

Project M&M Sorteermachine

René Hammink (3068269), Wout Hoenekamp (3032949)

June 2023

**UNIVERSITY
OF TWENTE.**

Inhoudsopgave

1 Abstract	3
2 Introductie	3
3 Eisen	4
3.1 materialen	4
3.2 Knoppen	4
3.3 Autmatische stop	4
3.4 Statemachine	4
4 Design	5
4.1 impact van eisen op design	5
4.2 keuzes van het design	5
4.3 Het Design	6
4.3.1 De sorteermachine	6
4.3.2 De Code	7
5 Testen van het prototype	8
5.1 Testen	8
5.1.1 Sluitertijd	8
5.1.2 Servo 1	8
5.1.3 Servo 2	9
5.1.4 De gehele machine	9
5.2 Analyse	10
6 Conclusie	12
7 Referenties	13

1 Abstract

In deze designopdracht wordt een M&M-sorteermachine gebouwd aan de hand van de Featherboard M4, twee servo's en een kleurensensor. In het uiteindelijk gemaakte prototype, worden de servo's gebruikt om de M&M's te verplaatsen van de invoer tot de bakjes voor de verschillende kleuren. De kleurensensor wordt gebruikt om de kleuren van de M&M's te herkennen.

Het gemaakte prototype sorteert 25 M&M's in rond de 45 seconden van de invoer tot de bakjes, met ongeveer 7 M&M's in de verkeerde bakjes of naast de bakjes gevallen. Hiermee is het design doel van de sorteermachine grotendeels gehaald, maar is er zeker nog ruimte voor verbetering.

2 Introductie

In dit project wordt er een apparaat gebouwd waarmee verschillende kleuren M&M's zo snel mogelijk gesorteerd moeten worden. Deze M&M sorteermachine, wordt volledig gebouwd uit schroot materiaal, denk hierbij bijvoorbeeld aan karton. Er wordt met behulp van een Featherboard twee servo's en een kleursensor aangestuurd om van de 25 M&M's zo snel mogelijk de kleur te herkennen en deze in de juiste bakjes te sorteren. Hierbij wordt gebruik gemaakt van de Featherboard M4 waarvoor in de programmeertaal Python, een code wordt geschreven om de M&M's zo snel mogelijk naar de juiste bakjes te sorteren.

3 Eisen

Er zijn een paar eisen gesteld waar het uiteindelijke design voor de sorteermachine aan moet voldoen:

3.1 materialen

Allereerst moet de gehele M&M sorteermachine gebouwd worden van schroot materiaal en kleine bevestigingsmaterialen. Hieronder vallen bijvoorbeeld karton en pvc-buizen, maar ook lijm, tireps en tape.

Verder zijn er ook maar beperkte materialen beschikbaar wat betreft de elektronica, de belangrijkste hiervan zijn twee servos en een kleurensensor.

3.2 Knoppen

Daarnaast moet het systeem een aan/uit knop bevatten om het systeem mee aan of uit te zetten. Deze moeten ten alle tijden werken. Zodra deze knop voor het eerst wordt ingedrukt moet er een tijd gaan lopen om de tijd te timen die de sorteermachine erover doet om de M&M's te sorteren. De volgende keer dat de knop ingedrukt wordt, moet de sorteermachine stoppen en uitgaan.

3.3 Autmatische stop

Zodra de laatste M&M van de 25 M&M's die oorspronkelijk in de invoer van het sorteersysteem zaten, gesorteerd is, moet het systeem automatisch stoppen. Ook moet dan weergegeven worden hoe lang er gedaan is om alle M&M's te sorteren.

3.4 Statemachine

De laatste eis is dat de code voor de featherboard geschreven wordt als een statemachine. Een statemachine is een machine die maar in een staat/state tegelijk kan zijn. Deze staat verandert alleen naar een andere staat bij een bepaalde input, meerdere inputs of gebeurtenissen.

4 Design

4.1 impact van eisen op design

Van de verschillende eisen die gesteld zijn voor het uiteindelijke design, is de beperking van de te gebruiken materialen tot schroot materiaal de eis die het meest invloed heeft gehad op het uiteindelijk design. Doordat de sorteermachine gemaakt is van materialen zoals karton, moest er goed opgelet worden dat de hele constructie stootbestendig was en stevig bleef staan. Hiervoor zijn er ook verschillende pilaren gemonteerd in de vorm van een pvc-buis die de verschillende lagen karton in de lucht houden en zorgen dat de machine niet uit elkaar valt.

Verder zijn er maar twee servos en een kleurensensor beschikbaar. Waarvan het meest uitdagende de beperkte hoeveelheid servos was. Om de M&M's met twee servos naar de juiste bak te brengen, is één servo gebruikt om de M&M's van de plek waar deze binnengaan, naar de kleurensensor te brengen, en vervolgens weer door naar de tweede servo. Deze stuurt vervolgens de M&M's aan de hand van de gemeten kleur naar het juiste bakje.

De overige drie eisen hebben vooral een impact gehad op het schrijven van de code. Het schrijven van code in de vorm van een statemachine heeft geen directe invloed gehad op de snelheid van de code en de sorteermachine zelf. Deze manier van coderen heeft er vooral voor gezorgd dat eventuele fouten in de code er snel uitgefilterd konden worden. Ook de overige twee eisen hebben geen invloed op de snelheid van de sorteermachine gehad. Deze zijn gemonteerd om de machine veilig uit te kunnen schakelen mocht er iets fout gaan, en om de tijd te meten die het kost voor de machine om een bepaald aantal M&M's te sorteren.

4.2 keuzes van het design

In het uiteindelijk design is gekozen om vooral te focussen op het correct meten en sorteren van de M&M's en later op de snelheid. Dit heeft meerdere gevolgen gehad voor het uiteindelijke design.

Allereerst voor de kleurensensor. Hierbij is gekozen voor een grotere sluitertijd, deze bepaalt hoelang de kleurensensor meet, om zo te kunnen zorgen dat van de M&M's consistent de juiste kleur wordt gemeten. Dit zorgt er wel voor dat de tijd die het kost per M&M om deze te sorteren iets omhoog gaat.

Ook is er tijdens het maken van de sorteermachine, veel tijd gegaan naar het kalibreren van de kleurensensor. Dit houdt in dat er per kleur M&M nauwkeurig wordt bepaald welke RGB-code (Red,Green,Blue) hier het best bij past, om zo het aantal fout gesorteerde M&M's te verkleinen.

4.3 Het Design

4.3.1 De sorteermachine

Het uiteindelijk design bestaat uit twee delen: Het M&M Meet deel, en het M&M distributie deel. In het eerste deel worden de M&M's naar de kleurensensor gebracht, daar op kleur gemeten, en vervolgens worden ze verder gebracht naar het M&M distributie deel. In het M&M distributie deel worden de M&M's naar het juiste bakje geleid. Een afbeelding van het totale design, is te zien in Figuur 1 hieronder.



Figuur 1: Het totale Design

Het meet deel Het meet deel van het design bestaat uit een buis waaruit de M&M's binnenvallen, een halve ronde schijf met een rond gat erin, deze wordt gedraaid met behulp van een servo, en een kleurensensor. Dit is te zien in Figuur 2. Hier is de bewegende schijf goed op te zien. Hier vallen de M&M's vanuit de buis op en blijven hier op liggen. Zodra de schijf draait, zal het gat in deze schijf precies onder de M&M's terecht komen. Hier valt een M&M in en de volgende M&M blijft liggen op de M&M eronder. Vervolgens beweegt het gat met de M&M erin richting de kleurensensor waar de servo weer stopt met bewegen.

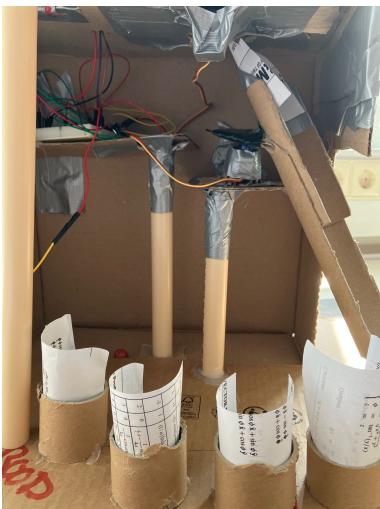
Vervolgens meet deze kleurensensor de kleur van de M&M en draait de schijf door totdat het gat in de schijf uitlijnt met een gat op de kartonlaag waar deze op gemonteerd is, hier valt de M&M doorheen het distributie deel in.



Figuur 2: Het meet deel

De kleurensensor werkt met van red-green-blue (RGB) kleurcodes. Om een kleur te koppelen aan de RGB-kleurcode is een code geschreven die aan de hand van de kleurensensor de RGB-kleurcode leest. Om dit te kalibreren zijn de verschillende kleuren M&M's in de sorteermachine gedaan, en is de RGB-kleurcode gemeten en gekoppeld aan de kleur van de M&M.

[1][3][2]



Figuur 3: Het distributie deel

Het distributie deel Zodra de kleur is gemeten, wordt dit doorgegeven naar het distributie deel. Hier draait servo 2 (de distributie servo) de 'glijbaan', zo dat de te sorteren M&M in het juiste bakje terecht komt. Een foto van het distributiedeel is ook te zien in Figuur 3. Aan de hand van de kleur die gemeten is, draait de servo naar de aangegeven hoek bij deze kleur, zodat de glijbaan wijst naar de plek waar de M&M terecht moet komen. In dit geval zijn dit de bakjes te zien in Figuur 3.

Voor de bakjes zijn dikke ronde schijfjes van kartongemaakt. Om ervoor te zorgen dat de M&M's niet uit deze bakjes stuiteren, zijn er papiertjes aan de buitenkant van deze bakjes geplakt die de muren van deze bakjes verlengen. Dit is ook te zien in Figuur 3.

[1][3]

4.3.2 De Code

Om de machine te laten werken, is een code geschreven en geupload in de Featherboard M4. Deze code is in de vorm van een statemachine geschreven, deze code is apart bijgeleverd. Voor de verschillende states zijn oorspronkelijk functies gedefinieerd, om deze later op te roepen als de machine in een bepaalde state zit. Omdat dit zorgde voor een onduidelijk en niet werkend script, is er uiteindelijk voor gekozen om deze functies gewoon direct neer te zetten bij de bijbehorende state.

In het script worden eerst de verschillende inputs en outputs gedefinieerd, denk hierbij aan de twee servo's, de kleurensensor en de aan/uit knop.

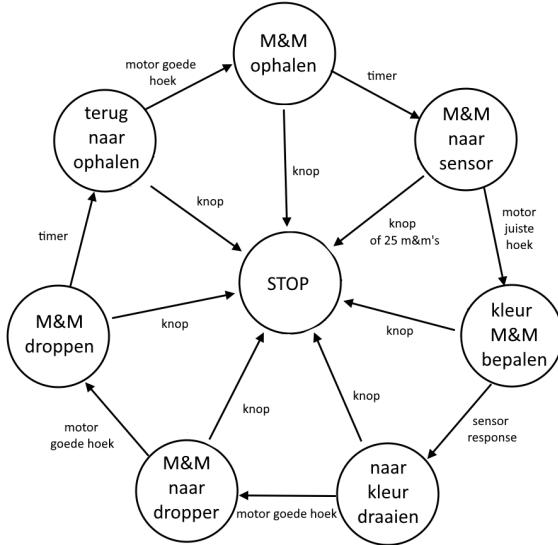
Hieronder zijn de verschillende states, en de RGB-kleurcodes voor de verschillende kleuren M&M's gedefinieerd. Hier staan ook de verschillende hoeken voor de servo's gedefinieerd. Voor servo 1 zijn dit de hoek waar het gat in de schijf uitlijnt met de invoer, de hoek waar het gat in de schijf uitlijnt met de kleurensensor en de hoek waar het gat in de schijf uitlijnt met het gat waar de M&M's doorheen moeten vallen naar de tweede servo. Voor servo 2 zijn dit de hoeken waarop de verschillende bakjes voor de verschillende kleuren liggen.

Vervolgens gaat de machine aan zodra de knop wordt ingedrukt en controleert deze steeds in welke state de sorteermachine is.

De verschillende states zijn in Figuur 4 als een state-diagram te zien. In Figuur 5 is nog extra ingezoomd op de substates van de state: "Naar kleur draaien"

In de code staan de verschillende states allemaal onder elkaar met if/elif statements. In elk van de verschillende states, staat dan weer aangegeven wat de voorwaarde is om door te gaan naar een andere state.

Het koppelen van de gemeten RGB-kleurcode aan een van de voorgedefinieerde RGB-kleurcode wordt zo aangepakt: elke RGB-kleurcode wordt gezien als een punt in 3D. Er zijn voorgedefinieerde punten voor rood, oranje, bruin, blauw, geel en groen, oftewel de kleuren voor de M&M's. Van elk van deze gemeten punten (de gemeten RGB-kleurcode van de M&M) wordt vervolgens berekend bij welke van de voorgedefinieerde punten deze het dichtstbij ligt. Dit is de kleur van de M&M. [1][3][2]



Figuur 4: Het statediagram van de sorteermachine

5 Testen van het prototype

5.1 Testen

Er zijn verschillende testen uitgevoerd om het uiterste te vinden van de sorteermachine, hierin is vooral gevarieerd in de snelheid van de servos en de sluitingstijd van de kleurensensor. Bij de verschillende testen, stond vooral centraal dat de maximale snelheid werd gevonden, waarbij alle M&M's nog steeds in de juiste bakjes werden gesorteerd.

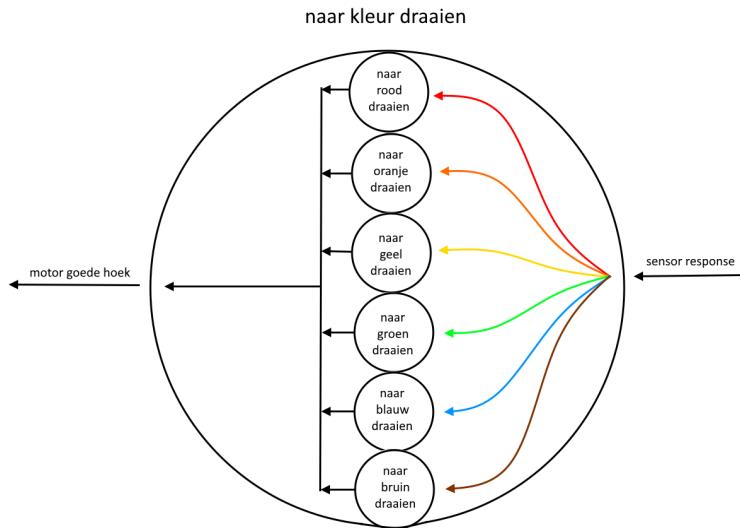
5.1.1 Sluitertijd

Allereerst is er getest voor de sluitertijd. Om dit te testen zijn verschillende kleuren M&M's in de sorteermachine gedaan en is deze vervolgens aan gezet. Vervolgens is gekeken wat de kortste sluitertijd is waarop de kleurensensor nog de M&M's in de juiste plek sorteerde/de juiste kleur meette. Hiervoor is er een sluitertijd gevonden van 300 milliseconden.

Bij deze sluitertijd worden sommige M&M's nog wel door elkaar gehaald, waarvan de RGB-kleurcode veel overeenkomt. Denk hierbij bijvoorbeeld aan de oranje M&M's, de rode M&M's en de bruine M&M's.

5.1.2 Servo 1

Voor servo 1, dit is de servo die vast zit aan de schijf te zien in Figuur 2, is ook geprobeert de optimale snelheid te vinden. Bij deze test moest vooral gekeken naar de M&M's, deze mochten namelijk door de snelheid van de servo niet weg stuiteren of uit de schijf vallen. Ook is een mogelijke valkuil dat de servo zelf de snelheid aangegeven door de code niet kan bijnouden. In dit geval zal de servo niet even ver draaien als de code zelf omdat de servo bijvoorbeeld nog niet bij de 60 graden is aangekomen terwijl dit volgens de geschreven code al wel is gebeurt. Het kan ook zo zijn dat de servo juist te snel beweegt, waardoor deze nog een beetje verder doorschiet dan de hoek waar deze eigenlijk stil op moet staan. Dit kan weer invloed hebben op de gemeten kleur door de kleurensensor. Om dit



Figuur 5: De substates van "Naar kleur draaien"

te meten zijn er meerdere M&M's in de machine gestopt en is de snelheid van de servo steeds wat hoger gezet om zo de maximale snelheid te vinden waarbij de M&M's niet uit de machine schieten en goed worden rondgebracht door de schijf. Uiteindelijk is de snelheid van de servo gevonden als 5.5 in de geschreven code. De uitleg over deze code is te zien onder het kopje De Code bij het Design.

5.1.3 Servo 2

Voor servo 2, dit is de servo die de M&M's naar de bakjes geleidt, is het vooral van belang dat de servo niet doorschiet en de M&M's naast de bakjes laat vallen. De snelheid van deze servo is getest door verschillende M&M's in de sorteermachine te doen en de snelheid langzaam omhoog te zetten. Voor deze servo is uiteindelijk een snelheid gevonden van 6 in de geschreven code. De uitleg over deze code is ook weer te zien onder het kopje De Code bij het Design. Ook is er, om de servo sneller te kunnen laten bewegen, een range gekozen rondom de hoek waar deze moet stoppen, om zo ervoor te zorgen dat de servo de plek waar deze stil moet staan niet overschiet. Zonder deze range zal de servo steeds heen en weer blijven gaan net buiten de aangegeven hoek.

5.1.4 De gehele machine

De uiteindelijk test voor de gehele sorteermachine is drie keer uitgevoerd. Voor de eerste test waren dit de uitkomsten:

Het totale aantal misplaatste M&M's: 1

De totale tijd om alle M&M's te sorteren: 47 seconden.

Voor de tweede test waren dit de uitkomsten:

Het totale aantal misplaatste M&M's: 7

De totale tijd om alle M&M's te sorteren: 45 seconden.

Voor de derde test waren dit de uitkomsten:

Het totale aantal misplaatste M&M's: 7

De totale tijd om alle M&M's te sorteren: 43 seconden.

Tijdens elk van deze testen, werden de blauwe, groene en gele M&M's telkens volledig correct gesorteerd. Maar de bruine, rode en oranje werden soms nog door elkaar gehaald, vooral door de eerder besproken RGB-kleurcodes die bij deze kleuren erg op elkaar lijken.

Ook moet vermeld worden dat bij test 1 de 25 M&M's niet allemaal tegelijk in de machine werden gegooid met de hiervoor bedoelde pvc-buis, maar achter elkaar waardoor de machine niet in een keer het gewicht van de 25 M&M's die naar beneden vallen, op zich krijgt.

Dit was ook wat er fout ging bij test twee. Door het gewicht van de 25 vallende M&M's, zakte het platform waar de schijf op ligt een beetje in, waardoor de ruimte tussen de pvc-buis via waar de M&M's naar binnen komen, en de schijf die de M&M's verplaatst, wat vergroot werd. Hierdoor vielen er al twee M&M's gelijk aan het begin van de test uit de sorteermachine.

5.2 Analyse

Als er nu terug gekeken wordt op het gehele ontwerp, dan zijn er een paar dingen die minder goed uit de testen zijn gekomen en die beter hadden gekunt.

Als eerst de bakjes waarin de M&M's gesorteerd werden. Door de vorm en de grootte van deze bakjes, kwam het redelijk vaak voor dat de M&M's of naast de bakjes vielen, of gelijk weer uit het bakje stuiterde. De bakjes die gemaakt zijn, zijn namelijk niet erg groot en hebben muren die niet erg hoog zijn wat het weer makkelijker maakt voor de M&M's om eruit te stuiteren. Ook de verlengingen van de muren met het papier heeft hierbij niet genoeg geholpen met het verbeteren van de bakjes.

Om dit in een volgend design te verbeteren, kunnen de bakjes langwerpig worden gemaakt zodat de M&M's niet makkelijk meer naar voren uit de bakjes kunnen stuiteren.

Daarnaast kunnen de bodems van de bakjes wat zachter gemaakt worden en onregelmatiger, om zo te voorkomen dat de M&M's gaan stuiteren. Om ervoor te zorgen dat de M&M's niet meer naast de bakjes terecht komen, kunnen alle bakjes aan elkaar vastzitten, zodat er geen ruimte meer is tussen de bakjes.

Verder kunnen de kleurensensor en de M&M die op kleur gemeten wordt, meer worden afgesloten van het buitenlicht. Dit buitenlicht kan namelijk voor extra ruis zorgen waardoor het lastiger is voor de kleurensensor om de juiste kleur te meten. Een volledige donkere ruimte voor de kleurensensor, waarbij het enige licht van de led van de kleurensensor komt, zou dus voor duidelijkere metingen kunnen zorgen. Wel moeten, als dit gebouwd wordt, de RGB-kleurcodes voor de M&M's opnieuw gekalibreerd worden.

Om te voorkomen wat er bij test twee fout ging, de M&M's die gelijk op het begin eruit vielen, kan de verdieping waarop de schijf en servo 1 op staan (zie Figuur 1) wat beter ondersteund worden. Denk hierbij bijvoorbeeld aan een soort van paal die de verdieping beter omhoog houdt. Hierdoor blijft de afstand tussen de pvc-buis waar de M&M's op het begin in zitten, en de schijf constant even groot, zodat de M&M's hier niet meer tussenuit kunnen vallen.

Een andere verbetering had nog in de snelheid kunnen zitten. Door bijvoorbeeld de twee servo's tegelijk te laten bewegen of andere bewegingen tegelijk te laten verlopen had er nog veel tijd bespaard kunnen worden tijdens het sorteren. Dit kan bijvoorbeeld door servo 1 van de sensor naar het gat waar de M&M doorheen moet vallen, tegelijk te laten draaien met servo 2 die naar het juiste bakje moet draaien.

De code zelf kan ook nog geoptimaliseerd worden. Door te kijken naar de verschillende onderdelen in de code en te meten welke de meeste tijd kosten, kan gekeken worden waar er mogelijk nog verbetering mogelijk is.

Het uiteindelijke prototype haalde soms ook de rode M&M's, de bruine M&M's en de oranje M&M's door elkaar. Een andere manier van de kleurenmeten, zou kunnen helpen om het verschil in RGB-kleurcode tussen deze kleuren groter te maken. Bijvoorbeeld door de kleuren van de M&M's met elkaar te vergelijken. Dit doe je bijvoorbeeld door een referentie punt te nemen voor de eerste M&M en door vervolgens te meten hoe erg deze kleur verandert. Als dit bijvoorbeeld in combinatie gebruikt wordt met het puntensysteem waar de gemaakte sorteermachine gebruik van maakt, dan kan ervoor gezorgd worden dat de sorteermachine ook de kleuren die op elkaar lijken beter kan onderscheiden.

6 Conclusie

Het doel van deze designopdracht was om een M&M sorteermachine te bouwen, die 25 M&M's sorteert met behulp van twee servo's en een kleurensensor. Het uiteindelijke prototype was in staat om 18 van de 25 M&M's juist te sorteren, waar de grootste fout vooral zat in het design van de bakjes waarin de M&M's gesorteerd moesten worden. Dit is ook te lezen in de Analyse. Ook moest het uiteindelijk design de M&M's zo snel mogelijk sorteren, voor dit prototype is er een snelste tijd gemeten van 43 seconden.

Het gemaakte prototype werkt met een paar fout gesorteerde M&M's wel volgens de gestelde eisen. Maar er is nog veel ruimte voor verbeteringen, die beschreven staan in de Analyse.

7 Referenties

Referenties

- [1] Lady Ada. *Adafruit Feather M4 Express*. URL: <https://learn.adafruit.com/adafruit-feather-m4-express-atsamd51/circuitpython-servo>. (bekeken op: 29-06-2023).
- [2] Bill Earl. *Adafruit Color Sensors*. URL: <https://learn.adafruit.com/adafruit-color-sensors/python-circuitpython>. (bekeken op: 29-06-2023).
- [3] Kattni Rembor. *CircuitPython Essentials*. URL: <https://learn.adafruit.com/circuitpython-essentials/circuitpython-digital-in-out>. (bekeken op: 22-06-2023).