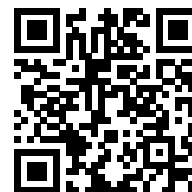




Workshop programmeren – Micro:bit

Werkblad Micro:bit

Werkblad bij stap 1



Bedenk twee vragen die je aan de mensen van Alius wilt stellen als je op bezoek gaat.

- Vraag 1

- Vraag 2

Ken je iemand die hier werkt, of een van de beroepen heeft die je in de video ziet?

Zou je er zelf willen werken? Waarom?

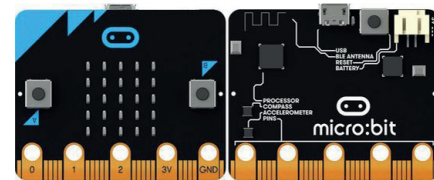


Workshop programmeren – Micro:bit

Werkblad Micro:bit

Werkblad bij stap 8

Programmeren met de micro:bit



- P1.** Pak de laptop en ga naar de website <https://makecode.microbit.org/> en klik op de knop 'nieuw project'. Je mag zelf een naam kiezen voor je project.
- P2.** Links op de website zie je een digitale micro:bit. Deze reageert net als een 'echte' micro:bit. Rechts naast de digitale micro:bit zie je knoppen in verschillende kleuren zoals 'basis' en 'invoer'. Daarmee geef je je micro:bit opdrachten. Als je op zo'n knop drukt, krijg je het menu met allerlei instructies die erbij horen, zoals 'toon lichtjes' of 'toon nummer'.
- P3.** Aan de rechterkant van het scherm staan twee blokken: 'bij opstarten' en 'de hele tijd'. Alles dat je in het blokje 'bij opstarten' sleept, gebeurt als je de micro:bit aanzet. Wat bij 'de hele tijd' staat wordt steeds opnieuw achter elkaar door uitgevoerd. Dit heet een **'loop'**.

Servomotor programmeren

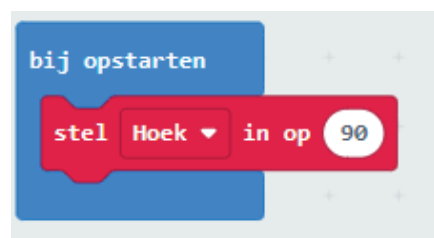
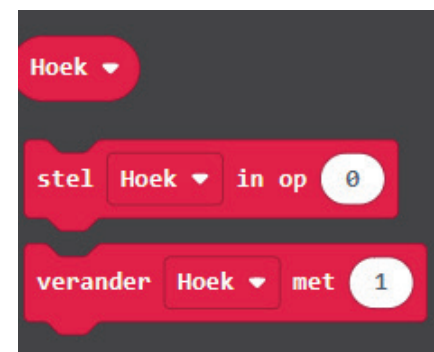
- P4.** Wat is nu eigenlijk een servomotor? Dit is een soort slimme motor, deze motor weet hoe ver hij is gedraaid. Daardoor kan je tegen de motor zeggen: 'Draai naar 90 graden'. In onze code voor de micro:bit kunnen we dus zulke opdrachten invoeren. Een servomotor wordt ook wel eens een stappenmotor genoemd.
- P5.** Een variabele gebruiken programmeurs om getallen, tekst of andere dingen op te slaan en te gebruiken. Deze variabele kan je op een ander moment weer ophalen om precies hetzelfde te gebruiken of je kan er iets nieuws in opslaan.
- P6.** We gaan de variabele 'hoek' invoeren. Klik in de lijst op 'variabelen' en klik op 'maak een variabele...'. Daar type je "hoek" in en klik je op OK.



In de lijst zijn drie blokken bijgekomen:

- 'Hoek' met afgeronde hoeken
- 'Stel [hoek] in op 0'
- 'Verander [hoek] met 1'.

Kies de eerste variabele 'Hoek'. Sleep het blok 'Stel [hoek] in op 0' in het blok 'bij opstarten' en verander het getal 0 naar 90.

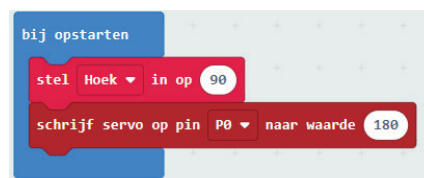




Workshop programmeren – Micro:bit

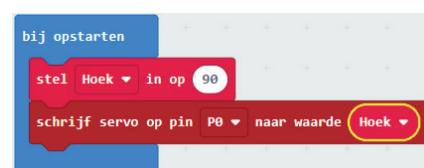
Werkblad Micro:bit

- P7.** Klik op 'geavanceerd' in het menu daarna 'pinnen'. Je ziet nu een aantal opties. Wij gebruiken 'schrijf servo op pin [p0] naar waarde 180'. Sleep dit blok in het blok 'bij opstarten' onder het andere blokje dat daarin zit.



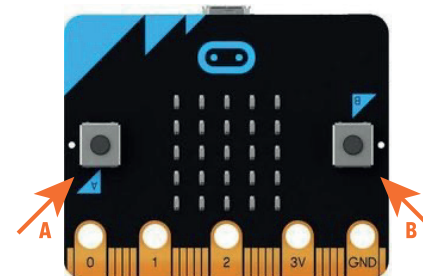
Als we de code nu op de micro:bit zouden zetten, draait de servomotor 180 graden rond.

- P8.** Wij willen dat de servo naar een hoek van 90 graden draait. Hiervoor gaan we onze variabele gebruiken. Klik in het menu op de knop 'variabelen' en sleep het kleine blokje 'hoek' op de plek waar nu 180 staat.



Wat er nu gebeurt is dat de micro:bit de variabele 'Hoek' bekijkt en leest wat hierin staat. Hierin staat 90 want dat hebben we in het blokje daarboven gezegd. Als we nu de code op de micro:bit zetten, dan draait de servo naar een hoek van 90 graden, precies zoals wij willen.

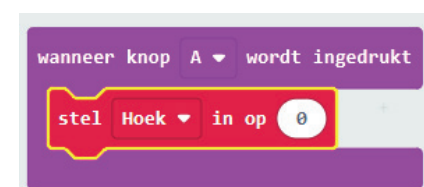
De servomotor is nu helemaal klaar om te bedienen, maar nu gaan we leren hoe je dat doet. Als je goed kijkt, zie je op de micro:bit twee zwarte knopjes zitten (zie de pijlen). Met deze knopjes kan je de servomotor naar links en rechts laten draaien.



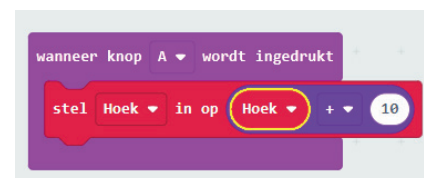
- P9.** Klik op de knop 'Invoer' in het menu. Bovenaan in dit menu staat het blok met de tekst 'wanneer knop [A] wordt ingedrukt'. Sleep dit blok naar een lege plek in het scherm waar ook 'bij opstarten' en 'de hele tijd staat'. We gaan straks aan de slag met knop 'B'.

- P10.** We hebben nu ingesteld dat er een blok wordt geactiveerd door op de knop 'A' te drukken. Nu gaan we de opdracht geven die dan moet worden uitgevoerd. Om de servomotor naar een nieuwe hoek te laten draaien, gaan we twee dingen doen. |

1. We gaan de variabele 'hoek' verhogen met tien.
 2. Daarna vertellen we de servomotor dat die naar deze nieuwe hoek moet draaien.
- Sleep uit het menu 'Variabelen' het blokje 'stel [hoek] in op 0' naar het blokje 'wanneer knop [A] wordt ingedrukt'.



- P11.** We willen de 'hoek' niet op 0 instellen, maar met 10 verhogen. Dat doen we met het rekenblokje. Sleep uit het menu 'Rekenen' het blokje '0 [+] 0' naar de plek in het blokje 'wanneer knop [A] wordt ingedrukt'.



Pak uit de lijst variabelen [hoek] en zet deze op de eerste '0' en zet op de plek waar nu de 0 staat een 10.

Nu wordt steeds als je het knopje A indrukt op de micro:bit de variabele hoek met 10 opgehoogd.



Workshop programmeren – Micro:bit

Werkblad Micro:bit

Tussenstand 1

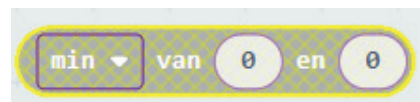
We hebben ingesteld dat de servo motor start op 90, dus het wordt:

Als je op 'A' klikt gaat de hoek naar $90 + 10 = 100$, klik je nogmaals op 'A' dan is Hoek 110, enzovoorts...

Dit gaat goed totdat we bij 180 graden uitkomen. De servomotor die wij gebruiken kan namelijk maar van 0 graden tot 180 graden draaien. Dus als we nu nog een keer zouden klikken, wordt 'hoek' 190 en daar kan onze motor niks mee.

P12. Gelukkig is er een blokje dat ons kan helpen. Dit blokje staat in het menu 'Rekenen' en het blokje heet '[min] van 0 en 0'. Dit blokje geeft de kleinste waarde van de twee waardes die erin staan terug.

Bijvoorbeeld: Als er op de eerste plek 40 staat en op de tweede plek 80, dan geeft dit blokje 40 terug. Want 40 is kleiner dan 80.



Sleep dit blokje op een lege plek in het venster met de rest van de blokjes. Het blokje heeft nu een grijzige kleur gekregen. De micro:bit-site wil hiermee aangeven dat het blokje op deze plek niets doet. Dat gaan we aanpassen.

Vul op de plek van de eerste 0 180 in. Het blokje zal dus nooit meer dan 180 teruggeven, omdat het blokje altijd zal kiezen voor het kleinste getal.



P13. Enkele stappen terug heb je het blokje 'hoek [+] 10' gemaakt. Sleep dit blokje op de plek van de 0.

 **Tip:** het helpt als je je muis op het paarse gedeelte houdt wanneer je klikt.

Dit hele blok geeft de kleinste waarde van de 2 terug. Dat is dus of 180 of wat de variabele 'hoek' is, plus 10. Sleep dit blokje op de plek waar je net het blokje 'hoek [+] 10' vandaan hebt gehaald. Op deze plek staat nu een 0.





Workshop programmeren – Micro:bit

Werkblad Micro:bit

Tussenstand 2

Wanneer er op het knopje A wordt gedrukt, worden alle opdrachten die daarin staan uitgevoerd. We hebben 'stel [hoek] in op ...', die de waarde van 'hoek' zal veranderen. De waarde die hoek krijgt, wordt bepaald door het blokje '[min] van 180 en hoek + 10'. Oftewel 180 of als 'hoek' + 10 kleiner is dan 180, dan wordt het dat getal.

P14. Het enige dat je nu nog moet doen om de servomotor te laten bewegen, is een blokcode gebruiken dat we al eerder hebben gebruikt: namelijk 'schrijf servo op pin [p0] naar waarde hoek'. Probeer deze nu zelf te maken.

 **Tip:** je kan een blokje code kopiëren (dupliceren) door er met je rechter muis op te klikken en te kiezen voor 'dupliceren'.

Dit willen we bereiken:



P15. Nu ga je hetzelfde doen voor knop B maar dan zo dat de servomotor de andere kant op draait. Om het onszelf makkelijk te maken gaan we niet elk blokje opnieuw in het veld slepen, maar we gaan het blokje van knop A dupliceren (kopiëren) en aanpassen.

Om een blokje te dupliceren klik je met je rechtermuisknop op een blokje en selecteer je uit het menu 'dupliceren'.



P16. De website van micro:bit geeft opnieuw aan dat er nu niks gedaan wordt met deze code (de grijzige kleur). Dat komt omdat we nu twee blokjes hebben die reageren op het drukken op knop A. In plaats van knop A selecteer je nu knop B in het blokje 'wanneer knop A wordt ingedrukt'. Het blok krijgt hierdoor weer een normale kleur.

Als je nu de code op de micro:bit zou zetten en op knop B klikt, draait de motor dezelfde kant op als bij knop A. Dat komt omdat alles wat in blokje 'wanneer knop B wordt ingedrukt', nog steeds hetzelfde is als bij blokje A. We willen nog steeds dat de variabele 'hoek' wordt aangepast. Dat laten we dus staan, maar in plaats van min klikken we op max. Nu staat er 'max van 180 en hoek + 10'.





Workshop programmeren – Micro:bit

Werkblad Micro:bit

P17. De code geeft het grootste getal van de 2 terug. Dus als het eerste getal 180 is en het tweede getal 50, dan kiest de code voor 180. Je hebt misschien onthouden dat je servomotor van 0 naar 180 graden kan draaien. In plaats van 180 voeren we dus 0 in op de eerste plek.

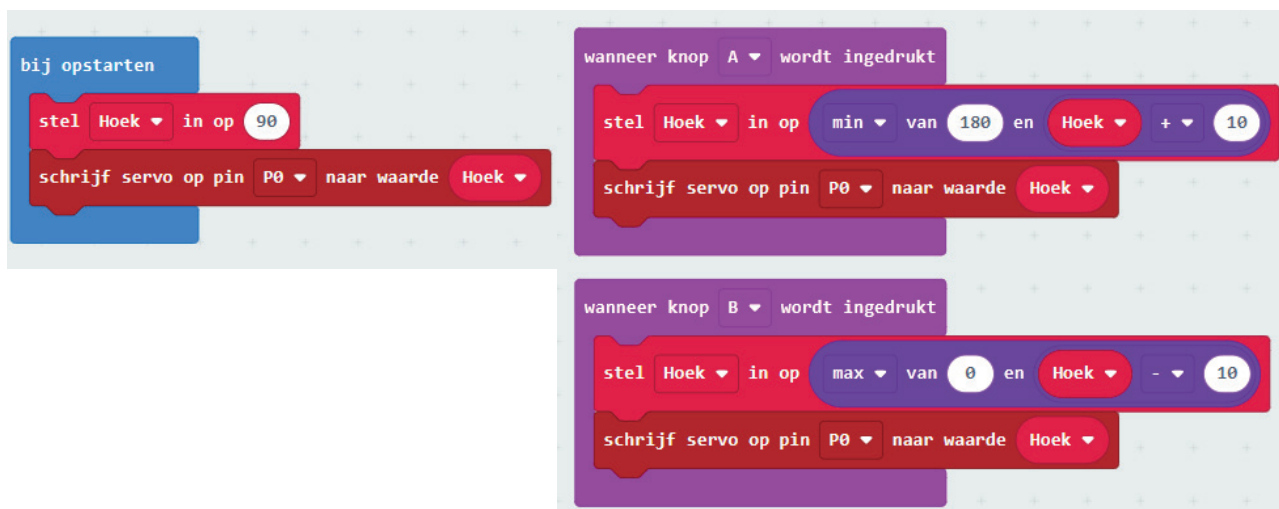
Als je nu nog het plusje verandert naar een minnetje wordt er 10 afgehaald van 'hoek', in plaats van opgeteld.



Tussenstand 3

Kijk eens naar de digitale micro:bit links in je scherm. Klik op de knoppen A en B op de micro:bit. Als je alles goed hebt gedaan, beweegt de servomotor naar links en naar rechts als je op de knopjes A en B klikt. Ze blijven net zo lang bewegen totdat 'hoek' bij 0 of 180 komt, de servomotor gaat dan niet meer verder. Zo willen we het hebben!

Overzicht code tot nu toe:





Workshop programmeren – Micro:bit

Werkblad Micro:bit

Zonnecel programmeren

P18. Nu gaan we aan de slag met de zonnecel. Deze heb je al vastgemaakt aan de servomotor. We gaan via de micro:bit de zonnecel laten weten hoeveel licht er op de cel valt. We hebben de zonnecel verbonden met pin '1', dus via die pin lees je uit hoeveel licht er op de cel valt.

Met het blok 'de hele tijd' ga je ervoor zorgen dat de micro:bit veranderingen op de zonnecel direct ziet. Mocht dit blokje niet op het beginscherm staan kun je deze vinden bij het menu 'basis'.



Alles dat in dit blok staat, wordt achter elkaar doorgedaan. Het begint dus steeds opnieuw. Dit heet een **loop**. Denk maar aan een cirkel!

P19. Afhankelijk van hoeveel licht er op de zonnecel valt, gaan we meer of minder lichtjes laten branden. We moeten dus het programma de keuze laten maken. Dit doen we door middel van een 'als <waar> dan' blokje. Wat er tussen de haakjes staat noemen we een **conditie**, dit betekent: wanneer iets wel of niet gedaan moet worden.

In het menu 'Logisch' hebben we 2 opties: 1 blokje met alleen 'als <waar> dan' en een blokje met 'als <waar> dan anders'. Dit tweede blokje voert alles wat in 'anders' staat uit wanneer het gegeven dat tussen de eerste haakjes staat onwaar is. Dit blokje kiezen we en zetten dit in 'de hele tijd'.



Je ziet ook een plusje staan in dat blokje. Als je hierop klikt verschijnt er een nieuw gedeelte met 'anders als <> dan'. Hier kan je weer een nieuwe conditie invullen. Klik net zo lang op het plusje totdat je vier plekken hebt waar je blokjes in kan stoppen. Hiermee gaan we het licht dat op de zonnecel valt in 4 stappen invoeren:

1. Geen licht
2. Weinig licht
3. Normaal licht
4. Veel licht

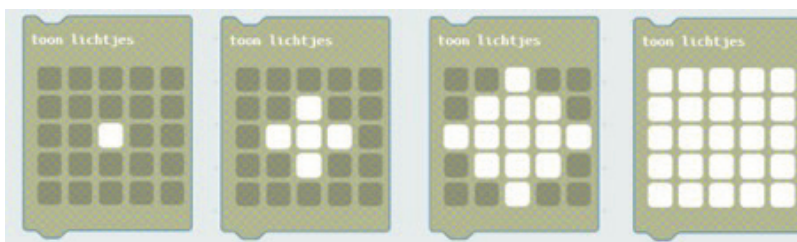
Dat doen we door de lichtjes op de micro:bit aan of uit te zetten.



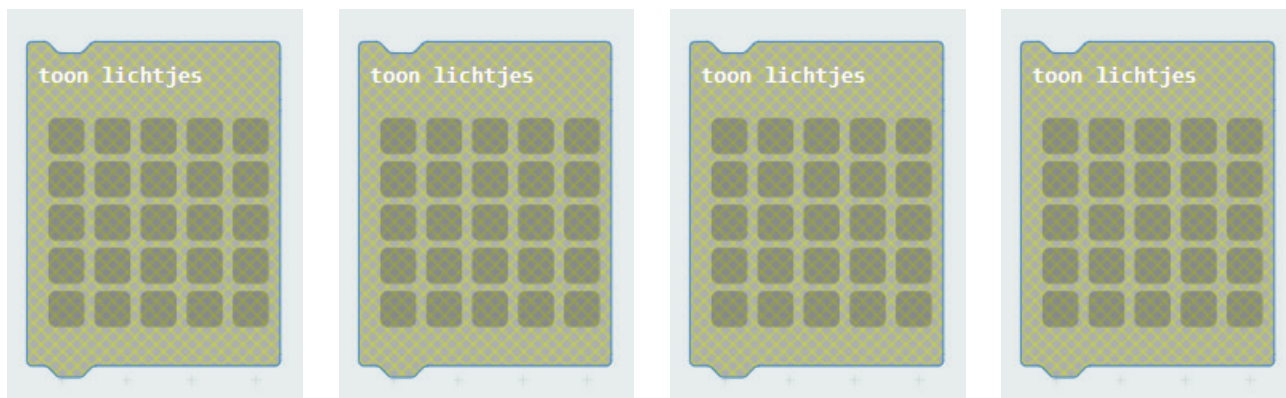
Workshop programmeren – Micro:bit

Werkblad Micro:bit

P20. Het aan- en uitzetten van lichtjes op de micro:bit doe je met het blokje 'toon lichtjes' in het menu 'Basis'. Sleep vier van deze blokjes in het scherm en maak voor elke hoeveelheid licht een patroontje, dit mag je helemaal zelf bepalen. Hieronder zie je een voorbeeld:



Trek dit voorbeeld hieronder over en teken het patroon welke je hebt bedacht zodat je weet welk patroon bij welke hoeveelheid licht hoort.



Geen licht

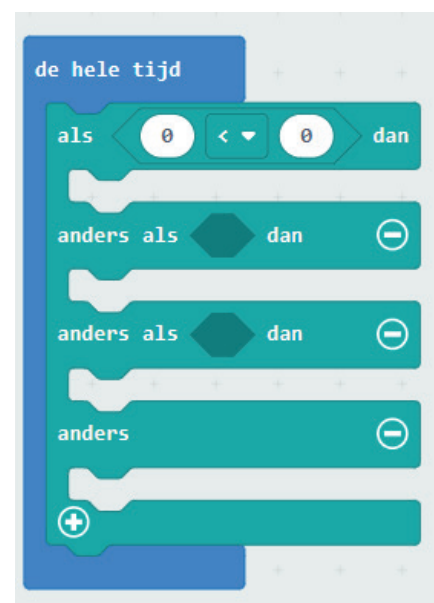
Weinig licht

Normaal licht

Veel licht

P21. In het menu 'Logisch' staat het blokje ' $0 < 0$ ', sleep deze op de plek waar nu 'waar' staat. Dit blokje vergelijkt wat er in het eerste vakje staat met wat er in het tweede vakje staat.

Het blokje kijkt of 0 kleiner is dan 0. Dat is natuurlijk niet zo, dus het blokje geeft onwaar terug.





Workshop programmeren – Micro:bit

Werkblad Micro:bit

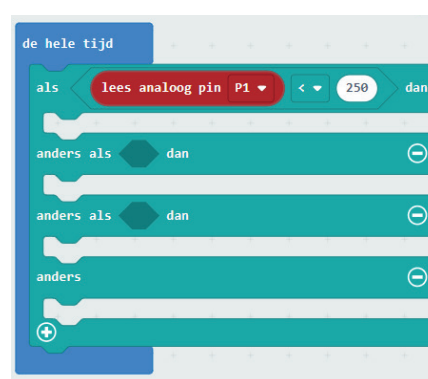
P22. Wij willen iets gaan vergelijken met hoeveel licht er op de zonnecel valt. Om dat te doen hebben we nog een blokje code nodig. Dit staat in het menu 'pinnen' onder 'geavanceerd'. Uit dit menu sleep je 'lees analoog pin [p0]' naar de plek van de eerste 0.

 **Tip:** let op dat je niet per ongeluk het blokje 'lees digitaal pin [p0]' pakt, dit werkt niet.

Verander [p0] naar [p1.] Hiermee zeg je tegen de code: kijk naar pin '1'. Hier hebben we namelijk de zonnecel aangesloten.

Het blokje dat we zojuist hebben toegevoegd, geeft aan hoeveel licht er op onze zonnecel valt. Dit doet het met een getal tussen 1 en 1023, waarbij 1 geen licht is en 1023 heel veel licht.


Wat we nu willen doen is het licht vergelijken met een getal dat we zelf kiezen (tussen de 1 en 1023). 250 is een goed getal om te meten of er weinig licht is. Vul het getal 250 in op de plek waar nu nog 0 staat.

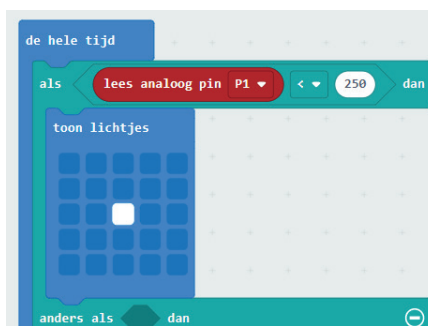


P23. Sleep het blok met 'toon lichtjes', welke je wilt gebruiken voor weinig licht, onder de zojuist gemaakte regel. Als we de code nu op de micro:bit zetten dan gaat er één lichtje branden als er weinig licht op de zonnecel valt.

Om te zorgen dat er meer lichtjes gaan branden naarmate er meer licht op de micro:bit valt, gaan we de rest van de 'anders als <> dan' vakjes ook invullen met precies dezelfde blokjes (met een ander patroon) als in het 'als <lees analoog pin[p1] < 250 dan'.

We gaan alleen de getallen veranderen naar 600 en 900.

 **Tip:** maak weer gebruik van 'dupliceren'. Dit lukt als je met je rechtermuisknop op het hele blok <lees analoog pin [p1] < 250> gaat staan.





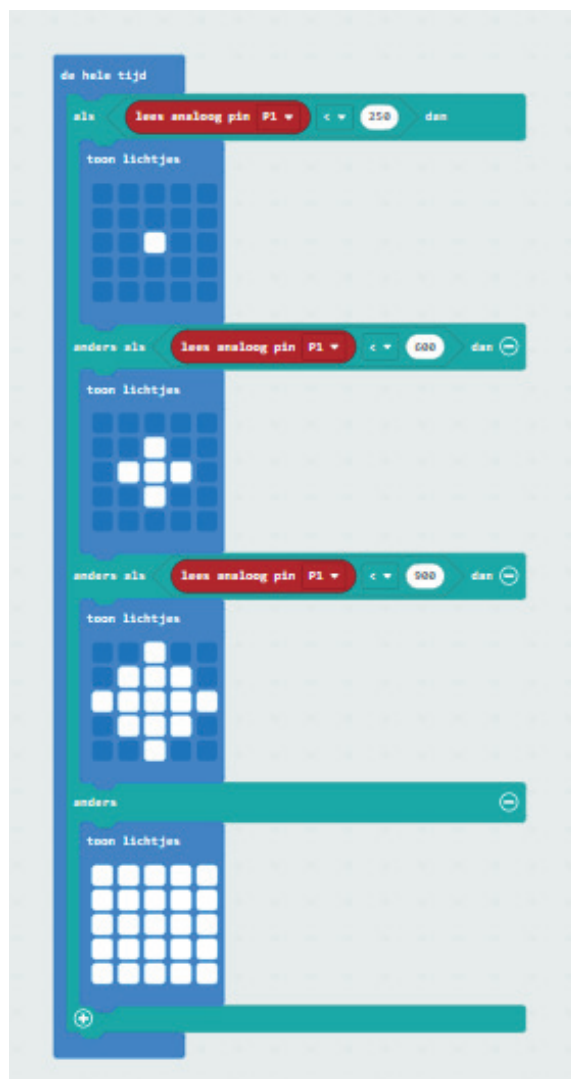
Workshop programmeren – Micro:bit

Werkblad Micro:bit

P24. We gaan de 'toon lichtjes' blokjes op de juiste plek slepen. Het blokje voor weinig licht komt in het eerste lege stuk. Het blokje voor normaal licht in het tweede lege stuk en het blokje voor veel licht komt in het lege stuk onder 'anders'.

Dit 'anders' stuk wordt uitgevoerd als geen van de blokjes die erboven staan worden uitgevoerd, en met hoe wij het hebben ingevuld kan dat alleen nog zijn als er veel licht valt op de zonnecel.

P25. Klik onderaan de pagina op de grote knop 'Downloaden'. Je krijgt nu een bestand met de naam van je project + .hex op het einde, sla dit bestand ergens op waar je er zo bij kan en geef het je eigen naam.





Workshop programmeren – Micro:bit

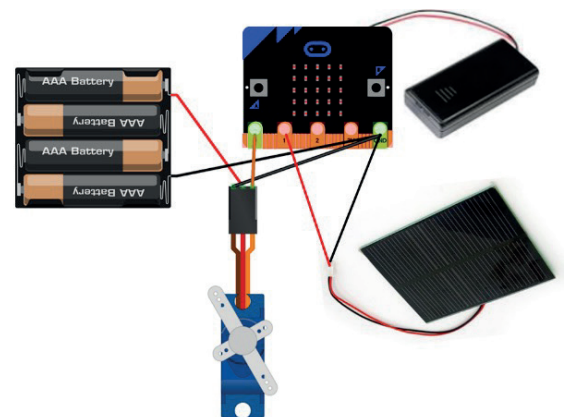
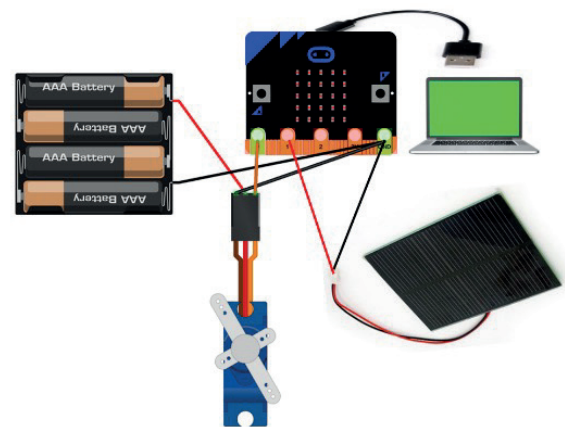
Werkblad Micro:bit

Micro:bit aansluiten

P26. Zorg dat de kleine batterijhouder niet is aangesloten op de micro:bit en sluit de micro:bit met de kabel aan op de laptop.

Sleep het bestand dat je net gedownload hebt in de folder van de micro:bit. Schrik niet, er gaan wat lampjes knipperen op de micro:bit. Dit betekent dat de micro:bit de code aan het lezen is en daarna weer opstart.

Wanneer er geen lampjes meer knipperen maak je de micro:bit los van de computer en sluit je de **kleine batterijhouder** aan.



Nu geven, als het goed is gelukt, de lichtjes aan hoeveel licht er op de zonnecel valt en kun je de servomotor naar links en naar rechts draaien met de knopjes A en B!

P27: Zorg dat je alle onderdelen netjes opbergt en de batterijen uit de houders haalt. Deze lever je los in bij je docent, zodat hij of zij kan zorgen dat ze weer opgeladen kunnen worden. Haal alles voorzichtig uit elkaar. Je school moet de schade vergoeden als onderdelen kapot zijn of ontbreken zonder dat dit is gemeld. Geef het dus door als iets ontbreekt of kapot is gegaan.

Sluit met je klas af en test de zonnepanelen eventueel eens buiten! Maak een foto van alle werkstukken samen en stuur deze op naar info@hetkop.nl!



Workshop programmeren – Micro:bit
Werkblad Micro:bit

Mede mogelijk gemaakt door:

