



DIGITAL TWIN VAN DE DOBOT: HET TERUG KUNNEN KOPPELEN VAN MECHANISCHE LASTEN NAAR DE GEBRUIKER

Een ontwerprapport om vast te stellen wat de nodige ontwikkelstappen zijn om een digitale tweeling te creëren van een machine. Met een korte introductie op het analyseproces voor predictief onderhoud.

AFSTUDEERSCHRIPTIE BATENBURG JB SYSTEMS

WESTLANDSEWEG 190, 3131 HX

VLAARDINGEN, 03 JUNI 2022

VERSIE 1.0

STAGIAIR:	ÖMER SEKMEN	
BEDRIJFSBEGELEIDER(S):	TIM VERWAAL	T
	RAOUL MOLENKAMP	KIM A.J. VERBERT
STAGEBEGELEIDER 1:	GERARD M. TUK	
STAGEBEGELEIDER 2:		

Versiebeheer

Versie	Datum	Wijzigingen/Opmmerkingen	Auteur
0.1	08-05-2022	Eerste concept versie.	Ömer Sekmen
0.2	13-05-2022	Aanpassingen n.a.v. bespreking van versie 0.1 met stagebegeleider op 12-05-2022.	Ömer Sekmen
0.3	23-05-2022	Tweede concept versie.	Ömer Sekmen
0.4	26-05-2022	Aanpassingen n.a.v. bespreking van versie 0.3 met R. Molenkamp op 25-05-2022.	Ömer Sekmen
0.5	30-05-2022	Derde concept versie.	Ömer Sekmen
1.0	03-06-2022	Eerste definitieve versie.	Ömer Sekmen

Voorwoord

Deze afstudeerscriptie is voortgekomen uit het documenteren van de afstudeerstage van de opleiding mechatronica aan De Haagse Hogeschool. Tijdens de afstudeerperiode van 17 weken is de opdracht uitgevoerd voor Batenburg JB Systems.

De opdracht is uitgevoerd met een onderzoekend perspectief. Hierbij is vooral onderzocht naar hoe een digital twin toegepast kan worden en naar de uitdagingen die daarbij komen kijken. De kern van de scriptie gaat uit van de kennis dat vergaard is in de afgelopen vier leerjaren van de opleiding.

Graag bedank ik Tim Verwaal die als bedrijfsbegeleider zijn kennis en ervaringen heeft gedeeld. Van de opleiding bedank ik graag Kim Verbert die tijdens het afstuderen veel ondersteuning heeft geboden zowel rondom de technische aspecten van het afstudeerproces als inhoudelijke begeleiding.

Tot slot wil ik graag de werknemers van JB Systems bedanken voor een aangename en zeer leerzame periode. Met nadruk Raoul Molenkamp die inbreng heeft gehad in het ontwikkelproces van de scriptie en Wilbert Blom die zijn – veel gewaardeerde – kennis in Unity met mij heeft gedeeld.

Vlaardingen, juni 2022
Ömer Sekmen

Samenvatting

Vanwege de stijgende hoeveelheid nieuwe ontwikkelingen rondom het digitaliseren van machines, wil JB Systems een begin maken aan het implementeren van deze functionaliteiten in hun eigen simulaties.

Dit rapport beschrijft het digitaliseringsproces van de DOBOT, een kleinschalige robotarm. Het doel van dit rapport is om uiteindelijk een beeld te krijgen van de uitdagingen in het digitaliseringsproces. Ook wordt duidelijk hoe bestaande projecten aangepast kunnen worden om aan de nieuwe functionaliteiten te voldoen. Het streven is om klanten te kunnen voorzien van een tijdige waarschuwing over mogelijk onderhoud aan een machine, dit wordt preventief onderhoud genoemd.

Dit rapport is geschreven met de volgende hoofdvraag centraal: *Wat zijn de ontwikkelstappen die nodig zijn om de DOBOT te digitaliseren, voor het toepassen van predictief onderhoud?*

Deze vraag wordt beantwoord door middel van drie deeldoelstelling die elk een stap verder zetten in het digitaliseringsproces. Deze doelstellingen luiden als volgt: Het vergaren van data van de last aan de end-effector, het omzetten van de fysieke DOBOT naar een digitaal model en het uitvoeren van een krachtenanalyse aan de hand van de verzamelde data. Bij het ontwikkelen van de bovenstaande deeldoelstelling is er geen rekening gehouden met het geven van een definitief oordeel over de onderhoudsstaat van de DOBOT.

De volgende resultaten zijn opgeleverd om de drie deeldoelstelling te behalen. Er is een loadsensor ontwikkeld die met behulp van rekstroken de kracht aan de end-effector van de DOBOT kan meten. Dit doet de sensor met een nauwkeurigheid van 0.5 N. Om de sensor te kunnen bevestigen aan de DOBOT is er een end-effector ontworpen. Daarnaast wordt er door middel van operationele versterkers het signaal versterkt tot signalen die uitgelezen kunnen worden door de DOBOT.

De DOBOT is digitaal te volgen aan de hand van een digital twin simulatie. De simulatie beschikt over een interface waarop belangrijke data zoals de hiervoor gemeten last getoond kan worden. Ook is er een mogelijkheid om via de interface een FEM analyse aan te roepen die aan de hand van een vooraf opgestelde krachtenmodel vervormingen in de verschillende objecten kan berekenen.

Concluderend blijkt dat de grootste uitdaging ligt bij het vergaren van de juiste data in voldoende hoeveelheid. Een tweede uitdaging is om een juiste systematische benadering van de krachten in de constructie te maken. Het virtuele model kan immers niet beter zijn dan de benaderingen die ervoor gesteld zijn.

Inhoudsopgave

Voorwoord	2
Samenvatting	3
1 Inleiding	8
2 Probleemanalyse	9
2.1 Functies	9
3 Conceptvorming & -toetsing	11
3.1 Programma van Eisen	11
3.1.1 Toelichting van enkele eisen	12
3.2 Conceptoplossingen	13
3.3 Conceptoplossingen toetsen	15
3.4 Het gekozen concept	17
4 Systeemarchitecturen van het gekozen concept	18
5 Het ontwerp van de loadmeter	20
5.1 Mechanisch Ontwerp	20
5.1.1 Spanningsmodule	20
5.1.2 End-effector	22
5.2 Elektrisch Ontwerp	23
5.2.1 Stap 1: Brug van Wheatstone	23
5.2.2 Stap 2: Versterking 1	25
5.2.3 Stap 3: Versterking 2	26
5.2.4 Hardware lowpass-filters	28
5.2.5 Meetpunten van de schakeling	28
5.3 Software Ontwerp	29
5.3.1 Filter algoritme selectie	29
5.3.2 De adaptieve reset van het kalman filter	32
6 De digital twin simulatie van de DOBOT	33
6.1 De opbouw van de virtuele arm	33
6.1.1 De verschillende coördinatenstelsels	33
6.2 Meebewegen met de werkelijke arm	34
6.2.1 Bewegingsbeperkingen toepassen op het virtuele ontwerp	34
6.3 Visuele terugkoppeling van de krachtenanalyse	36
6.3.1 De verschillende oppervlaktegeometrieën van een object	36
6.3.2 Het koppelen van hoekpunten van de verschillende geometrieën en de uitvoerwaarden	37
6.4 Interactie met de software	38
7 De krachtenanalyse van een last aan de DOBOT	40
7.1 Het krachtenmodel	40
7.2 FEM analyse met behulp van Calculix	41
7.2.1 Automatisering van de analyse	41
8 Testresultaten & evaluatie	43
8.1 Systeemtesten	43
9 Conclusies & Aanbevelingen	46

Referenties	48
A Algemene bijlage	50
B Testplannen	65
B1 Systeemtesten	66
C Uitgebreide berekeningen	73
C1 Berekeningen hoofdstuk 5 Het ontwerp van de loadmeter	73
C2 Berekeningen hoofdstuk 6 De digital twin simulatie van de DOBOT	77
C3 Berekeningen hoofdstuk 7 De krachtenanalyse van een last aan de DOBOT	78
D Programma listings	85
E Opdrachtvoorstel	89

Lijst van figuren

1	Lagenmodel van het digitaliseringspad.	9
2	Use case diagram van de belangrijkste functies.	10
3	De bewegingsvrijheden van de DOBOT.	14
4	Renders van het gekozen eindconcept.	17
5	Sequence diagrammen van de algemene werking van het systeem	18
6	De volledige Block Definition Diagram van het systeem.	19
7	Opbouw van een rekstrook.	20
8	Spanningsmodule als een buis.	20
9	De definitieve spanningsmodule.	22
10	Bevestigingsonderdelen geleverd door DOBOT.	22
11	De definitieve end-effector.	23
12	De brug van Wheatstone zonder balanceer-circuit.	23
13	De brug van Wheatstone met balanceer-circuit.	24
14	Benadering van weerstandswaarde R_{15} en R_{16}	25
15	Het inwendige circuit van de INA122.[10]	25
16	Circuit om V_{ref1} mee in te stellen.	26
17	Blokdiagram van alle versterkingsstappen.	26
18	Deelschakeling van versterkingsstap 2.	27
19	Het elektrisch circuit weergeven in een blokdiagram.	27
20	stapresponsie van diverse filter algoritmen.	30
21	Stabiliteitstest van diverse filter algoritmen.	31
22	Kalman filter met en zonder adaptieve reset.	32
23	De coördinatenstelsels van de drie conversieprogramma's.	33
24	de DOBOT in zijn oorspronkelijke stand, beide armen 45° (<i>links</i>). De DOBOT met een Front Arm rotatie van 60° en een Rear Arm rotatie van 30° (<i>rechts</i>).	34
25	Verduidelijkingen van de contactpunten.	35
26	Soorten oppervlaktegeometrieën.	36
27	Het deformeren van een object.	37
28	Beide panelen van de gebruikersinterface.	39
29	De 10 staaf/balk elementen.	40
30	De activity diagram van het uitvoeren van een FEM analyse	42

A1	Sequence diagrammen van de belangrijkste functies.	51
A2	Opvang van ongewenste krachten.	52
A3	De gerealiseerde loadmeter met de end-effector.	52
A4	Elektrische schakeling loadmeter.	53
A5	VeeCAD schakeltekening van de loadmeter.	54
A6	Loadmeter PCB.	55
A7	Het effect van een laagdoorlaatfilter op V_{out1}	57
A8	Het effect van een laagdoorlaatfilter op V_{out2}	57
A9	Het effect van een laagdoorlaatfilter op V_{cc}	57
A10	Section view van de DOBOT.	58
A11	Exploded view van de tien DOBOT onderdelen.	58
A12	Hiërarchie van de objecten in Unity.	59
A13	Toelichting translatie van contactpunten.	61
A14	De conversiestappen om het CAD model om te zetten naar een SMSTL.	62
A15	De HMI van de digital twin.	63
A16	Initiële configuratie van de onderdeel Front Arm	64
C1	De brug van Wheatstone met benamingen en annotaties.	73
C2	CAD modellen van onderdelen rear arm en foot.	78
C3	Vrijlichaamsschema's van onderdeel rear arm.	78
C4	Vrijlichaamsschema en CAD model van onderdeel rear arm connector 1.	79
C5	Vrijlichaamsschema en CAD model van onderdeel rear arm connector 2.	79
C6	Vrijlichaamsschema en CAD model van onderdeel rear arm connector 3.	80
C7	Vrijlichaamsschema en CAD model van onderdeel front arm connector.	80
C8	Vrijlichaamsschema en CAD model van onderdeel front arm.	81
C9	Vrijlichaamsschema en CAD model van onderdeel end-effector.	82
C10	Vrijlichaamsschema en CAD model van onderdeel wing.	82

Lijst van tabellen

1	Het programma van eisen.	12
2	Deel van het morfologisch overzicht met conceptoplossingen.	13
3	Bepaling van de weegfactoren per functie.	16
4	Keuzetabel conceptoplossingen.	16
5	Resultaten van vergelijkingstest van diverse filter algoritmen.	31
6	Behaalde systeemtesten.	44
7	Behaalde eisen.	45
A1	Eisen met betreffende functies.	50
A2	Morfologisch overzicht van alle conceptoplossingen.	50
A3	Onderdelenlijst schakeling loadmeter.	56
A4	Algemene kosten.	56
A5	Onderdelentabel section view A10 en exploded view A11.	58
A6	Contactparen van de basisobjecten.	60
B1	Systeemtest 1. Bewegingsvrijheid van de DOBOT.	66
B2	Systeemtest 2. Het meetbereik van de loadmeter.	67
B3	Systeemtest 3. Gegevens van de loadmeter.	68
B4	Systeemtest 4. Belasting op de loadmeter.	69
B5	Systeemtest 5. Gegevens van de Digital Twin.	70

B6	Systeemtest 6. Gegevens van de krachtenanalyse.	71
B7	Systeemtest 7. Overige vaste eisen	72
C1	Variabelen in de krachtberekeningen.	83

Lijst van listings

D1	Benadering Weerstand R_{15} en R_{16}	85
D2	Berekening van het simpel voortschrijdend gemiddelde.	86
D3	Berekening van het gewogen voortschrijdend gemiddelde.	86
D4	Berekening van het exponentieel voortschrijdend gemiddelde.	86
D5	Werking van de 1€ filter. [1]	87
D6	Werking van het kalman filter. [2]	88

1 Inleiding

JB Systems richt zich op het ontwikkelen van industriële softwareoplossingen in diverse markten. De softwareoplossingen variëren van besturingssystemen voor machines tot het maken van simulaties die gebruikt worden voor trainingsdoeleinden.

In de recente jaren wordt er veel ontwikkeld om de grens tussen simulatie en werkelijkheid te verkleinen. Deze ontwikkelingen geven gebruikers de mogelijkheid om meer toezicht te hebben op de status van de werktuigen. Dit heeft ervoor gezorgd dat de vraag bij JB Systems om deze ontwikkelingen mee te nemen in simulaties is vergroot.

JB Systems is benieuwd naar hoe deze ontwikkelingen toegepast kunnen worden in bestaande projecten. Daarvoor is er een opdracht tot stand gekomen waarbij er een stap wordt gezet in het digitaliseren van een machine met als streven het kunnen uitvoeren van preventief onderhoud. Ook wil het team met de opdracht duidelijkheid krijgen over waar in het ontwikkelproces de uitdagingen liggen.

Het uiteindelijke wenkend perspectief van dit project is om klanten te kunnen voorzien van een tijdige waarschuwing over mogelijk onderhoud aan een machine. Dit pad kan beschreven worden met de volgende lagen.

In laag 1 wordt het ontwerp in een 3D engine, zoals Unity 3D, gemodelleerd en heeft het dezelfde bewegingsbeperkingen als de werkelijke machine. In laag 2 wordt het model uitgebreid met ingangssignalen die uitgelezen worden vanuit het fysieke model. Tenslotte kan er in laag 3 verschillende analyses toegepast worden om onder andere een conclusie te kunnen trekken over de onderhoudsstaat van een machine.

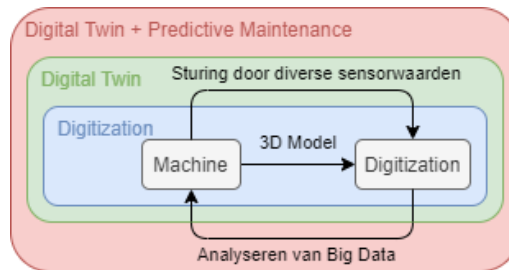
Het hoofddoel van de opdracht is om de ontwikkelstappen vast te leggen die nodig zijn bij het digitaliseren van een machine/systeem. Hoewel sommige onderdelen in dit rapport toepassingsspecifiek zijn, worden er in latere hoofdstukken onderwerpen behandeld die in het algemeen van toepassing zijn in het digitaliseringsproces. Dit doel wordt bereikt door het digitaliseringsproces uit te voeren op de DOBOT, een kleinschalig robotarm. Om het hoofddoel te kunnen halen zijn er deelvaststelling opgesteld, deze zijn: het uitlezen van een last op de DOBOT, de DOBOT simuleren als een digital twin en het uitvoeren van een krachtenanalyse aan de hand van de uitgelezen last.

De algemene hoofdvraag die in dit rapport wordt behandeld luidt als volgt: *"Wat zijn de ontwikkelstappen die nodig zijn om de DOBOT te digitaliseren, voor het toepassen van predictief onderhoud?"*

Dit rapport is geschreven op volgorde van de ontwikkelingen die gemaakt zijn tijdens het digitaliseren van de DOBOT. In hoofdstuk 3 worden als eerst de ontwerpkeuzes onderzocht. Daarbij worden er mogelijk concepten opgesteld en getoetst. Na de conceptkeuze wordt in hoofdstuk 5 de module ontworpen dat zorgt voor de belangrijkste schakel, de sensor voor het ingangssignaal. Vervolgens wordt er in hoofdstuk 6 beschreven hoe de simulatie van de digital twin van de DOBOT werkt. Na de terugkoppeling van de simulatie wordt laag 3 van het digitaliseringspad behandeld met een krachtenanalyse, dit is opgenomen in hoofdstuk 7. Tenslotte wordt het volledige systeem getest en wordt er een conclusie uitgebracht in hoofdstukken 8 en 9.

2 Probleemanalyse

Het digitaliseren van een machine is een vrij recente ontwikkeling die vele nieuwe aspecten van een machine kan laten zien. Dit geeft als probleem dat de nodige stappen om een machine te digitaliseren onbekend zijn. De term digitaliseren heeft niet één betekenis, maar kan verdeeld worden in meerdere stadia. Dit is kort behandeld in de inleiding, hieronder is in figuur 1 een toelichting te vinden.



Figuur 1: Lagenmodel van het digitaliseringspad.

Laag 1: Digitization

Het ontwerp is in een 3D engine gemodelleerd en kan bewegen als de werkelijke machine.

Laag 2: Digital twin

Het digitaal model wordt gestuurd door middel van meerdereingangssignalen die uitgelezen worden van de machine. Het mee simuleren van de niet-sturing gerelateerde meetwaarden zoals temperaturen of krachten vallen ook binnen deze laag. De digitale machine ervaart hetzelfde als de werkelijke machine.

Laag 3: Digital twin + analyse van data

Tenslotte kan er een begin worden gemaakt aan de vervolgstap: analyse van vergaarde data. Hier worden historische sensordata geanalyseerd om een conclusie te kunnen trekken over diverse aspecten van de machine. In het geval van predictive maintenance wordt deze analyse vervolgens teruggekoppeld aan de machine in de vorm van een tijdig waarschuwing voor een onderhoudsbeurt.

Het doel van de opdracht is om het model van de DOBOT aan het eind van het afstudeertraject minstens tot laag 2 te ontwikkelen. Daarbij wordt er in de vorm van een FEM krachtenberekening, een deel analyse uit laag 3 meegenomen. Het belang van de opdracht is dat het aan het einde van het traject duidelijker is wat de nodige stappen zijn om het digitaliseringsproces toe te passen in een bestaand of toekomstig project.

2.1 Functies

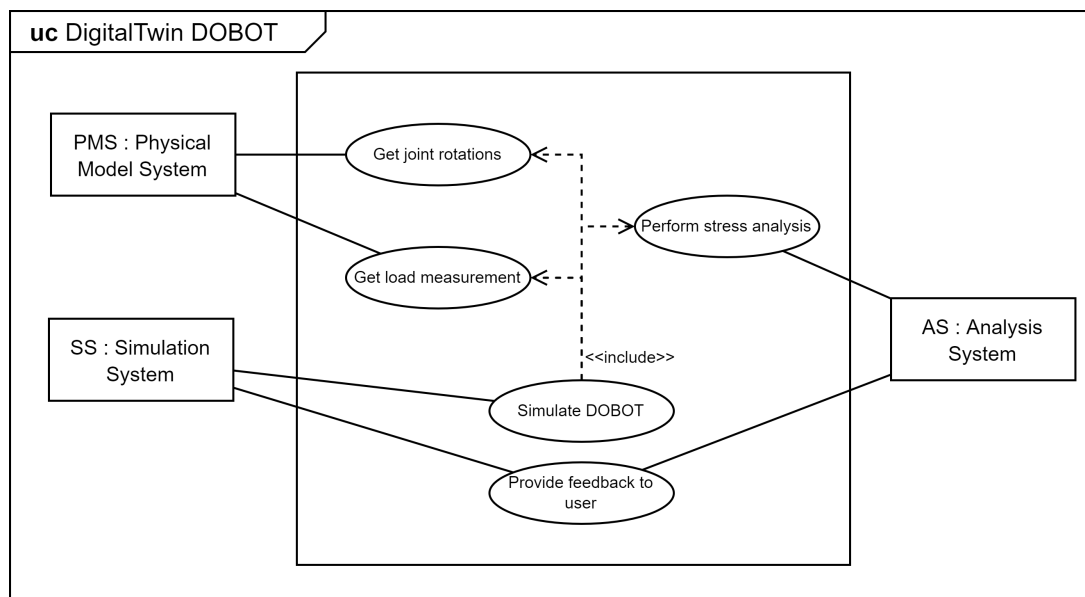
Om dit stappenplan op te kunnen zetten worden er drie deelopdrachten uitgevoerd. Elk van deze deelopdrachten behandelt een belangrijke functie uit het lagenmodel dat hierboven is beschreven. Met deze functies worden er in hoofdstuk 3 concepten opgesteld om uiteindelijk tot een definitief ontwerp te komen. De functies luiden als volgt:

- **Load meten:** Meten van de last op de end-effector. Hiermee kunnen de krachten in de constructie van de DOBOT bekend worden.
- **End-effector belasten:** Het aanbrengen van een kracht op de end-effector. De DOBOT moet op een herhaalbaar manier belast worden.

- **DOBOT simuleren:** Een digital twin maken die meebeweegt met de DOBOT.
- **Krachtenanalyse uitvoeren:** Vervormingen van de DOBOT berekenen aan de hand van de gemeten load. Hiermee worden de kritieke punten van de constructie berekend.
- **Krachtenanalyse terugkoppelen:** De vervormingen terugkoppelen naar de gebruiker. Hiermee worden de berekende kritieke punten weergegeven aan de gebruiker.

Er is voor de bovenstaande functionaliteiten gekozen, omdat het uitgangspunt van de opdracht predictive maintenance is. Het meten van de last aan de end-effector is daarbij cruciaal om iets te kunnen zeggen over spanningen in de constructie van de DOBOT.

De functionaliteiten worden uitgevoerd door de betreffende actors, deze verdeling is terug te zien in figuur 2. Actors worden door Delligatti gedefinieerd als personen, organisaties of andere systemen die een interactie aangaan met een systeem [3, p. 53–54]. De actoren worden na concepttoetsing gedefinieerd in hoofdstuk 4 door middel van een block definition diagram.



Figuur 2: Use case diagram van de belangrijkste functies.

3 Conceptvorming & -toetsing

In het voorgaande hoofdstuk zijn de functionaliteiten waaraan voldaan moet worden besproken. In dit hoofdstuk worden er eerst eisen opgesteld aan het ontwerp om ervoor te zorgen dat deze functionaliteiten behaald worden. Vervolgens worden de conceptoplossingen besproken en getoetst om tot een definitief ontwerp te komen.

3.1 Programma van Eisen

Om de opdracht af te bakenen wordt er een eisenpakket opgesteld dat later gebruikt zal worden als een controlelijst om te bepalen of de opdracht succesvol is afgerond. In dit hoofdstuk wordt het eisenpakket doorgenomen en waar nodig toegelicht.

De eisen worden opgedeeld in twee categorieën:

- **MoSCoW:**
Om eisen prioriteit te geven wordt de MoSCoW-methode toegepast. Deze methode kaart de Must have's, Should have's, Could have's en Won't have's aan van een project. De eisen waaraan in ieder geval voldaan moeten worden voor een succesvolle afronding worden aangegeven met must have eisen. Should have eisen zijn wensen die niet direct noodzakelijk zijn voor het succes van de opdracht. Wel kunnen deze eisen invloed hebben bij het nemen van beslissingen. Could have eisen zullen behandeld worden als hier tijd voor over is. Tenslotte dienen de won't have eisen als verduidelijking om aan te geven welke eisen er niet in het project behandeld zullen worden.
- **Vast of variabel:**
Dezelfde eis kan zowel variabel als vast gedefinieerd worden. Een variabele eis geeft de mogelijkheid om een concept te toetsen op de mate waarin dit voldoet aan de eis. Een concept met een hogere gemiddelde score krijgt voorrang in het ontwerpproces.

Het eisenpakket is weergegeven in tabel 1. Onder de tabel worden er enkele eisen toegelicht. Deze zijn in de tabel aangegeven met een ster.

	Eisennr	Eisen	Must have	Should have	Could have	Won't have	Vast	Variabel	Functioneel	Realisatie
Algemeen	1	De bouw van de DOBOT blijft onveranderd.	x				x			x
	2*	Bewegingen van de dobot mogen niet begrensd worden.		x				x	x	
	3*	Alle originele componenten van de DOBOT worden gebruikt.	x				x			x
	4	De totale kosten van de opdracht is onder €150.	x					x		x
Loadmeting	5*	Het domein van de loadmeting ligt minstens tussen 0 N en 5 N.	x					x	x	
	6*	De loadmeter kan minstens belast worden met een variabele kracht tussen 0 N en 5 N.	x					x	x	
	7	Het analoge uitgangssignaal van de loadmeting heeft een bereik van 0 V tot 3.3 V.	x				x			x
	8	Als spanningsbron wordt er gebruik gemaakt van een 12 V uitgang van de DOBOT.	x				x			x
	9*	De gemeten kracht heeft een nauwkeurigheid van ± 0.5 N.		x				x	x	
	10	Het protoboard is een stripboard (TriPad) bestaande uit drie geclusterde koperen strips.	x				x			x
	11	Het protoboard voor de loadmeting wordt gefixeerd aan het werkoppervlak met PCB afstandhouders.		x			x			x
	12	Het protoboard beschikt over mogelijkheden om het nul-last signaal te kalibreren	x				x			
	13*	Het protoboard wordt zo ontwikkeld dat componenten individueel verwisseld kunnen worden.	x				x			x
	14	Het protoboard heeft één kabel / componenten per soldeerpad.	x				x			x
	15	Het vermogen van de versterker is niet meer dan 0.25 W.	x				x		x	
	16	De constructie van de loadmeter vangt radiale krachten en (torsie)momenten op.		x				x	x	
	17	De loadmeter meet axiale krachten die op de end-effector uitgevoerd worden.		x				x	x	
	18	De last op de endeffector is onafhankelijk van de positie van de armen			x			x		x
	19	De loadmeter wordt ontworpen als een custom PCB.				x	x			x
	20	Er worden externe spanningsbronnen gebruikt in het ontwerp van de loadmeter				x	x			x
Simulatie	21	Alle drie bewegingsassen bewegen in de simulatie mee met de DOBOT.	x				x		x	
	22	De simulatie heeft een verversnelheid van minimaal 20 Hz. (excl. krachtenanalyse)		x			x		x	
	23	De DOBOT wordt gesimuleerd in Unity 3D.		x			x			x
	24	De simulatie houdt rekening met botsende en wrijvende oppervlakten.				x	x		x	
Krachtenanalyse	25	De vervorming van de DOBOT bij een axiale kracht aan de end-effector is visueel uit de krachtenanalyse te halen.			x		x		x	
	26	De vervormingen uit de krachtenanalyse worden teruggekoppeld naar de 3D simulatie in Unity.			x		x		x	
	27	Bij de krachtenanalyse worden de armen van de DOBOT niet versimpeld tot staven.			x		x			x
	28	Radiale krachten en (torsie)momenten worden meegenomen in de analyse.				x	x		x	

Tabel 1: Het programma van eisen.

3.1.1 Toelichting van enkele eisen

2. Bewegingen van de DOBOT mogen niet begrensd worden.

Deze variabele eis geeft de mate aan waarin de DOBOT zijn oorspronkelijke bewegingsvrijheden behoudt. De drie vrijheidsgraden worden samen met de translatieassen waarin de end-effector beweegt weergegeven in figuur 3.

3. Alle originele componenten van de DOBOT worden gebruikt.

Onder de originele componenten vallen de stappenmotoren en de twee armen van de DOBOT. Deze worden niet vervangen door een conceptoplossing van een deelfunctie

5. Het domein van de loadmeting ligt minstens tussen 0 N en 5 N.

De conceptoplossing voor de functie, load meten, meet op zijn minst een ingangskracht tussen 0 N en 5 N. Dit is een variabele eis en zal beoordeeld worden op de mate waarin dit ingangsbereik varieert bij de conceptoplossingen. Het bereik kan ook negatief worden, dit geeft aan dat het mogelijk is dat de loadmeter drukkrachten kan meten.

6. De loadmeter kan minstens belast worden met een variabele kracht tussen 0 N en 5 N.

Deze variabele eis zegt wat over de mate waarop de end-effector belast wordt. Er wordt uitgegaan van een minimale variabele trekkracht tussen de 0 N en 5 N. Net als de loadmeting kan de last negatief of groter dan 5 N worden.

9. De gemeten kracht heeft een nauwkeurigheid van ± 0.5 N.

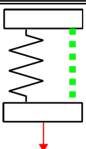
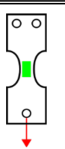
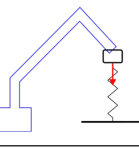
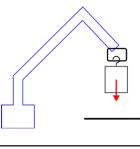
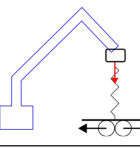
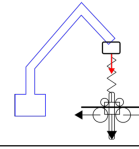
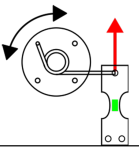
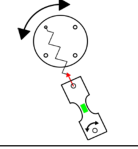
De gemiddelde van n aantal metingen hebben een maximale systematisch fout van 0.5 N in beide richtingen.

De conceptoplossingen zullen in de volgende deelhoofdstukken per functie beschreven worden.

3.2 Conceptoplossingen

De volledige lijst met conceptoplossingen om de functies uit het vorige deelhoofdstuk uit te kunnen voeren zijn in tabel A2 weergegeven. In tabel 2 is er een deel van deze oplossingen weergegeven. Vele aspecten van de opdracht zijn van tevoren al vastgelegd en is het daarom niet mogelijk om een keuze te maken voor sommige conceptoplossingen. De keuzemogelijkheden liggen bij deze opdracht vooral in de eerste twee punten. De conceptoplossingen worden per functie in de volgende stukken kort toegelicht.

Tabel 2: Deel van het morfologisch overzicht met conceptoplossingen.

Functies	Conceptoplossingen					
	1	2	3	4	5	6
Load meten						
	Veer + afstandmeter	Rekstrook				
End-effector belasten						
	Vaste veer	Kalibratiegewichten	Vaste veer + horizontale geleider	Verticaal bewegende veer + horizontale geleider	Servomotor met een torsieveer	Servomotor met een trekveer

Functie 1: load meten

Er zijn twee oplossingen bedacht waarop de last op de end-effector gemeten kan worden.

Opl. 1 - Veer + afstandmeter:

Deze oplossing bestaat uit een mechanisme van twee platformen die onder elkaar opgehangen worden door middel van lineaire geleiders en een veer. Aan de hand van de afstand tussen de platformen en de veerconstante van de veer kan de kracht op het onderste platform benaderd worden.

Opl. 2 - Rekstrook:

Deze oplossing maakt gebruik van een rekstrook om de last te bepalen. Wanneer er een mechanische spanning wordt aangebracht op het materiaal, verandert de weerstand van het rekstrook. Door deze verandering kan er een differentiaalspanning gemeten worden die afhankelijk is van de last op het materiaal.

Functie 2: end-effector belasten

Het is gewenst om de last die op de DOBOT werkt (softwarematig) aan te kunnen sturen. Om dit te realiseren worden er hieronder een aantal deeloplossingen behandeld. Deze deeloplossingen kunnen verder ingedeeld worden in twee categorieën; schijnbare belastingen en werkelijke belastingen. Bij werkelijke belastingen ervaart de DOBOT de last in de volledige constructie van de arm. Dit komt dichtst bij de werkelijkheid waarbij de DOBOT een massa optilt. Bij schijnbare belastingen wordt er lokaal, door middel van een servomotor, een spanning aangebracht in het materiaal waarop een rekstrook bevindt. Deze spanning bootst een last na zonder dat de gehele constructie van de DOBOT belast wordt.

Oplossingen voor schijnbare belastingen kunnen enkel gecombineerd worden met oplossing rekstrook voor de functie load meten. Het voordeel van een schijnbare belasting tegenover een werkelijke belasting is dat de gehele arm van de DOBOT niet fysiek belast wordt, dat verkleint de kans op falen. Daarnaast kan de spanning op de rekstrook op alle posities van de DOBOT geregeld worden, waarbij dat bij enkele conceptoplossingen niet mogelijk is. Het nadeel is dat de conceptoplossingen in categorie schijnbare belasting deels het doel van de opdracht mislopen. Het is namelijk belangrijk dat de virtuele versie hetzelfde ervaart als de fysieke DOBOT.

Eerst zullen de conceptoplossingen uit categorie werkelijke belastingen behandeld worden:

Opl. 1 - Vaste veer:

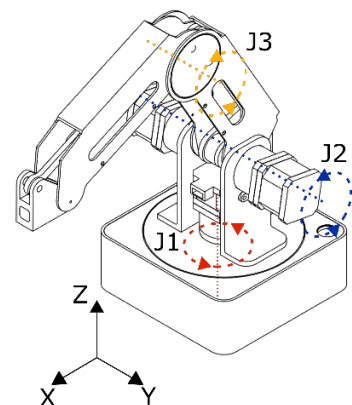
De vaste veer is een oplossing waarbij de kracht gevarieerd kan worden door de veer uit te rekken met de robotarm. De veer wordt bevestigd aan een vast punt op het werkoppervlak. Dit houdt in dat de end-effector alleen in de Z-as kan verplaatsen recht boven de veer. Het voordeel van deze oplossing is dat de kracht die op de end-effector werkt softwarematig geregeld kan worden.

Een nadeel is dat de kracht die op de end-effector werkt afhankelijk is van de Z-positie van de end-effector. Dit heeft als gevolg dat de maximale kracht alleen op de maximale Z-positie uitgeoefend kan worden. Dezelfde kracht kan dus niet bereikt worden om de minimale Z-positie.

Opl. 2 - Vaste veer + horizontale geleider:

Deze conceptoplossing is een uitbreiding op de vorige oplossing. Hierbij wordt de veer bevestigd aan lineaire geleiders aan het werkoppervlak. In de vorige oplossing is de veer, en daarmee de end-effector, bevestigd aan een vaste X en Y coördinaat. Dit zorgt voor een enkele bewegingsvrijheid in de Z-as, waarmee de kracht geregeld kan worden. Hierbij is de kracht afhankelijk van de Z-positie en daarmee hoeken J2 en J3. Voor een willekeurige combinatie van hoeken J2 en J3 is er maar één kracht mogelijk.

Deze oplossing zorgt ervoor dat er met behulp van lineaire geleiders de positie van de veer in de X-as mee kan bewegen. Dit heeft als gevolg dat de end-effector alleen nog een bewegingsbeperking heeft in de Y-as. Zo is de kracht alleen nog afhankelijk van de Z-positie. Een kracht op de end-effector kan met verschillende hoeken van J2 en J3 bereikt worden.



Figuur 3: De bewegingsvrijheden van de DOBOT.

Opl. 3 - Verticaal bewegende veer + horizontale geleider:

Zoals de vorige conceptoplossing bouwt deze oplossing verder op de voorgaande oplossing. Hierbij wordt de uiteinde van de veer met een actuator aangestuurd in de Z-richting. Dit betekent dat de kracht niet meer afhankelijk is van de Z-positie van de end-effector. Voor elke positie van de end-effector kan de veer aangestuurd worden om een willekeurige kracht te introduceren.

Opl. 4 - Kalibratiegewichten

Bij deze conceptoplossing wordt er een last geïntroduceerd door middel van losse kalibratiegewichten. Dit zijn exacte gewichten die met een haak aan de loadmeter gehangen worden. Het voordeel van het gebruik van losse gewichten is dat het een eenvoudig ontwerp is en dat men exact weet hoeveel massa er opgehangen wordt aan de loadmeter. Daarnaast worden de vrijheidsgraden minimaal beperkt en kan er een willekeurige kracht en positie gecombineerd worden.

Het nadeel is dat de last niet variabel is en niet softwarematig geregeld kan worden. Het komt echter wel overeen met de werkelijkheid, waarbij de last aan een kraan niet verandert.

De volgende twee conceptoplossingen zijn uit de categorie schijnbare belastingen.

Opl. 5 - Servomotor met een torsieveer:

Met behulp van een torsieveer en een servomotor wordt er een mechanische spanning geïntroduceerd op een rekstrook. De hoek van de servomotor bepaald het moment dat geleverd wordt door de torsieveer en daarmee de kracht op het rekstrook.

Opl. 6 - Servomotor met een trekveer:

Deze conceptoplossing werkt op hetzelfde principe als de vorige oplossing. Het verschil is dat deze oplossing geen mechanische spanning levert met een torsieveer maar met een trekveer. Hierbij regelt de servomotor de geleverde kracht niet direct met een hoekverdraaiing maar met de verplaatsing van de arm ten gevolgen van de hoekverdraaiing.

3.3 Conceptoplossingen toetsen

Om de losse conceptoplossingen te combineren tot een geheel concept, worden deze individueel beoordeeld op de variabele eisen. Deze aanpak is een gemodificeerde aanpak van Methodisch Ontwerpen [4]. Hier kiest Siers ervoor om de structuurbepaling van diverse concepten uit het morfologisch overzicht te halen. Hieruit komen er meerdere gehele concepten die vervolgens getoetst worden op de variabele eisen.

Door de beperkte keuzemogelijkheid van de conceptoplossingen is ervoor gekozen om de aanpak van Siers om te keren. Er wordt eerst gekeken naar hoe de individuele conceptoplossingen scoren op de gestelde eisen. Op basis van de hoogst scorende conceptoplossingen wordt er één definitief concept opgesteld. De resultaten van deze weging worden verder besproken met de opdrachtgever om uiteindelijk tot een eindconcept te komen.

In dit hoofdstuk worden de conceptoplossingen getoetst volgens de Kesselringmethode. Met deze onderbouwing kan er uiteindelijk in het volgende hoofdstuk een definitief concept opgesteld worden.

Door de keuze om toetsing en conceptvorming om te wisselen kunnen eisen alleen getoetst worden op de betreffende functies. In tabel A1 is weergegeven welke eisen en functies bij elkaar horen.

Ook zijn er per functie weegfactoren bepaald voor de eisen, deze zijn te vinden in tabel 3. De weegfactoren geven aan welke eisen prioriteit hebben over andere.

Elk cel in tabel 3 dient gelezen te worden met de volgende vraag in gedachte: *is eis [eis in rij] belangrijker dan eis [eis in kolom]*? Is het antwoord hierop ja, wordt er een 1 genoteerd in de cel, anders een 0. Aan het eind wordt de som bepaald van elke rij, dit getal is de weegfactor van de eis.

De weegfactoren worden gebruikt om de conceptoplossingen te toetsen, dit is weergegeven in tabel 4. In dit tabel wordt er eerste een score van 1-10 gegeven voor hoe goed de oplossing voldoet aan de eis. Dit cijfer wordt vervolgens vermenigvuldigd met de weegfactor, dit geeft een totale score. Aan de hand van de totale score worden de hoogst scorende conceptoplossingen aangegeven met een groen gekleurde cel. De individuele conceptoplossingen zijn terug te vinden in tabel A2 en deelhoofdstuk 3.2.

Tabel 3: Bepaling van de weegfactoren per functie.

Eis	4	5	9	16	17	weegfactor
4	-	0	0	0	0	1
5	1	-	0	1	0	3
9	1	1	-	1	1	5
16	1	0	0	-	0	2
17	1	1	0	1	-	4

(a) Functie: Load meten

Eis	2	4	6	18	weegfactor
2	-	1	0	0	2
4	0	-	0	0	1
6	1	1	-	0	3
18	1	1	1	-	4

(b) Functie: End-effector belasten

Tabel 4: Keuzetabel conceptoplossingen.

Eis	Opl. 1	Opl. 2	Weegfactor
4	8	5	1
5	3	9	3
9	4	9	5
16	8	6	2
17	10	6	4
totaal	93	113	150
totaal %	62%	75%	100%

(a) Functie: Load meten

Eis	Opl. 1	Opl. 2	Opl. 3	Opl. 4	Opl. 5	Opl. 6	Weegfactor
2	5	9	7	7	10	10	2
4	7	8	7	3	6	6	1
6	6	7	6	7	9	7	3
18	3	8	6	8	8	8	4
totaal	47	79	63	70	85	79	100
totaal %	47%	79%	63%	70%	85%	79%	100%

(b) Functie: End-effector belasten

3.4 Het gekozen concept

Uit de resultaten van het vorige hoofdstuk blijkt snel dat het gebruik van kalibratiegewichten de voorkeur krijgt. Dit komt voornamelijk, omdat het eenvoudig is om de last aan de end-effector te veranderen en de last niet afhankelijk is van de stand van de armen. Zo kan de DOBOT in elke positie belast worden met een willekeurige kracht. In overleg met de opdrachtgever is besloten dat de conceptoplossingen in categorie schijnbare belastingen komen te vervallen, omdat deze de DOBOT niet fysiek belasten.

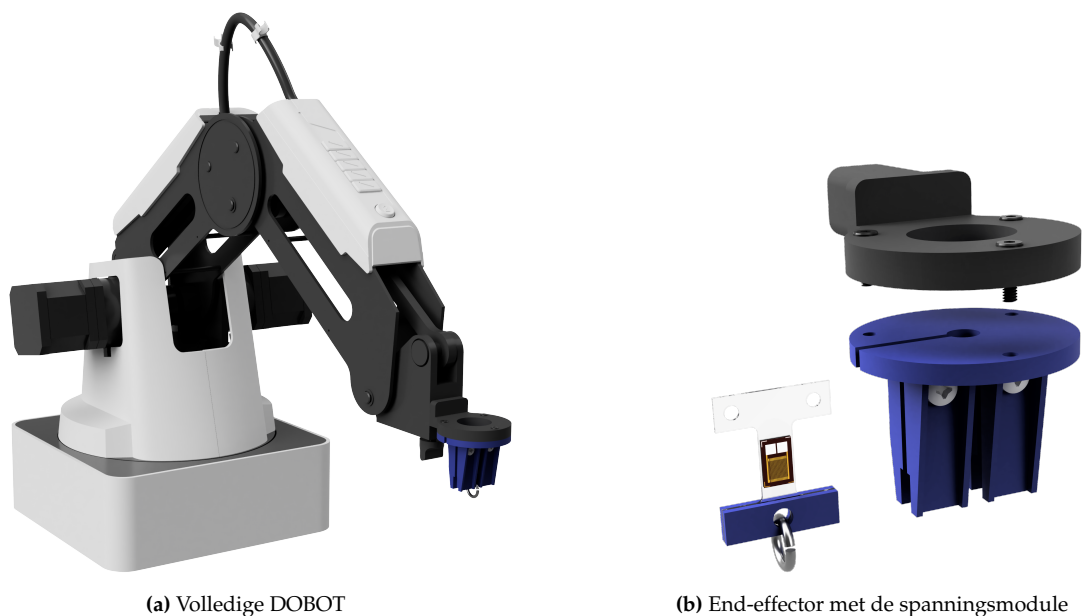
In het eindconcept wordt de kracht gemeten met behulp van rekstroken. De reden dat deze oplossing de voorkeur heeft gekregen op de veer en afstandsmeter komt door de resolutie en nauwkeurigheid van de meting.

De bewegingen die de DOBOT fysiek maakt worden in Unity[5] gesimuleerd met een overeenkomend model. Daarbij communiceert de software in Unity periodiek met de DOBOT. Er is gekozen voor Unity, omdat JB Systems ervaring heeft in het maken van simulaties met dit programma.

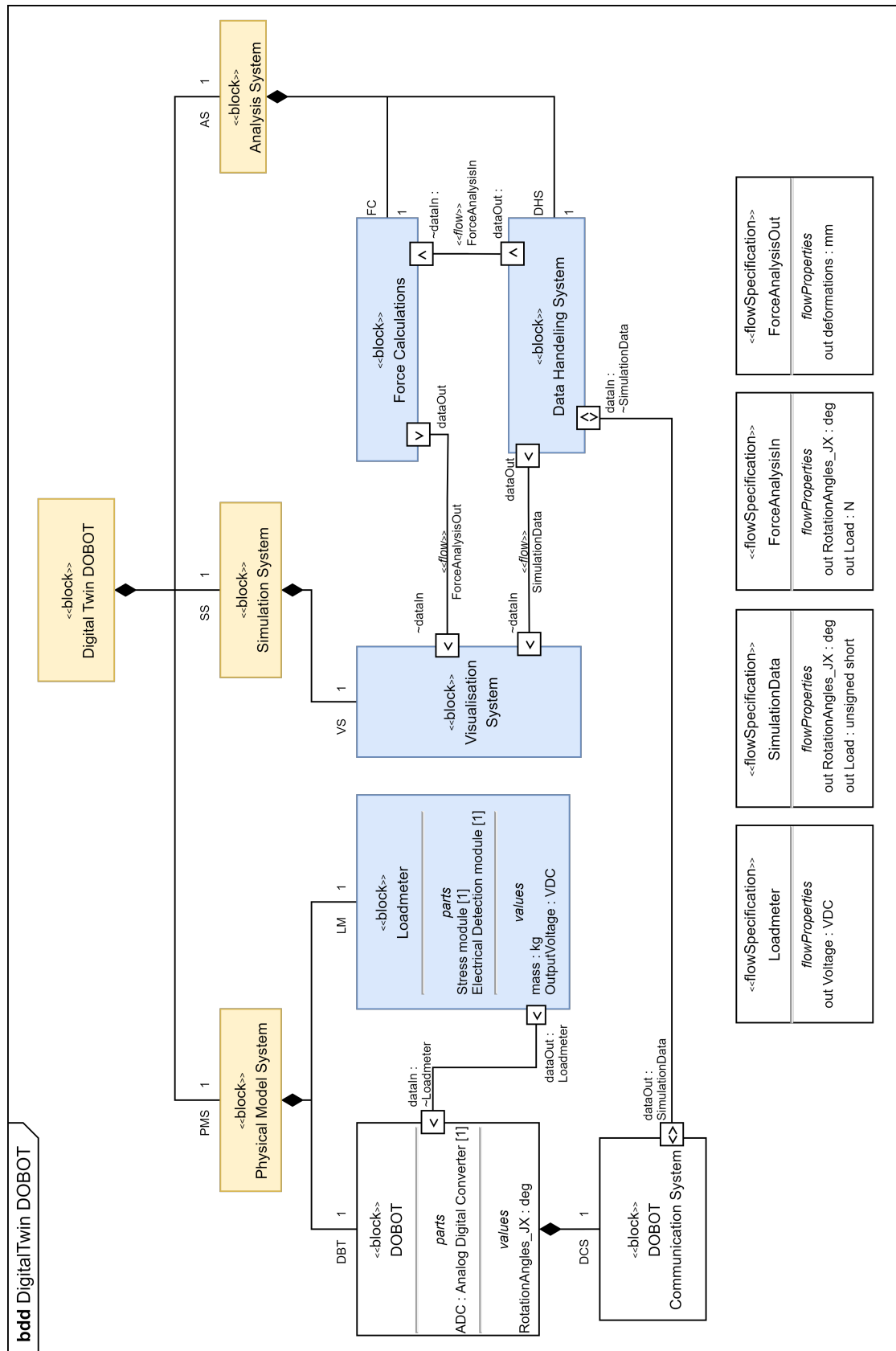
Voor de krachtenanalyse wordt er gebruik gemaakt van CalculiX, een open-source softwarepakket om krachtsommen op te lossen door gebruik te maken van de eindige-elementenmethode (EEM)/ finite element method (FEM). Dit programma maakt het mogelijk om diverse belastingen op 3D modellen te weergeven. Er is gekozen voor Calculix, omdat het een van de meest gebruikte FEM softwarepakketen is. Daarnaast zijn er diverse fora die ondersteuning kunnen bieden.

Als laatst worden de resultaten uit de berekeningen van CalculiX teruggekoppeld aan het model in Unity. Deze resultaten worden niet met hetzelfde ververssnelheid weergegeven in de simulatie. Dit komt door de vele berekeningen die uitgevoerd moeten worden door de onderliggende FEM software.

In figuur 4 is een render van het gekozen eindconcept weergegeven. Hierop is te zien dat alleen de end-effector is aangepast voor de krachtmeting. In de komende hoofdstukken wordt deze keuze en de rest van het ontwerp in detail behandeld.



Figuur 4: Renders van het gekozen eindconcept.

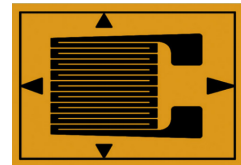


Figuur 6: De volledige Block Definition Diagram van het systeem.

5 Het ontwerp van de loadmeter

Rekstroken worden in de industrie regelmatig gebruikt voor diverse meettoepassingen. Zo kan er in een complexe constructie op een lokale plek het verband tussen mechanische spanning, kracht en rek gelegd worden. In de civiele techniek zijn de toepassingen terug te vinden op bijvoorbeeld bruggen en gebouwen. Door mechanische informatie over een bepaalde tijd te verzamelen, worden de bewegingen van constructies bekend. Vaak wordt er gebruikt gemaakt van de geometrie van de rekstrook door er twee of meer te combineren in verschillende oriëntaties. Op deze manier worden krachten in bepaalde richtingen gefilterd en kunnen enkel de gewenste krachten gemeten worden.

Een rekstrook bestaat uit een folie met daarop een dunne elektrische geleider. De geleider is een draad gewikkeld in een zigzagpatroon van evenwijdige lijnen, te zien in figuur 7. Nadat de folie op een constructieoppervlak gelijmd is, kan het patroon meerekken met de constructie. Door deze uitrekking vervormt het geleidingsmateriaal en verandert de weerstand van het draad. De verandering in weerstand kan met behulp van een elektrische schakeling omgezet worden naar een spanningssignaal die evenredig veranderd met de uitrekking.



Figuur 7: Opbouw van een rekstrook.

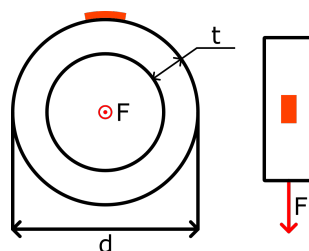
In hoofdstuk 3.4 is er geconcludeerd dat de last door middel van rekstroken bepaald zal worden. In dit hoofdstuk worden de ontwerpkeuzes die hiervoor gemaakt worden behandeld. Dit wordt deels gedaan door middel van beknopte berekeningen en voorbeelden. Voor uitgebreide berekeningen wordt er regelmatig verwezen naar bijlage C. Dit hoofdstuk is ingedeeld in drie deelhoofdstukken, die elk een kerndiscipline van de opleiding mechatronica behandelen.

5.1 Mechanisch Ontwerp

Het mechanisch ontwerp kent twee onderdelen; de spanningsmodule en de end-effector. De spanningsmodule wordt belast met een kracht en de end-effector wordt gebruikt om de spanningsmodule aan de DOBOT te bevestigen. Uit het ontwerpproces is er voor de spanningsmodule een I-profiel van 0.5 mm polycarbonaat gekomen. De end-effector is een 3D geprint onderdeel gemaakt van PETG. In de volgende stukken worden de stappen behandeld die doorlopen zijn om tot dit ontwerp te komen.

5.1.1 Spanningsmodule

Het eerste idee bij het ontwerpen van de spanningsmodule is om gebruik te maken van een dunne buis. Dit met de reden dat een buis eenvoudig te bevestigen is aan de end-effector. Nog een reden is dat de stijfheid van de buis ervoor zorgt dat radiale krachten opgevangen worden door de buis en dus niet resulteren in rekking van het materiaal. Hiermee zou de eis om te filteren op axiale krachten behaald zijn.



Figuur 8: Spanningsmodule als een buis.

De belangrijkste ontwerpcriterium van de spanningsmodule is de grootte van het weerstandsverschil bij een last. Om te berekenen hoeveel de weerstanden van de rekstroken zullen veranderen bij maximale belasting, wordt er gebruik gemaakt van formule (1). De rek van het materiaal bij een gegeven kracht wordt bepaald met een afgeleide formule van de wet van Hooke (2b). Waar de wet van Hooke (2a) het verband tussen kracht, veerconstante en uitrekking vastlegt, doet vergelijking (2b) dat met (trek)spanning, elasticiteitsmodulus en rek.

$$F = c \cdot x \quad (2a)$$

$$\Delta R = GF \cdot \epsilon \cdot R \quad (1)$$

$$\sigma = E \cdot \epsilon \quad (2b)$$

Waarbij,

$\Delta R [Ω]$	Het verschil in weerstand	$F [N]$	Kracht op de	$\sigma [N/m^2]$	(Axiale)spanning in de
GF	Gauge factor rekstrook		spanningsmodule		spanningsmodule
$\epsilon [m/m]$	Rek	$c [N/m]$	Veerconstante	$E [N/m^2]$	Elasticiteitsmodulus
$R [Ω]$	De onbelaste weerstand van de rekstrook	$x [m]$	Uitrekking	$\epsilon [m/m]$	Rek

$$\Delta R = GF \cdot \frac{\sigma}{E} \cdot R = GF \cdot \frac{F}{A \cdot E} \cdot R = GF \cdot \frac{F}{\pi \cdot t(d-t) \cdot E} \cdot R \quad (3)$$

Waarbij: $A [m^2]$ = Het belastingoppervlakte van de spanningsmodule

$t [m]$ = De dikte van de buis

$d [m]$ = De buitendiameter van de buis

Formule (1) kan herschreven worden tot formule (3) met behulp van formules (C1) en (C2). De variabele die in deze vergelijking nog open staan voor verandering zijn de elasticiteitsmodulus E en de belastingoppervlakte A. Na een korte marktonderzoek is er gekozen voor een acrylaat buis met een buitendiameter van 5 mm en een dikte van 1 mm. De keuze is gemaakt op basis van de minimale diameter die de buis nodig heeft waarop een rekstrook bevestigd kan worden. Daarnaast heeft acrylaat de gewenste materiaaleigenschappen met een gemiddelde elasticiteitsmodulus van 2.8 MPa.

De gekozen buis geeft een weerstandsverschil bij maximale belasting van:

$$\Delta R = 2 \cdot \frac{5}{1 \cdot 10^{-3} \pi (5 \cdot 10^{-3} - 1 \cdot 10^{-3}) \cdot 2.8 \cdot 10^9} \cdot 350 = 0.099 \Omega \quad (4)$$

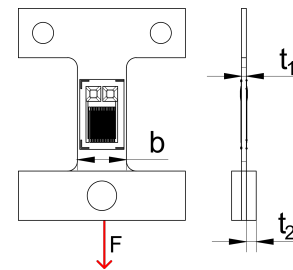
In hoofdstuk 5.2 wordt berekend wat de gevolgen zijn van het bovenstaande weerstandsverschil op de differentiaalspanning aan de uitgang van de brug van Wheatstone. Voor nu wordt er geconcludeerd dat deze differentiaalspanning zo groot mogelijk moet zijn. Het weerstandsverschil van 0.099 Ω resulteert in een differentiaalspanning van 70 μV , dit is te laag om er betrouwbare metingen uit te halen. Hiermee is de oplossing om een buis te gebruiken voor de spanningsmodule komen te vervallen.

Voor een tweede oplossing is er gekeken naar dun plaatmateriaal. Hierbij wordt de breedte van de doorsnede bepaald door de breedte van de rekstrook en de diepte door de dikte van het plaatmateriaal. Zo is het belastingoppervlakte van formule (3) zo klein mogelijk. Het ontwerp van de definitieve spanningsmodule is te zien in figuur 9.

Als plaatmateriaal is er gekozen voor 0.5 mm polycarbonaat, aangegeven met t_1 . Dit heeft als reden dat polycarbonaat een soortgelijke elasticiteitsmodulus heeft als de hiervoor gekozen acrylaat en en daarbij het dunste materiaal dat beschikbaar is.

De gekozen rekstroken hebben een breedte van 4 mm. Er wordt rekening gehouden met een marge van 1 mm aan beide kanten, aangegeven met t_2 . Waardoor de dwarsdoorsnede van het rekkende deel van de spanningsmodule afmetingen van 6 mm x 0.5 mm heeft, aangegeven met $(b \times t_1)$. Het weerstandsverschil met het aangepaste oppervlak bedraagt 0.417Ω .

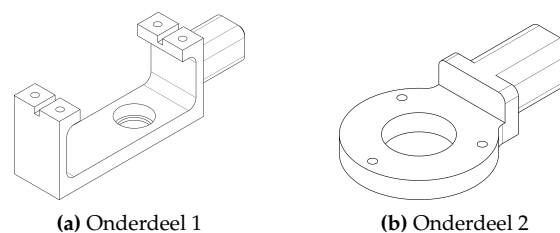
Een verschil van 0.417Ω geeft een differentiaalspanning van 4.28 mV (C3). Dit is 60 keer groter dan de vorige oplossing. Het resultaat kan niet verder verbeterd worden zonder ander materiaal te gebruiken. Daarom is besloten om dit ontwerp tot het definitief ontwerp van de spanningsmodule te maken. Ook wordt in hoofdstuk 5.2.1 door middel van het gebruik van meerdere rekstroken, het totale weerstandsverschil verdubbeld.



Figuur 9: De definitieve spanningsmodule.

5.1.2 End-effector

Bij het ontwerpen van de end-effector is er gekeken naar de meegeleverde onderdelen van de DOBOT. Daarbij zijn er twee bevestigingsonderdelen die gebruikt kunnen worden om toebehoren te bevestigen aan de DOBOT. De end-effector wordt vervolgens om één van deze twee onderdelen heen ontworpen. Eerst zal er een keuze gemaakt moeten worden welke bevestigingsonderdeel er gebruikt zal worden. De twee keuzes zijn weergegeven in figuur 10.

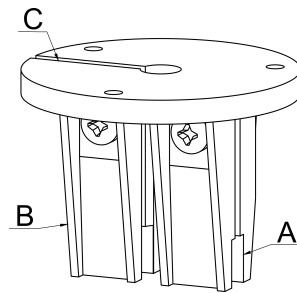


Figuur 10: Bevestigingsonderdelen geleverd door DOBOT.

De oorspronkelijke toepassing van onderdeel 1 in figuur 10a is om een servomotor te kunnen bevestigen aan de DOBOT. Deze servomotor dient als een vierde as, J4, en maakt het mogelijk om de end-effector 360° te roteren. Onderdeel 2 wordt gebruikt als een penhouder. Door de kleine vormfactor van onderdeel 2 en de ligging van de bevestigingsgaten, is er gekozen om het ontwerp van de end-effector te baseren op onderdeel 2.

Met deze eisen is het ontwerp in figuur 11 tot stand gekomen. Door de schuine ribben, te zien op punt B, wordt ervoor gezorgd dat de constructie stijf blijft en de spanningsmodule niet vervormt in ongewenste richtingen. De krachten F_x , M_y en M_z in figuur A2 worden hiermee opgevangen. Krachten F_y en M_x worden opgevangen door de spanningsmodule zelf. Dit komt door het oppervlaktetraagheidsmoment van het I-profiel. Zo heeft alleen de kracht F_z significante invloed op het belasten van de rekstrook.

De onderkant van de spanningsmodule is aan beide kanten verstevigd met een 3D geprint onderdeel van 1 mm. Op punt A is hier ruimte voor gehouden. Ook is er op punt C een opening om het verwisselen van de spanningsmodule te vereenvoudigen.



Figuur 11: De definitieve end-effector.

5.2 Elektrisch Ontwerp

In het vorige deelhoofdstuk is bepaald dat de rekstroken een maximale verschil in weerstand zullen hebben van 0.417Ω . Dit verschil moet echter nog omgezet worden naar een elektrische spanning, zodat het signaal via de ADC van de DOBOT uitgelezen kan worden. In dit deelhoofdstuk worden de stappen doorlopen van hoe dit opgelost is met het elektrische ontwerp. Als volgorde wordt daarbij het verloop van het signaal aangehouden. In figuur A4 en A5 is de gehele schakeling te zien.

Het elektrische ontwerp bestaat uit één meetfase en twee opeenvolgende versterkingsfases. De meetfase maakt gebruik van een brug van Wheatstone met een balanceer circuit om de mechanische spanning te meten. Er worden twee versterkingsfases gebruikt om de hoeveelheid ruis te minderen.. Als laatste is er gebruik gemaakt van enkele laagdoorlaatfilters om ongewenste hoge frequenties aan de uitgangen te filteren.

Berekeningen en simulaties van het elektrische ontwerp worden niet uitgebreid in detail behandeld. De achterliggende theorie wordt kort toegelicht in de vorm van formules en eindberekeningen. Voor uitgebreide berekeningen en simulaties wordt er gerefereerd naar de bijlage. Voor de berekeningen is het boek Electronics, A Systems Approach[6] gebruikt. De simulaties en schakelingen zijn gemaakt in Autodesk Eagle[7], LTSpice[8] en QUCS[9].

5.2.1 Stap 1: Brug van Wheatstone

De eerste stap die genomen wordt in het ontwerp is om het weerstandsverschil om te zetten naar een spanningsverschil. Dit gebeurt met behulp van de brug van Wheatstone. Deze schakeling is in essentie twee spanningsdelers, te zien in figuur 12.

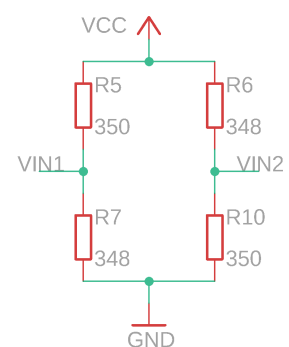
Bij het gebruik van vier identieke weerstanden kan met behulp van vergelijking (5) en (6) de spanning op punten V_{in1} en V_{in2} berekend worden.

$$V_{in1} = \frac{R_7}{R_5 + R_7} \cdot V_{cc} \quad (5) \quad V_{in2} = \frac{R_{10}}{R_6 + R_{10}} \cdot V_{cc} \quad (6)$$

V_{in1} en V_{in2} zijn hierbij de uitgangen van de eerste stap. In de volgende stap wordt er gekeken naar de differentiaalspanning tussen deze twee punten. Deze differentiaalspanning wordt met formule (7) beschreven:

$$V_{diff} = V_{in2} - V_{in1} \quad (7)$$

In eerste instantie is ervoor gekozen om één rekstrook te gebruiken. In figuur 12 is dat weerstand R_{10} . Wanneer deze belast wordt resulteert dit in een spanningsverandering op punt V_{in2} . Met de maximale

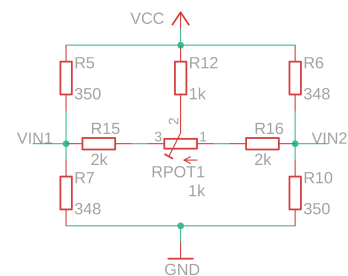


Figuur 12: De brug van Wheatstone zonder balanceer-circuit.

weerstandsverschil, berekend in hoofdstuk 5.1.1, van 0.417Ω is dat een differentiaalspanning van 4.28 mV (vgl. (C3))

Om de verhouding tussen de onzuiverheden van het signaal, door middel van ruis, en het signaal zo klein mogelijk te houden, is het gewenst om de differentiaalspanning zo groot mogelijk te krijgen. In het vorige deelhoofdstuk is dit bij het ontwerpen van de spanningsmodule gedaan door het oppervlak van de dwarsdoorsnede te minderen, waardoor ΔR van de rekstrook groter wordt. Een andere manieren om de differentiaalspanning te verhogen is om gebruik te maken van de symmetrie van de schakeling. In het voorbeeld hierboven wordt de spanning op punt V_{in2} groter en blijft de spanning op punt V_{in1} onveranderd. Door van R_5 ook een rekstrook te maken die op dezelfde manier en mate belast wordt als rekstrook R_{10} , wordt ervoor gezorgd dat de spanning in punt V_{in1} verlaagd. Doordat V_{in1} kleiner en V_{in2} groter wordt, verdubbeld V_{diff} naar 8.57 mV (vgl. (C4)).

Balanceren van het circuit. In de voorgaande berekeningen is er uit gegaan van ideale componenten waarbij de weerstanden in de schakeling exact 350Ω zijn. Dit resulteert in een exacte differentiaalspanning van 0 mV in de onbelaste toestand. Echter hebben componenten in de werkelijkheid systematische en dynamische toleranties. Zo kan de daadwerkelijke weerstandswaarde van een 350Ω weerstand met een tolerantie van 1% tussen 346.5Ω en 353.5Ω liggen. Ook veranderen waarden dynamisch door de warmte die gegenereerd wordt door de weerstand zelf. Om deze onzekerheden tegen te gaan en de nullast in te kunnen stellen op een gegeven spanningswaarde wordt de brug van Wheatstone uitgebreid met een balanceer-circuit, te zien in figuur 13.



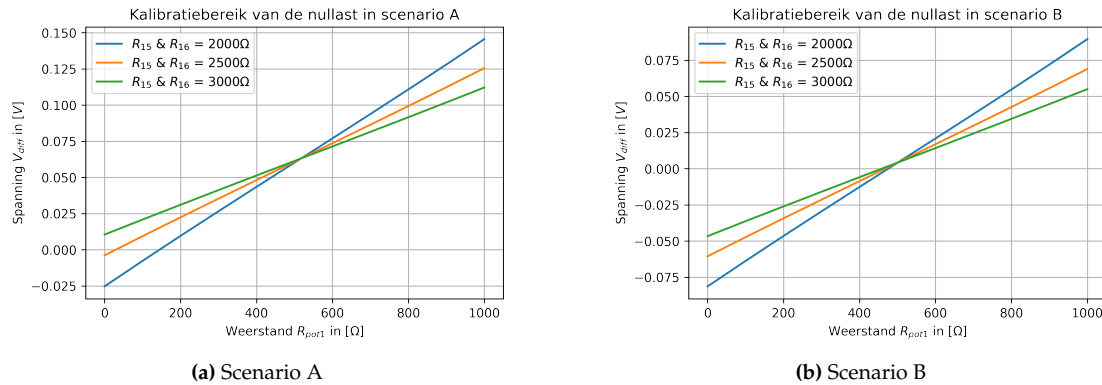
Figuur 13: De brug van Wheatstone met balanceer-circuit.

Door het aanpassen van trimmer potentiometer R_{pot1} kan de schakeling gebalanceerd worden. De totale hoeveelheid die V_{diff} kan veranderen met de uiterste waarden van R_{pot1} wordt bepaald door de omliggende weerstanden R_{12} , R_{15} en R_{16} . Gewenst is om de waarden zo te kiezen dat R_{pot1} de volledige onzekerheidsbereik kan kalibreren. Om tot de juiste waarden te komen van de omliggende weerstanden worden weerstanden R_5 , R_6 , R_7 en R_{10} gemeten. De weerstanden met de kleinste systematische fouten worden handmatig geselecteerd voor gebruik, hierdoor kan de tolerantie verlaagd worden van 1% naar 0.25% . Voor de rekstroken, R_5 en R_{10} , is de basiswaarde 350Ω en voor de weerstanden, R_6 en R_7 , is dat 348Ω .

Voor de onderstaande berekeningen wordt er uit gegaan van twee uiterste scenario's:

	Scenario A		Scenario B
R_6	$348 - 0.25\% \cdot 348 = 347.13 \Omega$	R_6	$348 + 0.25\% \cdot 348 = 348.87 \Omega$
R_7	$348 - 0.25\% \cdot 348 = 347.13 \Omega$	R_7	$348 + 0.25\% \cdot 348 = 348.87 \Omega$
R_5	$350 + 0.25\% \cdot 350 = 350.88 \Omega$	R_5	$350 - 0.25\% \cdot 350 = 349.13 \Omega$
R_{10}	$350 + 0.25\% \cdot 350 = 350.88 \Omega$	R_{10}	$350 - 0.25\% \cdot 350 = 349.13 \Omega$

Bij het berekenen van de weerstandswaarde van $R_{15} = R_{16}$ worden er een aantal regels vastgesteld. R_{12} blijft op 1000Ω en $R_{15} = R_{16}$ om het kalibratiebereik symmetrisch te houden. Door de complexe schakeling is ervoor gekozen om R_{15} en R_{16} numeriek te benaderen door middel van een python script met matrix berekeningen, te vinden in hoofdstuk C1. In figuur 14 zijn de resultaten van hiervan te zien. De gebruikte code is terug te vinden in listing D1.



Figuur 14: Benadering van weerstandswaarde R_{15} en R_{16} .

In de twee grafieken van figuur 14 is op de y-as het bereik van waarde V_{diff} in het domein van potentiometer R_{pot1} , op de x-as, te zien. De gewenste differentiaalspanning van een gekalibreerd systeem ligt op 1 mV, dit zal verder toegelicht worden in hoofdstuk 5.2.2. Een weerstandswaarde van 2 kΩ is in beide scenario's voldoende om V_{diff} te kalibreren naar 1 mV.

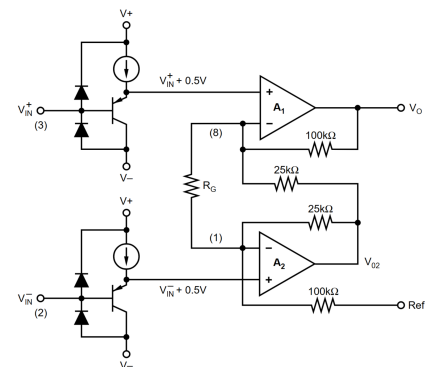
5.2.2 Stap 2: Versterking 1

Het signaal van de loadmeter wordt uitgelezen met de ADC van de DOBOT. De 12-bits ADC kan een spanning van 0 V tot 3.3 V omzetten naar een digitale waarde met een resolutie van 4096 stappen. Dit geeft een ingangsresolutie met stappen van $\frac{3.3}{2^{12}} = 0.81\text{mV}$. Zonder versterking van het signaal uit de brug van Wheatstone betekent dat er ongeveer 15 stappen van de 4096 gebruikt worden. Dit gecombineerd met de ruis op het uitgangssignaal resulteert in onbetrouwbare en onnauwkeurige meetwaarden.

Om het maximale van de resolutie te gebruiken dient het uitgangssignaal van de brug van Wheatstone versterkt te worden van 0 V tot 3.3 V. Hiervoor zijn er diverse schakelingen mogelijk die gebruik maken van op-amps. Voor de eerste versterkingsstap is er gekozen voor een instrumentatieversterker (in-amp) bestaande uit twee inwendige op-amps. De schakeling voor deze in-amp is te zien in figuur 15. Deze in-amp wordt niet met losse op-amps gerealiseerd maar wordt aangeschaft als een geïntegreerde schakeling. De keuze ligt hiervoor bij de INA122 [10].

De INA122 gebruikt een ingangsbuffer en twee interne op-amps om het verschil in spanning aan de ingangen te versterken. Door de hoge ingangsimpedantie zorgt de ingangsbuffer ervoor dat de volledige spanning V_{diff} overgedragen wordt aan de versterker. Daarnaast is er onder andere voor een instrumentatieversterker gekozen voor de lage input drift, input offset en een hoge common-mode rejection ratio (CMRR).

In het vorige hoofdstuk is er aangenomen dat de gewenste differentiaalspanning 1 mV is, dit komt door de input offset spanning van de INA122. De input offset is de differentiaalspanning die nodig is over de punten V_{in}^+ en V_{in}^- om de in-amp actief te stellen. Bij de INA122 ligt deze waarde op 250 μV . Met een ruime marge wordt er een differentiaalspanning van 1 mV vastgesteld.



Figuur 15: Het inwendige circuit van de INA122.[10]

Uit de datasheet van de INA122[10] wordt vergelijking (8) voor de uitgangsspanning vastgesteld.

$$V_{out1} = (V_{in1}^+ - V_{in1}^-)G_1 + V_{ref1} \quad , \text{ met } G_1 = \left(5 + \frac{200 \text{ k}\Omega}{R_G}\right) \quad (8)$$

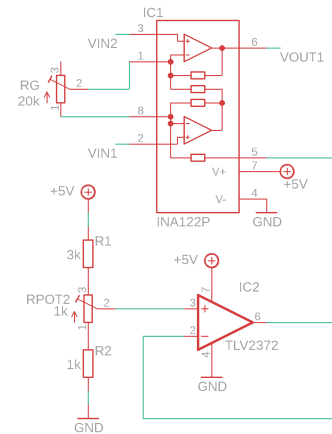
Het voordeel van het gebruiken van twee versterkingsstappen is dat de gain per stap laag kan blijven. Dit heeft positieve gevolgen op meerdere aspecten zoals insteltijd of de power supply rejection ratio (PSRR) van de op-amp. Voor de eerste stap wordt er een versterkingsfactor niet groter dan 100 aangehouden. De versterkingsfactor wordt berekend met behulp van de algemene vergelijking (9), geeft $G_1 = \frac{0.6 \text{ V}}{0.009 \text{ V}} \approx 66.7$. Daaruit volgt $R_G \approx 3243 \Omega$.

$$G = \frac{V_{out1}^+ - V_{out1}^-}{V_{in1}^+ - V_{in1}^-} \quad (9)$$

Daarbij is het ook gewenst om de uitgangsspanning zo ver mogelijk van de $V+$ en $V-$ (3.3V en 0V) lijnen af te houden. Dicht op deze lijnen verzadigd het signaal en is de versterking niet lineair. Om dit tegen te gaan zorgen de eerder genoemde ingangsbuffers voor nog een functionaliteit; rail-to-rail input en output. Dit maakt het mogelijk om met ingangs- en uitgangsspanningen tot 0.1V van $V+$ en $V-$ te werken. Met de bovenstaande gegevens wordt er een uitgangsspanningsbereik van 1.2V tot 1.8V vastgesteld.

Om deze uitgangsspanningen te kunnen bereiken moet V_{ref1} , pin 5 op figuur 16, op het juiste referentieniveau ingesteld worden. Om dit te realiseren is er een circuit ontworpen bestaande uit één op-amp en drie weerstanden, waaronder één potentiometer. De op-amp dient als een buffer, hiermee worden de impedanties voor en na de buffer gescheiden. Dit zorgt ervoor dat de weerstanden in het instelcircuit geen invloed hebben op de versterkingsfactor van de in-amp en dat de 5 V bron niet belast wordt door het circuit na de buffer. V_{ref1} kan vervolgens ingesteld worden door middel van potentiometer R_{pot2} . $V_{ref1} = 1.2 \text{ V}$ (C7).

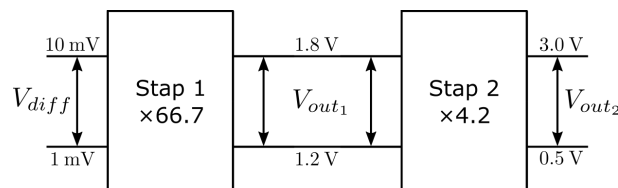
De gebruikte op-amp is de general-purpose op-amp TLV2372 [11].



Figuur 16: Circuit om V_{ref1} mee in te stellen.

5.2.3 Stap 3: Versterking 2

De laatste versterkingsstap dient het uitgangssignaal in het gewenste spanningsbereik te krijgen. Dit bereik wordt vastgesteld met een minimum en maximum van 0.5 V en 3 V. Om opnieuw de lineariteit van de versterking te maximaliseren wordt het minimum en maximum respectievelijk niet op 0 V en 3.3 V gesteld. Alle deelspanningen en versterkingen zijn in figuur 17 weergegeven.

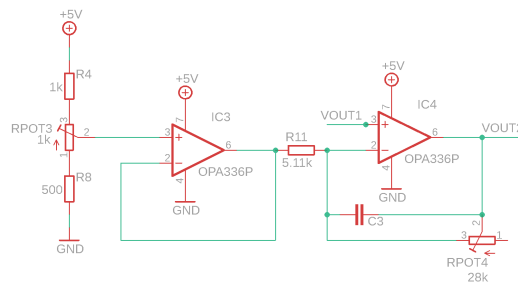


Figuur 17: Blokdiagram van alle versterkingsstappen.

Stap 2 maakt gebruik van een inverterende versterker bestaande uit één buffer- en één versterker op-amp. Uitgangsspanning V_{out2} kan vervolgens met vergelijking (10) beschreven worden.

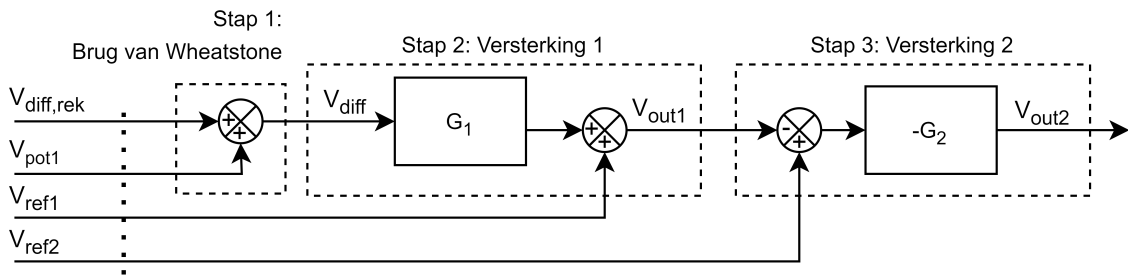
$$V_{out2} = -G_2(V_{ref2} - V_{out1}) \quad , \text{ met } G_2 = \frac{R_{pot4}}{R_{11}} = \frac{R_{pot4}}{5.11 \text{ k}\Omega} \quad (10)$$

Nogmaals wordt vergelijking (9) gebruikt om de versterkingsfactor van stap 2 te berekenen, dat geeft $G_2 = \frac{2.5 \text{ V}}{0.6 \text{ V}} \approx 4.2$. Daaruit volgt dat $R_{pot4} \approx 21.5 \text{ k}\Omega$. De laatste stap is om V_{ref2} op het juiste spanningsniveau te krijgen. Hiervoor wordt dezelfde schakeling van stap 1 gebruikt, echter hebben de weerstanden andere waarden om het instelbereik aan te passen. $V_{ref2} = 1.085 \text{ V}$ (C8). De schakeling van de tweede versterkingsstap is te zien in figuur 18.



Figuur 18: Deelschakeling van versterkingsstap 2.

In bijlage C1 is beschreven hoe de schakeling gekalibreerd kan worden. In figuur 19 is het circuit weergegeven met behulp van een blokschema, het verloop van het signaal is hier duidelijk in te zien.



Figuur 19: Het elektrisch circuit weergegeven in een blokdiagram.

De totale formule voor V_{out2} is weergegeven in vergelijking (11).

$$V_{out2} = -G_2(V_{ref2} - ((V_{in}^+ - V_{in}^-)G_1 + V_{ref1})) \quad (11)$$

5.2.4 Hardware lowpass-filters

Het uitgangssignaal van de loadmeter wordt gemeten met een oscilloscoop om te meten hoe groot de spanningsschommelingen zijn. Indien deze oscillaties significant groot zijn, zal het signaal gefilterd moeten worden. Aan de linkerkant van figuren A7, A8 en A9 zijn de oscillaties te zien van de respectievelijke signalen V_{out1} , V_{out2} en V_{cc} .

De metingen in de figuren zijn uitgevoerd met een wisselstroom koppeling. Dat betekent dat alleen de spanningsschommelingen in het signaal weergegeven worden op de scope, het DC-element van het signaal wordt hierbij weggelaten. Het V_{cc} voedingssignaal wordt gestabiliseerd met een bypass condensator. Het filteren van de uitgangssignalen gebeurt op basis van een zogeheten actieve laagdoorlaatfilter. In dit hoofdstuk wordt alleen het ontwerp van de laagdoorlaatfilter van signaal V_{out2} behandeld.

De schakeling van figuur 18 is hiervoor uitgebreid met een condensator, C3. De kantelfrequentie wordt berekend met vergelijking (C9). De gewenste kantelfrequentie is niet gespecificeerd in een eis, daarom wordt er een algemene schatting gemaakt om de kantelfrequentie op 50 Hz te nemen. Hiermee wordt ervoor gezorgd dat wisselspanningen met een frequentie boven de 50 Hz niet aan de uitgang komen. Een kantelfrequentie van 50 Hz geeft een minimale condensator waarde van 11 nF.

5.2.5 Meetpunten van de schakeling

Om fouten in de schakeling eenvoudig op te kunnen sporen, kan het elektrisch potentiaal van elk zogeheten net gemeten worden. Een net wordt gedefinieerd als een connectiepunt in de schakeling. elk groene ononderbroken lijn in figuur A4 is een net. In figuur A5 zijn de meetpunten aangegeven met een ruit.

De spanningen van deze punten kunnen nu vergeleken worden met de overeenkomende spanningen in de simulatie die met LTspice wordt uitgevoerd. Als de spanningen van deze punten afwijken van de simulatie, kan dat betekenen dat er een fout is in de schakeling of dat de schakeling niet correct is gekalibreerd.

5.3 Software Ontwerp

Naast een relatief snelle elektrisch laagdoorlaatfilter wordt de output van de loadmeter ook softwarematig gefilterd. Dit gebeurt in Unity na het opvragen van de waarden van de ADC. Om een softwarefilter te kunnen implementeren is er een kort onderzoek gedaan naar algoritmen die gebruikt kunnen worden.

Hieronder is er een lijst met drie algoritmen die uit het onderzoek zijn gekomen.

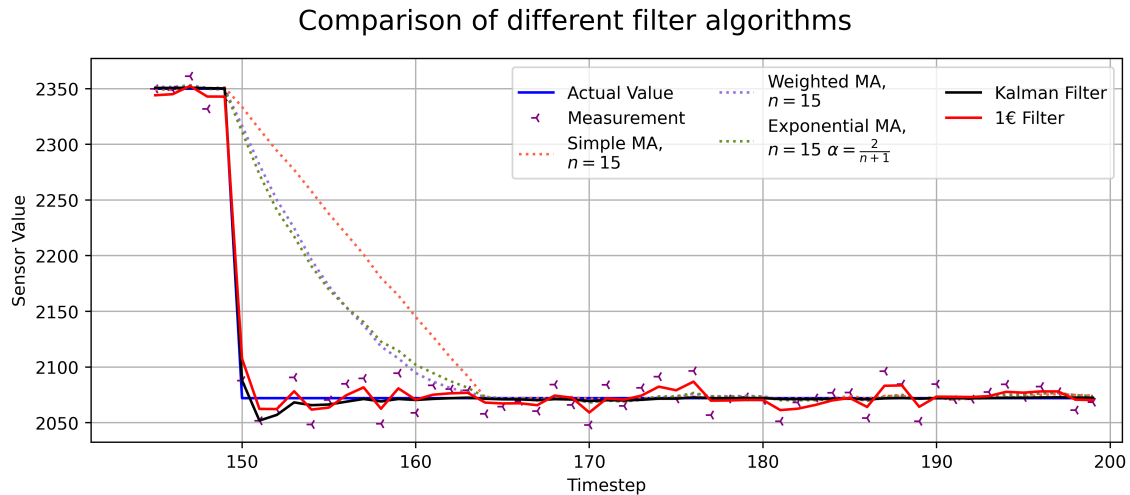
- **Het voortschrijdend gemiddelde:** het voortschrijdend gemiddelde, moving average, kijkt naar het gemiddelde van n aantal opeenvolgende meetpunten. Dit kan uitgebreid worden met twee variaties: het gewogen en exponentieel voortschrijdend gemiddelde.
- **Kalman filter [2]:** het kalman filter is een algoritme dat op basis afwijkingen van voorspellingen en meetwaarden, snel een toestand in kan schatten. Het voordeel van het kalman filter is dat deze een geheugen werking heeft waarbij de nieuwe voorspellingen voortbouwen op voorgaande meetwaarden.
- **1€ filter [1]:** deze filter is ontwikkeld om signalen in interactieve systemen die veel ruis bevatten te stabiliseren. Het voordeel is dat deze filter zeer snel reageert op veranderingen in een signaal. De werking is beschreven in een paper [12](Casiez et al., 2012).

De bovenstaande keuzes zijn gebaseerd op adaptief filtersnelheid en complexiteit van de implementatie. De algoritmen zijn getest en vergeleken met behulp van een python script. Uitgebreide definities van de gebruikte algoritmen kunnen in bijlage C gevonden worden. De resultaten van de test worden hieronder kort behandeld.

5.3.1 Filter algoritme selectie

Om het juiste filteralgoritme te kunnen kiezen is het belangrijk om de karakteristieken van het signaal duidelijk te krijgen. De sensor zal voornamelijk belast worden in stappen van 50g. Dat betekent dat de last stabiel op één waarde blijft tot er een gewicht bijgevoegd wordt. Dit gedrag wordt nagebootst met een python script zodat de prestaties van de verschillende algoritmen vergeleken kunnen worden.

Als eerst worden de algoritmen getest op de snelheid waarop de gefilterde waarde de werkelijke waarde bereikt. Om dit te meten wordt er in de test een verandering aangebracht aan de daadwerkelijke waarde. Vervolgens wordt er gekeken naar hoe de verschillende algoritmen hierop reageren, dit wordt de responsie genoemd. Figuur 20 toont de resultaten van deze test.

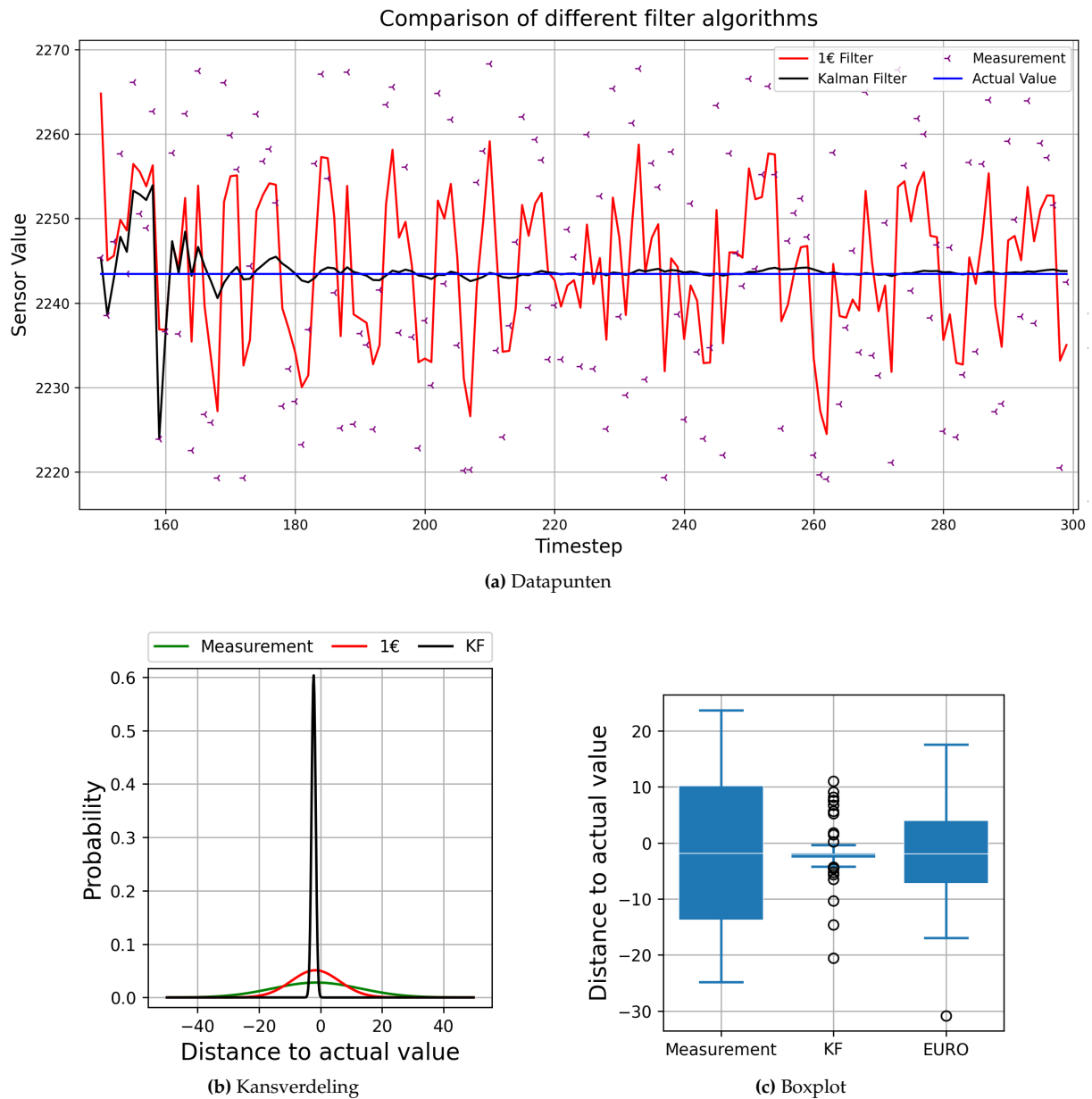


Figuur 20: stapresponsie van diverse filter algoritmen.

Uit de resultaten is te zien dat de drie vormen van het voortschrijdend gemiddelde het langzaamst reageren op verandering. Alle drie vormen van het voortschrijdend gemiddelde in figuur 20 zijn berekend met een periode van 15 metingen. Deze waarde is een middenweg tussen stabiliteit en snelheid. Omdat de deze drie algoritmen in vergelijking met de rest van de algoritme langzamer reageren, is er besloten om deze uit de selectie te halen.

Om een selectie te maken uit de resterende algoritmen is er gekeken naar de prestatie van de filters bij een langdurige meting van een statische last. Hiervoor wordt de variabele daadwerkelijke waarde voor een langdurige tijd constant gehouden. De resultaten van deze testen zijn te zien in tabel 5, één van deze resultaten is ook uitgezet in figuur 21a.

In de grafiek zijn er in totaal vier datalijnen te zien: de daadwerkelijke waarde (blauw), de twee gefilterde waarde (Kalman filter, zwart en 1 σ filter, rood) en de meetpunten (paars). Er is duidelijk te zien dat de meetpunten een onzekerheid hebben van ± 25 sensorstappen. Dit komt overeen met de uitgangsfuctuaties die gemeten zijn met een oscilloscoop in figuur A8b. Deze onzekerheid heeft meerdere oorzaken. Een van de oorzaken is de ontwerpkeuzes die zijn gemaakt bij het kiezen van de componenten voor de filters. Een ander belangrijke oorzaak is dat het circuit gevoelig is voor zowel externe signalen zoals de magnetische velden die gegenereerd worden in de omgeving, als voor fluctuaties door het circuit zelf. Denk hierbij aan de warmte die gegenereerd wordt door de weerstanden en versterkers. Dit heeft invloed op de prestatie van de componenten.



Figuur 21: Stabiliteitstest van diverse filter algoritmen.

Tabel 5: Resultaten van vergelijkingstest van diverse filter algoritmen.

Meting	μ Toevallige Fout			σ Standaarddeviatie van het verschil		
	Meetpunten	1€	KF	Meetpunten	1€	KF
1	1.943	1.840	1.754	14.441	9.009	3.685
2	1.238	1.022	2.213	14.449	8.4462	3.055
3	0.005	-0.164	0.657	14.756	8.945	1.614
4	-0.445	-0.359	-1.932	14.242	8.102	2.325
5	0.836	0.908	0.3801	14.562	8.4644	2.193

Bij een meting op een willekeurig moment is het belangrijk dat de standaarddeviatie van het verschil tussen de gefilterde waarde en de daadwerkelijke waarde zo klein mogelijk is. Daarom is er een keuze gemaakt om gebruik te maken van het kalman filter, ondanks dat deze een grotere toevallige fout heeft.

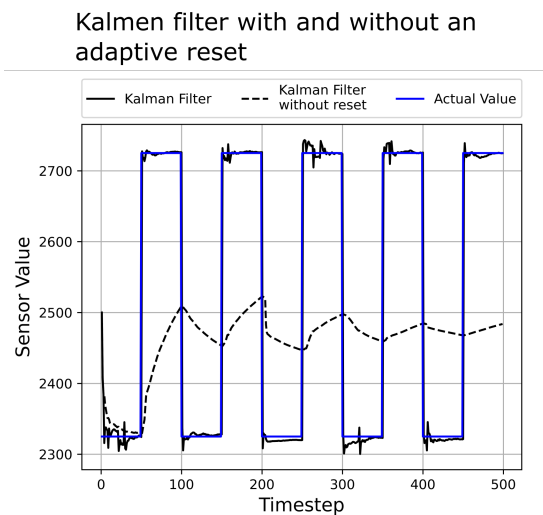
5.3.2 De adaptieve reset van het kalman filter

Het geïmplementeerde kalman filter is uitgebreid met een adaptieve reset, zodat het mogelijk is om veranderingen in het signaal met een hoge snelheid te kunnen verwerken naar de uitgang van het filter. Zonder deze reset zal de kalman gain de 0 benaderen, dit resulteert dat de nieuwe voorspellingen (vergelijking (C13)) steeds minder afhangen van de nieuwe metingen. De gevolgen hiervan zijn weergegeven in figuur 22.

Na een hoeveelheid tijd zal het kalman filter zonder reset stabiliseren op een waarde tussen alle voorgaande metingen.

Om dit te voorkomen wordt er bij het updaten van de filter gekeken naar de afwijking van de vorige voorspelling en het nieuwe meetpunt. Als deze afwijking groter is dan x aantal standaarddeviaties van de afgelopen y meetwaarden sinds de vorige reset, wordt de kalman gain op 1 gezet en de nieuwe voorspelling geplaatst op de nieuwe meetwaarde. Na de reset zet de filter zijn normale werking toe en nadert de uitgang de daadwerkelijke waarde.

De gefilterde waarde is te zien in de gebruikersinterface van de Unity simulate. Daarnaast wordt de waarde ook gebruikt om de krachten in het model te bereken door middel van een krachtenanalyse. Dit wordt in de komende hoofdstukken verder behandeld.



Figuur 22: Kalman filter met en zonder adaptieve reset.

6 De digital twin simulatie van de DOBOT

Een digital twin is een digitale representatie van een fysiek systeem. Het maakt het mogelijk om het systeem te kunnen bestuderen en een nauwkeurig beeld te geven over onder andere de onderhoudsstaat van het systeem. Dit wordt gedaan door sensoren op kritieke posities te plaatsen en de sensorwaarden real-time uit te lezen. Deze waarden worden samen met een log van voorgaande data geanalyseerd en teruggekoppeld aan het systeem.

Dit hoofdstuk beschrijft hoe de digitale representatie van de DOBOT is opgezet. Daarbij wordt ook de gebruikersinterface tussen systeem en gebruiker toegelicht. De digitale representatie is ontwikkeld in Unity met behulp van diverse programma's om objecten te converteren en aan te passen.

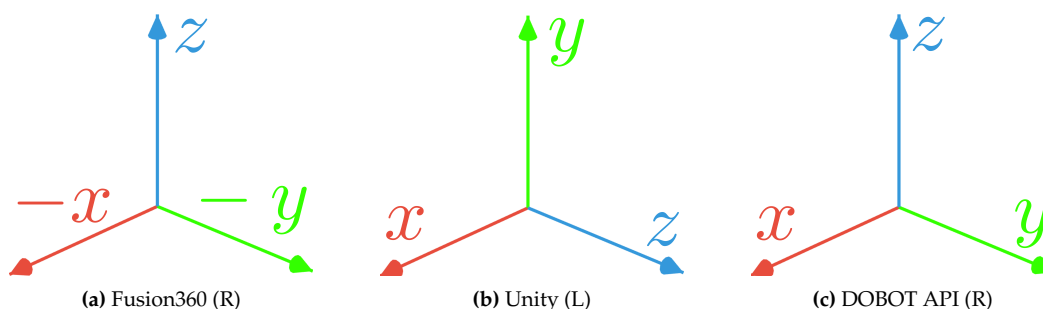
6.1 De opbouw van de virtuele arm

Door middel van meegeleverde CAD tekeningen is het model al digitaal beschikbaar in een CAD formaat. Dit model bestaat uit 86 losse componenten. Het aansturen van al deze losse onderdelen zou niet alleen veel ontwikkeltijd kosten maar zou ook veel rekenkracht eisen van de simulatie. Veel componenten worden daarom gegroepeerd tot één sub-assembly aan de hand van de identieke bewegingen die ze maken. In figuren A10 en A11 zijn deze sub-assemblies weergegeven. In totaal bestaat de virtuele DOBOT uit tien onderdelen, hierna te noemen als basisonderdelen.

De basisonderdelen worden door middel van meerdere stappen geconverteerd tot een *Surface Mesh STL (SMSTL)*, dit wordt later in hoofdstuk 6.3.1 in detail verder behandeld. Deze SMSTL wordt gebruikt om het object te importeren in Unity. Het 3D object springt tijdens de conversies door een aantal verschillende coördinatenstelsels. Het is belangrijk om deze verschillen bij te houden, zodat er hiermee bij het aansturen rekening gehouden kan worden met eventuele transformatiematrixen die toegepast moeten worden.

6.1.1 De verschillende coördinatenstelsels

De drie programma's die gebruikt worden tijdens de ontwikkelingen gebruiken elk een ander coördinaatstelsel, er dient hier rekening mee gehouden te worden. Deze verschillen in coördinaatstelsels zijn in figuur 23 weergegeven.



Figuur 23: De coördinatenstelsels van de drie conversieprogramma's.

De coördinatenstelsel van de DOBOT API wordt als hoofdstelsel beschouwd. Wanneer er gewerkt wordt in Fusion[13] of Unity zal er dus een transformatie plaatsvinden. De bijhorende transformatiematrixen zijn te vinden in (C14), (C15) en (C16).

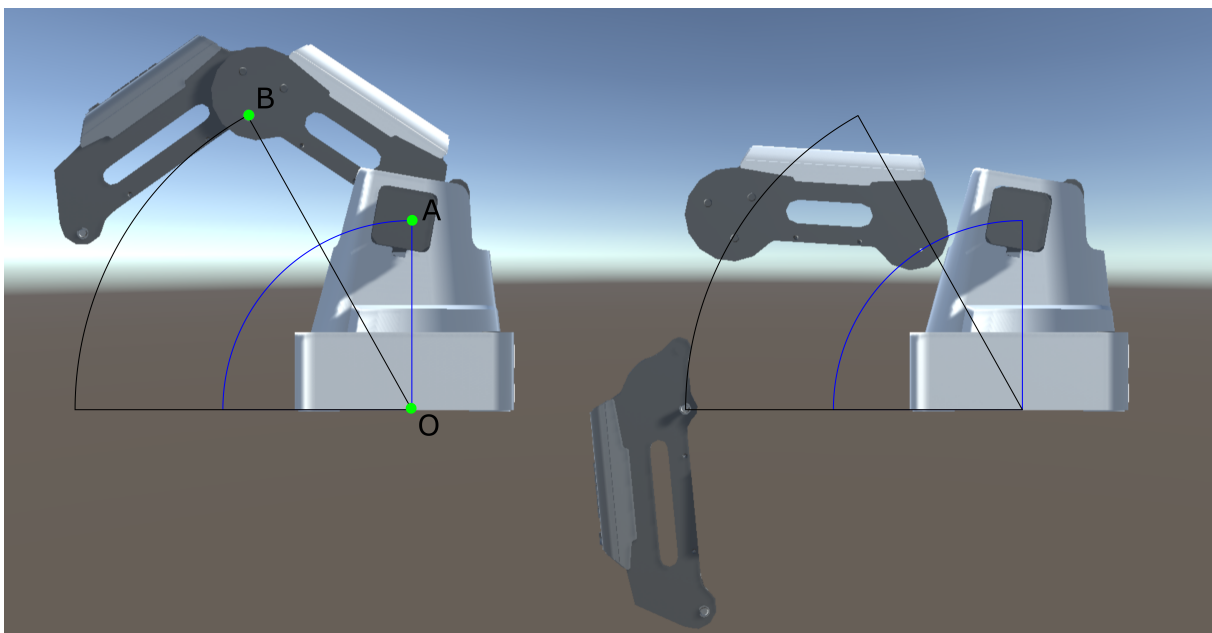
6.2 Meebewegen met de werkelijke arm

De end-effector van de DOBOT kan in drie graden van vrijheid bewegen, deze bewegingsassen zijn te zien in figuur 3. De bewegingsassen worden aangestuurd door middel van stappenmotoren. Door het gebruik van stappenmotoren worden de posities van de motoren, en daarmee de armen, nauwkeurig bijgehouden. Hierdoor kunnen posities direct opgevraagd worden zonder dat deze eerst gefilterd moeten worden, deze waarden worden vervolgens doorgevoerd naar de simulatie.

6.2.1 Bewegingsbeperkingen toepassen op het virtuele ontwerp

Om ervoor te zorgen dat de virtuele DOBOT als één geheel beweegt moeten de basisonderdelen in Unity geassembleerd worden met een aantal bewegingsbeperkingen. In dit deelhoofdstuk wordt de problematiek van dit proces samen met de oplossing behandeld. Als voorbeeld wordt er later in het hoofdstuk onderdeel **Rear Connector 1** bevestigd aan onderdeel **Foot**. Benamingen van de onderdelen zijn te vinden in tabel A5.

De basisonderdelen worden in Unity in elkaar genest zodat de transformaties in het bovenliggend object meegaan met de onderliggende objecten. Dit zorgt ervoor dat de rotatie in de Z-as alleen in onderdeel **Foot** plaats hoeft te nemen. In figuur A12 is de hiërarchie van objecten in Unity te zien. Voor rotaties van een bovenliggend object om de Y-as geeft dit echter een probleem, te zien in figuur 24, omdat de onderliggende objecten roteren om de oorsprong(O) en niet om hun respectievelijke rotatiepunten (A&B). Dit heeft als gevolg dat de contactpunten van twee objecten elkaar niet meer raken.

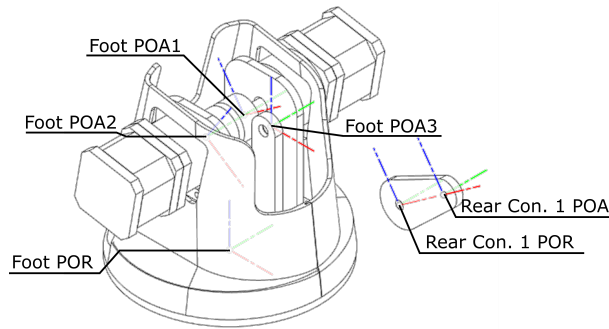


Figuur 24: de DOBOT in zijn oorspronkelijke stand, beide armen 45° (links). De DOBOT met een Front Arm rotatie van 60° en een Rear Arm rotatie van 30° (rechts).

Om dit probleem op te lossen moet er alleen een verplaatsing plaatsvinden waarmee de contactpunten van beide objecten in dezelfde positie komen. Het verplaatsen van objecten levert echter nog een probleem met zich mee. Het is namelijk niet mogelijk om de absolute afstanden van de contactpunten op te vragen in Unity.

Contactpunten Voordat de verplaatsingen verder worden behandeld is het belangrijk om de benamingen van de contactpunten duidelijk te krijgen. De basisobjecten bestaan elk uit twee soorten punten:

- **Point Of Rotation (POR):** Dit is het punt waar het basisonderdeel omheen roteert. Per basisonderdeel is er maar één POR gedefinieerd.
- **Point Of Attachment (POA):** Dit is het punt waar de POR van een ander basisonderdeel aan bevestigd wordt. Per basisonderdeel is er minimaal één POA gedefinieerd, dit kunnen er meerdere zijn voor meerdere onderdelen.



Figuur 25: Verduidelijkingen van de contactpunten.

Een contactpunt wordt gedefinieerd als een samenvoeging van de POR van onderdeel 1 en de POA van onderdeel 2. Voor de duidelijkheid is er in figuur 25 een voorbeeld gegeven van de contactpunten van onderdelen **Rear Connector 1** en **Foot**. In dit geval moet punt POR van **Rear Connector 1** verplaatsen naar punt POA 1 van **Foot**. **Rear Connector 1** en **Foot** worden respectievelijk gezien als child en parent met positievectoren $\vec{r}_{c_{POR}}$ en $\vec{r}_{p_{POA}}$. Een lijst van alle contactparen is te vinden in figuur A6.

Hiermee wordt de nieuwe positie van **Rear Connector 1** aangegeven met (12).

$$\vec{r}_{c_{nieuw}} = \vec{r}_c + (\vec{r}_{p_{POA}} - \vec{r}_{c_{POR}}) \quad (12)$$

Waarbij: \vec{r}_c = De absolute positievector van het middelpunt van de bounding box van child object.

$\vec{r}_{p_{POA}}$ = De absolute positievector van punt POA van parent object.

$\vec{r}_{c_{POR}}$ = De absolute positievector van punt POR van child object.

Zoals eerder aangegeven kunnen de absolute posities van de contactpunten niet direct opgevraagd worden, omdat deze in een lager object hiërarchie vallen, zie figuur A12. In plaats daarvan retourneert Unity bij aanvraag de relatieve positie vanaf het middelpunt van de bounding box van het hoofdobject. De nieuwe positie van de **Rear Connector 1** wordt aangegeven met (13). Een verduidelijking van deze translatie is te vinden in figuur A13.

$$\vec{r}_c = \vec{r}_p + (\vec{r}_{p_{POA}} - \vec{r}_{c_{POR}}) \quad (13)$$

Waarbij: \vec{r}_c = De absolute positievector van het middelpunt van de bounding box van child object.

\vec{r}_p = De absolute positievector van het middelpunt van de bounding box van parent object.

$\vec{r}_{p_{POA}}$ = De relatieve positievector van punt POA vanuit het middelpunt van parent object.

$\vec{r}_{c_{POR}}$ = De relatieve positievector van punt POR vanuit het middelpunt van child object.

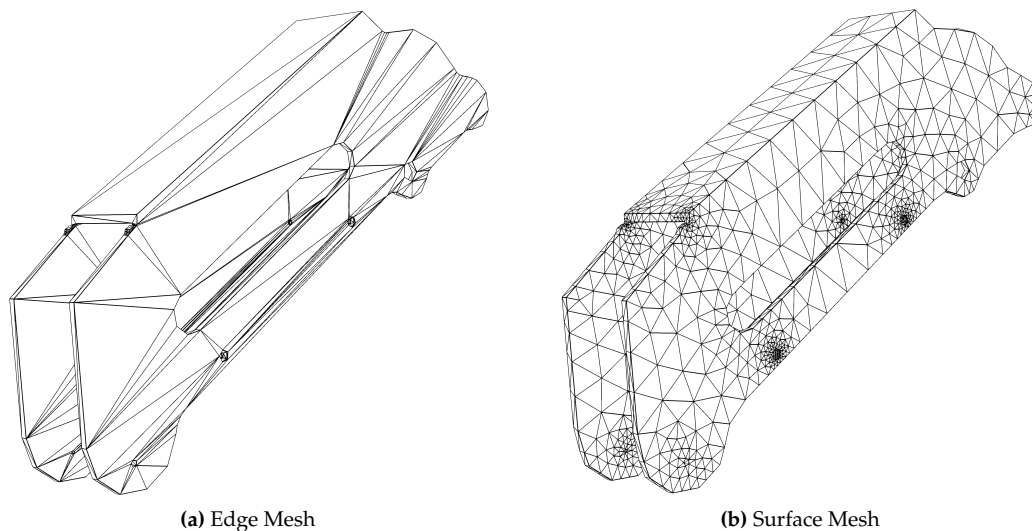
6.3 Visuele terugkoppeling van de krachtenanalyse.

Naast het meebewegen laat de digital twin ook de vervormingen zien die optreden bij een gemeten kracht aan de end-effector. De details van de krachtenanalyse wordt in hoofdstuk 7 uitgebreid behandeld. Dit deelhoofdstuk beschrijft hoe de output van deze analyse teruggekoppeld wordt naar Unity. Daarbij wordt het basisonderdeel **Front Arm** als voorbeeld gebruikt, de stappen zijn voor rest van de onderdelen identiek.

6.3.1 De verschillende oppervlaktegeometrieën van een object

Een digitaal 3D object is opgebouwd uit hoekpunten die samengevoegd worden tot 2D cellen die bij elkaar een 3D mesh vormen. Zoals eerder genoemd wordt er in eerste instantie gebruik gemaakt van een edge mesh om het object te visualiseren. Het doel van de edge mesh is om de contouren van het object met zo min mogelijk punten te beschrijven. Dit heeft als resultaat dat het object bestaat uit grote egale oppervlakten met strakke lijnen. Dit type mesh is niet ideaal om vervormingen in het object aan te tonen.

Vervormingen in het model worden aangebracht door de individuele hoekpunten te transleren met een vervormingsvector. Dit betekent dat de 2D cellen die aan een getransleerde hoekpunt vastzitten meebewegen met dit punt. Dit proces zal bij een edge mesh ernaar toe leiden dat er grote oppervlakte meebewegen. Dit is niet gewenst en wordt de edge mesh verfijnd tot een surface mesh, zodat niet de contour maar het oppervlak van een object prioriteit krijgt. Beide geometrieën voor de **Front Arm** zijn weergegeven in figuur 26.



Figuur 26: Soorten oppervlaktegeometrieën.

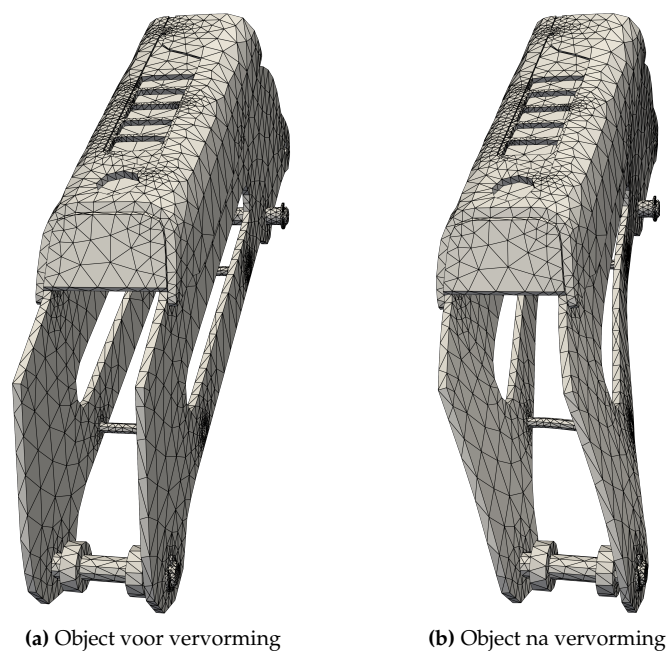
De krachtenanalyse gebruikt tijdens de FEM berekeningen en baseert de uitvoer ook op een gegeneerde surface mesh. In het volgende deelhoofdstuk wordt er beschreven hoe de surface mesh dat gebruikt is in Unity, gekoppeld wordt aan de uitvoer van de krachtenanalyse.

6.3.2 Het koppelen van hoekpunten van de verschillende geometrieën en de uitvoerwaarden

Het uiteindelijk gebruikte surface mesh wordt geconverteerd van een CAD tekening in een aantal stappen. Deze stappen zijn weergegeven in het linker helft van figuur A14. Tijdens deze conversie gaat het object door meerdere bestandsformaten. Dit heeft als gevolg dat er bepaalde tekenelementen verloren gaan in de conversie. Zo wordt het model in stap 4 van figuur A14 omgezet van een .vol (Net-Gen) formaat naar een .stl (SMSTL) formaat. Het NetGen formaat beschrijft een 3D object met zowel 1D-elementen (lijnen), 2D-elementen (driehoeken) en 3D element (viervlakken). Het STL formaat ondersteunt daarentegen alleen 2D elementen en zullen de andere elementen samen met hun betreffende hoekpunten vervallen tijdens de conversie.

Het .vol formaat wordt gebruikt door de krachtenanalyse en zullen de hoekpunten die wegvallen in het Unity model aanwezig zijn in de uitvoer van de krachtenanalyse. Hierdoor vervalt het verband tussen het object in Unity en de krachtenanalyse en kan de verplaatsingsvector niet één op één overgenomen worden, het object in Unity heeft immers minder hoekpunten dan het object in de krachtenanalyse. Om dit verband weer terug te krijgen wordt de index van elk hoekpunt in het Unity-model gekoppeld met de index van het dichtstbijzijnde hoekpunt in het krachtenanalyse-model. Dit koppelingsproces gebeurt in twee stappen, te vinden in het rechter helft van figuur A14.

Omdat beide modellen niet veranderen hoeft het conversie- en koppelingsproces maar één keer tijdens ontwikkeling uitgevoerd te worden. De koppelingen worden opgeslagen in een .csv tabel, zodat deze snel en eenvoudig uitgelezen kunnen worden tijdens het uitvoeren van de simulatie. Het uitvoerbestand van de krachtenanalyse bestaat uit een tabel met twee hoofdkolommen: index van hoekpunt (krachtenanalyse-model) en de verplaatsingsvector. Dit uitvoerbestand wordt in Unity uitgelezen en worden de betreffende hoekpunten, met behulp van de koppelingen, verplaatst met de respectievelijke verplaatsingsvectoren.



Figuur 27: Het deformeren van een object.

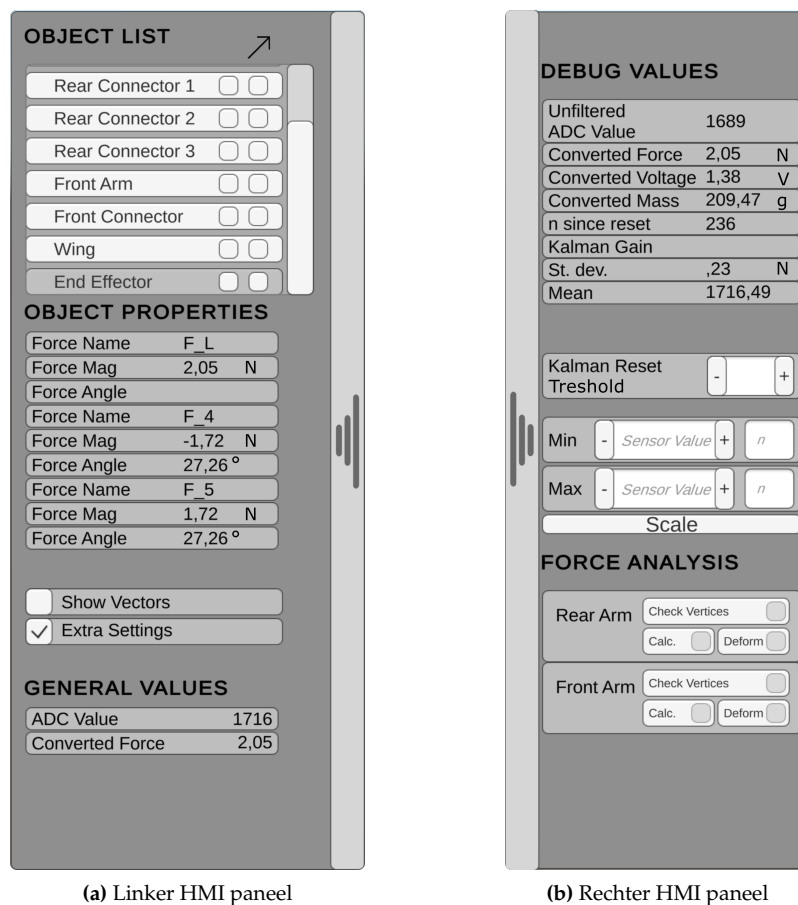
6.4 Interactie met de software

Voor de gebruiker is er een interface ontwikkeld waarmee de digital twin aangestuurd kan worden. Tevens is het ook mogelijk om hierop de beschikbare gegevens uit te lezen. Bij het ontwerpen zijn de powerpoint slides van het vak Human Machine Interaction[14] aangehouden. In figuren 28 en A15 is de gebruikersinterface te zien. Dit hoofdstuk behandelt de werking en detaillering van de verschillende secties in de interface.

De interface is ingedeeld in een linker en rechter helft. De linkerhelft bevat secties: Object List, Object Properties en General Values. Object list is een set met knopelementen waarmee basisonderdelen geselecteerd mee kunnen worden. Na selectie verschijnt er in de sectie Object Properties informatie over de krachten die in het geselecteerde onderdeel aanwezig zijn. Sectie General Values toont de algemene waarden, in de huidige status bestaat dit enkel uit de gefilterde ADC waarde en de geconverteerde kracht die gemeten wordt met de loadmeter. Ook bevat het linker paneel een selectievak om de extra opties in het rechter paneel te tonen. Beide panelen kunnen tevens ook getoond en verborgen worden door te klikken op het lichtgrijze deel.

Het rechter paneel bevat extra opties waarmee de digital twin aangestuurd kan worden. Ten eerste worden de gegevens die te maken hebben met de werking van het programma getoond. Dit maakt het makkelijk om eventuele fouten op te sporen. Vervolgens kan de reset grenslijn van het kalman filter aangepast worden in het tekstvlak **Kalman Reset Threshold**. De derde set met opties bestaat uit vier tekstvelden en een knop. Deze set dient voor de softwarekalibratie van de loadmeter. Door de minimale en maximale waarde van de sensor bij de respectievelijke aantal gewichten aan te geven, kan de sensor geijkt worden.

De laatste set met opties hebben te maken met aanbrengen van vervormingen in een object door middel van de uitgevoerde FEM analyse. Hoofdstuk 7 beschrijft de werking verder in detail. Voor er een vervorming aangetoond kan worden moet er eerst een aantal controles uitgevoerd worden. Deze controleren of de, in hoofdstuk 6.3.2, genoemde koppelingen correct zijn. Als deze controle succesvol uitgevoerd is, kan de FEM analyse uitgevoerd worden met **Calc.** en de vervormingen aangetoond worden met **Deform.**



(a) Linker HMI paneel

(b) Rechter HMI paneel

Figuur 28: Beide panelen van de gebruikersinterface.

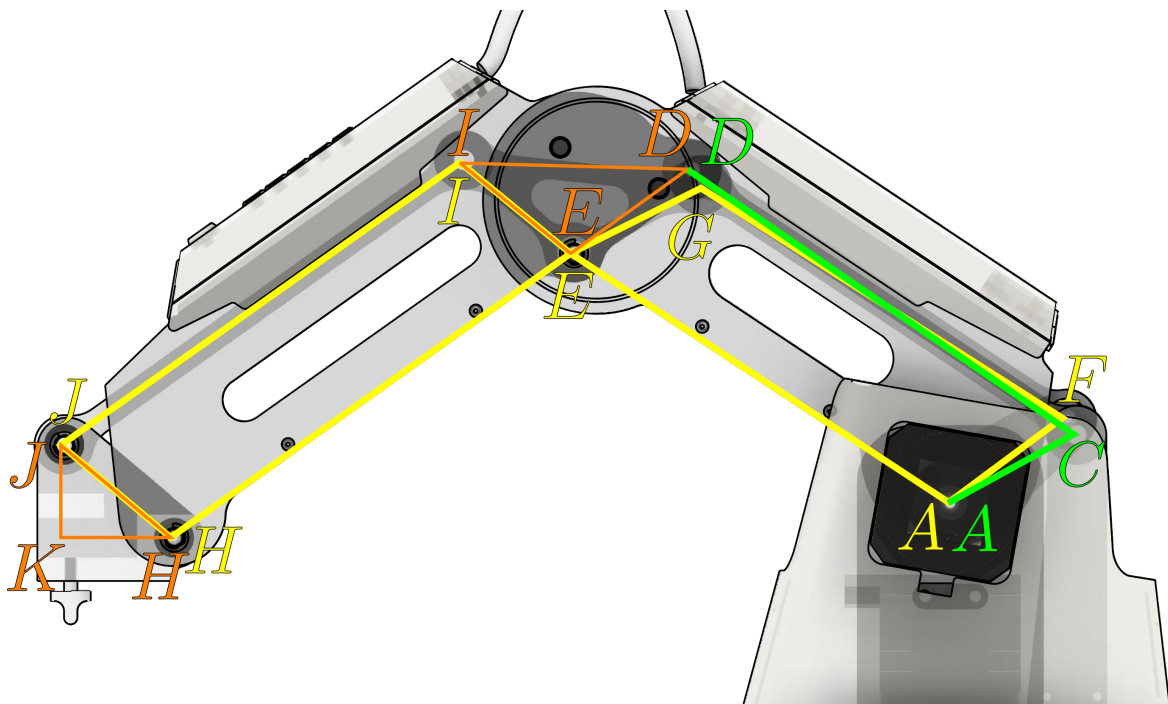
7 De krachtenanalyse van een last aan de DOBOT

In het vorige hoofdstuk is behandeld hoe de virtuele DOBOT meebeweegt met de werkelijke machine. Om de digital twin een stap verder te brengen wordt er een krachtenanalyse uitgevoerd met ingangssignalen vanuit de DOBOT. Hiervoor wordt er gebruikt gemaakt van de FEM analyse software PrePoMax [15] en de onderliggende CalculiX [16]. Dit hoofdstuk behandelt de werking van dit proces.

7.1 Het krachtenmodel

Het krachtenmodel van een werktuig of machine is vaak complex, waardoor het niet altijd eenvoudig is om een FEM analyse uit te voeren. De actieve krachten op een werktuig moeten immers bekend zijn om als invoer gebruikt te worden in de analyse. Dit zorgt ervoor dat de kwaliteit van de uitkomsten van de FEM analyse afhankelijk zijn van de kwaliteit van het krachtenmodel.

Het is daarom belangrijk om een middenweg te vinden in de complexiteit van het model. Een te simpel model heeft als gevolg dat de krachten niet representatief zijn van de krachten die in de werkelijke constructie voorkomen. Een te complex model daarentegen kan zorgen voor moeilijke instelparameters en krachten in deelcomponenten die niet afgeleid kunnen worden. Dit deelhoofdstuk behandelt het gekozen krachtenmodel dat is opgezet voor de DOBOT.



Figuur 29: De 10 staaf/balk elementen.

In figuur 29 zijn de bevestigingselementen van de 10 basisonderdelen schematisch weergegeven. De bevestigingselementen worden verdeeld in twee groepen; staven en balken. Onderdelen **rear connector 2**, **rear connector 3** en **front connector** vallen onder staven, deze kunnen geen schuifspanningen en momenten dragen. Staven worden enkel belast op trek- en drukspanningen. De rest van de basisonderdelen worden beschouwd als balken die daarentegen wel belast kunnen worden met interne schuifspanningen en momenten.

Het krachtenmodel en de daarbij horende vrijlichaamsschema's van elk onderdeel zijn opgesteld in hoofdstuk C3. Ook worden de onbekende krachten vastgesteld met behulp van een matrix berekening. Om deze berekening uit te kunnen voeren is er eerst een set met lineaire vergelijkingen opgesteld. De individuele vergelijkingen worden aangegeven met een nummer na vergelijkingen (C17) tot (C24).

De matrix dat uit de set van vergelijkingen ontstaat is weergegeven in vergelijking (C25). Om de berekening uit te kunnen voeren moeten matrix (C25) en (C27) volledig ingevuld worden. De twee matrixen tellen in totaal zeven onbekenden ($\alpha, \beta, \gamma, \delta, \varepsilon, \zeta$ en F_L), dit geeft echter een probleem, omdat er maar drie invoervariabele uitgelezen worden van de DOBOT (α, β en F_L). De rest van de variabelen zijn aangegeven in de deeltabellen van tabel C1.

De resterende hoeken kunnen bepaald worden door te kijken naar de parallellogramconstructie van de DOBOT, in figuur 29 aangegeven met gele lijnen. Hier is te zien dat de hoeken in de vrijlichaamsschema's γ en δ overeenkomen met hoek α . En dat hoeken ε en ζ overeenkomen met hoek β . De hoeken waarop de krachten staan worden aangegeven in tabel C1b.

7.2 FEM analyse met behulp van Calculix

FEM analyses geven de mogelijkheid om de kritische plekken van complexe constructies aan te tonen. Zoals in hoofdstuk 6.3 beschreven is wordt de geometrie van een object verdeeld in kleine delen met een onderlinge afhankelijkheid aan elkaar. Zo kunnen de krachten in een volledige constructie snel berekend worden.

De initiële configuratie van de FEM analyse wordt gedaan in de gebruikersinterface van PrePoMax, na deze configuratie kunnen de berekeningen via een console commando aangestuurd worden. Een schermafbeelding van de initiële configuratie van onderdeel **Front Arm** is te zien in figuur A16. Hieronder is een lijst met de configuratiepunten die ingesteld dienen te worden.

- **Mesh:** Instellingen die met het model te maken hebben.
 - **Parts:** De individuele subonderdelen waar een onderdeel uit bestaat.
- **Materials:** Definities van materialen en hun eigenschappen die in het model voorkomen.
- **Sections:** Definities van combinaties van een of meer **Parts**. Dit wordt gebruikt om aan te geven dat bijvoorbeeld part 1 en 2 van het materiaal ABS plastic gemaakt is en part 3 en 4 van aluminium 6061.
- **Constraints:** Constraints definiëren hoe de individuele **parts** bevestigd zijn aan elkaar.
- **Steps:** Steps bevat instellingen die met krachten te maken hebben.
 - **Field outputs:** Welke gegevens de uitvoer van de krachtenanalyse bevat.
 - **Boundary Conditions:** Vaste punten van het object (Reactiekrachten).
 - **Loads:** De krachten dat het object ondervindt.

Eenmaal geconfigureerd wordt er een .inp bestand aangemaakt. Dit bestand bevat alle configuratie instelling die voorafgaand ingesteld zijn.

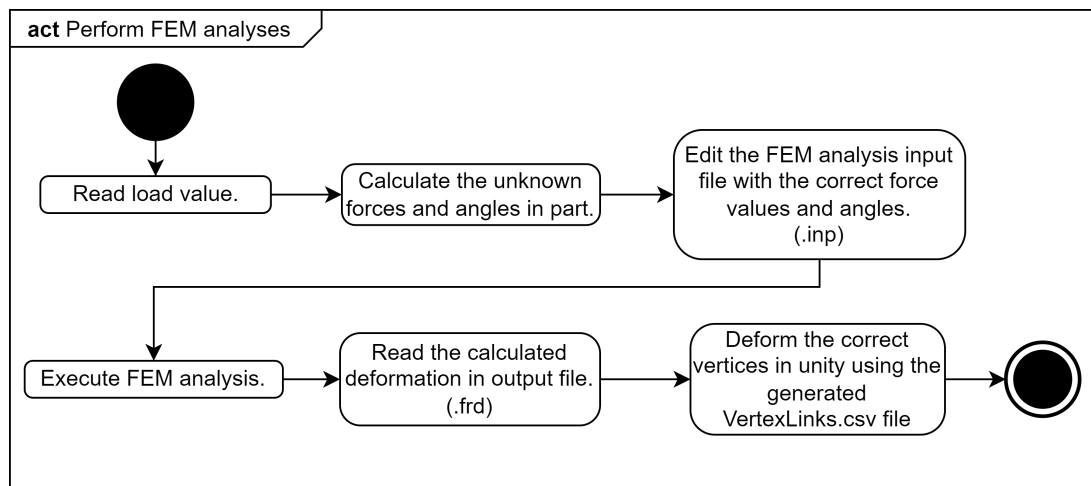
7.2.1 Automatisering van de analyse

In hoofdstuk 6.3 is beschreven hoe de oppervlaktegeometrie van de digital twin is opgebouwd. Deze geometrie wordt gecombineerd met de uitkomsten van de FEM analyse om vervormingen in het

object aan te brengen. Dit deelhoofdstuk beschrijft het automatiseren van het tweede gedeelte, de FEM analyse, van dit proces.

Zodra er via de gebruikersinterface van de digital twin een FEM analyse opgevraagd wordt, leest de DOBOT de drie invoervariabele uit (α , β en F_L). Vervolgens worden de resulterende krachten en de daarbij horende hoeken in alle objecten berekend. De respectievelijke krachten en hoeken van het te analyseren onderdeel kunnen vervolgens aangepast worden in het .inp bestand.

Dit proces is weergegeven in de activity diagram te vinden in figuur 30.



Figuur 30: De activity diagram van het uitvoeren van een FEM analyse

8 Testresultaten & evaluatie

Om de behaalde eisen aan te kunnen tonen worden er volgens het V-Model drie sets met testen opgesteld. Elk opeenvolgende set, test een ander integratieniveau van het ontwerp.

Om te beginnen worden de individuele functionaliteiten van de deelontwerpen loadmeter, digital twin en krachtenanalyse getest met unittesten. Zo kunnen de deelontwerpen onafhankelijk van een ander deelontwerp getest worden. Vervolgens wordt met integratietesten het raakvlak tussen twee deelsystemen getest. Dit zijn functionaliteiten zoals het sturen van de gemeten last naar de krachtenanalyse. Als laatste wordt de volledige functionaliteit getest met de systeemtesten. De systeemtesten kunnen gezien worden als acceptatietesten, waar alle gebruikerseisen getoetst worden.

Door de beperkte hoeveelheid tijd zijn alleen de systeemtesten gedocumenteerd. De unittesten en integratietesten zijn tijdens het ontwikkelproces uitgevoerd.

8.1 Systeemtesten

Volledige testplannen voor de systeemtesten zijn te vinden in bijlage hoofdstuk B1. Hier is te zien dat er meerdere eisen gecombineerd zijn in een testplan. Daarnaast is duidelijk aangegeven wat de invoer, uitvoer en slagingscriteria van de test is om de eisen te doen slagen.

In dit deelhoofdstuk worden de resultaten van de systeemtesten kort besproken en geëvalueerd wat dit betekend voor het eindresultaat. In tabel 6 zijn de resultaten van de systeemtesten te zien. Tabel 7 toont de individuele eisen die zijn behaald.

Systeemtest 1 - 0/1 eis behaald.

Deze systeemtest is **niet behaald**. De eis in deze test stelde dat de bewegingen van de DOBOT niet begrensd mogen worden. De gekozen conceptoplossing voor de functie end-effector belasten zijn vaste kalibratiegewichten. Deze hebben een lengte van 11 cm en raken het werkoppervlak dus als de end-effector zich lager dan 11 cm boven het werkoppervlak bevindt. Dit probleem kan opgelost worden door de DOBOT zo te positioneren aan de zijkant van een tafel, zodat de end-effector niet boven een tafel zweeft. In dit geval zal de eis en daarmee de test behaald zijn.

Systeemtest 2 - 3/3 eisen behaald.

Deze systeemtest is **volledig behaald**. In deze test worden de eisen getest die te maken hebben met de uiterste waarden van de loadmeter. De test is volledig behaald dat betekend dat de end-effector belast kan worden van 0 N tot 5 N. Daarnaast zorgt de elektrische schakeling ervoor dat deze uiterste krachtwaarden geregeld worden naar een uitgangsspanning van 0 V tot 3.3 V.

Systeemtest 3 - 2/2 eisen behaald.

Deze systeemtest is **volledig behaald**. Het ontwerp kan, na het iken van loadmeter, een last meten met een nauwkeurigheid van 0.5 N. Dit betekend dat de krachtenanalyse dat volgt een realistischer beeld geeft. Daarnaast verbruikt de schakeling niet meer dan 0.25 W.

Systeemtest 4 - 1/2 eisen behaald.

Deze systeemtest is **deels behaald**. Systeemtest 4 test ten eerste of de constructie van de end-effector de radiale krachten en de momenten opvangt. Vervolgens wordt er getest of de loadmeter axiale krachten kan meten. de eerste eis is niet behaald, omdat de sensor ook reageerde op belastingen in andere richtingen dan de axiale richting. Dit betekent dat de sensor te gevoelig is of dat de constructie niet volledig krachten kan opvangen.

De tweede eis is behaald, omdat de sensor de last kan meten in de axiale richting. Voor betrouwbare metingen moet er echter wel rekening gehouden worden dat de kracht enkel axiaal en niet schuin belast wordt.

Systeemtest 5 - 2/2 eisen behaald.

Deze systeemtest is **volledig behaald**. In deze test is er bepaald of de digital twin simulatie van de DOBOT aan de gestelde eisen voldoet. Rotatieassen J0, J1 en J2 bewegen in mee in de digital twin. Daarbij ligt de ververssnelheid van zowel de communicatie met de DOBOT als de Unity applicatie gemiddeld boven de 30 Hz.

Systeemtest 6 - 2/2 eisen behaald.

Deze systeemtest is **volledig behaald**. In systeemtest 6 zijn de eisen rondom de krachtenanalyse getest. De eisen stellen dat de vervormingen in objecten, ten gevolgen van een kracht aan de end-effector, visueel in de digital twin applicatie te zien zijn. Vervormingen kunnen enkel voor object front arm weergegeven worden, omdat er geen tijd was om de ontwikkelingen voor de resterende objecten door te voeren. Vooral nog wordt de eis als behaald gesteld.

Systeemtest 7 - 15/17 eisen behaald.

Deze systeemtest is **deels behaald**. Systeemtest 7 bestaat uit zeventien gebundelde eisen waarvan het niet nodig is om een testprocedure op te zetten. Dit zijn eisen waarbij een succesvolle afronding met een ja of nee vastgesteld kan worden. In deze test zijn er twee eisen niet behaald. De eerste eis is dat het protoboard (elektrische schakeling) bevestigd wordt aan het werkoppervlak met PCB afstandhouders. De tweede eis is dat de armen van de DOBOT in de krachtenanalyse niet versimpeld worden tot staven/-balken. De eerste eis is niet behaald, omdat er geen tijd voor over was. De prioriteiten werden gesteld op het afmaken van de andere deelfuncties. De tweede eis is niet behaald, omdat de benadering van de constructie te ingewikkeld werd. Dit is ook vermeld in hoofdstuk 7.1.

Het niet halen van de eerste eis heeft geen grote gevolgen voor de opdracht, eis 2 daarentegen wel. De krachten die namelijk bepaald worden aan de hand van de krachtenanalyse zijn gebaseerd op het krachtenmodel. Door de keuze te maken om de armen te versimpelen tot staven/balken is het model versimpeld en zijn daarmee de krachten minder representatief.

Tabel 6: Behaalde systeemtesten.

Systeemtest	Resultaat	Percentage behaalde eisen
1	Niet behaald	0%
2	Volledig behaald	100%
3	Volledig behaald	100%
4	Deels behaald	50%
5	Volledig behaald	100%
6	Volledig behaald	100%
7	Deels behaald	88.2%

Tabel 7: Behaalde eisen.

Eis	Resultaat	Eis	Resultaat
2	<input type="checkbox"/>	1	<input checked="" type="checkbox"/>
5	<input checked="" type="checkbox"/>	3	<input checked="" type="checkbox"/>
6	<input checked="" type="checkbox"/>	4	<input checked="" type="checkbox"/>
7	<input checked="" type="checkbox"/>	8	<input checked="" type="checkbox"/>
9	<input checked="" type="checkbox"/>	10	<input checked="" type="checkbox"/>
15	<input checked="" type="checkbox"/>	11	<input type="checkbox"/>
16	<input type="checkbox"/>	12	<input checked="" type="checkbox"/>
17	<input checked="" type="checkbox"/>	13	<input checked="" type="checkbox"/>
21	<input checked="" type="checkbox"/>	14	<input checked="" type="checkbox"/>
22	<input checked="" type="checkbox"/>	18	<input checked="" type="checkbox"/>
24	<input checked="" type="checkbox"/>	19	<input checked="" type="checkbox"/>
25	<input checked="" type="checkbox"/>	20	<input checked="" type="checkbox"/>
26	<input checked="" type="checkbox"/>	23	<input checked="" type="checkbox"/>
28	<input checked="" type="checkbox"/>	27	<input type="checkbox"/>

(a) Functionele eisen

(b) Realisatie eisen

In totaal zijn er 24 van de 28 eisen behaald. Dit geeft een totale score van 85.7%.

9 Conclusies & Aanbevelingen

Conclusies

De vraag van JB Systems is om de ontwikkelstappen die nodig zijn bij het digitaliseren van een machine/systeem vast te leggen. Deze hoofdvraag is in dit rapport beantwoord met behulp van drie deeldoelstelling:

- **Het uitlezen van een last op de DOBOT.**

Het waarnemen van een fysieke machine is een van de belangrijkste stappen in het digitaliseringsproces. Omdat het streven van de afstudeeropdracht is om klanten te kunnen voorzien van een tijdige waarschuwing over mogelijk onderhoud, is de last aan de end-effector het belangrijkste dat waargenomen dient te worden. Hiervoor is er een sensor ontwikkeld die door middel van rekstroken de mechanische spanningen in een materiaal ten gevolge van een last kan meten.

Naast deze sensor is er een bevestigingsonderdeel ontwikkeld waarmee de sensor aan de DOBOT bevestigd wordt. Om de sensor nauwkeurig uit te kunnen lezen met de DOBOT is er ook een versterkingsschakeling ontwikkeld waarmee het bereik van het signaal van 10 mV versterkt wordt tot 3 V. De sensor heeft een nauwkeurigheid van 0.5 N.

- **De DOBOT simuleren als een digital twin.**

Om de DOBOT digitaal te kunnen volgen dient er een 3D simulatie aanwezig te zijn. Er is gekozen om de DOBOT te simuleren in Unity3D. Het meegeleverde CAD model is hiervoor aangepast met twee geometrieën; een edge mesh STL (EMSTL) en een surface mesh STL (SMSTL). De EMSTL wordt gebruikt om de bewegingen van de werkelijke DOBOT over te nemen. De SMSTL wordt gebruikt om vervormingen aan te geven die uit de krachtenanalyse komen.

De digital twin kan de beweging van de werkelijke DOBOT volledig volgen. Onder bewegingen vallen de rotaties om assen J0, J1 en J2. Daarnaast biedt de digital twin een interface waarmee de gebruiker de DOBOT aan kan sturen en waarden uit kan lezen. De interface stelt waarden zoals ADC-waarde, geconverteerde kracht aan de end-effector en geconverteerde massa aan de end-effector beschikbaar. Dit kan in het vervolg uitgebreid worden met verschillende analyses die oordeel geven over de staat van de DOBOT.

- **Het uitvoeren van een krachtenanalyse aan de hand van de uitgelezen last.**

De uitvoer van de krachtenanalyse gebeurt op basis van een vooraf gestelde krachtenmodel en de gemeten last. De uitkomsten worden bij oplevering van de opdracht alleen gebruikt om vervormingen in een aantal objecten aan te tonen. Dit kan in het vervolg uitgebreid worden met materiaalkundige analyses om bijvoorbeeld vermoeiing in het materiaal van een constructie aan te tonen. Dit laat zien waar de zwakke punten van een constructie zich bevinden.

"Wat zijn de ontwikkelstappen die nodig zijn om de DOBOT te digitaliseren, voor het toepassen van predictief onderhoud?"

Concluderend is de bovenstaande hoofdvraag vanuit meerdere punten belicht. Hoewel de resultaten van de digital twin niet zover zijn dat deze geïntegreerd kunnen worden in de bestaande simulaties, heeft de opdracht ervoor gezorgd dat er een introductie is gemaakt in de volgende stap van het digitaliseringsproces.

Uit het digitaliseringsproces van de DOBOT blijkt dat de grootste uitdaging ligt bij het vergaren van de juiste data in voldoende hoeveelheid. Een tweede uitdaging is om een juiste systematische benadering

van de krachten in de constructie te maken. Het virtuele model kan immers niet beter zijn dan de benaderingen die ervoor gesteld zijn.

Aanbevelingen

Om de geleverde oplossingen een stap verder te brengen zijn er een aantal punten waarbij er verbeteringen doorgevoerd kunnen worden. Hieronder volgen er korte toelichting van deze punten.

In de conclusie is vermeld dat het waarnemen van de juiste waarden cruciaal is. Dit kan door middel van de volgende punten verbeterd worden.

1. Het bevestigen van de rekstroken

De rekstroken zijn bevestigd met behulp van Loctite 401, snelhardende lijm voor poreuze materialen. Dit heeft er echter voor gezorgd dat de rekstrook ligt gecorrodeerd is. Dit heeft nadelen gehad op de gevoeligheid en de totale weerstandsverandering van de rekstrook. Aanbevolen wordt om geschikte soort lijm te gebruiken om de rekstroken mee te bevestigen.

2. Het ontwerp van de end-effector

Zoals uit de testresultaten te zien is vangt de end-effector krachten in ongewenste richtingen niet op. Hier kan in verdiept worden om ervoor te zorgen dat alleen de krachten in de axiale richting meetellen in de meting.

3. Het ontwerp van de elektrische schakeling

Het elektrisch ontwerp van de loadmeter heeft meerdere punten waarop verbetering bereikt kan worden, dit komt vooral door de beperkte kennis in de elektronica.

Ten eerste wordt er aanbevolen om het ontwerpproces van de filters nogmaals te itereren. Hierbij dient er beter gekeken te moeten worden naar de kantelfrequenties van het signaal.

Ten tweede kan er nogmaals geïtereerd worden over het ontwerp van beide versterkers. Beide versterkingsstappen hebben nu respectievelijke versterkingen van 66.7 en 4.2. De nauwkeurigheid van de meting kan eventueel verbeterd worden door de verhouding tussen deze twee versterkingen aan te passen. Naast het aanpassen van de versterkingen kan er ook gekeken worden naar andere schakelingen waarmee dezelfde versterking bereikt kan worden.

De digital twin simulatie in unity kan aantrekkelijker gemaakt worden, door een omgeving toe te voegen waarin de virtuele DOBOT geplaatst kan worden. De virtuele DOBOT zweeft nu in de lucht en kan de schaal van de opstelling daardoor moeilijk ingeschat worden.

Voor de krachtenanalyse wordt er aanbevolen om de benadering van de constructie nogmaals te modelleren. In dit verslag zijn de armen versimpeld tot blaken en staven. Door de constructie complexer te modelleren kan ervoor gezorgd worden dat de benaderingen van de krachten en spanningen beter overeenkomen met de werkelijkheid.

Als laatste kan er onderzocht worden of het mogelijk is om de FEM analyse uit te voeren op een volledige assembly. In dit verslag zijn de FEM analyses uitgevoerd op deelassemblys (de basisonderdelen van hfst. 6.1). Dit kan ervoor zorgen dat de vorige aanbeveling komt te vervallen, aangezien de volledige assembly ervoor zorgt dat de hoeken correct overgedragen worden.

Referenties

- [1] J.T. de Balsch. Noise Filtering Using 1EURO Filter. 2020. URL: <https://jaantollander.com/post/noise-filtering-using-one-euro-filter/>.
- [2] A. Becker. Kalman Filter in one dimension. 2022. URL: <https://www.kalmanfilter.net/kalman1d.html>.
- [3] L. Delligatti. SysML Distilled. A Brief Guide To The Systems Modeling Language. Addison-Wesley Professional, 2013.
- [4] F.J. Siers. Methodisch ontwerpen. 3de ed. Noordhoff, 2004.
- [5] Unity. <https://unity.com/>. Versie 2020.3.30f1, Windows 10. 2022.
- [6] N. Storey. Electronics, A systems approach. 5de ed. Pearson Education Limited, 2013.
- [7] Autodesk Inc. Eagle. <https://www.autodesk.com/products/eagle>. Versie 9.6.2, Windows 10. 2022.
- [8] Inc. Analog Devices. LTspice. <https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html>. Versie 17.0.32.0, Windows 10. 2021.
- [9] Quite Universal Circuit Simulator. <http://qucs.sourceforge.net/>. Versie 0.0.19, Windows 10. 2019.
- [10] Single Supply, MicroPower INSTRUMENTATION AMPLIFIER. SBOS069. Burr-Brown Corporation. Okt 1997. URL: <https://www.ti.com/lit/ds/symlink/inal22.pdf>.
- [11] TLV237x 500-uA/Ch, 3-MHz Rail-to-Rail Input and Output Operational Amplifiers With Shutdown. SLOS270F. Rev. Oct 2006. Texas Instruments. Mrt 2001. URL: <https://www.ti.com/lit/ds/symlink/tlv2372.pdf>.
- [12] G. Casiez, N. Roussel en D. Vogel. „1-EURO Filter: A Simple Speed-based Low-pass Filter for Noisy Input in Interactive Systems”. In: HAL open science (2012).
- [13] Autodesk Inc. Fusion 360. <https://www.autodesk.eu/products/fusion-360>. Versie 2.0.13162, Windows 10. 2022.
- [14] S.D. de Jong. „Human Machine Interaction. Ontwerpen van interfaces voor mechatronische systemen; een user centered design benadering.” 2016.
- [15] M. Borovinšek. PrePoMax. <https://prepomax.fs.um.si/>. Versie 1.3.0, Windows 10. 2022.
- [16] G. Dhondt en K. Wittig. CALCULIX. <http://www.calculix.de/>. Versie 2.19, Windows 10. 2021.
- [17] Domein HBO Engineering. Bacheloropleidingen Engineering. 2016. URL: https://www.hbo-engineering.nl/_asset/_public/profiel_v3.pdf.

Bijlagen

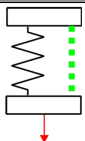
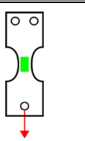
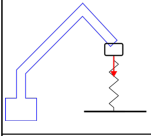
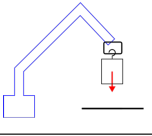
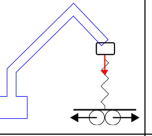
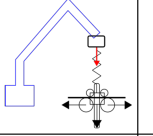
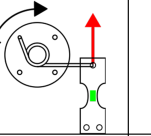
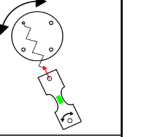




A	Algemene bijlage	50
B	Testplannen	65
B1	Systeemtesten	66
C	Uitgebreide berekeningen	73
C1	Berekeningen hoofdstuk 5 Het ontwerp van de loadmeter	73
C2	Berekeningen hoofdstuk 6 De digital twin simulatie van de DOBOT	77
C3	Berekeningen hoofdstuk 7 De krachtenanalyse van een last aan de DOBOT	78
D	Programma listings	85
E	Opdrachtvoorstel	89

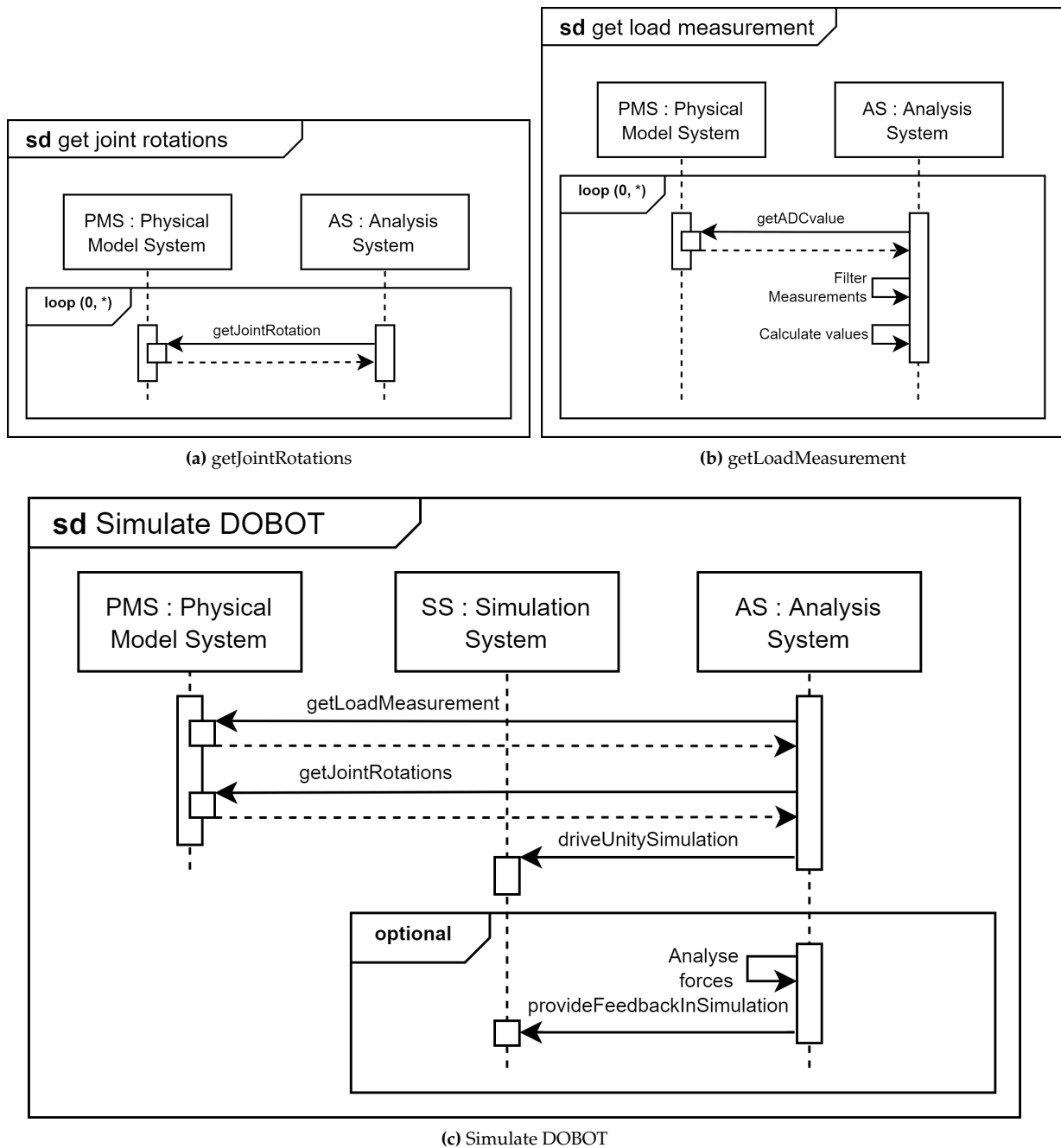
A Algemene bijlage

Tabel A1: Eisen met betreffende functies.

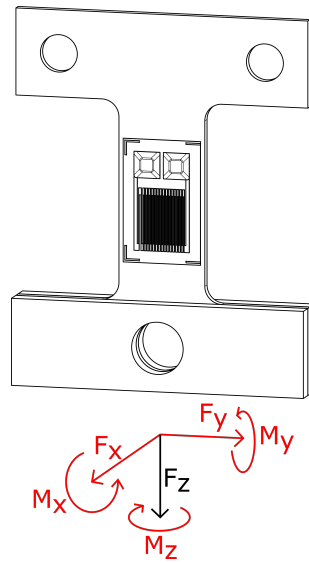
Eis	Functie 1 Load Meten		Functie 2 End-Effector Belasten					
	1	2	1	2	3	4	5	6
2. Bewegingen van de DOBOT mogen niet begrensd worden.			X	X	X	X	X	X
4. De totale kosten van de opdracht is onder €150.	X	X	X	X	X	X	X	X
5. Het domein van de loadmeting ligt minstens tussen 0 N en 5 N.	X	X						
6. De loadmeter kan minstens belast worden met een variabele kracht tussen 0 N en 5 N.			X	X	X	X	X	X
9. De gemeten kracht heeft een nauwkeurigheid van ± 0.5 N.	X	X						
16. De constructie van de loadmeter vangt radiale krachten en (torsie)momenten op.	X	X						
17. De loadmeter meet axiale krachten die op de end-effector uitgevoerd worden.	X	X						
18. De last op de end-effector is onafhankelijk van de positie van de armen			X	X	X	X	X	X

Tabel A2: Morfologisch overzicht van alle conceptoplossingen.

Functies	Conceptoplossingen					
	1	2	3	4	5	6
Load meten						
	Veer + afstandmeter	Rekstrook				
End-effector belasten						
	Vaste veer	Kalibratiegewichten	Vaste veer + horizontale geleider	Verticaal bewegende veer + horizontale geleider	Servomotor met een torsieveer	Servomotor met een trekveer
Dobot simuleren						
	Unity 3D	CAD programma				
Krachtenanalyse uitvoeren						
	Fusion 360 stress analyses	Externe FEM analyse softwarepakket				
Krachtenanalyse terugkoppelen						
	Verbogen model terugkoppelen naar simulatie in Unity	Verbogen model visueel aantonen in FEM softwarepakket				



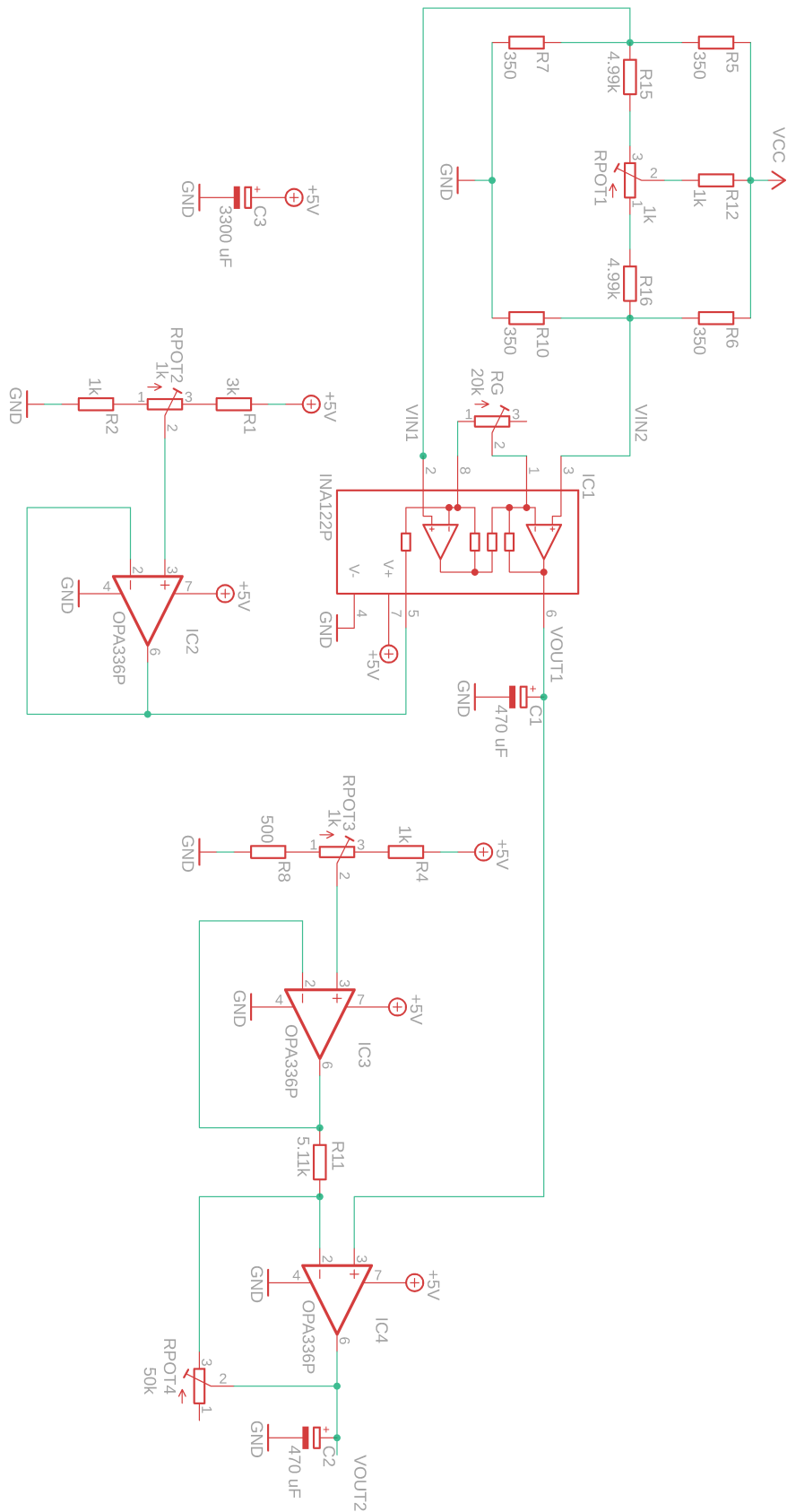
Figuur A1: Sequence diagrammen van de belangrijkste functies.



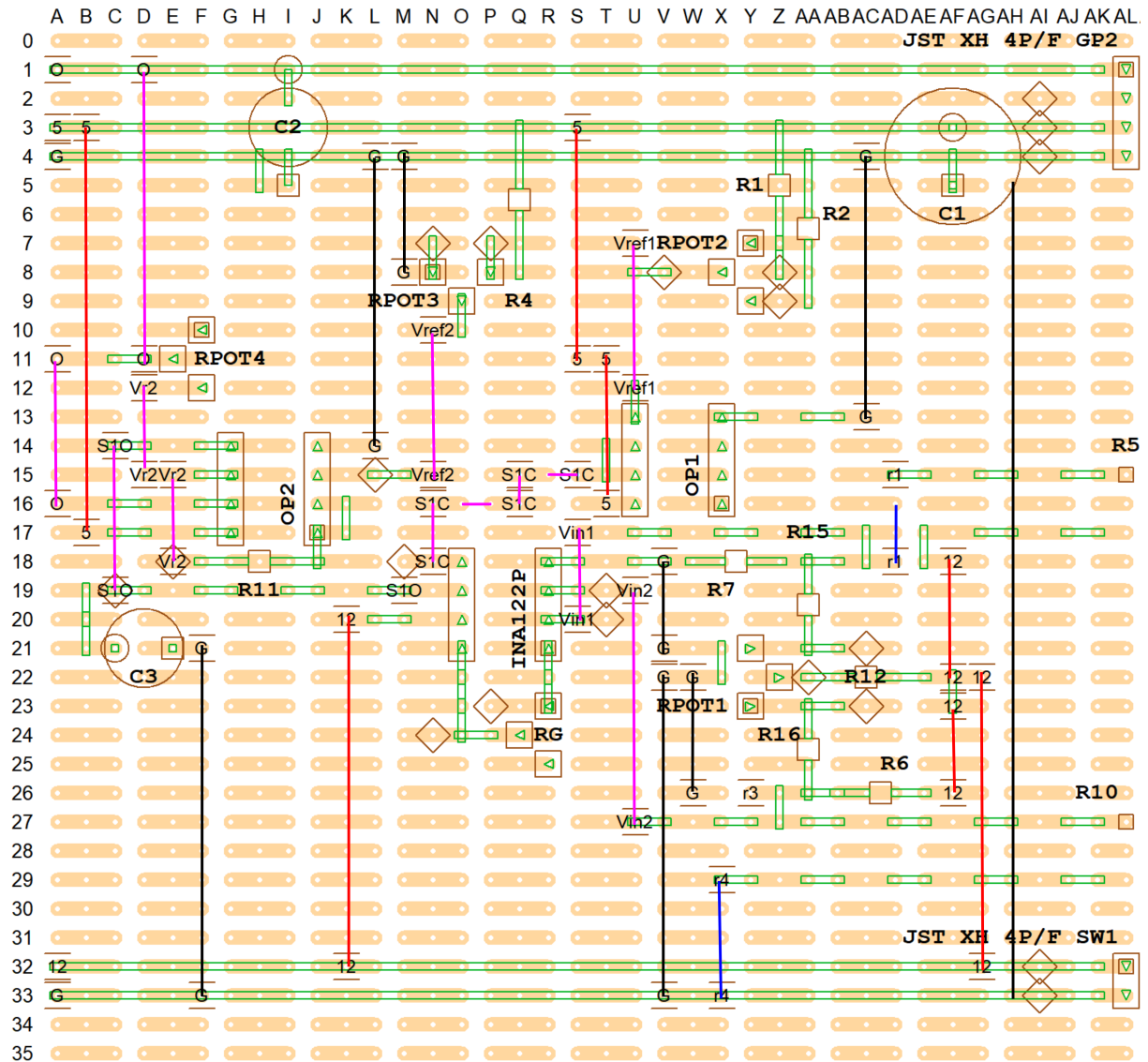
Figuur A2: Opvanging van ongewenste krachten.



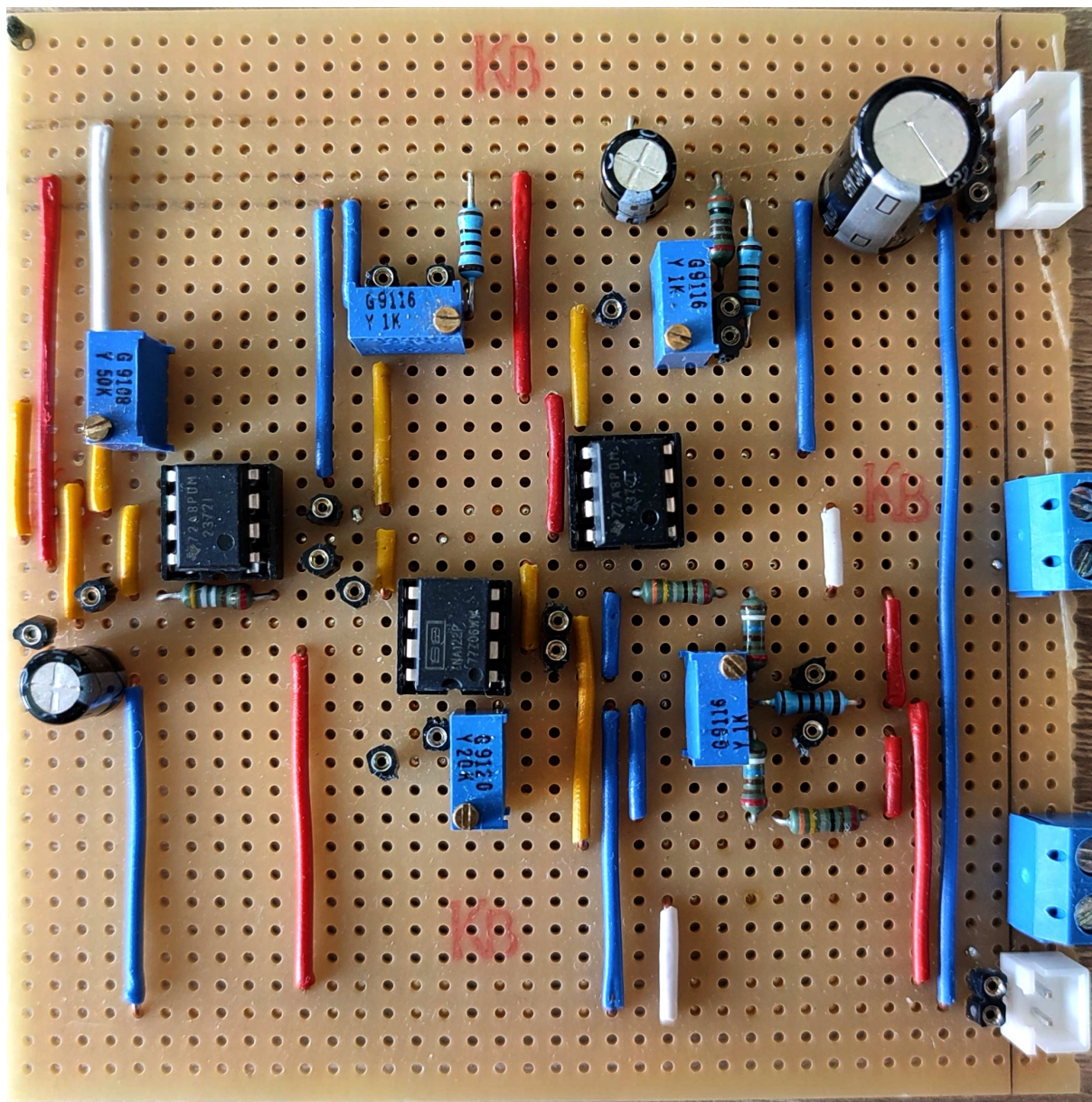
Figuur A3: De gerealiseerde loadmeter met de end-effector.



Figuur A4: Elektrische schakeling loadmeter.



Figuur A5: VeeCAD schakeltekening van de loadmeter.



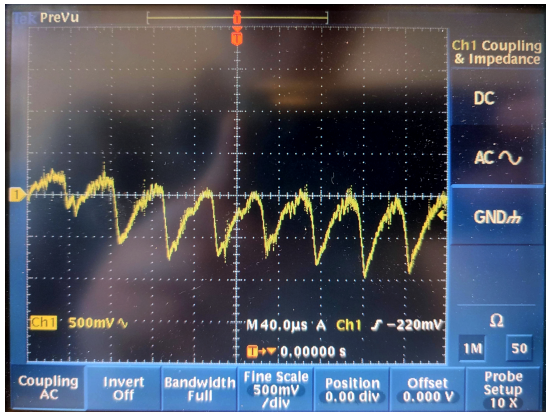
Figuur A6: Loadmeter PCB.

Tabel A3: Onderdelenlijst schakeling loadmeter.

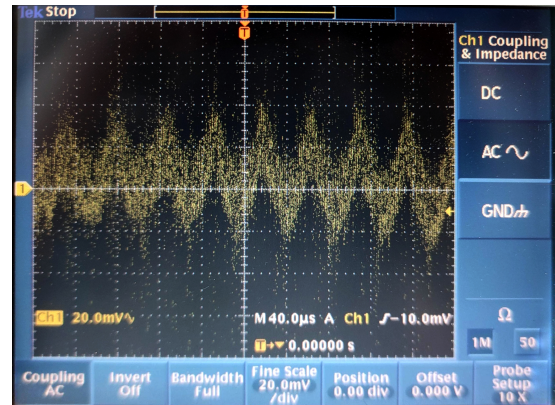
Naam/Namen	Component	Aantal	Beschikbaar
R5, R10	Rekstrook 348 Ω	2	<input type="checkbox"/>
R6, R7	Weerstand 350 Ω 1%	2	<input checked="" type="checkbox"/>
R _{pot1} , R _{pot2} , R _{pot3}	Trimmer potentiometer 1 k Ω	3	<input checked="" type="checkbox"/>
R _{pot4}	Trimmer potentiometer 50 k Ω	1	<input checked="" type="checkbox"/>
R15, R16	Weerstand 4.99 k Ω 1%	2	<input checked="" type="checkbox"/>
R _G	Trimmer potentiometer 20 k Ω	1	<input checked="" type="checkbox"/>
R1	Weerstand 3 k Ω 1%	1	<input checked="" type="checkbox"/>
R2, R4	Weerstand 1 k Ω 1%	2	<input checked="" type="checkbox"/>
R8	Weerstand 500 Ω 1%	1	<input checked="" type="checkbox"/>
R11	Weerstand 5.11 k Ω 1%	1	<input checked="" type="checkbox"/>
C3	Condensator 3.3 mF	1	<input checked="" type="checkbox"/>
C1, C2	Condensator 470 μ F	2	<input checked="" type="checkbox"/>
IC1	Instrumentatie Versterker INA122	1	<input type="checkbox"/>
IC2, IC3, IC4	Operationele Versterker TLV2372	3	<input type="checkbox"/>
-	DIP-8 socket	4	<input type="checkbox"/>
-	JST XH 2-pin male	1	<input type="checkbox"/>
-	JST XH 4-pin male	1	<input type="checkbox"/>
-	Klemschroefblok 2 polen	2	<input checked="" type="checkbox"/>
-	Tripad PCB	1	<input checked="" type="checkbox"/>
-	Female Header pin	22	<input checked="" type="checkbox"/>

Tabel A4: Algemene kosten.

Component	Aantal	Totale Kosten incl BTW. excl. verzendkosten
v Rekstrook 348 Ω	2	€ 1,15
Instrumentatie Versterker INA122	1	€ 8,01
Operationele Versterker TLV2372	3	€ 7,05
DIP-8 socket	4	€ 0,72
JST XH 2-pin male	1	€ 0,20
JST XH 4-pin male	1	€ 0,13
0.5 mm polycarbonaat plaat	1	€ 5,99
Kalibratie gewichten set	1	€ 48,77

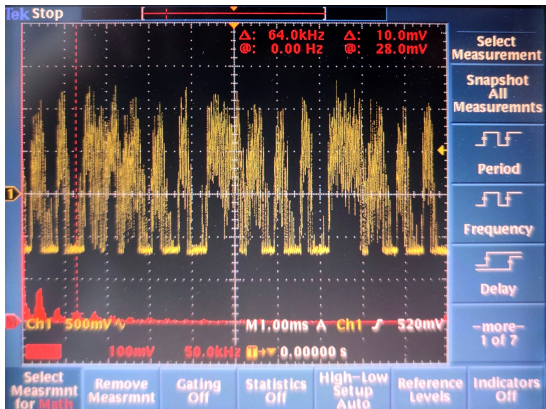


(a) Ongefilterd signaal V_{out1}

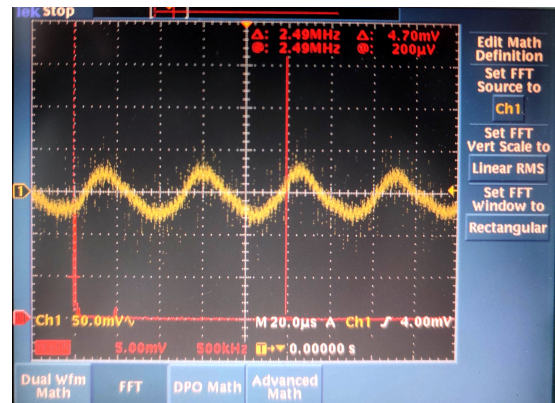


(b) Gefilterd signaal V_{out1}

Figuur A7: Het effect van een laagdoorlaatfilter op V_{out1} .

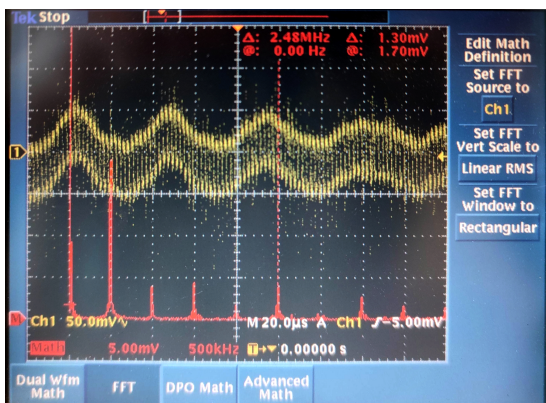


(a) Ongefilterd signaal V_{out2}

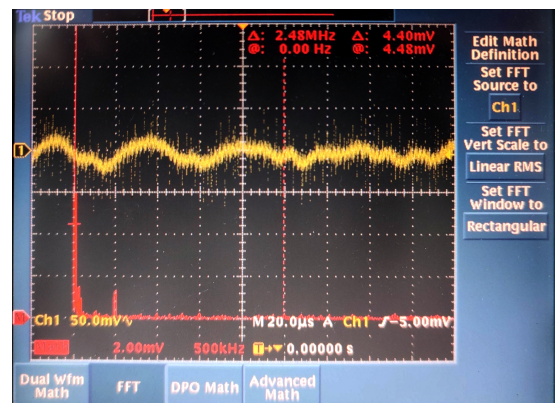


(b) Gefilterd signaal V_{out2}

Figuur A8: Het effect van een laagdoorlaatfilter op V_{out2} .

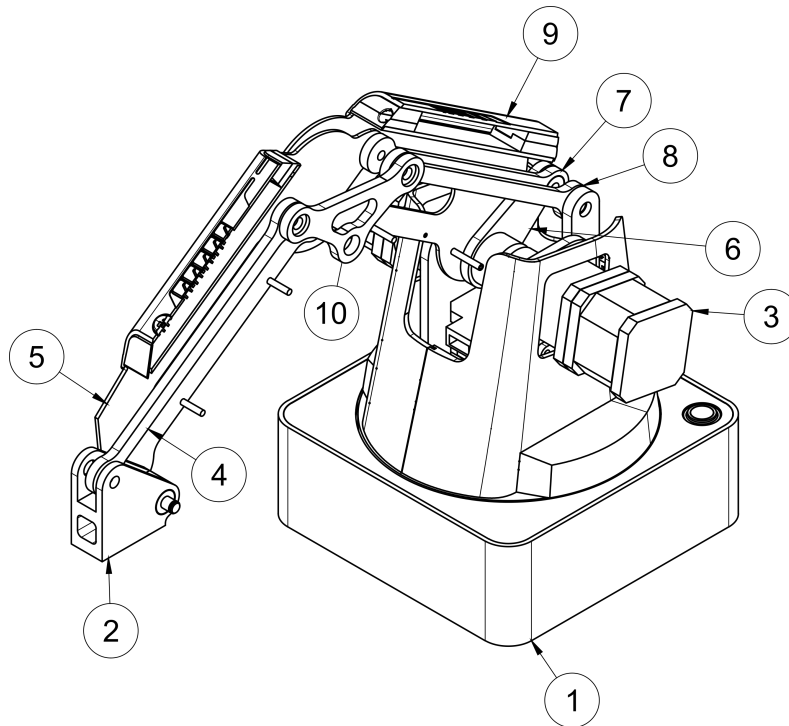


(a) Ongefilterd signaal V_{cc}

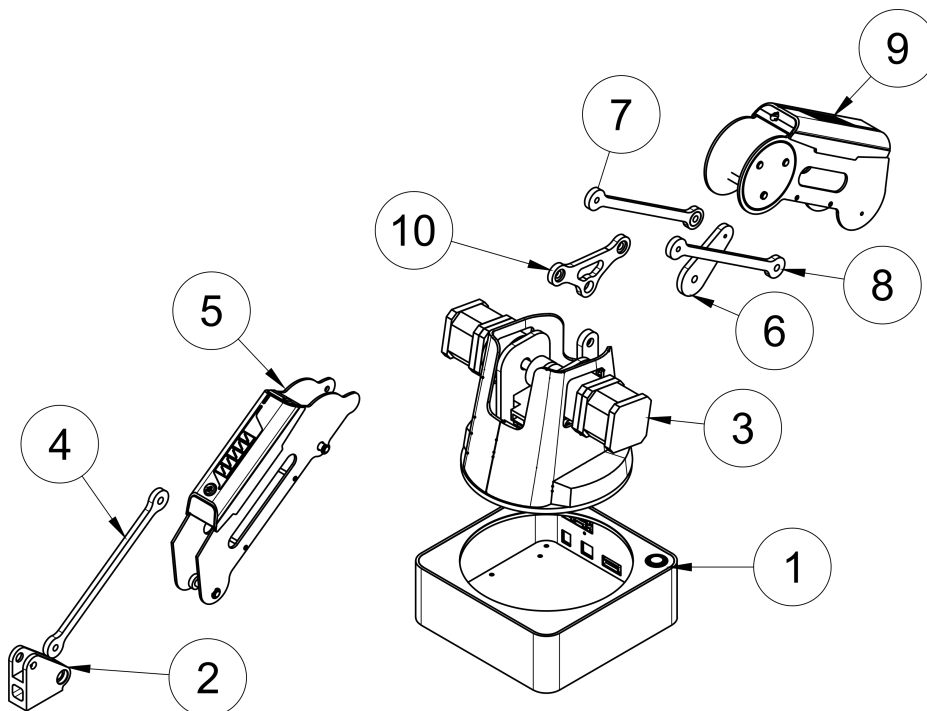


(b) Gefilterd signaal V_{cc}

Figuur A9: Het effect van een laagdoorlaatfilter op V_{cc} .



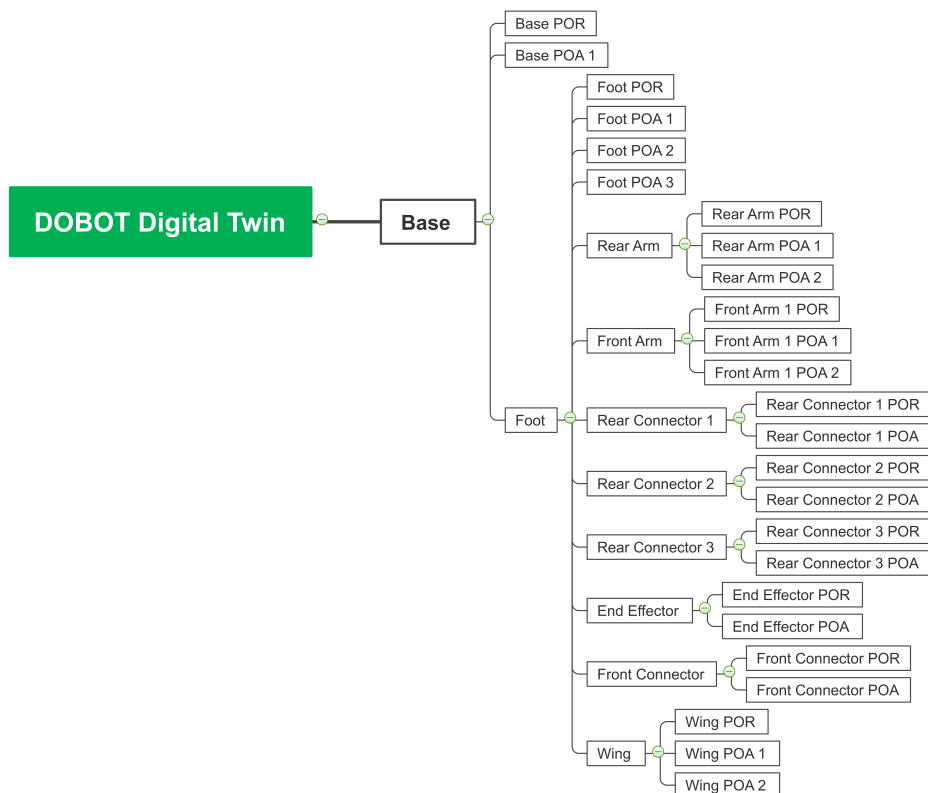
Figuur A10: Section view van de DOBOT.



Figuur A11: Exploded view van de tien DOBOT onderdelen.

Tabel A5: Onderdelentabel section view **A10** en exploded view **A11**.

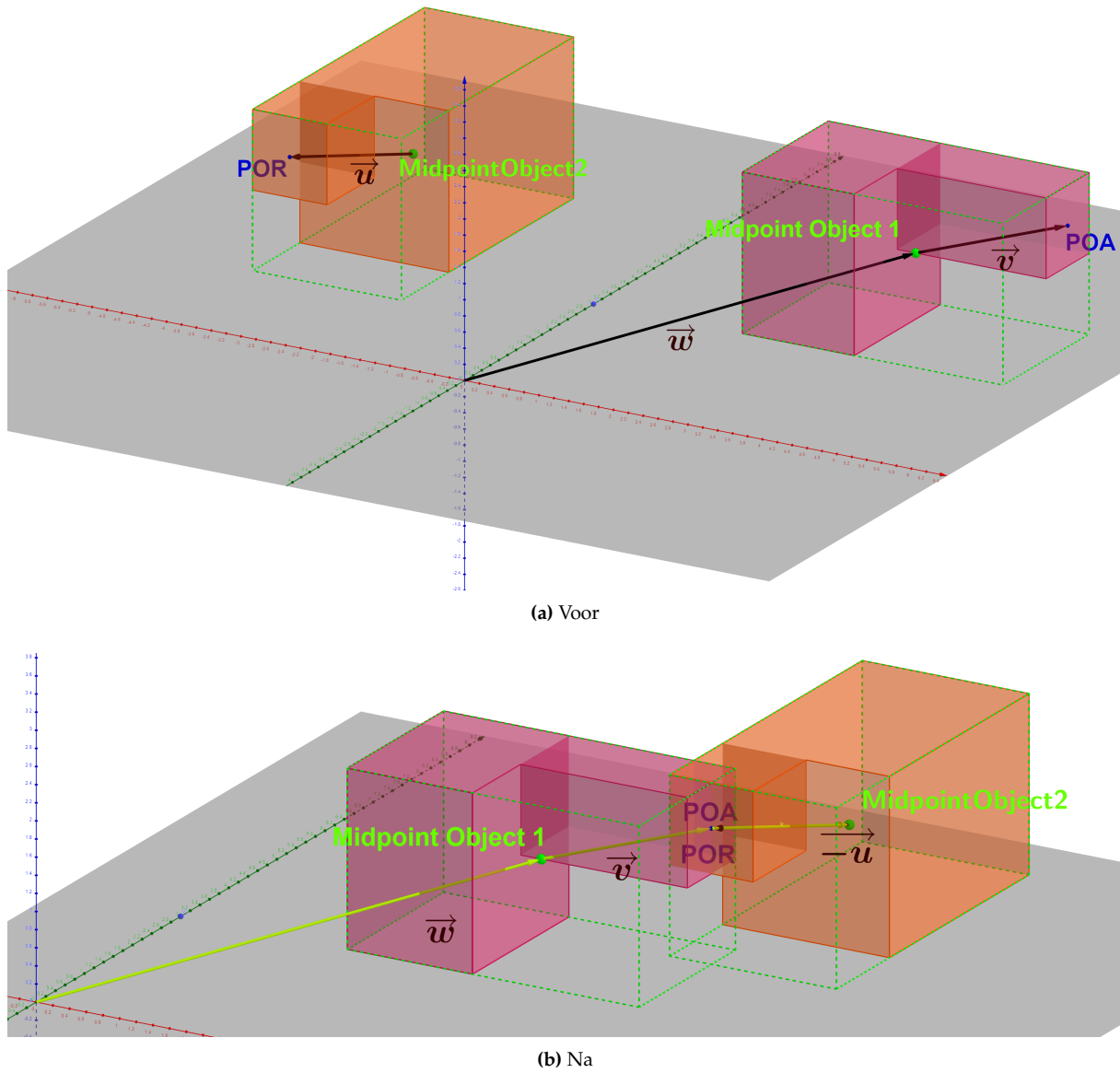
1 Base	3 Foot	5 Front Arm	7 Rear Connector 2	9 Rear Arm
2 End-Effector	4 Front Connector	6 Rear Connector 1	8 Rear Connector 3	10 Wing



Figuur A12: Hiërarchie van de objecten in Unity.

Tabel A6: Contactparen van de basisobjecten.

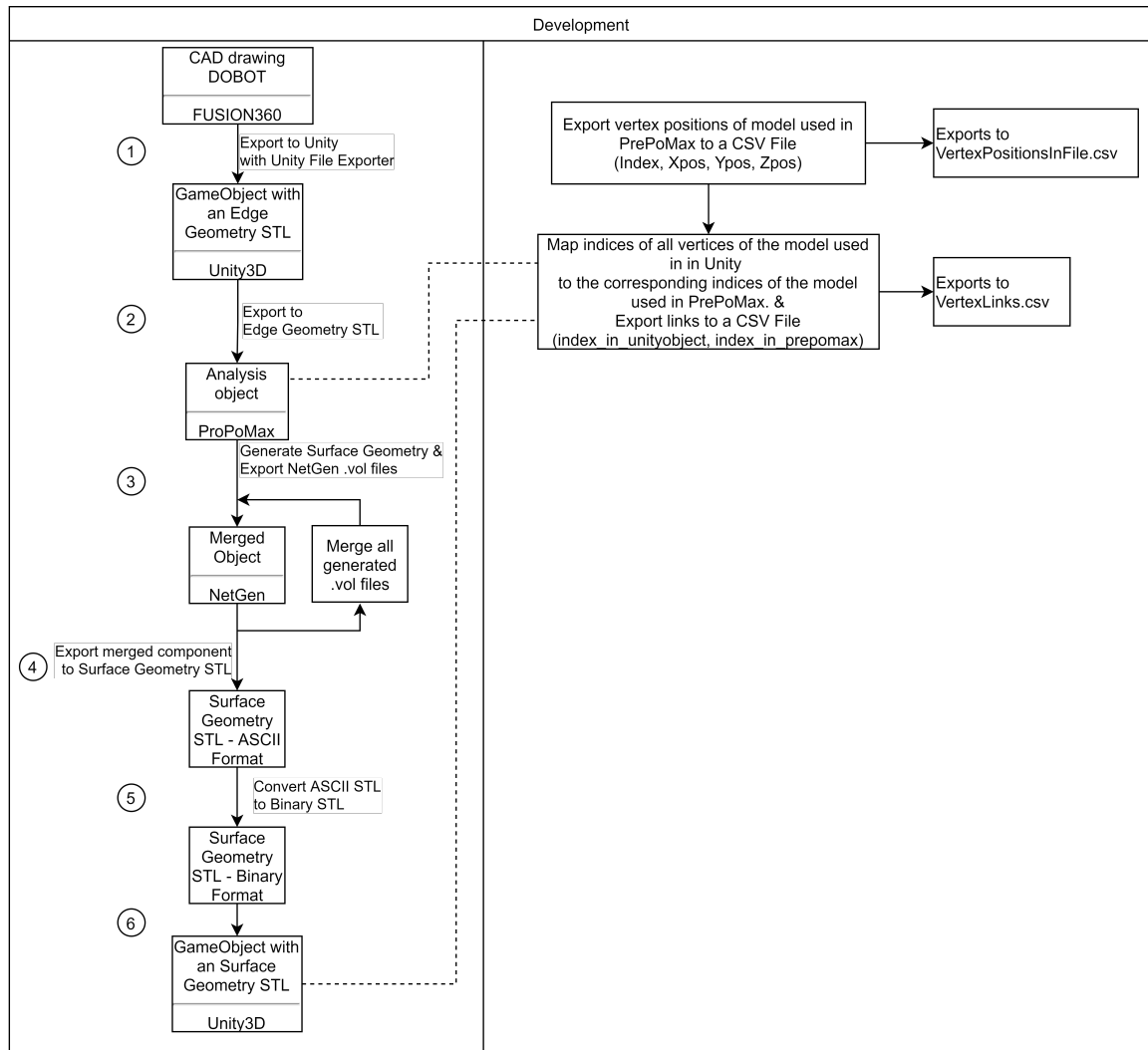
Basisonderdeel	Connectie 1	Connectie 2
Base	Base POR Base POA	(0,0,0) Foot POR
Foot	Foot POR Foot POA 1 Foot POA 2 Foot POA 3	Base POA Rear Connector 1 POR Rear Arm POR Rear Connector 3 POR
Rear Arm	Rear Arm POR Rear Arm POA 1 Rear Arm POA 2	Foot POA 2 Front Arm POR Wing POR
Front Arm	Front Arm POR Front Arm POA 1 Front Arm POA 2	Rear Arm POA 1 Rear Connector 2 POA End Effector POA
Rear Connector 1	Rear Connector 1 POR Rear Connector 1 POA	Foot POA 1 Rear Connector 2 POR
Rear Connector 2	Rear Connector 2 POR Rear Connector 2 POA	Rear Connector 1 POA Front Arm POA 1
Rear Connector 3	Rear Connector 3 POR Rear Connector 3 POA	Foot POA 3 Wing POA 1
End Effector	End Effector POR End Effector POA	Front Connector POA Front Arm POA
Front Connector	Front Connector POR Front Connector POA	Wing POA 2 End Effector POR
Wing	Wing POR Wing POA 1 Wing POA 2	Rear Arm POA 2 Rear Connector 3 POA Front Connector POR



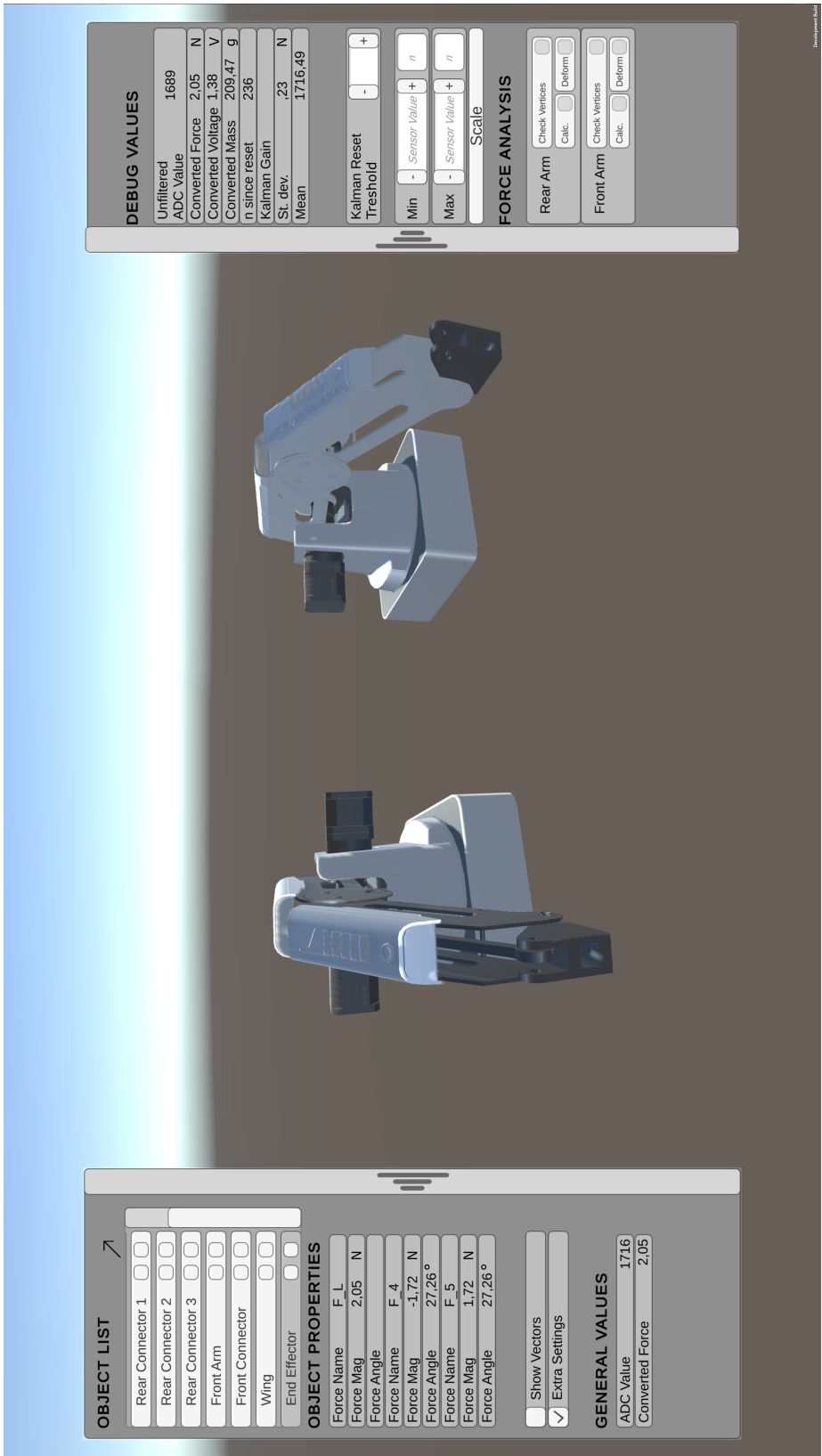
Figuur A13: Toelichting translatie van contactpunten.

Waarbij:

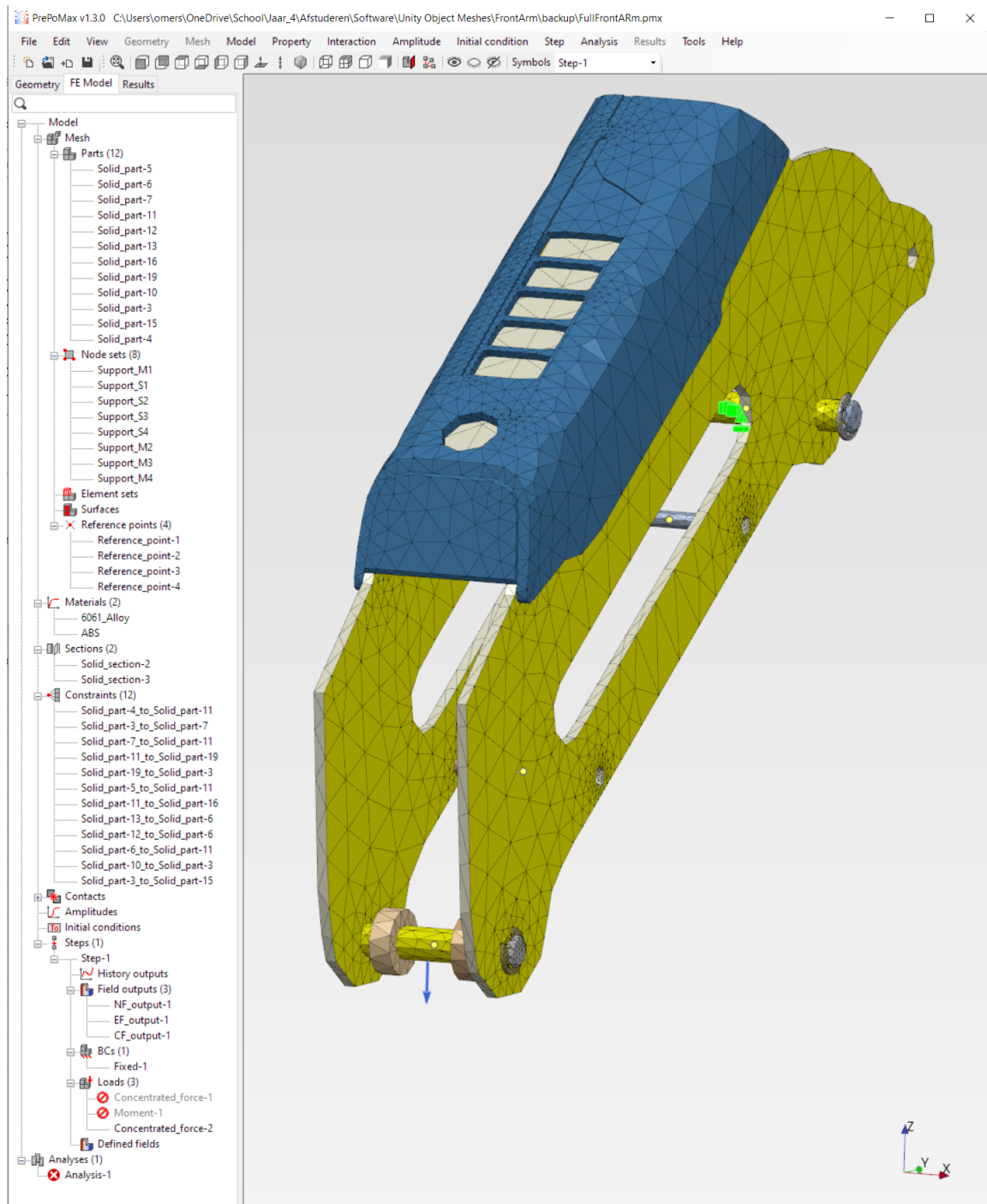
$$\begin{aligned}\vec{w} &= \vec{r}_p \\ \vec{v} &= \vec{r}_{p_{POA}} \\ \vec{u} &= \vec{r}_{c_{POR}}\end{aligned}$$



Figuur A14: De conversiestappen om het CAD model om te zetten naar een SMSTL.



Figuur A15: De HMI van de digital twin.



Figuur A16: Initiële configuratie van de onderdeel Front Arm.

B Testplannen

In de systeemtesten zullen de eisen die gesteld zijn aan het begin van de opdracht getoetst worden. Dit wordt gedaan in samenhang met de opdrachtgever en wordt er besloten of de opdracht succesvol behaald is. Hieronder worden er een aantal algemene punten behandeld die nodig zullen zijn bij het testen.

Vorbereidingen:

Hardware benodigdheden	Software benodigdheden
1 DOBOT robotarm	1 Digital twin simulatie
2 End-effector met de spanningsmodule	2 PrePoMax V1.3.0
3 Set kalibratiegewichten	3 Python 3.10.4 met benodigde pakketten
4 Multimeter	
5 PC/Laptop	

Testomgeving 1:

Testomgeving 1 is de algemene testomgeving waar de komende testen uitgevoerd gaan worden. Hier wordt de DOBOT bevestigd aan de rand van een tafel. De DOBOT wordt zo georiënteerd dat het werkgebied van de robotarm niet boven de tafel is. De testomgeving beschikt over stopcontacten om al het apparatuur van elektriciteit te voorzien.

B1 Systeemtesten

Tabel B1: Systeemtest 1. Bewegingsvrijheid van de DOBOT.

Systeemtest 1:					
Eisnummer:	2: Bewegingen van de DOBOT mogen niet begrensd worden.				
Voorwaarden:	<p>De test wordt uitgevoerd in testomgeving 1.</p> <table> <tr> <th>Hardware benodigdheden</th><th>Software benodigdheden</th></tr> <tr> <td> 1 DOBOT robotarm 2 End-effector met de spanningsmodule 3 Set kalibratiegewichten 4 PC/Laptop </td><td> 1 Python 3.10.4 met benodigde pakketten </td></tr> </table>	Hardware benodigdheden	Software benodigdheden	1 DOBOT robotarm 2 End-effector met de spanningsmodule 3 Set kalibratiegewichten 4 PC/Laptop	1 Python 3.10.4 met benodigde pakketten
Hardware benodigdheden	Software benodigdheden				
1 DOBOT robotarm 2 End-effector met de spanningsmodule 3 Set kalibratiegewichten 4 PC/Laptop	1 Python 3.10.4 met benodigde pakketten				
Testinvoer:	Als testinvoer zullen beide hoeken met behulp van een programma aangestuurd worden.				
Testuitvoer:	De arm beweegt in zijn volledige bewegingsbereik.				
Criteria:	Er ontstaan geen botsingen tussen de DOBOT en zijn omgeving.				
Testprocedure:	<ol style="list-style-type: none"> 1. De DOBOT wordt aangesloten op de laptop/pc. 2. De DOBOT wordt aangezet. 3. De houder van de gewichtenset wordt opgehangen aan de end-effector. 4. Vul de houder met de massaschijven. 5. Er wordt een programma uitgevoerd waarmee de DOBOT bewogen wordt. 6. Observeer of de constructie botst met de omgeving. 7. Schakel de DOBOT uit. 8. Ga na of het criterium is behaald. 				

Tabel B2: Systeemtest 2. Het meetbereik van de loadmeter.

Systeemtest 2:					
Eisnummers:	<p>5: Het domein van de loadmeting ligt minstens tussen 0 N en 5 N.</p> <p>6: De loadmeter kan minstens belast worden met een variabele kracht tussen 0 N en 5 N.</p> <p>7: Het analoge uitgangssignaal van de loadmeting heeft een bereik van 0.5 V tot 3.3 V.</p>				
Voorwaarden:	<p>De test wordt uitgevoerd in testomgeving 1.</p> <table> <tr> <th>Hardware benodigdheden</th><th>Software benodigdheden</th></tr> <tr> <td> <ul style="list-style-type: none"> 1 DOBOT robotarm 2 End-effector met de spanningsmodule 3 Set kalibratiegewichten 4 Multimeter 5 PC/Laptop </td><td> <ul style="list-style-type: none"> 1 Digital twin simulatie </td></tr> </table>	Hardware benodigdheden	Software benodigdheden	<ul style="list-style-type: none"> 1 DOBOT robotarm 2 End-effector met de spanningsmodule 3 Set kalibratiegewichten 4 Multimeter 5 PC/Laptop 	<ul style="list-style-type: none"> 1 Digital twin simulatie
Hardware benodigdheden	Software benodigdheden				
<ul style="list-style-type: none"> 1 DOBOT robotarm 2 End-effector met de spanningsmodule 3 Set kalibratiegewichten 4 Multimeter 5 PC/Laptop 	<ul style="list-style-type: none"> 1 Digital twin simulatie 				
Testinvoer:	Belasting van de spanningsmodule door middel van losse 50 g massaschijven.				
Testuitvoer:	Uitgangsspanning V_{out2}				
Criteria:	De uitgangsspanning is in de onbelaste toestand 0 ± 0.25 V en 3 ± 0.25 V bij een volledige belasting van 5 N. De onzekerheden zijn gebaseerd op eis 9.				
Testprocedure:	<ul style="list-style-type: none"> 1. De DOBOT wordt aangesloten op de laptop/pc. 2. De DOBOT wordt aangezet. 3. De houder van de gewichtenset wordt opgehangen aan de end-effector. 4. De elektrische schakeling wordt gekalibreerd met de stappen beschreven in bijlage C1. 5. Meet met de uitgangsspanning op punt D1 van figuur A5 6. Vul de houder met twee massaschijven. 7. Herhaal stap 5 en 6 tot de houder volledig gevuld is met 10 schijven. 8. Ga na of het criterium is behaald. 9. Schakel de DOBOT uit. 				

Tabel B3: Systeemtest 3. Gegevens van de loadmeter.

Systeemtest 3:													
Eisnummers:	9: De gemeten kracht heeft een nauwkeurigheid van ± 0.5 N. 15: Het vermogen van de versterker is niet meer dan 0.25 W.												
Voorwaarden:	De test wordt uitgevoerd in testomgeving 1. <table border="1"> <thead> <tr> <th>Hardware benodigdheden</th><th>Software benodigdheden</th></tr> </thead> <tbody> <tr> <td>1 DOBOT robotarm</td><td>1 Digital twin simulatie</td></tr> <tr> <td>2 End-effector met de spanningsmodule</td><td></td></tr> <tr> <td>3 Set kalibratiegewichten</td><td></td></tr> <tr> <td>4 Multimeter</td><td></td></tr> <tr> <td>5 PC/Laptop</td><td></td></tr> </tbody> </table>	Hardware benodigdheden	Software benodigdheden	1 DOBOT robotarm	1 Digital twin simulatie	2 End-effector met de spanningsmodule		3 Set kalibratiegewichten		4 Multimeter		5 PC/Laptop	
Hardware benodigdheden	Software benodigdheden												
1 DOBOT robotarm	1 Digital twin simulatie												
2 End-effector met de spanningsmodule													
3 Set kalibratiegewichten													
4 Multimeter													
5 PC/Laptop													
Testinvoer:	Belasting van de spanningsmodule door middel van losse 50 g massaschijven.												
Testuitvoer:	1. Het elektrisch vermogen van de schakeling. 2. De gefilterde meetwaarde.												
Criteria:	1. Het totale elektrisch vermogen is minder dan 0.25 W. 2. De totale fout in de gefilterde meetwaarde ligt tussen 0.5 N.												
Testprocedure:	1. De DOBOT wordt aangesloten op de laptop/pc. 2. Verbind de multimeter in serie met de 5V voedingslijn, zodat de stroomsterkte door de schakeling gemeten kan worden. 3. Start de digital twin applicatie. 4. De DOBOT wordt aangezet. 5. De houder van de gewichtenset wordt opgehangen aan de end-effector. 6. De elektrische schakeling wordt gekalibreerd met de stappen beschreven in bijlage C1. 7. Vul de houder met een willekeurige hoeveelheid massaschijven. 8. Lees in de digital twin applicatie de gefilterde waarde voor F_L uit. 9. Meet de stroomsterkte door de schakeling. 10. Schakel de DOBOT uit. 11. Herhaal de test vanaf stap 2, maar sluit de multimeter dit keer in serie aan met de 12V voedingslijn. 12. Ga na of de criteria zijn behaald.												

Tabel B4: Systeemtest 4. Belasting op de loadmeter.

Systeemtest 4:									
Eisnummers:	16: De constructie van de loadmeter vangt radiale krachten en (torsie)momenten op. 17: De loadmeter meet axiale krachten die op de end-effector uitgevoerd worden.								
Voorwaarden:	<p>De test wordt uitgevoerd in testomgeving 1.</p> <table> <tr> <th>Hardware benodigdheden</th><th>Software benodigdheden</th></tr> <tr> <td>1 DOBOT robotarm</td><td>1 Digital twin simulatie</td></tr> <tr> <td>2 End-effector met de spanningsmodule</td><td></td></tr> <tr> <td>3 PC/Laptop</td><td></td></tr> </table>	Hardware benodigdheden	Software benodigdheden	1 DOBOT robotarm	1 Digital twin simulatie	2 End-effector met de spanningsmodule		3 PC/Laptop	
Hardware benodigdheden	Software benodigdheden								
1 DOBOT robotarm	1 Digital twin simulatie								
2 End-effector met de spanningsmodule									
3 PC/Laptop									
Testinvoer:	Belasten van de spanningsmodule in de x, y en z richtingen.								
Testuitvoer:	De gefilterde meetwaarde								
Criteria:	De gefilterde meetwaarde toont alleen een significante verandering bij belastingen in de z richting.								
Testprocedure:	<ol style="list-style-type: none"> 1. De DOBOT wordt aangesloten op de laptop/pc. 2. De DOBOT wordt aangezet. 3. Start de digital twin applicatie. 4. De elektrische schakeling wordt gekalibreerd met de stappen beschreven in bijlage C1. 5. Knoop een stuk touw vast aan de spanningsmodule. 6. Trek voorzichtig aan het touw in de X richting om de spanningsmodule in een geïsoleerde richting te belasten 7. Observeer de gefilterde meetwaarde in de digital twin applicatie. 8. Herhaal de test nogmaals twee keer vanaf stap 6, dit keer met belastingen in de y en z richtingen. 9. Schakel de DOBOT uit. 10. Ga na of het criterium is behaald. 								

Tabel B5: Systeemtest 5. Gegevens van de Digital Twin.

Systeemtest 5:							
Eisnummers:	21: Alle drie bewegingsassen bewegen in de simulatie mee met de DOBOT. 22: De simulatie heeft een ververssnelheid van minimaal 20 Hz. (excl. krachtenanalyse)						
Voorwaarden:	De test wordt uitgevoerd in testomgeving 1. <table border="1"> <thead> <tr> <th>Hardware benodigdheden</th><th>Software benodigdheden</th></tr> </thead> <tbody> <tr> <td>1 DOBOT robotarm</td><td>1 Digital twin simulatie</td></tr> <tr> <td>2 PC/Laptop</td><td></td></tr> </tbody> </table>	Hardware benodigdheden	Software benodigdheden	1 DOBOT robotarm	1 Digital twin simulatie	2 PC/Laptop	
Hardware benodigdheden	Software benodigdheden						
1 DOBOT robotarm	1 Digital twin simulatie						
2 PC/Laptop							
Testinvoer:	Het fysiek bewegen van de DOBOT.						
Testuitvoer:	Het virtueel bewegen van de DOBOT.						
Criteria:	1. De drie bewegingsassen(J0, J1, J2) bewegen in de simulatie mee met de werkelijke DOBOT. 2. De waarden in de simulatie hebben een ververssnelheid van 20 Hz.						
Testprocedure:	1. De DOBOT wordt aangesloten op de laptop/pc. 2. De DOBOT wordt aangezet. 3. Start de digital twin applicatie. 4. Druk op de knop " Start Random Movement ". Dit voert zowel een script uit om de DOBOT te bewegen en houdt gedurende de beweging de gemiddelde ververssnelheid van de waarden bij. 5. Observeer of de virtuele DOBOT meebeweegt. 6. Observeer de gemiddelde ververssnelheid in de digital twin applicatie. 7. Schakel de DOBOT uit. 8. Ga na of de criteria zijn behaald.						

Tabel B6: Systeemtest 6. Gegevens van de krachtenanalyse.

Systeemtest 6:											
Eisnummers:	<p>25: De vervormingen van de DOBOT bij een axiale kracht aan de end-effector zijn visueel uit de krachtenanalyse te halen.</p> <p>26: De vervormingen uit de krachtenanalyse worden teruggekoppeld naar de 3D simulatie in Unity.</p>										
Voorwaarden:	<p>De test wordt uitgevoerd in testomgeving 1.</p> <table> <tr> <th>Hardware benodigdheden</th><th>Software benodigdheden</th></tr> <tr> <td>1 DOBOT robotarm</td><td>1 Digital twin simulatie</td></tr> <tr> <td>2 End-effector met de spanningsmodule</td><td>2 PrePoMax V1.3.0</td></tr> <tr> <td>3 Set kalibratiegewichten</td><td></td></tr> <tr> <td>4 PC/Laptop</td><td></td></tr> </table>	Hardware benodigdheden	Software benodigdheden	1 DOBOT robotarm	1 Digital twin simulatie	2 End-effector met de spanningsmodule	2 PrePoMax V1.3.0	3 Set kalibratiegewichten		4 PC/Laptop	
Hardware benodigdheden	Software benodigdheden										
1 DOBOT robotarm	1 Digital twin simulatie										
2 End-effector met de spanningsmodule	2 PrePoMax V1.3.0										
3 Set kalibratiegewichten											
4 PC/Laptop											
Testinvoer:	Hoeken J1 en J2 en kracht F_L worden met een druk op een knop gebruikt om een krachtenanalyse uit te voeren.										
Testuitvoer:	Een vervormd object in de digital twin applicatie.										
Criteria:	De vervormingen ten gevolgen van de testinvoer zijn zichtbaar in de objecten in de digital twin applicatie. Daarbij wordt ook de vervormingsschaal weergegeven.										
Testprocedure:	<ol style="list-style-type: none"> 1. De DOBOT wordt aangesloten op de laptop/pc. 2. De DOBOT wordt aanzet. 3. Start de digital twin applicatie. 4. De houder van de gewichtenset wordt opgehangen aan de end-effector. 5. De elektrische schakeling wordt gekalibreerd met de stappen beschreven in bijlage C1. 6. Vul de houder met een willekeurige hoeveelheid massaschijven. 7. Druk op de knop "Check Vertex Links" om te controleren of de vooraf gegenereerde bestanden van het geselecteerde object correct zijn. 8. Wacht tot het proces succesvol afgerond is. Afronding wordt aangegeven met een groene vink. 9. Druk op de knop "Calculate" de FEM analyse te starten van het geselecteerde object met de actuele invoervariabelen. 10. Wacht tot het proces succesvol afgerond is. Afronding wordt aangegeven met een groene vink. 11. Druk op de knop "Deform Object" om de geselecteerde object te vervormen met de uitvoer van de FEM analyse. 12. Schakel de DOBOT uit. 13. Ga na of de criteria zijn behaald. 										

Tabel B7: Systeemtest 7. Overige vaste eisen

Systeemtest 7:	
Eisnummers:	<p>1: De bouw van de DOBOT blijft onveranderd.</p> <p>3: Alle originele componenten van de DOBOT worden gebruikt.</p> <p>4: De totale kosten van de opdracht is onder €150.</p> <p>7: Het analoge uitgangssignaal van de loadmeting heeft een bereik van 0 V tot 3.3 V.</p> <p>8: Als spanningsbron wordt er gebruik gemaakt van een 12 V uitgang van de DOBOT.</p> <p>10: Het protoboard is een stripboard (TriPad) bestaande uit drie geclusterde koperen strips.</p> <p>11: Het protoboard voor de loadmeting wordt gefixeerd aan het werkoppervlak met PCB afstandhouders.</p> <p>12: Het protoboard beschikt over mogelijkheden om het nul-last signaal te kalibreren.</p> <p>13: Het protoboard wordt zo ontwikkeld dat componenten individueel verwisseld kunnen worden.</p> <p>14: Het protoboard heeft één kabel / componenten per soldeerpad.</p> <p>18: De last op de endeffector is onafhankelijk van de positie van de armen.</p> <p>19: De loadmeter wordt NIET ontworpen als een custom PCB.</p> <p>20: Er worden GEEN externe spanningsbronnen gebruikt in het ontwerp van de loadmeter.</p> <p>23: De DOBOT wordt gesimuleerd in Unity 3D.</p> <p>24: De simulate houdt GEEN rekening met botsende en wrijvende oppervlakten.</p> <p>27: Bij de krachtenanalyse worden de armen van de DOBOT niet versimpeld tot staven.</p> <p>28: Radiale krachten en (torsie)momenten worden NIET meegenomen in de analyse.</p>
Voorwaarden:	De test wordt uitgevoerd in testomgeving 1.
Criterium:	De bovenstaande eisen zijn behaald als deze met een ja beantwoord kunnen worden. Zo niet zijn deze niet behaald.

C Uitgebreide berekeningen

C1 Berekeningen hoofdstuk 5 Het ontwerp van de loadmeter

Deelhoofdstuk 5.1 Mechanisch Ontwerp

$$\sigma = \frac{F}{A} 11 \quad (C1)$$

$$\epsilon = \frac{\sigma}{E} \quad (C2)$$

Deelhoofdstuk 5.2 Elektrisch Ontwerp

Berekening van V_{diff} met weerstand R_{10} als rekstrook:

$$V_{diff} = \frac{R_{rek}R_5 - R_6R_7}{(R_{rek} + R_6)(R_7 + R_5)} \cdot V_{cc} = \frac{350.417 \cdot 350 - 350 \cdot 350}{(350.417 + 350)(350 + 350)} \cdot 12 = 4.28 \text{ mV} \quad (C3)$$

Berekening van V_{diff} met weerstand R_{10} en R_5 als rekstroken:

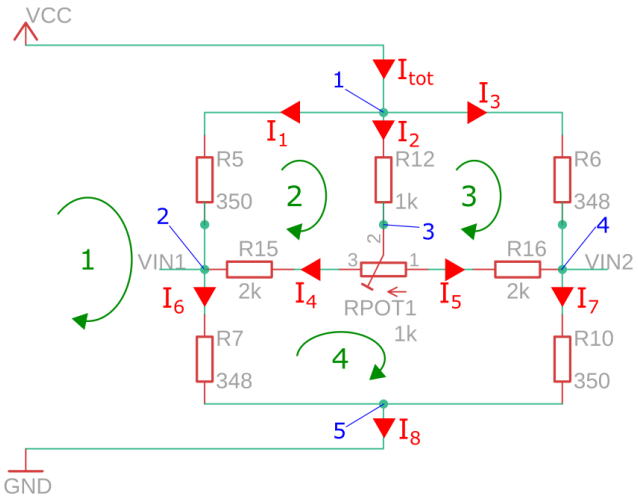
$$V_{diff} = \frac{(R_{rek}^2 - R_6R_7)}{(R_{rek} + R_6)(R_{rek} + R_7)} \cdot V_{ref} = \frac{(350.417^2 - 350 \cdot 350)}{(350.417 + 350)(350.417 + 350)} \cdot 12 = 8.57 \text{ mV} \quad (C4)$$

Benadering van weerstanden R_{15} en R_{16} met behulp van de twee wetten van Kirchhoff:

Bij de volgende berekeningen wordt de schakeling en benamingen in figuur C1 gehanteerd. Als eerst wordt de eerste wet van Kirchhoff toegepast om de stromen door alle knooppunten vast te stellen:

Knooppunt	KCL
1	$I_{tot} = I_1 + I_2 + I_3$
2	$I_6 = I_1 + I_4$
3	$I_5 = I_2 - I_4$
4	$I_7 = I_3 + I_5 = I_2 + I_3 - I_4$
5	$I_8 = I_6 + I_7 = I_1 + I_2 + I_3 = I_{tot}$

Vervolgens worden de spanningen in de gesloten lussen vastgesteld met behulp van de tweede wet van Kirchhoff:



Figuur C1: De brug van Wheatstone met benamingen en annotaties.

Lus	KVL
1	$I_1R_5 + I_6R_7 = V_{cc}$
2	$I_2R_{12} + I_4R_{pot1} - I_1R_5 + I_4R_{15} = 0$
3	$I_3R_6 - I_5R_{16} - I_5(1000 - R_{pot1}) - I_2R_{12} = 0$
4	$I_5(1000 - R_{pot1}) + I_5R_{16} + I_7R_{10} - I_6R_7 - I_4R_{15} - I_4R_{pot1} = 0$

Met het bovenstaande kunnen er een set van lineaire vergelijkingen opgesteld worden:

$$\begin{aligned} (R_5 + R_7) \cdot I_1 + & 0 \cdot I_2 + 0 \cdot I_3 + (R_7) \cdot I_4 = V_{cc} \\ (-R_5) \cdot I_1 + & (-R_{12}) \cdot I_2 + 0 \cdot I_3 + (R_{pot1} + R_{15}) \cdot I_4 = 0 \\ 0 \cdot I_1 + (-R_{16} - (1000 - R_{pot1}) - R_{12}) \cdot I_2 + & (R_6) \cdot I_3 + (R_{16} + (1000 - R_{pot1})) \cdot I_4 = 0 \\ (-R_7) \cdot I_1 + & ((1000 - R_{pot1}) + R_{16} + R_{10}) \cdot I_2 + (R_{10}) \cdot I_3 + & (- (1000 - R_{pot1}) - R_{16} - R_{10} - R_7 - R_{15} - R_{pot1}) \cdot I_4 = 0 \end{aligned}$$

Deze set van vergelijkingen kan ten slotte met een matrix berekening opgelost worden, te zien in vergelijking C5. Hieruit worden de stroomsterkten door de takken van de schakeling bekend.

$$\underbrace{\begin{bmatrix} R_5 + R_7 & 0 & 0 & R_7 \\ -R_5 & -R_{12} & 0 & R_{pot1} + R_{15} \\ 0 & -R_{16} - (R_{tot} - R_{pot1}) - R_{12} & R_6 & R_{16} + (R_{tot} - R_{pot1}) \\ -R_7 & (R_{tot} - R_{pot1}) + R_{16} + R_{10} & R_{10} & -(R_{tot} - R_{pot1}) - R_{16} - R_{10} - R_7 - R_{15} - R_{pot1} \end{bmatrix}}_A \underbrace{\begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix}}_X = \underbrace{\begin{bmatrix} V_{cc} \\ 0 \\ 0 \\ 0 \end{bmatrix}}_B \quad (C5)$$

Waarbij $X = A^{-1}B$. Nu de stroomsterkten bekend zijn kunnen alle individuele spanningen over de weerstanden berekend worden met behulp van de wet van Ohm. Dit maakt het mogelijk om V_{in1} en V_{in2} op een alternatieve methode te kunnen berekenen, gegeven in vergelijking C6.

$$V_{diff} = V_{in2} - V_{in1} = I_6 R_7 - I_7 R_{10} = (I_4 + I_4) R_7 - (I_2 + I_3 - I_4 + I_3) R_{10} \quad (C6)$$

Voor het benaderen van de weerstandswaarde R_{15} en R_{16} wordt de bovenstaande berekeningen uit gevoerd door middel van een python script. Daarbij wordt er gelust door de waarden van weerstand R_{15} en R_{16} en door potentiometer R_{pot1} , om tot de juiste curve te komen. Dit wordt in hoofdstuk 5.2.1 behandeld.

Instellen van de referentiespanningen van beide versterkingsstappen

$$V_{ref1} = V_{out1} - (V_{out1}^+ - V_{out1}^-) G_1 = 1.8 - 0.009 \frac{0.6}{0.009} = 1.2 \text{ V} \quad (C7)$$

$$V_{ref2} = \frac{1}{-2.5} V_{out2} + V_{out1} = \frac{1}{-2.5} 3 + 1.8 \approx 1.086 \text{ V} \quad (C8)$$

$$f = \frac{1}{2\pi R_{pot4} C_3} \quad (C9)$$

Het kalibreren van de loadmeter

1. R_{pot1} wordt gekalibreerd tot V_{diff} gelijk is aan 1 mV. Dit is het balanceer circuit.
2. R_G wordt gekalibreerd tot het verschil van V_{out1} tussen vollast en nullast gelijk is aan 0.6 V. Hiermee wordt gain stage 1 ingesteld.
3. R_{pot2} wordt gekalibreerd tot V_{out1} bij nullast gelijk is aan 1.2 V en bij vollast gelijk is aan 1.8 V. Dit is de offset d.m.v. V_{ref1} van stage 1.
4. R_{pot4} wordt gekalibreerd tot het verschil van V_{out2} tussen vollast en nullast gelijk is aan 2.5 V. Hiermee wordt gain stage 2 ingesteld.
5. R_{pot3} wordt gekalibreerd tot V_{out2} bij nullast gelijk is aan 0.5 V en bij vollast gelijk is aan 2.5 V. Dit is de offset d.m.v. V_{ref2} van stage 2.

Deelhoofdstuk 5.3 Software Ontwerp

In dit deelhoofdstuk worden de algoritmen voor de software filters verder toegelicht en verduidelijkt. Daarbij worden ook de keuzes voor de instelparameters en kalibratiemethoden voor elke filter in de test toegelicht.

1. Het voortschrijdend gemiddelde

De code is te vinden in listing D2.

Instelparameters: N = aantal meetpunten, $N = 15$

Vergelijking:

$$SMA_n = \frac{1}{N} \sum_{i=0}^{N-1} x_i \quad (C10)$$

Waarbij: x_i = Meetwaarde in set met index i.

2. Het gewogen voortschrijdend gemiddelde:

De code is te vinden in listing D3.

Instelparameters: N = Aantal meetpunten, $N = 15$
 w_i = Weegfactor voor waarde met index i, $w_i = \frac{N-i}{N} = 1 - \frac{i}{N}$

Vergelijking:

$$WMA_n = \frac{\sum_{i=0}^{N-1} (w_i x_{N-i})}{\sum_{i=0}^{N-1} w_i} \quad (C11)$$

Waarbij: x_{N-i} = Meetwaarde in set met index N-i

x_N = De nieuwste waarde in de reeks

x_1 = De oudste waarde in de reeks

3. Het exponentieel voortschrijdend gemiddelde:

De code is te vinden in listing D4.

Instelparameters: N = aantal meetpunten, $N = 15$
 α = wegingscoëfficiënt, $\alpha = \frac{2}{N+1}$

Vergelijking:

$$EMA_n = \frac{\sum_{i=0}^{N-1} ((1-\alpha)^{N-i-1} x_{i+1})}{\sum_{i=0}^{N-1} (1-\alpha)^{N-i-1}} \quad (C12)$$

Waarbij: x_{i+1} = Meetwaarde in set met index i+1

x_N = De nieuwste waarde in de reeks

x_1 = De oudste waarde in de reeks

4. 1€ Filter:

De code is te vinden in listing D5.

Instelparameters: $f_{c_{min}}$ = minimum cutoff frequency, $f_{c_{min}} = 0.00238$
 β = speed coefficient, $\beta = 0.00874$

Vergelijking:

zie paper [12](Casiez et al., 2012).

5. Kalman Filter:

De code is te vinden in listing D6.

Instelparameters:	KG_{min} = Ondergrens voor KG	$KG_{min} = 0$
	KG_{max} = Bovengrens voor KG	$KG_{max} = 1$
	$p_{n,0}$ = Initiële onzekerheid in voorspelling	$p_{n,0} = 11 \times 10^7$
	$r_{n,0}$ = Initiële onzekerheid in meting	$r_{n,0} = 200$
	$\hat{x}_{n,0}$ = Initiële voorspelling	$\hat{x}_{n,0} = 2500$
	z_0 = Initiële meting	$z_0 = 2500$
	k = Reset trigger niveau	$k = 2$

Vergelijkingen:**Updaten van de kalman gain:**

$$KG_{min} \leq KG \leq KG_{max} \quad KG = \begin{cases} \frac{p_{n,n-1}}{p_{n,n-1} + r_n}, & \text{als } \left| \frac{z_n - \hat{x}_{n,n-1}}{\sigma_n} \right| < k \text{ of } n < 2, \\ \frac{p_{n,0}}{p_{n,0} + r_0}, & \text{anders.} \end{cases} \quad (C13a)$$

Maken van een nieuwe voorspelling:

$$\hat{x}_{n,n} = \begin{cases} \hat{x}_{n,n-1} + KG(z_n - \hat{x}_{n,n-1}), & \text{als } \left| \frac{z_n - \hat{x}_{n,n-1}}{\sigma_n} \right| < k \text{ of } n < 2, \\ z_n, & \text{anders.} \end{cases} \quad (C13b)$$

Updaten van de onzekerheid in de voorspelling

$$p_{n,n} = \begin{cases} (1 - KG)p_{n,n-1}, & \text{als } \left| \frac{z_n - \hat{x}_{n,n-1}}{\sigma_n} \right| < k \text{ of } n < 2, \\ p_{n,0}, & \text{anders.} \end{cases} \quad (C13c)$$

Updaten van de onzekerheid in de meting

$$r_n = \sigma_n^2 \quad (C13d)$$

Waarbij: KG = Kalman Gain

n = Huidige index van meting, sinds de vorige reset.

$\hat{x}_{n,n}$ = Nieuwe voorspelling.

$\hat{x}_{n,n-1}$ = Vorige voorspelling.

σ_n = De standaarddeviatie van de meetset met n metingen, sinds de vorige reset.

σ_n^2 = De variantie van de meetset met n metingen, sinds de vorige reset.

$p_{n,n-1}$ = Vorige onzekerheid in voorspelling.

$p_{n,n}$ = Nieuwe onzekerheid in voorspelling.

z_n = Meting.

r_n = Onzekerheid in meting.

C2 Berekeningen hoofdstuk 6 De digital twin simulatie van de DOBOT

Benodigde transformatiematrixen.

DOBOT API \Rightarrow Fusion360, $\theta = 180$

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{C14})$$

Fusion360 \Rightarrow DOBOT API, $\theta = 180$

$$\begin{bmatrix} \cos(-\theta) & -\sin(-\theta) & 0 \\ \sin(-\theta) & \cos(-\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{C15})$$

DOBOT API \Leftrightarrow Unity

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (\text{C16})$$

C3 Berekeningen hoofdstuk 7 De krachtenanalyse van een last aan de DOBOT

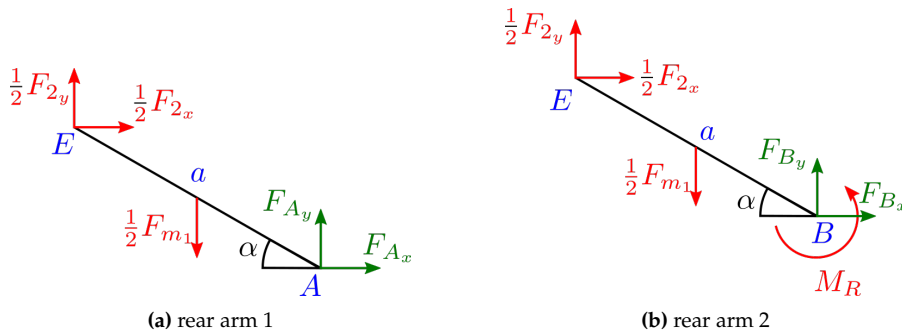
De vrijlichaamsschema's van de onderdelen van de DOBOT zijn in de onderstaande figuren te zien. De getallen naast sommige vergelijkingen is de index van de rij van matrix (C25).

Rear arm 1 & 2:

De Rear Arm heeft twee vrijlichaamsschema's omdat deze bevestigd wordt aan de Foot op twee punten zonder een doorlopende as.



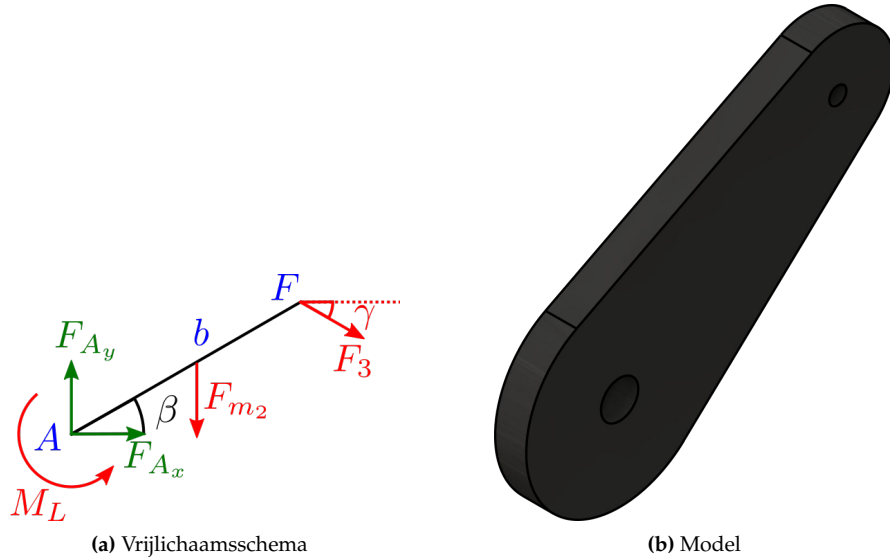
Figuur C2: CAD modellen van onderdelen rear arm en foot.



Figuur C3: Vrijlichaamsschema's van onderdeel rear arm.

$$\begin{aligned} \sum F_x &= 0; & F_{2x} + F_{Ax} + F_{Bx} &= 0 & (1) & (C17a) \\ \sum F_y &= 0; & F_{2y} - F_{m1} + F_{Ay} + F_{By} &= 0 & (12) & (C17b) \\ \sum M_{A\&B} &= 0; & \frac{1}{2} a \cos(\alpha) F_{m1} - a \sin(\alpha) F_{2x} - a \cos(\alpha) F_{2y} + M_R &= 0 & (2) & (C17c) \\ & & \frac{1}{2} a \cos(\alpha) F_{m1} - a \sin(\alpha) F_{2x} - a \cos(\alpha) F_{2y} + M_R &= 0 & & \end{aligned}$$

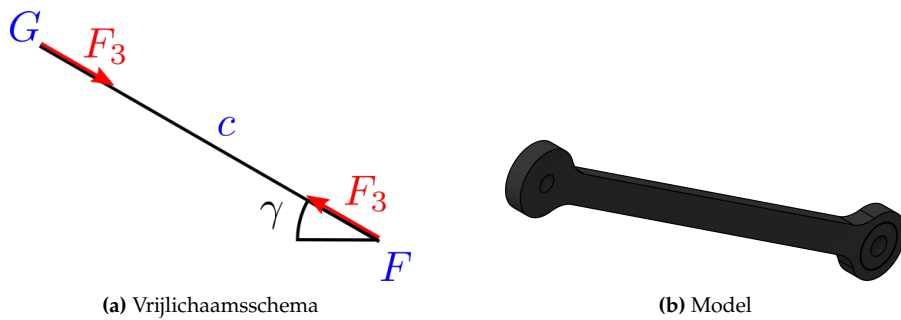
Rear arm connector 1:



Figuur C4: Vrijlichaamsschema en CAD model van onderdeel rear arm connector 1.

$$\begin{aligned} \sum F_x &= 0; & \cos(\gamma)F_3 + F_{Ax} &= 0 & (3) & (C18a) \\ \sum F_y &= 0; & -b \cos(\beta)F_{m2} - \sin(\gamma)F_3 + F_{Ay} &= 0 & (15) & (C18b) \\ \sum M_A &= 0; & -\frac{1}{2}b \cos(\beta)F_{m2} - b \cos(\beta) \sin(\gamma)F_3 - b \sin(\beta) \cos(\gamma)F_3 + M_L &= 0 & (4) & (C18c) \\ & & -\frac{1}{2}b \cos(\beta)F_{m2} - b \sin(\beta + \gamma)F_3 + M_L &= 0 & & \end{aligned}$$

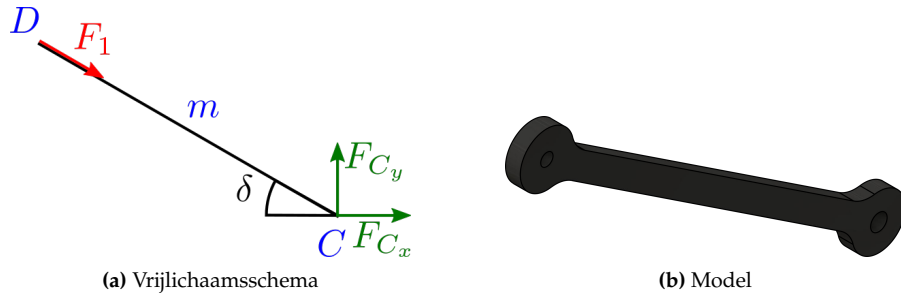
Rear arm connector 2:



Figuur C5: Vrijlichaamsschema en CAD model van onderdeel rear arm connector 2.

$$\begin{aligned} \sum F_x &= 0; & \cos(\gamma)F_3 - \cos(\gamma)F_3 &= 0 & (C19a) \\ \sum F_y &= 0; & \sin(\gamma)F_3 - \sin(\gamma)F_3 &= 0 & (C19b) \end{aligned}$$

Rear arm connector 3:



Figuur C6: Vrijlichaamsschema en CAD model van onderdeel rear arm connector 3.

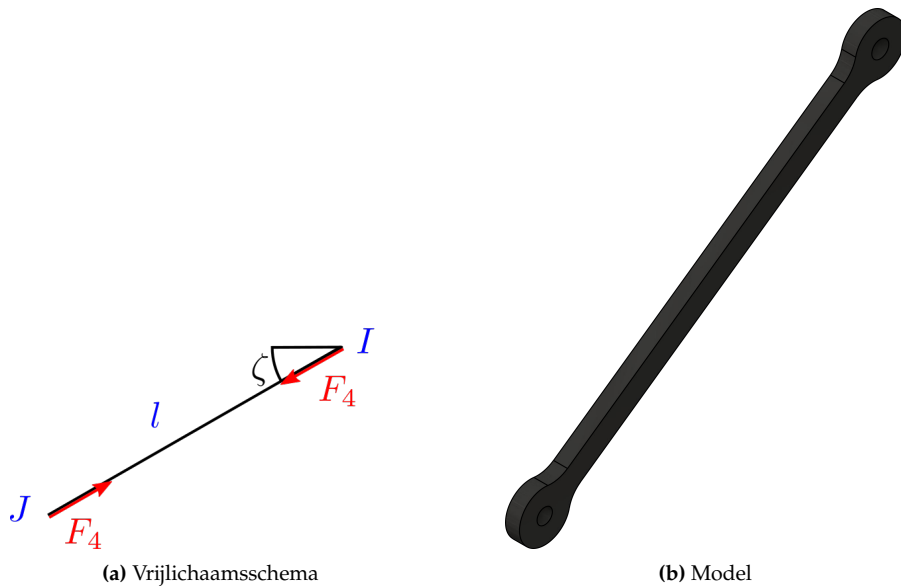
$$\sum F_x = 0;$$

$$\sum F_y = 0;$$

$$\cos(\delta)F_1 - F_{C_x} = 0 \quad (13) \quad (C20a)$$

$$\sin(\delta)F_1 - F_{C_y} = 0 \quad (14) \quad (C20b)$$

Front arm connector:



Figuur C7: Vrijlichaamsschema en CAD model van onderdeel front arm connector.

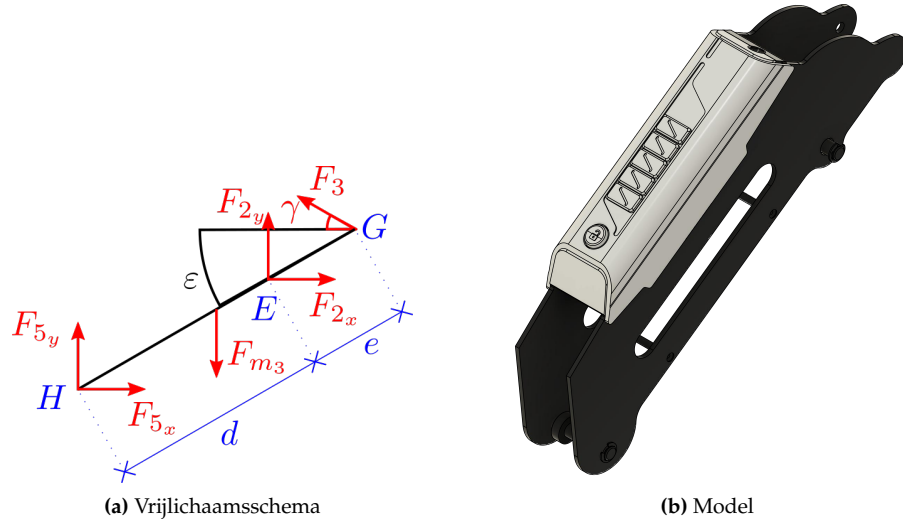
$$\sum F_x = 0;$$

$$\sum F_y = 0;$$

$$\cos(\zeta)F_4 - \cos(\zeta)F_4 = 0 \quad (C21a)$$

$$\sin(\zeta)F_4 - \sin(\zeta)F_4 = 0 \quad (C21b)$$

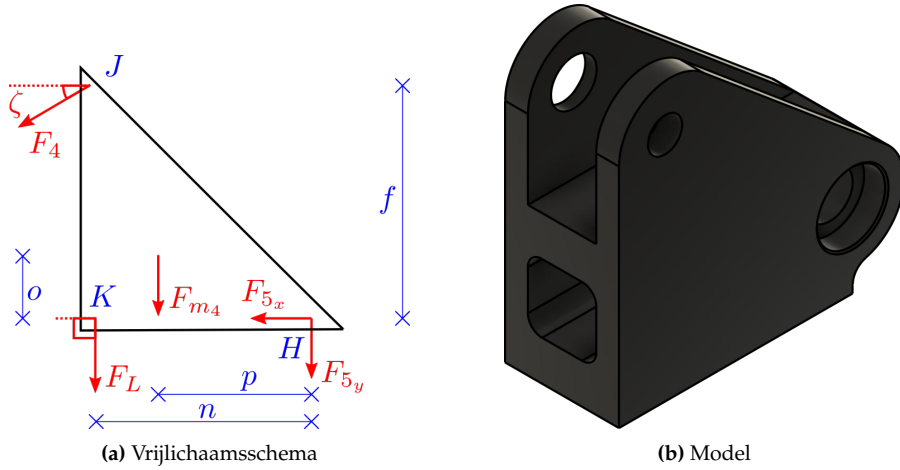
Front arm:



Figuur C8: Vrijlichaamsschema en CAD model van onderdeel front arm.

$$\begin{aligned} \sum F_x &= 0; & F_{5x} + F_{2x} - \cos(\gamma)F_3 &= 0 & (5) \quad (C22a) \\ \sum F_y &= 0; & -F_{m3} + F_{5y} + F_{2y} + \sin(\gamma)F_3 &= 0 & (C22b) \\ \sum M_H &= 0; & -\frac{1}{2}(d+e)\cos(\epsilon)F_{m3} - d\sin(\epsilon)F_{2x} + d\cos(\epsilon)F_{2y} + (d+e)\cos(\epsilon)\sin(\gamma)F_3 + & & \\ & & (d+e)\sin(\epsilon)\cos(\gamma)F_3 &= 0 & (6) \quad (C22c) \\ & & -\frac{1}{2}(d+e)\cos(\epsilon)F_{m3} - d\sin(\epsilon)F_{2x} + d\cos(\epsilon)F_{2y} + (d+e)\sin(\gamma+\epsilon)F_3 &= 0 & \end{aligned}$$

End-effector:



Figuur C9: Vrijlichaamsschema en CAD model van onderdeel end-effector.

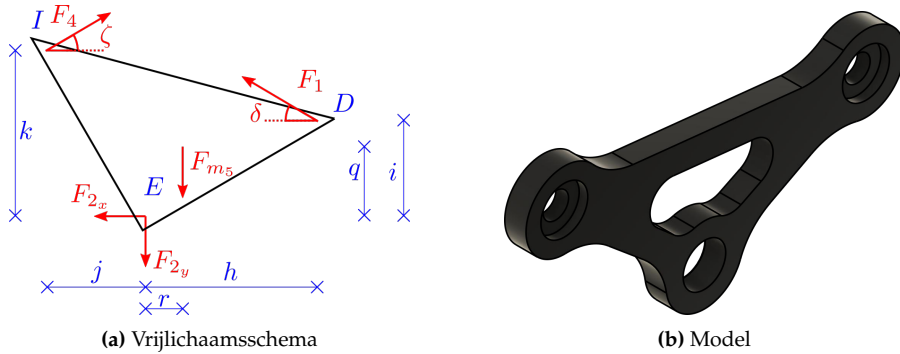
$$\begin{aligned}\sum F_x &= 0; \\ \sum F_y &= 0; \\ \sum M_H &= 0\end{aligned}$$

$$-\cos(\zeta)F_4 + F_{5x} = 0 \quad (7) \quad (C23a)$$

$$-F_{m4} - \sin(\zeta)F_4 + F_{5y} = 0 \quad (11) \quad (C23b)$$

$$pF_{m4} + nF_L + f\cos(\zeta)F_4 + n\sin(\zeta)F_4 = 0 \quad (8) \quad (C23c)$$

Wing:



Figuur C10: Vrijlichaamsschema en CAD model van onderdeel wing.

$$\begin{aligned}\sum F_x &= 0; \\ \sum F_y &= 0; \\ \sum M_E &= 0;\end{aligned}$$

$$-\cos(\delta)F_1 + \cos(\zeta)F_4 - F_{2x} = 0 \quad (9) \quad (C24a)$$

$$-F_{m5} + \sin(\delta)F_1 + \sin(\zeta)F_4 - F_{2y} = 0 \quad (C24b)$$

$$-rF_{m5} + i\cos(\delta)F_1 + h\sin(\delta)F_1 - j\sin(\zeta)F_4 - k\cos(\zeta)F_4 = 0 \quad (10) \quad (C24c)$$

Tabel C1: Variabelen in de krachtberekeningen.

variabele	waarde
a	0.135 m
b	0.043 m
c	0.135 m
d	0.147 m
e	0.043 m
f	0.028 m
h	0.038 m
i	0.020 m
j	0.033 m
k	0.028 m
l	0.147 m
m	0.135 m
n	0.033 m
o	0.003 m
p	0.020 m
q	0.017 m
r	0.002 m

(a) lengten

kracht	variabele	waarde
F_L	θ_{F_L}	-90°
F_1	θ_{F_1}	δ
F_3	θ_{F_3}	γ
F_4	θ_{F_4}	ζ
F_2	θ_{F_2}	$\arctan(\frac{F_{2y}}{F_{2x}})$
F_5	θ_{F_5}	$\arctan(\frac{F_{5y}}{F_{5x}})$
F_a	θ_{F_a}	$\arctan(\frac{F_{ay}}{F_{ax}})$
F_b	θ_{F_b}	$\arctan(\frac{F_{by}}{F_{bx}})$
F_c	θ_{F_c}	$\arctan(\frac{F_{cy}}{F_{cx}})$

(b) Hoeken van krachten

variabele	waarde
α	Uitlezen van DOBOT
β	Uitlezen van DOBOT
γ	α
δ	α
ε	β
ζ	β

(c) hoeken

Massa		Kracht	
m_1	0.600 kg	F_{m_1}	5.886 N
m_2	0.020 kg	F_{m_2}	0.196 N
m_3	0.400 kg	F_{m_3}	3.924 N
m_4	0.056 kg	F_{m_4}	0.549 N
m_5	0.018 kg	F_{m_5}	0.177 N

(d) massa's

variabele	waarde
F_L	Uitlezen van loadmeter
F_1	?
F_3	?
F_4	?
F_{2x}	?
F_{2y}	?
F_{5x}	?
F_{5y}	?
F_{ax}	?
F_{ay}	?
F_{bx}	?
F_{by}	?
F_{cx}	?
F_{cy}	?
M_L	?
M_R	?

(e) krachten/momenten

In tabel C1(c) wordt er gebruik gemaakt van de parallellogramconstructie van de DOBOT om zes hoeken

te reduceren naar twee hoeken.

Met 15 onbekenden wordt er met behulp de vergelijkingen (C17) t/m (C22) een 15x15 matrix opgesteld. Hieronder volgen de A (C25), B (C26) en X(C27) matrixen.

$$\begin{bmatrix}
 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & -a \sin \alpha & -a \cos \alpha & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & \cos \gamma & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & -b \sin(\beta + \gamma) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
 0 & -\cos \gamma & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & (d + e) \sin(\gamma + \varepsilon) & 0 & -d \sin \varepsilon & d \cos \varepsilon & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & -\cos \zeta & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & f \cos \zeta + n \sin \zeta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -\cos \delta & 0 & \cos \zeta & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 i \cos \delta + h \sin \delta & 0 & -j \sin \zeta - k \cos \zeta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & -\sin \zeta & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
 \cos \delta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
 \sin \delta & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
 0 & -\sin \varepsilon & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix} \quad (C25)$$

$$\begin{bmatrix} F_1 & F_3 & F_4 & F_{2x} & F_{2y} & F_{5x} & F_{5y} & F_{ax} & F_{ay} & F_{bx} & F_{by} & F_{cx} & F_{cy} & M_L & M_R \end{bmatrix}^T \quad (C26)$$

$$\begin{bmatrix} 0 & -\frac{1}{2} a \cos \alpha F_{m_1} & 0 & \frac{1}{2} b \cos \beta F_{m_2} & 0 & \frac{1}{2} (d + e) \cos \varepsilon F_{m_3} & 0 & -n F_L - p F_{m_4} & 0 & r F_{m_5} & F_{m_4} & F_{m_1} & 0 & 0 & b \cos \beta F_{m_2} \end{bmatrix}^T \quad (C27)$$

D Programma listings

Listing D1: Benadering Weerstanden R_{15} en R_{16} .

```

1  import matplotlib.pyplot as plt
2  import numpy as np
3
4  R_3 = 0
5  R_9 = 0
6  R_5 = 350 + 0.0025*350
7  R_10 = 350 + 0.0025*350
8  R_6 = 348 - 0.0025*348
9  R_7 = 348 - 0.0025*348
10 R_15 = 2E3
11 R_16 = 2E3
12 R_12 = 1000
13 Rpot1_tot = 1000
14 Rpot1 = 0
15 V_in = 12
16
17 x_list = []
18 y_list = []
19 legend_list = []
20 for i in range(2000,2500,3000): #range(2000,2001,1):
21     temp_x = []
22     temp_y = []
23     for j in range(0,1001,100):
24         Rpot1 = j
25         R_15 = i
26         R_16 = i
27
28         R_mat = np.array([
29             [R_3+R_9+R_5+R_7, 0, 0, R_7],
30             [-R_5, R_12, 0, Rpot1 + R_15],
31             [0, -R_16-(Rpot1_tot-Rpot1)-R_12, R_6, R_16+(Rpot1_tot-Rpot1)],
32             [-R_7, (Rpot1_tot-Rpot1)+R_16+R_10, R_10, -(Rpot1_tot-Rpot1)-R_16-R_10-R_7-R_15-Rpot1]
33         ])
34         U_mat = np.array([
35             [-V_in],
36             [0],
37             [0],
38             [0],
39         ])
40
41         I_mat = np.matmul(np.linalg.inv(R_mat),U_mat)
42         I_1 = np.abs(I_mat[0][0])
43         I_2 = np.abs(I_mat[1][0])
44         I_3 = np.abs(I_mat[2][0])
45         I_tot = np.abs(I_mat[0][0] + I_mat[1][0] + I_mat[2][0])
46
47         temp_x.append(j)
48         Vin1 = 12-I_tot*R_3-I_1 * R_5
49         Vin2 = 12-I_tot*R_3-I_3 * R_7
50
51         temp_y.append(Vin2-Vin1)
52         x_list.append(temp_x)
53         y_list.append(temp_y)
54         legend_list.append(i)
55
56 for list in range(0, len(y_list)):
57     plt.plot(x_list[list], y_list[list], label="$R_{15}$ & $R_{16}$ = " +
58             str(legend_list[list]) + "$\Omega$")
59
60
61 plt.grid()
62 plt.legend()
63 plt.title('Kalibratiebereik van de nullast in scenario A')
64 plt.xlabel('Weerstand $R_{pot1}$ in [$\Omega$]')
65 plt.ylabel('Spanning $V_{diff}$ in [V]')
66 plt.gcf().set_dpi(300)
67 plt.show()

```

Listing D2: Berekening van het simpel voortschrijdend gemiddelde.

```
1 def simple_running_mean(x, N):
2     cumsum = 0
3     if(len(x) >= N):
4         for i in range(0,N):
5             cumsum += x[-movingAverageFrame:][i]
6         return cumsum/N
7     else:
8         for i in range(0,len(x)):
9             cumsum += x[-len(x):][i]
10        return cumsum/len(x)
```

Listing D3: Berekening van het gewogen voortschrijdend gemiddelde.

```
1 def weighted_running_mean(x, N):
2     cumsum = 0
3     w = np.flip(1 - np.arange(0, 1, 1/N))
4     if(len(x) >= N):
5         for i in range(0,N):
6             cumsum += x[-movingAverageFrame:][i] * w[-N:][i]
7         return cumsum/sum(w[:len(x)])
8     else:
9         for i in range(0,len(x)):
10            cumsum += x[-len(x):][i] * w[-len(x):][i]
11        return cumsum/sum(w[:len(x)])
```

Listing D4: Berekening van het exponentieel voortschrijdend gemiddelde.

```
1 def exponential_running_mean(x, N, alpha):
2     Num = 0
3     Den = 0
4     if(len(x) >= N):
5         for i in range(0,N):
6             Num += np.power((1-alpha), (N-i)-1) * x[-movingAverageFrame:][i]
7             Den += np.power((1-alpha), (N-i)-1)
8         return Num/Den
9     else:
10        for i in range(0,len(x)):
11            Num += np.power((1-alpha), (len(x)-i)-1) * x[-len(x):][i]
12            Den += np.power((1-alpha), (len(x)-i)-1)
13        return Num/Den
```

Listing D5: Werking van de 1€ filter. [1]

```
1 def smoothing_factor(t_e, cutoff):
2     r = 2 * math.pi * cutoff * t_e
3     return r / (r + 1)
4
5
6 def exponential_smoothing(a, x, x_prev):
7     return a * x + (1 - a) * x_prev
8
9
10 class OneEuroFilter:
11     def __init__(self, t0, x0, dx0=0.0, min_cutoff=1.0, beta=0.0,
12                 d_cutoff=1.0):
13         """Initialize the one euro filter."""
14         # The parameters.
15         self.min_cutoff = float(min_cutoff)
16         self.beta = float(beta)
17         self.d_cutoff = float(d_cutoff)
18         # Previous values.
19         self.x_prev = float(x0)
20         self.dx_prev = float(dx0)
21         self.t_prev = float(t0)
22
23     def __call__(self, t, x):
24         """Compute the filtered signal."""
25         t_e = t - self.t_prev
26
27         # The filtered derivative of the signal.
28         a_d = smoothing_factor(t_e, self.d_cutoff)
29         dx = (x - self.x_prev) / t_e
30         dx_hat = exponential_smoothing(a_d, dx, self.dx_prev)
31
32         # The filtered signal.
33         cutoff = self.min_cutoff + self.beta * abs(dx_hat)
34         a = smoothing_factor(t_e, cutoff)
35         x_hat = exponential_smoothing(a, x, self.x_prev)
36
37         # Memorize the previous values.
38         self.x_prev = x_hat
39         self.dx_prev = dx_hat
40         self.t_prev = t
41
42     return x_hat
```

Listing D6: Werking van het kalman filter. [2]

```

1 def kalmanUpdate(_ErrorEstimate, _ErrorMeasurement, _Estimate, _Measurement, _minMaxKG = [0,1]):
2     _newEstimate = 0.
3     _newEstimateError = 0.
4
5     #1. Update Kalman gain.
6     _KG = var_check(_ErrorEstimate/(_ErrorEstimate+_ErrorMeasurement),_minMaxKG[0], _minMaxKG[1])
7
8     #2. Make new estimate.
9     _newEstimate = _Estimate + _KG*(_Measurement-_Estimate)
10
11    #3. Update error in estimate.
12    _newEstimateError = (1 - _KG)*_ErrorEstimate
13
14    return [_KG, _newEstimate, _newEstimateError]
15
16 #Code in main loop
17 distance = Measurement[-1]-KalmanEstimates[-1]
18 stdev_set = np.std(Measurement[-stepsAfterReset:])
19 if(np.std(Measurement[-stepsAfterReset:]) != 0 and
20    abs((distance)/(stdev_set)) > kalman_reset_treshhold and
21    stepsAfterReset>kalmanResetImmunity):
22     KalmanEstimatesErrors[-1] = InitErrorEstimate
23     KalmanEstimates[-1]= Measurement[-1]
24     stepsAfterReset = 1
25 else:
26     ret = kalmanUpdate(KalmanEstimatesErrors[-1],
27                        ErrorMeasurement,
28                        KalmanEstimates[-1],
29                        Measurement[-1],
30                        [0, 1])
31     KG = ret[0]
32     KalmanEstimates.append(ret[1])
33     KalmanEstimatesErrors.append(ret[2])
34     ErrorMeasurement = np.var(Measurement[-stepsAfterReset:]).item()

```

E Opdrachtvoorstel

Naam student:	Ömer Sekmen	Bedrijf:	Batenburg JB Systems
Studentnummer:	18022618	Adres bedrijf:	Westlandseweg 190 3131 HX Vlaardingen
Afstudeerperiode:	Shift 3 7 februari 2022 – 03 juni 2022	Afdeling:	Research and Development
Bedrijfsbegeleider	Ir. Tim Verwaal	Functie bedrijfsbegeleider & opleidingsniveau	R&D Engineer WO Mechanical engineering + 15 jaar werkervaring

Achtergrond:

JB Systems richt zich voornamelijk op het ontwikkelen van industriële software-oplossingen in diverse markten. De software oplossingen kunnen variëren van besturingssystemen van een machine door middel van een PLC/SCADA oplossing, tot het maken van simulaties waarop de operationele werking van diverse offshore apparatuur gesimuleerd kan worden voor trainingsdoeleinde.

De afstudeeropdracht maak deel uit van afdeling R&D. Hier worden onder andere de eerder genoemde simulaties ontwikkelt. Daarnaast houdt de afdeling zich ook bezig met nieuwe technologieën zoals: **Robotics, Big data, Machine vision, Machine learning, Haptics, VR/AR, Digitization, Test automation**. Het doel is om de benoemde ontwikkelingen terug te koppelen naar een product waar de klant gebruik van kan maken of om vervolg ontwikkelingen van JB systems te ondersteunen.

Aanleiding:

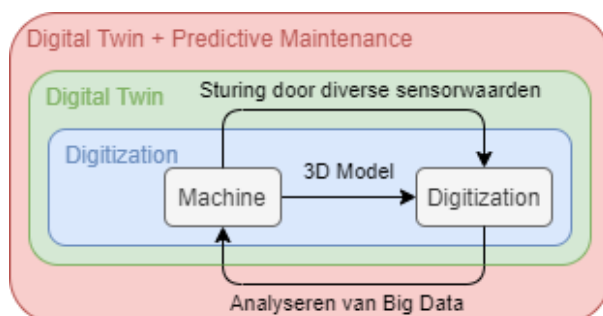
De aanleiding van de opdracht is om een stap te zetten naar het preventief onderhouden (predictive maintenance) van machines. Om hier een begin aan te kunnen maken moet het bestaande ontwerp eerst gedigitaliseerd worden, door bijvoorbeeld een digital twin. Het is belangrijk dat de digital twin ook feedback krijgt van het bestaande ontwerp.

Tijdens de opdracht wordt er gebruik gemaakt van de DOBOT Magician¹, te zien in figuur 1.



Figuur 1 Dobot Magician

In het geval van de DOBOT zou de digital twin onder andere de hoeken van de armen of de belaste load als input krijgen, zodat de robotarm een-op-een overgenomen kan worden in de simulatie.



Figuur 2 lagen-model predictive maintenance

In figuur 2 is er op een versimpelde manier weergegeven wat de benodigde stappen zijn om een vorm van predictive maintenance toe te kunnen passen.

Laag 1: Digitization

Het ontwerp is in een 3D engine, zoals Unity, gemodelleerd en kan bewegen als de werkelijke machine.

¹ <https://www.dobot.cc/dobot-magician/product-overview.html>

Laag 2: Digital twin

Het model wordt gestuurd door middel van meerdere ingangssignalen die uitgelezen worden van de machine. Het mee simuleren van de niet-sturing gerelateerde meetwaarde zoals temperaturen of loadkrachten valt ook binnen deze laag. De digitale machine leeft mee met de werkelijke machine.

Laag 3: Digital twin + predictive maintenance

Tenslotte kan er een begin worden gemaakt aan de vervolg stap: predictive maintenance. Hier worden historische sensordata geanalyseerd om een conclusie te kunnen trekken over de onderhoudsstaat van een machine. Deze analyse wordt vervolgens teruggekoppeld aan de machine in de vorm van een tijdig onderhoudsbeurt.

Doelstelling:

Het doel is om het model van de DOBOT aan het eind van het afstudeertraject minstens tot laag 2 te ontwikkelen. Daarbij wordt een deel analyse van laag 3 meegenomen in de ontwikkelingen. Dit zijn vooral sterkte berekeningen om te simuleren hoe en waar de DOBOT belast wordt. Het idee is om, bij een willekeurige beweging met een load, doorbuigingen en kritische mechanische zwaktepunten zichtbaar te maken door middel van een FEM analyse.

Een overzicht van de drie hoofdtaken die omvat worden door de afstudeeropdracht zijn hieronder, per vakgebied, in een lijst weergegeven. Hierbij zijn de startsituaties en de mogelijke oplossingen die uiteindelijk opgeleverd worden aangegeven.

1. Een loadmeting uitvoeren:

Elektrotechnisch: Om belastingen aan de end-effector te kunnen meten zal er een loadmeting plaats vinden. Dit wordt gedaan met een rekstrookje. Hiervoor zal er een elektrische schakeling ontworpen moeten worden, waar bijvoorbeeld het signaal versterkt kan worden of waarbij het nul-last signaal gekalibreerd kan worden.

Mechanisch: Er moet een onderdeel ontworpen worden waar het rekstrookje aan bevestigd kan worden. De testgewichten dienen aan dit onderdeel te hangen.

2. Digital twin simuleren:

Informatica: De DOBOT moet overgezet worden naar een 3D omgeving, JB systems gebruikt hierbij Unity. Het model dient dicht bij de werkelijkheid te komen, waarbij de simulatie gestuurd wordt door sensoren op de robotarm. 3D modellen van de robot zijn beschikbaar, het overtekenen van onderdelen valt dus buiten de opdracht.

Ook wordt er een algemene beweging geprogrammeerd, waarbij de robot in een zo groot mogelijk deel van zijn bereik beweegt. Deze beweging dient als een stress-test en vormt een basis voor de volgende taak.

3. Stress analyses toepassen:

Mechanisch: Als laatst worden de mechanische sterktes van de robotarm geanalyseerd. Dit wordt gedaan door middel van een FEM simulatie met softwareprogramma's zoals PrePoMax² + CalculiX³ (Of andere soortgelijke programma's).

Het liefst worden de uitkomsten van deze analyses geëxporteerd naar Unity, waar de maximale belastingen real-time weergegeven kunnen worden.

² <https://prepomax.fs.um.si/>

³ <http://www.calculix.de/>

Ruwe startplanning met mijlpalen:

Op de volgende pagina is er een ruwe planning te vinden met een algemene lijst van taken die uitgevoerd zullen worden. Deze planning zal gedurende het afstuderen veranderen.

[illegible]