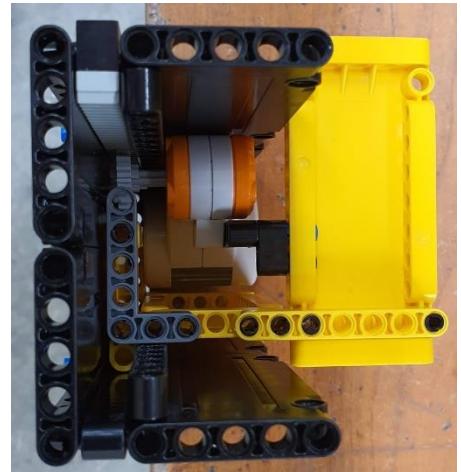
Plaats een knop per verdieping.



### Doe!

Als je op knop 1 drukt, moet de cabine naar verdieping 1 gaan. Als je op knop 2 drukt moet de cabine naar verdieping 2 gaan en voor knop 3 naar 3. Het scherm laat zien op welke verdieping de cabine hangt.

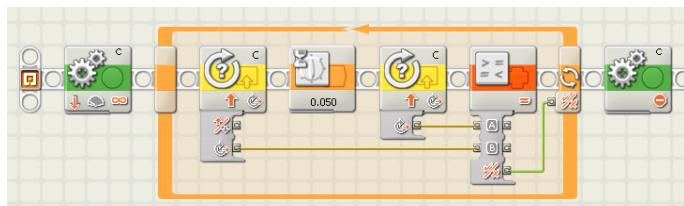
Pro: Als je de NXT aanzet, zou hij zelf moeten uitzoeken waar de cabine is. Typisch doe je dit door de cabine naar beneden (of naar boven) te bewegen tot hij een knop raakt (de nulstand). Je kunt ook de truc gebruiken die hieronder staat: moeilijkere software, makkelijkere hardware.



### Programma

Zelf te bepalen

Om de nulstand van de cabine te bepalen, kun je de sensor in de motor gebruiken. Laat de motor zachtjes (!) omlaag draaien tot twee uitlezingen hetzelfde zijn. Dan immers zit de cabine vast tegen de grond. Stop met lus-en en zet de motor uit.



## 26 Keukenwekker

## Compleet project

### Bouw

Maak een keukenwekker.

Dat wil zeggen:

- Een wijzer aan een motor
- Een wijzerplaat met “streepjes”
- Een knop om het alarm uit te zetten



### Doeleind

We maken een keukenwekker die van 0 tot 60 seconde werkt.

- Als je de NXT aanzet, staat de wijzer op 0 (omhoog).
- Je moet de wijzer met de hand verdraaien naar bijvoorbeeld 10, zoals op het plaatje hierboven.
- Je start de wekker door op de knop (bovenop de wekker) te drukken.
- De NXT laat de wijzer elke seconde een stapje terug draaien (hoeveel graden is een stapje?).
- Het zou leuk zijn als je tik en tak hoort terwijl de wijzer terug stapt.
- Als de wijzer weer op nul staat, stopt de wijzer met bewegen en gaat het alarm af.
- Het alarm stop je door op de knop te drukken.
- Natuurlijk kun je de keukenwekker daarna nog een keer gebruiken.

### Programma

Zelf te bepalen

## 27 Ballensorteerder

## Compleet project

### Bouw

Maak een bal sorteer machine.

- Er is een voorraad ballen, in het plaatje hiernaast ligt de te sorteren voorraad op de schuine rails.
- Monteer een kleurensensor die bepaalt of de bal rood of blauw is.
- Bouw een mechanisme dat de rode bal in het linker en de blauwe bal in het rechter vak kan duwen.
- Wij hebben nog een tweede motor gebruikt: die voorkomt (blokkeerpen) dat de ballen naar beneden rollen als de sorteermotor naar links of naar rechts beweegt.



### Doeleind

De ballen rollen tot een aanslag.

Als de sensor geen bal ziet, stopt de machine.

Als de sensor een rode of blauwe bal ziet gaat de blokkeer-pen omhoog (motor 2).

De sorteermotor (motor 1) duwt de bal naar links als hij rood is, en naar rechts als hij blauw is.

De sorteermotor gaat weer in de middenstand.

De blokkeerpen gat weer weg.

Het programma herhaalt zich.

### Programma

Zelf te bepalen

## 28 Colorcoaster

## Compleet project

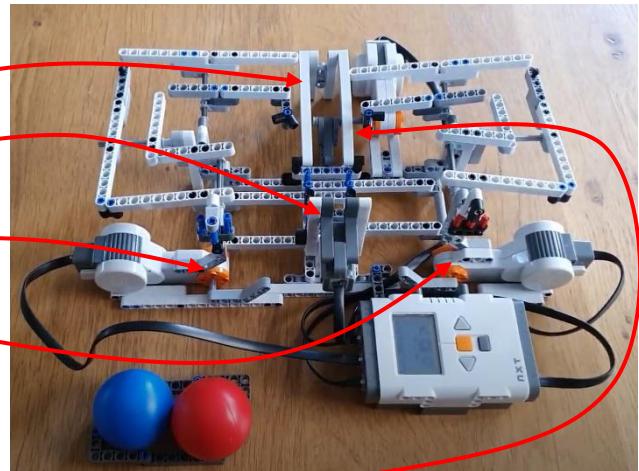
### Bouw

We maken een rollercoaster voor ballen. De ballen worden ook nog gesorteerd op kleur.

In het midden achter zie je een opvangrails die omlaag naar voren loopt.

Midden voor is een kleuren sensor die kijkt of de bal rood of blauw is.

Er zijn twee liften, links voor de blauwe en rechts voor de rode ballen. De liften brengen de ballen omhoog, waarna ze via de (respectievelijk linker en rechter) rails terugrollen naar de opvangrails.

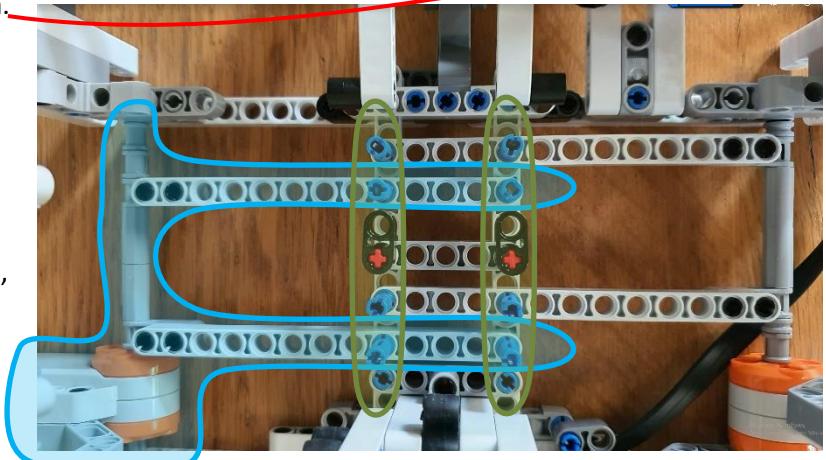


Een derde motor bedient de blokkeerpin.

### Tip 1: liften

De linker en rechter lift zijn hetzelfde. In het plaatje hiernaast is de linker lift blauw omcirkeld.

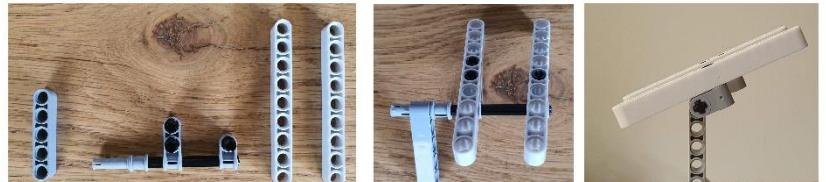
Let op, dat beide liften "vingers" hebben, en dat die vingers samen weer het einde van de opvang rails vormen (groen).



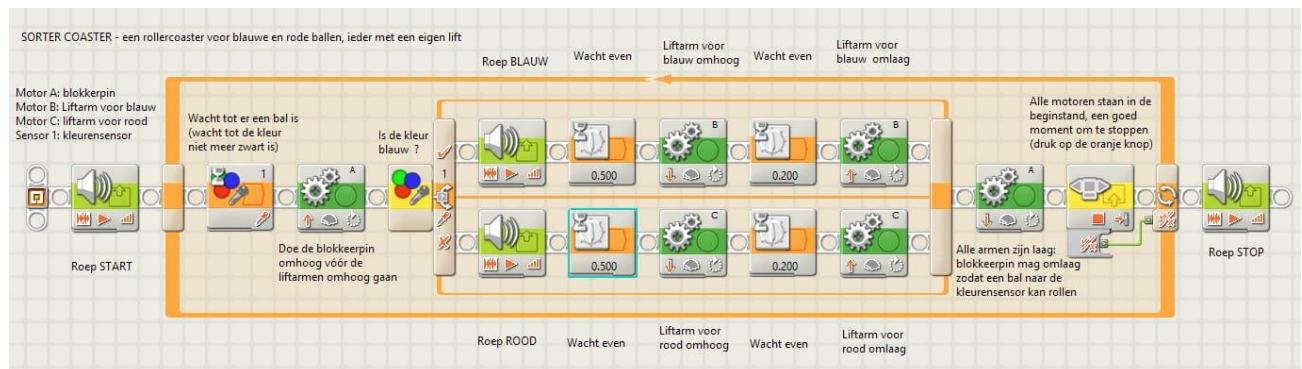
### Tip 2: schuine rails

Een rollercoaster heeft aflopende rails nodig. Helaas is schuin bouwen in lego niet makkelijk.

Hiernaast een hint in drie stapjes:



### Programma



## 29 Line follower

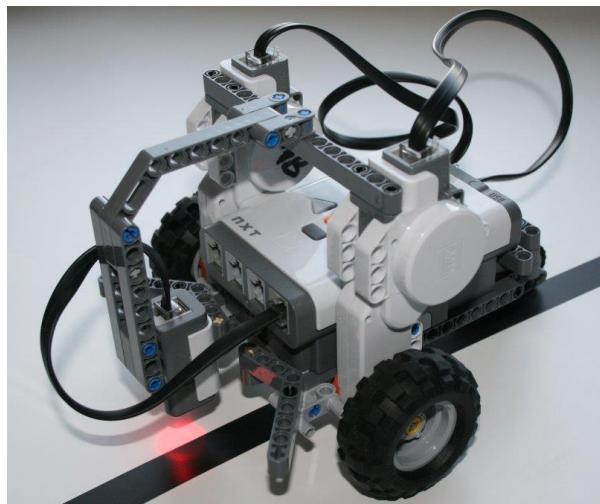
## Compleet project, P(ID)

### Bouw

Bouw een auto die kan draaien; bijvoorbeeld één wiel links met een motor en één wiel rechts met een motor en een zwenkwiel achter.

Zet vooraan, laag bij de grond een lijn sensor: een licht sensor.

Hiernaast zie je een voorbeeld van zo'n auto. Hij staat op de lijn die hij probeert te volgen.



### Doe

Het doel is dat de auto continue rijdt, maar wel de lijn blijft volgen.

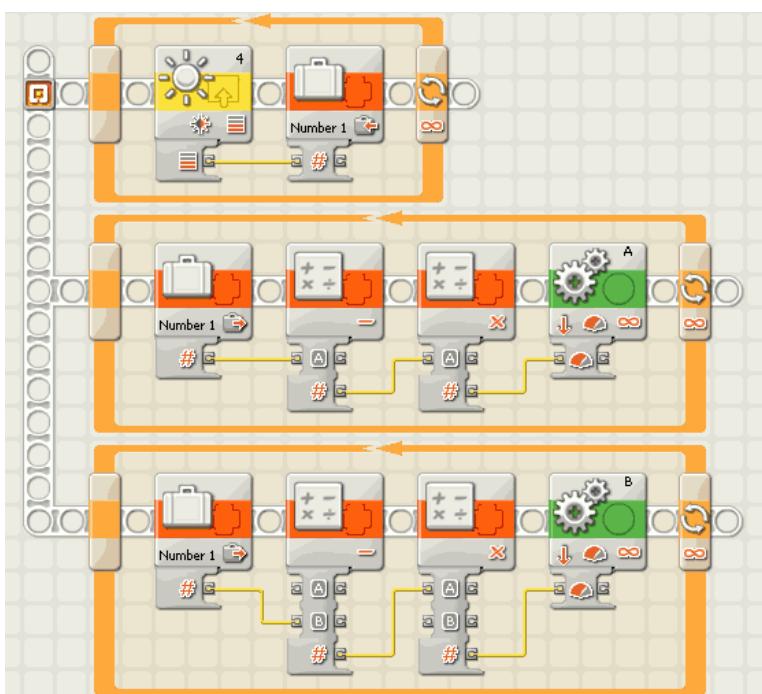
### Programma

Er is een belangrijk principe: de auto volgt niet de zwarte lijn, maar de grens tussen zwarte en wit. Als je goed kijkt zie je dat op de foto: de licht vlek zit half op zwart en half op het wit aan de rechter kant.

Wat doet het programma: als de sensor wit ziet, dan is de auto te veel rechts en dus stuurt hij naar links. Andersom, als de sensor zwart ziet, dan is de auto te veel links en dus stuurt hij naar rechts.

Als je dit helemaal mooi wil doen: hoe witter hoe meer naar links, hoe zwarter hoe meer naar rechts.

Hieronder zie je een taak die de sensor leest en de waarde in de koffer "Number 1" plaatst. Daaronder zien we een taak voor het linker wiel en een taak voor het rechter wiel, beide gebruiken "Number 1".



#### Pro Tip "kalibreren":

De licht-waarde van de sensor is tussen 0 en 100. Maar in de praktijk veel minder. Dat moet je kalibreren. Bij ons: voor zwart 20 en voor wit 60.

De motoren moeten een vermogen krijgen tussen 0 en 100.

$$\text{VermogenA} = (\text{Sensor} - \text{Zwart}) \times \text{Snelheid}$$

$$\text{VermogenB} = (\text{Wit} - \text{Sensor}) \times \text{Snelheid}$$

Voor ons: Zwart=20, Wit=60 en Snelheid=2.5.

Sensor	VermogenA	VermogenB
20	0	100
30	25	75
40	50	50
50	75	25
60	100	0

# 30 Gripper

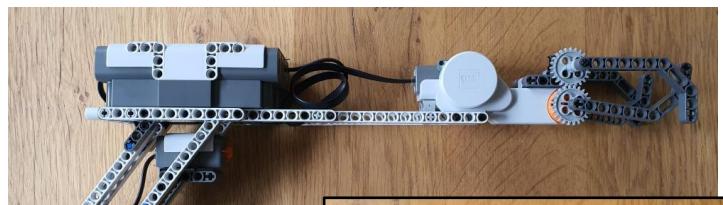
**MyBlock**

## Bouw

Bouw een “hand op afstand”. Dit bestaat uit een motor die de vingers open en dicht kan doen, en een druk knop om de hand te bedienen.

Zorg voor een *aanslag* voor open en dicht.

Zorg voor een stevig handvat.



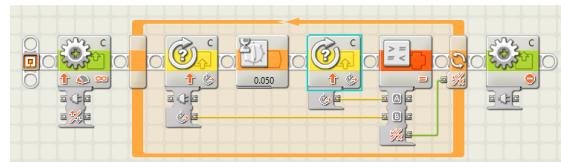
## Doe!

Als je de knop indrukt is, gaat de hand dicht – tot hij niet meer verder kan.

Als je de knop los is, gaat de hand open – tot hij niet meer verder kan.

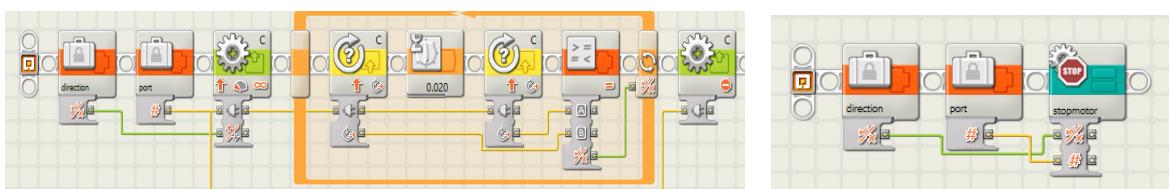
## Programma 1

Als je de NXT aanzet (de knop is dan niet ingedrukt) moet de hand open gaan tot hij niet meer verder kan (tot hij tegen de aanslag aan komt). De truc die we doen is de motor aanzetten, wachten tot de ingebouwde draaisensor geen verschil meer ziet, en dan de motor uitzetten. **Schrijf dat programma eerst.**



## MyBlock

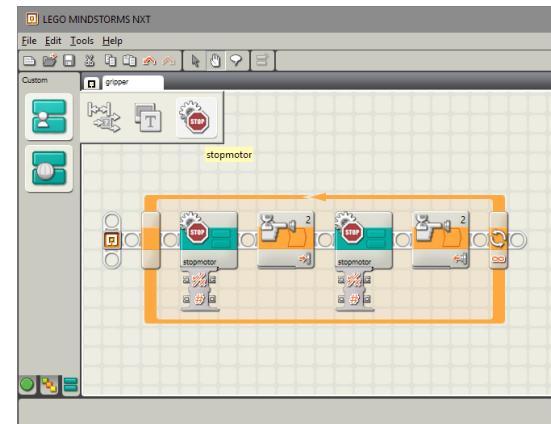
Precies zo'n zelfde stuk hebben we nodig als we knop indrukken, behalve dat de motor de andere kant op moet draaien. De makkelijke weg is dit stuk kopiëren en de motorrichting omdraaien. **Maar wij gaan een eigen blok maken.** Daarvoor moeten we eerst de draairichting (en ook maar de poort) met een koffertje zetten (plaatje links). Daarna ga je in het Bewerk menu en kies je Maak MyBlock. Verzin een naam (bijvoorbeeld “stopmotor”), omschrijving en plaatje voor je MyBlock, en ineens is je programma veel korter (plaatje rechts). Test of het nog steeds werkt. Test of het ook werkt voor hand dicht.



## Programma 2

Als je eenmaal dit MyBlock hebt is er iets bijzonders, dit MyBlock staat nu in je derde (“blauwige”) menu.

Het hele programma voor de gripper is nu kort. Denk er wel aan dat je in het stopmotor blok de poort en richting goed invult.

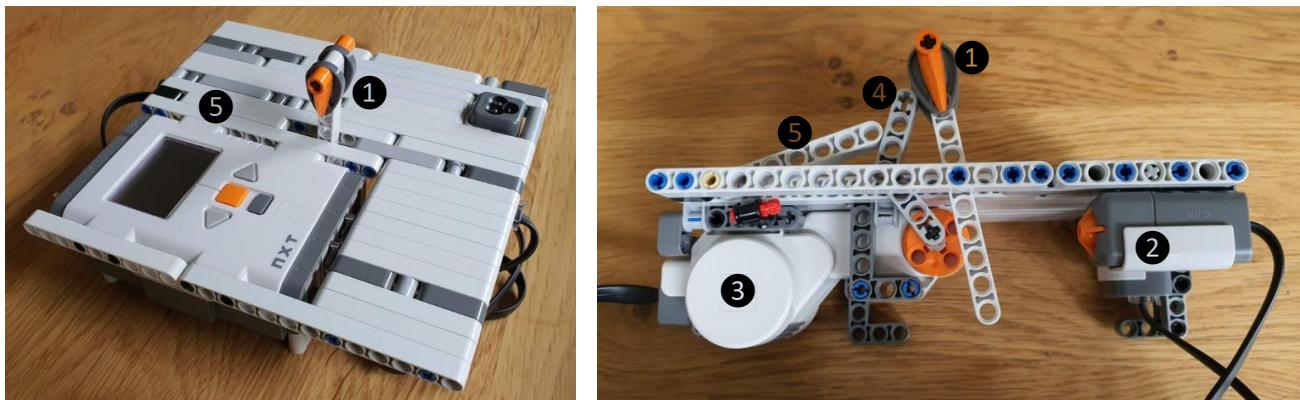


## 31 Don't Push

Timer

### Bouw

We gaan een “useless machine” maken. Deze machine heeft een grote aan/uit knop (de grote oranje handel). Als je de machine aanzet zal hij zichzelf weer uitzetten, door met een armpje de handel terug te duwen.



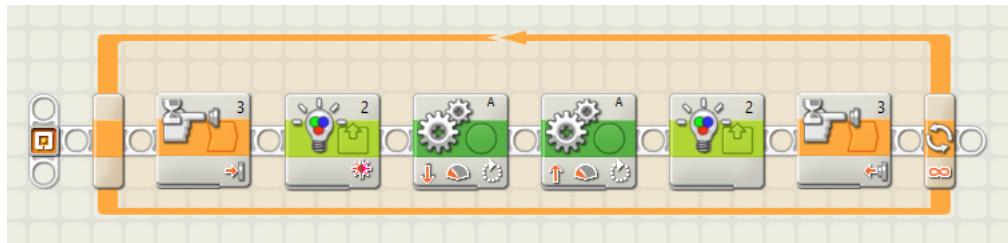
Op de rechter foto zie je de handel (1), die tegen de sensor (2) drukt. De motor (3) heeft een arm (4) die de handel terugduwt. Hij duwt daarbij ook de afdekplaat (5) omhoog.

Naast de motor en de druk sensor, hebben we ook nog een kleuren sensor gebruikt. Die gebruiken we als aan/uit lampje: hij gaat rood branden als de machine “aan staat” en gaat uit als de machine “uit staat”.

### Doe!

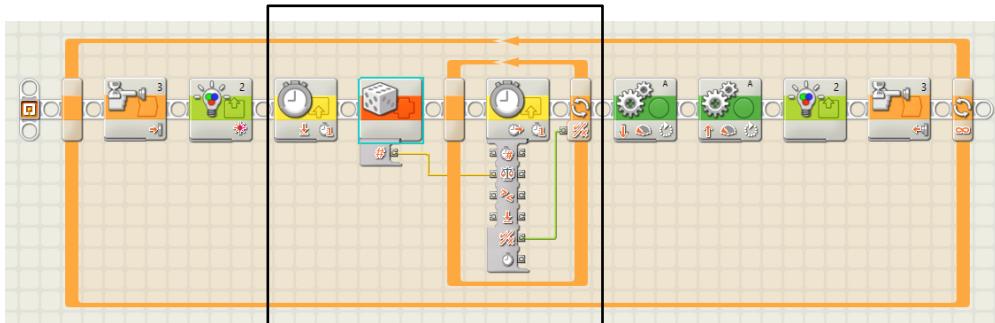
Als je de handel overhaalt, gaat de rode lamp branden. De motor zet de handel weer in de uitstand. De rode lamp gaat uit.

### Programma 1



### Programma 2

Bovenstaande programma zet de machine meteen uit. Wij willen dat dat pas na een tijdje gebeurt. En ook nog eens dat “dat tijdje” soms wat langer duurt en soms wat eerder is. We gebruiken een dobbelsteen om de wachttijd te kiezen, en een timer om zolang te wachten als de dobbelsteen aangeeft.



#### Waarschuwing

De timer werkt met *millisecondes*.

De dobbelsteen moet dus een willekeurig getal tussen 0 en bijvoorbeeld 8000 kiezen. 8000 milliseconden is 8 seconde.

## 32 Rock, paper, scissors

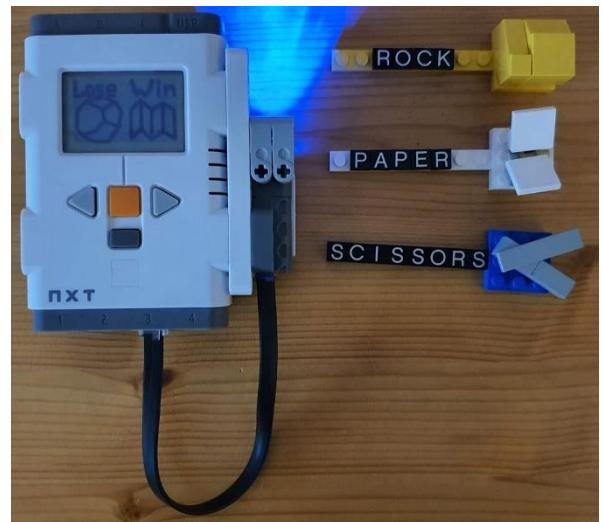
## Softwareproject, eigen tekening

### Bouw

We gaan een robot maken die rock, paper, scissors met je speelt. Je hoeft bijna niks te bouwen. Wij gebruiken een kleurensensor en drie bouwwerkjes voor rock, paper, scissors, van drie verschillende kleuren. Een "aanslag" erbij is wel handig, want als je het bouwwerkje te dicht bij de kleurensensor houdt ziet hij zwart.

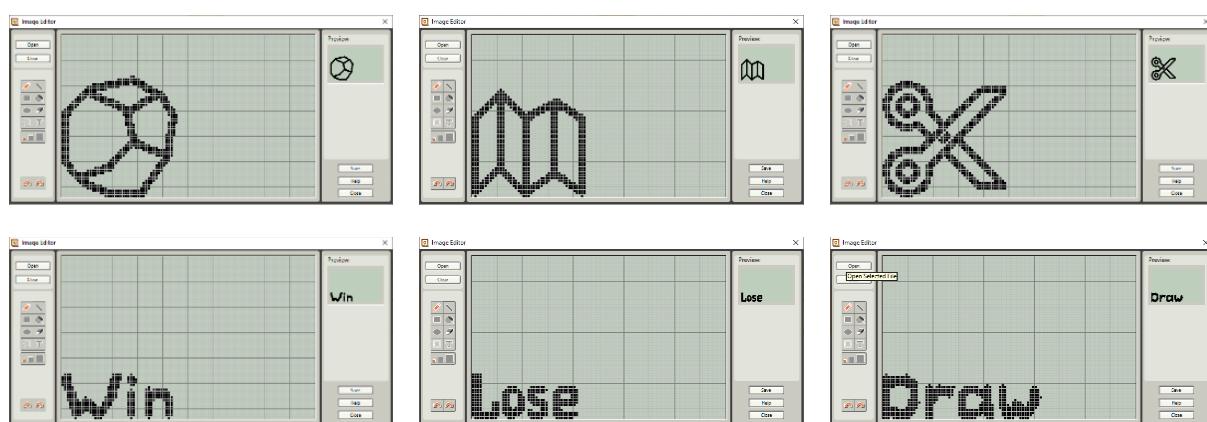
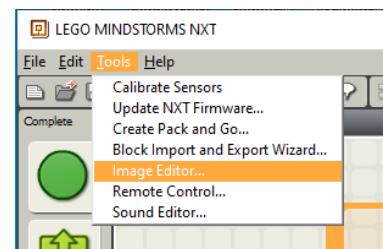
### Doe!

- Door een van drie gekeurde objecten bij de kleuren sensor te houden voert de *user* zijn keuze in.
- Daarna maakt de *computer* een keuze (gebruik het dobbelsteen blok).
- Het programma zal vervolgens met een tekeningetje de keuze van de *user* en van de *computer* op het scherm tonen.
- Ten slotte komt bovenaan het scherm te staan wie gewonnen heeft.



### Tekenen

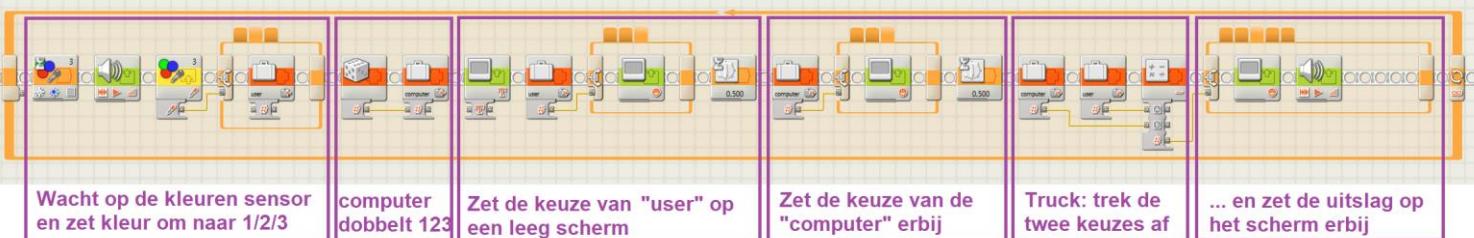
Het is leuk om tekeningen te maken; de Lego app heeft een tekenprogramma. Teken plaatjes, voor rock, paper, en scissors, en voor winst, verlies, en gelijkspel.



**Tip:** Teken altijd links onder

### Programma

Maak twee variabelen aan "user" en "computer" waar de waardes 1, 2, of 3 in staan, voor rock, paper, en scissors. Hieronder staat een schets van een mogelijk programma.



Wij laten een piepjoe horen als de mens een zet doet. De uitslag vieren we met een verdrietig of blij muziekje.

## 33 Reactie tester

## Softwareproject

### Bouw

We gaan een robot maken die je reactietijd meet. Je hoeft bijna niks te bouwen. Er is een drukknop die je moet indrukken zo gauw de lamp aangaat. Wij gebruiken een kleurensensor omdat die een lamp heeft.

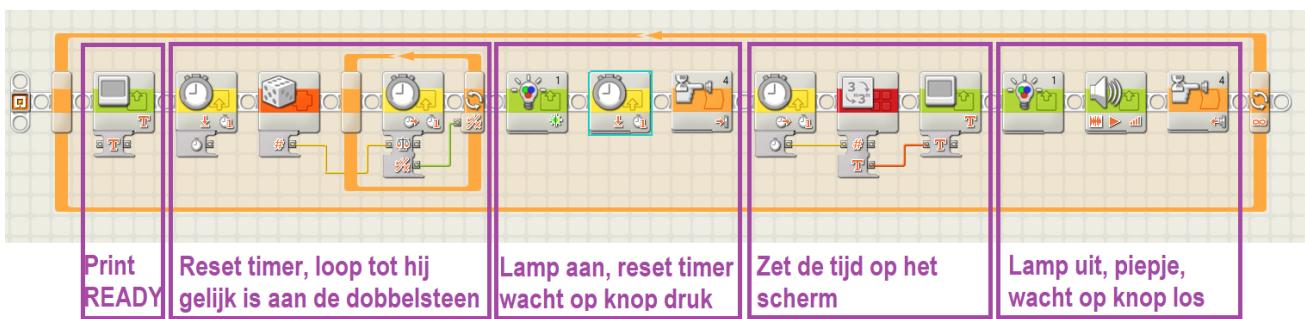


### Doe!

- Aan het begin van het spel staat er READY... op het scherm.
- Na een "random" aantal seconde (tussen 3 en 15) gaat de lamp aan.
- De gebruiker moet op de knop drukken, en we vertellen hoe snel dat ging.

### Programma 1

Hieronder staat een voorbeeld programma met uitleg in paars.



### Programma 2

Er is één ding jammer in programma 1: je kunt vals spelen. Als er READY op het scherm staat kun je de knop al meteen indrukken. In de versie hieronder testen we tijdens het wachten of de knop al ingedrukt is.

We gebruiken een variabele 'Te vroeg' om bij te houden of er te vroeg gedrukt is.



**Pro tip:** Als je de kleuren sensor gebruikt (in plaats van de lichtsensor) kun je extra gemeen doen. De licht sensor heeft maar één lampje, maar de kleuren sensor heeft meerdere kleuren.

Je zou de speler kunnen afleiden met een "verkeerde" kleur. De reactie test gebruikt dan het *groene* lampje. Maar af en toe, in de tijd-wacht-lus wordt het lampje *rood*.

Je mag alleen bij groen drukken. Bij rood ben je af (net als bij te vroeg drukken).

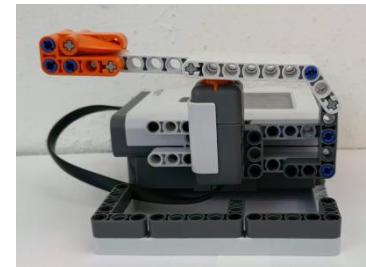
## 34 Morse

## Softwareproject

### Bouw

We gaan met morsecodes werken: piepjess coderen een boodschap.

Verbind een drukknop sensor met de NXT en maak er een mooi grote knop aan vast.



### Morse

De morsecode werkt met piepjess. Druk je de schakelaar kort in dan krijg je een korte piep (•), druk je hem lang in, dan krijg je een lange (–). Combinaties van lang en kort coderen een letter. Bijvoorbeeld R is • – •.

Hieronder zie je een codeertabel (van letter naar code) en daaronder een decodeertabel (code naar letter).

A    •–	E    •	I    ••	M    --	Q    ---•-	U    ••-	Y    -•---
B    -•••	F    •••-	J    •---	N    -•	R    •••	V    •••-	Z    - -••
C    -•••	G    -••	K    -•-	O    ---	S    •••	W    •--	
D    -••	H    ••••	L    •-••	P    •--•	T    -	X    -••-	

1	2	3    • ? ?	3    - ? ?	4    •• ? ?	4    •- ? ?	4    -• ? ?	4    -- ? ?
E    •	I    ••	S    •••	D    -••	H    ••••	L    ••••	B    -•••	Z    - -••
T    -	A    •-	U    ••-	K    -•-	V    •••-	?    •••-	X    -••-	Q    - -•-
	N    -•	R    -••	G    -••	F    ••••	P    ••--•	C    -•••	?    - - -•
	M    --	W    •--	O    ---	?    ••--	J    •---	Y    -•--	?    - - - -

### Programma 1

Schrijf een programma dat een piep laat horen zolang je de knop indrukt (piep stopt als je loslaat).

### Programma 2

Als programma 1, maar als je de knop loslaat staat er op het scherm of de knop kort of lang is ingedrukt. Je gebruikt hiervoor een timer. Lees in de pro tip hoe tijd gebruikt wordt bij morse (de-)codering.

### Programma 3

Als programma 2, maar nu kun je de knop meerdere keren indrukken. Elke keer wordt de hele code getoond. Tip: sla de code op als een getal: 1121 codeert dus F.

### Programma 4

Als programma 3, maar nu komt echt de letter op het scherm. Maak een myblock waar een getal in gaat en een letter uit komt. Natuurlijk kun je een heel woord intypen.

**Pro tip:** de truc bij morse is *tijd*.

Je spreekt een tijdsinterval af bijvoorbeeld 0.1 seconde.

Dat betekent dat een punt 0.1s duurt en dat een streep 0.3s duurt.

De wachttijd tussen twee symbolen is ook weer 0.1s en de wachttijd tussen twee letters is weer 0.3s.

In beide gevallen is 0.2 dus het beslis punt.

### Andere ideeën

Als je twee NXTs hebt zou je *Bluetooth* kunnen gebruiken om écht morsecodes naar de ander te sturen.

Je zou ook een *recorder* kunnen maken. De tijden van de piepjess die je intypt worden in een bestand opgeslagen. Met een ander programma kun je ze afspelen.

## 35 Radar

## Tekenen met coördinaten

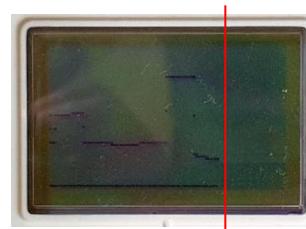
### Bouw

We gaan een radar maken. We gebruiken de ultrasoon sensor om afstand te meten. Deze sensor zit op een as die door de motor langzaam heen en weer gedraaid wordt. De afstanden worden op het scherm van de NXT getekend.



### Doe!

De radar draait 180 graden. Tijdens het draaien wordt onderaan het scherm een voortgangslijn opgebouwd. De lijn loopt op van links naar rechts, en loopt gelijk op met de radar die van 0 naar 180 draait. Op het linker plaatje zie je de lijn op  $\frac{3}{4}$  (rood), op het rechter plaatje is de radar net helemaal rond. Een tweede "lijn" (puntenwolk) laat de gemeten afstand zien.



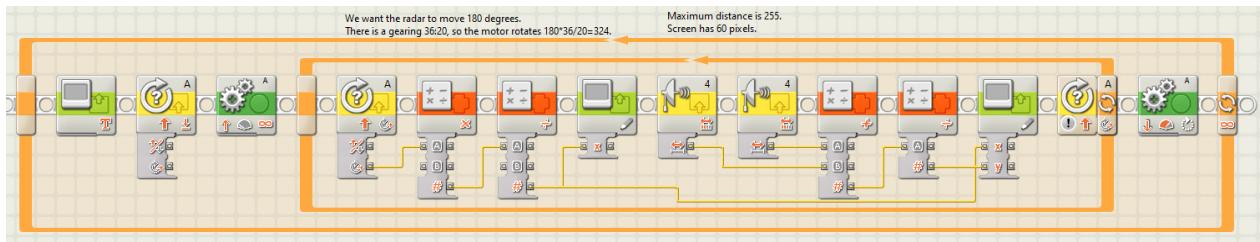
### Programma

Het is een uitdaging om de motor gelijk te laten lopen met de lijn. Het plaatje hieronder laat zien hoe wij dat gedaan hebben. Nadat het scherm gewist is, zetten we de draaihoek sensor in de motor op nul. Daarna zetten we de motor aan (langzaam), en begint de lus die de voortgangslijn tekent.



In de lus meten we steeds de motor draaihoek en rekenen die om naar een getal tussen 0 en 100 – zoveel puntjes zijn er namelijk op het scherm in horizontale (x) richting. We noemen dit de x-coördinaat, en tekenen een puntje op  $(x,0)$ . De lus stopt als de draaihoek van de radar 180 is. Vanwege de tandwielen heeft de motor dan meer gedraaid (bij ons  $36/20 \times$  zover).

Als je de voortgangslijn voor elkaar hebt gaan we de afstandsmetingen toevoegen. We doen steeds twee metingen met de ultrasoon sensor om meetfouten uit te middelen. We tellen die op en delen dat zodat er een getal tussen 0 en 60 uit komt – zoveel puntjes zijn er namelijk op het scherm in verticale (y) richting. We noemen dit de y-coördinaat, en tekenen een puntje op  $(x,y)$ .



In plaats van de motor te stoppen draaien we de radar weer terug naar nul positie, en hebben we een lus om het geheel gezet, zodat de radar heen-en-weer blijft draaien, en steeds een nieuw afstandsplatje tekent.

---

# Ideeën

---

## Projecten

- Programmeerbare auto (input voor aantal-rotaties; input voor speed)
- Muziekmachine: speel deuntje dat te programmeert met een kleuren strip
- Rijd in een cirkel en zeg home na een vol rondje, daarna een achtje rijden
- Record en playback dansen; 2 motoren met armen en een hoofd?
- Record en playback: piano met vier knoppen
- Muntgeld teller (spaarpot)
- Raad volgende nummer (1..10) is hoger of lager
- "Etch-a-Sketch" op NXT scherm  
<http://www.nxtprograms.com/scribble/index.html>
- "Simon says" met 4 knoppen en de kleuren sensor in het midden als 4 kleuren led
- Gebruik een tweede NXT om de eerste (een auto) met bluetooth te besturen
- Flipperkast
- Caver game: rupsband (papieren band) met obstakels, lichtsensor ziet ze, knoppen schuift sensor  
<https://m.facebook.com/groups/SPIKEcommunity/permalink/1432303443814550>
- Bar code scanner [http://www.nxtprograms.com/NXT2/checkout\\_scanner/index.html](http://www.nxtprograms.com/NXT2/checkout_scanner/index.html)
- Area calculator [http://www.nxtprograms.com/volume\\_calc/index.html](http://www.nxtprograms.com/volume_calc/index.html)
- Whack a brick (SW only) <http://www.nxtprograms.com/whack-a-brick/index.html>