

WEEKENDSCHOOL – PROGRAMMEREN –

LES 2B – BEGELEIDERS INSTRUCTIE

1. INTRODUCTIE

Dit is de beschrijving van de extra opgave (2B) van de tweede les programmeren. Het geeft achtergrond informatie, maar ook de uitgewerkte stappen.

2. LES 2B: DE UITWERKINGEN VAN DE OPDRACHTEN

2.1 JITTER VAN DE JOYSTICK

Bij de joystick moeten we analoge waarden meten. Het blijkt dat in sommige opstellingen de servo's, als ze in de uiterste positie staan mechanisch geblokkeerd worden terwijl het motortje probeert verder te draaien. Dat hoor je ook omdat de servo dan wat zoemt.

De betekenis is dat de servo stroom neemt en kennelijk een variabele stroom, want je ziet de gemeten waarden aan de uitgang van de joystick dan wat op en neer dansen. We noemen dat *jitter*. Dat komt omdat door de variabele stroom die de servo neemt de spanning wordt beïnvloedt. Weliswaar zit er een stabilisatie op het rode bordje (als de 9V voeding is aangesloten) of op de Arduino (voor het geval de USB voeding wordt gebruikt), maar kennelijk is er toch nog een effect.

Dat betekent dat de joystick ook andere waarden zal geven als je hem niet aanraakt.

Ze kunnen die jitter verminderen (maar niet helemaal kwijtraken) door de servo's in een andere positie te sturen (b.v. 90 graden). Ze zouden ook de servo's kunnen afkoppelen, maar ik wil eigenlijk niet stimuleren dat ze aan draden gaan trekken.

2.2 JOYSTICK BEPAALT POSITIE OF BEWEGING?

Je kunt de positie van de joystick direct de positie laten bepalen of de snelheid van de beweging.

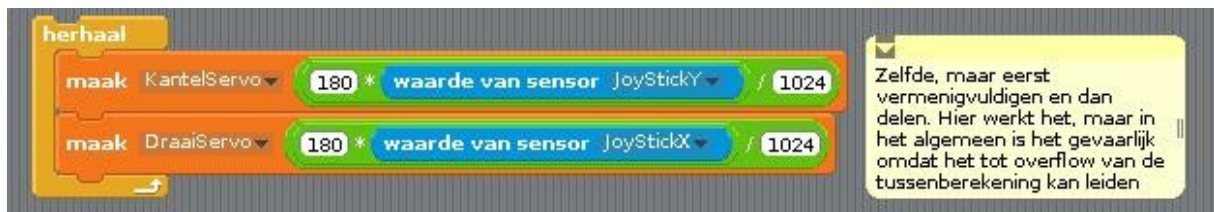
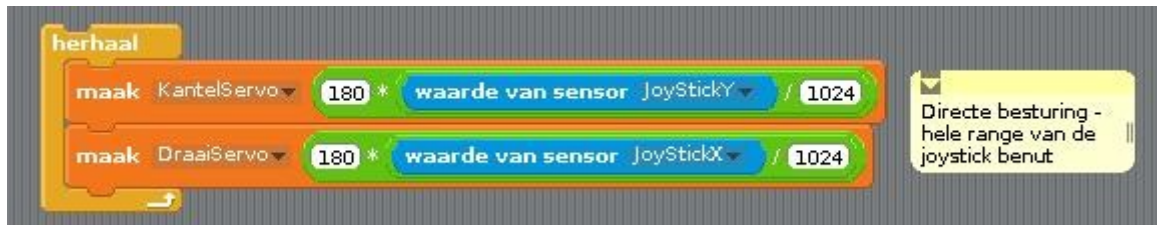
2.3 JOYSTICK BEPAALT POSITIE

OPDRACHT 3: Maak wat hierboven staat en test het uit. **Laat het zien aan je begeleider.**



Omdat de joystick een bereik heeft van 0 tot 1024 en de servo slechts van 0 tot 180 werkt heeft de joystick slechts op in een beperkt gebied invloed.

OPDRACHT 4: Pas het programma aan op deze manier. Doe het voor de DraaiServo en voor de KantelServo. Kijk wat het resultaat is en **laat het zien aan je begeleider**.



De ervaring leert dat je beter eerst kunt delen en dan pas vermenigvuldigen om overflow te voorkomen.

OPDRACHT 5: Pas het programma aan zodat de richting van de bewegingen logisch zijn.



2.4 JOYSTICK BEPAALT BEWEGING

Alternatief is dat ze de joystick waarden gebruiken om bij de X en Y coördinaten op te tellen. Dan moeten ze een bij de variabelen voor de X en de Y positie waarde optellen die afgeleid is van de joystick waarde. Als midden 512 is (theoretisch zou dat moeten, maar door de mechanische uitvoering van de joystick zou het ook ergens anders kunnen liggen), dan zou dit bijvoorbeeld zo kunnen

$$\text{DraaiServo} = \text{DraaiServo} + 0,01 * (\text{JoystickX} - 512)$$

Wacht 0,05 seconde

Door de waarde 0,1 en 0,05 te variëren kunnen ze de gevoeligheid aanpassen. De wachttijd heeft invloed op de reactiesnelheid (hoe kleiner hoe beter) en de belasting van de computer. Bedenk dat er Arduino via ScratchClient ca. 10 berichten per seconde verstuurt, dus de 0,05 is waarschijnlijk al zo gevoelig als het kan.

OPDRACHT 6: Laat het bestaande blok staan, maar zorg dat het niet meer gestart wordt als je op de groene vlag klikt.

Zie boven. De daar getoonde versies hebben al geen verbinding met de startvlag.

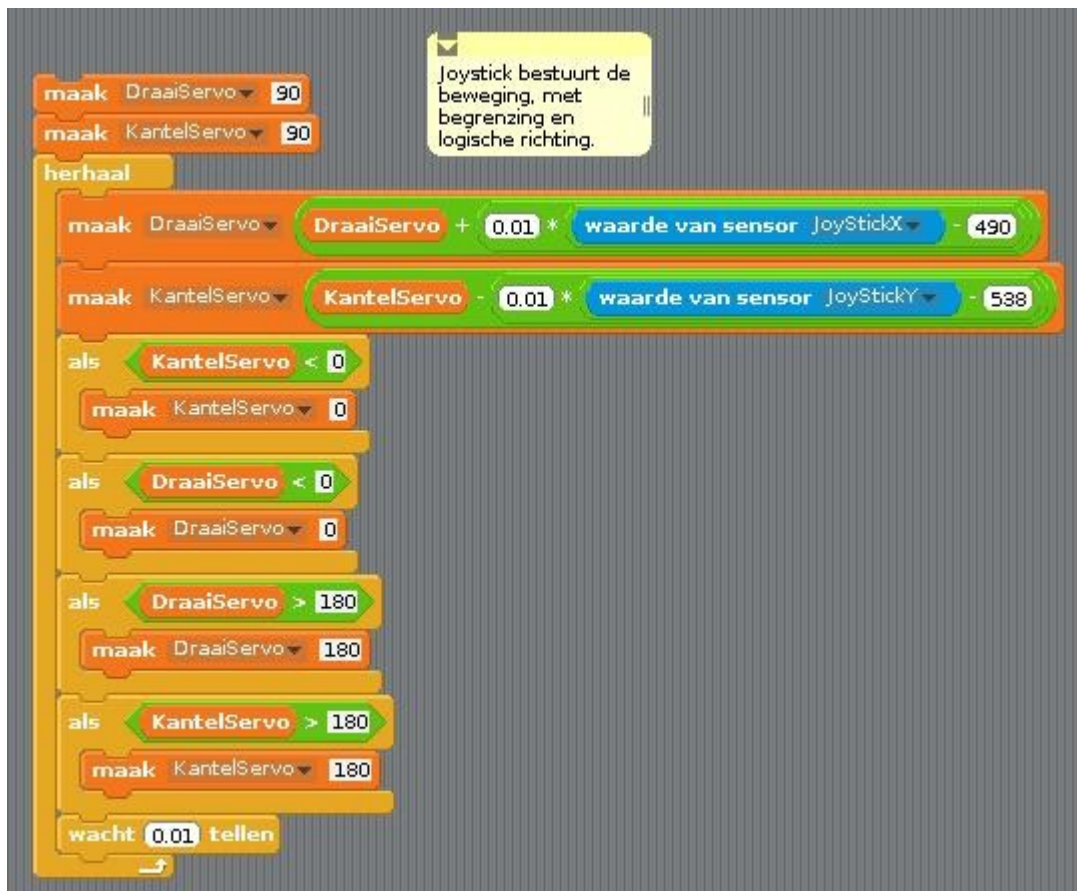
OPDRACHT 7: Maak een nieuw blok met de formule hierboven. Doe dit ook voor de kantelservo. En probeer het uit. En zet de servo's eerst in het midden, dus op 90 (als je op de groene vlag klikt).



OPDRACHT 8: Pas het programma aan zodat het logisch werkt.

Zie bij de opdracht hieronder.

OPDRACHT 9: Pas het programma zo aan dat de variabelen niet buiten de 0 tot 180 kunnen komen.



OPDRACHT 10: Bedenk wat voor oplossingen je zou kunnen verzinnen om dat wel meer effect te laten hebben en bespreek het met je begeleider.

Hier wordt de afwijking gekwadrateerd, maar wel met behoud van teken. Daardoor worden kleine uitslagen nog kleiner en grote uitslagen groter. Door het kwadrateren gaat het teken verloren, dus daar moet je apart iets voor doen.



2.5 GEBRUIK VAN DE JOYSTICK OM TE TEKENEN

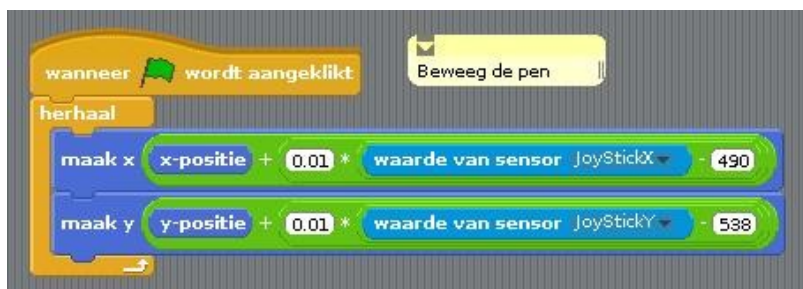
OPDRACHT 14: Zorg met bovenstaande blokken dat je de pen op en neer kunt bewegen door de joystickknop op in te drukken en los te laten zoals boven besproken.



OPDRACHT 15: Bedenk de blokken die je nodig hebt om dit te maken en doe het ook voor de Y-positie.

Zie bij de volgende opdracht.

OPDRACHT 16: Kijk naar de waarde van de sensor JoystickX en JoystickY en als die anders zijn dan 512, verander dan de waarde 512 in het programma naar wat de middenpositie echt is. Let op: dit zal waarschijnlijk anders zijn voor X dan voor Y.



OPDRACHT 18: Zorg dat de lijndikte verandert als je de linker knop indrukt en kijk of het werkt.



OPDRACHT 19: Maak hetzelfde voor het veranderen van de kleur van de lijn door het drukken op de rechter knop.



OPDRACHT 20: De lijndikte wordt nu steeds dikker, maar hoe zou je kunnen zorgen dat de lijndikte weer dunner wordt? Bedenk iets en bespreek met je begeleider.

Je kunt de knop als toggle switch gebruiken en daarmee de richting omkeren. Dus indrukken en de lijn wordt dikker. Loslaten en weer indrukken en de lijn wordt dunner. Dit is hetzelfde als bij een dimmer voor een lamp die met een enkele drukschakelaar werkt.



OPDRACHT 21: Maak hetzelfde wat je voor lijndikte hebt gedaan ook voor de kleur zodat je die in twee richtingen kunt veranderen.



OPDRACHT 22: Test uit wat er gebeurt als je dat blok weer actief maakt.

Dit blok doet *pen op* zodra als de muis losgelaten is. Dus als je met de joystickknop de pen naar beneden probeert te krijgen, wordt dat onmiddellijk ongedaan gemaakt doordat de muis niet ingedrukt is.



OPDRACHT 23: Pas het nu zo aan dat het tekenen met de muis het tekenen met de joystick niet in de weg zit en omgekeerd.



De essentie van de wijziging is dat eerder de pen op / pen neer opdrachten nu bestuurd worden door het ingedrukt en losgelaten **worden** en niet door het ingedrukt of losgelaten **zijn**.

OPDRACHT 24: Als je hier bent aangekomen dan mag je iets anders bedenken om te toe te voegen.

Bespreek met de leerling wat ze zouden willen toevoegen. Ze zouden bijvoorbeeld het reageren op knoppen kunnen vervangen door reageren op toetsen in plaats van op knoppen zodat ze het ook thuis kunnen doen.