

# Ontwikkelen van een multifunctionele robotarm voor sensing en acting op een autonome fieldrobot

**Matthieu SERGEANT**

Promotoren:

Ing. Rico Claeys

Ing. Klaas Philips

Co-promotoren:

Dr. Ing. Tim Van De Gucht

Dr. Ir. Simon Cool

Masterproef ingediend tot het behalen van de  
graad van master of Science in de industriële  
wetenschappen: *Elektromechanica,*  
*Mechatronica*

Academiejaar 2017-2018



# Ontwikkelen van een multifunctionele robotarm voor sensing en acting op een autonome fieldrobot

**Matthieu SERGEANT**

Promotoren:

Ing. Rico Claeys

Ing. Klaas Philips

Co-promotoren:

Dr. Ing. Tim Van De Gucht

Dr. Ir. Simon Cool

Masterproef ingediend tot het behalen van de  
graad van master of Science in de industriële  
wetenschappen: *Elektromechanica,*  
*Mechatronica*

Academiejaar 2017-2018

© Copyright KU Leuven

Zonder voorafgaande schriftelijke toestemming van zowel de promotor(en) als de auteur(s) is overnemen, kopiëren, gebruiken of realiseren van deze uitgave of gedeelten ervan verboden. Voor aanvragen i.v.m. het overnemen en/of gebruik en/of realisatie van gedeelten uit deze publicatie, kan u zich richten tot KU Leuven Technologiecampus Gent, Gebroeders De Smetstraat 1, B-9000 Gent, +32 92 65 86 10 of via e-mail [iiw.gent@kuleuven.be](mailto:iiw.gent@kuleuven.be).

Voorafgaande schriftelijke toestemming van de promotor(en) is eveneens vereist voor het aanwenden van de in deze masterproef beschreven (originele) methoden, producten, schakelingen en programma's voor industrieel of commercieel nut en voor de inzending van deze publicatie ter deelname aan wetenschappelijke prijzen of wedstrijden.



Eindverhandeling tot het verkrijgen van de graad van master of Science in de industriële wetenschappen Elektromechanica.

Academiejaar 2017 - 2018

Opleiding Elektromechanica, faculteit industriële ingenieurswetenschappen Technologicampus Gent

Deze masterproef verliep in samenwerking met het Instituut voor Landbouw-, Visserij- en Voedingsonderzoek.

door

Matthieu Sergeant

### ***Samenvatting.***

Automatisering en precisieapplicaties kennen een opmars binnen de landbouw, zowel om ecologische als economische redenen. Gedreven door deze evolutie wordt er volop onderzoek uitgevoerd naar nieuwe precisielandbouwtechnologie. Een van de trends is het gebruik van veldrobots, die door hun lage gewicht de bodem minder verstoren en autonoom en dus arbeidsbesparend kunnen werken.

De opdracht van deze masterproef was tweedelig. Er werd gestart van een robot die initieel enkel bestond uit een onderbouw, bestaande uit een uitschuifbaar frame met rupsaandrijving. Om deze robot in het veld te kunnen inzetten was er nog nood aan een bovenbouw waarop verschillende werktuigen kunnen bevestigd worden. In het eerste deel van deze masterproef werd dit ontworpen en ontwikkeld. Voor gebruik op het veld is er echter nog nood aan werktuigen. In het tweede deel van deze masterproef werd een robotarm ontworpen, om het oogsten van producten toe te laten, maar ook om sensoren in een gewenste positie te brengen of om stalen te nemen van de bodem of het gewas.

Om de robotarm een zo groot mogelijk gebruiksspectrum te geven werd er gekozen om de arm van zes vrijheidsgraden te voorzien. Na het ontwerp werden alle stukken vervaardigd om vervolgens geassembleerd te worden. Om de arm naar een positie te kunnen sturen, was het noodzakelijk om de voorwaartse en inverse kinematica uit te werken. Deze werden volledig wiskundig uitgewerkt binnen de praktische grenswaarden van het bereik. Hierdoor kan de robotarm, aan de hand van de gevonden kinematica, volledig geprogrammeerd worden voor gebruik op het veld.



Eindverhandeling tot het verkrijgen van de graad van master of Science in de industriële wetenschappen Elektromechanica.

Academiejaar 2017 - 2018

Opleiding Elektromechanica, faculteit industriële ingenieurswetenschappen Technologicampus Gent

Deze masterproef verliep in samenwerking met het Instituut voor Landbouw-, Visserij- en Voedingsonderzoek.

door

Matthieu Sergeant

### ***Abstract.***

Automation and precision applications are on the rise in agriculture, both for ecological and economical reasons. Driven by this evolution, a lot of research is being done into new precision farming technologies. One of the trends is the use of field robots, which, due to their low weight, are less disruptive to the soil and can work autonomously and therefore labor-saving.

The assignment of this master's thesis consisted of two parts. The basis of the thesis was a robot which was initially only composed of a substructure, made up of a telescopic frame on caterpillars. In order to be able to use this robot in the field, there was still need for a superstructure where various tools could be attached to. In the first part of this master's thesis this was designed and subsequently manufactured.

However, for use in the field, there is still a need for tools. In the second part of this master's thesis a robotic arm was designed to allow the harvesting of products, but also to place sensors in a desired position or to take samples from the soil or the crop.

The question now was, how many degrees of freedom are required for the widest possible spectrum of use? It was decided to provide the arm with six degrees of freedom because the biggest possible workspace could be reached. After the design, all the pieces were manufactured and then assembled. To be able to send the robot arm to a certain position, the forward and inverse kinematics were required. These were developed completely mathematically within the practical limits of the range of the arm. This allows the robot arm, based on the kinematics that were found, to be fully programmed for use on the field.

Keywords:

Robotic arm, autonomous fieldrobot, agriculture, precision farming, farming automation



## **VOORWOORD / BEDANKING**

In samenwerking met KULeuven en ILVO kwam deze masterproef tot stand. Tijdens het schrijven van deze thesis werd ik door de beide partijen ondersteund en leerde ik veel bij. Daarvoor wil ik ook mijn dank uitdrukken. In het bijzonder wil ik mijn promotoren aan ILVO bedanken, Tim Van De Gucht en Simon Cool, voor hun goede begeleiding en hulp bij het maken van dit eindwerk. Zeker wil ik ook de promotoren van KULeuven, Rico Claeys en Klaas Philips, bedanken voor hun goede raad en voor het beantwoorden van mijn vragen. Een speciale bedanking wil ik nog uitschrijven voor Koert Bruggeman, die mij altijd te woord stond bij vragen over de berekening van de robotica. Tot slot wil ik ook iedereen bedanken die geholpen heeft bij de praktische uitvoering van deze masterproef, waaronder Tim De Bock, Jan Van Mullem en het andere technisch personeel van ILVO.

Matthieu Sergeant



## **AANSPRAKELIJKHEID**

Deze resultaten kunnen door ILVO of zijn aangestelde, worden gebruikt onder eigen verantwoordelijkheid. Noch KULeuven, noch zijn medewerkers, noch de auteur van deze masterproef kunnen echter op welke wijze dan ook aansprakelijk gesteld worden voor welke gevolgen dan ook van het gebruik van resultaten van dit werk, hetzij rechtstreeks of onrechtstreeks. In het bijzonder kan niet worden gegarandeerd dat de veiligheid voldoende verzekerd is op basis van voorliggende tekst en is de conformiteit met de wetgeving niet gegarandeerd.

Lotenhulle, 30-04-2018

## OPDRACHTOMSCHRIJVING

ILVO, het instituut voor Landbouw-, Visserij- en voedingsonderzoek voert onder andere onderzoek uit naar precisielandbouw. Dit vanuit de nood om efficiënter aan landbouw te doen. Door een doorgedreven automatisering en mechanisatie is het mogelijk om akkerbouw zowel economisch als ecologisch voordeliger te maken.

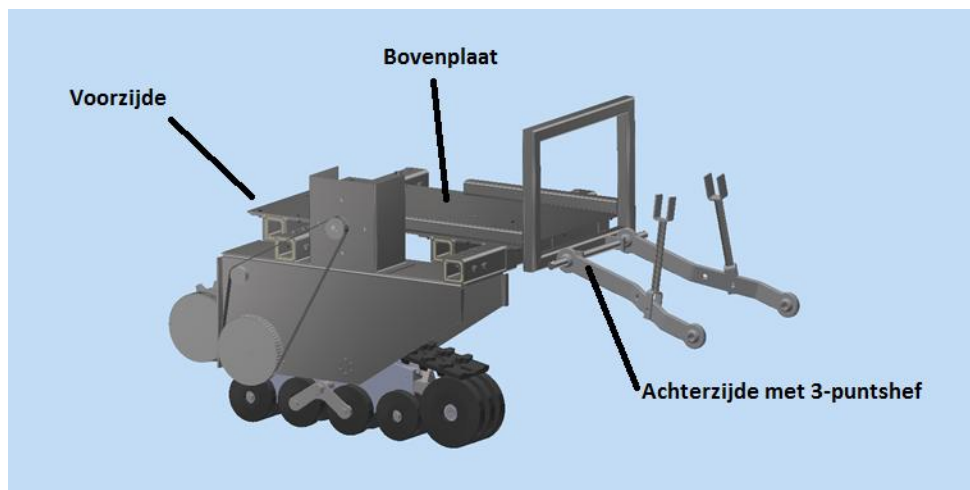
# ILVO

Instituut voor Landbouw-,  
Visserij- en Voedingsonderzoek

**Figuur iii-1 Logo ILVO [1]**

Een van de projecten is de AgroBot, een onderzoek dat reeds over verschillende jaren loopt en waarbij reeds enkele studenten van KU Leuven via hun masterproef aan meegeholpen hebben. De AgroBot is een autonoom navigerende robot voor toepassingen op het veld en kan zowel sensing (monitoren van bodem en gewas via sensoren) als acting taken hebben. Positiebepaling gebeurt via een RTK-GPS<sup>1</sup> systeem en een IMU<sup>2</sup>. Verder zal ook beeldverwerking geïmplementeerd worden om tussen de rijen te kunnen navigeren.

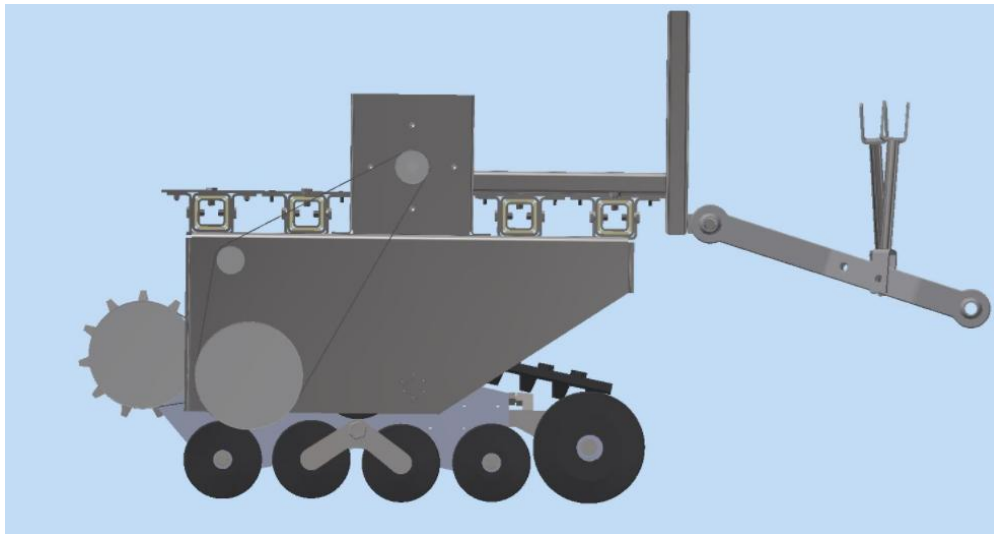
De vorige generaties waren klein en enkel geschikt voor indoor gebruikt (sensing platform, verkennend onderzoek). De 3<sup>e</sup> generatie zal groter en krachtiger zijn, zodat dat de robot effectief op het veld kan ingezet worden om landbouwkundige taken uit te voeren. Deze robot zal nog steeds een prototype zijn en dient als proof - of - concept. Men wil aantonen wat de mogelijkheden zijn van een dergelijke robot en dit aan de hand van onderzoek op het veld zelf. De robot kan met verschillende sensoren en machines uitgerust worden om een multi-inzetbaarheid te garanderen. De robot is momenteel nog in ontwikkeling.



**Figuur iii-2 3D – Zicht AgroBot V3 (Rups slechts aan één zijde getekend)**

<sup>1</sup> Real Time Kinematic – Global Positioning System

<sup>2</sup> Inertial Measurement Unit



**Figuur iii-3 Zijaanzicht AgroBot V3**

Dit jaar worden er verschillende werktuigen onderzocht:

- Strooimachine
- Spuitmachine
- Schoffelmachine
- Multifunctionele robotarm

Deze masterproef zal de ontwikkeling van de multifunctionele robotarm behandelen.

Een robotarm kan de veldrobot toelaten om groenten en fruit te oogsten, stalen te nemen van de bodem of het gewas of om sensormetingen uit te voeren waarbij een hogere beweeglijkheid vereist is. Ook onkruidbestrijding is een mogelijke toepassing. De arm moet zo multifunctioneel mogelijk zijn, verschillende eindactuatoren moeten kunnen gemonteerd worden.

Om de machines aan de robot te bevestigen op een gestandaardiseerde manier werd gekozen voor een driepuntshefinrichting. De bevestiging van de driepuntheef aan de robot is het eerste deel van de opdracht van deze masterproef. Eens dit initiële werk uitgevoerd is, moet vervolgens de robotarm uitgewerkt worden aan de hand van volgende randvoorwaarden:

**Tabel iii-1 Randvoorwaarden Robotarm**

	<b>Randvoorwaarden</b>
<b>Reikwijdte</b>	1 m
<b>Payload</b>	2 kg
<b>Budget</b>	€1,500
<b>Extra</b>	Mobiliteit

Als laatste randvoorwaarde dient de mobiliteit in de gaten gehouden te worden. De robotarm moet namelijk aan de autonome veldrobot kunnen bevestigd worden.

Vooraf zal bekeken worden als aan de hand van deze randvoorwaarden een robotarm kan aangekocht worden. Indien dit niet het geval is, dan zal de robotarm met behulp van CAD<sup>3</sup> getekend worden.

Indien de robotarm zelf ontworpen wordt, dan zal de nadruk vooral liggen op de assemblage en een korte programmering om de robotarm te testen. In het geval van aankoop zal de nadruk liggen op het monteren van de arm op de AgroBot en de programmering van de robotarm.

Bij deze masterproef is geen geheimhouding vereist.

---

<sup>3</sup> Computer Aided Design



# INHOUD

Voorwoord / Bedanking .....	i
Aansprakelijkheid .....	ii
Opdrachtoomschrijving .....	iii
Mijlpalenplanning .....	vi
Inhoud .....	vii
1 Inleiding .....	1
1.1 Algemeen .....	1
1.2 Bedrijfsvoorstelling .....	1
2 Literatuurstudie .....	3
2.1 Precisielandbouw .....	3
2.1.1 Wat is precisielandbouw? .....	3
2.1.2 Noodzaak aan precisielandbouw .....	4
2.1.3 Opsplitsing precisielandbouw .....	4
2.2 Machines in de precisielandbouw .....	6
2.2.1 Algemene automatisering bij tractorfabrikanten .....	6
2.2.2 Fieldrobots .....	7
2.3 ILVO AgroBot .....	9
2.3.1 Eerdere projecten .....	9
2.3.2 Derde generatie AgroBot - in productie .....	11
2.4 Robotarmen .....	14
2.4.1 Mogelijke systemen voor een robotarm .....	14
2.4.2 Robotarmen op de markt .....	22
2.4.3 Robotarmen in de landbouw .....	25
3 Probleemstelling & Doel .....	28
4 Bovenbouw AgroBot V3 .....	29
4.1 Ontwerp .....	29
4.1.1 Algemeen .....	29
4.1.2 Aanhechting werktuigen .....	29
4.1.3 Materiaalkeuze en bewerkingstechniek .....	30
4.1.4 Afmetingen .....	30
4.2 Eindresultaat ontwerp .....	31
4.3 Praktisch eindresultaat .....	31
5 AgroBot A.R.M. ....	32

5.1	Keuze type arm.....	32
5.1.1	Factoren.....	32
5.1.2	Score verschillende robotarmen.....	33
5.2	Benodigde vrijheidsgraad bepalen.....	33
5.2.1	Enkele mogelijke toepassingen.....	34
5.2.2	Conclusie.....	34
5.3	Ontwerp.....	35
5.3.1	Randvoorwaarden.....	35
5.3.2	Uiterlijk.....	35
5.3.3	Uiteindelijke ontwerp.....	36
5.3.4	Praktisch eindresultaat.....	38
6	Praktische uitwerking en berekeningen (A.R.M.).....	40
6.1	Materiaalkeuze.....	40
6.1.1	Frame.....	40
6.1.2	Assen (verbindingsflenzen).....	40
6.1.3	Andere componenten.....	40
6.2	Motoren.....	41
6.2.1	Statisch koppel.....	41
6.2.2	Dynamisch koppel.....	43
6.2.3	Conclusie berekeningen.....	50
6.2.4	Motoren.....	51
6.3	Controle overbrengingen.....	53
6.3.1	Tandriem- en kettingoverbrenging.....	53
6.3.2	Bout - en spieoverbrengingen.....	56
6.4	Lagers.....	61
6.4.1	Type.....	61
6.4.2	Montage.....	62
6.5	Sturing en programmering.....	64
6.5.1	Programmeeromgeving: Arduino.....	64
6.5.2	Controllino Mega.....	64
6.6	Drivers.....	65
6.6.1	Algemeen.....	65
6.6.2	Keuze drivers.....	66
6.7	Elektrische aansluiting.....	67
6.7.1	Schema.....	67
6.7.2	Korte toelichting aansluitingen.....	68

6.8	Veiligheidsvoorschriften .....	68
6.8.1	Noodstop.....	68
6.8.2	Andere veiligheidsinrichtingen.....	68
7	Bepalen van de voorwaartse en inverse kinematica van de AgroBot A.R.M. ....	70
7.1	Inleiding .....	70
7.2	Voorwaartse kinematica .....	71
7.2.1	Kinematische keten .....	71
7.2.2	Denavit-Hartenberg (DH) conventie .....	73
7.2.3	Denavit-Hartenberg (DH) parameters.....	76
7.2.4	Transformatie tussen twee aangrenzende coördinatenframes .....	79
7.2.5	Eigenlijke berekening van de voorwaartse kinematica.....	83
7.2.6	Numeriek voorbeeld voorwaartse kinematica .....	87
7.3	Inverse Kinematica:.....	93
7.3.1	Kinematische ontkoppeling.....	93
7.3.2	Grenswaarden rotatiehoeken & definiëring kwadranten .....	94
7.3.3	Beginparameters.....	96
7.3.4	Omrekenen van Tool Center Point tot pols.....	98
7.3.5	Inverse positiekinematica .....	98
7.3.6	Inverse oriëntatiekinematica.....	106
7.3.7	Numeriek voorbeeld inverse kinematica .....	107
7.4	Conclusie voorwaartse en inverse kinematica.....	112
8	Testen.....	113
8.1	Algemeen.....	113
8.2	Controle van de draairichting .....	113
8.3	Controle van het houdkoppel .....	113
8.4	Sturen van de robot naar een bepaalde positie .....	114
8.5	Herhaalbaarheid .....	115
8.6	Speling.....	116
8.7	Conclusie testen .....	116
9	Kostprijscalculatie Robotarm.....	117
10	Projecten en bemerkingsen .....	118
10.1	Projecten voor volgende jaren.....	118
10.1.1	Zes-assig programmeren .....	118
10.1.2	Invoeren kalibrering en beginpositie .....	118
10.1.3	Koppelen visiesysteem.....	118
10.1.4	Toepassingsgericht programmeren .....	118



10.1.5 Invoeren van extra veiligheidsmaatregelen.....	118
10.2 Mogelijke aanpassingen.....	119
10.2.1 Keuze motoren.....	119
10.2.2 Keuze overbrengingen .....	119
10.2.3 Lagermontage .....	119
10.2.4 Montage armdelen .....	119
10.3 Bemerkingen.....	119
11 Conclusie.....	121
12 Bibliografie.....	122
13 Opmerkingen .....	128
Bijlage A Promotoren.....	A1
Bijlage B Lijst met gegevens voor de berekening van het koppel.....	B1
Bijlage C Berekening bouten & spieën .....	C1
Bijlage D Elektrisch schema .....	D1
Bijlage E MatLab code.....	E1
1) Voorwaartse kinematica: positiebepaling (symbolisch).....	E1
2) Voorwaartse kinematica positiebepaling en Eulerhoeken (getalwaarden) .....	E2
3) MatLab code berekenen totale rotatiematrix (symbolisch).....	E3
4) MatLab code berekening inverse oriëntatiekinematica .....	E4
Bijlage F Extra voorbeelden inverse positiekinematica .....	F1
1) Controle van $\theta_1$ in kwadrant 4B.....	F1
2) Controle verschillende oriëntaties met wijzigende $\theta_2$ en $\theta_1 = 30^\circ$ .....	F1
3) Controle verschillende oriëntaties met wijzigende $\theta_2$ en $\theta_1 = -30^\circ$ .....	F3
Bijlage G Arduino testprogramma.....	G1
Bijlage H Lijst met alle gebruikte materialen voor de kostprijsanalyse .....	H1
Bijlage I Technische dossiers.....	I1



# 1 INLEIDING

---

## 1.1 Algemeen

Zowel automatisering en precisieapplicaties kennen een opmars binnen de landbouw, gedreven door economische en ecologische redenen. Het 'Instituut voor Landbouw-, Visserij- en Voedingsonderzoek' (ILVO) voert volop onderzoek uit binnen deze topics. Eén van de trends is het gebruik van veldrobots, die door hun lage gewicht bodemverdichting vermijden en autonoom en hierdoor dus arbeidsbesparend kunnen werken. Naast het oplossen van de problematiek van de bodemverdichting, kunnen veldrobots nog veel andere toepassingen hebben. Ze kunnen 's nachts gebruikt worden en met meerdere tegelijk. Zo hoeft de landbouwer niet te investeren in grote, zware machines om tijdig zijn veld te kunnen bewerken. Bovendien kunnen robots preciezer werken.

In 2010 richtte ILVO daarom het AgroBot-project op. Hierin wordt onderzocht wat de mogelijkheden zijn van veldrobots. Na enkele prototypes die indoor getest werden, was het dit jaar tijd om een derde prototype te ontwikkelen om echt op het veld te gebruiken.

Bij aanvang van de masterproef was reeds de onderbouw volledig uitgewerkt. Deze bestaat uit een uitschuifbaar frame met rupsaandrijving. Om verschillende werktuigen te kunnen gebruiken, moet er eerst een bovenbouw ontworpen worden. Hiervoor wordt vooraf een korte literatuurstudie gedaan, waarin verschillende veldrobots zullen bekeken worden. Om de AgroBot effectief op het veld in te zetten, is er ook nood aan werktuigen. In het tweede deel van deze masterproef wordt het gebruik van een robotarm onderzocht.

## 1.2 Bedrijfsvoorstelling

Het 'Instituut voor Landbouw-, Visserij - en Voedingsonderzoek', waarvan het logo weergegeven is in figuur 1-1, is een onderzoeksinstituut dat instaat voor multidisciplinair onderzoek, onafhankelijk van de private sector. Zo verrichten ze onderzoek naar duurzame landbouw, visserij en voeding op vlak van economisch, maar ook op maatschappelijk als ecologisch standpunt. Vanuit dit onderzoek wordt toegepaste kennis opgebouwd die dient ter verbetering van productie én product, zodat de efficiëntie (ook hier zowel op vlak van economie, ecologie) stijgt, met aansluitend een blijvende, of zelfs verbeterde kwaliteit of veiligheid van het eindproduct.



**Figuur 1-1 Logo ILVO [2]**

ILVO heeft volgende onderzoeksgroepen:

- Gewassen
- Plantengenetica en Fysiologie
- Plantenteelt en bodem
- Gewasbescherming
- Veehouderij
- Aquatisch milieu en kwaliteit
- Visserij en aquatische productie
- Voeding
- Klimaat
- Milieu en PAS
- Plattelandsontwikkeling
- Bedrijfs - en sectorontwikkeling
- Bio - economie
- Agrotechniek

Deze masterproef situeert zich in de onderzoeksgroep Agrotechniek, die gelegen is te Burgemeester Van Gansberghelaan 115, Merelbeke.

De onderzoeksgroep Agrotechniek richt zich voornamelijk op het ontwikkelen van vernieuwende en milieuvriendelijke landbouwsystemen. Ook bestaande technieken worden verbeterd en samen met nieuwe technieken geïmplementeerd in een innovatief geheel. Er is een sterke expertise rond emissies en spuittechniek.

Nieuwer is de innovatieve precisielandbouw, waarbij aan de hand van GPS en sensoren, al dan niet autonoom, met hoge precisie aan landbouw gedaan kan worden [3].

## 2 LITERATUURSTUDIE

---

In deze literatuurstudie wordt eerst kort het concept precisielandbouw toegelicht. Hierin wordt gedefinieerd wat precisielandbouw precies is, gevolgd door enkele technieken. Vervolgens wordt een mogelijke toepassing binnen precisielandbouw besproken, namelijk geautomatiseerde en/of autonome machines (meer specifiek veldrobots). Tot slot kan men aan een veldrobot verschillende werktuigen monteren, een voorbeeld daarvan is een geautomatiseerde robotarm.

### 2.1 Precisielandbouw

#### 2.1.1 Wat is precisielandbouw?

Dit is een landbouwvorm waarbij zowel planten als dieren op heel nauwkeurige wijze behandeld worden. Dit kan zowel in tijd als in ruimte. In de akkerbouw wordt bijvoorbeeld niet één veld beschouwd, maar de variatie binnen een veld [4].

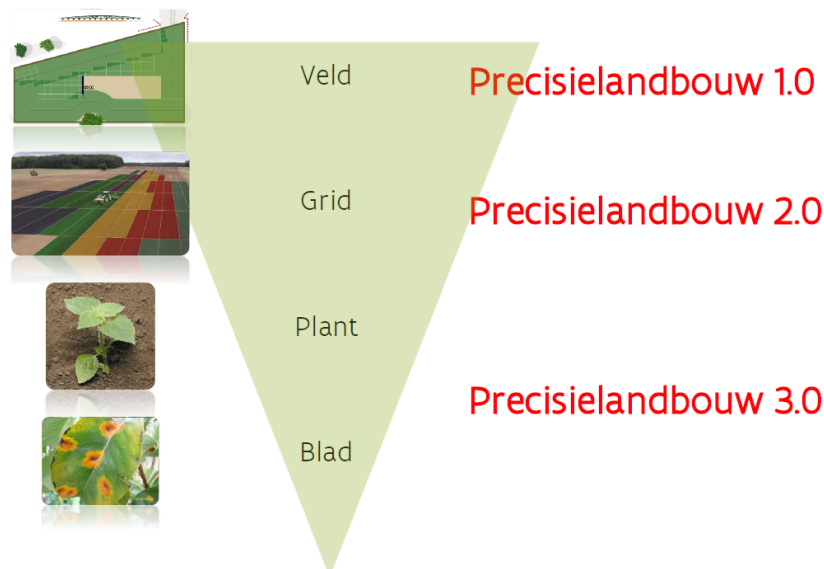
##### 2.1.1.1 Wat is er nodig?

- Plaatsbepaling
- Detectie via sensoren
- Analysering
- Uitvoering door middel van actuatoren

Dit alles vereist een grondige multidisciplinaire kennis van agricultuur en technologie.

##### 2.1.1.2 Hoe nauwkeurig is precisielandbouw?

In figuur 2-1 is een onderverdeling te zien van de verschillende toepassingsgebieden van de precisielandbouw.



**Figuur 2-1 Precisielandbouw [4]**

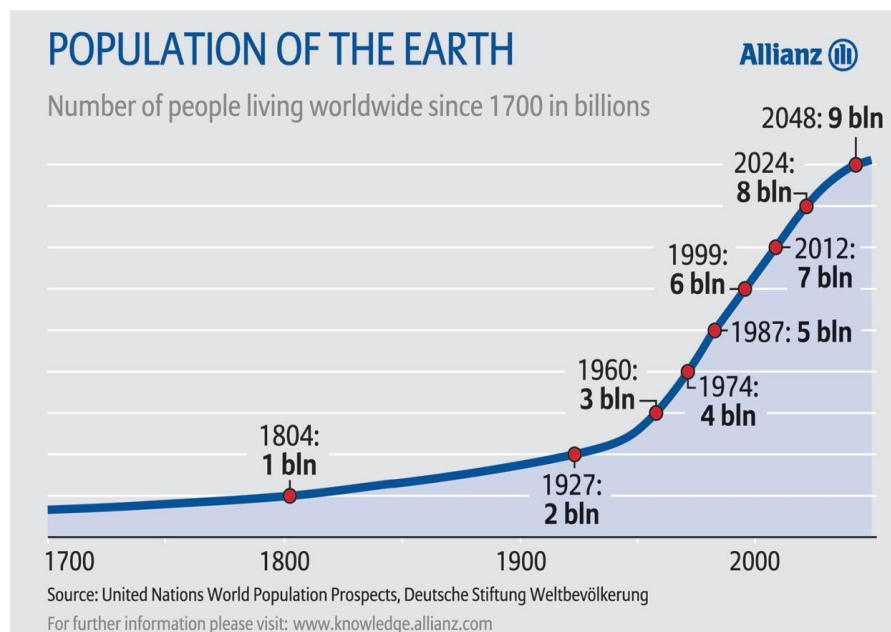
- Precisielandbouw 1.0: Nauwkeurige uniforme veldbewerking door middel van plaatsbepaling met satellieten. Zo wordt onder andere overlap vermeden.

- Precisielandbouw 2.0: Landbouw op 'grid'-niveau. Het veld wordt opgedeeld in partities door middel van sensormetingen als vochtigheid, bodemsamenstelling, etc. Dit via sensoren in het veld, luchtbeelden, etc.
- Precisielandbouw 3.0: Behandeling op plantniveau. Dit staat echter nog in kinderschoenen, hiervoor is verder onderzoek noodzakelijk.

## 2.1.2 Noodzaak aan precisielandbouw

Zoals te zien in figuur 2-2, stijgt de wereldpopulatie sterk. Momenteel staat de teller op ongeveer 7.6 miljard mensen, terwijl tegen 2048 een wereldbevolking verwacht wordt van 9 miljard mensen. Momenteel lijdt ongeveer één op negen honger. Daarbovenop komt het feit dat de stijging in de wereldbevolking zich voornamelijk situeert op de plaatsen waar reeds honger heerst. Het huidige landbouwsysteem moet bijgestuurd worden om deze bevolkingsgroei en de daarbij horende hongerproblematiek het hoofd te kunnen bieden [5] [6] [7].

De nood aan hogere opbrengsten dringt zich op. De landbouw moet efficiënter kunnen voorzien in de voedselvoorziening. Op biologisch vlak kan dit door zaadverfijning en genetische veredeling om sterkere, meer opbrengstrijke planrassen te ontwikkelen, maar ook op agrotechnisch vlak zijn veel verbeteringen mogelijk.

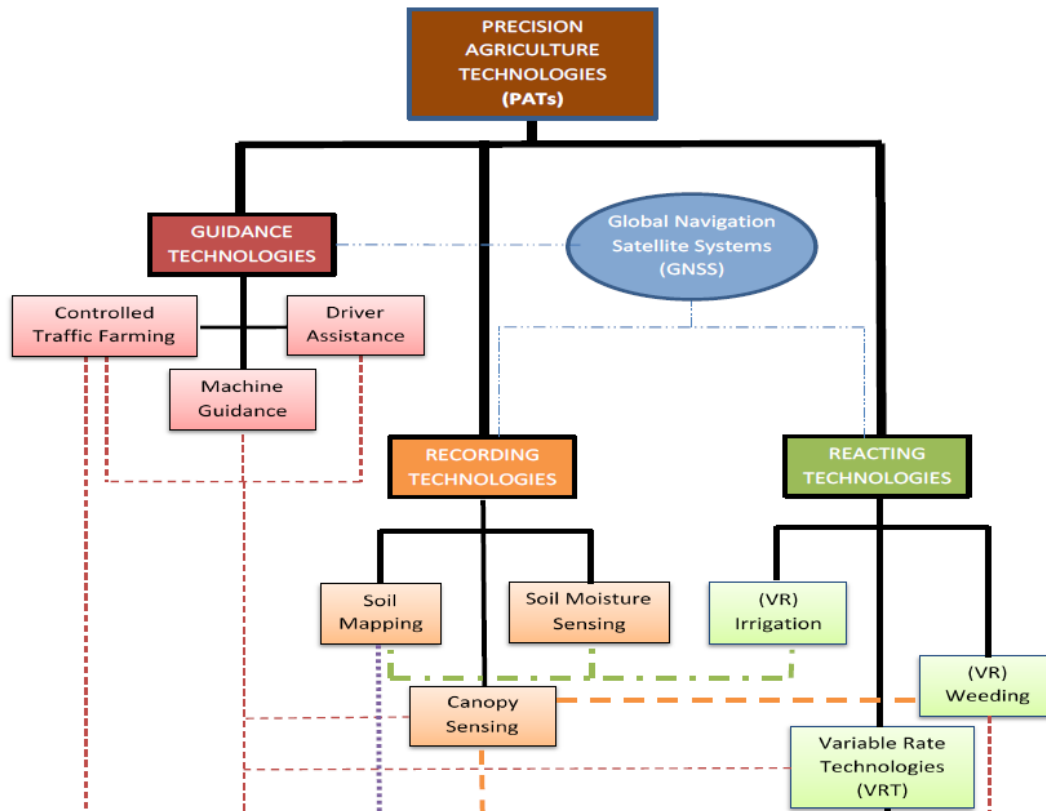


Figuur 2-2 Wereldpopulatie [8]

## 2.1.3 Opsplitsing precisielandbouw

Bij precisielandbouw gaat men over van een conventioneel landbouwsysteem naar een (deels-) geautomatiseerd systeem, dit omdat er met de huidige technologie nauwkeuriger en dus efficiënter kan gewerkt worden dan wat op zuiver menselijk vlak mogelijk is [4].

Precisielandbouw kunnen we opsplitsen volgens het schema afgebeeld in figuur 2-3:



Figur 2-3 Precision Agriculture Technologies [4]

### 2.1.3.1 Guidance technologies

Aan de hand van begeleidende technologie wordt het veld op een efficiëntere manier bewerkt. Met efficiënt wordt hier bedoeld dat er minder bewerkbaar grondoppervlak verloren gaat, alsook dat er nauwkeuriger geoogst wordt.

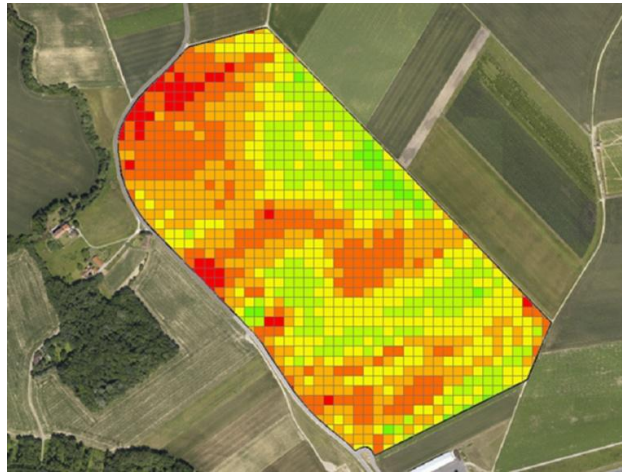
- **Controlled Traffic Farming:** Dit zorgt ervoor dat er zo weinig mogelijk sporengangen gemaakt worden in het veld. Waar de landbouwer vroeger ongeveer gelijke afstanden aanhield tussen de sporengangen, kan dit nu veel preciezer. Dit heeft als voordeel dat er minder bodemverdichting is en dat er bijvoorbeeld nergens teveel of te weinig gespreid wordt. Om de bodemverdichting in ontstane sporengangen extra te verminderen, kan er in de trekker een systeem geïnstalleerd worden waarmee de druk in de banden kan geregeld worden. Door op het veld op lage druk te rijden, wordt de bodem ter hoogte van de sporengangen minder aangedrukt.
- **Machine Guidance:** Door verschillende sensoren te koppelen aan de machines en tractor, krijgt de bestuurder feedback en informatie tijdens het werken.
- **Driver Assistance:** Bijvoorbeeld autopiloot die op basis van RTK GPS data de tractor bestuurt.

### 2.1.3.2 Recording technologies

Op figuur 2-4 is te zien hoe het land opgedeeld wordt in kleinere segmenten (soil mapping). Dit kan bijvoorbeeld een kaart zijn waarop de grondvochtigheid aangeduid is. Aan de hand van deze kaarten kan een intelligent bewateringssysteem de dosering plaats specifiek regelen.

Enkele methodes om deze gegevens te verkrijgen:

- Satellietbeelden
- Foto's via drones
- Sensoren in het veld
- Sensoren op de machine, mobiele platformen



**Figuur 2-4 Soil mapping [9]**

### **2.1.3.3 Reacting technologies**

Dit zijn technologieën die inwerken op de data van de recording technologies.

Met Variable Rate technology (VR) kan er op verschillende plaatsen in het veld een verschillende hoeveelheid 'input' toegevoerd worden. Enkele voorbeelden van inputs zijn: zaad, sproeistof, mest, ...

Verder kan ook op verschillende plaatsen meer of minder gewied worden, alsook irrigatie zal op verschillende plaatsen in het veld andere hoeveelheden kennen.

## **2.2 Machines in de precisielandbouw**

De hierboven besproken technieken worden momenteel volop onderzocht en waar mogelijk reeds geïmplementeerd. Hieronder een voorbeeld bij gevestigde tractorfabrikanten gevolgd door een korte inleiding tot veldrobots. Tot slot een bespreking van AgroBot van ILVO.

### **2.2.1 Algemene automatisering bij tractorfabrikanten**

Bij vele tractorfabrikanten wordt er al heel wat automatisering en begeleiding geïmplementeerd in de nieuwe trekkers en machines. Hieronder een aantal mogelijke hulpmiddelen van John Deere [10]:

- AutoTrac Universal: Automatische stuureenheid
- AutoTrac Controller: Controller waarbij componenten van verschillende merken kunnen geïmplementeerd worden
- AutoTrac Vision: Front-camera waarmee jonge scheuten kunnen herkend worden en waarmee tijdens het sproeien mooi tussen de planten kan gereden worden
- AutoTrac RowSense: Begeleidingssysteem om bv. maïs zo efficiënt mogelijk te oogsten

Dergelijke automatisering is te vinden bij de meeste fabrikanten, wat aangeeft dat precisielandbouw erg in opkomst is.



## 2.2.2 Fieldrobots

Wereldwijd worden nieuwe veldrobots ontwikkeld, zowel op onderzoeksniveau als op industrieel niveau. Hieronder een aantal robots die momenteel in ontwikkeling of op de markt zijn.

### 2.2.2.1 *Strawberry picking robot*

De aardbeiplukrobot van AgroBot is samengesteld uit een verrijdbaar chassis met een set grijpers die aardbeien kunnen lokaliseren, identificeren en plukken naargelang hun grootte en hun rijpheid. Daarna worden de geplukte aardbeien naar de inpakruimte getransporteerd. Daar kunnen ze manueel ingepakt worden. In afbeelding 2-5 is de robot te zien in een serre met aardbeien [11].



Figuur 2-5 Aardbeiplukrobot AgroBot [11]

### 2.2.2.2 *Kongskilde Vibro Crop Robotti*

De Vibro Crop Robotti is een autonome wiedrobot van Kongskilde. De robot navigeert met behulp van RTK-GPS. De navigatie is afgesteld om zoveel mogelijk op bestaande paden te rijden en zo een verdere verdichting van de bodem tegen te gaan [12] [13].

Het hoofddoel van de Vibro Crop Robotti is het autonoom schoffelen van een perceel zodat bespaard kan worden op arbeid. De robot is demonteerbaar, waardoor andere werktuigen kunnen aangehecht worden.

In figuur 2-6 is de robot te zien met een schoffelinstallatie.



Figuur 2-6 Kingskolde Vibro Crop Robotti [13]

### 2.2.2.3 Grassbots

Weideland bezit grote hoeveelheden biomassa dat potentieel kan gebruikt worden voor biogas [14].

Het doel van de Grassbots is om semi-autonoom, via vooraf uitgestippelde routes, gras te maaien op een economisch rendabele en milieuvriendelijke manier. De robot, zoals te zien op figuur 2-7, zou vooral ingezet worden in oevergebieden waar andere machines door hun omvang en gewicht niet goed functioneren. Door de lichte constructie is er een minimum aan schade ten gevolge van indrukking van de grond [15].



Figuur 2-7 GrassBot [15]

### 2.2.2.4 BoniRob

De BoniRob is een universeel robotplatform ontwikkeld door Amazone. Het ontwerp leent zich ertoe om allerlei teelthandelingen autonoom uit te voeren. Er kunnen namelijk op het basisplatform verschillende werktuigen gekoppeld worden. Dit gebeurt met behulp van een uniek koppelingssysteem. Deze multi-inzetbaarheid zorgt ervoor dat veel universiteiten en onderzoeksinstituten werktuigen ontwikkelen die compatibel zijn met deze veldrobot. [16] [17].

Door een autonoom platform als de BoniRob zal het mogelijk zijn om met kleine machines diverse teelten autonoom uit te voeren. Hierdoor kan de machine een veel kleinere vormfactor aannemen dan de huidige machines (hoe breder, hoe minder werkuren). Dit zorgt ervoor dat de bodem minder beschadigd wordt.

In figuur 2-8 staat de BoniRob klaar om getest te worden.



Figuur 2-8 Amazone BoniRob [17]

### 2.2.2.5 IBEX2

IBEX2 is een project van InnovateUK dat autonome spuitrobots maakt. In tegenstelling tot veel andere veldrobots is de IBEX2 voorzien van een rupsaandrijving waarmee hij makkelijk kan navigeren op ruwe ondergrond. (Bv. weideland, hellingen, ...) Door middel van on-board camera's en software kan onkruid herkend worden en vervolgens plaatselijk worden bespoten met herbicide [18] [19].



Figuur 2-9 IBEX2 Weed Sprayer [19]

### 2.2.2.6 Andere

Bovenstaande lijst is zeer beperkt en schetst een beeld over de mogelijkheden van autonome robots. Momenteel zijn er nog veel andere veldrobots in ontwikkeling, waarvan sommige dicht bij het productiestadium staan, terwijl andere projecten nog in de kinderschoenen staan. Belangrijk om op te merken blijft dat er een grote interesse is in precisielandbouw en dat er dus volop onderzoek naar verricht wordt.

## 2.3 ILVO AgroBot

In onderstaande uitleg worden kort de verschillende generaties van de AgroBot toegelicht.

### 2.3.1 Eerdere projecten

- Ontwerp en bouw van een robot voor deelname aan het FieldRobot event - Rico Claeys / Peter Bettens - 2010 - 2011
- De AgroBot - Jeroen Clauw / Andy Vandenbroucke - 2011 - 2012
- De AgroBot - Joris De Pauw - 2012 - 2013
- Ontwerp van een multifunctionele robotarm voor de AgroBot - Andy - / Benny Du Four - 2013 - 2014

In 2010-2011 ontwierpen Rico Claeys en Peter Bettens de eerste generatie van de AgroBot (AgroBot V1) om deel te nemen aan het FieldRobot event, een jaarlijkse competitie tussen verschillende Europese hogescholen en universiteiten. De robot werd tijdens het event onderworpen aan een aantal testen. Het zelfstandig navigeren van de robot door middel van sensoren en camera's was een van de belangrijkste taken. Het einddoel van dit event was het bouwen van een veldrobot die verschillende landbouwkundige taken op zich kan nemen (bv. plukken rijpe groenten, besproeien onkruid, ...). In dit eindwerk is het ontwerp van een

robot, maar ook het automatisch detecteren en besproeien van onkruid opgenomen. In figuur 2-10 is de eerste generatie te zien [20].



**Figuur 2-10 AgroBot V1 - Rico Claeys en Peter Bettens [21]**

Door de toenemende belangstelling voor automatisering in de landbouw en aldus het verlichten van het werk van de landbouwer, volgde in 2011-2012 een vervolgstudie door Jeroen Clauw en Andy Vandenbroucke op het initiële werk van Rico Claeys en Peter Bettens. Aangezien de eerste generatie moeite had om te navigeren in het ruwe akkerland, werd er besloten om een nieuw chassis te ontwikkelen op basis van het Bobcat - principe<sup>4</sup>. De AgroBot V2 kwam tot stand. Het nieuwe chassis was uitschuifbaar, zodat de spoorbreedte kon aangepast worden. Zo was de robot aanpasbaar voor gewassen met een verschillende rijafstand. Daarnaast werd ook de navigatie geoptimaliseerd door met een 2D laserscanner te werken. Alle sensoren en actuators werden verbonden met een Arduino. Uiteindelijk werd een goed functionerende robot bekomen die op moeilijke ondergronden kon navigeren en die obstakels op zijn pad kon ontwijken. In figuur 2-11 wordt de AgroBot V2 afgebeeld [22].



**Figuur 2-11 AgroBot V2 - Jeroen Clauw en Andy Vandenbroucke [22]**

In 2012-2013 werkte Joris de Pauw verder aan de AgroBot V2. Zijn opzet was om het bestaande platform te verbeteren, door onder andere de aandrijving te vervangen door vierwielaandrijving in plaats van driewiel. De aansturing moest uitgewerkt worden alsook

---

<sup>4</sup> Bobcat-Principe: sturen van het voertuig door de wielen aan linker - en rechterzijde in tegenovergestelde richting aan te sturen [125]

moesten er verschillende extra componenten geïntegreerd worden. Eenmaal de aansturing in orde was, werd er gekeken voor een praktische toepassing voor de AgroBot. Hier werd gekozen voor het plukken van aardbeien. Een manier om de aardbeien te detecteren werd beschreven en praktisch uitgevoerd. In figuur 2-12 is de AgroBot V2.1 zien [23].



**Figuur 2-12 AgroBot V2.1 - Joris de Pauw [23]**

Tot slot werkten Andy - en Benny Du Four in 2013-2014 verder op de AgroBot V2.1. Er werd een nieuw, langer chassis gebouwd en een nieuwe bovenplaat ontworpen om een robotische arm te huisvesten. Deze werd ontworpen via het gelede-arm-principe en werd door middel van riemoverbrengingen en stappenmotoren aangedreven. Het doel van deze robotarm was om, met behulp van het aangekoppelde visiesysteem (dieptecamera), onkruid te kunnen besproeien. Het geheel werd verbonden met een Arduino microcontroller die ook gebruikt werd voor de navigatie. Hieronder in figuur 2-13 de AgroBot V2.2 [24].



**Figuur 2-13 AgroBot V2.2 - Andy - en Benny Du Four [24]**

Nadat het project enkele jaren onaangeroerd gebleven is, is het in 2017 terug opgepikt met het ontwerp van een derde generatie AgroBot.

## **2.3.2 Derde generatie AgroBot - in productie**

### **2.3.2.1 Inleiding**

De bedoeling van de derde generatie is om een robot te bouwen die krachtig en groot genoeg is om op het veld te werken. Er werd een geheel nieuw chassis ontwikkeld dat meer dan twee keer zo groot is dan de voorgaande modellen. Om een beeld te scheppen van de afmetingen wordt tabel 2-1 gebruikt. Hierin worden de oppervlaktes van de bovenplaten vergeleken tussen de verschillende robots.

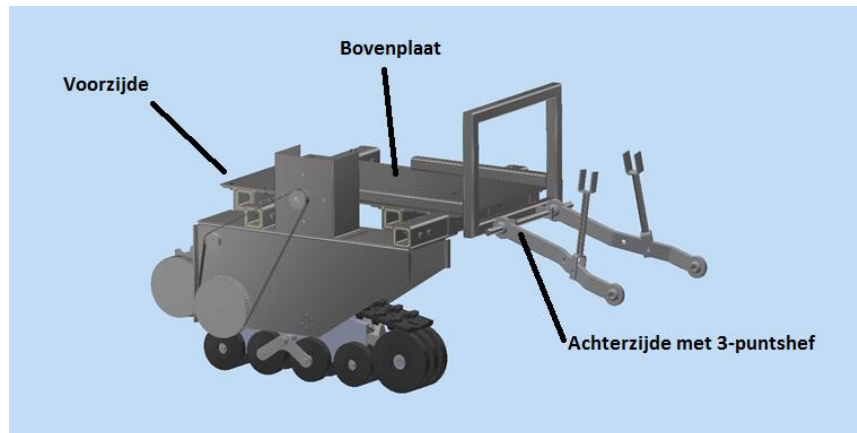
**Tabel 2-1 Afmetingen bovenplaat AgroBot V1-2 in vergelijking met V3**

	Lengte (m)	Breedte (m)	Oppervlakte (m <sup>2</sup> )
<b>AgroBot V1-V2</b>	0,6	0,4	0,24
<b>AgroBot V3</b>	1	0,6	0,6
Verschil in oppervlakte: factor <b>2,5</b>			

De robot blijft echter een proof-of-concept waarmee men wil aantonen wat de mogelijkheden zijn van veldrobots. De taken die de robot zal uitvoeren zullen in functie zijn van lopende onderzoeken, op de verworven kennis kan dan later worden verder gebouwd. De navigatie op het veld zal op autonoom gebeuren met behulp van sensoren en GPS.

Tot slot zal op de robot een gestandaardiseerde driepuntshef voorzien worden, zodat verscheidene machines kunnen aangesloten worden.

In figuur 2-14 is het ontwerp van de onderbouw en bovenbouw van de derde generatie AgroBot te zien, zoals deze ontworpen was aan het begin van het academiejaar 2017-2018.



**Figuur 2-14 AgroBot V3 - 3D ontwerp**

### **2.3.2.2 Werkelijke afmetingen**

In de voorgaande tabel werden de afmetingen van de basisplaat gegeven om een vergelijk te geven tussen de huidige en de vorige generaties van de AgroBot.

Doordat de derde generatie werkt met een uitschuifbaar rupssysteem kan de robot over een smaller of breder gangpad rijden. Dit zorgt voor een variabele spoorbreedte. Bovendien zullen op deze robot werktuigen bevestigd worden, waardoor de robot een stuk langer uitvalt dan enkel de basisplaat.

In tabel 2-2 worden de werkelijke afmetingen van de onderbouw van de robot gegeven.

**Tabel 2-2 Dimensies onderbouw AgroBot V3**

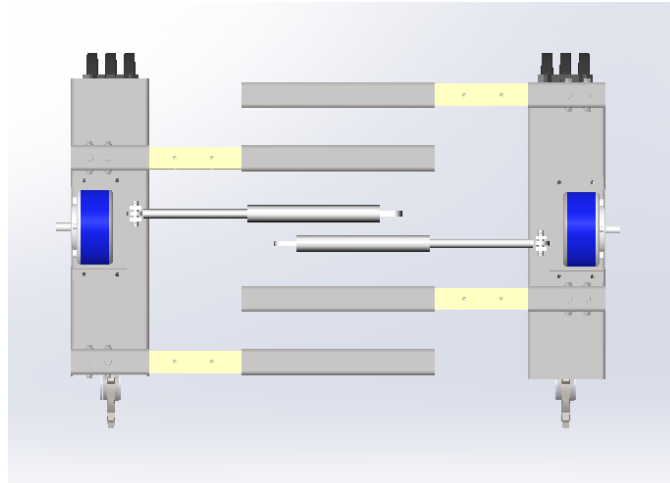
	Dimensies
<b>Lengte (m)</b>	1,01
<b>Hoogte (m)</b>	0,58
<b>Breedte (m)</b>	0,95 - 1,45
<b>Massa (kg)</b>	± 250

Opmerkelijk is ook de hoge massa. Dit komt door het gebruik van veel roestvast staal en zal ervoor zorgen dat er een krachtige aandrijving nodig is. Dit zorgt er wel voor dat er zwaardere werktuigen kunnen gebruikt worden. Indien er kans zou zijn op kantelen kan er later nog een systeem aangebouwd worden om gewichten aan de voorkant toe te voegen.

### 2.3.2.3 Opbouw & Onderdelen

De robot bestaat uit twee delen, de bovenbouw en de onderbouw.

De onderbouw werd reeds uitgewerkt. Het frame van de onderbouw is volledig vervaardigd uit roestvast staal. Aan de hand van afbeelding 2-15 kunnen een aantal belangrijke componenten benoemd worden.

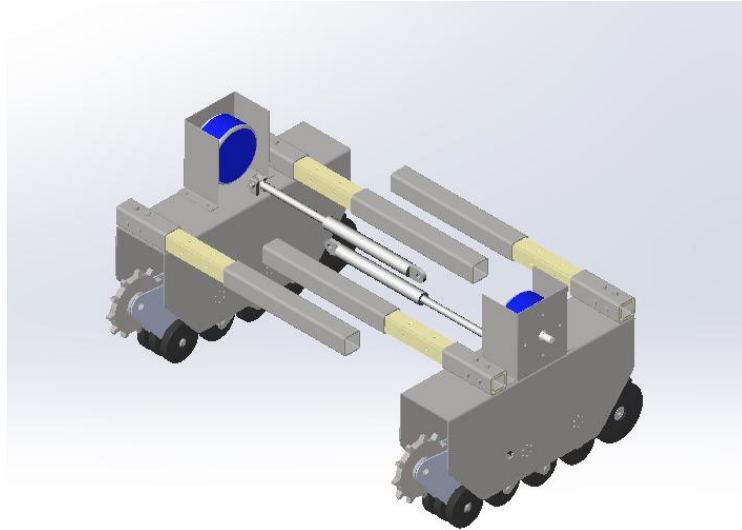


**Figuur 2-15 Onderbouw - bovenaanzicht - eigen bewerking van bestaand ontwerp**

Vooreerst zijn centraal twee (lichtgrijze) hydraulische cilinders zichtbaar. Deze zorgen ervoor dat de gangbreedte van de robot kan aangepast worden. Ook duidelijk zichtbaar hier zijn de geleidingen voor het uitschuifstelsel. Het geheel bestaat uit een buis-in-buissysteem. Door middel van de gele teflonstrips wordt een wrijvingsgeringe beweging bekomen.

De rupsen zullen aangedreven worden door middel van elektromotoren, gevoed door een 48-55V DC batterij. In de bovenstaande afbeelding zijn deze motoren in het blauw afgebeeld. Deze motoren zullen door middel van riemoverbrengingen afzonderlijk of gelijk de rupsen aandrijven. Ter verduidelijking figuur 2-16. Hierbij zijn de rupsen niet zichtbaar, omdat deze in het oorspronkelijke ontwerp niet inbegrepen zijn. Wat wel zichtbaar is, zijn de verschillende rupswielen en de getande wielen waarin de rupsen haken.

Door de motoren ongelijk te bekrachtigen (Bobcat) zal een draaibeweging uitgevoerd worden. Bij gelijk bekrachtigen zal de robot in een rechte lijn bewegen.




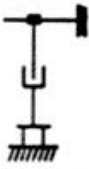
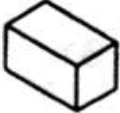

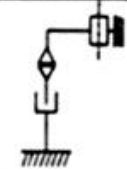







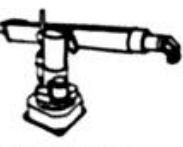


**Figuur 2-16 Onderbouw - isometrisch - eigen bewerking van bestaand ontwerp**

De bovenbouw wordt in figuur 2-14 voorgesteld door een vlakke plaat met daar een driepuntshef aan bevestigd. Vanuit deze basis zal een nieuwe, geschikte bovenbouw ontwikkeld worden.

## 2.4 Robotarmen

### 2.4.1 Mogelijke systemen voor een robotarm

In figuur 2-17 is een kort overzicht te zien van de mogelijke systemen die gebruikt worden bij robotarmen. Deze worden in komende paragrafen toegelicht.

Principle	Kinematic Structure	Workspace	Principle	Kinematic Structure	Workspace
 <p><b>Cartesian Robot</b></p>			 <p><b>SCARA Robot</b></p>		
 <p><b>Cylindrical Robot</b></p>			 <p><b>Articulated Robot</b></p>		
 <p><b>Spherical Robot</b></p>					

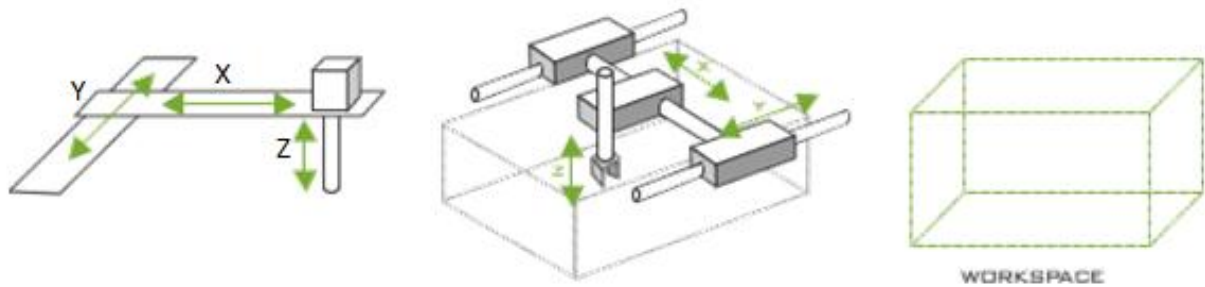
**Figuur 2-17 Types robotarmen [25]**

#### 2.4.1.1 Cartesiaanse robot

De cartesiaanse robot, ook vaak gekend onder de naam 'lineaire robot', is een robot die in de industrie vaak gebruikt wordt voor pick-and-placedoelinden. De robot bestaat uit drie



hoofdassen die ten opzichte van elkaar kunnen bewegen. Conventioneel zijn dit de X-, Y- en Z-as. De hoofdassen en het werkveld worden afgebeeld in figuur 2-18.



**Figuur 2-18 Vrijheidsgraden - Werkveld [26]**

Door hun vaste structuur hebben deze robots over het algemeen een goede precisie en herhaalbaarheid.

Een cartesiaanse robot die aan beide zijden ondersteund wordt, wordt een gantry-robot<sup>5</sup> genoemd. Deze robots zijn identiek aan cartesiaanse robots qua werking [27].



**Figuur 2-19 Gantry - robot [28]**

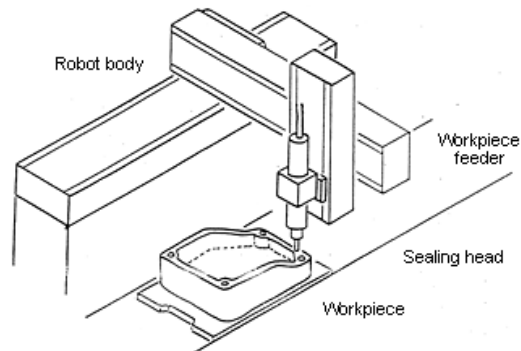
### ***Enkele toepassingen van cartesiaanse robots***

In figuren 2-20 tot en met 2-22 worden enkele toepassingen gegeven van de cartesiaanse robot.

---

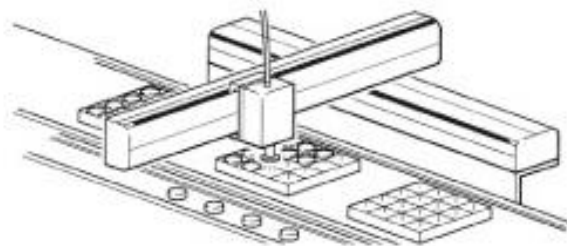
<sup>5</sup> Gantry = Portaal

- Dichtingen plaatsen



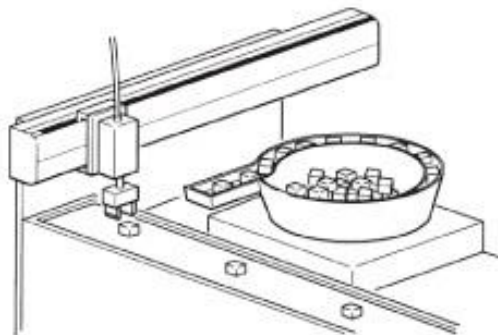
**Figuur 2-20 Dichtingen plaatsen [29]**

- Pick-and-place



**Figuur 2-21 Palletiseren [29]**

- Onderdelen aanvoeren



**Figuur 2-22 Onderdelen aanvoeren [29]**

#### **2.4.1.2 Cilindrische robot**

Bij deze robot liggen de assen volgens het cilindrisch coördinatenstelsel. Er zijn drie bewegingsrichtingen: een Z-beweging (omhoog, omlaag), een R-beweging (uitschuiven manipulator) en een  $\theta$ -beweging (draaien rond de as). In figuur 2-23 zijn deze bewegingen weergegeven alsook het cilindrisch werkveld dat bekomen wordt.

Over het algemeen worden de axiale verplaatsingen uitgevoerd door pneumatische cilinders, waar de rotatie bekomen wordt door middel van een elektrische motor en veranding [30].



**Figuur 2-23 Vrijheidsgraden - werkveld [26]**

Het cilindrisch coördinatenstelsel is extra handig bij objecten die over rotationele symmetrie beschikken. Als voorbeeld kan een lasrobot beschouwd worden die de binnenzijdes van twee buizen moet aaneen lassen.

Enkele toepassingen opgesomd:

- Assemblage
- Puntlassen
- Gaslassen
- Machinegereedschap aanbrengen/stockeren

In figuur 2-24 wordt een cilindrische robot van Hudson Robotics gebruikt om in laboratoriumomgeving testspecimen te stockeren en terug aan te brengen.



**Figuur 2-24 Cilindrische robot van Hudson Robotics [31]**

### **2.4.1.3 Sferische robot**

Bij de sferische robot zijn de assen volgens het sferische (bolcoördinaten) assenstelsel. Er zijn wederom drie bewegingsrichtingen. De axiale R-beweging waarlangs de actuator uitschuift, en de twee sferische bewegingen ( $\theta$ ,  $\varphi$ ). Het bol-cilindrische werkveld is te zien op figuur 2-25, samen met de verschillende bewegingsrichtingen [32].



**Figuur 2-25 Vrijheidsgraden - werkveld [26]**

### **Enkele toepassingen van sferische robots**

Het toepassingsgebied van de sferische robot is gelijk aan dat van de cilindrische robot.

Hieronder, in figuur 2-26, wordt de "Stanford Arm" weergegeven. Dit is een sferische robotarm met zes vrijheidsgraden, ontworpen door Victor Scheinman en is een van de eerste robots die ontworpen werd om met een computer te besturen [33].



**Figuur 2-26 Stanford Arm [33]**

#### **2.4.1.4 SCARA robot**

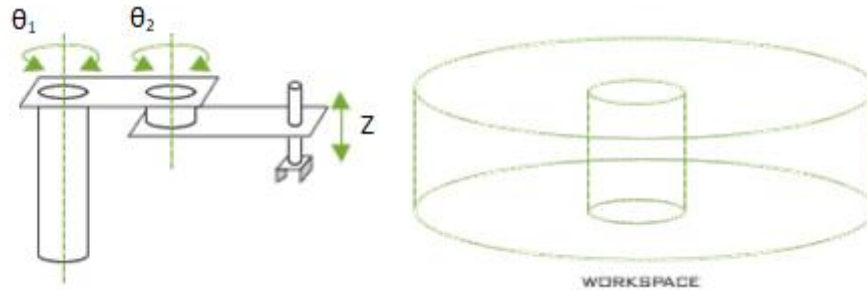
De SCARA<sup>6</sup> robot wordt gekenmerkt doordat alle assen parallel zijn aan elkaar. Er kan in het X-Y-veld bewogen worden door middel van rotatie van de armdelen ten opzichte van de basis, maar ook ten opzichte van elkaar. Tenslotte kan er bewogen worden in de Z-richting via een axiale actuator. De bewegingsmogelijkheden en het werkveld zijn zichtbaar in figuur 2-27. Opmerkelijk aan de SCARA robot is dat het cilindrisch werkveld een stuk ruimer is dan dat van de cilindrische robot. Dit komt doordat de arm kan 'dichtplooiën', waardoor het gebied dicht bij de rotatie-as van de basis beter bereikbaar is.

Met "Selective Compliance" wordt bedoeld dat de arm flexibel is in de X- en Y-richting, maar star in de Z-richting. De A in SCARA staat voor "Assembly" of "Articulated". Het toepassingsgebied laat zich hierdoor eenvoudig raden.

---

<sup>6</sup> SCARA = Selective Compliance Assembly Robot Arm / Selective Compliance Articulated Robot Arm

SCARA robotarmen zijn over het algemeen sneller dan Cartesiaanse robots, nemen minder plaats in beslag en kunnen makkelijk op een plat vlak gemonteerd worden. Echter is dit type robotarm over het algemeen duurder dan de cartesiaanse tegenhanger. De programmering is een stuk complexer, doordat de robotarm op verschillende manieren een bepaalde positie kan bereiken. Hierdoor is de berekening van de inverse kinematica<sup>7</sup> vereist [34].

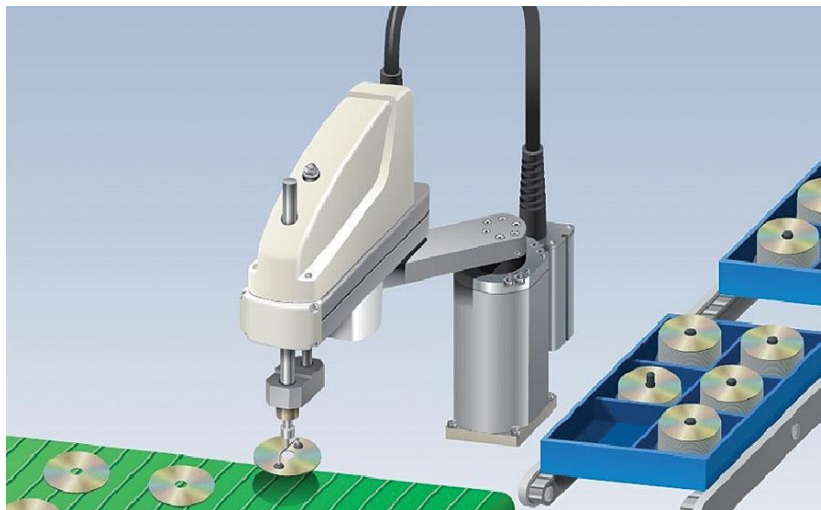


**Figuur 2-27** Vrijheidsgraden - werkveld [26]

### **Enkele toepassingen van SCARA Robots**

Dit type robot zal vaak gebruikt worden voor assembleerdoeleinden in omgevingen waar hygiëne van groot belang is, zoals in de voedingsindustrie.

Een ander voorbeeld is te zien in figuur 2-28, waar een SCARA robotarm gebruikt wordt voor het sorteren en plaatsen van CD's in trays.

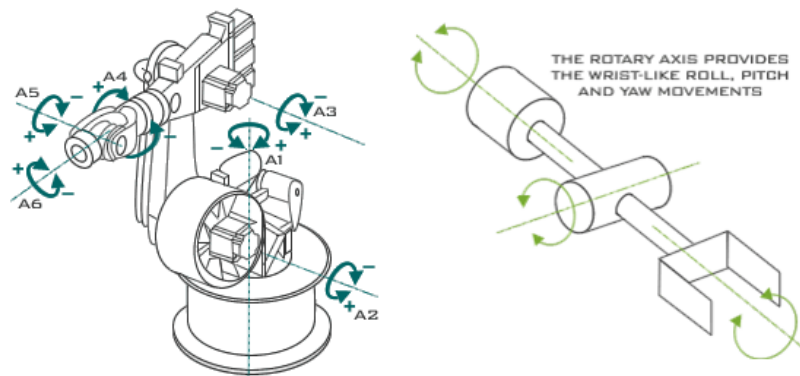


**Figuur 2-28** SCARA robotarm gebruikt voor sorteren en plaatsen van CD's in trays [35]

---

<sup>7</sup> Inverse kinematica = het berekenen van de verschillende rotatiehoeken uit een gewenste positie en oriëntatie van de eeffector

### 2.4.1.5 Gelede arm robot



**Figuur 2-29 Vrijheidsgraden gelede arm robot [26]**

Bij de gelede arm robot kunnen alle armdelen ten opzichte van elkaar roteren. In figuur 2-29 wordt een zes-assige arm voorgesteld. De rotatiehoeken worden hier aangeduid met A1 - A6. Opmerkelijk zijn de drie rotaties A4-A6 die samenkomen in de pols.

Doordat elk armdeel ten opzichte van elk ander armdeel kan bewegen worden complexe bewegingsvergelijkingen bekomen. De voorwaartse en inverse kinematica moeten uitgewerkt worden. Dit houdt in dat aan de hand van de hoeken waarin de armdelen staan, bepaald kan worden wat de exacte positie is in een ruimtelijk gebied, of dat het punt en de oriëntatie in de ruimte gekend zijn en dat hieruit de hoeken van de verschillende armdelen berekend kunnen worden. De complexiteit van deze kinematica hangt nauw samen met het aantal vrijheidsgraden.

Door de hoge bewegingsmogelijkheid is de gelede arm robot een van de meeste veelzijdige robots in de industrie. Vaak wordt deze arm ook vergeleken met een menselijke arm, waarbij men vaak uitgaat van een 6-DOF robot zoals in bovenstaande figuur. Bewegingen A1 en A2 stellen de schouder voor, A3 de elleboog en A4, A5 en A6 de pols [36] [37].

#### **Enkele toepassingen van een gelede arm robot**

Door zijn veelzijdigheid kan de gelede arm robot voor tal van doeleinden gebruikt worden. In feite is deze geschikt voor alle toepassingen van de voorgaande robotarmen.

In figuur 2-30 is een assemblagelijijn te zien waar tal van zes-assige robots gebruikt worden.



**Figuur 2-30 Industriële gelede arm robots in een Tesla fabriek [38]**

#### 2.4.1.6 Delta (parallele) robot

De delta robot is een type parallelle robot die bestaat uit drie (of meer) armen die verbonden zijn met een gemeenschappelijk scharnier aan de onderzijde. Het parallelle in de delta robot slaat op het feit dat het onderste platform steeds parallel is met het bovenste platform. Dit komt door het gebruik van parallellogrammen in de armen, waardoor de oriëntatie van de eeffector behouden blijft.

In figuur 2-31 wordt de delta robot voorgesteld, samen met het werkingsgebied.



**Figuur 2-31 Delta robot - Werkveld [26]**

Indien een rotatie van de eeffector vereist is, dan kan een extra motor aangebracht worden die de centrale as doet draaien, zoals te zien is op figuur 2-32.



**Figuur 2-32 Delta robot met roterende actuator [39]**

De delta robot wordt gekenmerkt door zijn snelheid, precisie en herhaalbaarheid.

Een belangrijke bemerking bij de delta robot is dat eventuele fouten in de beweging van de armen door elkaar (deels) uitgemiddeld worden of op zijn minst niet versterkt worden. Dit waar bij seriële robots de fout telkens opgebouwd wordt [40] [41] [42].

#### **Enkele toepassingen van een delta robot**

Dit type robot wordt voornamelijk gebruikt voor pick-en-place doeleinden, zoals ook te zien in figuur 2-33.

Andere toepassingen zijn te vinden in onder andere de medische industrie waar ze door hun stijfheid kunnen gebruikt worden voor operaties. Ook worden ze gebruikt voor assemblagedoeleinden voor elektronica in 'clean rooms'. Tot slot worden deze robots sinds kort ook gebruikt in de 3D-printindustrie.



**Figuur 2-33 Delta robot voor sorteren van pralines [43]**

## 2.4.2 Robotarmen op de markt

De vraag luidt nu: zijn er robotarmen te koop op de markt die makkelijk inzetbaar zijn op een mobiele veldrobot, binnen het vooropgelegde budget?

Het antwoord hierop is neen. Er is op de markt geen enkele industriële robotarm te koop die binnen het budget valt. Bovendien zijn de beschikbare robotarmen (die buiten het budget vallen) niet geschikt voor gebruik op een mobiele applicatie omdat ze vaak voorzien moeten worden van een grote stuurkast. Tenslotte worden ze over het algemeen gevoed met wisselspanning, wat niet aanwezig is op de veldrobot.

Aangezien er geen geschikte industriële robotarmen zijn, zal er geen uitgebreide literatuurstudie gedaan worden. Voor de volledigheid echter, worden hieronder enkele robotarmen van gevestigde fabrikanten kort beschreven. Deze kunnen later dienen als inspiratie voor het eigen ontwerp. Voor aanvang van het onderzoekwerk werd eerst bepaald welk type robotarm bruikbaar zou kunnen zijn, waarna het aantal gewenste vrijheidsgraden bepaald werd. De beslissing viel op een zes-assige gelede arm robot. Meer toelichting hierover is te vinden in hoofdstuk vijf.

In de opdrachtomschrijving worden een aantal randvoorwaarden opgelegd, deze zijn ook weergegeven in tabel 2-3. Omdat voornamelijk het budget een beperkende factor is, zal deze achterwege gelaten worden bij een initiële zoektocht. Het mobiele aspect dient ook niet beschouwd te worden, aangezien elk van deze robots een aanzienlijke stuurkast heeft.

**Tabel 2-3 Vereisten robotarm**

	<b>Randvoorwaarden</b>
<b>Reikwijdte</b>	1 m
<b>Payload</b>	2 kg
<b>Budget</b>	€1,500
<b>Extra</b>	Mobiliteit
<b># Assen</b>	6

Hieronder is een korte lijst te vinden van zes-assige robotarmen die een gelijkaardige reikwijdte en payload hebben. Deze liggen allemaal in een prijsbereik tussen €25,000 en €75,000.



#### **2.4.2.1 ABB IRB 1200 5/900**

Met een payload van vijf kilogram is de IRB 1200 krachtiger dan nodig, de reikwijdte daarentegen is net te weinig. Beide parameters zijn echter nog binnen de grenzen van het bruikbare.

Optioneel is deze robot ook beschikbaar met een IP67 - rating, waardoor gebruik in vochtige omgevingen geen probleem vormt.

In tabel 2-4 zijn de belangrijkste specificaties van de robotarm gegeven, in figuur 2-34 is deze afgebeeld [44] [45].



**Figuur 2-34 ABB IRB 1200 5/900 [45]**

#### **2.4.2.2 FANUC M-6iB/2HS**

De FANUC M-6iB is een robotarm die zo goed als volledig overeenkomt met de vooropgestelde randvoorwaarden. De HS in de naam staat voor High Speed. Door deze hogere snelheden is de payload lager dan bij gelijkaardige modellen. Opmerkelijk is de massa, die een stuk hoger is dan die van de IRB 1200.

In tabel 2-4 zijn de belangrijkste specificaties van de robotarm gegeven, in figuur 2-35 wordt de robotarm afgebeeld [46].



**Figuur 2-35 FANUC M-6iB/2HS [46]**

### 2.4.2.3 KUKA KR6-R900

De KR6-R900 is bijna identiek aan de IRB 1200. De reikwijdte, payload en massa zijn ongeveer gelijk. Opmerkelijk is hier de herhaalbaarheid, want deze is dubbel zo goed als bij de IRB 1200.

Dit model is ook beschikbaar met IP67-rating.

In tabel 2-4 zijn de belangrijkste specificaties van de robotarm gegeven, in figuur 2-36 wordt de robotarm afgebeeld [47].



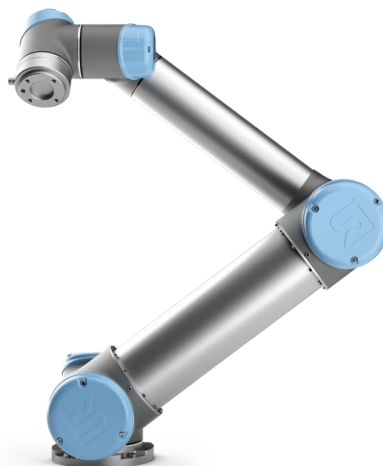
Figuur 2-36 KUKA KR6-R900 [47]

### 2.4.2.4 Universal Robots UR5

De UR5 past niet helemaal in de lijst, aangezien het geen 'industriële robot' is. Deze robotarm is ontwikkeld om samen te werken met de operator. Met de UR5 kan er op een veilige manier geautomatiseerd worden, afschermingen zijn niet nodig.

De reikwijdte van de robotarm ligt net buiten het bruikbare gebied. Opmerkelijk is de erg lage massa in vergelijking met de andere robotarmen. De herhaalbaarheid, hoewel minder goed dan bij de andere robots, is nog steeds voldoende.

In tabel 2-4 zijn de belangrijkste specificaties van de robotarm gegeven, in figuur 2-37 wordt de robotarm afgebeeld [47].



Figuur 2-37 Universal Robots UR5 [48]

### 2.4.2.5 Tabel met gegevens

In tabel 2-4 worden de gegevens van de besproken robotarmen weergegeven.

Tabel 2-4 Gegevens robots [46] [47] [48] [49]

	ABB	FUNAC	KUKA	Universal Robots
Type	IRB 1200	M-6iB/2HS	KR6-R900	UR5
Payload	5 kg	2 kg	6 kg	5 kg
Reikwijdte	901 mm	951 mm	901 mm	850 mm
Herhaalbaarheid	0.06 mm	0.08 mm	0.03 mm	0.1 mm
Massa	54 kg	135 kg	52 kg	18.4 kg

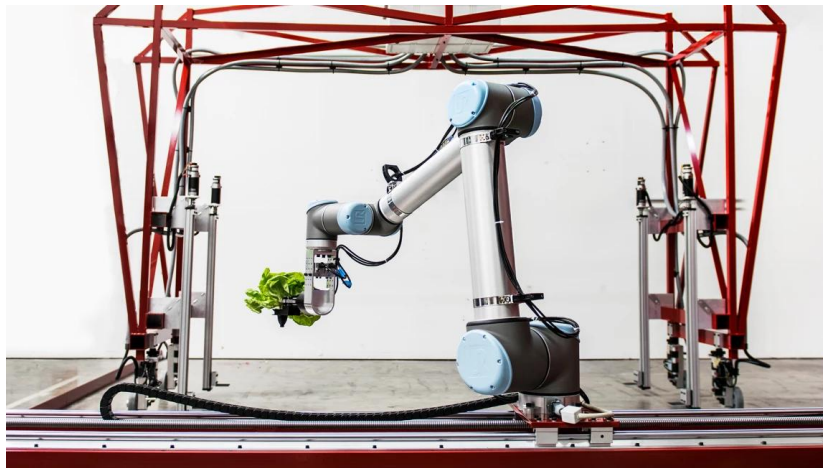
## 2.4.3 Robotarmen in de landbouw

Binnen de landbouw zijn er enkele projecten waarbij robotarmen geïmplementeerd worden voor het automatiseren van oogstsystemen. In deze sectie worden enkele van deze armen toegelicht [50].

### 2.4.3.1 Iron Ox Lettuce Robot

De Iron Ox, te zien op figuur 2-38, is ontworpen om sla te verplanten tussen verschillende trays. Met behulp van een lineaire geleiding kan de robotarm van de ene zijde naar de andere bewegen. Plantdetectie gebeurt met behulp van stereovisie waardoor een driedimensioneel beeld van elke plant verkregen wordt [51].

De gebruikte robot is van fabrikant Universal Robots, analoog aan de UR5 die reeds besproken werd in de vorige paragraaf.



Figuur 2-38 Iron Ox Lettuce Robot [52]

### 2.4.3.2 MIT Robot Gardener

Studenten aan MIT (Massachusetts Institute of Technology) ontwikkelden een mobiele robot, te zien op figuur 2-39, om gewassen van water te voorzien en de vruchten te plukken. Bij deze robot is een robotarm bevestigd op het rijdend gedeelte.

De planten zijn allemaal voorzien van sensoren die draadloos communiceren met de robot. Als bijvoorbeeld de grondvochtigheid te laag is, wordt de robot geroepen [53].



**Figuur 2-39 MIT Robot Gardener [53]**

#### **2.4.3.3 Komkommer plukrobot**

Deze plukrobot is een project van Wageningen U&R. Omdat ongeveer dertig procent van alle kosten bij serrekwekerijen naar lonen gaat, onderzochten zij de impact van het gebruik van een gerobotiseerde installatie.

De beschikbare robots op de markt voldeden niet voor gebruik in serres, waarna er een eigen robot ontwikkeld werd, waarop een robotarm bevestigd werd. Met behulp van een lineaire geleiding kan de robotarm langs de planten bewegen.

Een extra moeilijkheid bij het plukken van komkommers is dat zowel de plant als de vrucht groen zijn. Door gebruik te maken van stereovisie kunnen de komkommers gedetecteerd worden, waarna met een afzonderlijke camera de rijpheid gecontroleerd wordt [54] [55].



**Figuur 2-40 Komkommer plukrobot [55]**

#### **2.4.3.4 Berry Nice**

De Berry Nice - robot werd ontwikkeld voor het plukken van rijpe aardbeien. Deze detecteert zowel de vrucht als de rijpheid met behulp van stereovisie camera's. De werking is analoog aan de komkommer plukrobot en de Iron Ox. De robot beweegt ook hier met behulp van een lineaire geleiding, zoals te zien op figuur 2-41 [50].



**Figuur 2-41 Berry Nice [56]**

#### **2.4.3.5 Conclusie**

Drie van de vier genoemde robots werken via eenzelfde concept, namelijk het plaatsen van een robotarm met visiesysteem op een lineaire geleiding. Dit blijkt ook zo bij nog andere fabrikanten en onderzoekscentra. Doordat de robot op een geleiding bevestigd is, heeft deze weinig last van externe factoren zoals trillingen. De detectie gebeurt zo goed als altijd met behulp van stereovisie.

### 3 PROBLEEMSTELLING & DOEL

---

Om de veldrobot effectief op het veld in te zetten, moet er eerst een nieuwe bovenbouw ontworpen worden. Deze dient voldoende robuust te zijn en ook geschikt om er op een universele wijze verschillende werktuigen op te bevestigen.

In de landbouw is de omgeving erg ongestructureerd en is er geen enkele plaats gelijk. De gewassen kunnen makkelijk beschadigd worden, waardoor extra voorzichtigheid nodig is. Dit en nog veel andere factoren zorgen ervoor dat een studie in verband met automatisering belangrijke kennis kan opleveren voor latere toepassingen.

De besproken robotarmen in de landbouw worden telkens gemonteerd op een lineaire geleiding. Omdat er weinig onderzoeksprojecten lopende zijn waarbij een robotarm effectief gebruikt wordt op een veldrobot, kan onderzoek hiernaar leiden tot nieuwe, interessante inzichten.

Uit de literatuurstudie blijkt reeds dat een industriële robotarm aankopen om tal van redenen geen valabele optie is. De belangrijkste reden is dat de aankoopprijs van een robotarm ver boven het beschikbare budget ligt. Een eigen ontwerp dient gemaakt te worden dat later kan gekoppeld worden aan de veldrobot.

Voor het ontwerp wordt er van uitgegaan dat er geen rekening dient gehouden te worden met trillingen en dat de robotarm enkel zal gebruikt worden als de veldrobot stilstaat tijdens het bewegen van de arm.

## 4 BOVENBOUW AGROBOT V3

---

Om de AgroBot op het veld in te zetten, is een stevige bovenbouw nodig waaraan tal van werktuigen kunnen bevestigd worden. In dit hoofdstuk wordt een beknopte toelichting gegeven hoe het ontwerp tot stand gekomen is.

### 4.1 Ontwerp

Doordat de bovenbouw moest afgestemd worden op de reeds beschikbare onderbouw, werd het ontwerp van Jordi Demarest geraadpleegd. Dit was getekend in Inventor [57], waardoor het niet meteen bruikbaar was. Na importeren in Solidworks [58] moesten alle koppelingen tussen de componenten opnieuw aangemaakt worden. Vanuit deze basis werd gestart aan het ontwerp van de bovenbouw.

#### 4.1.1 Algemeen

De bovenbouw vormt een behuizing voor alle regelcomponenten, hydraulische groep en de batterij. Daarom werd een constructie uitgewerkt die eenvoudig door middel van plaatwerk kan afgesloten worden.

Door gebruik te maken van stevige profielen heeft de bovenbouw een massa van ongeveer honderd kilogram. Eenmaal alles voorzien is op de bovenbouw zal een totaalgewicht van de gehele AgroBot van ongeveer vierhonderd kilogram bekomen worden.

Achteraan de robot kunnen via een universele aanhechting verschillende werktuigen gekoppeld worden. Indien deze werktuigen de robot doen kantelen, zullen er frontgewichten moeten voorzien worden.

#### 4.1.2 Aanhechting werktuigen

De aanhechting van de werktuigen gebeurt door middel van een universele driepuntshef. In figuur 4-1 is een gelijkaardige hef te zien op een Kubota mini trekker.

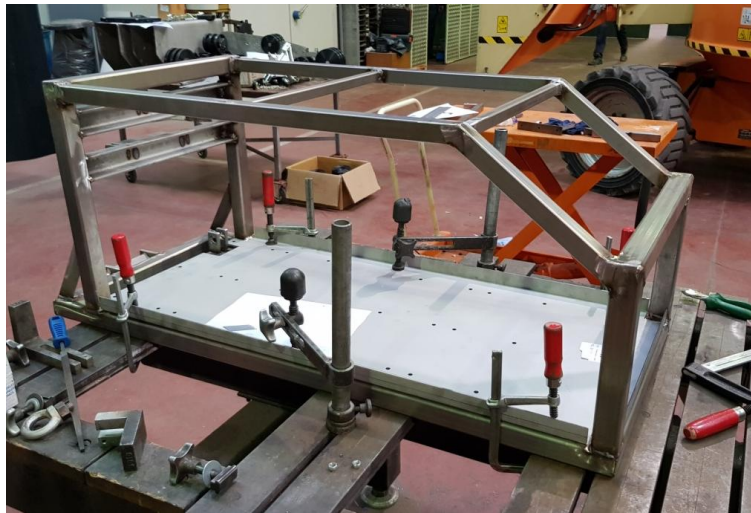
Deze hef kan omhoog en omlaag bewegen door middel van een hydraulische cilinder.



Figuur 4-1 Driepuntshef op een kubota [59]

### 4.1.3 Materiaalkeuze en bewerkingstechniek

De bovenbouw werd volledig vervaardigd in roestvast stalen profielen. Deze werden aaneen gelast tot een robuuste constructie. De lasconstructie is in opbouw te zien op figuur 4-2.

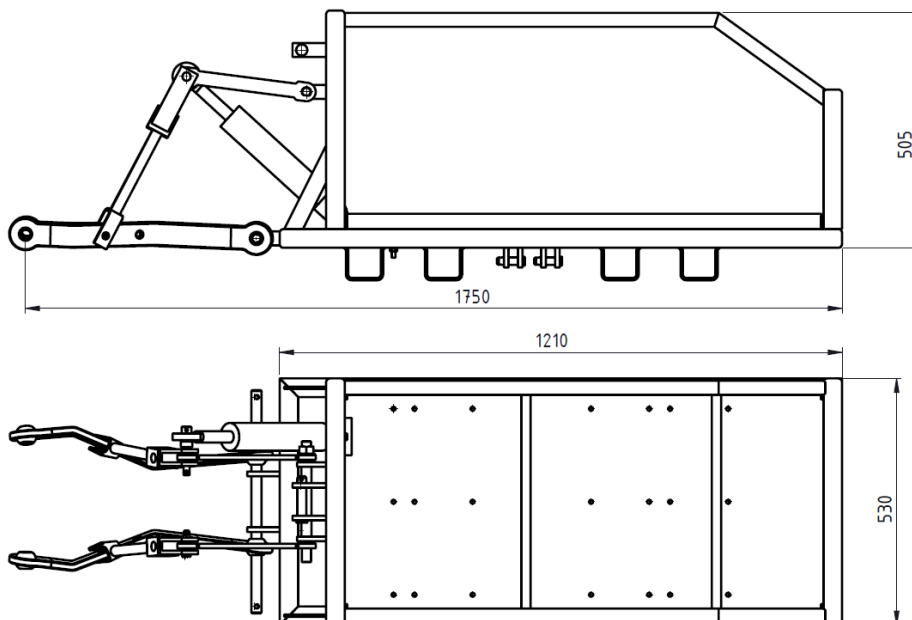


Figuur 4-2 Lasconstructie in opbouw

De verschillende assen die gebruikt worden bij de bovenbouw worden allemaal vastgepend in houders die op het frame gelast zijn. Zo kunnen de hef en de hydraulische cilinders gekoppeld worden aan de bovenbouw.

### 4.1.4 Afmetingen

In figuur 4-3 zijn de belangrijkste afmetingen van de bovenbouw zichtbaar.

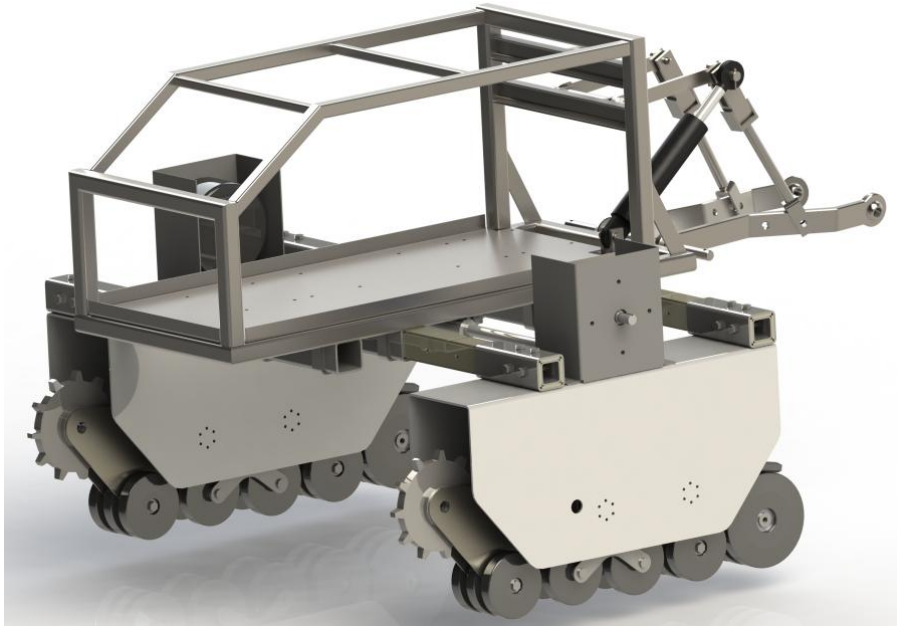


Figuur 4-3 Zij- en bovenaanzicht met maten



## 4.2 Eindresultaat ontwerp

In figuur 4-4 is een render van het eindresultaat van het ontwerp te zien.



Figuur 4-4 AgroBot eindresultaat - render

## 4.3 Praktisch eindresultaat

In figuur 4-5 is de AgroBot te zien bij de eerste test in een grasveld.



Figuur 4-5 AgroBot op grasveld

## 5 AGROBOT A.R.M.

---

Het robotarmproject werd omgedoopt tot de AgroBot A.R.M., waarbij de A voor Articulated staat, de R voor Robotic en de M voor Manipulator. In dit hoofdstuk wordt kort de keuze van het type robotarm besproken, waarna het aantal vrijheidsgraden toegelicht wordt. Tenslotte wordt het ontwerp besproken.

### 5.1 Keuze type arm

De keuze van het type robotarm volgt uit het vergelijken van een aantal factoren eigen aan de verschillende beschikbare types. Door het toekennen van een gewicht aan de factoren en een score per type kan zo eenvoudig het meest gunstige model gekozen worden.

#### 5.1.1 Factoren

De factoren van de types robotarmen worden in deze paragraaf kort toegelicht. Bij de keuze wordt er geen rekening gehouden met de eerste drie factoren, omdat ze hier niet relevant zijn of een kleine impact hebben op de keuze. De andere factoren worden in tabel 5-1 opgesomd met daarnaast een gewichtsfactor.

##### 5.1.1.1 *Kostprijs*

Aangezien de robotarm zelf zal ontworpen worden, is de kostprijs voornamelijk afhankelijk van de ontwerpkeuzes en minder van het type robotarm. Niettegenstaande zal er bij de verschillende types een licht verschillende kostprijs zijn doordat er een verschillend aantal motoren zal gebruikt worden.

##### 5.1.1.2 *Gewicht*

Het gewicht volgt voornamelijk uit de keuze van de materialen. Het gewichtsverschil tussen de verschillende types robotarmen zal daarom klein zijn.

##### 5.1.1.3 *Integratie*

Het spreekt voor zich dat de robotarm te integreren moet zijn op de veldrobot. De parallelle en cartesische robot worden daarom niet opgenomen in de keuze van het type robotarm. Deze zijn door hun ontwerp niet geschikt om gemonteerd te worden op de veldrobot. Alle andere types zijn wel geschikt voor implementatie.

##### 5.1.1.4 *Aansturing*

Op de veldrobot is er enkel een 48 VDC spanningsbron aanwezig. Er is geen persluchtgroep voorzien, waardoor pneumatische actuatoren niet gebruikt kunnen worden.

##### 5.1.1.5 *Complexiteit*

Er zijn twee soorten complexiteit, de complexiteit van de sturing en de complexiteit van het ontwerp.

De complexiteit van de sturing is voornamelijk afhankelijk van het aantal vrijheidsgraden, terwijl de complexiteit van het ontwerp afhankelijk is van het type robotarm. Uiteindelijk gaan deze twee vaak hand in hand, want een robotarm met veel vrijheidsgraden heeft over het algemeen ook een complexer ontwerp.

### 5.1.1.6 Werkingsgebied

Het werkingsgebied hangt volledig af van het type robotarm. De opbouw van de vrijheidsgraden zal ervoor zorgen dat een bepaald werkingsgebied bekomen wordt. Hoe groter dit gebied, hoe beter.

### 5.1.1.7 Functionaliteit

Tenslotte dient het gekozen type robotarm geschikt te zijn voor tal van toepassingen.

**Tabel 5-1 Gewicht van de verschillende factoren**

	<b>Gewicht</b>
<b>Aansturing</b>	1
<b>Complexiteit</b>	2
<b>Werkingsgebied</b>	3
<b>Functionaliteit</b>	4

## 5.1.2 Score verschillende robotarmen

Aa alle types wordt een score gegeven op tien, naargelang hoe goed ze presteren op vlak van de factoren beschreven in 5.1.1. Deze scores zijn intuïtief opgesteld met behulp van de verworven kennis uit de literatuurstudie.

Tenslotte worden de scores opgeteld met hun gewicht ingerekend. Het type robotarm met de hoogste totale score is het meest geschikt. Dit is te zien in tabel 5-2.

**Tabel 5-2 Score van de verschillende types robotarmen**

	<b>Cilindrische robot</b>	<b>Sferische robot</b>	<b>SCARA robot</b>	<b>Gelede arm robot</b>
<b>Aansturing</b>	3	5	10	10
<b>Complexiteit</b>	10	7	6	3
<b>Werkingsgebied</b>	4	6	8	10
<b>Functionaliteit</b>	5	6	7	10
<b>Totaal</b>	<b>55</b>	<b>61</b>	<b>74</b>	<b>86</b>

Uit deze scores kan opgemaakt worden dat de gelede arm robot het meest geschikt is, voornamelijk door het grote werkingsgebied en de ideale functionaliteit.

## 5.2 Benodigde vrijheidsgraad bepalen

Om de benodigde vrijheidsgraad te bepalen is het interessant om naar enkele mogelijke toepassingen te kijken. Vooraf is het interessant om te weten dat een robot met zes vrijheidsgraden theoretisch gezien dezelfde posities kan bereiken als een robot met zeven vrijheidsgraden, omdat dit het punt is vanaf wanneer totale bewegingsvrijheid bekomen wordt. Daarom zal dit ingesteld worden als maximaal aantal vrijheidsgraden [60].

Voor voorbeelden van zes-assige robotarmen wordt verwezen naar paragraaf 2.4.2.

### 5.2.1 Enkele mogelijke toepassingen

Naast elke toepassing wordt een richtlijn gegeven van het aantal benodigde vrijheidsgraden.

De vrijheidsgraden werden intuïtief bepaald door na te gaan welke flexibiliteit vereist is om een bepaalde handeling uit te voeren. Bij het plukken van vruchten bijvoorbeeld moet de robotarm zich kunnen positioneren in het X-Y-Z-vlak, maar de grijper moet ook zodanig gedraaid kunnen worden dat een vrucht schuin kan vastgegrepen worden (om eventuele schade aan de plant te voorkomen). Daarom zijn voor deze toepassing zes vrijheidsgraden vereist. Onkruidverdelging daarentegen kan in feite met drie vrijheidsgraden omdat ervoor gekozen kan worden om de spuitmond in een vaste oriëntatie te verplaatsen. Analoge redeneringen kunnen opgebouwd worden voor de andere toepassingen.

- Plukken van vruchten (6)
- Specifieke onkruidverdelging (3)
- Grondstalen nemen (4)
- Ziekteherkenning met camera en GPS-datapunten (4)
- Manipuleren van de omgeving om voorbij een obstakel te geraken (5)
- Sensoren die een verhoogde mobiliteit vereisen (4)
- Pick and place in serre (5)

### 5.2.2 Conclusie

Omdat vooraf nog niet geweten is welke taken de robotarm gaat uitvoeren op de AgroBot, wordt ervoor gekozen om het ontwerp zo ruim mogelijk te maken, zodat de robotarm voor alle hierboven beschreven toepassingen inzetbaar is. Daarom is er gekozen om de robotarm van zes vrijheidsgraden te voorzien.

In figuur 5-1 is een Universal Robots robotarm te zien met de zes vrijheidsgraden aangeduid.



Figuur 5-1 Robotarm met zes vrijheidsgraden [61]

## 5.3 Ontwerp

### 5.3.1 Randvoorwaarden

Bij het ontwerp dienen de randvoorwaarden opgelegd in de opdrachtomschrijving in acht genomen te worden. Deze worden voor de volledigheid in tabel 5-3 nogmaals weergegeven.

Tabel 5-3 Randvoorwaarden

	Randvoorwaarden
Reikwijdte	1 m
Payload	2 kg
Budget	€1,500
Extra	Mobiliteit

Door de omvang van het budget kunnen enkel eenvoudige bewerkingstechnieken als laseren, plooiën, frezen en draaien gebruikt worden. Bovendien zal de payload sterk afhangen van de gebruikte materialen en het budget.

De reikwijdte, gecombineerd met de payload, zal enkel kunnen behaald worden als de eigen massa van de arm niet te hoog is. Anders moeten er grotere motoren aangekocht worden die misschien niet binnen het budget vallen.

De mobiliteit kan enkel gegarandeerd worden als de robotarm bestaat uit een relatief lichte constructie.

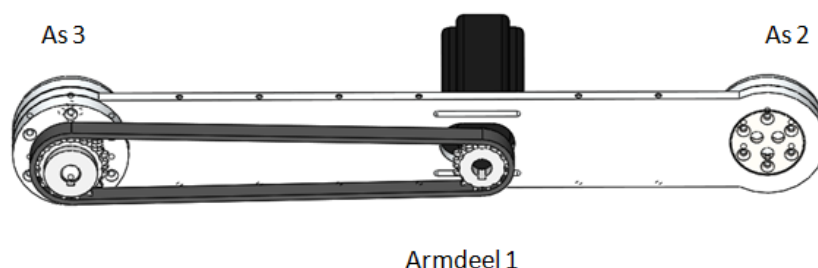
### 5.3.2 Uiterlijk

Het uiterlijk van de robot is voornamelijk gebaseerd op robotarmen uit de industrie, maar ook op projecten van ingenieurs over de wereld. Hieronder worden kort een aantal aspecten van het ontwerp besproken.

#### 5.3.2.1 Plaatsing motoren

Bij zes-assige industriële robotarmen worden de motoren voornamelijk geplaatst ter hoogte van de scharnierpunten van de verschillende armdelen. Bij het ontwerp van de AgroBot A.R.M. werd dit toegepast, met het verschil dat de motoren telkens naar achter geplaatst zijn op de verschillende armdelen. Dit om het extra koppel veroorzaakt door het gewicht van de motoren zo laag mogelijk te houden.

Dit wordt weergegeven in figuur 5-2, waarbij de motor op armdeel één verbonden is met de as drie waarrond armdeel twee roteert.



Figuur 5-2 Armdeel 1

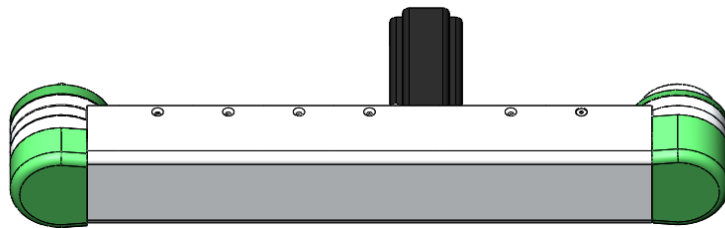
### 5.3.2.2 Opspanning kettingen/riemen

Zoals te zien op figuur 5-2 wordt de motor bevestigd op de arm ter hoogte van de sleuven. De bouten om de motor vast te zetten, worden door de flens van de motor en door gleuven van het armdeel gebracht. Door de motor naar links of rechts te bewegen als de riem-/of ketting gemonteerd is, kan deze opgespannen worden.

### 5.3.2.3 Afdekkappen

Op de robotarm werden verschillende afdekkappen aangebracht. Deze hebben een dubbel nut. Op de lange armdelen zorgen ze voor een extra torsiestijfheid. Bovendien zijn ze nodig om bewegende machineonderdelen af te schermen van de buitenwereld uit veiligheidsoverwegingen.

Op figuur 5-3 is armdeel één te zien met een aluminium afdekkap en twee 3D-geprinte kappen.



Figuur 5-3 Armdeel 1 met kappen

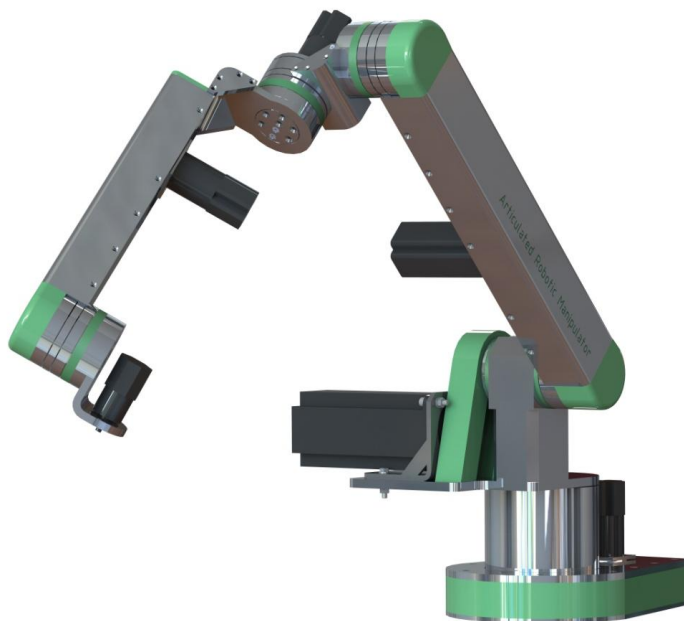
### 5.3.2.4 Eindeffector

Momenteel is er nog geen eindeffector voorzien op de robotarm. Deze kan later gekoppeld worden aan de zesde motor die reeds voorzien is.

## 5.3.3 Uiteindelijke ontwerp

### 5.3.3.1 Algemeen

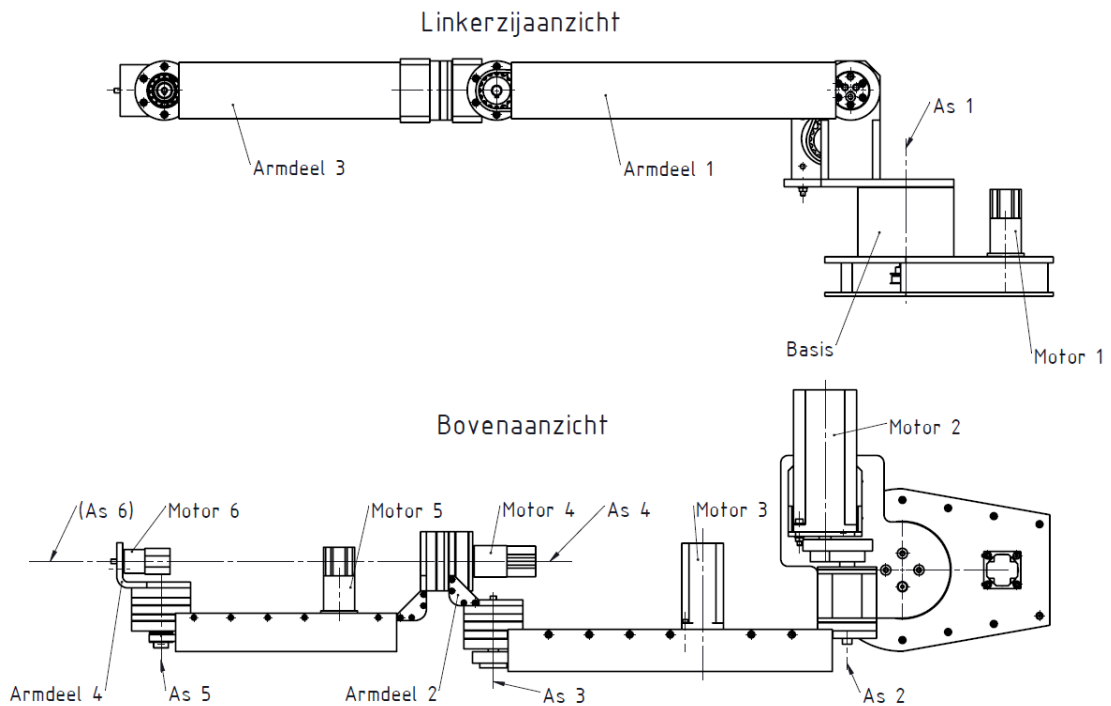
Voor de technische tekeningen wordt verwezen naar het tekendossier "AgroBot A.R.M.". In figuur 5-4 wordt het uiteindelijke ontwerp weergegeven.



Figuur 5-4 Uiteindelijke ontwerp AgroBot A.R.M.

### 5.3.3.2 *Figuur met benoeming motoren, assen en armdelen*

In figuur 5-5 worden de verschillende motoren, assen en armdelen benoemd.

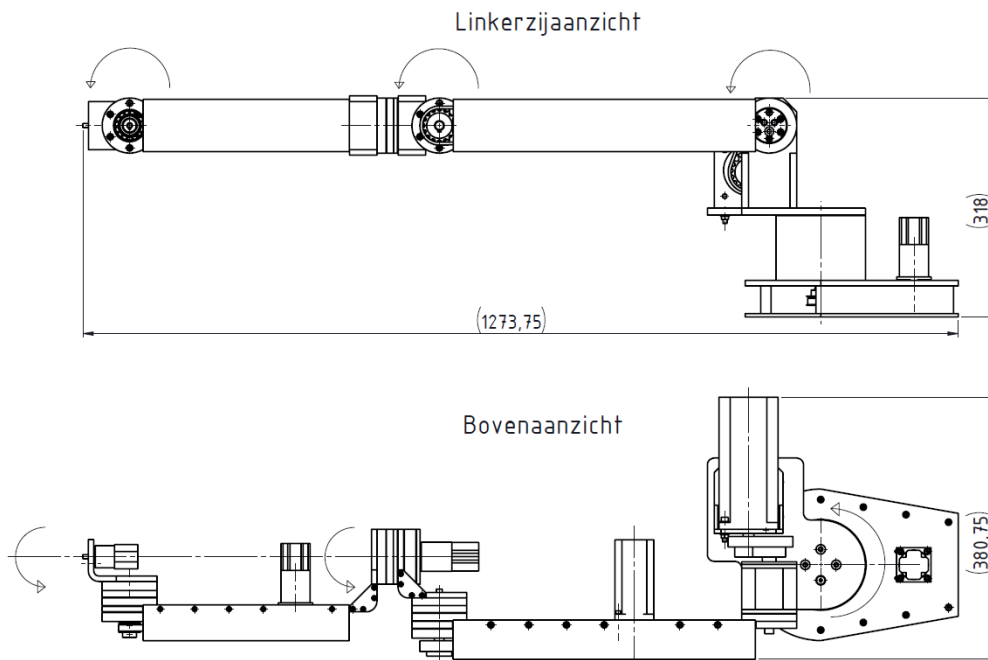


**Figuur 5-5 Benoeming Motoren, assen en armdelen**

### 5.3.3.3 *Afmetingen*

#### 5.3.3.3.1 *Buitenmaten en rotaties*

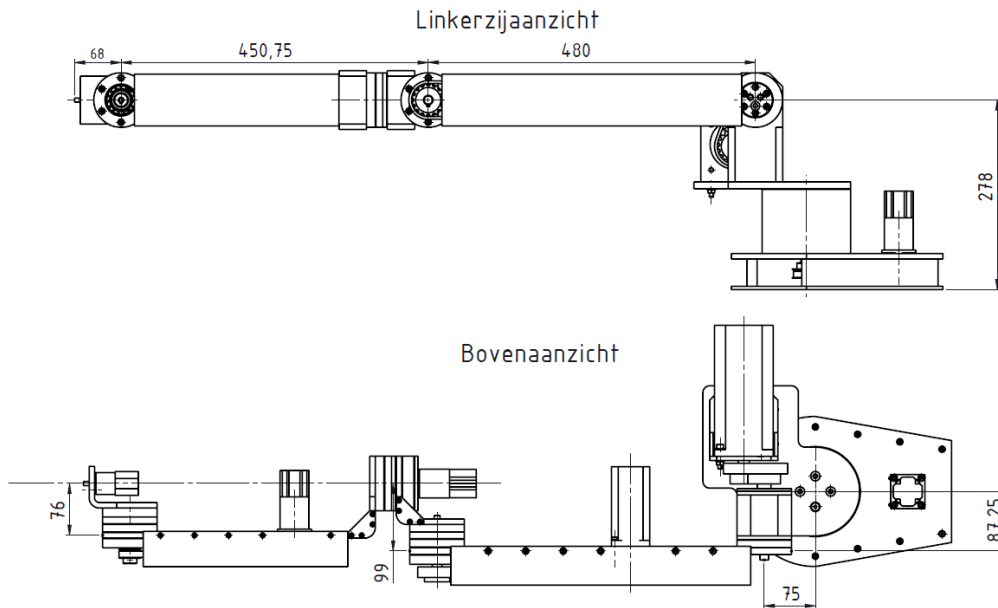
Op figuur 5-6 zijn de buitenmaten en de mogelijke rotaties van de robotarm weergegeven.



**Figuur 5-6 Buitenmaten en rotaties**

### 5.3.3.2 Maten belangrijk voor latere berekeningen

Bij latere berekening van de voorwaartse en inverse kinematica worden een aantal afmetingen gebruikt van de robotarm. Deze zijn weergegeven op figuur 5-7.



**Figuur 5-7 Belangrijke maten**

### 5.3.4 Praktisch eindresultaat

Op figuur 5-8 en 5-9 is de A.R.M. te zien in de testfase.



**Figuur 5-8 AgroBot A.R.M. testfase**





**Figuur 5-9 AgroBot A.R.M. testfase**

## 6 PRAKTISCHE UITWERKING EN BEREKENINGEN (A.R.M.)

---

### 6.1 Materiaalkeuze

In de volgende paragrafen worden kort de gebruikte materialen toegelicht. Er wordt een onderscheid gemaakt tussen het frame, de assen en andere, aangekochte componenten. Kenmerkend voor de gehele robotarm is corrosiebestendigheid, wat nodig is voor gebruik op het veld.

#### 6.1.1 Frame

Het frame van de robotarm is volledig vervaardigd uit aluminium. Dit is namelijk een stuk lichter dan staal, waardoor er minder krachtige motoren moeten gebruikt worden. Deze keuze was noodzakelijk, omdat motoren met een hoger koppel snel een stuk duurder en bovendien moeilijker te vinden zijn, zeker mits de arm gevoed wordt met een 48V DC spanningsbron.

Er werden verschillende varianten aluminium gebruikt naargelang de benodigde nabewerking. Zo werd de aluminiumlegering EN AW-5754 gebruikt voor stukken die gelaserd en geplooid moesten worden. Dit is een legering van aluminium en magnesium die een uitstekende corrosiebestendigheid heeft. Bovendien is dit materiaal sterker dan een ongelegeerde variant. De vervormbaarheid is goed, maar de verspaanbaarheid niet. Er worden lange spanen gevormd en de stukken zijn moeilijk nauwkeurig na te bewerken [62] [63]. Voor de componenten die enkel gedraaid en gefreesd moesten worden is er gekozen voor EN-AW 6082. Deze variant is beter geschikt om te verspanen en kent dezelfde, goed corrosiebestendigheid.

#### 6.1.2 Assen (verbindingsflenzen)

De assen en de verbindingsflenzen zijn vervaardigd uit roestvast staal. Door deze keuze zijn de stukken zwaarder, maar hebben ze een grotere treksterkte en een hogere hardheid. Dit is in het bijzonder handig omdat hier soms grote koppels op komen [64] [65].

De gebruikte variant is AISI 304, een veelgebruikte legering die gekenmerkt wordt door de toevoeging van achttien procent Chroom en acht procent Nikkel. Hierdoor heeft dit type roestvast staal een erg goede weerstand tegen corrosie. Bovendien is het goed verspaanbaar, waardoor het ideaal is voor deze toepassing.

#### 6.1.3 Andere componenten

##### 6.1.3.1 Lagers

De lagers zijn vervaardigd uit staal en dienen zowel voor de werking als voor de weerstand tegen corrosie geregeld gesmeerd te worden.

##### 6.1.3.2 Bouten, moeren, spieën

Voor de bouten, moeren en spieën werd er ook telkens gekozen voor roestvast staal. Waar mogelijk is er geopteerd geweest voor de AISI 316 varianten en waar dit niet mogelijk was voor AISI 304. AISI 316 is namelijk nog meer corrosiebestendig. Deze keuze is er gekomen omdat de bouten en moeren vaak aan de buitenkant van de robotarm zitten, waardoor ze meer blootgesteld worden aan corrosieve media. Bovendien staan de bouten, moeren en spieën vaak onder spanning waardoor spanningscorrosie in de hand gewerkt wordt.

### 6.1.3.3 Tandriemschijven, kettingwielen en kettingen

De tandriemschijven zijn vervaardigd uit aluminium, echter werd de variant niet meegegeven door de fabrikant. Voor latere berekeningen wordt er van uitgegaan dat het gaat om EN-AW 2017A. Dit is een aluminiumlegering die door riemschijffabrikant BRECO [66] gebruikt wordt. Het heeft een hoge treksterkte en hardheid in vergelijking met ongelegeerd aluminium.

De kettingwielen en ketting zijn momenteel vervaardigd in C45-Staal, dit is echter een tijdelijke oplossing voor het testen van de robotarm. Dit staal heeft een koolstofgehalte van vijfenveertig procent en heeft een hogere sterkte dan het aluminium dat gebruikt wordt voor de tandriemschijven. Doordat het staal niet extra gelegeerd is, is het niet corrosiebestendig. De kettingen en kettingwielen zullen tijdig moeten gesmeerd worden. Op termijn kunnen deze componenten vervangen worden door roestvast stalen varianten.

## 6.2 Motoren

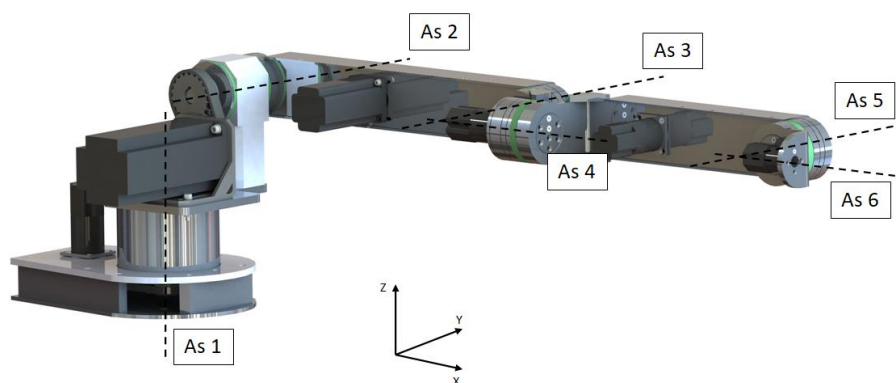
De berekeningen ter bepaling van het benodigde motorkoppel worden in feite twee maal uitgevoerd. De eerste maal bij een initieel ontwerp zonder motoren, om zo een inschatting te kunnen maken over het benodigde koppel. Daarna worden de motoren gekozen en worden de berekeningen opnieuw gedaan met de massa's van de motoren ingerekend.

De berekeningen hieronder zijn de tweede set berekeningen, waarbij de motoren reeds gekozen waren. De toelichting van deze keuze volgt in 6.2.4. Tevens zijn de berekeningen telkens ten opzichte van de assen, dus zonder overbrenging.

### 6.2.1 Statisch koppel

Bij de statische berekening wordt telkens het koppel uitgerekend dat op een as heerst als de arm zich in de meest belaste toestand bevindt. Dit geval doet zich voor als de robotarm volledig uitgestrekt is, met de maximale payload van twee kilogram op het einde. Deze toestand is voorgesteld in figuur 6-1.

Assen één en zes worden niet of nauwelijks statisch belast en zullen daarom niet in de berekening van de statische belasting opgenomen worden.



Figuur 6-1 Aanduiding assen

#### 6.2.1.1 Gegevens nodig voor de berekeningen

In bijlage B is een tabel terug te vinden met alle gegevens die nodig zijn voor de berekening van de koppels.

In deze tabel worden de massa's van bouten en moeren verwaarloosd. Als compensatie worden alle massa's van de andere componenten per 5 gram naar boven afgerond. Tandriemschijven en tandriemen worden telkens verondersteld een massa van 100 gram te hebben.

### 6.2.1.2 Statische berekening

Om de totale statische belasting te berekenen in een bepaalde motor, volstaat het om het deel na deze motor als één rechte, statische staaf te beschouwen. Daarna wordt het koppel dat veroorzaakt wordt door alle onderdelen op deze staaf berekend.

De koppelberekening gebeurt volgens volgende formule:

	$T_{statisch,As} = \sum \text{Gewicht} * \text{hefboom} = \sum (m * \text{hefboom}) * g \text{ [Nm]}$ $\text{met } g \cong 10 \frac{m}{s^2}$	(6.1)
--	--	-------

Dit houdt in dat de som genomen wordt van elke massa met zijn hefboomsarm ten opzichte van de beschouwde as.

De onderstaande berekeningen respecteren de rechterhandregel volgens het assenstelsel weergegeven in figuur 6-1.

### 6.2.1.3 Assen twee, drie, vier en vijf

Hieronder worden de benodigde statische momenten uitgerekend per as.

#### As twee:

	$T_{statisch,As2}$ $= (1050 * 240 + 1800 * 210 + \dots + 175 * 975 + 2000 * 998.75) * 10$ $= 762828955 \frac{10^{-3}kg * 10^{-3}m * m}{s^2} = 76.28 Nm$	(6.2)
--	---	-------

#### As drie:

	$T_{statisch,As3}$ $= (155 * 30 + 65 * 45 + \dots + 175 * 495 + 2000 * 998.75) * 10$ $= 23.06 Nm$	(6.3)
--	---	-------

#### As vier:

	$T_{statisch,As4} = (45 * 42.50 + 680 * 35 + \dots + 210 * 30 + 70 * 25) * 10$ $= 1.98 Nm$	(6.4)
--	--	-------

In de getekende stand is het koppel negatief volgens de rechterhandregel. Echter kan er rond as vier geroteerd worden met honderdtachtig graden, waardoor een positief koppel bekomen wordt. Om latere verwarring te vermijden, wordt het koppel hier daarom positief gekozen.

**As vijf:**

	$T_{statisch,As5} = (400 * 19.25 + 175 * 44.25 + 2000 * 68) * 10$ $= 1.51 Nm$	(6.5)
--	---	-------

## 6.2.2 Dynamisch koppel

Onder dynamisch koppel kan het koppel verstaan worden dat nodig is om de arm gefixeerd te houden, gesommeerd met het koppel dat nodig is om de arm te versnellen. Deze component is afhankelijk van de massa en de versnelling. De berekening kan opgesplitst worden in de berekening van assen één en vier, waarbij de verschillende armdelen geen invloed hebben op elkaar. Daarnaast zijn er nog assen twee, drie en vijf die een gecombineerd koppel zullen veroorzaken.

### 6.2.2.1 Algemeen

Na het berekenen van de hoekversnelling kan het dynamisch koppel bekomen worden met behulp van onderstaande formule:

	$T_{dynamisch} = T_{statisch} + J * \alpha = T_{statisch}$	(6.6)
--	--	-------

Met J = massatraagheidsmoment,  $\alpha$  = hoekversnelling

Omdat het massatraagheidsmoment vaak moeilijk te berekenen is, kan deze formule ook vereenvoudigd geschreven worden:

	$T_{dynamisch} = T_{statisch} + m * a * hefboom$	(6.7)
--	--	-------

Met m = massa, a = versnelling.

Om de correctheid te valideren, volstaat het om te controleren als de eenheden van beide formules gelijk zijn.

	$J * \alpha = m * a * hefboom \Leftrightarrow [J * \alpha] = [m * a * hefboom]$	(6.8)
--	---	-------

	$[J * \alpha] = kgm^2 * \frac{1}{s^2} = Nm$	(6.9)
--	---	-------

	$[m * a * hefboom] = kg * \frac{m}{s^2} * m = Nm$	(6.10)
--	---	--------

De eenheden zijn gelijk aan elkaar, dus kan de vereenvoudigde formule gebruikt worden.

## 6.2.2.2 Bepalen van de versnelling

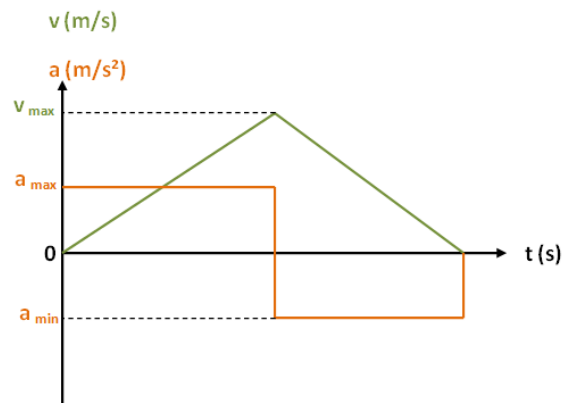
### 6.2.2.2.1 Snelheids- en versnellingsprofiel

Omdat in de gebruikte software geen rukbegrenzing aanwezig is, zal de arm telkens maximaal versnellen en vertragen. Deze versnelling wordt bepaald door het maximale koppel dat de motoren kunnen leveren.

Voor de koppelberekening wordt ervan uitgegaan dat elke as een verdraaiing van negentig graden kan maken binnen de twee seconden. Deze versnelling kan mogelijk niet bereikt worden door het gebruik van overbrengingen. Niettegenstaande is het interessant om in de berekeningen met een bepaalde veiligheidsmarge te werken.

Door te stellen dat een verdraaiing van negentig graden mogelijk is binnen de twee seconden, kan een grensvoorwaarde voor de koppelberekening opgesteld worden. Het koppel moet zo berekend worden dat de gebruikte motoren (met overbrenging) de last theoretisch zodanig kunnen versnellen dat er een hoekrotatie van negentig graden binnen de twee seconden kan gebeuren. Omdat het niet mogelijk is om de werkelijke versnelling te weten, is het mogelijk dat deze hoger ligt dan de vooropgestelde versnelling die berekend wordt uit de grensvoorwaarde. Om met dit scenario rekening te houden wordt een factor van anderhalf ingevoerd waarmee de berekende hoekversnelling vermenigvuldigd moet worden. Door deze veiligheidsfactor kan gesteld worden dat de snelheid volgens een driehoekig profiel loopt.

In figuur 6-2 wordt de snelheid en versnelling voorgesteld in functie van de tijd.



Figuur 6-2 Snelheids - en versnellingsprofiel

### 6.2.2.2.2 Hoekversnelling

Vanuit het driehoeksprofiel is duidelijk te zien dat er over een gelijke duur versneld en vertraagd wordt. Met de vooropgestelde grensvoorwaarde wil dit zeggen dat de as bij het omschakelen van versnelling naar vertraging met precies vijfenveertig graden groteerd is.

De beweging kan beschouwd worden als een eenparig (roterende) versnelde beweging met volgende vergelijking. De veiligheidsfactor van anderhalf werd reeds opgenomen.

	$\theta = \theta_0 + \omega_0 t + \frac{\alpha}{1.5} t^2$	(6.11)
--	---	--------

Met  $\theta_0$  de hoekverdraaiing en  $\omega_0$  de hoeksnelheid bij aanvang.

Doordat zowel de initiële hoekverdraaiing als de hoeksnelheid gelijk zijn aan nul, kan de hoekversnelling eenvoudig als volgt berekend worden:

	$\alpha = 1.5 * \frac{\theta}{t^2} = 1.5 * \frac{45^\circ * \frac{2\pi}{360^\circ}}{1^2} = \frac{3\pi \text{ rad}}{8 \text{ s}^2}$	(6.12)
--	--	--------

Uit deze hoekversnelling kan eenvoudig de tangentiële versnelling van iedere as bepaald worden die gebruikt wordt in de vereenvoudigde formule van de dynamische koppelberekening.

	$a = L_{armdeel} * \alpha$	(6.13)
--	----------------------------	--------

Met  $L_{armdeel}$  de lengte van de indeffector tot de beschouwde as.

#### 6.2.2.2.3 Hoeksnelheid

Uit de berekende hoekversnelling kan nu eenvoudig de maximale hoeksnelheid bepaald worden:

	$\omega_{maximaal} = \omega_0 + \alpha * \Delta t = \alpha * 1 = \frac{3\pi \text{ rad}}{8 \text{ s}}$	(6.14)
--	--	--------

### 6.2.2.3 Dynamische berekening assen één, vier en zes

Bij de berekening van assen één, vier en zes wordt de algemene formule beschouwd.

#### 6.2.2.3.1 As één

De componenten beschreven bij de statische berekening hebben een totale massa van elf en een halve kilogram. De basis van de arm weegt ongeveer acht kilogram.

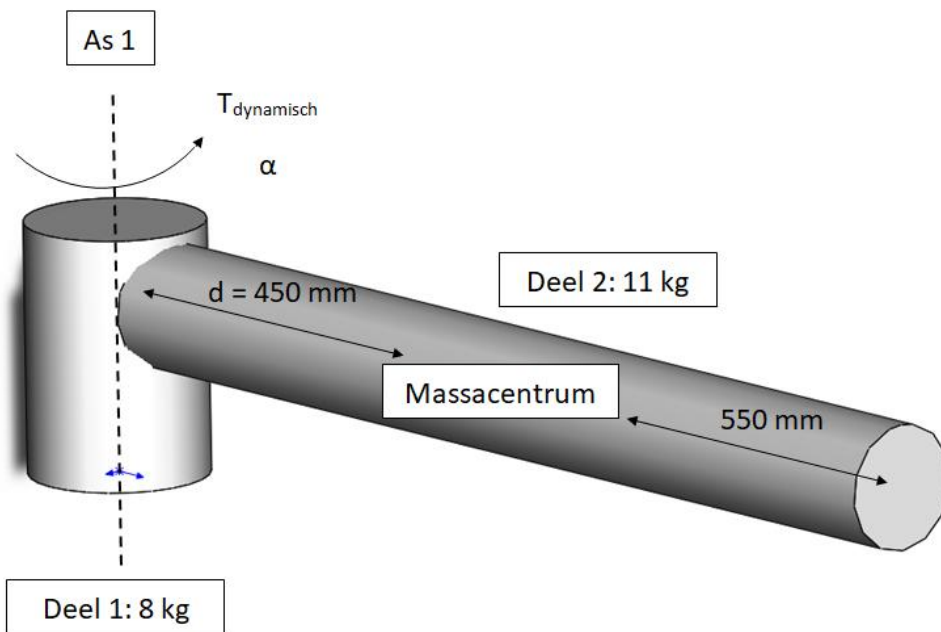
Vervolgens dient het massatraagheidsmoment bepaald te worden. Omdat dit voor heel wat rekenwerk zou zorgen, wordt hiervoor een vereenvoudiging ingevoerd.

De arm wordt opgesplitst in een basisdeel dat bestaat uit as één en het blok met daarop de motor voor as twee en in de arm zelf (waaruit de statische koppels berekend werden), die bestaat uit alle componenten na as twee. Beide worden beschouwd als volle cilinders, de gegevens zijn terug te vinden in tabel 6-1.

**Tabel 6-1 Gegevens dynamische berekening as één**

	Deel 1	Deel 2
<b>massa (kg)</b>	8	11
<b>straal (m)</b>	0.15	0.10
<b>lengte (m)</b>	0.25	1
<b>afstand tot rotatie-as (m)</b>	0	0.45
<b>statisch moment (Nm)</b>	0	0

In figuur 6-3 is een schematische voorstelling te zien.



**Figuur 6-3 Schematische voorstelling robotarm**

Het massa traagheidsmoment wordt berekend door de som te nemen van de twee traagheidsmomenten. Bij de arm zelf wordt de stelling van Steiner toegepast om de afstand tot de rotatie-as in rekening te brengen [67].

	$J_{As1} = J_{Deel1} + (J_{Deel2} + d^2 * m_{Deel2})$	(6.15)
	$J_{As1} = \frac{1}{2} m_{Deel1} R_{Deel1}^2 + \left( \frac{1}{12} m_{Deel2} * l_{Deel2}^2 + d^2 * m_{Deel2} \right)$ $= \frac{1}{2} * 8 * 0.15^2 + \left( \frac{1}{12} * 11 * 1^2 + 0.450^2 * 11 \right) = 3.23 \text{ kgm}^2$	(6.16)

Het koppel volgt vervolgens uit de vermenigvuldiging van het massa traagheidsmoment met de hoekversnelling:

	$T_{dynamisch,As1} = T_{statisch,As1} + J * \alpha = 3.23 * \frac{3\pi}{8} = 3.81 \text{ Nm}$	(6.17)
--	---	--------

#### 6.2.2.3.2 As vier

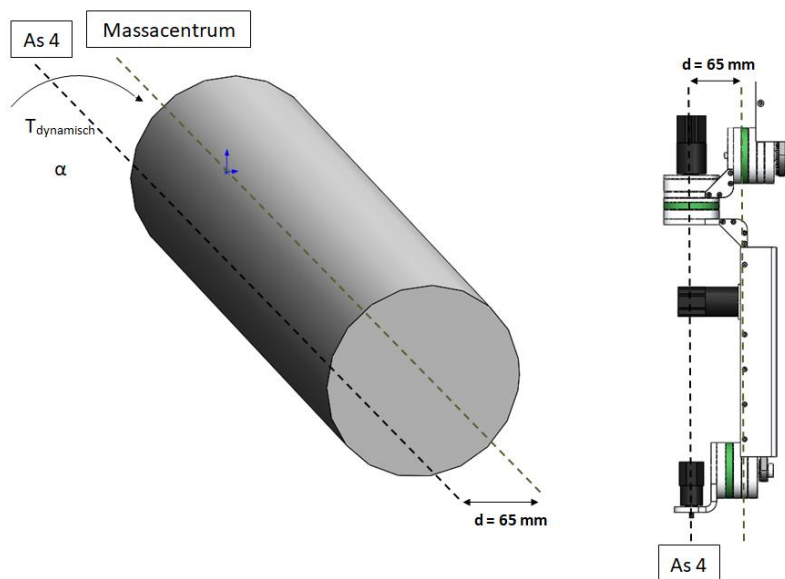
Op analoge wijze wordt as vier uitgewerkt. Hier worden de armdelen die invloed hebben op de as vervangen door een massieve cilinder die met zijn massacentrum verschoven ligt ten



opzichte van de rotatie-as. Aanvullende gegevens kunnen in tabel 6-2 gevonden worden. In figuur 6-4 is een schematische voorstelling te zien.

**Tabel 6-2 Gegevens dynamische berekening as vier**

	<b>As 4</b>
<b>massa (kg)</b>	4.50
<b>straal (m)</b>	0.10
<b>lengte (m)</b>	0.45
<b>afstand tot rotatie-as (m)</b>	0.065
<b>statisch moment (Nm)</b>	1.97



**Figuur 6-4 Schematische voorstelling momentwerking op as vier**

De berekening is nu analoog aan die van de eerste as:

$T_{dynamisch,As4} = T_{statisch,As4} + J * \alpha$ $= T_{statisch,As4} + \left( \frac{1}{2} * m_{As4} R_{As4}^2 + d^2 * m_{As4} \right) * \alpha$ $= 1.97 + \left( \frac{1}{2} * 4.5 * 0.1^2 + 0.065^2 * 4.5 \right) * \frac{3\pi}{8} = 2.07 Nm$	(6.18)
---	--------

De dynamische component is hier verwaarloosbaar klein.

### 6.2.2.3.3 As zes

De berekening van as zes is analoog. In tabel 6-3 zijn de benodigde gegevens weergegeven. Deze gegevens zijn aannames van wat een mogelijke belasting kan zijn. De berekening dient beschouwd te worden als richtlijn.

Er wordt verondersteld dat een pompoen van twee kilogram moet gerotereerd worden rond zijn massacentrum. De aslijn van de pompoen ligt dus in het verlengde van as zes.

Tabel 6-3 Gegevens dynamische berekening as zes

	As 6
massa (kg)	2
straal (m)	0.20
lengte (m)	0.40
statisch moment (Nm)	0

De berekening:

	$T_{dynamisch,As6} = T_{statisch,As6} + \left(\frac{1}{2} * m_{As4} R_{As4}^2\right) * \alpha$ $= 0 + \left(\frac{1}{2} * 2 * 0.2^2\right) * \frac{3\pi}{8} = 0.05 Nm$	(6.19)
--	--	--------

Het bekomen koppel is verwaarloosbaar klein. Voor de keuze van de motoren wordt een moment van 1 Nm verondersteld.

#### 6.2.2.4 Dynamische berekening assen twee, drie en vijf.

Omdat de assen twee, drie en vijf parallel liggen, zijn hun bewegingen niet onafhankelijk. Als enkel as twee versnelt en as drie en vijf blijven star, zal er een ander benodigd koppel uitgekomen worden dan als assen drie en vijf ook een versnelling ondergaan, los van de initiële versnelling.

Het slechtst mogelijke geval voor de koppelberekening van een bepaalde as, is als deze as en alle assen die parallel staan tegelijk maximaal aangestuurd worden.

Voor de berekening van deze assen wordt de aangepaste formule voor het dynamisch koppel gebruikt. Dit houdt in dat eerst de hoekversnellingen moeten omgezet worden in tangentiële versnellingen.

##### 6.2.2.4.1 Versnelling

Uit 6.2.2.2.2 volgt de volgende formule voor het omvormen van de hoekversnelling naar de tangentiële versnelling:

	$a = L_{armdeel} * \alpha$	(6.20)
--	----------------------------	--------

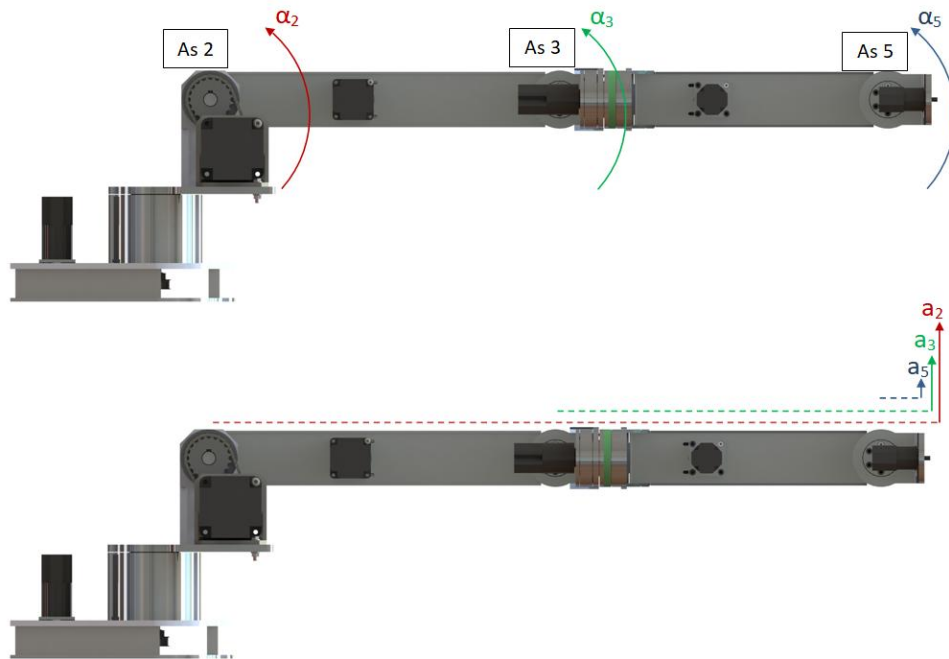
Er werd berekend dat de hoekversnelling telkens  $3\pi/8$  bedraagt.

Indien deze formule toegepast wordt op de versnelling rond de tweede as, dan moet als parameter  $L_{armdeel}$  de afstand van het TCP tot as twee genomen worden. Analooq geldt dit voor assen drie en vijf.

	$a_2 = (480 + 450.75 + 68) * 10^{-3} * \frac{3\pi}{8} = 1.17 \frac{m}{s^2}$	(6.21)
--	---	--------

	$a_3 = (450.75 + 68) * 10^{-3} * \frac{3\pi}{8} = 0.62 \frac{m}{s^2}$	(6.22)
	$a_5 = 68 * 10^{-3} * \frac{3\pi}{8} = 0.080 \frac{m}{s^2}$	(6.23)

De versnellingen worden in figuur 6-5 uitgetekend ter verduidelijking.



**Figuur 6-5 Verduidelijkende figuur bij de omvorming van de hoekversnelling**

Met behulp van deze tangentiële versnellingen kan nu het benodigd koppel berekend worden.

**As twee:**

Het maximaal benodigd koppel in as twee wordt bekomen als zowel as twee, drie en vijf maximaal aangestuurd worden.

De dynamische componenten worden van de verschillende armdelen opgeteld. De formule ziet er uiteindelijk als volgt uit:

	$T_{dynamisch,As2} = T_{statisch,As2} + \sum_{As\ 2-As\ 3-As\ 5} m * a * hefboom$	(6.24)
	$T_{dynamisch,As2} = T_{statisch,As2} + \sum_{As\ 2-As\ 3-As\ 5} \frac{T_{statisch,As} * a}{g}$ $met\ g \cong 10 \frac{m}{s^2}$	(6.25)

	$T_{dynamisch,As2} = 76.28 + \frac{76.28 * 1.17}{10} + \frac{23.06 * 0.62}{10} + \frac{1.51 * 0.080}{10}$ $= 86.47 Nm$	(6.26)
--	--	--------

De dynamische component stelt ongeveer tien procent van het geheel voor en is dus zeker niet te verwaarlozen. Analoog worden de dynamische koppels voor as drie en vijf berekend.

**As drie:**

Hier wordt het deel beschouwd van as drie tot aan het tool center point.

	$T_{dynamisch,As3} = T_{statisch,As3} + \sum_{As\ 3-As\ 5} \frac{T_{statisch,As} * a}{g}$	(6.27)
	$T_{dynamisch,As2} = 23.06 + \frac{23.06 * 0.62}{10} + \frac{23.06 * 0.080}{10} = 24.50 Nm$	(6.28)

**As vijf:**

Bij deze as is er geen keten van versnellingen. De dynamische component is enkel afhankelijk van de versnelling rond de vijfde as.

	$T_{dynamisch,As5} = T_{statisch,As5} + \frac{T_{statisch,As5} * a}{g}$	(6.29)
	$T_{dynamisch,As2} = 1.51 + \frac{1.51 * 0.08}{10} = 1.52 Nm$	(6.30)

**6.2.3 Conclusie berekeningen**

Aan de hand van de berekende waarden kunnen nu de motoren gekozen worden. Aangezien de koppels die hierboven weergegeven worden, berekend zijn met de grensvoorwaarde in gedachte, is het wenselijk dat de motoren (met overbrenging) een hoger koppel kunnen leveren dan hier is beschreven.

Uit de berekeningen kan ook opgemaakt worden dat de dynamische component vaak verwaarloosbaar klein is. Dit komt doordat de armdelen slechts kleine versnellingen ondergaan. Om later rekenwerk te vereenvoudigen worden alle statische koppels vermenigvuldigd met een factor 1.25 en vervolgens naar boven afgerond. De gebruikte koppels in verdere berekeningen zijn in tabel 6-4 weergegeven. Deze koppels zijn vermoedelijk een stuk hoger dan de werkelijke koppels, dit zorgt ervoor dat de robotarm in praktijk een hogere payload zal hebben dan beschreven.

**Tabel 6-4 Afgeronde koppels**

	<b>Koppel (Nm)</b>
<b>As 1</b>	5
<b>As 2</b>	108
<b>As 3</b>	31
<b>As 4</b>	3
<b>As 5</b>	2
<b>As 6</b>	1

## **6.2.4 Motoren**

### **6.2.4.1 Stappenmotor versus Servomotor**

Stappenmotoren en servomotoren verschillen op een aantal belangrijke vlakken, namelijk de manier waarop ze gebouwd zijn, hoe ze gestuurd worden en de manier waarop ze hun koppel leveren [68] [69] [70].

#### *6.2.4.1.1 Constructie*

Het grootste verschil in de constructie is het aantal polen van de elektromotoren. Zo heeft een stappenmotor veel polen (bv. 100), terwijl een servomotor slechts een aantal polen heeft (bv. 10).

De polen zijn een natuurlijk stoppunt voor de motor. Aangezien een stappenmotor veel polen heeft, zijn er veel stabiele posities. Hierdoor kan een stappenmotor gestuurd worden naar een bepaalde hoek zonder dat hierbij terugkoppeling nodig is. Servomotoren hebben een positie-encoder nodig die hun positie bijhoudt.

#### *6.2.4.1.2 Sturing*

Een stappenmotor sturen is erg eenvoudig. Om de motor met één stap te verdraaien (per rotatie zijn er evenveel stappen als poolparen) dient er slechts een pulstrein gestuurd te worden. Aangezien vooraf geweten is wat de hoekverdraaiing per stap is, kan er zo eenvoudig tot een bepaalde rotatiehoek gekomen worden.

Een servomotor daarentegen leest het verschil uit tussen de gewenste positie en de huidige positie (met behulp van de encoder). Vervolgens wordt de juiste stroom gestuurd om de beweging uit te voeren.

Door de eenvoudige sturing van stappenmotoren zijn ze een stuk goedkoper dan servomotoren.

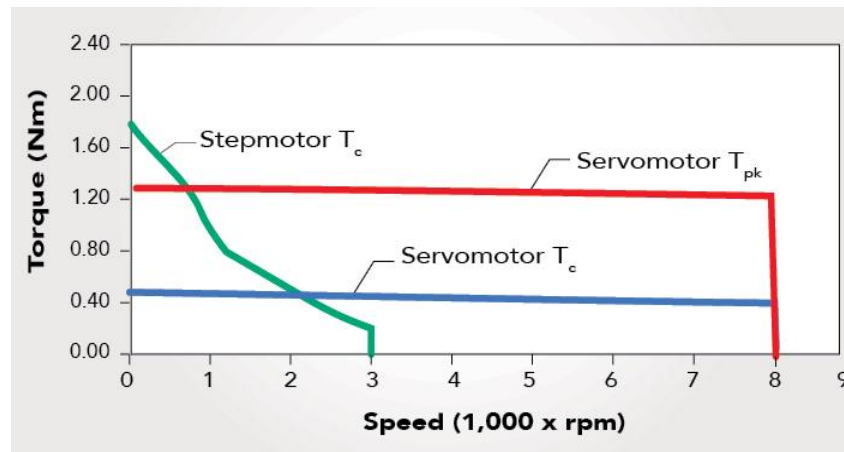
#### *6.2.4.1.3 Koppel en snelheid*

Stappenmotoren kunnen door hun ontwerp een constant houdkoppel leveren. Bij lage toerentallen leveren ze een hoger koppel per volume dan bij servomotoren. Echter daalt hun koppel sterk bij stijgend toerental. Bovendien kunnen stappenmotoren niet overbelast worden, omdat ze dan het risico lopen om stappen over te slaan. Servomotoren daarentegen hebben geen houdkoppel, maar kunnen tot twee maal hun nominaal koppel leveren voor korte tijd. Hun nominale koppel neemt marginaal af met stijgend toerental.

In figuur 6-6 zijn bovenstaande bemerkingen duidelijk te zien. Hierbij staat  $T_c$  voor het nominale koppel en  $T_{pk}$  voor het piekkoppel.

#### 6.2.4.1.4 Conclusie

Stappenmotoren zijn geschikt voor applicaties waarbij lage snelheden en hoog houdkoppel nodig zijn, terwijl servomotoren eerder geschikt zijn voor toepassingen waarbij hogere snelheden en hoge positieprecisie belangrijk zijn.



Figuur 6-6 Koppel van een stappenmotor en een servomotor [69]

#### 6.2.4.2 Keuze type motor

Voor een gelede arm robot is het gebruik van servomotoren iets beter dan stappenmotoren. Het werken volgens een closed-loop systeem (met terugkoppeling), zorgt ervoor dat er met zekerheid een positie bereikt wordt. Aangezien stappenmotoren open-loop systemen zijn (zonder terugkoppeling), kan de positie niet gegarandeerd worden, zeker niet bij overbelasting.

In praktijk werd er echter gekozen voor stappenmotoren, voornamelijk omdat deze een stuk goedkoper zijn dan servomotoren. Bovendien is de aansturing ervan erg gemakkelijk, waardoor er een korte leercurve is. Via eindeloopsensoren kan er indien nodig voor positieterugkoppeling gezorgd worden.

#### 6.2.4.3 Keuze van de stappenmotoren met gearbox en de overbrengingsratio van de riem-/kettingoverbrenging

De keuze van de motoren gebeurde, om organisatorische redenen, voor de eindfase van het ontwerp. Er werd uitgegaan van het - op dat moment berekende - statische koppel, dat vervolgens met een factor twee vermenigvuldigd werd. Hierdoor zijn sommige motoren en overbrengingen een stuk overgedimensioneerd.

De gekozen motoren met hun overbrengingen zijn te vinden in tabel 6-5. Deze gegevens werden uit de datasheets gehaald van de verschillende motoren en zijn in de documentatie van het tekendossier "AgroBot A.R.M." terug te vinden.

Tabel 6-5 Gegevens motoren

	Motor 1	Motor 2	Motor 3	Motor 4	Motor 5	Motor 6
<b>Benodigd koppel</b>	5	108	31	3	2	1
<b>Gekozen motor</b>	17HS15-1684S	34HS31-5504S	23HS22-2804S	17HS15-1684S	17HS15-1684S	14HS13-0804S
<b>Houdkoppel (Nm)</b>	0.39	4.50	1.03	0.39	0.39	0.18
<b>Gekozen gearboxratio</b>	15/1	30/1	50/1	15/1	10/1	5/1
<b>Gekozen overbrenging riem/ketting</b>	1.25/1	1.25/1	1.42/1	1/1	1/1	1/1
<b>Uiteindelijk houdkoppel (Nm)</b>	7.35	168.75	73.13	5.85	3.90	0.90

Uit deze tabel is duidelijk op te maken dat de gewenste koppels behaald worden, met uitzondering van de laatste motor. Dit vormt echter geen probleem, omdat de 1 Nm-grens daar arbitrair gekozen werd. Motor drie is sterk overgedimensioneerd.

Om de robotarm sneller te laten bewegen is het mogelijk om later de overbrengingsverhouding die gebruikt werden bij de motoren te verlagen door de kettingwielen of tandriemschijven te vervangen.

## 6.3 Controle overbrengingen

In deze paragraaf worden de verschillende types overbrengingen gecontroleerd.

### 6.3.1 Tandriem- en kettingoverbrenging

Initieel werden alle assen met de motoren verbonden door middel van tandriemen. Deze zijn namelijk erg goedkoop en onderhoudsvriendelijk. Ze hebben geen smering nodig om de goede werking te garanderen. Bovendien is het gebruik van een tandriem op zich eenvoudiger doordat ze flexibeler zijn en makkelijker op te spannen [71].

Bij de praktische uitvoering bleek echter dat het gebruik van een tandriem bij as twee en drie voor problemen zorgde. Door de omvang van het overgebrachte koppel begon de tandriem te 'slippen' op de tandriemschijf. Door extra hard op te spannen kon tijdelijk een effectieve koppeloverdracht bekomen worden. Aangezien die geen garantie gaf tot probleemloos gebruik, werd er geopteerd voor het gebruik van een kettingaandrijving ter vervanging van de tandriemen bij as twee en drie.

Dit heeft als voordeel dat de robot preciezer zal kunnen stoppen, doordat de ketting niet rekt zoals de riem. Bovendien is het niet meer mogelijk om te slippen. Het nadeel aan een kettingoverbrenging is dat ze moet gesmeerd worden, zeker omdat zowel de ketting als het kettingwiel vervaardigd zijn uit C45-staal en niet uit roestvast staal of aluminium zoals de

overige componenten. Op termijn zouden de kettingen en kettingwielen kunnen vervangen worden door alternatieven in roestvast staal.

### 6.3.1.1 Kettingoverbrenging

Doordat de ketting per vijf meter gekocht moest worden en er slechts een klein deel nodig was voor de robotarm, werd de keuze voornamelijk gemaakt op vlak van herbruikbaarheid voor andere projecten. Daarom is er gekozen voor kettingwielen met dubbele vertanding.

#### 6.3.1.1.1 Type

De kettingen en kettingwielen zijn genormeerd volgens ISO 606 (DIN 8187) en zijn vervaardigd uit C45-staal. De belangrijkste parameter van de ketting is de steek, namelijk 9.525 mm of 3/8 inch. Een deel van de ketting wordt afgebeeld in figuur 6-7.



**Figuur 6-7 Ketting ISO 606 06B-2 [72]**

#### 6.3.1.1.2 Berekening

Op voorhand is intuïtief duidelijk dat de gekozen ketting overgedimensioneerd zal zijn voor het aandrijven van as twee en as drie van de robotarm. Aangezien tweemaal dezelfde ketting gebruikt wordt, is het slechts interessant om de kracht in de ketting te berekenen bij de as waarbij het hoogste koppel aanwezig is, namelijk as twee.

In de gegevens meegeleverd door de fabrikant van de ketting is te zien dat de kracht tot breuk minimaal 17600N bedraagt [73].

Nu kan eenvoudig de trekkracht in de ketting berekend worden door de kracht te beschouwen die ter hoogte van de steekcirkel op het kettingwiel heerst ten gevolge van het overgebrachte koppel. Deze kracht zal zowel bij aandrijfzijde als bij gedreven zijde gelijk zijn, doordat het koppel mee transformeert met de overbrengingsverhouding, dat dan weer bepaald wordt door het aantal tanden of de respectievelijke steekcirkels van de kettingwielen.

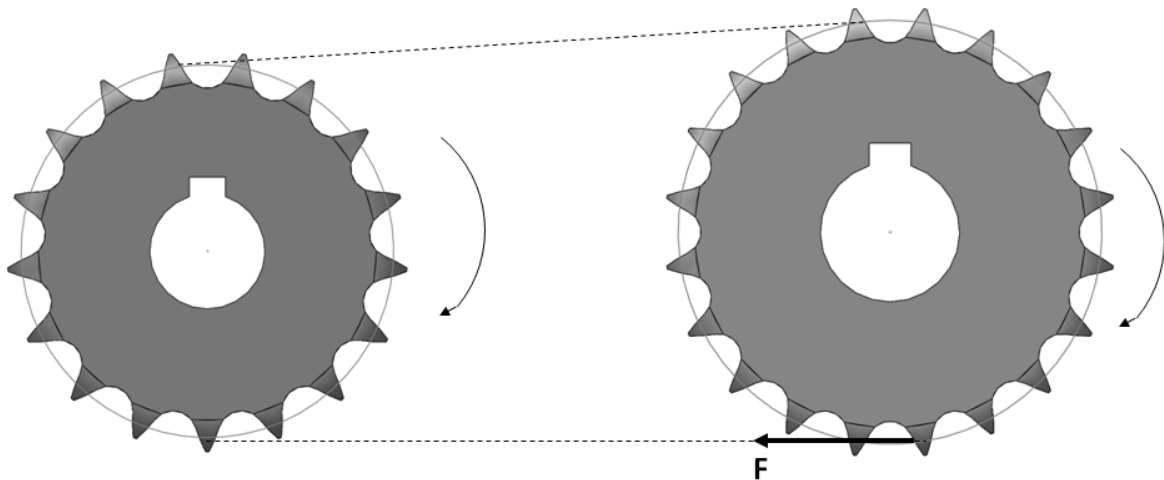
De berekende kracht is de kracht op de tand van het tandwiel, maar uit de derde wet van Newton kan gesteld worden dat deze kracht een gelijke kracht in tegengestelde richting veroorzaakt in de ketting.

Hieronder een korte berekening aan de aangedreven zijde:

	$F = \frac{T}{\frac{\text{steekcirkel}}{2}} = \frac{108000 \text{ Nmm}}{\frac{60.89}{2} \text{ mm}} = 3547.38 \text{ N}$	(6.31)
--	--	--------



Deze kracht op het kettingwiel ligt volgens de richting van de ketting en is zichtbaar op figuur 6-8.



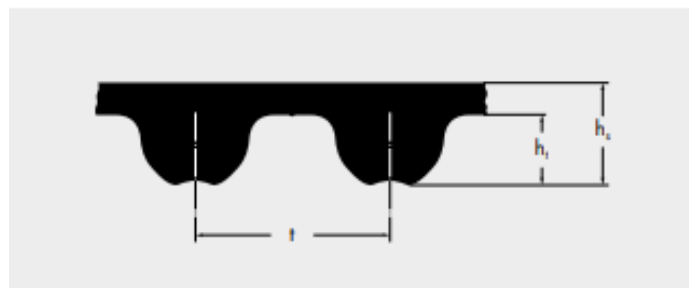
**Figuur 6-8 Voorstelling kracht op tandwiel**

Uit deze berekening is duidelijk dat de ketting zeker voldoende sterk zal zijn.

### **6.3.1.2 Tandriemoverbrenging**

#### **6.3.1.2.1 Type: Omega 3M**

Voor de tandriemoverbrengingen werd er telkens gebruik gemaakt van tandriemen van het merk Optibelt. Er werd gekozen voor tandriemen met een breedte van negen millimeter en als profielaanduiding 3M. In figuur 6-9 is dit tandriemprofiel zichtbaar.



**Figuur 6-9 Optibelt Omega 3M profiel [74]**

Belangrijke parameters die te zien zijn op dit profiel, zijn de tandhoogte  $h_t$  van 1.10 mm, de totale riemdikte  $h_s$  van 2.30 mm en tot slot de tandsteek  $t$  van 3 mm. De spanningsdraad in de tandriem is vervaardigd uit gevlochten glasvezel. Hierdoor hebben de riemen een hoge treksterkte en flexibiliteit en een lage rek [75].

De tandriemen kunnen gekoppeld worden met tandriemschijven van Optibelt, namelijk de HTD tandriemschijven met profiel 3M. Deze tandriemschijven zijn vervaardigd uit aluminium.

#### **6.3.1.2.2 Berekening**

Omdat er telkens dezelfde tandriem gekozen is, met als enige verschil de lengte, is het enkel nuttig om de tandriem na te rekenen die het grootste koppel overbrengt. Deze tandriem maakt de verbinding tussen motor één en as één. Het overgebrachte koppel aan de gedreven zijde is 5 Nm. De kracht in de tandriem zal zowel aan de kleinste als grootste tandriemschijf even groot zijn.

De trekkracht in de tandriem kan analoog berekend worden aan de trekkracht in de ketting:

	$F = \frac{T}{\frac{\text{steekcirkel}}{2}} = \frac{5000 \text{ Nmm}}{\frac{38.2}{2} \text{ mm}} = 261.78 \text{ N}$	(6.32)
--	--	--------

Een tandriem zal zo goed als nooit knappen door een te hoge trekkracht. Eerst zullen de tanden afslijten, waardoor de riem zal beginnen slippen in de tandriemschijf.

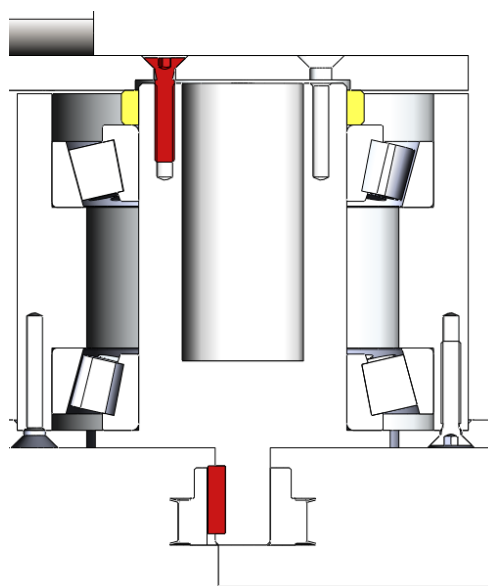
Aangezien er door de fabrikant geen concrete gegevens worden vrijgegeven, kan dit moeilijk gecontroleerd worden. Er mag echter aangenomen worden dat een kracht van een kleine driehonderd Newton geen probleem zal vormen voor de tandriem.

### 6.3.2 Bout - en spieoverbrengingen

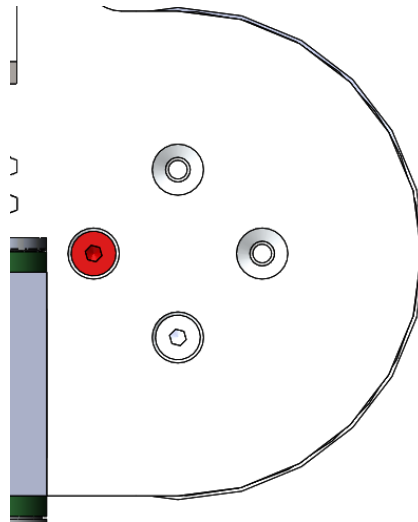
Om de robotarm gemakkelijk te kunnen monteren en demonteren, is er voor gekozen om persverbindingen zoveel mogelijk te vermijden.

De verschillende armdelen worden op de assen aan de ene zijde vastgezet met bouten. Aan de andere zijde is er telkens een spieoverbrenging met een tandriemschijf of kettingwiel, waarop via een tandriem of ketting het koppel van de motoren overgebracht wordt. De berekeningen van de spieën gebeurt telkens aan de zijde van de gedreven as, aangezien er mag van uitgegaan worden dat de spieën die meegeleverd zijn door de motorfabrikant correct gedimensioneerd zijn. Als de motor gebruikt wordt binnen het opgegeven koppel dat de fabrikant meegeeft zouden er zich geen problemen mogen voordoen. Bovendien zal het koppel aan de gedreven zijde hoger zijn dan aan de drijvende zijde waar gebruik werd gemaakt van overbrengingen die groter zijn dan één.

In figuur 6-10 is een doorsnede zichtbaar van deze twee verbindingen, gevolgd door een extra verduidelijkend aanzicht in figuur 6-11. Bij dergelijke overbrengingen is het belangrijk te berekenen wat de afschuifspanningen zijn in de bouten, spieën en de tandriemschijven of kettingwielen, zodat met zekerheid een breuk vermeden kan worden.



**Figuur 6-10 Doorsnede boutverbinding (zijaanzicht)**



**Figuur 6-11 Verduidelikend aanzicht boutverbinding (bovenaanzicht)**

Hieronder volgt de berekening van de eerste as, op gelijkaardige wijze werden alle assen gecontroleerd.

### 6.3.2.1 As één - Motor één

Via deze as wordt een maximaal koppel overgebracht van 5 Nm. De boutverbinding bestaat uit vier bouten M6 uit RVS(70). De spie heeft als afmetingen 5x5x20 en is ook vervaardigd in RVS.

#### 6.3.2.1.1 Gegevens voor de berekening

Vooraf aan de berekening zijn in tabel 6-6 de nodige gegevens te vinden.

**Tabel 6-6 Gegevens berekening schuifsterkte as één**

Overgebracht koppel $T = 5 \text{ Nm}$			
Boutverbinding		Spie	
Bouttype	M6	B	5 mm
Aantal	4	$L_{\text{eff}}$	15 mm
Kerndiameter	5 mm	$\varnothing_{\text{as}}$	16 mm
Buitendiameter	5.7 mm	$t_1$	2.3 mm
Afstand C-C	22.5 mm	$t_2$	3 mm
$\sigma_{\text{toel,bout}}$	700 N/mm <sup>2</sup>	$\sigma_{\text{toel,tandriemschijf}}$	540 N/mm <sup>2</sup>
$T_{\text{toel,bout}}$	350 N/mm <sup>2</sup>	$T_{\text{toel,tandriemschijf}}$	270 N/mm <sup>2</sup>

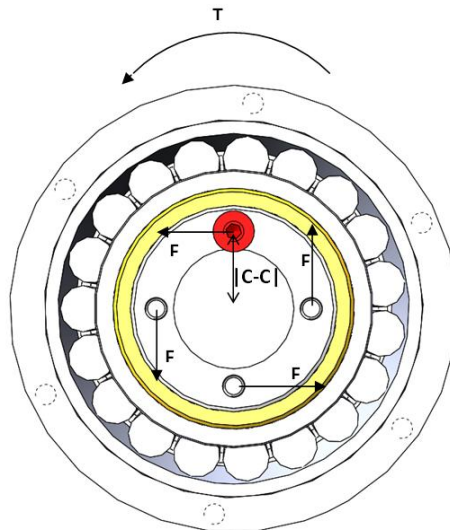
#### 6.3.2.1.2 Berekenen afschuifsterkte bouten

Om de sterkte van de bouten te controleren beschrijft Roloff/Matek [76] twee mogelijke berekeningen. De eerste mogelijkheid is het bepalen van de benodigde totale doorsnede van de bouten, rekening houdende met aandraaimoment en de voorspankracht. De tweede berekening beschouwt de bouten als pennen die enkel belast worden op schuifspanning, waarna de maximale afschuifkracht kan gecontroleerd worden. Omdat deze tweede methode

een stuk intuïtiever is en geen gebruik maakt van tabellen en complexe formules, wordt deze toegepast.

Aangezien dit een vereenvoudigde manier van berekening is, zal er bij controle van de maximale schuifspanning gerekend worden met de 0.5 keer de maximale normaalspanning (volgens het criterium van Tresca) in plaats van 0.8 keer de maximale normaalspanning, zoals normaal gehanteerd wordt .

In figuur 6-12 is een koppel T te zien van 5Nm. Dit koppel wordt overgebracht door middel van vier bouten M6.



**Figuur 6-12 Boutverbinding - krachten/koppel**

In onderstaande formule is de afstand C-C de afstand tussen het centrum van de as en het centrum van de bouten. De kracht in elke bout ten gevolge van het koppel wordt als volgt berekend:

	$F = \frac{T}{\#bouten * afstand C - C} = \frac{5000 \text{ Nmm}}{4 * 22.5 \text{ mm}} = 55.55 \text{ N}$	(6.33)
--	---	--------

Deze kracht veroorzaakt schuifspanningen in de bouten die als volgt kunnen berekend worden:

	$\tau = \frac{F}{A} = \frac{55.55 \text{ N}}{\frac{(5 + 5.7)^2 * \pi}{4}} = 2.45 \frac{\text{N}}{\text{mm}^2}$	(6.34)
--	--	--------

	$\tau_{max} = \frac{3}{2} * \tau = 3.67 \frac{\text{N}}{\text{mm}^2}$	(6.35)
--	---	--------

Bij het berekenen van de oppervlakte van de bouten wordt de diameter gekozen als het gemiddelde van de kerndiameter de buitendiameter, zoals deze te vinden zijn in het tabellenboek voor metaaltechniek [77]. De bekomen schuifspanning is de schuifspanning in

het centrum van de bout en is een maat voor de gemiddelde schuifspanning in de bout. Omdat de hoogste schuifspanning, gelegen aan de rand van buitendiameter van de bout, hoger is, wordt telkens een factor van anderhalf ingerekend als extra veiligheidsmarge.

Vervolgens wordt er gecontroleerd als deze schuifspanning lager is dan de maximaal toegelaten schuifspanning.

Doordat het gaat om een bout in roestvast staal met kengetal 70, is de maximaal toegelaten normaalspanning  $\sigma_{toel}$  gelijk aan 700 N/mm<sup>2</sup>.

Vervolgens wordt de maximale schuifspanning berekend volgens het criterium van Tresca, dat stelt dat een schuifspanning twee maal gevaarlijker is dan een normaalspanning [78].

	$\tau_{toel} = 0.5 * \sigma_{toel} = 350 \frac{N}{mm^2}$	(6.35)
--	--	--------

Het is duidelijk zichtbaar dat de schuifspanning in dit geval niet voor een breuk zal zorgen. De reden dat bouten M6 gekozen zijn, is dan ook niet uit gevaar voor breuk door schuifspanning, maar omdat in dit specifieke geval een groot deel van de robotarm rust op de plaat die vastgemaakt is op deze as.

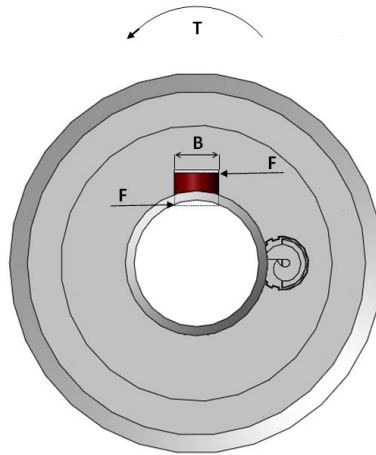
#### 6.3.2.1.3 Berekening afschuifsterkte spie

Om vervolgens de afschuifspanning te berekenen in de spie worden figuren 6-13 en 6-14 gegeven. Hierop zijn de benodigde afmetingen aangeduid. In figuur 6-13 staan de krachten die op de spie werken ten gevolge van het tegenwerkend moment, ontwikkeld door de tandriemschijf. Met onderstaande formule wordt de schuifspanning in een spie berekend [79]:

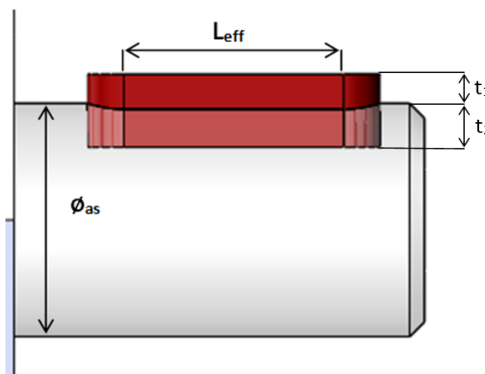
	$\tau = \frac{T}{B * L_{eff} * \frac{\phi_{as}}{2}} = \frac{5000}{5 * 15 * 8} = 8.33 \frac{N}{mm^2}$	(6.36)
--	--	--------

De toegelaten schuifspanning wordt wederom berekend volgens het criterium van Tresca. De treksterkte van RVS spieën is niet gegeven, daarom wordt deze gekozen op de minimale treksterkte van roestvast staal A4 (AISI316), namelijk 480 N/mm<sup>2</sup> [80].

Ook hier is het duidelijk dat er geen risico voor afschuiving is. De spieën dienen in feite niet nagerekend te worden, omdat deze bij correct gebruik telkens sterker zijn dan de as of naaf. Bij de berekening van de andere assen wordt dit deel niet opgenomen.



**Figuur 6-13** Vooraanzicht spie met schuifkrachten ten gevolge van het koppel T



**Figuur 6-14** Zijaanzicht met extra afmetingen

#### 6.3.2.1.4 Berekening vlaktespanning ten gevolge van een spie

Uit de vorige paragraaf kan werd vastgesteld dat de schuifspanning van een spie in feite niet moet nagerekend worden, maar dat het wel interessant is om de vlaktespanning in de as of de naaf te berekenen. Vooraf wordt er afgewogen welke component uit het sterkste materiaal vervaardigd is.

Voor deze berekening wordt figuur 6-13 geadapteerd. Daar zijn de krachten weergegeven op de spie, waar bij deze berekening de krachten weergegeven zijn op de as en de naaf, zoals te zien op figuur 6-15. De spanningen op as, naaf en spie worden gelijk verondersteld. Aangezien in dit specifieke geval de naaf in aluminium vervaardigd is en de as in roestvast staal, wordt enkel de naaf beschouwd.

De spanning wordt als volgt berekend [81]:

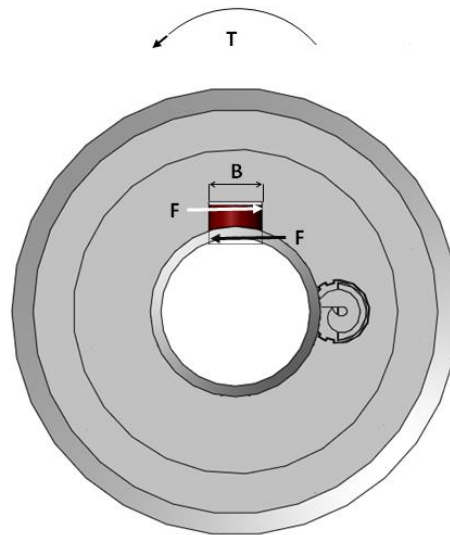
	$\sigma = \frac{T}{t_1 * L_{eff} * \frac{\phi_{as}}{2}} = \frac{5000}{2.3 * 15 * 8} = 18.12 \frac{N}{mm^2}$	(6.38)
--	---	--------

Omdat de gebruikte aluminiumlegering voor de tandriemschijven niet vermeld wordt in de documentatie, kan de maximale normaalspanning niet bepaald worden. Daarom wordt er, zoals eerder beschreven bij de materiaalkeuze, vanuit gegaan dat de treksterkte gelijk is aan

die van de aluminiumlegering volgens EN-AW 2017A, namelijk  $\pm 300 \text{ N/mm}^2$  [82]. De toegelaten vlaktespanning voor aluminium wordt berekend als volgt [83]:

	$\sigma_{toel\text{-}vlaktedruk} = \frac{\sigma_{toel}}{2} = 150 \frac{N}{mm^2}$	(6.39)
--	--	--------

Het is duidelijk dat de toegelaten vlaktedruk veel hoger is dan de berekende.



**Figuur 6-15 Krachten op de naaf**

#### 6.3.2.1.5 Conclusie

Zowel de boutverbinding als de spie zullen door het overgebrachte koppel een schuifspanning te verduren krijgen. Eveneens zal er een kleine vlaktespanning ontstaan in de tandriemschijven en assen.

#### 6.3.2.2 Andere assen

De andere assen werden op analoge manier berekend. Deze berekeningen zijn telkens in tabelvorm terug te vinden in bijlage C.

#### 6.3.2.3 Conclusie

Na de berekening van de bouten en de spieën valt op te merken dat de meeste onderdelen overgedimensioneerd zijn. Bij massaproductie zouden alle bouten en spieën herrekend kunnen worden, om zo goedkopere componenten te gebruiken. Echter is de extra marge voor dit prototype een pluspunt, want dit maakt het mogelijk om later nog aanpassingen te doen aan het ontwerp zonder dat er extra rekening moet gehouden worden met de sterkte van deze verbindingen.

## 6.4 Lagere

### 6.4.1 Type

Tijdens het ontwerp van de robotarm moest er gekozen worden tussen glijlagers en wentellagers. Doordat de lagerhuizen uit aluminium gemaakt zijn dat minder hard is dan de stalen assen, zouden glijlagers kunnen zorgen voor slijtage in het huis. Bovendien kan er

door de momentwerking op de armen een grote radiale krachtcomponent komen op het glijlager, waardoor plaatselijke belasting kan optreden. Om die redenen werd er gekozen om wentellagers te gebruiken, meer specifiek kegellagers.

Kegellagers zijn een veilige keuze, omdat deze zowel radiale als axiale krachtcomponenten kunnen opnemen. Voor de assen bij de armdelen zouden kogellagers ook voldoen, omdat daar de axiale krachten slechts gering zijn. Echter, doordat de kegellagers door hun omvang niet veel duurder waren dan gewone kogellagers, werd er beslist om telkens hetzelfde type lager te gebruiken. Kegellagers worden gekenmerkt door hun buiten- en binnenring met conische loopbanen met daartussen kegelvormige rollen. In afbeelding 6-16 wordt een enkelrijig kegellager afgebeeld [84].



**Figuur 6-16 Kegellager 30206 J2-Q [85]**

## **6.4.2 Montage**

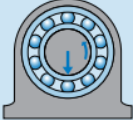
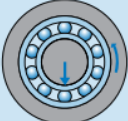
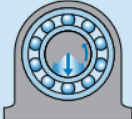
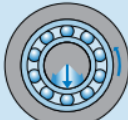
De montage en demontage van een kegellager is eenvoudig doordat de buiten- en binnenring afzonderlijk kunnen gemonteerd worden. Aan de hand van de gekozen passing kan de montage-techniek bepaald worden.

### **6.4.2.1 Passing**

Met behulp van de montagegids van SKF wordt de correcte passing bepaald. Een onderdeel van die gids is te zien in figuur 6-17. Eerst dienen de werkomstandigheden bepaald te worden. Doordat de armdelen meebewegen met de rotatie, gaat het om een last die mee roteert met de binnenring, terwijl de buitenring vaststaat. Voor dit type belasting raadt SKF een vaste passing aan van de buitenring en een losse passing voor de binnenring [86].

De buitenring zal met behulp van een hydraulische pers in het huis geperst worden, terwijl de binnenring over de as kan glijden.



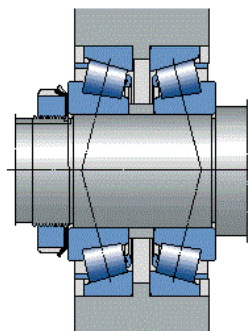
Conditions of rotation and loading				
Operating conditions	Schematic illustration	Load condition	Example	Recommended fits
Rotating inner ring Stationary outer ring Constant load direction		Rotating load on inner ring Stationary load on outer ring	Belt-driven shafts	Interference fit for inner ring Loose fit for outer ring
Stationary inner ring Rotating outer ring Constant load direction		Stationary load on inner ring Rotating load on outer ring	Conveyor idlers Car wheel hub bearings	Loose fit for inner ring Interference fit for outer ring
Rotating inner ring Stationary outer ring Load rotates with inner ring		Stationary load on inner ring Rotating load on outer ring	Vibratory applications Vibrating screens or motors	Interference fit for outer ring Loose fit for inner ring
Stationary inner ring Rotating outer ring Load rotates with outer ring		Rotating load on inner ring Stationary load on outer ring	Gyratory crusher (Merry-go-round drives)	Interference fit for inner ring Loose fit for outer ring

**Figuur 6-17 Rotatie en belastingsvoorwaarden**

De tolerantieklasse voor de as en het huis kunnen ook bepaald worden met behulp van de montagegids. Voor de as geldt tolerantieklasse g6, voor het huis M7.

#### 6.4.2.2 Opstelling

Doordat een éénrijig kegellager slechts aan één zijde axiale belasting kan opnemen, wordt dit type lager gebruikelijk per twee geïnstalleerd. Voor de robotarm is er gekozen voor een O-opstelling, zoals te zien is in figuur 6-18.



**Figuur 6-18 O-opstelling lager [87]**

## 6.5 Sturing en programmering

### 6.5.1 Programmeeromgeving: Arduino

Arduino is een gratis open-source platform dat gebaseerd is op eenvoudig te gebruiken software en hardware. Het logo is te zien in figuur 6-19. Arduino borden zijn standaard voorzien van ingangen en uitgangen die vanuit het platform eenvoudig in C++ te programmeren zijn.

Door de korte leercurve van Arduino kunnen tal van eenvoudige projecten gerealiseerd worden. Dit brengt met zich mee dat de borden op grote schaal kunnen gefabriceerd worden, waardoor de kostprijs erg laag is.

Doordat Arduino open-source is, kunnen andere printplaatfabrikanten hun eigen versie maken [88].



Figuur 6-19 Arduino logo [89]

### 6.5.2 Controllino Mega

Controllino maakte gebruik van de technologie van Arduino om zo een geïndustrialiseerde variant van Arduino te maken. De borden worden beschreven als industry-ready PLC's. Het logo van Controllino is te zien in figuur 6-20.



Figuur 6-20 Controllino logo [90]

Er wordt gebruik gemaakt van Controllino controllers, omdat deze een hogere betrouwbaarheid hebben. Bovendien beschikken ze over een ethernetpoort, waardoor ze makkelijk in een netwerk op te nemen zijn.

Om alle motoren aan te sturen zijn er achttien uitgangen nodig. Uit het assortiment van Controllino voldoet enkel de Mega aan deze eis. Deze is afgebeeld in figuur 6-21.



Figuur 6-21 Controllino Mega [91]

De Controllino Mega beschikt over 23 digitale outputs (DO) en 15 analoge inputs (AI). Door het hoge aantal inputs kan er eenvoudig gebruik gemaakt worden van bedieningsknoppen of joysticks om de robotarm in een bepaalde positie te brengen.

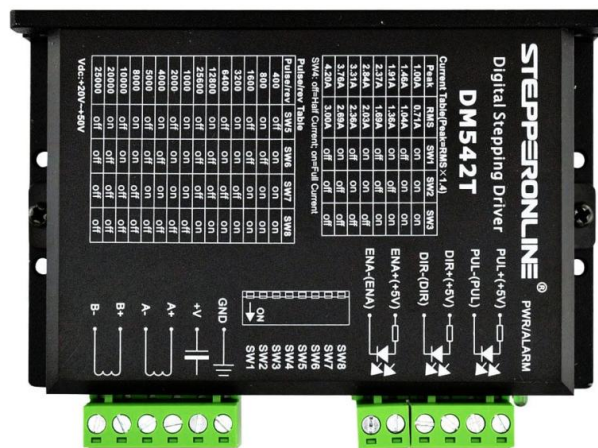
Alle ingangen en uitgangen zijn op de klemmenrij beschikbaar op 12/24V of op de pinheaders op 5V. Bij de aansluiting dient hier rekening mee gehouden te worden [91].

## 6.6 Drivers

### 6.6.1 Algemeen

De verschillende motoren kunnen niet rechtstreeks aangesloten worden op de Controllino, omdat ze een te hoge stroom vragen. Daarom worden ze aangesloten op drivers, toestellen die een lage-stroom stuursignaal omzet in een hoge-stroom signaal dat de motor stuurt [92].

In figuur 6-22 wordt een digitale stepping driver weergegeven. In deze paragraaf wordt de werking uitgelegd.



Figuur 6-22 Digitale driver [93]

#### 6.6.1.1 Aansluitingen

Het lage-stroom stuursignaal wordt verzonden via de Controllino en komt toe op de PUL/DIR/ENA-klemmen. Hierbij dient de PUL-klem (pull) om de de pulsen te sturen, de DIR-klem (direction) om de draairichting van de motor te bepalen en de ENA-klem (enable) om de motor al dan niet te bekrachtigen.

De signalen worden vervolgens omgezet naar hoge-stroom signalen die op de klemmen A+/A-/B+/B- komen.

De driver wordt gevoed door middel van een 48V DC bron via de +V- en GND-klem.

#### 6.6.1.2 Stroomtabel

Aan de hand van de gegevens die bij de motor meegeleverd zijn, kunnen via de stroomtabel de schakelaars SW1-3 aan de zijkant van de driver juist ingesteld worden, zodat de driver de juiste motorstroom levert.

Standaard staan deze schakelaars zodanig ingesteld dat de minimale stroom geleverd wordt. Echter zal vaak het maximale koppel van de motor niet behaald kunnen worden met deze instelling. Door de schakelaars zodanig te zetten dat een hogere stroom geleverd wordt

dan de nominale stroom per fase van de motoren, zal een hoger koppel mogelijk zijn. Dit zal ervoor zorgen dat de motoren te warm worden en hierdoor een verkorte levensduur hebben.

Tenslotte kan met schakelaar SW4 de stroom gehalveerd worden. Dit is ideaal om testen uit te voeren.

### **6.6.1.3 Pulsen/omwenteling tabel**

Met behulp van schakelaars SW5-8 kan het aantal benodigde pulsen om één omwenteling te verdraaien ingesteld worden. Deze benaming is dubieus, want het aantal benodigde pulsen voor een omwenteling van een motor is afhankelijk van het aantal polen en de overbrengingsverhouding. Hieronder een kort rekenvoorbeeld:

Stel dat het aantal pulsen per omwenteling dient berekend te worden voor de motor 17HS15-1684S met een totale overbrengingsverhouding van 10/1.

Uit de datasheets wordt een staphoek van  $1.8^\circ$  gevonden, wat overeenkomt met 200 poolparen. Per omwenteling moeten er dus 200 pulsen gestuurd worden. Dit wordt vervolgens vermenigvuldigd met factor 10 door het gebruik van een 10/1-overbrenging. Om één omwenteling te maken zijn dus 2000 pulsen nodig.

Vervolgens kunnen op de driver de schakelaars zodanig geplaatst worden zodat het aantal pulsen per toer op de driver overeenkomt met die het benodigd aantal pulsen van de motor. Als nu bijvoorbeeld 4000 pulsen per toer ingesteld wordt, zal de motor de helft trager draaien.

## **6.6.2 Keuze drivers**

Aangezien de motoren aangekocht werden bij OMC Stepper Online, is het aangewezen om de bijhorende drivers ook daar te kopen. Na overleg met de leverancier werd duidelijk dat voor de robotarm best digitale drivers gebruikt worden. Door digitale drivers te gebruiken, produceren de motoren namelijk minder warmte en lawaai terwijl een hoger koppel bereikt kan worden in vergelijking met analoge drivers.

### **6.6.2.1 Spanningsbron 48V**

De hoofdspanningsbron van de AgroBot levert 48V. De drivers worden zoveel mogelijk gekozen zodat deze spanning binnen hun toegelaten spanningsbereik valt. Binnen het gamma van Stepper Online kan gekozen worden uit de DM520T, DM556T en DM860T.

Bij de productomschrijving blijkt de DM520T specifiek voor motoren van het type Nema 17 en Nema 23 te zijn. De driver heeft een stroombereik van 1-4.2A en is hiermee geschikt voor motoren één, drie, vier en vijf.

Voor de Nema 34 kan er gekozen worden tussen de DM556T en de DM860T. Met een stroombereik tussen de 1.8A en 5.6A is de DM556T net krachtig genoeg, maar zonder enige marge. Daarom wordt er gekozen voor de DM860T, die een stroombereik van 2.4-7.2A heeft.

### **6.6.2.2 Spanningsbron 24V**

Voor de Nema 14 is er geen enkele driver beschikbaar die op 48V kan gevoed worden. Al deze drivers hebben namelijk een minimale uitgangsstroom hoger dan de stroom nodig voor de gekozen motor. Aangezien de controller met 24V gevoed wordt, kan deze bron ook dienen om de driver van de zesde motor te voeden.

De enige geschikte driver in het gamma van Stepper Online is de DM320T met een stroombereik van 0.3-2.2A. Dit is ideaal voor de gekozen motor.

In tabel 6-7 zijn de nodige gegevens nogmaals te vinden van elke motor en driver.

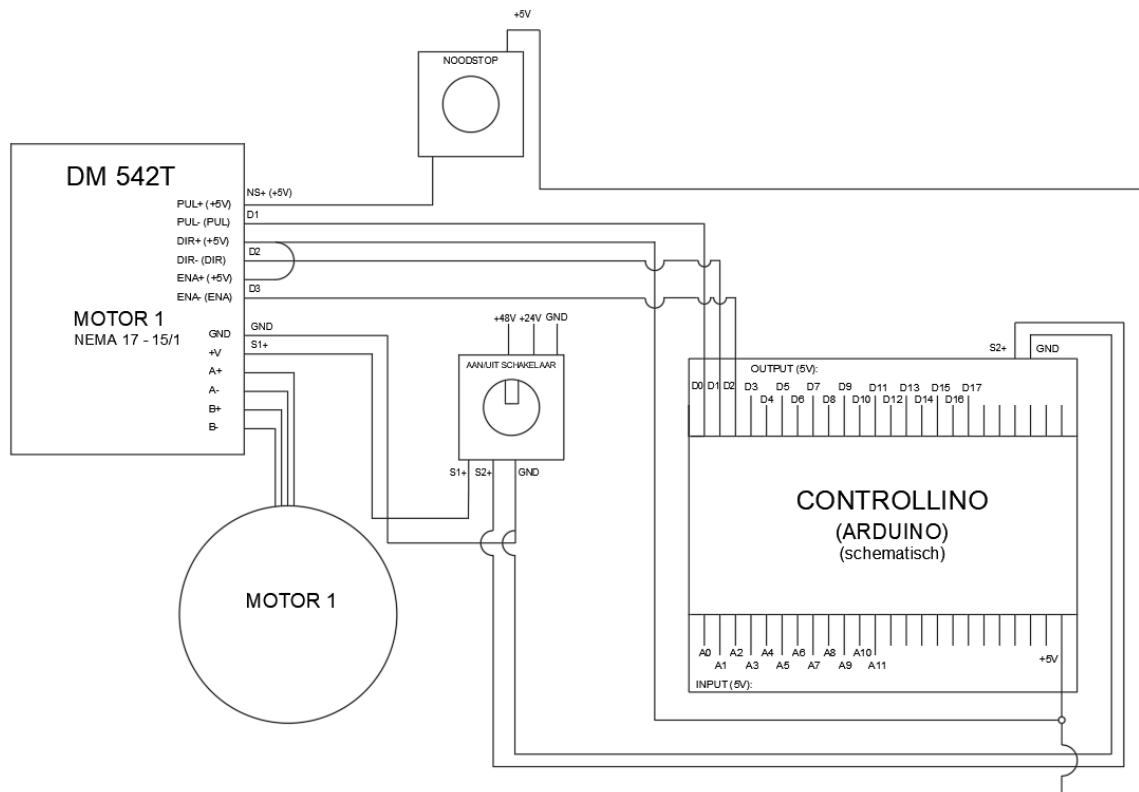
Tabel 6-7 Gegevens motoren en drivers

Motor	Stroom / fase (A)	Driver	Stroombereik (out) (A)	Spanningsbereik (in) (V)
Motor 1, 4, 5 - Nema 17	1.68	DM 320T	0.3-2.2	18-30 (DC)
Motor 2 - Nema 34	5.50	DM 542T	1.0-4.2	20-50 (DC)
Motor 3 - Nema 23	2.80	DM 860T	2.4-7.2	36-110 (DC)
Motor 6 - Nema 14	0.80			

## 6.7 Elektrische aansluiting

### 6.7.1 Schema

De motoren van de robotarm worden via de drivers van stroom voorzien. Deze drivers worden aangestuurd via de Controllino. In figuur 6-23 wordt de bekabeling van één motor weergegeven. In bijlage D is de volledige aansluiting terug te vinden.



Figuur 6-23 Elektrisch aansluitschema

## **6.7.2 Korte toelichting aansluitingen**

### **6.7.2.1 Controllino**

De outputs van de Controllino worden aangeduid met de letter D gevolgd door het nummer van de uitgang, analoog geldt dit voor de ingangen, waarbij het nummer wordt voorafgegaan door de letter A. Via de uitgangen zullen de verschillende parameters van de drivers aangestuurd worden. Aan de ingangen kunnen schakelaars of knoppen aangesloten worden om zo de robotarm manueel te besturen. Dit kan handig zijn als de robot vastzit ten gevolge van een singulariteit of om de werking van de verschillende motoren afzonderlijk te testen.

### **6.7.2.2 Driver: DM542T**

De driver die voorgesteld wordt in het schema volgt uit de bepaling van de drivers in paragraaf 6.6. Deze wordt gevoed door de 48V-spanningsbron via de aan/uit schakelaar.

### **6.7.2.3 Motor 1**

De motor wordt aangesloten op de driver. De stuursignalen verzonden door de Controllino worden dan versterkt door de driver en naar de motor doorgegeven.

### **6.7.2.4 Noodstop**

De noodstop onderbreekt de spanning van het stuursignaal bij een gevaarlijke situatie. Meer over de noodstopwerking in paragraaf 6.8.1.

## **6.8 Veiligheidsvoorschriften**

### **6.8.1 Noodstop**

Bij het gebruik van de noodstop is ten allen tijden de veiligheid van de gebruiker prioritair. Daarom dient de robotarm bij het indrukken van de noodstop meteen te stoppen met alle handelingen, zonder dat deze naar beneden valt.

Dit wordt bekomen door de noodstop te plaatsen op het stuursignaal, waardoor de motoren niet meer aangestuurd worden na bekrachtiging van de noodstop. De motoren blijven gevoed door de drivers en behouden zo hun houdkoppel, waardoor ze blijven staan.

Nadelig aan deze opstelling is dat de voedingsspanning niet onderbroken wordt, waardoor er bijvoorbeeld een blijvend brandgevaar is. Indien de noodstop zou geplaatst worden op de voeding van de drivers, zal de robotarm naar beneden vallen bij het indrukken van de noodstop. Met een massa van ongeveer elf kilogram kan dit makkelijk voor ernstige verwondingen zorgen.

Een ideale situatie zou bekomen worden als de motoren zouden beschikken over een interne rem. Dergelijke systemen vallen echter niet binnen het opgelegde budget.

### **6.8.2 Andere veiligheidsinrichtingen**

Aangezien de robotarm op een veldrobot geplaatst zal worden, zijn er weinig mogelijkheden om de robotarm extra te beveiligen. Er kan namelijk geen afscherming geplaatst worden, ook kan er geen laserscanner of lichtgordijn gebruikt worden.

Door de gebruikers correct op te leiden en waarschuwingsstickers op de robotarm te plaatsen, zoals te zien in figuur 6-24, kan het aantal gevaarlijke situaties gereduceerd worden.



**BEWEGENDE  
ONDERDELEN**



**Figuur 6-24 Waarschuwing [94]**

Bij een verdere uitwerking van het prototype kan er een aanvullend onderzoek uitgevoerd worden om de robotarm veiliger te maken. Ook kan er uitgezocht worden hoe de noodstopfunctie veiliger kan, zodat de eindgebruiker minder risico loopt.

# 7 BEPALEN VAN DE VOORWAARTSE EN INVERSE KINEMATICA VAN DE AGROBOT A.R.M.

---

## 7.1 Inleiding

In dit hoofdstuk zal de voorwaartse en inverse kinematica van de zes-assige AgroBot A.R.M. uitgewerkt worden. Wegens de complexiteit van de materie en de gewenste verstaanbaarheid voor verder gebruik, zal in de mate van het mogelijke de nodige uitleg voorzien worden.

Bij het gebruik van de robotarm is het wenselijk om zowel de motoren van de arm afzonderlijk te kunnen sturen, als het kunnen sturen van de arm naar een gewenste positie in een driedimensionale ruimte. De controller zal in dit laatste geval moeten geprogrammeerd worden om de verschillende motoren correct aan te sturen, zodat de armdelen in de juiste oriëntatie komen, waardoor de gewenste positie bereikt wordt.

Voor aanvang van de wiskundige kant van het probleem, wordt hieronder kort beschreven wat kan verstaan worden onder voorwaartse en inverse kinematica.

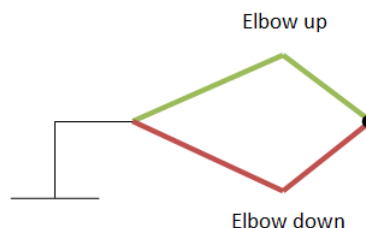
Het begrip kinematica wordt als volgt gedefinieerd:

*"Kinematica is een onderdeel van de klassieke mechanica die de beweging van punten, objecten en systemen beschrijft, zonder rekening te houden met de massa of de benodigde kracht om de beweging te veroorzaken."* [95]

Voorwaartse kinematica verwijst naar het gebruik van kinematische vergelijkingen om aan de hand van de gespecificeerde parameters van de verschillende gewrichten de eindpositie en de oriëntatie van de eindeffector te berekenen [96].

Inverse kinematica is zoals de naam doet vermoeden de inverse van de voorwaartse kinematica. Hier wordt uitgegaan van een bepaalde uitgangspositie met bijhorende oriëntatie. Hieruit kunnen de benodigde hoeken van de armdelen in de gewrichten bepaald worden aan de hand van de inverse kinematische vergelijkingen [97].

Belangrijk om weten, is dat er bij de inverse kinematica soms verschillende bewegingsmogelijkheden zijn om tot een bepaald punt te komen. Zo is er vaak een *elbow-up* of *elbow-down* configuratie. Dit wordt schematisch voorgesteld in figuur 7-1. Welk van de bewegingsmogelijkheden in werkelijkheid gebruikt wordt, zal afhangen van de voorafgaande parametrisering.



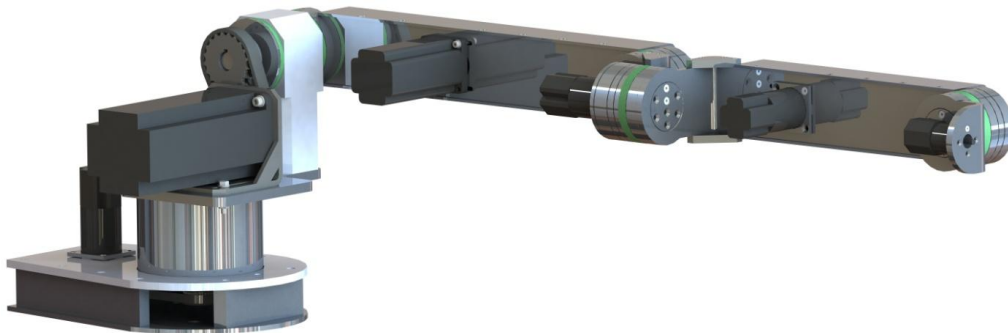
**Figuur 7-1 Elbow up - Elbow down**

Verder bestaat ook nog het probleem van singulariteiten. Dit zijn situaties waarbij er oneindig veel mogelijke configuraties zijn om naar het gewenste eindpunt te gaan. Tijdens de beweging naar een bepaald punt kan de robot in zo'n singulariteit terechtkomen, dit heeft als



gevolg dat de robot moet uitgeschakeld worden en manueel moet verplaatst worden naar een mogelijke positie. Singulariteiten kunnen vermeden worden door er rekening mee te houden bij de programmering [98].

In dit hoofdstuk volgt de mathematische uitwerking van de kinematica van de AgroBot A.R.M. die afgebeeld is op figuur 7-2.



**Figuur 7-2 AgroBot A.R.M.**

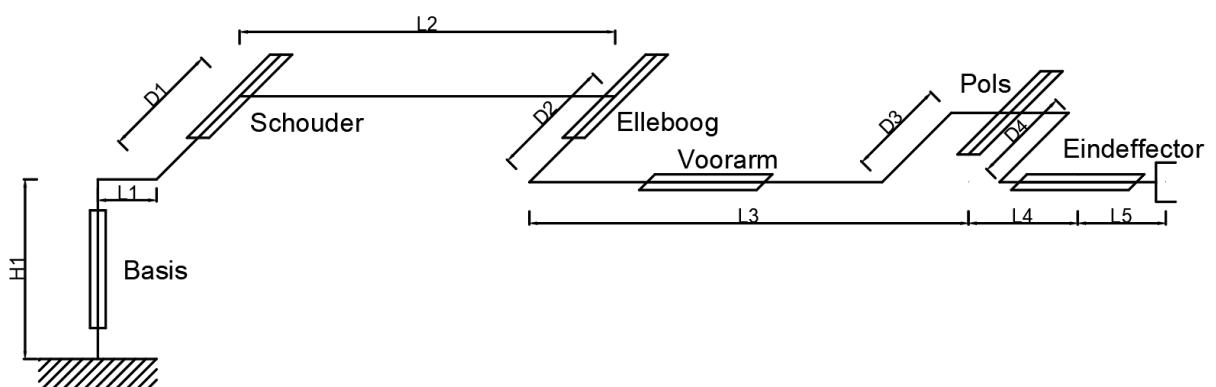
## 7.2 Voorwaartse kinematica

Zoals vermeld in 7.1 wordt de voorwaartse kinematica gebruikt om de positie en oriëntatie te vinden van de robotarm aan de hand van de hoeken waaronder de armdelen staan. [99] [100].

### 7.2.1 Kinematische keten

De eerste stap bestaat erin om een diagram uit te tekenen van de kinematische keten. Dit diagram geeft de verschillende armdelen van de robot weer, samen met hun onderlinge positie en oriëntatie.

In figuur 7-3 wordt het kinematisch diagram van de AgroBot A.R.M. weergegeven.



**Figuur 7-3 Kinematisch diagram (op basis van linkerzij aanzicht)**

Hierop zijn alle benodigde afmetingen weergegeven, zoals gemeten in het 3D model in Solidworks. Deze gegevens zijn noodzakelijk voor de berekening van zowel de voorwaartse als de inverse kinematica. In tabel 7-1 de aangeduide afmetingen.

**Tabel 7-1 Afmetingen tussen scharnieren**

	Afmeting (mm)		Afmeting (mm)
L1	75	H1	278
L2	480	D1	87.25
L3	450.75	D2	99
L4	56	D3	76
L5	12	D4	76

Verder zijn de verschillende gewrichten aangeduid in figuur 7-3. Deze zijn veelal vernoemd naar de gewrichten zoals bij de menselijke arm. De bedoeling hiervan is om makkelijk te kunnen volgen om welke rotaties het gaat. In tabel 7-2 staat een korte opsomming van de gewrichten.

**Tabel 7-2 Benaming gewrichten**

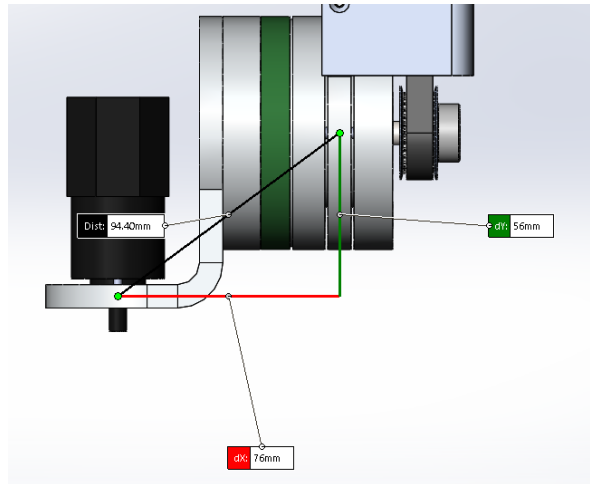
Gewricht #	Werkelijke benaming
1	Basis
2	Schouder
3	Elleboog
4	Voorarm
5	Pols
6	Eindeffector

Hierna moet er voor elke mogelijke rotatie een assenstelsel gedefinieerd worden, dit gebeurt volgens de Denavit-Hartenberg conventie. Deze assenstelsels liggen niet altijd in de gewrichten, om verwarring te voorkomen wordt er in paragraaf 7.2.2 een tabel gegeven met de benaming van de verschillende assenstelsels.

**Opmerking:**

De afmetingen van de verschillende armdelen ten opzichte van elkaar, werden telkens gemeten in de neutrale vezel van die armdelen.

Hieronder figuur 7-4 ter verduidelijking:



**Figuur 7-4 Meten afstanden**

De afstand H1 daarentegen is een uitzondering, daar die gemeten is van het centrum van het schoudergewricht tot aan de onderkant van de basisplaat en dus niet de neutrale vezel van de basisplaat. De onderkant van de basisplaat wordt beschouwd als de basis van het ruimtelijk vlak waarin de robot zal werken.

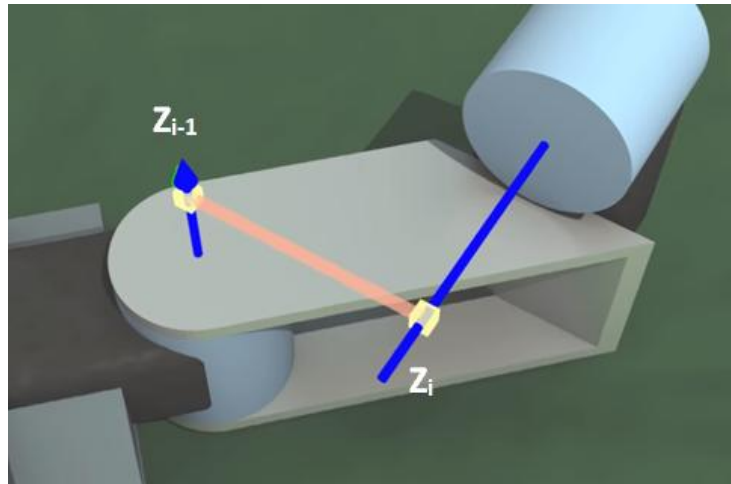
## 7.2.2 Denavit-Hartenberg (DH) conventie

Deze conventie bepaalt hoe de assenstelsels bij iedere rotatie zullen liggen. Belangrijk om vooraf op te merken is dat de assenstelsels volgens de rechterhandregel werken.

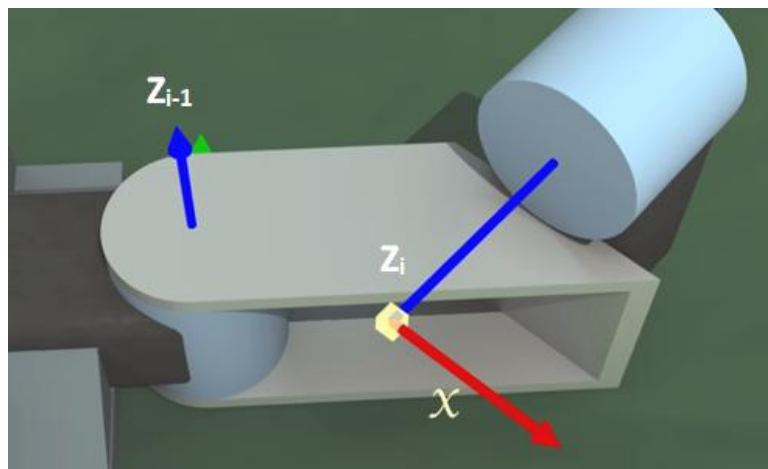
De Z-as wordt telkens gekozen volgens de as van de rotatie. Vervolgens moet de X-as zodanig gekozen worden zodat deze colineair<sup>8</sup> is met de gemeenschappelijke normaal. Deze gemeenschappelijke normaal wordt steeds bepaald tussen de Z-assen van twee opeenvolgende scharnieren en is de lijn die vanuit de aslijn van as  $Z_i$  loodrecht naar  $Z_{i-1}$  gaat. De nieuwe X-as zal zijn oorsprong hebben in het snijpunt van de gemeenschappelijke normaal met de  $Z_i$ -as en zal georiënteerd zijn volgens de gemeenschappelijke normaal. Met andere woorden is de nieuwe X-as telkens loodrecht op zowel  $Z_i$  als  $Z_{i-1}$ . Als er geen unieke loodrechte is, dan wordt de X-as gekozen in de richting van  $Z_{i-1}$  naar  $Z_i$ . Bijkomend moet de  $X_i$ -as telkens snijden met de  $Z_{i-1}$ -as. Dit brengt met zich mee dat de coördinaatframes niet steeds in de gewrichten zelf liggen. Figuur 7-5 toont duidelijk de gemeenschappelijke normaal, terwijl figuur 7-6 aantoont waar de X-as gelegen is [101] [102].

---

<sup>8</sup> Colineair = volgens dezelfde as



**Figuur 7-5 Gemeenschappelijke normaal (roze)**



**Figuur 7-6 X-as (rood)**

Tot slot wordt de Y-as zo gekozen zodat een rechtshandig assenstelsel bekomen wordt. Aan de hand van deze uitleg wordt hieronder het kinematisch diagram van de AgroBot A.R.M. bekomen.

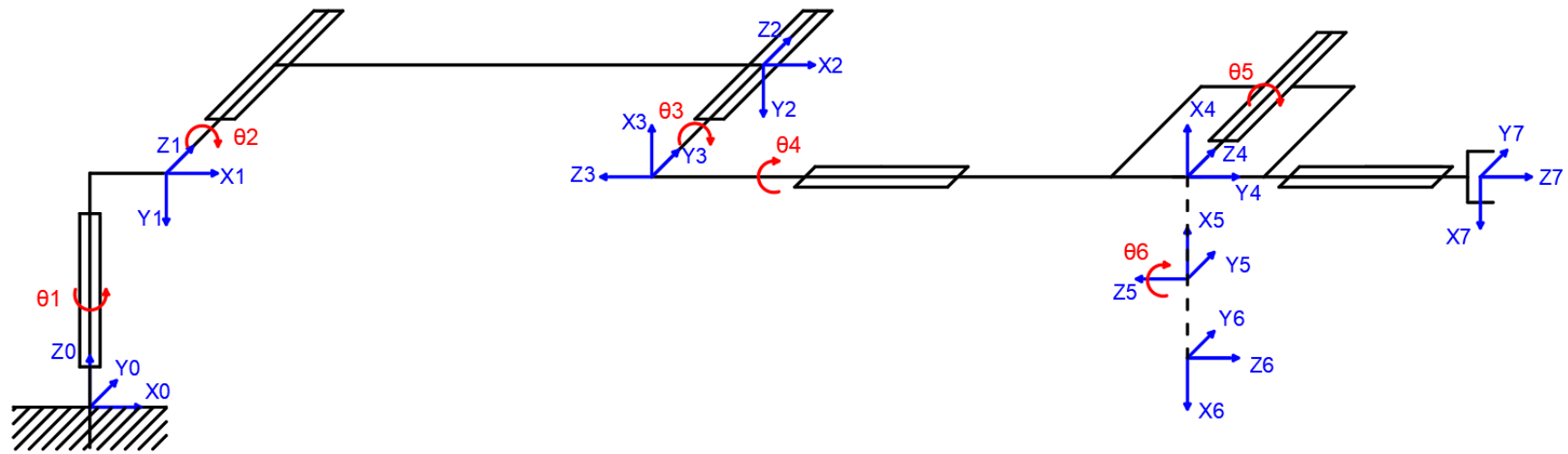
In tabel 7-3 worden alle assenstelsels benoemd. Sommige van deze namen komen overeen met de namen van de gewrichten, omdat een rotatie rond dit assenstelsel gelijk is aan een rotatie om het gewricht. Bij de pols zijn er meerdere rotaties mogelijk, vandaar de opsplitsing. Het laatste assenstelsel ligt in het Tool Center Point<sup>9</sup>.

De assenstelsels worden later ook beschreven als frames of coördinatenframes en worden door de letter F voorafgegaan.

In figuur 7-7 is het volledige kinematische diagram met de assenstelsels, bepaald via de DH-conventie, afgebeeld.

---

<sup>9</sup> Tool Center Point = TCP = centrale punt van de einaffector



Figuur 7-7 Kinematisch diagram volgens DH conventie

Tabel 7-3 Assenstelsels DH-conventie

Assenstelsel (frame)	Benaming
$F_0$	Basis
$F_1$	Schouder
$F_2$	Elleboog
$F_3$	Voorarm
$F_4$	Pols (rotatie $\theta_5$ )
$F_5$	Pols (rotatie $\theta_6$ )
$F_6$	Pols (TCP zonder offset)
$F_7$	TCP

Het basisassenstelsel  $F_0 (X_0, Y_0, Z_0)$  wordt als referentiepunt in de ruimte gekozen.

Waar assenstelsels  $F_4$ ,  $F_5$  en  $F_6$  samenkomen wordt het polsgewricht gedefinieerd. Dit om in de latere berekening van de inverse kinematica te kunnen terugrekenen naar dit punt. Verder is er nog het assenstelsel  $F_7$  in het TCP.

Alle rotaties rond de assenstelsels zijn positief volgens de rechterhandregel.

#### Opmerking 1:

Bij het basisassenstelsel mag de X-as vrij gekozen worden, omdat er geen gemeenschappelijke normaal is.

#### Opmerking 2:

De oorsprong van de X-as is niet steeds gelegen in het centrum van het gewricht, maar kan ook gelegen zijn in de open ruimte. Dit komt omdat de DH parameters die in paragraaf 7.2.3 uitgelegd worden, geen rekening houden met de fysieke plaats van de componenten, maar enkel met de beweging van de armdelen.

#### Opmerking 3:

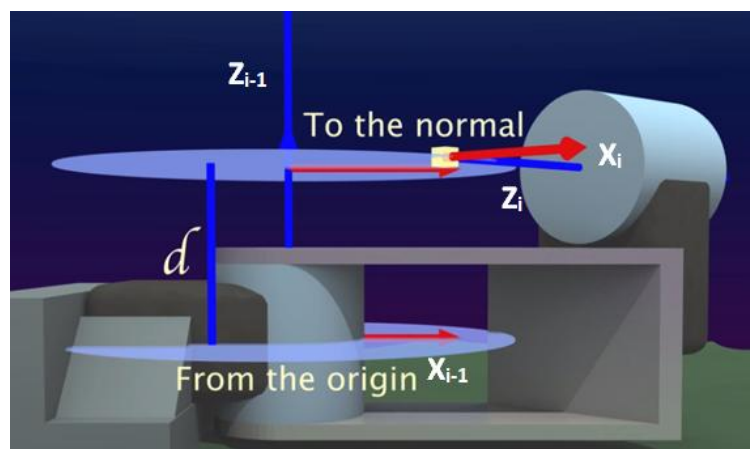
Het TCP-assenstelsel  $F_7$  worden in latere berekeningen gebruikt om de offset van tool center point ten opzichte het polsgewricht in rekening te brengen. Dit assenstelsel is echter niet gekoppeld aan een rotatie of een gewricht.

### 7.2.3 Denavit-Hartenberg (DH) parameters

De DH-parameters, die enkel geldig zijn voor de assenstelsels bekomen via de DH-conventie, worden gebruikt om de verschillende coördinatenframes te koppelen aan de armdelen in de kinematische keten. Aan de hand van de vier parameters:  $d$ ,  $\theta$ ,  $r$  en  $\alpha$  kunnen de transformaties van de verschillende assenstelsels ten opzichte van elkaar beschreven worden [103] [104] [105] [106].

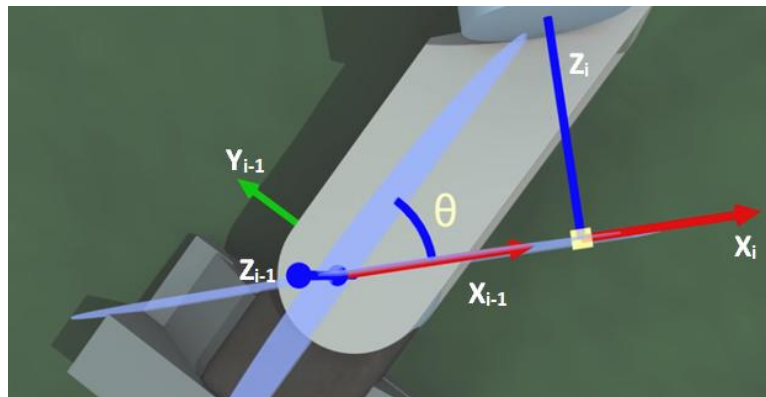
Een belangrijke opmerking is dat elk van deze parameters voorzien moet worden van een teken. Dit teken volgt uit de ligging van de assen en de gebruikte conventie voor positieve hoeken, namelijk de rechterhandregel.

De eerste parameter,  $d$ , is de afstand tussen de oorsprong van de as  $Z_{i-1}$  tot de gemeenschappelijke normaal. Dit wordt geïllustreerd in figuur 7-8.



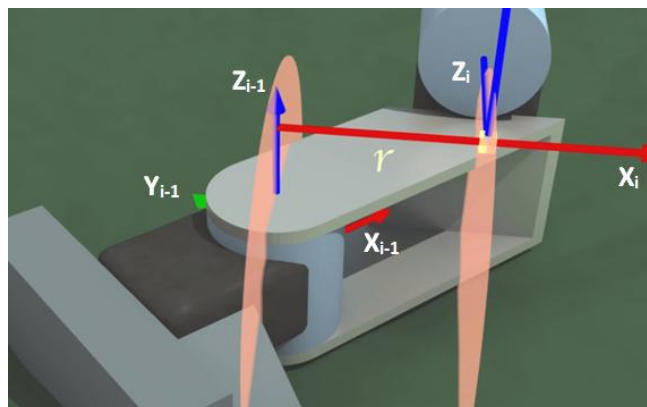
Figuur 7-8 Parameter  $d$

De tweede parameter,  $\theta$ , is de benodigde rotatie van de as  $X_{i-1}$  rond de as  $Z_{i-1}$  om parallel te staan met de  $X_i$ -as. Dit is duidelijk te zien in figuur 7-9. Later wordt deze hoek  $\theta$  beschouwd als de rotatiehoek van de armdelen.



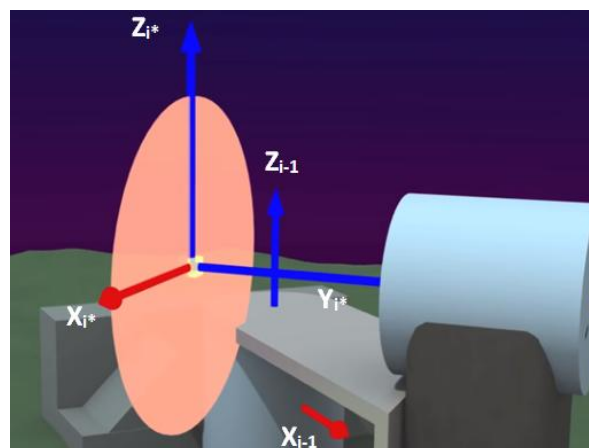
**Figuur 7-9 Parameter  $\theta$**

De derde parameter,  $r$ , is de lengte van de gemeenschappelijke normaal. Deze afstand is weergegeven in figuur 7-10. Dit kan ook eenvoudiger gezien worden als de afstand tussen het huidige en het vorige gewricht.



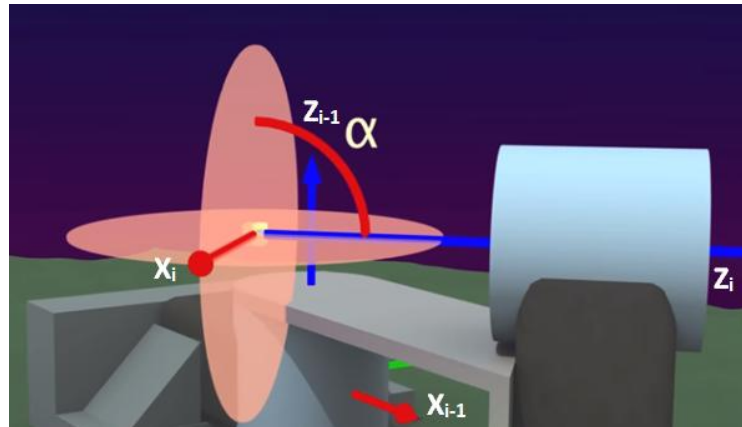
**Figuur 7-10 Parameter  $r$**

De laatste parameter,  $\alpha$ , is de hoek waarover  $X_i$  moet verdraaid worden om  $Z_i$  in de gewenste positie te krijgen, uitgaande van een rechtshandig assenstelsel en  $Z_i$  omhoog gericht.



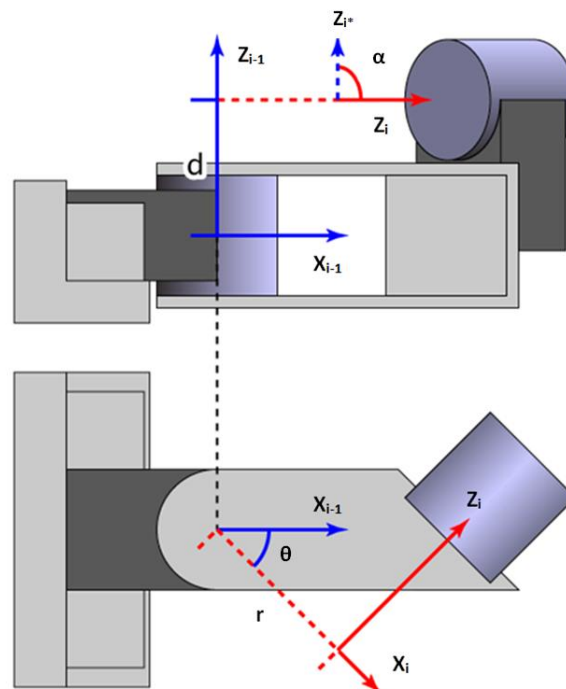
**Figuur 7-11 Beginsituatie**

In figuur 7-11 kan de beginsituatie beschouwd worden. In de gele oorsprong vertrekt de rode as  $X_i$ , gevormd met behulp van de gemeenschappelijke normaal. De  $Z_i$  - as wordt hier naar boven beschouwd en de  $Y_i$  - as naar rechts. Deze assen worden aangeduid met een sterretje om de tussensituatie aan te geven. Om de  $Z_i$  - as in de juiste richting te krijgen, namelijk als rotatie-as van het gewricht, moet deze over een hoek  $\alpha$  rond  $X_i$  verdraaid worden, zoals te zien in de figuur 7-12. De bekomen assen in deze figuur worden zonder sterretje voorgesteld, dit zijn de uiteindelijke assen.



**Figuur 7-12 Parameter  $\alpha$**

In figuur 7-13 worden alle parameters nogmaals weergegeven bij wijze van overzicht. Hierbij stelt de rode stippellijn de gemeenschappelijke normaal voor.



**Figuur 7-13 DH - Parameters [107]**

In tabel 7-4 staan de DH-parameters van de AgroBot A.R.M bepaald met behulp van bovenstaande uitleg.



Tabel 7-4 DH-Parameters A.R.M.

Frame #	r (mm)	d (mm)	$\alpha$ (rad)	$\theta$ (rad)
0-1	75	278	$-\pi/2$	$\theta_1$
1-2	480	87.25	0	$\theta_2$
2-3	0	-99	$\pi/2$	$\theta_{3^*}$
3-4	0	-450.75	$-\pi/2$	$\theta_4$
4-5	0	0	$\pi/2$	$\theta_5$
5-6	0	0	$\pi$	$\theta_{6^*}$
6-7	0	68	0	0

**Opmerking 1:**

De bovenstaande parameter  $\theta$  zal telkens wijzigen bij het verplaatsen van de armdelen ten opzichte van elkaar. In een later numeriek voorbeeld zullen concrete waarden ingegeven worden voor deze parameter. De hoeken  $\theta_i$  bij de DH-parameters komen overeen met de hoeken  $\theta_i$  in het kinematisch diagram.

**Opmerking 2:**

Er geldt:  $\theta_{3^*} = \theta_3 - \pi/2$ , en  $\theta_{6^*} = \theta_6 + \pi$ . Deze offset is een gevolg van de gekozen oriëntatie van het kinematisch diagram. Bij overgang van  $F_2$  naar  $F_3$  moet de as  $Z_2$  immers over een hoek  $\theta_3$  draaien van  $-90^\circ$  opdat de  $X_2$ - en  $X_3$ - assen volgens dezelfde richting zouden liggen. Een analoge redenering kan opgebouwd worden voor  $\theta_6$ . Bij de verdere berekeningen zullen deze hoeken altijd beschouwd worden inclusief hun offset.

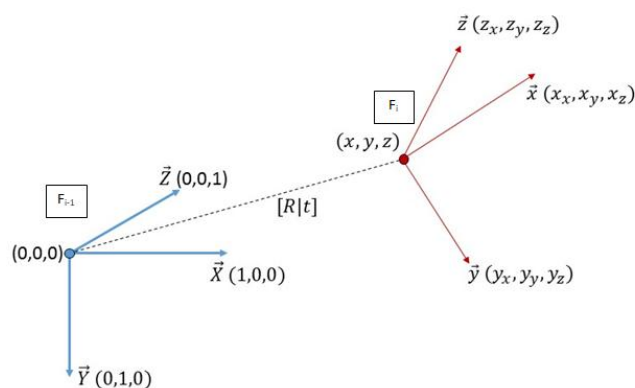
**Opmerking 3:**

Frame 7 stelt het extra ingevoerde assenstelsel voor aan het TCP.

**7.2.4 Transformatie tussen twee aangrenzende coördinatenframes**

**7.2.4.1 Transformatiematrix ter verduidelijking van rotatie en translatie**

Om de positie van de robotarm te kunnen bepalen is het nodig om te weten hoe de verschillende coördinatenframes zich verhouden ten opzichte van elkaar. Worden twee willekeurige coördinatenframes  $F_{i-1}$  en  $F_i$ , beschouwd, dan kunnen deze ten opzichte van elkaar geroteerd en getransleerd worden, zoals te zien in figuur 7-14 [108].



Figuur 7-14 Translatie en rotatie van twee frames [109]

Deze bewegingen ten opzichte van elkaar stellen een transformatie voor en worden voorgesteld door middel van een transformatiematrix T die gebaseerd is op de Denavit-Hartenberg conventie.

De transformatiematrix  ${}^{i-1}T_i$  om het coördinatenframe  $F_i$  te transformeren naar  $F_{i-1}$  wordt gegeven als volgt:

	${}^{i-1}T_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i * \cos\alpha_i & \sin\theta_i * \sin\alpha_i & r_i * \cos\theta_i \\ \sin\theta_i & \cos\theta_i * \cos\alpha_i & -\cos\theta_i * \sin\alpha_i & r_i * \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.1)
--	--	-------

Met de DH-parameters  $d$ ,  $\theta$ ,  $r$  en  $\alpha$  bepaald uit de onderlinge ligging van de assenstelsels.

Deze 4x4 transformatiematrix kan gesplitst worden in twee submatrices, namelijk een unieke translatiematrix en een unieke rotatiematrix. Uit de translatiematrix zal de positie bepaald worden van coördinatenframe  $F_i$  en uit de rotatiematrix volgt de oriëntatie van  $F_i$  ten opzichte van  $F_{i-1}$ .

	${}^{i-1}T_i = \begin{bmatrix} {}^{i-1}R_i & {}^{i-1}d_i \\ 0 & 1 \end{bmatrix}$	(7.2)
--	--	-------

Hieronder respectievelijk de unieke rotatiematrix en translatiematrix om van coördinatenframe  $F_i$  te transformeren naar  $F_{i-1}$ .

	${}^{i-1}R_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i * \cos\alpha_i & \sin\theta_i * \sin\alpha_i \\ \sin\theta_i & \cos\theta_i * \cos\alpha_i & -\cos\theta_i * \sin\alpha_i \\ 0 & \sin\alpha_i & \cos\alpha_i \end{bmatrix}$	(7.3)
--	---	-------

	${}^{i-1}d_i = \begin{bmatrix} r_i * \cos\theta_i \\ r_i * \sin\theta_i \\ d_i \end{bmatrix}$	(7.4)
--	---	-------

Deze redenering kan analoog opgebouwd worden voor de assenstelsels van de A.R.M..

#### **7.2.4.2 Transformatiematrices van de AgroBot A.R.M.**

Om de voorwaartse kinematica van de robotarm te bepalen, is het nodig om alle transformatiematrices uit te schrijven van de assenstelsels ten opzichte van elkaar. Hierbij worden de waarden uit tabel 7-4 gebruikt.

**Transformatie van frame  $F_1$  naar  $F_0$  (Schouder -> Basis):**

	${}^0T_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 * \cos\left(-\frac{\pi}{2}\right) & \sin\theta_1 * \sin\left(-\frac{\pi}{2}\right) & 75 * \cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 * \cos(0) & -\cos\theta_1 * \sin\left(-\frac{\pi}{2}\right) & 75 * \sin\theta_1 \\ 0 & \sin\left(-\frac{\pi}{2}\right) & \cos\left(-\frac{\pi}{2}\right) & 278 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.5)
	${}^0T_1 = \begin{bmatrix} \cos\theta_1 & 0 & -\sin\theta_1 & 75\cos\theta_1 \\ \sin\theta_1 & 0 & \cos\theta_1 & 75\sin\theta_1 \\ 0 & -1 & 0 & 278 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.6)

**Transformatie van frame  $F_2$  naar  $F_1$  (Elleboog -> Schouder):**

	${}^1T_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 * \cos(0) & \sin\theta_2 * \sin(0) & 480 * \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 * \cos(0) & -\cos\theta_2 * \sin(0) & 480 * \sin\theta_2 \\ 0 & \sin(0) & \cos(0) & 87.25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.7)
	${}^1T_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & 480 * \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & 480 * \sin\theta_2 \\ 0 & 0 & 1 & 87.25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.8)

**Transformatie van frame  $F_3$  naar  $F_2$  (Voorarm -> Elleboog):**

	${}^2T_3 = \begin{bmatrix} \cos\theta_3^* & -\sin\theta_3^* * \cos\left(\frac{\pi}{2}\right) & \sin\theta_3^* * \sin\left(\frac{\pi}{2}\right) & 0 * \cos\theta_3^* \\ \sin\theta_3^* & \cos\theta_3^* * \cos\left(\frac{\pi}{2}\right) & -\cos\theta_3^* * \sin\left(\frac{\pi}{2}\right) & 0 * \sin\theta_3^* \\ 0 & \sin\left(\frac{\pi}{2}\right) & \cos\left(\frac{\pi}{2}\right) & -99 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.9)
	${}^2T_3 = \begin{bmatrix} \cos\theta_3^* & 0 & \sin\theta_3^* & 0 \\ \sin\theta_3^* & 0 & -\cos\theta_3^* & 0 \\ 0 & 1 & 0 & -99 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.10)

**Transformatie van frame  $F_4$  naar  $F_3$  (Pols -> Voorarm):**

	${}^3T_4 = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 * \cos(-\frac{\pi}{2}) & \sin\theta_4 * \sin(-\frac{\pi}{2}) & 0 * \cos\theta_4 \\ \sin\theta_4 & \cos\theta_4 * \cos(-\frac{\pi}{2}) & -\cos\theta_4 * \sin(-\frac{\pi}{2}) & 0 * \sin\theta_4 \\ 0 & \sin(-\frac{\pi}{2}) & \cos(-\frac{\pi}{2}) & -450.75 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.11)
	${}^3T_4 = \begin{bmatrix} \cos\theta_4 & 0 & -\sin\theta_4 & 0 \\ \sin\theta_4 & 0 & \cos\theta_4 & 0 \\ 0 & -1 & 0 & -450.75 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.12)

**Transformatie van frame  $F_5$  naar  $F_4$  (Pols -> Pols):**

	${}^4T_5 = \begin{bmatrix} \cos\theta_5 & -\sin\theta_5 * \cos(\frac{\pi}{2}) & \sin\theta_5 * \sin(\frac{\pi}{2}) & 0 * \cos\theta_5 \\ \sin\theta_5 & \cos\theta_5 * \cos(\frac{\pi}{2}) & -\cos\theta_5 * \sin(\frac{\pi}{2}) & 0 * \sin\theta_5 \\ 0 & \sin(\frac{\pi}{2}) & \cos(\frac{\pi}{2}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.13)
	${}^4T_5 = \begin{bmatrix} \cos\theta_5 & 0 & \sin\theta_5 & 0 \\ \sin\theta_5 & 0 & -\cos\theta_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.14)

**Transformatie van frame  $F_6$  naar  $F_5$  (Pols (TCP zonder offset) -> Pols):**

	${}^5T_6 = \begin{bmatrix} \cos\theta_6^* & -\sin\theta_6^* * \cos(\pi) & \sin\theta_6^* * \sin(\pi) & 0 * \cos\theta_6^* \\ \sin\theta_6^* & \cos\theta_6^* * \cos(\pi) & -\cos\theta_6^* * \sin(\pi) & 0 * \sin\theta_6^* \\ 0 & \sin(\pi) & \cos(\pi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.15)
	${}^5T_6 = \begin{bmatrix} \cos\theta_6^* & \sin\theta_6^* & 0 & 0 \\ \sin\theta_6^* & -\cos\theta_6^* & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.16)

**Transformatie van frame F<sub>7</sub> naar F<sub>6</sub> (TCP -> Pols (TCP zonder offset)):**

	${}^6T_7 = \begin{bmatrix} \cos(0) & -\sin(0) * \cos(0) & \sin(0) * \sin(0) & 0 * \cos(0) \\ \sin(0) & \cos(0) * \cos(0) & -\cos(0) * \sin(0) & 0 * \sin(0) \\ 0 & \sin(0) & \cos(0) & 68 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.17)
	${}^6T_7 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 68 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.18)

## 7.2.5 Eigenlijke berekening van de voorwaartse kinematica

### 7.2.5.1 Algemeen

De voorwaartse kinematica van de volledige robotarm kan bekomen worden door het coördinatenframe van het TCP te transformeren tot aan het coördinaatframe van de basis. Er wordt telkens teruggerekend van F<sub>i</sub> naar F<sub>i-1</sub>, bij matrixvermenigvuldiging betekent dit dat er linksom vermenigvuldigd wordt.

	${}^0T_7 = {}^0T_1 * {}^1T_2 * {}^2T_3 * {}^3T_4 * {}^4T_5 * {}^5T_6 * {}^6T_7$	(7.19)
--	---	--------

Bij de berekening van de voorwaartse kinematica zal er telkens gerekend worden tot aan de pols. Doordat de pols en het TCP enkel verschillen in een offset volgens de Z-as, is de oriëntatie van de pols gelijk aan die van het TCP. Deze werkwijze is in het bijzonder handig omdat de offset van het TCP afhankelijk is van het type actuator en die dus kan verschillen bij bijvoorbeeld het inzetten van een andere toolkop [110].

### 7.2.5.2 Positie en oriëntatie pols

De positie en oriëntatie van de pols kunnen bekomen worden door uit te rekenen tot  ${}^0T_6$ . Hieronder de transformatiematrix van het polsgewricht tot aan het basisframe.

	${}^0T_6 = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$	(7.22)
--	--	--------

Met MatLab [111] worden de verschillende termen in de transformatiematrix symbolisch berekend aan de hand van de DH-parameters en de verschillende transformatiematrices, zoals beschreven in 7.2.3 en 7.2.4.2. De MatLab code voor deze symbolische berekening kan teruggevonden worden in bijlage E-1.

	$n_x = -s_6(c_4s_1 + c_{23}c_1s_4) - c_6(c_5(s_1s_4 - c_{23}c_1c_4) + s_{23}c_1s_5)$	(7.23)
	$n_y = c_6(c_5(c_1s_4 + c_{23}c_4s_1) - s_{23}s_1s_5) + s_6(c_1c_4 - c_{23}s_1s_4)$	(7.24)
	$n_z = s_{23}s_4s_6 - c_6(c_{23}s_5 + s_{23}c_4c_5)$	(7.25)
	$o_x = c_6(c_4s_1 + c_{23}c_1s_4) - s_6(c_5(s_1s_4 - c_{23}c_1c_4) + s_{23}c_1s_5)$	(7.26)

	$o_y = s_6(c_5(c_1s_4 + c_{23}c_4s_1) - s_{23}s_1s_5) - c_6(c_1c_4 - c_{23}s_1s_4)$	(7.27)
	$o_z = -s_6(c_{23}s_5 + s_{23}c_4c_5) - s_{23}c_6s_4$	(7.28)
	$a_x = s_5(s_1s_4 - c_{23}c_1c_4) - s_{23}c_1c_5$	(7.29)
	$a_y = -s_5(c_1s_4 + c_{23}c_4s_1) - s_{23}c_5s_1$	(7.30)
	$a_z = s_{23}c_4s_5 - c_{23}c_5$	(7.31)
	$p_x = 75c_1 + \frac{47}{4}s_1 + 450c_1c_2 - \frac{1803}{4}s_{23}c_1$	(7.32)
	$p_y = 75s_1 - \frac{47}{4}c_1 - \frac{1803}{4}s_{23}s_1 + 450c_2s_1$	(7.33)
	$p_z = 278 - 450s_2 - \frac{1803}{4}c_{23}$	(7.34)

In deze formules zijn afkortingen gebruikt voor de verschillende termen. In tabel 7-5 een legende van de gebruikte symbolen.

**Tabel 7-5 Legende symbolen**

$c_i$	$\cos\theta_i$
$s_i$	$\sin\theta_i$
$c_{ii+1}$	$\cos\theta_i * \cos\theta_{i+1} - \sin\theta_i * \sin\theta_{i+1} = \cos(\theta_i + \theta_{i+1})$
$s_{ii+1}$	$\sin\theta_i * \cos\theta_{i+1} + \cos\theta_i * \sin\theta_{i+1} = \sin(\theta_i + \theta_{i+1})$
$c_3/s_3/c_6/s_6$	$\cos\theta_3^*/\sin\theta_3^*/\cos\theta_6^*/\sin\theta_6^*$

### 7.2.5.3 Positie tool center point

De positie van het TCP kan eenvoudig berekend worden met behulp van de transformatiematrix  ${}^0T_6$  en de offset  $d_7$  uit tabel 7-4. Hieronder een korte uitwerking daarvan:

	$\text{Positie TCP} = {}^0T_6 * \begin{bmatrix} 0 \\ 0 \\ d_7 \\ 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 68 \\ 1 \end{bmatrix}$	(7.35)
	$\text{Positie TCP} = \begin{bmatrix} p_x + 68a_x \\ p_y + 68a_y \\ p_z + 68a_z \\ 1 \end{bmatrix}$	(7.36)

### 7.2.5.4 Bepalen van de Eulerhoeken

Verder kunnen uit de transformatiematrix  ${}^0T_6$  de Eulerhoeken bepaald worden. In de literatuur wordt vaak een variant van deze hoeken besproken, namelijk yaw, pitch en roll. [112].

Deze hoeken zijn gebaseerd op het eerste rotatietheorema van Euler [113]:

*Elk koppel van onafhankelijke orthogonale assenstelsels kan beschreven worden door een sequentie van rotaties (niet meer dan drie) rond de assen van het stelsel, waar geen twee opeenvolgende rotaties rond dezelfde as mogen gebeuren.*

*Euler's eerste rotatietheorema*

Kort samengevat zijn dit de rotatiehoeken van de assen van een assenstelsel  $F_i$  ten opzichte van een ander assenstelsel  $F_{i+1}$ . In de transformatiematrix  ${}^0T_7$  betekent dit dat de verschillende rotaties van het basissenstelsel worden beschouwd tot aan het assenstelsel in het TCP. Respectievelijk in  ${}^0T_6$  is dit van het basissenstelsel tot het assenstelsel in het polsgewricht.

Er wordt uitgegaan van een driedimensionaal rechtshandig vast assenstelsel dat op een bepaalde, vooraf gekende plaats in de ruimte gelegen is. Dit assenstelsel dient als basis voor de rotaties. Vervolgens wordt er een tweede assenstelsel gedefinieerd dat precies ligt zoals het vaste assenstelsel. Dit assenstelsel is mobiel en kan vrij verdraaid worden in de ruimte. Tenslotte wordt arbitrair een derde rechtshandig assenstelsel in de ruimte beschouwd. Hiervan moet de oriëntatie gedefinieerd worden aan de hand van rotaties van het mobiele assenstelsel volgens de assen van het vast assenstelsel.

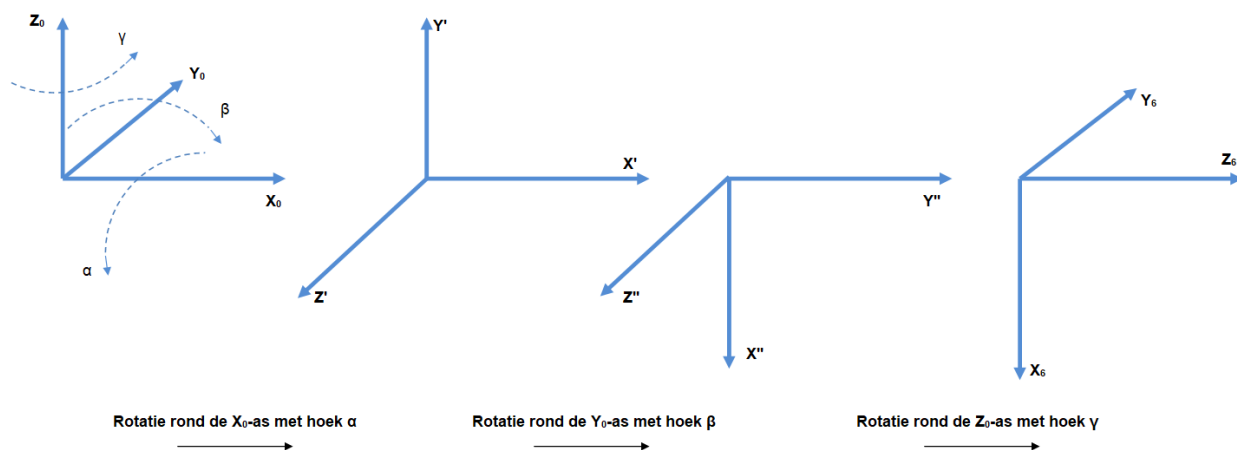
Deze rotaties kunnen op een verscheidenheid van manieren gebeuren. Algemeen wordt gesteld dat volgende stappen doorlopen worden:

1. Rotatie rond de x -, y - of z - as van het vaste frame met hoek  $\alpha$
2. Rotatie rond de x -, y - of z - as van het vaste frame met hoek  $\beta$
3. Rotatie rond de x -, y - of z - as van het vaste frame met hoek  $\gamma$

De volgorde waarmee deze rotaties gebeuren is belangrijk. Voor de AgroBot A.R.M. te beschrijven werd  $x \rightarrow y \rightarrow z$  gekozen als rotatiesequentie.

Als vast frame wordt hier het basisframe gekozen, wat wil zeggen dat de hoek  $\alpha$  correspondeert met een rotatie rond de x-as van het basisframe, de hoek  $\beta$  correspondeert met een rotatie om de y-as en de hoek  $\gamma$  met een rotatie om de z-as.

In figuur 7-15 zijn de verschillende stappen te zien voor de rotaties om het basisassenstelsel tot het assenstelsel in de pols van de AgroBot A.R.M., zoals te zien op het kinematisch diagram. De beschrijving hieronder is de transformatie van  $F_0$  naar  $F_6$ . Het is erg belangrijk om te begrijpen dat alle rotaties gebeuren rond de oorspronkelijke assen van  $F_0$ . In onderstaand geval is de eerste stap een rotatie  $\alpha$  rond de  $X_0$ -as van  $90^\circ$  waardoor een nieuw (mobiel) assenstelsel bekomen wordt. Dit nieuwe assenstelsel wordt in stap twee dan gerooteerd over de hoek  $\beta$  van  $90^\circ$  rond de  $Y_0$ -as. Dit is hetzelfde als roteren rond de Z'-as met dezelfde hoek  $\beta$ . Hieruit wordt weer een nieuw assenstelsel bekomen dat dan met hoek  $\gamma$  van  $90^\circ$  gerooteerd wordt rond de  $Z_0$ -as. Dit is hetzelfde als roteren met  $\gamma$  rond de X"-as. Onderstaande verdraaiingen zijn positief volgens de rechterhandregel.



**Figuur 7-15 Rotaties van het basisframe tot aan de pols met de configuratie zoals voorgesteld in het kinematisch diagram**

Vanuit de verschillende rotaties kunnen de basisrotatiematrices opgesteld worden [114]:

	$R_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$	(7.37)
	$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$	(7.38)
	$R_z(\gamma) = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$	(7.39)

Deze rotatiematrices stellen telkens een rotatie in één bepaalde richting voor en zijn onafhankelijk van elkaar. De volgorde van de rotaties bepaalt de totale rotatie, daarom dient dus steeds onthouden te worden om de rotaties volgens de juiste rotatiesequentie uit te voeren.

De totale rotatiematrix die bekomen wordt, is de vermenigvuldiging van de basisrotatiematrices in de omgekeerde volgorde van rotatie rond de assen. Dit omdat er voor gekozen is om de rotaties te beschrijven ten opzichte van het vast basisassenstelsel. Via MatLab wordt dit symbolisch uitgewerkt in bijlage E-3.

	$R(\gamma, \beta, \alpha) = R_z(\gamma)R_y(\beta)R_x(\alpha)$	(7.40)
	$R(\gamma, \beta, \alpha) = \begin{bmatrix} c_\beta c_\gamma & c_\gamma s_\alpha s_\beta - c_\alpha s_\gamma & s_\alpha s_\gamma + c_\alpha c_\gamma s_\beta \\ c_\beta s_\gamma & c_\alpha c_\gamma + s_\alpha s_\beta s_\gamma & c_\alpha s_\beta s_\gamma - c_\gamma s_\alpha \\ -s_\beta & c_\beta s_\alpha & c_\alpha c_\beta \end{bmatrix}$	(7.41)

In deze formules zijn afkortingen gebruikt voor de verschillende termen. In tabel 7-6 een legende van de gebruikte symbolen.



**Tabel 7-6 Legende symbolen**

$c_\alpha/c_\beta/c_\gamma$	$\cos\alpha/\cos\beta/\cos\gamma$
$s_\alpha/s_\beta/s_\gamma$	$\sin\alpha/\sin\beta/\sin\gamma$

Deze rotatiematrix zal vergeleken worden met de rotatiematrix uit transformatiematrix  ${}^0T_6$  (of  ${}^0T_7$ ). Via onderstaande formules kunnen de Eulerhoeken bepaald worden [115]:

	$\alpha = \text{atan2}\left(\frac{n_z}{\cos\beta}, \frac{o_z}{\cos\beta}\right)$	(7.42)
	$\beta = \text{atan2}\left(\sqrt{a_x^2 + a_y^2}, -a_z\right)$	(7.43)
	$\gamma = \text{atan2}\left(-\frac{a_x}{\cos\beta}, \frac{a_y}{\cos\beta}\right)$	(7.44)

**Opmerking 1:**

Bij de atan2 functie is het argument soms in een verschillende volgorde naargelang de gebruikte software. De waarden hier ingegeven zijn geschikt voor het berekenen met behulp van MatLab.

**Opmerking 2:**

Om de hoeken  $\alpha$  en  $\gamma$  te kunnen berekenen geldt:

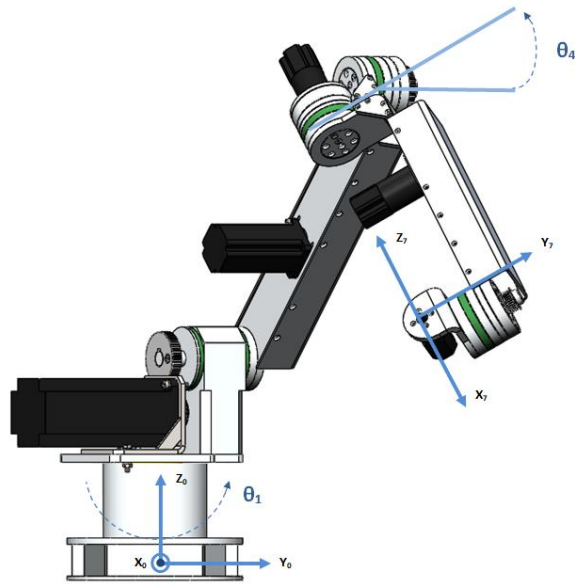
	$\cos\beta \neq 0 \rightarrow \beta \neq \pm \frac{\pi}{2}$	(7.45)
--	---	--------

Uit de transformatiematrix kan alle nodige informatie gehaald worden over de robot. Zowel de positie als de oriëntatie is gekend. In de volgende paragraaf wordt dit getest aan de hand van een numeriek voorbeeld.

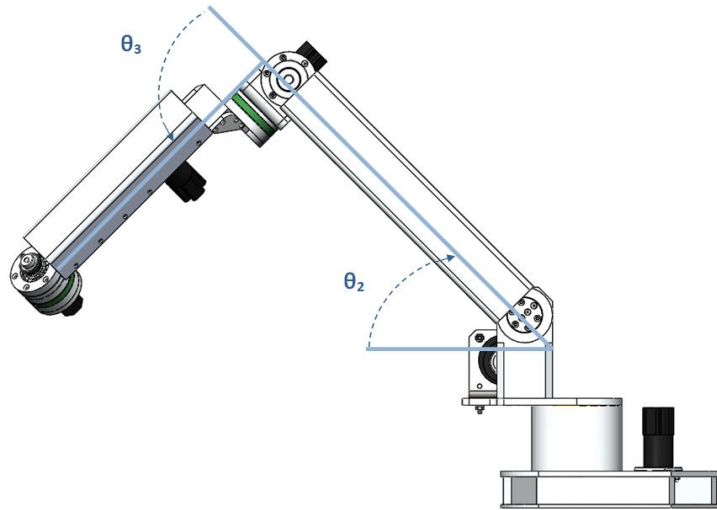
**7.2.6 Numeriek voorbeeld voorwaartse kinematica**

**7.2.6.1 Gegevens**

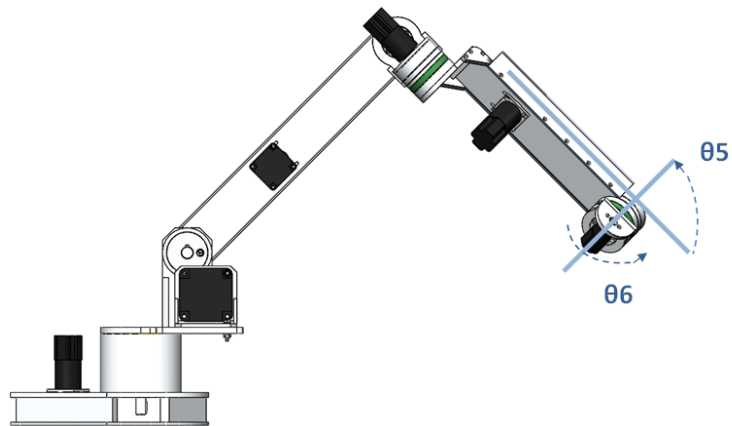
Figuur 7-16 tot en met 7-18 tonen de verschillende rotatiehoeken  $\theta_i$ .



**Figuur 7-16 Vooraanzicht A.R.M.**



**Figuur 7-17 Linkerzijaanzicht A.R.M.**



**Figuur 7-18 Rechterzijaanzicht A.R.M.**

In deze figuren zijn de hoeken telkens getekend volgens hun teken in tabel 7-7. Als er een minteken is, dan is de getekende hoek tegengesteld aan de positieve richting zoals weergegeven in het kinematisch diagram. In figuur 7-16 zijn de assenstelsels van de basisplaat en het TCP toegevoegd.

**Tabel 7-7 DH-parameters met getalwaarden**

Frame #	r (mm)	d (mm)	$\alpha$ (rad)	$\theta$ (rad)
0-1	75	278	$-\pi/2$	$\pi/6$
1-2	480	87.25	0	$-\pi/4$
2-3	0	-99	$\pi/2$	$\pi/2 - \pi/2$
3-4	0	-450.75	$-\pi/2$	$-\pi/4$
4-5	0	0	$\pi/2$	$-\pi/2$
5-6	0	0	$\pi$	$-\pi/2 + \pi$
6-7	0	68	0	0

### 7.2.6.2 Doel

Het doel van deze berekening is het bekomen van de XYZ-positie van de het TCP ( $X_7, Y_7, Z_7$ ) ten opzichte van de basis van de robotarm ( $X_0, Y_0, Z_0$ ), alsook de oriëntatie van de twee frames ten opzichte van elkaar.

### 7.2.6.3 Berekening

De berekening wordt uitgevoerd met behulp van MatLab (Bijlage E-2), volgens de theorie beschreven in 7.2.5. De DH-parameters worden met hun getalwaarden ingevuld en de positie en oriëntatie wordt bekomen.

Voor de parameters ingegeven in tabel 7-7 worden volgende waarden bekomen:

	$\text{Positie TCP} = {}^0T_6 * \begin{bmatrix} 0 \\ 0 \\ 68 \\ 1 \end{bmatrix} =$ $\begin{bmatrix} 0.0795 & -0.6124 & 0.7866 & 640.7925 \\ 0.8624 & -0.3536 & -0.3624 & 356.3940 \\ 0.5000 & 0.7071 & 0.5000 & 298.6829 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 68 \\ 1 \end{bmatrix}$	(7.46)
	$\text{Positie TCP} = \begin{bmatrix} 68 * (0.7866) + 640.7925 \\ 68 * (-0.3624) + 356.3940 \\ 68 * (0.5000) + 298.6829 \\ 1 \end{bmatrix} = \begin{bmatrix} 694.2790 \\ 331.7527 \\ 332.6829 \\ 1 \end{bmatrix}$	(7.47)
	$\beta = \text{atan2} \left( \sqrt{(-0.7866)^2 + (0.3624)^2}, -(-0.5000) \right) \xrightarrow{\text{degrees}} 120^\circ$	(7.48)
	$\alpha = \text{atan2} \left( \frac{0.5000}{\cos\left(\frac{\pi}{3}\right)}, \frac{0.7071}{\cos\left(\frac{\pi}{3}\right)} \right) \xrightarrow{\text{degrees}} 35.264^\circ$	(7.49)

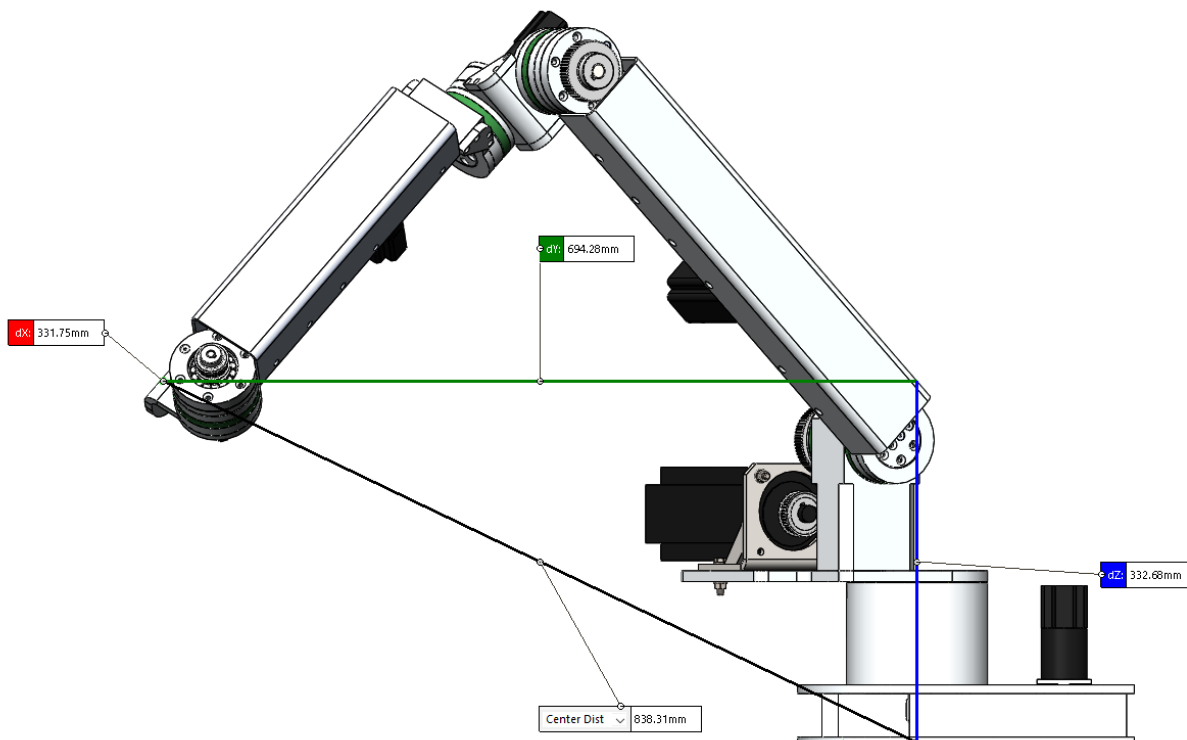
	$\gamma = \text{atan2}\left(\frac{-0.7866}{\cos\left(\frac{\pi}{3}\right)}, \frac{0.3624}{\cos\left(\frac{\pi}{3}\right)}\right) \xrightarrow{\text{degrees}} -114.736^\circ$	(7.50)
--	---	--------

**Tabel 7-8 Samenvatting: positie en oriëntatie van het TCP**

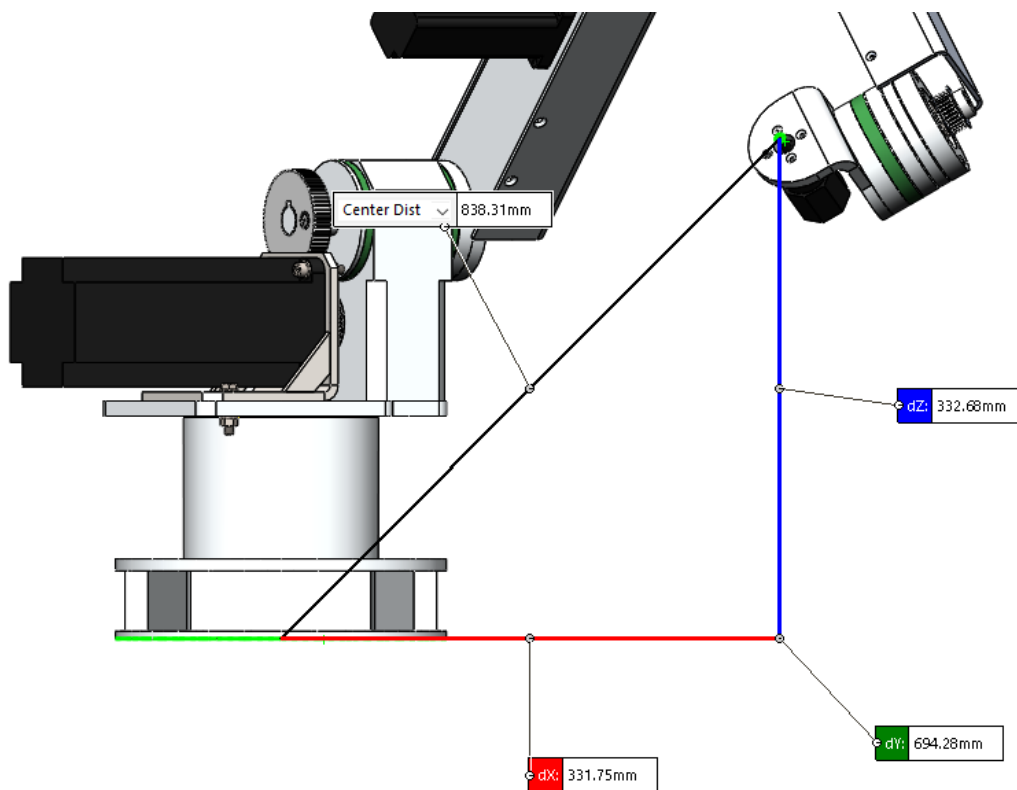
	Berekende waarde (mm)
<b>X</b>	694.2790
<b>Y</b>	331.7527
<b>Z</b>	332.6829
Berekende waarde (°)	
<b>α</b>	35.264
<b>β</b>	120
<b>γ</b>	-114.736

#### 7.2.6.4 Controle

Via de 'Measure' - tool in Solidworks kan de werkelijke positie van de eeffector bepaald worden. Aan de hand van deze waarden kan de voorwaartse kinematica eenvoudig gecontroleerd worden. De gemeten waarden worden hieronder, in figuur 7-19 en 7-20, weergegeven en daarna opgelijst in tabel 7-9.



**Figuur 7-19 Controlefiguur dY - dZ**



**Figuur 7-20 Controlefiguur dX - dZ**

**Opmerking:**

De waarden dX en dY in de figuren liggen anders dan in het kinematisch diagram. Volgens de gebruikte conventie geldt:  $dX = 694.20\text{mm}$  en  $dY = 331.75\text{mm}$ .

**Tabel 7-9 Controle berekende waarden**

	<b>Berekende waarde (mm)</b>	<b>Gemeten waarde (mm)</b>
<b>X</b>	694.2790	694.28
<b>Y</b>	331.7527	331.75
<b>Z</b>	332.6829	332.68

Deze waarden komen overeen met de berekende waarden uit MatLab. Omdat de Eulerhoeken in dit numerieke voorbeeld erg moeilijk manueel gecontroleerd kunnen worden, zal in 7.2.6.5 een extra voorbeeld uitgewerkt worden.

**7.2.6.5 Extra: controle Eulerhoeken**

Omdat de hoeken  $\theta_4$  en  $\theta_6$  ervoor zorgen dat er manueel moeilijk kan nagerekend worden, worden deze op nul gesteld.  $\theta_5$  wordt voor de eenvoud ook op nul gesteld.

De andere hoeken die in MatLab worden ingegeven, zijn te vinden in tabel 7-10.

**Tabel 7-10 Invulwaarden controle Eulerhoeken**

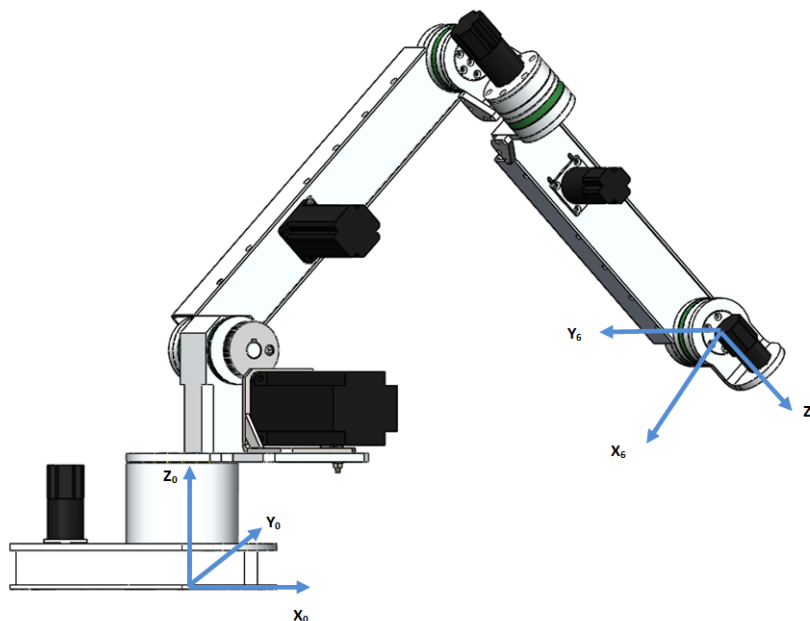
	Hoeken (°)
$\theta_1$	30
$\theta_2$	-45
$\theta_3$	90 ( - 90)
$\theta_4$	0
$\theta_5$	0
$\theta_6$	0 ( + 180)

Uit MatLab komen volgende waarden voor de Eulerhoeken:

**Tabel 7-11 Uitkomst MatLab**

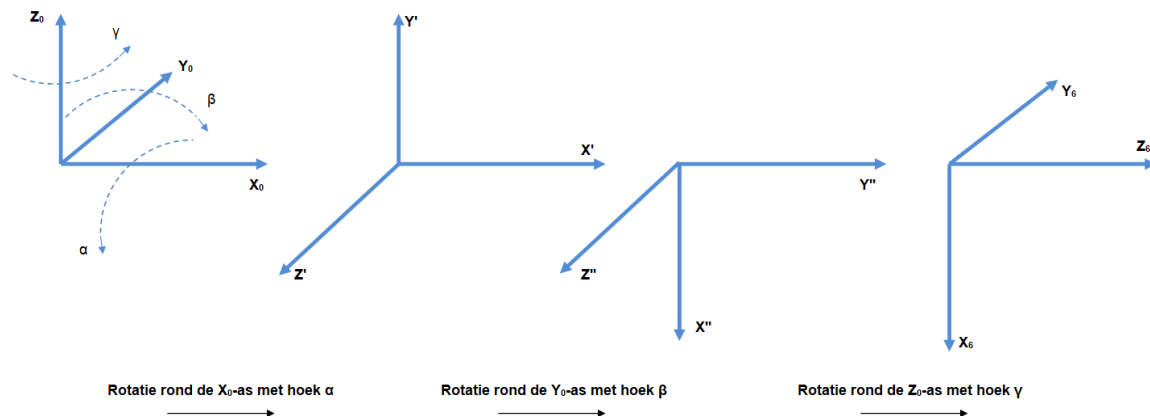
	Hoeken (°)
$\alpha$	-90
$\beta$	45
$\gamma$	-60

Dit komt overeen met volgende oriëntatie:



**Figuur 7-21 Controlefiguur Eulerhoeken**

Door te redeneren op basis van de drie onafhankelijke rotaties wordt nagegaan als de berekende hoeken correct zijn. Eerst wordt de redenering opgebouwd voor de rotaties van het basissenstelsel tot het assenstelsel in het polsgewricht, zoals getekend in het kinematisch diagram. Hieronder dezelfde figuur als figuur 7-15 ter verduidelijking:



**Figuur 7-22 Rotaties van het basisframe tot aan tool center point met de configuratie zoals voorgesteld in het kinematisch diagram**

1. Rotatie  $\alpha$  om de  $X_0$ -as met  $90^\circ$
2. Rotatie  $\beta$  om de  $Y_0$ -as met  $90^\circ$
3. Rotatie  $\gamma$  om de  $Z_0$ -as met  $90^\circ$

Na deze drie rotaties wordt het assenstelsel uit het polsgewricht bekomen.

Analoog kan deze sequentie opgebouwd worden voor dit voorbeeld:

1. Rotatie  $\alpha$  om de  $X_0$ -as met  $-90^\circ$  in plaats van  $90^\circ$
2. Rotatie  $\beta$  om de  $Y_0$ -as met  $45^\circ$  is in plaats van  $90^\circ$
3. Rotatie  $\gamma$  om de  $Z_0$ -as met  $-60^\circ$  is in plaats  $90^\circ$

Als deze assenstelsel in de ruimte uitgetekend worden, wordt hetzelfde assenstelsel in de pols bekomen als weergegeven op figuur 7-21.

Uit dit voorbeeld blijkt dat ook de Eulerhoeken correct zijn. De voorwaartse kinematica is numeriek bevestigd. Vanaf deze basis kan de inverse kinematica in de volgende paragraaf uitgewerkt worden.

## 7.3 Inverse Kinematica:

Er zijn verschillende manieren om de inverse kinematica van een robotarm op te lossen. De methode die hier gebruikt wordt, is gebaseerd op kinematische ontkoppeling. Dit is de splitsing van de inverse kinematica in twee deelproblemen, namelijk inverse positie- en inverse oriëntatiekinematica. Aan de hand van deze techniek kunnen de hoeken van alle armdelen bepaald worden als een gewenste positie en oriëntatie van de eendeffector gegeven is [116].

### 7.3.1 Kinematische ontkoppeling

Bij kinematische ontkoppeling wordt de arm opgesplitst ter hoogte van de pols, zodat er drie rotatievrijheden zijn in de pols en drie in het armdeel ervoor, verder de basisarm genoemd. De basisarm zal gebruikt worden voor het oplossen van de inverse positiekinematica, terwijl de pols gebruikt zal worden voor de inverse oriëntatiekinematica. Bij de AgroBot A.R.M. kan deze opsplitsing voor verwarring zorgen doordat de rotatie in  $F_4$  gelegen is voor de pols. Door die rotatie van de voorarm zal echter ook de pols meedraaien, waardoor de rotatie ook beschouwd mag worden in de pols.

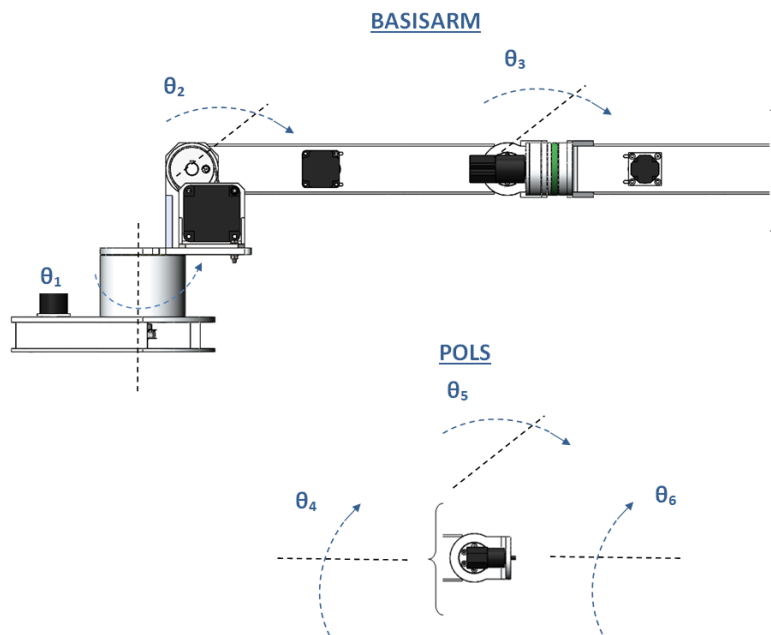
Deze methode kan en mag enkel gebruikt worden als de robotarm voldoet aan de Pieper voorwaarde [117]:

Om kinematische ontkoppeling toe te passen dienen drie opeenvolgende roterende assen te snijden in één punt in de ruimte.

*Piepervoorwaarde*

Aan deze voorwaarde is bij de AgroBot A.R.M. in de pols voldaan, doordat er daar drie assen samenkomen met als gemeenschappelijk snijpunt het polsgewricht.

In figuur 7-23 zijn de verschillende rotaties aangegeven op de gesplitste robotarm.



**Figuur 7-23 Kinematische ontkoppeling - rotaties per deel**

Door het invoeren van de ontkoppeling kunnen positie en oriëntatie als onafhankelijk van elkaar beschouwd worden. De berekening van de inverse kinematica van de twee delen zal op verschillende wijze gebeuren. De inverse positiekinematica zal goniometrisch opgelost worden aan de hand van de basisarm, terwijl de inverse oriëntatiekinematica analytisch zal opgelost worden aan de hand van de pols.

In toepassingen is het telkens gewenst om de eindeffector te sturen, en intuïtief zou de berekening uitgevoerd worden ter hoogte van het TCP. Zoals eerder beschreven is het TCP in feite een offset in de richting X,Y en Z vanuit de pols. Deze offset zal bepaald worden door parameter  $d_7$  van de DH-parameters en door de oriëntatie van de pols. Daarom dat er telkens zal gerekend worden naar de pols.

## 7.3.2 Grenswaarden rotatiehoeken & definiëring kwadranten

### 7.3.2.1 Grenswaarden

De berekening van de inverse kinematica is afhankelijk van de kwadranten waarin de armdelen zich bevinden. Door de keuze van de oriëntatie van het kinematisch diagram, kunnen de armdelen verschillende kwadranten doorlopen. Om deze kwadranten op voorhand te kennen en de latere berekeningen er op te kunnen afstemmen, worden nu



grenswaarden ingevoerd voor de hoeken. Deze theoretische grenswaarden zijn te vinden in tabel 7-12. De grenswaarden zijn reeds in functie van het mechanisch ontwerp, zonder dat er ergens obstructie is. Omdat de robotarm door de bekabeling of diverse andere factoren dit opgelegde bereik niet volledig kan afgaan, worden in tabel 7-12 ook de praktische grenswaarden ingevuld. Deze zijn belangrijk voor de latere programmering van de robot.

**Tabel 7-12 Grenswaarden hoeken**

Hoek	Theoretisch	Praktisch
$\theta_1$	$[-90^\circ; +90^\circ]$	$[-60^\circ; +60^\circ]$
$\theta_2$	$[-90^\circ; +90^\circ]$	$[-90^\circ; +70^\circ]$
$\theta_3$	$[0^\circ; +135^\circ]$	$[0^\circ; +125^\circ]$
$\theta_4$	$[-180^\circ; +180^\circ]$	$[-90^\circ; +90^\circ]$
$\theta_5$	$[-135^\circ; +135^\circ]$	$[-135^\circ; +135^\circ]$
$\theta_6$	$[-90^\circ; +90^\circ]$	$[-90^\circ; +90^\circ]$

**Opmerking:**

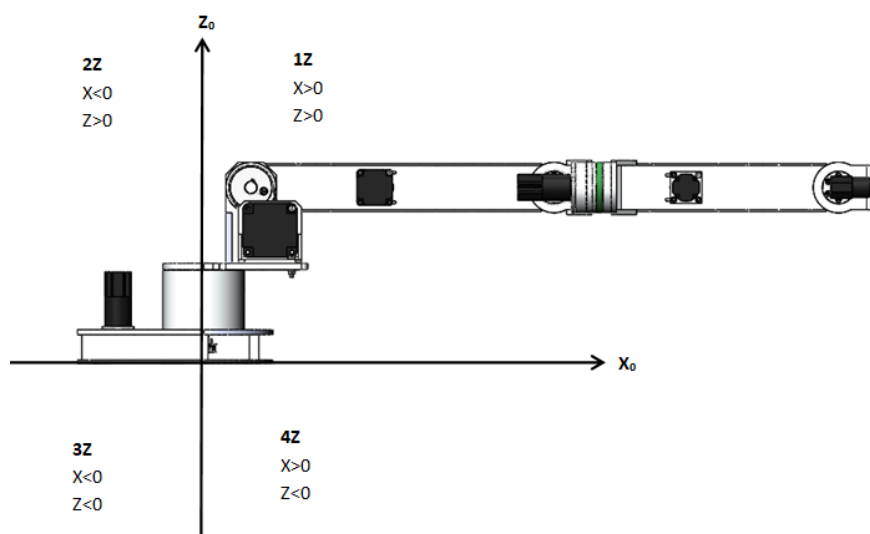
Om telkens een elbow-up configuratie te bekommen wordt  $\theta_3$  positief gekozen (de configuratie zoals in voorbeeld 7.2.6 is elbow-up). Indien later een elbow-down configuratie bekommen wil worden, zal hiermee rekening moeten gehouden worden bij het berekenen van  $\theta_3$ .

**7.3.2.2 Kwadranten**

De coördinaten bepalen in welk kwadrant de pols zich bevindt. Hieronder worden de twee kwadrantvormen voorgesteld. Deze zijn nuttig bij latere berekeningen en zullen ook gebruikt worden bij programmering van de robotarm.

**7.3.2.2.1 Kwadranten volgens het zijaanzicht**

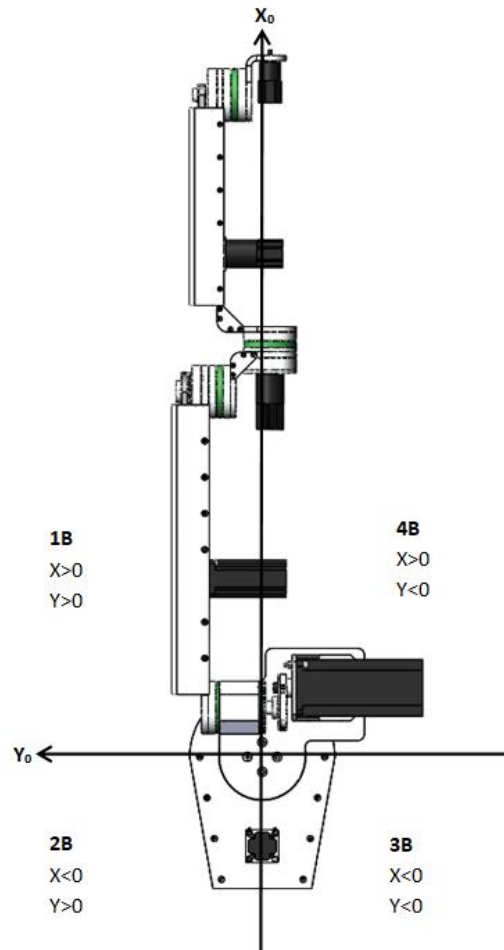
Deze kwadranten worden beschreven door het kwadrantnummer met toevoeging van de letter Z. De verschillende kwadranten zijn te zien in figuur 7-24.



**Figuur 7-24 Kwadranten volgens het zijaanzicht**

### 7.3.2.2.2 Kwadranten volgens het bovenaanzicht

Deze kwadranten worden beschreven door het kwadrantnummer met toevoeging van de letter B. De verschillende kwadranten zijn te zien in figuur 7-25.



**Figuur 7-25 Kwadranten volgens het bovenaanzicht**

### 7.3.3 Beginparameters

Bij aanvang van de berekeningen moet eerst een gewenste positie en oriëntatie van de eeffector gedefinieerd worden.

De positie bestaat uit de coördinaten van het laatste frame ( $F_7$ ). Vanaf dit punt wordt er via de opgegeven oriëntatie teruggerekend naar de pols. Hiermee wordt vervolgens de inverse positiëkinematica berekend.

De inverse oriëntatiekinematica wordt berekend aan de hand van een gewenste oriëntatie en de oriëntatie van de basisarm die volgt uit de inverse positiëkinematica.

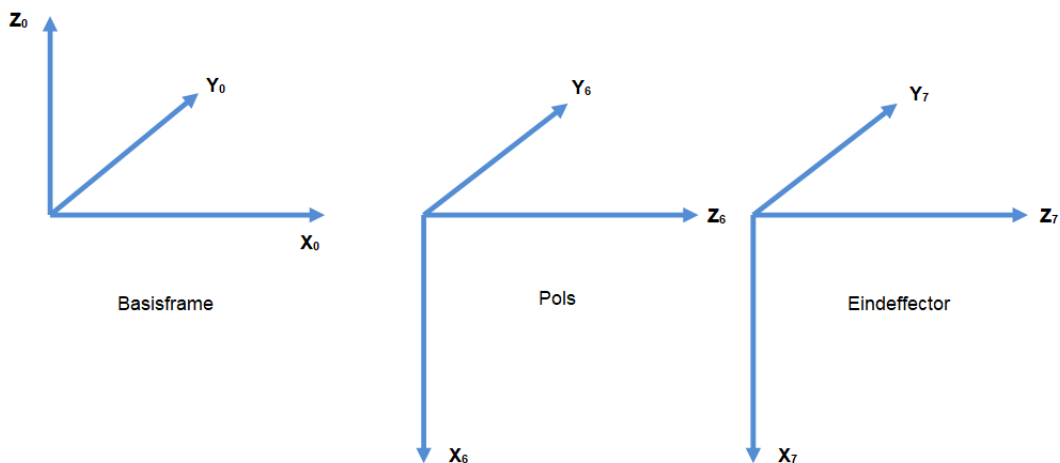
De oriëntatie wordt ingegeven in matrixvorm en stelt de oriëntatie van de eeffector voor ten opzichte van de basis. Aangezien het coördinatenframe van de eeffector enkel verschilt met dat van de pols met een offset volgens de Z-richting van de pols, is de oriëntatie van de pols gelijk aan die van de eeffector.

Omdat dit vrij abstract is om te interpreteren, wordt hieronder een kort voorbeeld neergeschreven.

### 7.3.3.1 Oriëntatie definiëren

Stel dat het gewenst is dat de eideffector horizontaal ligt in het verlengde van de robotarm, dan kan het coördinatenframe van de eideffector beschouwd worden zoals in het kinematisch diagram.

In figuur 7-26 worden zowel het basiscoördinatenframe als het coördinatenframe in de pols en eideffector weergegeven. Bij verdere berekeningen zal het frame van de pols gebruikt worden.



**Figuur 7-26 Oriëntatie van de basis - pols - eideffector**

Het basisframe wordt beschouwd als vast frame waarbij de oriëntatie van de assen kan voorgesteld worden als een eenheidsmatrix:

	$\text{Oriëntatie basisframe} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	(7.51)
--	--	--------

Vervolgens kan de oriëntatiematrix van de pols op twee manieren bekomen worden. In dit geval kan deze intuïtief bepaald worden, doordat gemakkelijk te zien is dat  $X_6$ -as ligt volgens de negatieve  $Z_0$ -as, de  $Y_6$ -as volgens de positieve  $Y_0$ -as en tot slot de  $Z_6$ -as volgens de positieve  $X_0$ -as. Volgende oriëntatiematrix wordt bekomen:

	$\text{Oriëntatie pols} = R = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$	(7.52)
--	---	--------

Bij een andere oriëntatie is het intuïtief bepalen van de oriëntatiematrix van de pols vaak niet mogelijk. Hierbij kan er gebruik gemaakt worden van de eulerhoeken. In dit specifieke geval geldt:  $\alpha=\beta=\gamma=90^\circ$ . Uit vergelijking 7.41 kan de oriëntatie van de pols gevonden worden.

	$R(\gamma, \beta, \alpha) = \begin{bmatrix} c_\beta c_\gamma & c_\gamma s_\alpha s_\beta - c_\alpha s_\gamma & s_\alpha s_\gamma + c_\alpha c_\gamma s_\beta \\ c_\beta s_\gamma & c_\alpha c_\gamma + s_\alpha s_\beta s_\gamma & c_\alpha s_\beta s_\gamma - c_\gamma s_\alpha \\ -s_\beta & c_\beta s_\alpha & c_\alpha c_\beta \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad (7.53)$	
--	---	--

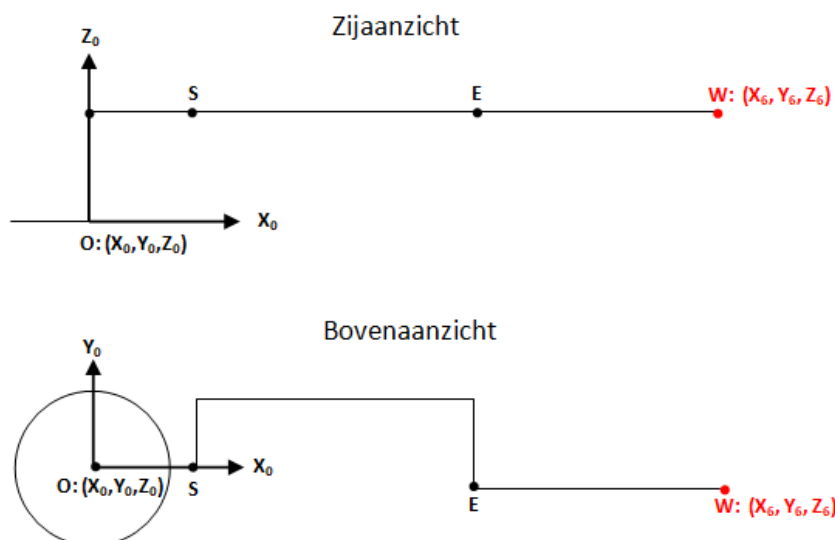
### 7.3.4 Omrekenen van Tool Center Point tot pols

Analoog aan vergelijking 7.35 kan de positie van de pols bekomen worden uit de positie van het TCP, als de oriëntatie van het TCP gekend is.

	$\text{Positie TCP} = \begin{bmatrix} X_{\text{TCP}} \\ Y_{\text{TCP}} \\ Z_{\text{TCP}} \\ 1 \end{bmatrix} = {}^0T_6 * \begin{bmatrix} 0 \\ 0 \\ 68 \\ 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 68 \\ 1 \end{bmatrix} \quad (7.54)$	
	$\text{Positie pols} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} X_{\text{TCP}} - 68 * (a_x) \\ Y_{\text{TCP}} - 68 * (a_y) \\ Z_{\text{TCP}} - 68 * (a_z) \\ 1 \end{bmatrix} \quad (7.55)$	

### 7.3.5 Inverse positiekinematica

Voor de berekening van de inverse positiekinematica wordt er gebruik gemaakt van het vereenvoudigd kinematisch diagram van de basisarm zoals te zien in figuur 7-27. Elk scharnier wordt voorgesteld door een zwart punt (O = oorsprong, S = schouder, E = elleboog), terwijl de pols (W = wrist) wordt voorgesteld door een rood punt. Vanuit dit diagram zal een bepaalde positie omgevormd worden tot een hoek. Bij de berekening worden steeds de coördinaten van de pols gegeven, zodat hieruit de eerste drie hoeken kunnen bepaald worden.



Figuur 7-27 Vereenvoudigd kinematisch diagram basisarm

### 7.3.5.1 Basisrotatie ( $\theta_1$ )

#### 7.3.5.1.1 Figuur met aantekeningen

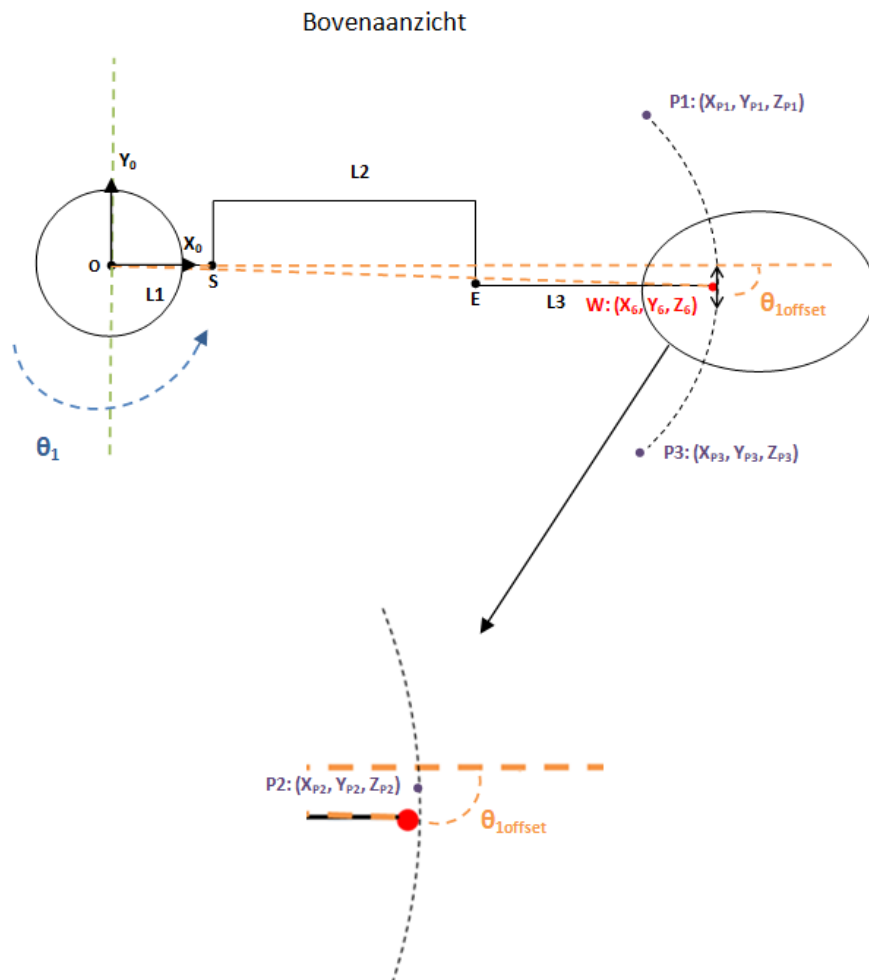
In figuur 7-28 zijn met behulp van de groene stippellijn de uiterste theoretische waarden van  $\theta_1$  aangegeven, zoals beschreven in tabel 7-12.

Voor de berekening is het belangrijk op te merken dat de pols ( $X_6, Y_6, Z_6$ ) niet op dezelfde lijn ligt als de X- as van de basis. Daarom wordt de hoek  $\theta_{1offset}$  gedefinieerd als de hoek tussen de oranje stippellijnen.

	$\theta_{1offset} = abs\left(\text{atan}\left(\frac{-11.75}{\sqrt{X^2 + Y^2}}\right)\right)$	(7.56)
--	--	--------

Bij deze formule zijn X en Y de coördinaten van de pols bij een bepaalde configuratie. In de rustpositie zullen dit  $X_6$  en  $Y_6$  zijn.

De offset zal opgeteld worden bij de hoek  $\theta_1$  als de pols zich in kwadrant 1B of 4B bevindt. In kwadranten 2B en 3B wordt de offset bij  $\theta_1$  in mindering gebracht.



**Figuur 7-28** Figuur met alle aantekeningen voor de berekening van  $\theta_1$

#### 7.3.5.1.2 Positieve rotatie ( $> \theta_{1offset}$ )

Stel dat de robotarm linksom (dus positief) roteert vanuit de beginpositie naar een punt  $P_1$  in de ruimte, zoals te zien op figuur 7-28, dan wordt  $\theta_1$  als volgt berekend:

	$\theta_1 = \text{atan}\left(\frac{Y_{P1}}{X_{P1}}\right) \pm \theta_{1offset}$	(7.57)
--	---	--------

#### 7.3.5.1.3 Positieve rotatie ( $\leq \theta_{1offset}$ )

Als de robotarm naar punt  $P_2$  roteert vanuit de beginpositie, dan is de positieve hoek kleiner dan  $\theta_{1offset}$ , ook hier geldt dezelfde formule. Voor het punt  $P_2$  op figuur 7-28 geldt:

	$\theta_1 = \text{atan}\left(\frac{Y_{P2}}{X_{P2}}\right) \pm \theta_{1offset}$	(7.58)
--	---	--------

#### 7.3.5.1.4 Negatieve rotatie

Stel dat de robotarm rechtsom (dus negatief) roteert vanuit de beginpositie naar een punt  $P_3$  in de ruimte, zoals te zien op figuur 7-28, dan wordt  $\theta_1$  als volgt berekend:

	$\theta_1 = \text{atan}\left(\frac{Y_{P3}}{X_{P3}}\right) \pm \theta_{1offset}$	(7.59)
--	---	--------

#### 7.3.5.1.5 Conclusie

Bij het zoeken van  $\theta_1$  dient er geen rekening gehouden te worden met de draairichting, daar het teken van de hoek automatisch volgt door het invullen van de coördinaten ten opzichte van de oorsprong. Het optellen of aftrekken van de offset is enkel afhankelijk van de kwadranten waarin de pols zich bevindt.

#### Opmerking:

Zowel bij de negatieve als de positieve rotatie in kwadranten 1B of 4B wordt opgeteld met  $\theta_{1offset}$ , ongeacht de richting van de rotatie. Dit lijkt onlogisch op het eerste zicht, daarom hieronder een korte toelichting.

Volgens de gekozen conventie zal het punt  $P_1$  een positief X- en Y- coördinaat hebben, waardoor de atan functie positief uitkomt. De afgelegde hoek is echter groter, dus dient opgeteld te worden met  $\theta_{1offset}$ . Het geheel komt positief uit.

Als punt  $P_3$  beschouwd wordt, dan wordt duidelijk dat hier ook een positief X- coördinaat is, maar dat het Y- coördinaat negatief is. De atan functie komt hier negatief uit. De afgelegde hoek is echter kleiner, omdat er reeds een offset is. Daarom wordt  $\theta_{1offset}$  erbij opgeteld, er wordt een kleinere negatieve hoek bekomen.

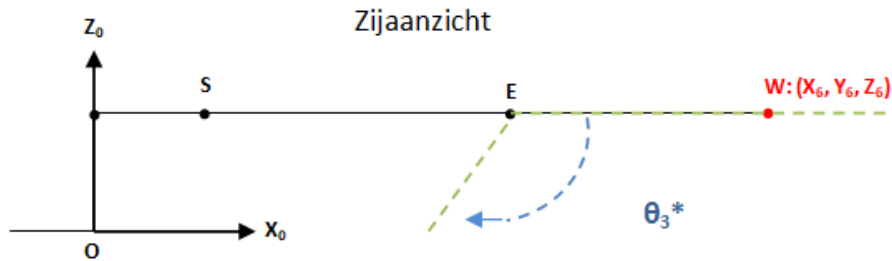
#### 7.3.5.2 Elleboogrotatie ( $\theta_3$ )

Voordat de schouderrotatie kan berekend worden, is het handiger om eerst de elleboogrotatie te beschouwen. Het berekenen van  $\theta_3$  gebeurt aan de hand van de positie van de pols met zowel  $\theta_2$  en  $\theta_3$  verschillend van nul. Een aantal standen zijn mogelijk, de berekening zal echter telkens gelijk zijn.

Belangrijk om op te merken is dat  $\theta_3$  steeds positief moet zijn, omdat telkens een elbow-up configuratie gewenst is.

### 7.3.5.2.1 Bereik elleboogrotatie

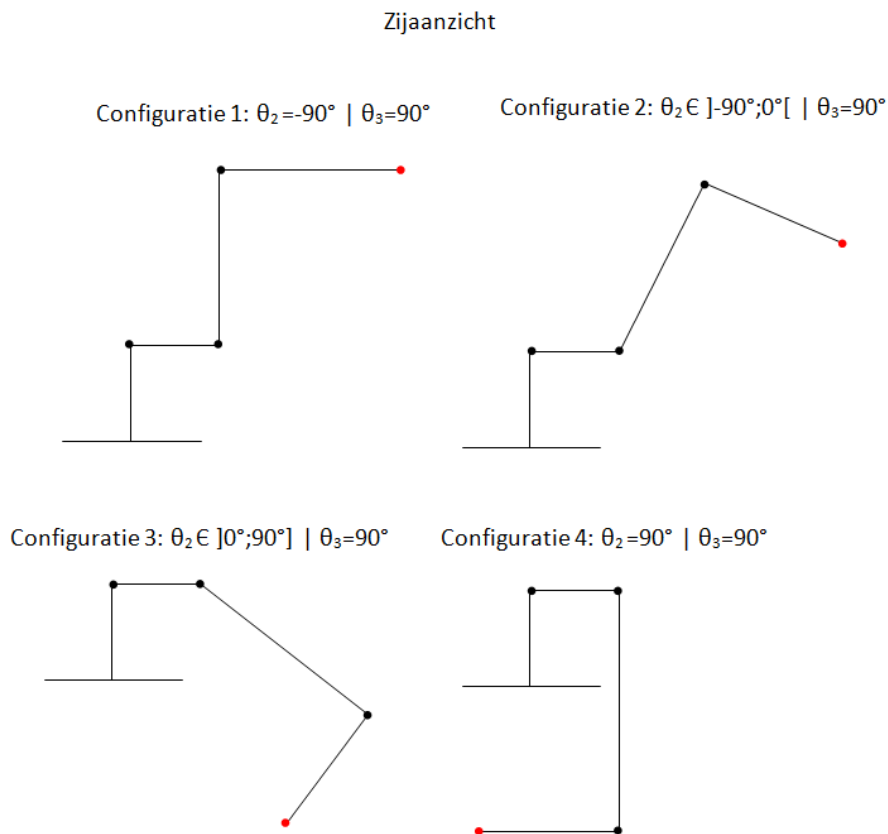
In figuur 7-29 kan het theoretische bereik van de elleboogrotatie waargenomen worden.



**Figuur 7-29 Bereik elleboogrotatie**

### 7.3.5.2.2 Configuraties

In figuur 7-30 zijn vier verschillende configuraties te zien. In tegenstelling tot wat intuïtief zou verondersteld worden, kunnen deze configuraties op dezelfde wijze opgelost worden. Belangrijk is dat de elbow-up configuratie ten allen tijde behouden blijft, dit is namelijk de voorwaarde opdat de werkwijze in alle vier de configuraties gelijk zou zijn.



**Figuur 7-30 Verschillende configuraties met variërende  $\theta_2$  en  $\theta_3 = 90^\circ$**

### 7.3.5.2.3 Oplossing voor $\theta_3$ aan de hand van de tweede configuratie

In figuur 7-31 wordt de gebruikte configuratie voor de berekening afgebeeld.

De eerste stap bestaat erin om de afstand tussen de wrist  $W$  en het scharnierpunt  $S$  van de schouder te berekenen. In dit geval is  $\theta_1$  verschillend van nul en wordt het punt  $P_1$  (=positie wrist bij die oriëntatie) genomen voor de berekening:

	$R = \left( \sqrt{X_{P_1}^2 + Y_{P_1}^2} \right) \pm 75 \text{ (* zie opmerking)}$	(7.60)
	$D = \sqrt{R^2 + \Delta Z^2} \text{ met } \Delta Z = Z_{P_1} - Z_S$	(7.61)

#### Opmerking:

Bij de berekening van  $R$  dient 75 opgeteld te worden als de pols zich in het kwadrant 3Z bevindt. In de kwadranten 1Z, 4Z moet 75 in mindering gebracht worden. De kwadranten volgens het bovenaanzicht zorgen voor geen bijzondere gevallen door de gekozen begrenzing van  $\theta_1$ .

Vervolgens wordt de cosinusregel toegepast in de driehoek SEP1:

	$D^2 = L2^2 + L3^2 - 2 * L2 * L3 * \cos(180^\circ - \theta_3)$ Met $\cos(180^\circ - \theta_3) = -\cos(\theta_3)$	(7.62)
--	--	--------

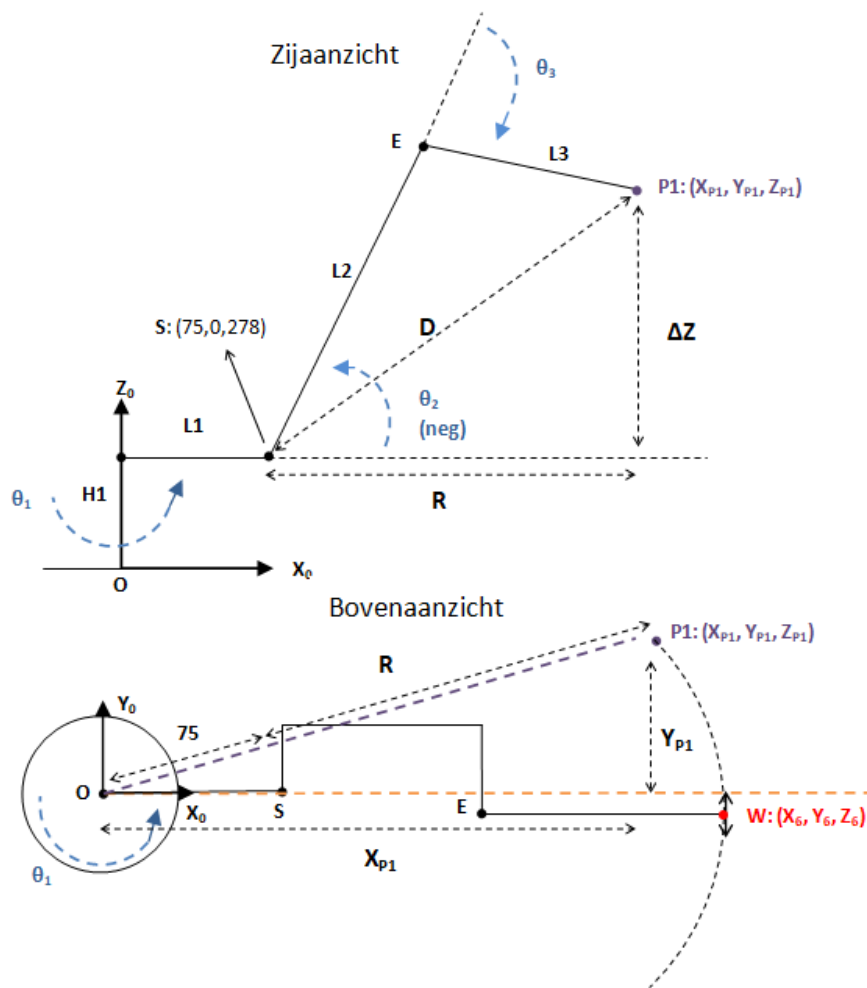
Hieruit volgt:

	$\cos(\theta_3) = \frac{D^2 - L2^2 - L3^2}{2 * L2 * L3}$	(7.63)
--	--	--------

Hieruit kan eenvoudig  $\theta_3$  berekend worden met de acos-functie. Deze functie is in vele programmeeromgevingen onnauwkeurig bij hoeken naderend naar nul. Daarom zal er gebruik gemaakt worden van de atan-functie:

	$\theta_3 = \text{atan} \left( \frac{\sin(\theta_3)}{\cos(\theta_3)} \right)$ Met $\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)}$	(7.64)
--	---	--------





Figuur 7-31 Berekening  $\theta_3$  - tweede configuratie

#### 7.3.5.2.4 Conclusie

Bij het berekenen van  $\theta_3$  zal er gewerkt worden vanuit een bepaalde configuratie. In deze configuratie mag  $\theta_2$  zijn volledige theoretische bereik doorlopen, zolang de elbow-up configuratie gerespecteerd wordt.

#### 7.3.5.3 Schouderrotatie ( $\theta_2$ )

Tot slot wordt de schouderrotatie berekend. Deze volgt uit de berekening van de elleboogrotatie. Er wordt verder gewerkt vanaf figuur 7-31, door het toevoegen van de hoeken  $\alpha$  en  $\beta$ . Alle aanpassing zijn te zien op figuur 7-32 en zijn in het roodbruin getekend. Deze figuur geldt voor de configuraties 1, 2 en 3 te zien in figuur 7-30. Vervolgens wordt in de 4<sup>e</sup> configuratie uitgewerkt volgens figuur 7-33.

##### 7.3.5.3.1 Configuraties 1, 2, 3 -> Kwadranten 1Z, 4Z

De hoek  $\theta_2$  kan als volgt gevonden worden:

	$\theta_2 = (-\alpha - \beta) = -(\alpha + \beta)$	(7.65)
--	--	--------

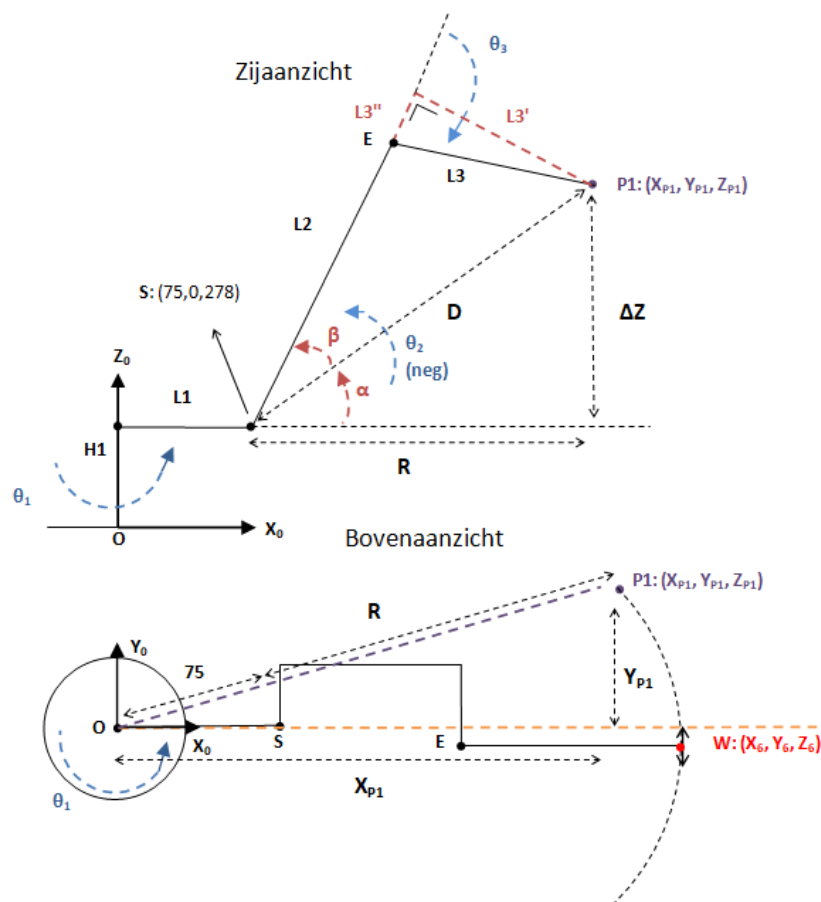
De hoek  $\theta_2$  is negatief door de keuze van de positieve draairichting.

De hoek  $\alpha$  volgt uit de atan-functie:

	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right)$	(7.66)
	$\text{met } R = (\sqrt{X_{P1} + Y_{P1}}) \pm 75 \xrightarrow{\text{Kwadrant 1Z,4Z}} R = (\sqrt{X_{P1} + Y_{P1}}) - 75$	(7.67)

De hoek  $\beta$  kan vervolgens bepaald worden door de hulplijnen  $L3'$  en  $L3''$  te tekenen en hierna de atan-functie te gebruiken:

	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right)$	(7.68)
	$\text{met } L3' = L3 \sin(\theta_3) \text{ en } L3'' = L3 \cos(\theta_3)$	(7.69)



**Figuur 7-32 Berekening  $\theta_2$  (figuur geadapteerd van fig. 7-31)**

### 7.3.5.3.2 Configuratie 4 -> Kwadrant 3Z

De werkwijze bij de 4<sup>e</sup> configuratie is gelijk aan die van de andere configuraties. Kenmerkend aan deze configuratie is dat de pols in het kwadrant 3Z ligt.

Om de berekening van de schouderrotatie te tonen wordt er een variant getekend op de 4<sup>e</sup> configuratie, met  $\theta_2 \in ]0^\circ;90^\circ]$  en de pols in kwadrant 3Z gelegen. Deze configuratie, 4<sup>e</sup> genoemd, is te zien in figuur 7-33.

De hoek  $\theta_2$  kan als volgt gevonden worden:

	$\theta_2 = (-\alpha + \beta)$	$(7.70)$
--	--------------------------------	----------

De hoek  $\theta_2$  is positief door de keuze van de positieve draairichting.

De hoek  $\alpha$  volgt eenvoudig uit de atan-functie:

	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right)$	$(7.71)$
--	---	----------

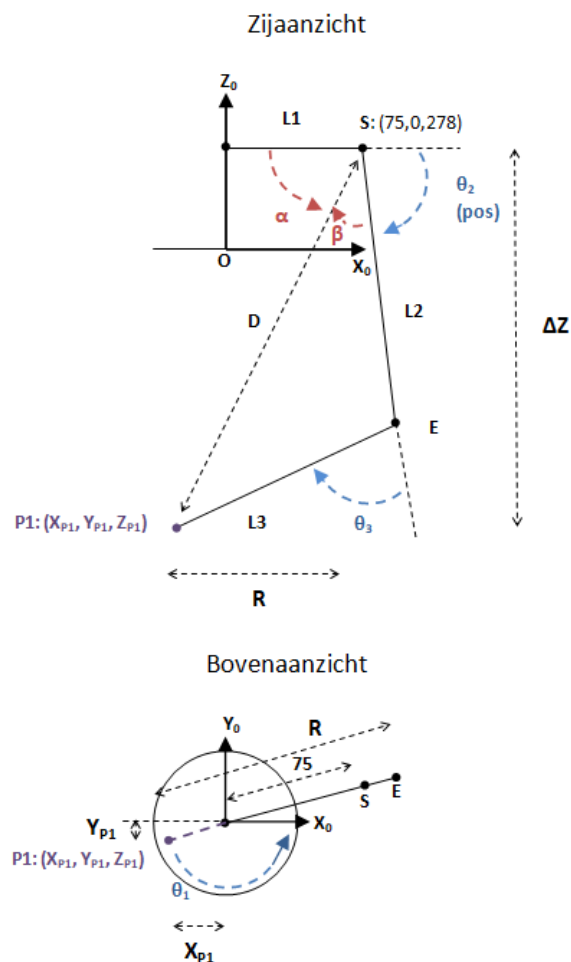
	$\text{met } R = (\sqrt{X_{P1} + Y_{P1}}) \pm 75 \xrightarrow{\text{Kwadrant 3Z}} R = (\sqrt{X_{P1} + Y_{P1}}) + 75$	$(7.72)$
--	--	----------

Bij R wordt nu 75 opgeteld doordat de pols in kwadrant 3Z ligt.

De hoek  $\beta$  kan hierna bepaald op analoge wijze als bij 7.3.5.3.1.

	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right)$	$(7.73)$
--	---	----------

	$\text{met } L3' = L3 \sin(\theta_3) \text{ en } L3'' = L3 \cos(\theta_3)$	$(7.74)$
--	--	----------



**Figuur 7-33 Berekening  $\theta_2$  volgens de configuratie 4'**

### 7.3.6 Inverse oriëntatiekinematica

De inverse oriëntatiekinematica kan op verschillende manieren berekend worden. Hieronder wordt er een analytische variant uitgewerkt. Hierbij dient eerst de inverse positiekinematica opgelost te worden, alvorens er kan overgegaan worden tot de berekening van de inverse oriëntatiekinematica [116] [118].

Vooraf aan de berekening wordt oriëntatie van de eendeffector meegegeven in matrixvorm, zoals beschreven in 7.3.3.

Deze rotatiematrix, verder ook R genoemd, beschrijft hoe het coördinatenframe van de pols (of eendeffector) gelegen is ten opzichte van het basisframe. Met andere woorden is de rotatiematrix R van de eendeffector gelijk aan de totale rotatiematrix  ${}^0R_6$  van de robotarm in die specifieke oriëntatie.

Indien een gewenste eindpositie gegeven is, kan via deze oriëntatie teruggerekend worden naar de pols zoals beschreven in 7.3.4. De polscoördinaten worden verder beschreven als X, Y en Z.

	$\text{Positie pols} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} X_{TCP} - 68 * (a_x) \\ Y_{TCP} - 68 * (a_y) \\ Z_{TCP} - 68 * (a_z) \\ 1 \end{bmatrix}$	(7.75)
--	---	--------

Uit de inverse positiekinematica kan vervolgens de rotatiematrix  ${}^0R_3$  opgemaakt worden uit:

	${}^0R_3 = {}^0R_1 * {}^1R_2 * {}^2R_3$	(7.76)
--	---	--------

Dit gebeurt analoog aan de berekening van de transformatiematrices gebruikt bij de voorwaartse kinematica.

Vervolgens kan de oriëntatie van de pols ten opzichte van de basisarm gevonden worden door de rotatiematrix van de basisarm te inverteren of te transponeren (deze bewerkingen zijn in het geval van de rotatiematrix gelijk, want de matrix is orthogonaal) en vervolgens te vermenigvuldigen met de totale rotatiematrix.

	${}^3R_6 = {}^0R_3^{-1} * {}^0R_6 = {}^0R_3^T * {}^0R_6$	(7.77)
--	--	--------

De matrix die hieruit bekomen wordt, kan algemeen geschreven worden als:

	${}^3R_6 = \begin{bmatrix} n_{xR6-3} & o_{xR6-3} & a_{xR6-3} \\ n_{yR6-3} & o_{yR6-3} & a_{yR6-3} \\ n_{zR6-3} & o_{zR6-3} & a_{zR6-3} \end{bmatrix}$	(7.78)
--	---	--------

Vervolgens kunnen hier de hoeken  $\theta_4$ ,  $\theta_5$  en  $\theta_6$  berekend worden. Er kan gekozen worden tussen twee sets van formules om die hoeken te berekenen. De keuze tussen de eerste en de tweede set wordt in het geval van de AgroBot A.R.M. bepaald door het bereik waarbinnen de hoeken moet vallen.

In MatLab wordt dit als volgt uitgewerkt:

	<code>if (abs(theta4)&lt;90)    (abs(theta5)&lt;135)    (abs(theta6)&lt;90)</code> <code>FORMULESET 1</code> <code>else FORMULESET 2</code> <code>end</code>	
--	---	--

Dit houdt in dat de eerste set formules zal gebruikt worden indien de absolute waarde van de hoeken  $\theta_4$ ,  $\theta_5$  en  $\theta_6$  kleiner is dan  $90^\circ$ . Indien deze grens niet overschreden wordt, zal de andere set formules gebruikt worden. Deze stellen de omgeklapte variant van de pols voor.

#### Formuleset 1

	$\theta_4 = \text{atan2}(a_{yR6-3}, a_{xR6-3})$	(7.79)
	$\theta_5 = \text{atan2}\left(-\sqrt{1 - a_{zR6-3}^2}, -a_{zR6-3}\right)$	(7.80)
	$\theta_6 = \text{atan2}(-o_{zR6-3}, n_{zR6-3})$	(7.81)

#### Formuleset 2

	$\theta_4 = \text{atan2}(-a_{yR6-3}, a_{xR6-3})$	(7.82)
	$\theta_5 = \text{atan2}\left(\sqrt{1 - a_{zR6-3}^2}, -a_{zR6-3}\right)$	(7.83)
	$\theta_6 = \text{atan2}(o_{zR6-3}, n_{zR6-3})$	(7.84)

### Conclusie

Om de inverse oriëntatiekinematica uit te rekenen moet er telkens een positie en een oriëntatie gekozen worden. Aan de hand van de inverse positiekinematica en de vooraf bepaalde oriëntatie kan de oriëntatiekinematica berekend worden.

### Opmerking: Singulariteiten

Doordat de laatste drie rotaties samenkomen in één punt, kan er een situatie ontstaan waarbij de robot 'in de knoop' ligt. Deze situatie doet zich voor als  $\theta_5$  ofwel nul ofwel honderdtachtig graden is. Hierbij is het mogelijk dat een gewenste oriëntatie bekomen kan worden door een rotatie van  $\theta_4$  of  $\theta_6$ . Aangezien deze rotaties rond dezelfde as zijn bij die bepaalde  $\theta_5$ -waarden, kan uit de inverse kinematica niet bepaald worden met welke hoek moet geroteerd worden. Zo'n situatie wordt ook wel een singulariteit genoemd. In een singulariteit kunnen uit de inverse kinematica dus oneindig veel oplossingen gevonden worden voor de hoeken  $\theta_4$  en  $\theta_6$  [119].

## 7.3.7 Numeriek voorbeeld inverse kinematica

### 7.3.7.1 Gegevens

In 7.2.6 wordt de voorwaartse kinematica berekend aan de hand van een aantal vooraf gekende hoeken. Hieruit wordt een positie van de eeffector bekomen alsook de oriëntatie en positie van de pols.

In het numerieke voorbeeld werden de hoeken uit tabel 7-13 gebruikt. Deze worden in volgende berekening gevalideerd met behulp van de inverse kinematica.

Tabel 7-13 Hoeken numeriek voorbeeld

	Hoeken (°)
$\theta_1$	30
$\theta_2$	-45
$\theta_3$	90
$\theta_4$	-45
$\theta_5$	-90
$\theta_6$	-90

De gewenste positie van de eeffector volgt uit tabel 7-14:

Tabel 7-14 Coördinaten Tool Center Point

	Berekende waarde (mm)
$X_{TCP}$	694.2790
$Y_{TCP}$	331.7527
$Z_{TCP}$	332.6829

De oriëntatie van de eeffector en dus ook de pols werd berekend met behulp van de voorwaartse kinematica, deze wordt hieronder overgenomen.

	$R = {}^0R_6 = \begin{bmatrix} 0.0795 & -0.6124 & 0.7866 \\ 0.8624 & -0.3536 & -0.3624 \\ 0.5000 & 0.7071 & 0.5000 \end{bmatrix}$	(7.85)
--	---	--------

### 7.3.7.2 Inverse positiëkinematica

#### 7.3.7.2.1 Omrekenen tot aan de pols

Met behulp van de gewenste oriëntatie kan met de offset van de eeffector teruggerekend worden tot de positie van de pols.

	$\text{Positie TCP} = \begin{bmatrix} 694.2790 \\ 331.7527 \\ 332.6829 \\ 1 \end{bmatrix} =$ $\begin{bmatrix} 0.0795 & -0.6124 & 0.7866 & p_x \\ 0.8624 & -0.3536 & -0.3624 & p_y \\ 0.5000 & 0.7071 & 0.5000 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 0 \\ 0 \\ 68 \\ 1 \end{bmatrix}$	(7.86)
--	---	--------

	$\text{Positie pols} = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} 694.2790 - 68 * (0.7866) \\ 331.7527 - 68 * (-0.3624) \\ 332.6829 - 68 * (0.5000) \\ 1 \end{bmatrix}$ $= \begin{bmatrix} 640.7925 \\ 356.3940 \\ 298.6829 \\ 1 \end{bmatrix}$	(7.87)
--	--	--------

Vanuit deze positie kunnen nu de hoeken  $\theta_1$  tot en met  $\theta_3$  berekend worden.

### 7.3.7.2.2 Basisrotatie ( $\theta_1$ )

De berekening van de basisrotatie volgens 7.3.3.1.

	$\theta_1 = \text{atan}\left(\frac{Y_p}{X_p}\right) + \theta_{1offset}$ $= \text{atan}\left(\frac{356.3940}{640.7925}\right) + 0.9181^\circ = 29.9999^\circ \cong 30^\circ$	(7.88)
	$\text{met } \theta_{1offset} = \text{abs}\left(\text{atan}\left(\frac{-11.75}{\sqrt{640.7925^2 + 356.3940^2}}\right)\right) = 0.9181^\circ$	(7.89)

Deze hoek komt overeen met de hoek ingegeven in het numeriek voorbeeld in 7.2.6.

### 7.3.7.2.3 Elleboogrotatie ( $\theta_3$ )

De berekening van de elleboogrotatie volgens 7.3.3.2.

	$R = \left(\sqrt{X_p^2 + Y_p^2}\right) \pm 75 \xrightarrow{\text{Kwadrant 1Z}}$ $= \left(\sqrt{640.7925^2 + 356.3940^2}\right) - 75 = 658.2337 \text{ mm}$	(7.90)
	$D = \sqrt{658.2337^2 + (298.6829 - 278)^2} = 658.5586 \text{ mm}$	(7.91)
	$\cos(\theta_3) = \frac{D^2 - L2^2 - L3^2}{2 * L2 * L3} = \frac{658.5586^2 - 480^2 - 450.75^2}{2 * 480 * 450.75}$ $= 0.0002861$	(7.92)
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	(7.93)
	$\theta_3 = \text{atan}\left(\frac{\sin(\theta_3)}{\cos(\theta_3)}\right) = \text{atan}\left(\frac{0.9999}{0.0002861}\right) = 89.9836^\circ \cong 90^\circ$	(7.94)

Deze hoek komt overeen met de hoek ingegeven in het numeriek voorbeeld in 7.2.6.

### 7.3.7.2.4 Schouderrotatie ( $\theta_2$ )

De berekening van de schouderrotatie volgens 7.3.3.3.

	$\theta_2 = -(\alpha + \beta) = -(1.7997^\circ + 43.1999^\circ) = 44.9969^\circ \cong 45^\circ$	(7.95)
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = \text{atan}\left(\frac{298.6829 - 278}{658.2337}\right) = 1.7997^\circ$	(7.96)
	$\text{met } R = (\sqrt{X_{P1} + Y_{P1}}) \pm 75 \xrightarrow{\text{Kwadrant 1Z}}$ $= (\sqrt{640.7925^2 + 356.3940^2}) - 75 = 658.2337 \text{ mm}$	(7.97)
	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right) = \text{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	(7.98)
	$\text{met } L3' = L3 \sin(\theta_3) \text{ en } L3'' = L3 \cos(\theta_3) \text{ en } \theta_3 = 90^\circ$	(7.99)

Deze hoek komt overeen met de hoek ingegeven in het numeriek voorbeeld in 7.2.6.

### 7.3.7.2.5 Conclusie inverse positiekinematica

Bij dit voorbeeld zijn alle hoeken die bekomen zijn met behulp van de inverse positiekinematica correct. Om aan te tonen dat deze hoeken ook correct zijn bij andere configuraties worden in bijlage F een aantal extra gevallen uitgewerkt.

### 7.3.7.3 Inverse oriëntatiekinematica

Aan de hand van de gevonden hoeken kan nu de rotatiematrix  ${}^0R_3$  berekend worden. Deze wordt vervolgens getransponeerd.

	${}^0R_3 = \begin{bmatrix} 0.6123 & -0.500 & -0.6125 \\ 0.3535 & 0.8660 & -0.3536 \\ 0.7072 & 0.0000 & 0.7070 \end{bmatrix}$	(7.100)
	${}^0R_3^T = \begin{bmatrix} 0.6123 & 0.3535 & 0.7072 \\ -0.5000 & 0.8660 & 0.0000 \\ -0.6125 & -0.3536 & 0.7070 \end{bmatrix}$	(7.101)

Aangezien zowel de oriëntatie van de basisarm en de eindeffector ten opzichte van de basis gekend is, kan hieruit de oriëntatie van de eindeffector ten opzichte van de basisarm berekend worden.

	${}^3R_6 = {}^0R_3^T * {}^0R_6 = \begin{bmatrix} 0.7071 & 0.0001 & 0.7071 \\ 0.7071 & -0.0000 & -0.7071 \\ -0.0001 & 1.0000 & -0.0001 \end{bmatrix}$	(7.102)
--	--	---------

Ten slotte kunnen nu de hoeken  $\theta_4$  tot en met  $\theta_6$  berekend worden aan de hand van de eerste set formules.

	$\theta_4 = \text{atan2}(-0.7071, 0.7071) = -45.0013^\circ \cong -45^\circ$	(7.103)
	$\theta_5 = \text{atan2}(-\sqrt{1 - 0.0001^2}, 0.0001) = -89.9935^\circ \cong -90^\circ$	(7.104)
	$\theta_6 = \text{atan2}(-1, 0.0001) = -89.9920^\circ \cong -90^\circ$	(7.105)



### 7.3.7.3.1 Conclusie inverse oriëntatiekinematica

Voor dit voorbeeld zijn alle hoeken bekomen met de inverse oriëntatiekinematica correct.

### 7.3.7.4 Extra: inverse kinematica van een pick-and-place robot

Omdat de oriëntatie in het voorgaande voorbeeld moeilijk te visualiseren is zonder hulp van software, wordt in dit voorbeeld een eenvoudig voor te stellen oriëntatie gekozen. De eindeffector ligt horizontaal, in het verlengde van de robotarm. Deze oriëntatie kan bijvoorbeeld gebruikt worden bij een pick-and-place robot.

Met behulp van MatLab (Bijlage E-4) worden de hoeken berekend.

#### 7.3.7.4.1 Gegevens

De positie van de eindeffector wordt gegeven in tabel 7-15.

**Tabel 7-15 Positie eindeffector**

	Positie (mm)
$X_{TCP}$	801.1396
$Y_{TCP}$	-11.75
$Z_{TCP}$	298.6829

De oriëntatie van de eindeffector ligt zoals beschreven in 7.3.3.1. en is voorgesteld in figuur 7-26. De oriëntatiematrix ziet er als volgt uit:

$\text{Oriëntatie pols} = R = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}$	(7.106)
---	---------

#### 7.3.7.4.2 Berekening

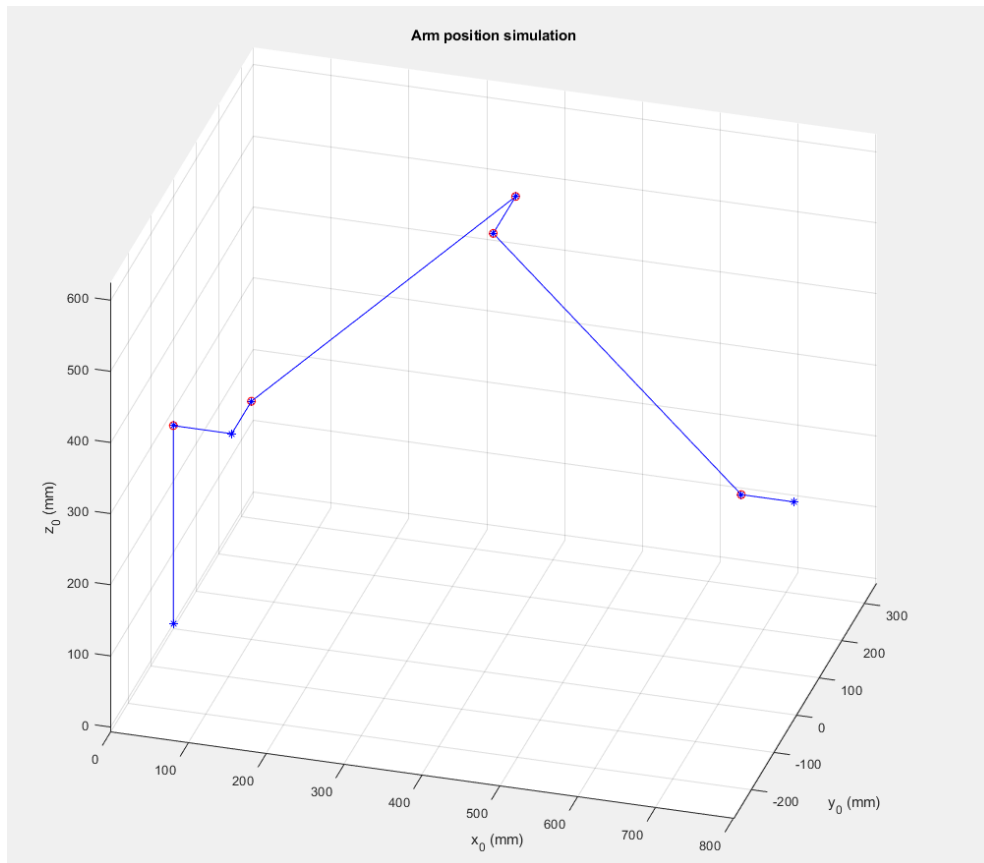
In tabel 7-16 zijn de berekende hoeken terug te vinden.

**Tabel 7-16 Berekende hoeken**

	Hoeken (°)
$\theta_1$	0
$\theta_2$	-45
$\theta_3$	90
$\theta_4$	0
$\theta_5$	-45
$\theta_6$	0

#### 7.3.7.4.3 Visualisatie in MatLab

Tenslotte wordt de robotarm gevisualiseerd in MatLab. Hier is de horizontale oriëntatie duidelijk zichtbaar.



**Figuur 7-34 Simulatie positie arm**

#### 7.3.7.4.4 Conclusie

De berekende hoeken zijn zowel bij de voorwaartse als inverse kinematica correct. In de simulatie is duidelijk te zien dat de gewenste oriëntatie gerespecteerd werd.

## 7.4 Conclusie voorwaartse en inverse kinematica

Door de voorwaartse en inverse kinematica te berekenen, is het nu mogelijk om de robotarm te sturen naar een bepaalde positie. De gebruikte methode zorgt er voor dat er telkens een unieke oplossing is.

Deze wiskunde dient vervolgens omgevormd te worden naar bruikbare code voor het aansturen van de verschillende motoren.

## 8 TESTEN

---

### 8.1 Algemeen

Na montage en bekabeling van de robotarm kon een korte testfase uitgevoerd worden.

Om alle individuele motoren te testen, dienden de drivers eerst correct ingesteld te worden. Door de drivers te koppelen aan de arduino, kunnen de motoren eenvoudig aangestuurd worden.

### 8.2 Controle van de draairichting

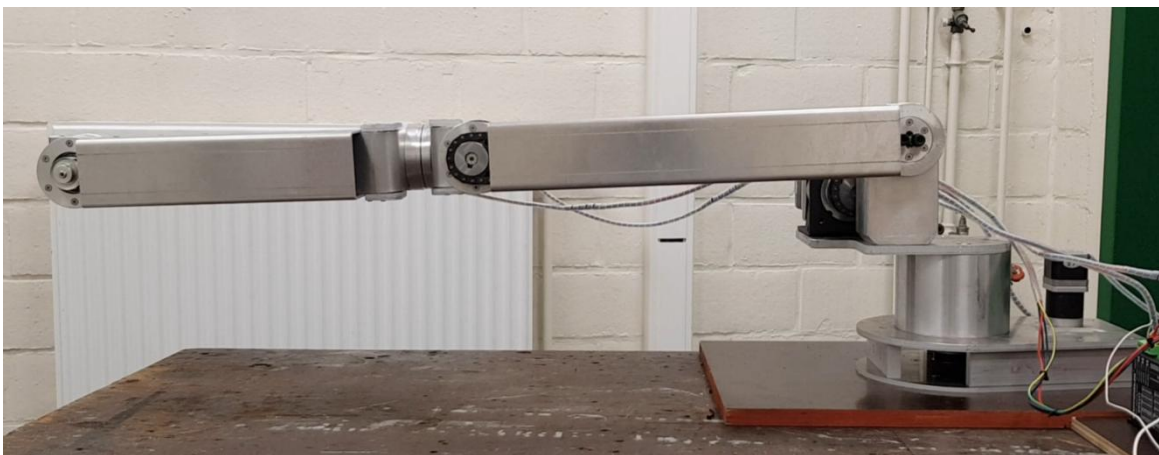
Deze eerste test gebeurde terwijl alle motoren van de arm gedemonteerd waren. Hierdoor kan de draairichting gecontroleerd worden. Na het succesvol uitvoeren van deze proef werden de motoren terug gemonteerd.

### 8.3 Controle van het houdkoppel

Als de drivers gevoed worden, dan worden de motoren bekrachtigd. Het koppel dat de motoren nu kunnen leveren is het houdkoppel.

Om te controleren als de motoren voldoende houdkoppel hebben, wordt de robotarm (onbekrachtigd) met de hand in een bepaalde houding gebracht. Hierna wordt de voeding aangelegd. Als de robotarm in de gewenste positie blijft, is het houdkoppel voldoende.

Deze slechts mogelijke situatie is als de robotarm volledig uitgestrekt is, zoals in figuur 8-1. Hierbij is het houdkoppel in motoren twee, drie, vier en vijf het hoogste. Uit deze test blijkt dat het houdkoppel voldoende hoog is voor deze motoren. De arm helt naar linksonder, niet doordat er te weinig koppel is, maar omdat de arm moeilijk alleen handmatig te positioneren was.



**Figuur 8-1 Robotarm houdkoppel**

Het houdkoppel in motor één is van ondergeschikt belang, omdat deze motor enkel dynamisch belast wordt. Motor zes werd niet gecontroleerd, omdat er nog geen bruikbare eindeffector is. Aangezien de andere motoren een voldoende hoog houdkoppel hebben, en motor zes op dezelfde manier gedimensioneerd is, is het veilig om aan te nemen dat het houdkoppel hier ook voldoende hoog zal zijn.

## 8.4 Sturen van de robot naar een bepaalde positie

Met de motoren terug gemonteerd en de draairichting gekend, is het nu mogelijk om de robot naar een bepaalde positie te sturen.

Vooraf dient bepaald te worden hoeveel pulsen er moeten gestuurd worden zodat de motor over de juiste hoek draait. Met onderstaande formule kan eenvoudig het aantal pulsen berekend worden die moeten gestuurd worden om de motor van een gedreven as één omwenteling te doen maken.

	$\frac{\text{Pulsen}}{\text{toer}} = \text{Overbrengingsverhouding} * \frac{\text{stappen}}{\text{toer}}$	(8.1)
--	---	-------

Met als overbrengingsverhouding het product van de overbrengingsverhouding van de motorgearbox en de ketting-/riemoverbrenging. Het aantal stappen per omwenteling is te vinden (en in te stellen) op de driver.

In figuur 8-2 is de robotarm te zien met motoren twee, drie, vier en vijf aangestuurd. In figuur 8-3 is een andere mogelijke pose weergegeven.



Figuur 8-2 Sturen robotarm pose 1



Figuur 8-3 Sturen robotarm pose 2

## 8.5 Herhaalbaarheid

Bij stappenmotoren is de belasting erg belangrijk, want bij overbelasting zal een stappenmotor stappen overslaan. Aangezien er geen encoders aanwezig zijn, is het niet mogelijk het werkelijk aantal afgelegde stappen te achterhalen. Daarom werd een korte test ingevoerd op de herhaalbaarheid.

De robotarm werd vijftig keer naar dezelfde positie gestuurd, daarna werd telkens de afstand gemeten zoals te zien op figuur 8-4. In tabel 8-1 worden de meetwaarden opgenomen.



**Figuur 8-4 Test herhaalbaarheid**

**Tabel 8-1 Meetwaarden**

# bewegingen	Afstand (mm)
0	22.2
5	22.1
10	22.2
15	22.2
20	22.1
25	22.3
30	22.2
35	22.6
40	22.3
45	22.0
50	22.1

Uit de meetwaarden is duidelijk op te maken dat er geen stappen overgeslaan worden en dat bijgevolg het maximale koppel niet overschreden wordt. Er mag er van uitgegaan worden dat de robot bij normaal gebruik dus telkens de beweging zal uitvoeren zoals opgelegd.

De precisie van de robot kan hieruit geraden worden als millimeter tot centimeterprecies. Uit verdere testen zal de werkelijke precisie blijken.

## 8.6 Speling

De werkelijke precisie zal grotendeels afhangen van de speling die er op de robotarm is. Omdat de arm voornamelijk gebruikt zal worden in de kwadrant 1Z (volgens het zijaanzicht), zal de belasting telkens naar beneden gericht zijn, waardoor de speling in die richting beperkt is. Bij overgang naar kwadrant 2Z zal de speling groter zijn, omdat de kettingen die armdelen één en twee aandrijven dan volgens de andere kant opgespannen zijn.

Deze speling wordt veroorzaakt door het niet volledig kunnen opspannen van kettingen of riemen. Ook zullen de tandriemen een beetje kunnen rekken, waardoor bij hogere belasting de precisie zal dalen.

Een mogelijke oplossing om de speling te reduceren is het rechtstreeks koppelen van de armdelen met de gearbox van de motor.

## 8.7 Conclusie testen

Met deze testen wordt aangetoond dat de robotarm werkt en correct gedimensioneerd is. Eenmaal de inverse kinematica geprogrammeerd is, kunnen verdere testen uitgevoerd worden.

Zo kan onder andere de snelheid en de precisie getest worden door de robotarm van het ene punt naar het ander te sturen. Dit gebeurt door middel van de inverse kinematica, die het aantal benodigde stappen zal uitrekenen. Met behulp van een tijdsmeting en de afgelegde euclidische afstand kan de snelheid van de eindeffector bepaald worden. Door het nameten van de eindpositie kan de precisie van de arm bepaald worden.

Hierna kan overgegaan worden tot het programmeren van de robot voor een concrete toepassing. Eén van de projecten waarvoor de robotarm zal gebruikt worden is het keren van bloemkolen die door middel van een transportband aangevoerd worden.

### Opmerking

In bijlage G zijn de verschillende testprogramma's te vinden die gebruikt werden. Bovendien is ook de code toegevoegd voor het besturen van de robotarm met joysticks, het sturen met behulp van hoeken en de vijf-assige inverse kinematica.

## 9 KOSTPRIJSCALCULATIE ROBOTARM

---

Met een budget van €1,500 euro werd de literatuurstudie aangevat. De vraag bestond erin als er met dit vooropgestelde budget een robotarm kon aangekocht worden met de juiste specificaties. Door de geringe omvang van het budget, was vooraf reeds zo goed als zeker dat de kans tot aankoop zeer klein zou zijn en dat er voor een eigen ontwerp zou moeten gezorgd worden. Na het nodige opzoekwerk bleek deze aannahme te kloppen, waarna de ontwerpfase begon.

Ook tijdens de ontwerpfase bleek een budget van €1,500 euro beperkt te zijn. Doordat alle stukken (met uitzondering van plooi- en laserstukken) binnenshuis bij ILVO vervaardigd werden, kon er op mechanisch vlak sterk bespaard worden. Het budget diende nu enkel tot het bekostigen van de aangekochte componenten en uitbestedingen.

Dit zorgde ervoor dat het mogelijk was om binnen het budget te blijven. Er moest nog steeds geopteerd worden voor goedkopere componenten op vlak van motorisatie en aansturing. Door het budget kon er niet geopteerd worden voor geavanceerde bewerkingstechnieken en moesten de componenten voor de robotarm met manuele machines kunnen vervaardigd worden.

In tabel H-1 in bijlage H zijn alle aangekochte componenten (inclusief BTW, exclusief verzendkosten) te vinden alsook de werken die buitenshuis zijn uitgevoerd. Componenten die in bulk zijn aangekocht (tabel H-2) worden enkel aangerekend voor het percentage dat ze verbruikt zijn. De initiële kost om deze componenten aan te kopen is hoog, maar overschotten kunnen later gebruikt worden. De gepresteerde uren door de techniekers binnen ILVO dienden niet opgenomen te worden in deze calculatie.

De totale kostprijs van de robotarm bedraagt €1,149.93. Het beoogde budget werd ruimschoots gerespecteerd.

## 10 PROJECTEN EN BEMERKINGEN

---

### 10.1 Projecten voor volgende jaren

#### 10.1.1 Zes-assig programmeren

Aangezien er in de Arduino programmeeromgeving niet gerekend kan worden met matrices, dienen deze allemaal handmatig vermenigvuldigd en ingevoerd te worden. Ook inverteren is niet mogelijk, waardoor ook deze handeling zelf moet uitgevoerd worden.

Omdat hier veel code voor nodig zou zijn en het geheugen van de Arduino slechts beperkt is, zal deze berekening in praktijk via MatLab gebeuren. Deze MatLab-waarden dienen dan doorgestuurd te worden, waarna de robotarm naar de gewenste positie kan gestuurd worden.

#### 10.1.2 Invoeren kalibrering en beginpositie

Omdat de stappenmotoren niet voorzien zijn van een encoder, is er geen terugkoppeling mogelijk.

Met behulp van eindeloopschakelaars kan een kalibratie voorzien worden. Deze dienen zodanig geplaatst te worden dat alle motoren aangestuurd worden tot hun maximaal mogelijke verdraaiing. In de uiterste stand van de armdelen worden de eindeloopschakelaars bediend, waardoor de beweging wordt stopgezet. Deze positie is gekend en van hieruit kan terug gestuurd worden naar een mogelijke beginpositie.

#### 10.1.3 Koppelen visiesysteem

Door middel van stereovisie kunnen objecten gedetecteerd worden. Een inleiding hierop kan gevonden worden in de thesis "Ontwerp van een multifunctionele robotarm voor de AgroBot" van Benny en Andy Du Four. Hierin werd een volledig hoofdstuk gewijd aan detectie van objecten in de omgeving [120].

#### 10.1.4 Toepassingsgericht programmeren

Nadat de zes-assige programmering en het koppelen van een visiesysteem kunnen enkele projecten uitgewerkt worden. Een korte opsomming:

- Bloemkoolproject (bloemkolen keren op transportband)
- Plukrobot
- Plaatsspecifiek sproeien
- Sensoren die extra mobiliteit vereisen
- ...

Doordat de robotarm beschikt over zes vrijheidsgraden is het gebruiksspectrum van de robot enorm groot.

#### 10.1.5 Invoeren van extra veiligheidsmaatregelen

In paragraaf 6.7 worden reeds enkele veiligheidsmaatregelen aangehaald. Een verder onderzoek kan extra inzichten geven, waardoor de gebruiker minder risico loopt.



## **10.2 Mogelijke aanpassingen**

### **10.2.1 Keuze motoren**

#### **10.2.1.1 Type**

Op termijn kunnen de stappenmotoren vervangen worden door equivalente servomotoren. Zo kan eenvoudig een terugkoppeling ingewerkt worden, waardoor het programmeren een stuk eenvoudiger kan. Dit brengt natuurlijk een duurdere kostprijs met zich mee.

#### **10.2.1.2 Motorgrootte**

De motoren werden gekozen op basis van het benodigde koppel. Echter zouden de Nema 17 motoren kunnen vervangen worden door Nema 23 motoren. Dit zou ervoor zorgen dat er een hoger houdkoppel is.

Bovendien zou de motor van de eindeffector kunnen vervangen worden door een motor die ook gevoed kan worden met een 48 VDC bron.

Door het gebruik van grotere motoren kan de overbrengingsverhouding verlaagd worden, waardoor de robotarm sneller kan bewegen.

### **10.2.2 Keuze overbrengingen**

Initieel werden alle overbrengingen uitgevoerd met behulp van versnellingskasten (gekoppeld aan de motoren) en tandriemschijven.

Deze werden reeds deels vervangen door kettingen. Deze zijn stijver en zullen minder speling toelaten.

### **10.2.3 Lagermontage**

De lageropspanning gebeurt nu door het aandraaien van bouten op de verschillende armdelen. Dit is echter niet ideaal voor de lagers, waardoor deze snel zullen slijten. De lagers kunnen eventueel vervangen worden door gewone kogellagers die op hun plaats worden gehouden door middel van verende elementen.

### **10.2.4 Montage armdelen**

De armdelen worden via de rotatie-assen op elkaar vastgebout. Dit zou op termijn kunnen vervangen worden, want uit testen bleek dat er snel speling op de armdelen komt doordat de flens waarop de bouten aangespannen zijn, uitslijt.

## **10.3 Bemerkingen**

In de literatuurstudie werden een aantal robotarmen voorgesteld die gebruikt werden in de landbouw. Deze werden telkens bevestigd op een lineaire geleiding. In de probleemstelling werd gesteld dat het gebruik op een veldrobot tot nieuwe inzichten kan leiden.

De reden dat andere projecten deze robots telkens op een lineaire geleiding plaatsen is omdat de robotarmen op een mobiele robot veel trillingen ondergaan, wat nefast is voor de precisie van de robot. Door de trillingen komt er namelijk snel speling op de precies afgewerkte componenten.

In de probleemstelling is ervoor gekozen om geen rekening te houden met de trillingen, omdat de robotarm enkel op een stilstaande robot zou gebruikt worden. Dit is echter een theoretische situatie die zich niet zal voordoen. Zelf als de robotarm stevig bevestigd is op de

veldrobot en niet gebruikt wordt tijdens het rijden, zal deze toch invloed en dus ook slijtage door de trillingen ondervinden.

Voor daadwerkelijk gebruik in het veld zullen er maatregelen moeten getroffen worden om de trillingen tot een minimum te beperken. Een aanpassing van het ontwerp kan er eventueel ook voor zorgen dat de trilling minder invloed heeft.

# 11 CONCLUSIE

---

In deze masterproef werd gestart van een bestaand concept, een veldrobot op rupsaandrijving. Na een korte literatuurstudie over verschillende veldrobots werd er overgegaan tot het ontwerp van een bovenbouw, die geschikt is om verschillende werktuigen op te bevestigen.

Eenmaal deze bovenbouw voltooid was, moesten er werktuigen ontworpen worden. Eén van deze werktuigen, een robotarm, vormde het hoofddoel van deze masterproef. Er werd afgewogen welk type robotarm het meest geschikt was en hoeveel vrijheidsgraden er nodig waren. Na een korte bespreking van de robotarmen in de industrie en de landbouw werd er geopteerd voor een eigen ontwerp. Het unieke aan dit project is het gebruik van een robotarm op een veldrobot, wat tot nieuwe inzichten kan leiden.

Bij het ontwerp werd er rekening gehouden met tal van factoren. Materiaalkeuze, motoren en overbrengingen waren erg belangrijk. Deze werden zo veel als mogelijk wiskundig nagerekend om de correctheid te staven.

Na de ontwerpfasen werden de componenten van de robotarm vervaardigd, waarna de assemblage volgde. Snel was duidelijk dat de tandriemoverbrengingen die oorspronkelijk voorzien waren, niet voor alle armdelen geschikt waren. Daarom werd er bij de armdelen met de grootste koppels geopteerd voor een ketting. Na deze aanpassing was de robot gebruiksklaar.

Voor er over werd gegaan tot de testfase werd eerst de volledige voorwaartse en inverse kinematica uitgewerkt. Aan de hand van deze kinematica kan zowel de eindpositie berekend worden uit een aantal gegeven hoeken als dat de hoeken kunnen berekend worden uit de eindpositie en gewenste oriëntatie.

Vervolgens kon de robot getest worden. Om te voldoen aan de opdrachtoomschrijving, was het essentieel dat de werking van de robotarm kon aangetoond worden. Dit werd bewezen door de robotarm statisch te belasten in volledig uitgestrekte stand, alsook door de robot een gegeven aantal bewegingen te laten doen om erna te controleren als de eindpositie gelijk bleef. Uit deze testen bleek dat alle motoren correct gedimensioneerd werden. Vervolgens werd de vijf-assige kinematica geprogrammeerd, zodat deze op een later tijdstip eenvoudig kan geïmplementeerd worden. De zes-assige kinematica die uitgewerkt werd, kan ook eenvoudig via MatLab geïmplementeerd worden.

Een laatste belangrijke factor van het project was de kostprijs. Deze werd volledig nagerekend. Het totaalbedrag komt neer op €1149.93, wat ruim onder het voorziene budget van €1,500 euro ligt.

Tenslotte worden enkele mogelijke projecten voor volgende jaren toegelicht, alsook een aantal richtlijnen voor mogelijke aanpassingen die kunnen doorgevoerd worden.

## 12 BIBLIOGRAFIE

---

- [1] „Logo\_ILVO\_2016,” [Online]. Available: [http://www.ilvo.vlaanderen.be/Portals/68/images/OverILVO/logo\\_ILVO\\_2016\\_lr.jpg](http://www.ilvo.vlaanderen.be/Portals/68/images/OverILVO/logo_ILVO_2016_lr.jpg). [Geopend 2017 oktober 14].
- [2] „Logo ILVO,” [Online]. Available: <https://www.ilvo.vlaanderen.be/NL/Pers-en-media/Logos#.Wul9b4iFOUk>. [Geopend 2 mei 2018].
- [3] „Over ILVO - Instituut voor Landbouw - en Visserijonderzoek (ILVO),” [Online]. Available: <http://www.ilvo.vlaanderen.be/language/nl-BE/NL/Over-ILVO.aspx#.WckPGsirSUK>. [Geopend 25 September 2017].
- [4] D. Nuyttens, *Precisielandbouw en Gewasbescherming*, Merelbeke: ILVO.
- [5] S. Cool, *Precision Agriculture - Presentatie*, Gent.
- [6] „Worldometers,” [Online]. Available: <http://www.worldometers.info/world-population/>. [Geopend 12 Februari 2018].
- [7] *Honger de wereld uit. Maar Hoe?*, Gent: Broederlijk Delen, 2018.
- [8] „Whizolosophy,” [Online]. Available: <https://whizolosophy.com/category/the-legacy-we-leave/article/population-of-the-earth>. [Geopend 12 Februari 2018].
- [9] „Airborne Technologies,” [Online]. Available: <http://www.airbornetechnologies.at/references/references-data-capturing/references-precision-farming-agriculture-management>. [Geopend 5 mei 2018].
- [10] „John Deere,” [Online]. Available: [https://www.deere.com/en\\_INT/products/equipment/agricultural\\_management\\_solutions/guidance\\_and\\_machine\\_control/guidance\\_and\\_machine\\_control.page](https://www.deere.com/en_INT/products/equipment/agricultural_management_solutions/guidance_and_machine_control/guidance_and_machine_control.page). [Geopend 6 november 2017].
- [11] „AgroBot SW6010,” [Online]. Available: <http://www.agrobot.com/products.html>. [Geopend 15 november 2017].
- [12] „Precisielandbouw Groenkennis,” [Online]. Available: <https://precisielandbouw.groenkennisnet.nl/display/EL/Kongskilde+Robotti>. [Geopend 15 november 2017].
- [13] „FroboMind Wiki,” [Online]. Available: [http://frobomind.org/oldwiki/index.php%3Ftitle=FroboMind\\_Robot:Kongskilde\\_Robotti.html](http://frobomind.org/oldwiki/index.php%3Ftitle=FroboMind_Robot:Kongskilde_Robotti.html). [Geopend 15 november 2017].
- [14] „INBIOM - Biomass,” [Online]. Available: <http://www.inbiom.dk/en/projects/completed-biomass-projects/grassbots>. [Geopend 15 november 2017].
- [15] „FroboMind Grassbot,” [Online]. Available: [http://frobomind.org/oldwiki/index.php%3Ftitle=FroboMind\\_Robot:GrassBots.html](http://frobomind.org/oldwiki/index.php%3Ftitle=FroboMind_Robot:GrassBots.html). [Geopend 15 november 2017].
- [16] „Precisielandbouw Amazone Bonirob,” [Online]. Available: <https://precisielandbouw.groenkennisnet.nl/display/EL/Amazone+Bonirob>. [Geopend 15 november 2017].
- [17] „Amazone BiniRob Feldroter,” [Online]. Available: <http://www.amazone.de/1800.asp>. [Geopend 15 november 2017].
- [18] „Ibex Automation Ltd,” [Online]. Available: <http://www.ibexautomation.co.uk/>. [Geopend 15 november 2017].
- [19] „Innovate UK - GOV,” [Online]. Available: <https://www.gov.uk/government/organisations/innovate-uk>. [Geopend 15 november 2017].

- [20] R. Claeys en P. Bettens, "*Ontwerp en bouw van een robot voor deelname aan het FieldRobot event*", Departement Gent: Katholieke Hogeschool Sint-Lieven, 2010-2011.
- [21] R. Claeys, Artist, *AgroBot V1*. [Art]. KULeuven, 2010-2011.
- [22] J. Clauw en A. Vandenbroucke, "De AgroBot", Departement Gent: Katholieke Hogeschool Sint-Lieven - Technologicampus Gent, 2011-2012.
- [23] J. de Pauw, "De AgroBot", Departement Gent: Katholieke Hogeschool Sint-Lieven - Technologicampus Gent, 2012-2013.
- [24] A. Du Four en B. Du Four, "Ontwerpen van een multifunctionele robotarm voor de AgroBot", Departement Gent: KULeuven - Technologicampus Gent, 2013-2014.
- [25] „Arch McGill,” [Online]. Available: <http://www.arch.mcgill.ca/prof/sijpkcs/ice-lecture/sii6.gif>. [Geopend 6 November 2017].
- [26] „Loop Technology,” [Online]. Available: <http://www.looptechnology.com/robotic-robot-types.asp#.WgAbxmj9SUK>. [Geopend 6 november 2017].
- [27] „Robotpark Academy,” [Online]. Available: <http://www.robotpark.com/academy/all-types-of-robots/stationary-robots/cartesian-gantry-robots/>. [Geopend 6 november 2017].
- [28] „SageRobot,” [Online]. Available: [http://www.sagerobot.com/sites/default/files/styles/full\\_post/public/Light%20Duty%20Gantry%2002\\_0.png?itok=cf5rCdNJ](http://www.sagerobot.com/sites/default/files/styles/full_post/public/Light%20Duty%20Gantry%2002_0.png?itok=cf5rCdNJ). [Geopend 6 november 2017].
- [29] „Toshiba-Machine,” [Online]. Available: <http://www.toshiba-machine.co.jp/en/product/robot/case/case02.html>. [Geopend 2017 november 6].
- [30] „RobotPark Academy,” [Online]. Available: <http://www.robotpark.com/academy/all-types-of-robots/stationary-robots/cylindrical-robots/>. [Geopend 6 november 2017].
- [31] „Hudson Robotics,” [Online]. Available: <http://hudsonrobotics.com/wp-content/uploads/2014/01/sciclops.jpg>. [Geopend 6 November 2017].
- [32] „RobotPark Academy,” [Online]. Available: <http://www.robotpark.com/academy/all-types-of-robots/stationary-robots/spherical-robots/>. [Geopend 6 november 2017].
- [33] „Stanford Infolab,” [Online]. Available: <http://infolab.stanford.edu/pub/voy/museum/pictures/display/1-Robot.htm>. [Geopend 6 November 2017].
- [34] „RobotPark Academy,” [Online]. Available: <http://www.robotpark.com/academy/all-types-of-robots/stationary-robots/scara-robots/>. [Geopend 6 november 2017].
- [35] „Assembly Mag,” [Online]. Available: <https://www.assemblymag.com/articles/93338-whats-new-with-scara-robots>. [Geopend 6 november 2017].
- [36] „Loop Technology Articulated / Jointed Arm Robot,” [Online]. Available: <http://www.looptechnology.com/robotic-robot-types.asp#.WgwaXEriaUI>. [Geopend 6 november 2017].
- [37] „RobotPark Academy Articulated Robots,” [Online]. Available: <http://www.robotpark.com/academy/all-types-of-robots/stationary-robots/robotic-arms-articulated-robots/>. [Geopend 6 november 2017].
- [38] „GlobalSpec Insights,” [Online]. Available: <http://insights.globalspec.com/article/4788/what-is-the-real-cost-of-an-industrial-robot-arm>. [Geopend 6 november 2017].
- [39] „Omron Industrial,” [Online]. Available: [https://industrial.omron.us/en/media/Delta\\_robot\\_XL\\_tcm849-100817.jpg](https://industrial.omron.us/en/media/Delta_robot_XL_tcm849-100817.jpg). [Geopend 6 november 2017].
- [40] „Wikipedia - Parallel Manipulator,” [Online]. Available: [https://en.wikipedia.org/wiki/Parallel\\_manipulator](https://en.wikipedia.org/wiki/Parallel_manipulator). [Geopend 6 november 2017].

- [41] „Wikipedia - Delta Robot,” [Online]. Available: [https://en.wikipedia.org/wiki/Delta\\_robot](https://en.wikipedia.org/wiki/Delta_robot). [Geopend 6 november 2017].
- [42] „Mechatronica Delft,” [Online]. Available: <http://www.mechatronicadelft.nl/wat-is-robotica/industriële-robot/delta-robot.html>. [Geopend 6 november 2017].
- [43] „Echord,” [Online]. Available: [http://www.echord.info/file/Attachments/wikis/website/prada/Images/Adept\\_Quattro\\_Packaging-line.jpg](http://www.echord.info/file/Attachments/wikis/website/prada/Images/Adept_Quattro_Packaging-line.jpg). [Geopend 6 november 2017].
- [44] „ABB IRB 1200,” [Online]. Available: <http://new.abb.com/products/robotics/industrial-robots/irb-1200>. [Geopend 20 November 2017].
- [45] „ABB IRB 1200 5/900,” [Online]. Available: <https://www.robots.com/robots/abb-irb-1200-5-0-9>. [Geopend 20 november 2017].
- [46] „RobotWorx Fanuc M-6iB/2HS,” [Online]. Available: <https://www.robots.com/robots/fanuc-m-6ib-2hs>. [Geopend 20 november 2017].
- [47] „KUKA KR 6 R900,” [Online]. Available: <https://www.robots.com/robots/kuka-kr-6-r900-sixx>. [Geopend 20 november 2017].
- [48] „Universal Robots UR5,” [Online]. Available: <https://www.robots.com/robots/universal-robot-ur5>. [Geopend 20 november 2017].
- [49] „IRB 1200 A compact, flexible, fast and functional small industrial robot,” [Online]. Available: <https://search-ext.abb.com/library/Download.aspx?DocumentID=9AKK106103A6066&LanguageCode=en&DocumentPartId=&Action=Launch>. [Geopend 20 november 2017].
- [50] „Intorobotics: Robots in Agriculture,” [Online]. Available: <https://www.intorobotics.com/35-robots-in-agriculture/>. [Geopend 21 november 2017].
- [51] „The Hydroponic, robotic future of farming in greenhouses,” [Online]. Available: <https://www.wired.com/story/the-hydroponic-robotic-future-of-farming-in-greenhouses-at-iron-ox/>. [Geopend 21 november 2017].
- [52] „This robot handles the entire process of growing lettuce by itself,” [Online]. Available: <https://www.fastcompany.com/40501937/this-robot-handles-the-entire-process-of-growing-lettuce-by-itself>. [Geopend 21 november 2017].
- [53] „Robot farm workers take to the field,” [Online]. Available: <https://www.newscientist.com/gallery/farm-robots/>. [Geopend 21 november 2017].
- [54] „Wageningen UR: An autonomous harvesting machine for cucumbers,” [Online]. Available: [https://www.wur.nl/upload\\_mm/5/5/7/c221711e-98da-4865-805a-8fc8531aa624\\_flyer\\_cucumber%20harvesting\\_robot\\_uk.pdf](https://www.wur.nl/upload_mm/5/5/7/c221711e-98da-4865-805a-8fc8531aa624_flyer_cucumber%20harvesting_robot_uk.pdf). [Geopend 21 november 2017].
- [55] „Cucumber harvesting robot,” [Online]. Available: <https://www.wur.nl/en/show/Cucumber-harvesting-robot.htm>. [Geopend 21 november 2017].
- [56] „\$50.000 strawberry picking robot to go on sale in Japan,” [Online]. Available: <https://www.cnet.com/news/50000-strawberry-picking-robot-to-go-on-sale-in-japan/>. [Geopend 21 november 2017].
- [57] *Inventor*, Autodesk, 2017.
- [58] *Solidworks*, Dassault Systems, 2017.
- [59] „DN Trailers - Mini tractoren,” [Online]. Available: <https://dn-trailers.be/minitractoren/>. [Geopend 1 december 2017].
- [60] „Robohub: How many axes does my robot need,” [Online]. Available: <http://robohub.org/how-many-axes-does-my-robot-need/>. [Geopend 13 februari 2018].
- [61] „Overview of robotic manipulators,” [Online]. Available: [http://surprisingfacts.com/wp-content/uploads/2017/07/1500503982\\_540\\_overview-of-robotic-manipulators-universal-robots-blog-of-the-company-top-3d-shop-geektimes.jpg](http://surprisingfacts.com/wp-content/uploads/2017/07/1500503982_540_overview-of-robotic-manipulators-universal-robots-blog-of-the-company-top-3d-shop-geektimes.jpg). [Geopend 13 februari 2018].

- [62] „Aluminium 5754,” [Online]. Available: <https://www.hego.nl/aluminium-5754/>. [Geopend 25 april 2018].
- [63] „Aluminium en aluminiumlegeringen,” [Online]. Available: [http://www.demetaalgids.nl/index.php?page=metalen&metaal=Aluminium+en+aluminium+legeringen&id=1&pagina=Soorten%2C+kwaliteiten%2C+eigenschappen&pagina\\_id=23](http://www.demetaalgids.nl/index.php?page=metalen&metaal=Aluminium+en+aluminium+legeringen&id=1&pagina=Soorten%2C+kwaliteiten%2C+eigenschappen&pagina_id=23). [Geopend 25 april 2018].
- [64] „AZO - Stainless steel AISI 304,” [Online]. Available: <https://www.azom.com/article.aspx?ArticleID=965>. [Geopend 25 april 2018].
- [65] „AZO - Aluminium EN AW 6082,” [Online]. Available: <https://www.azom.com/article.aspx?ArticleID=2813>. [Geopend 25 april 2018].
- [66] „BRECO Zahnscheiben,” [Online]. Available: [http://www.breco.de/content/downloads/products/en/Synchronising\\_pulleys.pdf](http://www.breco.de/content/downloads/products/en/Synchronising_pulleys.pdf). [Geopend 25 april 2018].
- [67] K. Bruggeman, „Massatraagheidsmoment - theoretische bepaling,” in *Servomotoren en toepassing*, Gent, KULeuven, 2017-2018, pp. 8-7.
- [68] „Stepper motors vs. Servo motors - selecting a motor,” [Online]. Available: <https://www.lifewire.com/stepper-motor-vs-servo-motors-selecting-a-motor-818841>. [Geopend 15 april 2018].
- [69] „Stepper motor or servomotor,” [Online]. Available: <https://www.motioncontroltips.com/stepper-motor-servomotor/>. [Geopend 15 april 2018].
- [70] „Stepper motor or Servo motor,” [Online]. Available: <https://www.kollmorgen.com/en-us/service-and-support/knowledge-center/white-papers/stepper-motor-or-servo-motor-which-should-it-be/>. [Geopend 15 april 2018].
- [71] „Ketting - en riemoverbrenging,” [Online]. Available: <https://www.niutec.nl/Mechanica/Ketting.pdf>. [Geopend 20 april 2018].
- [72] „OND Transmissions,” [Online]. Available: <https://dutch.alibaba.com/g/colored-roller-chain.html>. [Geopend 20 april 2018].
- [73] „ISO R606 - DIN 8187 Fenner Roller Chains,” [Online]. Available: [http://www.fptgroup.com/dss/docs/679\\_02\\_Fenner\\_Chain\\_Drives.pdf](http://www.fptgroup.com/dss/docs/679_02_Fenner_Chain_Drives.pdf). [Geopend 20 april 2018].
- [74] „Optibelt - Technical manual rubber timing belt drives,” [Online]. Available: [https://www.optibelt.com/fileadmin/extranet/power\\_transmission/technische\\_Unterlagen/OPTIBELT-TM-Rubber-Timing-Belt-Drives.pdf](https://www.optibelt.com/fileadmin/extranet/power_transmission/technische_Unterlagen/OPTIBELT-TM-Rubber-Timing-Belt-Drives.pdf). [Geopend 20 april 2018].
- [75] „Optibelt Technical manual,” [Online]. Available: [https://www.optibelt.com/fileadmin/extranet/power\\_transmission/technische\\_Unterlagen/OPTIBELT-TM-Rubber-Timing-Belt-Drives.pdf](https://www.optibelt.com/fileadmin/extranet/power_transmission/technische_Unterlagen/OPTIBELT-TM-Rubber-Timing-Belt-Drives.pdf). [Geopend 25 april 2018].
- [76] D. Muhs, H. Wittel, M. Becker, D. Jannasch en J. Vossiek, „Moment(schub)belastete Anschlüsse,” in *Roloff/Matek Machineonderdelen*, Vieweg, Springer, 2015, pp. 264-265.
- [77] W. De Clippeleer, B. Wellekens, W. Dax, K. Gundelfinger, W. Häffner, H. Itschner en G. Kotsch, „Schroeven,” in *Tabellenboek voor metaaltechniek*, Mechelen, Plantyn, 2000, p. 201.
- [78] M. Juwet, „Criterium van de grootste schuifspanning - Tresca,” in *Algemene sterkteleer voor machinebouw*, StuDie, 2011, pp. 219-223.
- [79] „Key / Spline Strenght calculation,” [Online]. Available: [http://www.roytech.co.uk/Useful\\_Tables/Keyways/key\\_strength.html](http://www.roytech.co.uk/Useful_Tables/Keyways/key_strength.html). [Geopend 19 april 2018].
- [80] „Stainless steels - 316,” [Online]. Available: <https://www.azom.com/properties.aspx?ArticleID=863>. [Geopend 19 april 2018].
- [81] M. Juwet, „Vlakke inlegspie,” in *Berekenen van machine-onderdelen*, StuDie, 2001, pp. IV-2.
- [82] „Aluminium EN-AW 2017A,” [Online]. Available: <https://www.alumeco.com/aluminium/sheets/en-aw-2017a/1-x-2000--1000-mm/p/402/23061>.

- [Geopend 20 april 2018].
- [83] M. Juwet, „Tabel 18: Toelaatbare vlaktedruk,” in *Berekenen van machine-onderdelen*, StuDie, 2001, p. 152.
- [84] „Lagertechnologie,” [Online]. Available: <http://www.mechanismen.be/mechanismen/lagers/lagertechnologie2.pdf>. [Geopend 25 april 2018].
- [85] „Taper roller bearing 30206 J2-Q,” [Online]. Available: <https://uk.rs-online.com/web/p/roller-bearings/2868319/>. [Geopend 25 april 2018].
- [86] „Bearing installation and maintenance guide,” [Online]. Available: [http://www.skf.com/binary/tcm:12-114679/140-710%20BIMG%2011\\_2012\\_tcm\\_12-114679.pdf](http://www.skf.com/binary/tcm:12-114679/140-710%20BIMG%2011_2012_tcm_12-114679.pdf). [Geopend 25 april 2018].
- [87] „SKF Binary,” [Online]. Available: [http://www.skf.com/binary/21-3541/1\\_\\_en\\_\\_0008f21\\_tcm\\_12-3541.gif](http://www.skf.com/binary/21-3541/1__en__0008f21_tcm_12-3541.gif). [Geopend 25 april 2018].
- [88] „What is Arduino,” [Online]. Available: <https://www.arduino.cc/en/Guide/Introduction>. [Geopend 24 april 2018].
- [89] „Arduino,” [Online]. Available: <https://www.arduino.cc/>. [Geopend 23 april 2018].
- [90] „Controllino,” [Online]. Available: <https://controllino.biz/>. [Geopend 23 april 2018].
- [91] „Controllino Mega,” [Online]. Available: <https://controllino.biz/controllino/mega/>. [Geopend 23 april 2018].
- [92] „What is a motor driver?,” [Online]. Available: <http://www.futureelectronics.com/en/drivers/motor-driver.aspx>. [Geopend 25 april 2018].
- [93] „Digital Stepping Driver DM542T,” [Online]. Available: <https://www.omc-stepperonline.com/stepper-motor-driver/digital-stepper-driver-10-42a-20-50vdc-for-nema-17-23-24-stepper-motor-dm542t.html>. [Geopend 25 april 2018].
- [94] „Machine Hazard Signs,” [Online]. Available: <https://www.mysafetysign.com/machine-hazard-signs>. [Geopend 25 april 2018].
- [95] „Encyclopaedia Britannica: Kinematics,” [Online]. Available: <https://www.britannica.com/science/kinematics>. [Geopend 24 februari 2018].
- [96] „Wikipedia: Forward Kinematics,” [Online]. Available: [https://en.wikipedia.org/wiki/Forward\\_kinematics#cite\\_note-1](https://en.wikipedia.org/wiki/Forward_kinematics#cite_note-1). [Geopend 24 februari 2018].
- [97] „Wikipedia: Inverse Kinematics,” [Online]. Available: [https://en.wikipedia.org/wiki/Inverse\\_kinematics](https://en.wikipedia.org/wiki/Inverse_kinematics). [Geopend 24 februari 2018].
- [98] „Robotiq: Why singularities can ruin day,” [Online]. Available: <https://blog.robotiq.com/why-singularities-can-ruin-your-day>. [Geopend 26 februari 2018].
- [99] R. N. Jazar, „Chapter 5: Forward Kinematics,” in *Theory of Applied Robotics*, Melbourne, Australia, Springer, 2010, pp. 233-323.
- [100] „Robotiq: How to calculate a robots forward kinematics in 5 easy steps,” [Online]. Available: <https://blog.robotiq.com/how-to-calculate-a-robots-forward-kinematics-in-5-easy-steps>. [Geopend 26 februari 2018].
- [101] „Tekkotsu Robotics: Kinematics,” [Online]. Available: <http://www.tekkotsu.org/Kinematics.html>. [Geopend 27 februari 2018].
- [102] R. N. Jazar, „Denavit-Hartenberg Notation,” in *Theory of Applied Robotics*, Australia, Springer, 2010, pp. 233-234.
- [103] „Society of Robots: Robot Arm Tutorial,” [Online]. Available: [http://www.societyofrobots.com/robot\\_arm\\_tutorial.shtml](http://www.societyofrobots.com/robot_arm_tutorial.shtml). [Geopend 26 februari 2018].
- [104] „Wikipedia: Denavit-Hartenberg Parameters,” [Online]. Available: [https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg\\_parameters](https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg_parameters).



- [Geopend 26 februari 2018].
- [105] „Youtube: Denavit-Hartenberg Reference Frame Layout,” [Online]. Available: <https://www.youtube.com/watch?v=rA9tm0gTIn8>. [Geopend 8 maart 2018].
- [106] R. N. Jazar, „Denavit-Hartenberg Notation,” in *Theory of Applied Robotics*, Australia, Springer, 2010, pp. 235-236.
- [107] „Wikipedia: Denavit- Hartenberg parameters,” [Online]. Available: [https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg\\_parameters](https://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg_parameters). [Geopend 8 maart 2018].
- [108] R. N. Jazar, „Transformation between two adjacent coordinate frames,” in *Theory of Applied Robotics*, Australia, Springer, 2010, pp. 242-243.
- [109] „Stack Overflow: How to find the rotation matrix between two coordinate systems,” [Online]. Available: <https://stackoverflow.com/questions/34391968/how-to-find-the-rotation-matrix-between-two-coordinate-systems>. [Geopend 8 maart 2018].
- [110] M. Almaged, „Forward and Inverse Kinematic Analysis and Validation of the ABB IRB 140 Industrial Robot,” *International Journal of electronics, mechanical and mechatronics engineering*, pp. 1381-1401, 2017.
- [111] „Matlab Ra2018,” [Online]. Available: <https://nl.mathworks.com/products/matlab.html>. [Geopend 19 maart 2018].
- [112] J. J. Craig, „X-Y-Z fixed angles,” in *Introduction to Robotics: Mechanics and Control*, Saddle River, New Jersey, Verenigde Staten, Pearson, 2005, pp. 41-43.
- [113] J. B. Kuipers, „Quaternions and rotation sequences,” *Geometry, Integrability and Quantization*, pp. 127-143, 1999.
- [114] „Mecademic: How is orientation in space represented with Euler angles?,” [Online]. Available: <https://www.mecademic.com/ressources/Euler-angles/Euler-angles>. [Geopend 25 maart 2018].
- [115] „AR2 Kinematic Model,” Chris Annin, Oregon, 2017.
- [116] R. N. Jazar, „Inverse kinematics,” in *Theory of Applied Robotics*, Melbourne, Springer, 2010, pp. 325-380.
- [117] P. J.N., „Inverse kinematica - Pieper voorwaarde,” in *Industrial Robots Programming*, Springer, 2007, p. 9.
- [118] M. Almaged, „Forward and Inverse Kinematic Analysis and Validation of the ABB IRB 140 Industrial Robot,” *International journal of electronics, mechanical and mechatronics engineering*, vol. 7, nr. 2, pp. 1383-1401, 2017.
- [119] A. Owen-Hill, „Why singularities can ruin your day,” [Online]. Available: <https://blog.robotiq.com/why-singularities-can-ruin-your-day>. [Geopend 23 maart 2018].
- [120] B. D. Four en A. D. Four, „Object detectie in de omgeving,” in *Ontwerpen van een multifunctionele robotarm voor de AgroBot*, Departement Gent, KULeuven - Technologiecampus Gent, 2013-2014, pp. 64-79.

## 13 OPMERKINGEN

---



## Bijlage A PROMOTOREN

Ing. A. Breda	Andreas.Breda@kuleuven.be
ir. K. Bruggeman	Koert.Bruggeman@kuleuven.be
ing. R.Claeys	Rico.Claeys@kuleuven.be
dr. ing. S.Coppieters	Sam.Coppieters@kuleuven.be
prof. dr. ir. D. Debruyne	Dimitri.Debruyne@kuleuven.be
dr. ing. S. Debruyne	Stijn.Debruyne@kuleuven.be
dr. ir. D. Degryse	Dominiek.Degryse@kuleuven.be
ing. B. De Man	Benny.DeMan@kuleuven.be
ing. W. D'Haese	Wim.DHaese@kuleuven.be
ing. J. D'hondt	Jordi.Dhondt@kuleuven.be
ing. P. Guns	Piet.Guns@kuleuven.be
ir. M. Juwet	Marc.Juwet@kuleuven.be
ir. I. Martens	Ignace.Martens@kuleuven.be
ing. W. Nica	Ward.nica@kuleuven.be
ing. K. Philips	Klaas.Philips@kuleuven.be
ir. K. Van Daele	Katrien.VanDaele@kuleuven.be
ir. K. Vanden Hautte	Koen.VandenHautte@kuleuven.be
ing. L. Vande Riviere	Luc.VanDeRiviere@kuleuven.be
ing. A. Van de Velde	Arne.VandeVelde@kuleuven.be
ing. I. Van De Voorde	Ivan.VanDeVoorde@kuleuven.be
ing. P. Vanhauwermeiren	Pieter.vanhauwermeiren@kuleuven.be

# Bijlage B LIJST MET GEGEVENS VOOR DE BEREKENING VAN HET KOPPEL

**Tabel B-1 Gegevens berekening statische belasting**

Component	Massa (g)	$\Delta X$ tot as 2 (mm)	Afstand tot as 2 (mm)	Afstand tot as 3 (mm)	Afstand tot as 5 (mm)	$\Delta Y$ tot as 4 (mm)	Afstand tot as 4 (mm)(y)
<b>As 2</b>							
Armdeel 1 basisplaat	1050	240	240			nvt	
Nema 23 50/1	1800	210	210			nvt	
Ketting wiel 06B-2 Z12	130	210	210			nvt	
Armdeel 1 verstevigingsprofiel	650	240	240			nvt	
Ketting	535.4	338.38	338.38			nvt	
Lager 2x 30204-J2	240	480	480			nvt	
Ketting wiel 06B-2 Z14	270	480	480			nvt	
<b>As 3</b>							
Armdeel 1 flens met armdeel 2	165	480	480			nvt	
Armdeel 1 flens lager	120	480	480			nvt	
<b>As 3</b>							
Flens verbinden as J3 met volgend armdeel	70	480	480	0		nvt	
Armdeel 2 flens met armdeel 1	190	480	480	0		nvt	
Nema 17 10/1	880	480	480	0		nvt	
Armdeel 2 - flens motor	155	510	510	30		nvt	
Armdeel 2 flens motor	65	525	525	45		nvt	
Verstevigingsplaatjes	45	525	525	45		nvt	
Armdeel 2 plooistuk	215	525	525	45		nvt	
Lager 2x 30204-J2	240	530	530	50		nvt	
Armdeel 2 flens met armdeel 3	95	545.9	545.9	65.9		nvt	
<b>As 4</b>							
As 4	135	555	555	75		nvt	
<b>As 4</b>							

Armdeel 3 flens met armdeel 2	250	565	565	85		0	0
Flens verbinden as met volgend armdeel	70	575	575	95		0	0
Verstevigingsplaatjes	45	580	580	100		42.5	42.5
Nema 17 15/1	680	689.25	689.25	209.25		35	35
Motor mount nema 17 met gearbox	65	689.25	689.25	209.25		40	40
Tandriemschijf 30-3M-9	50	689.25	689.25	209.25		91.75	91.75
Armdeel 3 plooistuk	640	702	702	222		76.75	76.75
Armdeel 3 verstevigingsprofiel	410	702	702	222		95.25	95.25
Tandriem	100	798.75	798.75	318.75		90	90
Lager 2x 30204-J2	240	930.75	930.75	450.75		75	75
Tandriemschijf 30-3M-9	100	930.75	930.75	450.75		98	98
Armdeel 3 flens met lager	125	930.75	930.75	450.75		80	80
Armdeel 3 flens met armdeel 4	165	930.75	930.75	450.75		70	70
As 5	160	930.75	930.75	450.75		65	65
<b>As 5</b>							
Armdeel 4 flens met armdeel 3	210	930.75	930.75	450.75	0	30	30
Flens verbinden as met volgend armdeel	70	930.75	930.75	450.75	0	25	25
Nema 14 5/1	400	950	950	470	19.25	0	0
Armdeel 4 plooistuk	175	975	975	495	44.25	0	0
<b>As 6</b>							
Maximale kracht	2000	998.75	998.75	518.75	68	0	0.00

## Bijlage C BEREKENING BOUTEN & SPIEËN

### 1) As 2 - Motor 2

Tabel C-1 Gegevens berekening schuifsterkte as 2

Overgebracht koppel $T = 108 \text{ Nm}$			
Boutverbinding		Spie	
Bouttype	M8	B	6 mm
Aantal	3	$L_{\text{eff}}$	14 mm
Kerndiameter	6 mm	$\varnothing_{\text{as}}$	20 mm
Buitendiameter	6.9 mm	$t_1$	2.8 mm
Afstand C-C	8.5 mm		
$T_{\text{toel,bout}}$	350 N/mm <sup>2</sup>	$T_{\text{toel,kettingwiel}}$	350 N/mm <sup>2</sup>
Berekende waarden			
$T_{\text{max,bout}}$	194.43 N/mm <sup>2</sup>	$T_{\text{kettingwiel}}$	275.51 N/mm <sup>2</sup>

Aangezien de tandriem bij deze as vervangen is door een stalen kettingwiel, wordt de berekening hier uitgevoerd met de treksterkte van C45-staal (700N/mm<sup>2</sup>) in plaats van aluminium.

De maximaal toegelaten waarden worden niet overschreden.

### 2) As 3 - Motor 3

Tabel C-2 Gegevens berekening schuifsterkte as 3

Overgebracht koppel $T = 31 \text{ Nm}$			
Boutverbinding		Spie	
Bouttype	M5	B	5 mm
Aantal	4	$L_{\text{eff}}$	13 mm
Kerndiameter	4 mm	$\varnothing_{\text{as}}$	14 mm
Buitendiameter	4.6 mm	$t_1$	2.3 mm
Afstand C-C	6 mm		
$T_{\text{toel,bout}}$	350 N/mm <sup>2</sup>	$T_{\text{toel,kettingwiel}}$	350 N/mm <sup>2</sup>
Berekende waarden			
$T_{\text{max,bout}}$	88.95 N/mm <sup>2</sup>	$T_{\text{kettingwiel}}$	148.11 N/mm <sup>2</sup>

Ook hier werd de aluminium tandriemschijf vervangen door een stalen kettingwiel.

De maximaal toegelaten waarden worden niet overschreden.

### 3) As 4 - Motor 4

Tabel C-3 Gegevens berekening schuifsterkte as 4

Overgebracht koppel T = 3 Nm			
Boutverbinding		Spie	
Bouttype	M4	B	3 mm
Aantal	2	L <sub>eff</sub>	13 mm
Kerndiameter	2.5 mm	∅ <sub>as</sub>	8 mm
Buitendiameter	2.9 mm	t <sub>1</sub>	1.4 mm
Afstand C-C	6 mm		
T <sub>toel,bout</sub>	350 N/mm <sup>2</sup>	T <sub>toel,tandriemschijf</sub>	270 N/mm <sup>2</sup>
Berekende waarden			
T <sub>max,bout</sub>	43.66 N/mm <sup>2</sup>	T <sub>tandriemschijf</sub>	41.21 N/mm <sup>2</sup>

De maximaal toegelaten waarden worden niet overschreden.

### 4) As 5 - Motor 5

Tabel C-4 Gegevens berekening schuifsterkte as 5

Overgebracht koppel T = 2 Nm			
Boutverbinding		Spie	
Bouttype	M4	B	3 mm
Aantal	2	L <sub>eff</sub>	13 mm
Kerndiameter	2.5 mm	∅ <sub>as</sub>	8 mm
Buitendiameter	2.9 mm	t <sub>1</sub>	1.4 mm
Afstand C-C	6 mm		
T <sub>toel,bout</sub>	350 N/mm <sup>2</sup>	T <sub>toel,tandriemschijf</sub>	270 N/mm <sup>2</sup>
Berekende waarden			
T <sub>max,bout</sub>	29.11 N/mm <sup>2</sup>	T <sub>tandriemschijf</sub>	27.47 N/mm <sup>2</sup>

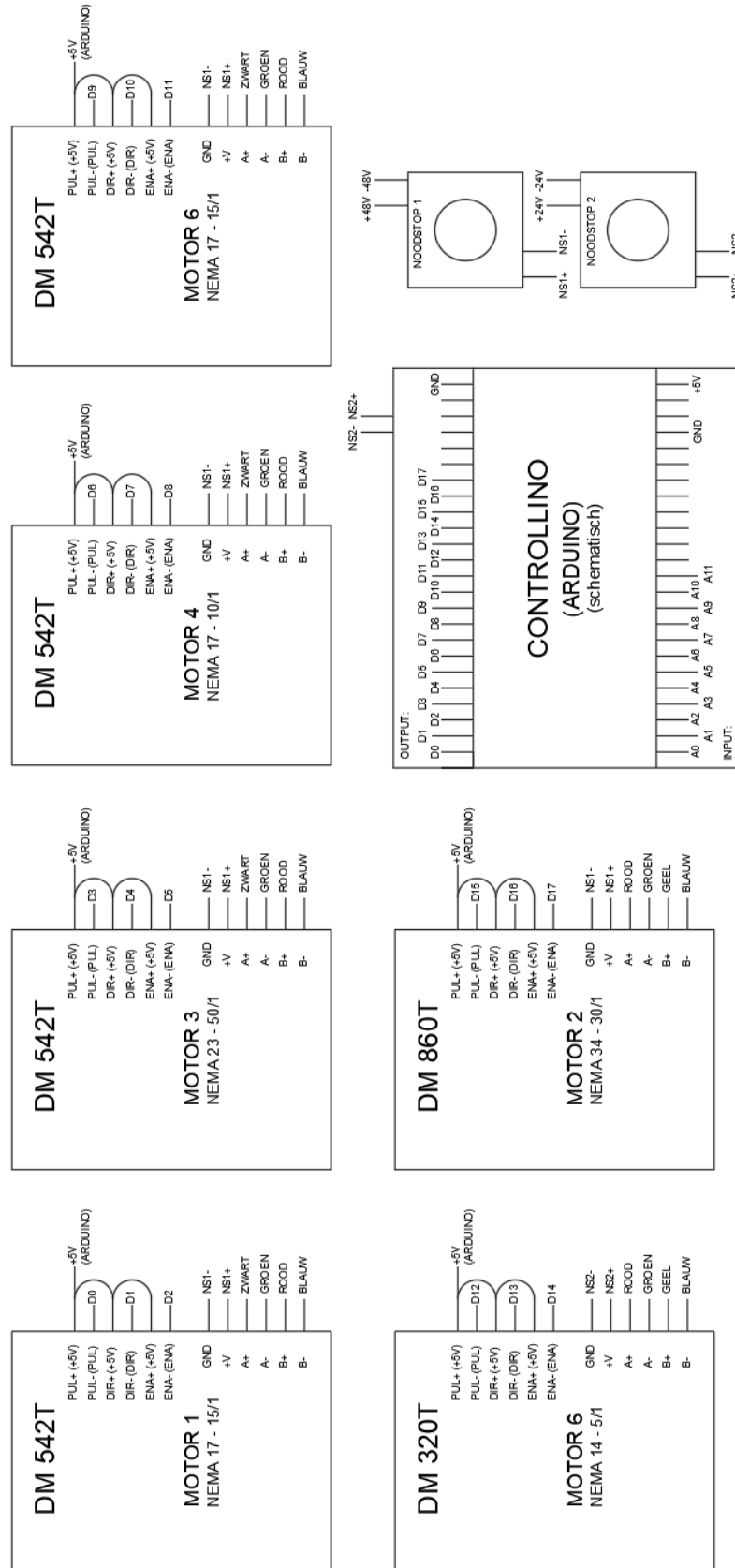
De maximaal toegelaten waarden worden niet overschreden.

### 5) As 6 - Motor 6

Aan de zesde as is nog geen werktuig bevestigd. De berekening is hier dan ook overbodig.



# Bijlage D ELEKTRISCH SCHEMA



# Bijlage E MATLAB CODE

## 1) Voorwaartse kinematica: positiebepaling (symbolisch)

```
function [ ] = Voorwaartse_kinematica_MATLAB_SYMBOLIC( )

% DEFINIËREN DENAVIT HARTENBERG PARAMETERS
r1 = 75; d1 = 278; alpha1 = -sym(pi)/2;
r2 = 450; d2 = 87.25; alpha2 = 0;
r3 = 0; d3 = -99; alpha3 = sym(pi)/2;
r4 = 0; d4 = -450.75; alpha4 = -sym(pi)/2;
r5 = 0; d5 = 0; alpha5 = sym(pi)/2;
r6 = 0; d6 = 0; alpha6 = sym(pi);
r7 = 0; d7 = 68; alpha7 = 0;

% DEFINIËREN SYMBOLISCHE HOEKEN
theta1 = sym('theta1');
theta2 = sym('theta2');
theta3ster = sym('theta3ster');
theta4 = sym('theta4');
theta5 = sym('theta5');
theta6ster = sym('theta6ster');

% DH TRANSFORMATIE FUNCTIE
function T = DHTRANSFORMATIE(ri, alphai, di, thetai)
T = [ cos(thetai), -1.*sin(thetai).*cos(alphai), sin(thetai).*sin(alphai),
ri.*cos(thetai) ;
sin(thetai), cos(thetai).*cos(alphai), -1.*cos(thetai).*sin(alphai),
ri.*sin(thetai);
0, sin(alphai), cos(alphai), di;
0, 0, 0, 1 ];
end

% DE 'DHTRANSFORMATIE'-FUNCTIE OPROEPEN OM DE VERSCHILLENDE
% TRANSFORMATIEMATRICES TE BEREKENEN

T10 = DHTRANSFORMATIE(r1,alpha1,d1,theta1)
T21 = DHTRANSFORMATIE(r2,alpha2,d2,theta2)
T32 = DHTRANSFORMATIE(r3,alpha3,d3,theta3ster)
T43 = DHTRANSFORMATIE(r4,alpha4,d4,theta4)
T54 = DHTRANSFORMATIE(r5,alpha5,d5,theta5)
T65 = DHTRANSFORMATIE(r6,alpha6,d6,theta6ster)

% BEREKENING VAN DE VOORWAARTSE KINEMATICA TOT AAN HET POLSGEWICHT (TCP
% ZONDER OFFSET)
T20 = T10*T21;
T20s = simplify(T20)
T30 = T20s*T32;
T30s = simplify(T30)
T40 = T30s*T43;
T40s = simplify(T40)
T50 = T40s*T54;
T50s = simplify(T50)
T60 = T50s*T65;
T60s = simplify(T60)

end
```

## 2) Voorwaartse kinematica positiebepaling en Eulerhoeken (getalwaarden)

```
function [ ] = Voorwaartse_kinematica_MATLAB( )

% DEFINIËREN DENAVIT HARTENBERG PARAMETERS

r1 = 75; d1 = 278; alpha1 = -pi/2;
r2 = 480; d2 = 87.25; alpha2 = 0;
r3 = 0; d3 = -99; alpha3 = pi/2;
r4 = 0; d4 = -450.75; alpha4 = -pi/2;
r5 = 0; d5 = 0; alpha5 = pi/2;
r6 = 0; d6 = 0; alpha6 = pi;
r7 = 0; d7 = 68; alpha7 = 0;

% GETALWAARDEN INGEVEN
theta1 = input ('GEEF DE WAARDE VAN THETA 1 (GRADEN) = ');
theta2 = input ('GEEF DE WAARDE VAN THETA 2 (GRADEN) = ');
theta3 = input ('GEEF DE WAARDE VAN THETA 3 (GRADEN) = ');
theta4 = input ('GEEF DE WAARDE VAN THETA 4 (GRADEN) = ');
theta5 = input ('GEEF DE WAARDE VAN THETA 5 (GRADEN) = ');
theta6 = input ('GEEF DE WAARDE VAN THETA 6 (GRADEN) = ');

% DE 'DHTRANSFORMATIE'-FUNCTIE OPROEPEN OM DE VERSCHILLENDE
% TRANSFORMATIEMATRICES TE BEREKENEN

T10 = DHTRANSFORMATIE(r1,alpha1,d1,theta1*pi/180)
T21 = DHTRANSFORMATIE(r2,alpha2,d2,theta2*pi/180)
T32 = DHTRANSFORMATIE(r3,alpha3,d3,(theta3-(90))*pi/180)
T43 = DHTRANSFORMATIE(r4,alpha4,d4,theta4*pi/180)
T54 = DHTRANSFORMATIE(r5,alpha5,d5,theta5*pi/180)
T65 = DHTRANSFORMATIE(r6,alpha6,d6,(theta6+(180))*pi/180)
T76 = DHTRANSFORMATIE(r7,alpha7,d7,0)

% DH TRANSFORMATIE FUNCTIE
function T = DHTRANSFORMATIE(ri,alphai,di,thetai)
T = [ cos(thetai), -1.*sin(thetai).*cos(alphai), sin(thetai).*sin(alphai),
ri.*cos(thetai) ;
sin(thetai), cos(thetai).*cos(alphai), -1.*cos(thetai).*sin(alphai),
ri.*sin(thetai);
0, sin(alphai), cos(alphai), di;
0, 0, 0, 1 ];
end

% BEREKENING VAN DE VOORWAARTSE KINEMATICA TOT AAN HET POLSGEWICHT (TCP
% ZONDER OFFSET)
T20 = T10*T21
T30 = T20*T32
T40 = T30*T43
T50 = T40*T54
T60 = T50*T65

% DE POSITIE VAN HET POLSGEWICHT (TCP ZONDER OFFSET)
Xtcp0 = T60(1,4);
Ytcp0 = T60(2,4);
Ztcp0 = T60(3,4);
POSTcp0 = [Xtcp0;Ytcp0;Ztcp0]

% DE POSITIE VAN DE EINDEFFECTOR (TCP MET OFFSET)
```

```

Ttcp = T60*[0;0;68;1];
Xtcp = Ttcp(1,1);
Ytcp = Ttcp(2,1);
Ztcp = Ttcp(3,1);
POSTcp = [Xtcp;Ytcp;Ztcp]

% VARIABELEN VAN DE ROTATIEMATRIX BINNEN T60
nx=T60(1,1);
ny=T60(2,1);
nz=T60(3,1);
ox=T60(1,2);
oy=T60(2,2);
oz=T60(3,2);
ax=T60(1,3);
ay=T60(2,3);
az=T60(3,3);
termsqrtbeta = ((ax^2+ay^2)^(1/2));

% BEPALEN VAN DE EULERHOEKEN
beta = atan2(termsqrtbeta,-az)*180/pi
alpha = atan2(nz/cos(beta),oz/cos(beta))*180/pi
gamma = atan2(-ax/cos(beta),ay/cos(beta))*180/pi

end

```

### 3) MatLab code berekenen totale rotatiematrix (symbolisch)

```

function [ ] = Rotatiematrix_MATLAB_SYMBOLIC( )

% DEFINIËREN SYMBOLISCHE HOEKEN
alpha = sym('alpha');
beta = sym('beta');
gamma = sym('gamma');

% DEFINIËREN BASISROTATIEMATRICES
Ryalpha = [1,0,0;
0,cos(alpha),-sin(alpha);
0, sin(alpha),cos(alpha)];

Rzbeta = [cos(beta),0,sin(beta);
0,1,0;
-sin(beta),0,cos(beta)];

Rxgamma = [cos(gamma),-sin(gamma),0;
sin(gamma),cos(gamma),0;
0,0,1];

Ryalpha
Rzbeta
Rxgamma

% BEREKENEN VAN DE TOTALE ROTATIEMATRIX
Rtot = Ryalpha*Rzbeta*Rxgamma

end

```

## 4) MatLab code berekening inverse oriëntatiekinematica

```
function [ ] = Inverse_kinematica_MATLAB( )

% DEFINIËREN DENAVIT HARTENBERG PARAMETERS

r1 = 75; d1 = 278; alpha1 = -pi/2;
r2 = 480; d2 = 87.25; alpha2 = 0;
r3 = 0; d3 = -99; alpha3 = pi/2;
r4 = 0; d4 = -450.75; alpha4 = -pi/2;
r5 = 0; d5 = 0; alpha5 = pi/2;
r6 = 0; d6 = 0; alpha6 = pi;
r7 = 0; d7 = 68; alpha7 = 0;

% GEEF DE GEWENSTE POSITIE IN VAN HET TCP
XTCP = 694.2790; %input ('X = '); %1=GELIJKE RICHTING,-1=TEGENGESTELDE
RICHTING,0=GEEN REKENING
YTCP = 331.7527; %input ('Y = ');
ZTCP = 332.6829; %input ('Z = ');

% GEEF DE GEWENSTE ORIËNTATIE VAN DE EINDEFFECTOR TEN OPZICHTE VAN HET
% BASISFRAME

RTCPBASIS = [0.0795, -0.6124,0.7866;
0.8624,-0.3536,-0.3624;
0.5,0.7071,0.5 ];

% BEREKENEN VAN DE POSITIE VAN DE WRIST
X = XTCP-68*RTCPBASIS(1,3)
Y = YTCP-68*RTCPBASIS(2,3)
Z = ZTCP-68*RTCPBASIS(3,3)

% BEREKENEN INVERSE POSITIEKINEMATICA
% Bepalen van kwadranten
    %Kwadranten volgens het zijaanzicht
    if (X>0) && (Z>0)
        kwZ = 1 ;
    elseif (X<0) && (Z>0)
        kwZ = 2 ;
    elseif (X<0) && (Z<0)
        kwZ = 3 ;
    elseif (X>0) && (Z<0)
        kwZ = 4 ;
    end
    %Kwadranten volgens het bovenaanzicht
    if (X>0) && (Y>0)
        kwB = 1 ;
    elseif (X<0) && (Y>0)
        kwB = 2 ;
    elseif (X<0) && (Y<0)
        kwB = 3 ;
    elseif (X>0) && (Y<0)
        kwB = 4 ;
    end

% Berekenen van de hoek thetal
if ((kwB==1) || (kwB==4))
    thetal = (atan(Y/X) + abs(atan(-11.75/((X^2+Y^2)^(1/2)))))*180/pi;
```

```

else
    theta1 = (atan(Y/X) - abs(atan(-11.75/((X^2+Y^2)^(1/2)))))*180/pi;

end

% Berekenen van de hoek theta3
if ((kwZ==1) || (kwZ==4))
    R = ((X^2+Y^2)^(1/2))-75 ;
else
    R = ((X^2+Y^2)^(1/2))+75;

end
D=((R^2+(Z-278)^2)^(1/2));
costheta3=(D^2-480^2-450.75^2)/(2*480*450.75);
sintheta3= ((1-costheta3^2)^(1/2));
theta3 = atan(sintheta3/costheta3)*180/pi;

% Berekenen van de hoek theta2
alfa = atan((Z-278)/R);
beta = atan((450.75*sintheta3)/(480+450.75*costheta3));
if ((kwZ==1) || (kwZ==4))
    theta2 = (-alfa-beta)*180/pi;
else
    theta2 = (-alfa+beta)*180/pi;

end

% WEERGEVEN VAN DE RESULTATEN
if(kwZ == 2) || (theta1 < -90) || (theta1 > 90) || (theta2 < -90) || (theta2 >
90) || (theta3 < 0) || (theta3 > 125) ('Dit punt in de ruimte valt buiten het
werkingsgebied')
else
    theta1
    theta2
    theta3

end

% ORIËNTATIEMATRIX R30

R10 = RTRANSFORMATIE(r1,alpha1,d1,theta1*pi/180);
R21 = RTRANSFORMATIE(r2,alpha2,d2,theta2*pi/180);
R32 = RTRANSFORMATIE(r3,alpha3,d3,(theta3-(90))*pi/180);

R20 = R10*R21;

R30 = R10*R21*R32;
transpose(R30);

% BEREKENEN INVERSE ORIËNTATIEKINEMATICA

R63 = (transpose(R30)*RTCPBASIS)

% VARIABELEN VAN DE ROTATIEMATRIX BINNEN T60
nx=R63(1,1);
ny=R63(2,1);
nz=R63(3,1);

```

```

ox=R63(1,2);
oy=R63(2,2);
oz=R63(3,2);
ax=R63(1,3);
ay=R63(2,3);
az=R63(3,3);
termsqrtTHETA5 = ((ax^2+ay^2)^(1/2));

% BEPALEN VAN DE EULERHOEKEN

termsqrtTHETA5 = ((1-az^2)^(1/2));
THETA5 = (atan2(-termsqrtTHETA5,-az)*180/pi);
THETA4 = atan2(ay,ax)*180/pi;
THETA5;
THETA6 = atan2(-oz,-nz)*180/pi;

if (abs(THETA4)<90) || (abs(THETA5)<135) || (abs(THETA6)<90)
THETA5 = (atan2(-termsqrtTHETA5,-az)*180/pi);
THETA4 = atan2(ay,ax)*180/pi
THETA5
THETA6 = atan2(-oz,nz)*180/pi

else

THETA5 = (atan2(termsqrtTHETA5,-az)*180/pi);
THETA4 = atan2(-ay,ax)*180/pi
THETA5
THETA6 = atan2(oz,nz)*180/pi
end

% R TRANSFORMATIE FUNCTIE
function R = RTRANSFORMATIE(ri,alphai,di,thetai)
R = [ cos(thetai), -1.*sin(thetai).*cos(alphai), sin(thetai).*sin(alphai);
sin(thetai), cos(thetai).*cos(alphai), -1.*cos(thetai).*sin(alphai);
0, sin(alphai), cos(alphai) ];
end
end

```

# Bijlage F EXTRA VOORBEELDEN INVERSE POSITIEKINEMATICA

## 1) Controle van $\theta_1$ in kwadrant 4B

In het numerieke voorbeeld in paragraaf 7.3.7 is hoek  $\theta_1$  gelijk aan  $30^\circ$ , hierdoor ligt de pols in het kwadrant 1B. Om na te gaan als de werkwijze klopt wordt de hoek  $\theta_1$  nu op  $-30^\circ$  gekozen en in de voorwaartse kinematica geplaatst.

In tabel F-1 zijn de ingevulde hoeken weergegeven alsook de polscoördinaten die bekomen worden via de voorwaartse kinematica.

Tabel F-1 Getalwaarden controle hoek  $\theta_1$  in kwadrant 4B

	Hoeken ( $^\circ$ )		Coördinaten (mm)
$\theta_1$	-30	X	629.0425
$\theta_2$	-45	Y	-376.7456
$\theta_3$	90 ( - 90)	Z	298.6829
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	Kwadranten	4B - 1Z

	$\theta_1 = -30.0001^\circ \cong -30^\circ$	
--	---	--

De berekende hoek is dezelfde als die ingegeven in de voorwaartse kinematica. De vooropgestelde werkwijze komt zo in beide kwadranten correct uit.

## 2) Controle verschillende oriëntaties met wijzigende $\theta_2$ en $\theta_1 = 30^\circ$

Hieronder een aantal configuraties ter bevestiging van de werkwijzen beschreven in 7.3.5.

### Configuratie 1

Tabel F-2 Configuratie 1:  $\theta_2 = -90^\circ$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken ( $^\circ$ )		Coördinaten (mm)
$\theta_1$	30	X	461.1879
$\theta_2$	-90	Y	252.6992
$\theta_3$	90 ( - 90)	Z	758.0000
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	Kwadranten	1B - 1Z



	$\theta_1 = 29.9997^\circ \cong 30^\circ$	
	$R = 450.8813$	
	$D = 658.5544$	
	$\cos(\theta_3) = 0.0002736$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9843^\circ \cong 90^\circ$	
	$\theta_2 = -(\alpha + \beta) = -89.9916^\circ \cong -90^\circ$	
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = 46.7917^\circ$	
	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right) = \text{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	

### Configuratie 3

Tabel F-3 Configuratie 3:  $\theta_2 \in [0^\circ; 90^\circ]$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken (°)		Coördinaten (mm)
$\theta_1$	30	X	88.7388
$\theta_2$	45	Y	37.6656
$\theta_3$	90 (- 90)	Z	-380.1396
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 (+ 180)	Kwadranten	1B - 4Z

	$\theta_1 = 29.9483^\circ \cong 30^\circ$	
	$R = 21.4016$	
	$D = 658.4875$	
	$\cos(\theta_3) = 0.00006979$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9960^\circ \cong 90^\circ$	
	$\theta_2 = -(\alpha + \beta) = -(-44.9376)^\circ \cong 45^\circ$	
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = -88.1375^\circ$	
	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right) = \text{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	

#### Configuratie 4

Tabel F-4 Configuratie 4:  $\theta_2 = 90^\circ$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken (°)		Coördinaten (mm)
$\theta_1$	30	X	-319.5340
$\theta_2$	90	Y	-198.0508
$\theta_3$	90 ( - 90)	Z	-202.0000
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	Kwadranten	3B - 3Z

	$\theta_1 = 30.0009^\circ \cong 30^\circ$	
	$R = 450.9336$	
	$D = 658.5903$	
	$\cos(\theta_3) = 0.0003826$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9781^\circ \cong 90^\circ$	
	$\theta_2 = (-\alpha + \beta) = 89.9883^\circ \cong 90^\circ$	
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = -46.7884^\circ$	
	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right) = \text{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	

Bij elk van de drie configuraties komt de hoek  $\theta_3$  correct uit, wat de methode bevestigt.

### 3) Controle verschillende oriëntaties met wijzigende $\theta_2$ en $\theta_1 = -30^\circ$

De configuraties uit de vorige paragraaf worden opnieuw beschouwd, maar nu met  $\theta_1 = -30^\circ$ . De configuraties worden gelijk benoemd, maar worden voorzien van een accent.

#### Configuratie 1'

Tabel F-5 Configuratie 1:  $\theta_2 = -90^\circ$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken (°)		Coördinaten (mm)
$\theta_1$	-30	X	449.4378
$\theta_2$	-90	Y	-273.0508
$\theta_3$	90 ( - 90)	Z	758.0000
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	Kwadranten	4B - 1Z

	$\theta_1 = -30.0003^\circ \cong -30^\circ$	
	$R = 450.8812$	
	$D = 658.5544$	
	$\cos(\theta_3) = 0.0002736$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9843^\circ \cong 90^\circ$	
	$\theta_2 = -(\alpha + \beta) = -89.9916^\circ \cong -90^\circ$	
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = 46.7917^\circ$	
	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right) = \text{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	

#### Configuratie 2'

Tabel F-6 Configuratie 1:  $\theta_2 = -90^\circ$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken (°)		Coördinaten (mm)
$\theta_1$	-30	X	629.0425
$\theta_2$	-45	Y	-376.7456
$\theta_3$	90 ( - 90)	Z	298.6829
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	Kwadranten	4B - 1Z

	$\theta_1 = -30.0001^\circ \cong -30^\circ$	
	$R = 658.2337$	
	$D = 658.5586$	
	$\cos(\theta_3) = 0.0002863$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9836^\circ \cong 90^\circ$	
	$\theta_2 = -(\alpha + \beta) = -44.9996^\circ \cong -45^\circ$	
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = 1.7797^\circ$	

	$\beta = \operatorname{atan}\left(\frac{L3'}{L2 + L3''}\right) = \operatorname{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	
--	---	--

Configuratie 3'

Tabel F-7 Configuratie 3:  $\theta_2 \in ]0^\circ;90^\circ]$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken (°)		Coördinaten (mm)
$\theta_1$	-30	<b>X</b>	76.9888
$\theta_2$	45	<b>Y</b>	-58.0172
$\theta_3$	90 ( - 90)	<b>Z</b>	-380.1396
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	<b>Kwadranten</b>	4B - 4Z

	$\theta_1 = -30.05167^\circ \cong -30^\circ$	
	$R = 21.4016$	
	$D = 658.4875$	
	$\cos(\theta_3) = 0.00006979$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9960^\circ \cong 90^\circ$	
	$\theta_2 = -(\alpha + \beta) = 44.9376^\circ \cong 45^\circ$	
	$\alpha = \operatorname{atan}\left(\frac{\Delta Z}{R}\right) = -88.1375^\circ$	
	$\beta = \operatorname{atan}\left(\frac{L3'}{L2 + L3''}\right) = \operatorname{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	

Configuratie 4'

Tabel F-8 Configuratie 4:  $\theta_2 = 90^\circ$  - Ingegeven hoeken + Coördinaten pols bekomen uit FK

	Hoeken (°)		Coördinaten (mm)
$\theta_1$	-30	<b>X</b>	-331.2840
$\theta_2$	90	<b>Y</b>	177.6992
$\theta_3$	90 ( - 90)	<b>Z</b>	-202.0000
$\theta_4$	-45		
$\theta_5$	-90		
$\theta_6$	-90 ( + 180)	<b>Kwadranten</b>	2B - 3Z

	$\theta_1 = -29.9434^\circ \cong -30^\circ$	
	$R = 450.9336$	
	$D = 658.5903$	
	$\cos(\theta_3) = 0.0003826$	
	$\sin(\theta_3) = \sqrt{1 - \cos^2(\theta_3)} = 0.9999$	
	$\theta_3 = 89.9781^\circ \cong 90^\circ$	
	$\theta_2 = (-\alpha + \beta) = 89.9883^\circ \cong 90^\circ$	
	$\alpha = \text{atan}\left(\frac{\Delta Z}{R}\right) = -46.7884^\circ$	
	$\beta = \text{atan}\left(\frac{L3'}{L2 + L3''}\right) = \text{atan}\left(\frac{450.75 \sin(90^\circ)}{480 + 450.75 \cos(90^\circ)}\right) = 43.1999^\circ$	

Ook hier is bij elk van de configuraties de hoek  $\theta_3$  correct, wat de methode bevestigt.

## Bijlage G ARDUINO TESTPROGRAMMA

```
// Definiëren pinnen (In de testfase werd een Arduino MEGA gebruikt)
const int stepPin1 = 22;
const int dirPin1 = 23;
const int enPin1 = 24;

const int stepPin2 = 25;
const int dirPin2 = 26;
const int enPin2 = 27;

const int stepPin3 = 28;
const int dirPin3 = 29;
const int enPin3 = 30;

const int stepPin4 = 31;
const int dirPin4 = 32;
const int enPin4 = 33;

const int stepPin5 = 34;
const int dirPin5 = 35;
const int enPin5 = 36;

const int stepPin6 = 37;
const int dirPin6 = 38;
const int enPin6 = 39;

// Definiëren pinnen joysticks
// Motoren 1&2
const int joy1Pin1 = 0; // "X-richting: Waarde tussen 0 en 1024, center 512"
const int joy1Pin2 = 1; // "Y-richting: Waarde tussen 0 en 1024, center 512"
const int joy1Pin3 = 2; // "knop - monostabiel"

// Motoren 3&4
const int joy2Pin1 = 2;
const int joy2Pin2 = 3;
const int joy2Pin3 = 3;

// Motoren 5&6
const int joy3Pin1 = 4;
const int joy3Pin2 = 5;
const int joy3Pin3 = 4;

// Overbrengingsverhouding van de verschillende motoren (gearbox +
ketting/riemoverbrenging
float gearratioM1 = 18.75;
```

```

float gearratioM2 = 37.5;
float gearratioM3 = 70.83;
float gearratioM4 = 10.0;
float gearratioM5 = 15.0;
float gearratioM6 = 5.0;

//Stappen per toer ingesteld op de driver
int driverRevM1 = 800 ;
int driverRevM2 = 8000 ;
int driverRevM3 = 400 ;
int driverRevM4 = 800 ;
int driverRevM5 = 400 ;
int driverRevM6 = 800 ;

float stepsperRevM1 = gearratioM1 * driverRevM1;
float stepsperRevM2 = gearratioM1 * driverRevM2;
float stepsperRevM3 = gearratioM1 * driverRevM3;
float stepsperRevM4 = gearratioM1 * driverRevM4;
float stepsperRevM5 = gearratioM1 * driverRevM5;
float stepsperRevM6 = gearratioM1 * driverRevM6;

// Definiëren van de hoeken
float theta1;
float theta2;
float theta3;
float theta4;
float theta5;
float theta6;

// Opstellen counters voor de verschillende motoren
int counterM1 = 0;
int counterM2 = 0;
int counterM3 = 0;
int counterM4 = 0;
int counterM5 = 0;
int counterM6 = 0;

char keuze = '0';

// Initialiseren van de pins
void setup() {

// Initialiseren seriële monitor
Serial.begin(9600);

//Instellen pins motor 1
pinMode(stepPin1, OUTPUT);

```

```
pinMode(dirPin1, OUTPUT);
pinMode(enPin1, OUTPUT);
digitalWrite(enPin1, HIGH);
delay(200);
digitalWrite(dirPin1, HIGH); //Stel de dirPin hoog voor een positieve verdraaiing, laag voor
een negatieve verdraaiing (volgens kinematisch diagram) -> Dient gecontroleerd te worden!
delay(200);
```

```
//Instellen pins motor 2
pinMode(stepPin2, OUTPUT);
pinMode(dirPin2, OUTPUT);
pinMode(enPin2, OUTPUT);
digitalWrite(enPin2, HIGH);
delay(200);
digitalWrite(dirPin2, HIGH);
delay(200);
```

```
//Instellen pins motor 3
pinMode(stepPin3, OUTPUT);
pinMode(dirPin3, OUTPUT);
pinMode(enPin3, OUTPUT);
digitalWrite(enPin3, HIGH);
delay(200);
digitalWrite(dirPin3, HIGH);
delay(200);
```

```
//Instellen pins motor 4
pinMode(stepPin4, OUTPUT);
pinMode(dirPin4, OUTPUT);
pinMode(enPin4, OUTPUT);
digitalWrite(enPin4, HIGH);
delay(200);
digitalWrite(dirPin4, HIGH);
delay(200);
```

```
//Instellen pins motor 5
pinMode(stepPin5, OUTPUT);
pinMode(dirPin5, OUTPUT);
pinMode(enPin5, OUTPUT);
digitalWrite(enPin5, HIGH);
delay(200);
digitalWrite(dirPin5, HIGH);
delay(200);
```

```
//Instellen pins motor 6
pinMode(stepPin6, OUTPUT);
pinMode(dirPin6, OUTPUT);
pinMode(enPin6, OUTPUT);
```



```

digitalWrite(enPin6, HIGH);
delay(200);
digitalWrite(dirPin6, HIGH);
delay(200);

//Instellen pins joystick 1
pinMode(joy1Pin1, INPUT);
pinMode(joy1Pin2, INPUT);
digitalWrite(joy1Pin3, HIGH);

//Instellen pins joystick 2
pinMode(joy2Pin1, INPUT);
pinMode(joy2Pin2, INPUT);
digitalWrite(joy2Pin3, HIGH);

//Instellen pins joystick 3
pinMode(joy3Pin1, INPUT);
pinMode(joy3Pin2, INPUT);
digitalWrite(joy3Pin3, HIGH);

while (! Serial);
}
void loop() {
// Hoofdprogramma: hierin kan de gebruiker kiezen uit een aantal methodes

counterM1;
counterM2;
counterM3;
counterM4;
counterM5;
counterM6;

Serial.println("Kies modus: 1: homing - 2: homing synchroon - 3: homing voor
herhaalbaarheidstest - 4: test herhaalbaarheid - 5: Bediening met joysticks - 6: positie zoals
in rekenvoorbeeld - 7: Inverse kinematica 5-assig - 8: Inverse kinematica 6-assig");
while (keuze == '0') {
if (Serial.available() > 0) {
keuze = Serial.read();
Serial.println(keuze);
// Keuze 1-4 = sturen met pulsen - Keuze 5-7 sturen met hoeken, pulsen daarruit
berekend.
if (keuze == '1') {
homingApart();
}
else if (keuze == '2') {
homingSynchroon();
}
else if (keuze == '3') {

```

```

    homingTestHerhaalbaarheid();
}
else if (keuze == '4') {
    testHerhaalbaarheid();
}

// Onderstaande methodes zijn reeds geprogrammeerd, maar dienen nog
geïmplementeerd te worden.
else if (keuze == '5') {
    bedieningJoysticks();
}
else if (keuze == '6') {
    positieAlsInVoorbeeld();
}
else if (keuze == '7') {
    Serial.println("Geef de gewenste X-waarde");
    float x = Serial.read();
    Serial.println("Geef de gewenste Y-waarde");
    float y = Serial.read();
    Serial.println("Geef de gewenste Z-waarde");
    float z = Serial.read();
    Serial.println("Geef de gewenste gripper Hoek");
    float gripperHoek = Serial.read();
    invKin5(x, y, z, gripperHoek, counterM1, counterM2, counterM3, counterM4, counterM5,
counterM6);
}
else if (keuze == '8') {
    Serial.println("Hier dient later de zes-assige inverse kinematica in uitgewerkt te
worden.");
}

else {
    Serial.println("Foutieve keuze");
}
}
keuze = '0';
}

void homingApart() {
// In deze methode worden de verschillende armdelen apart aangestuurd zodat de robotarm
naar een bepaalde positie gaat. Hiervoor worden enkel motoren twee en drie gebruikt.

//stap1
digitalWrite(dirPin2, HIGH);
for (int x = 0; x < 7000; x++) {
    digitalWrite(stepPin2, HIGH);

```

```

    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
}
delay(100);

//stap2
digitalWrite(dirPin3, LOW);
for (int x = 0; x < 1000; x++) {

    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}
delay(100);

//stap3
digitalWrite(dirPin2, HIGH);
for (int x = 0; x < 5000; x++) {
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
}
delay(100);

//stap4
digitalWrite(dirPin3, LOW);
for (int x = 0; x < 1000; x++) {

    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}
delay(100);

//stap5
digitalWrite(dirPin2, HIGH);
for (int x = 0; x < 2000; x++) {
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
}
delay(100);

```

```
//stap6
digitalWrite(dirPin3, LOW);
for (int x = 0; x < 1000; x++) {

    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}
delay(100);
```

```
//stap7
digitalWrite(dirPin2, HIGH);
for (int x = 0; x < 2000; x++) {
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
}
delay(100);
```

```
//stap8
digitalWrite(dirPin3, LOW);
for (int x = 0; x < 1000; x++) {

    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}
delay(100);
```

```
//stap9
digitalWrite(dirPin2, HIGH);
for (int x = 0; x < 9000; x++) {
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
}
delay(100);
```

```
//stap10
digitalWrite(dirPin3, LOW);
for (int x = 0; x < 2000; x++) {

    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
```

```

    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}
delay(100);

Serial.print("Einde homing apart");
}

```

```

void homingSynchroon() {

```

// In deze methode worden de verschillende armdelen synchroon aangestuurd om naar een bepaalde positie te gaan. Hiervoor worden motoren twee, drie en vijf gebruikt.

//Eerst de volledige arm opheffen zodat deze niet over de tafel sleept

```

digitalWrite(dirPin2, HIGH);
for (int x = 0; x < 7000; x++) {
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);

```

```

}
delay(1000);

```

//Opsplitsen van de bewegingen in 5000 stappen -> Door de verschillende motoren in de lus te zetten zullen deze telkens één stap uitvoeren. Door een digitalWrite(stepPin) van een bepaalde motor extra toe te voegen, zal deze motor meer stappen uitvoeren.

```

for (int s = 0; s < 5000; s++)
{

```

//M2

```

digitalWrite(dirPin2, HIGH);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);

```

```

//M3
digitalWrite(dirPin3, LOW);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);

//M5
digitalWrite(dirPin5, LOW);
digitalWrite(stepPin5, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin5, LOW);
delayMicroseconds(200);
}

Serial.print("Einde homing synchroon");
}

void homingTestHerhaalbaarheid() {
  // TE GEBRUIKEN NA HOMINGSYNCHROON

  //Initiële homing voor de herhaalbaarheidstest

  digitalWrite(dirPin2, LOW);
  digitalWrite(dirPin5, LOW);/*RECHTSOM DRAAIEN*/
  for (int x = 0; x < 11500; x++) {
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);

  }
  for (int x = 0; x < 900; x++) {
    digitalWrite(stepPin5, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin5, LOW);
    delayMicroseconds(200);

  }

  delay(1000);

  //Eénmalig uitvoeren van de lus

  for (int i = 0; i < 1; i++)
  {
    delay(2000); // One second delay
  }
}

```

```

for (int f = 0; f < 1000; f++)
{

    //M2
    digitalWrite(dirPin2, LOW);
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);

    //M3
    digitalWrite(dirPin3, HIGH);
    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);

    //M5
    digitalWrite(dirPin5, HIGH);
    digitalWrite(stepPin5, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin5, LOW);
    delayMicroseconds(200);
}
delay(1000);

```

```

for (int f = 0; f < 1000; f++)
{

    //M2
    digitalWrite(dirPin2, HIGH);
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);

```

```

digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);

//M3
digitalWrite(dirPin3, LOW);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);

//M5
digitalWrite(dirPin5, LOW);
digitalWrite(stepPin5, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin5, LOW);
delayMicroseconds(200);
}
delay(1000);
}

Serial.print("End homing voor test herhaalbaarheid");
}

void testHerhaalbaarheid() {
// TE GEBRUIKEN NA HOMINGTESTHERHAALBAARHEID
// Vijf maal uitvoeren van de lus
for (int i = 0; i < 5; i++)
{
delay(2000);

Serial.println("LINKSOM DRAAIEN");
for (int f = 0; f < 1000; f++)
{

//M2 Linksom
digitalWrite(dirPin2, LOW);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);

```



```

digitalWrite(stepPin2, LOW);
delayMicroseconds(200);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);

//M3 Linksom
digitalWrite(dirPin3, HIGH);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);

//M5
digitalWrite(dirPin5, HIGH);
digitalWrite(stepPin5, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin5, LOW);
delayMicroseconds(200);
}
delay(1000);

Serial.println("RECHTSOM DRAAIEN");

for (int f = 0; f < 1000; f++)
{
//M2 Linksom
digitalWrite(dirPin2, HIGH);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);

//M3 Linksom
digitalWrite(dirPin3, LOW);

```

```

digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);
digitalWrite(stepPin3, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin3, LOW);
delayMicroseconds(200);

//M5
digitalWrite(dirPin5, LOW);
digitalWrite(stepPin5, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin5, LOW);
delayMicroseconds(200);
}
delay(1000);
}

Serial.print("Einde test herhaalbaarheid");
}

void bedieningJoysticks() {
// In deze methode kunnen de verschillende motoren door middel van joysticks
aangestuurd worden.
int joystickbediening = 0;
int hoekverdraaiing = 5;
int waardeM1 = analogRead (joy1Pin1);
int waardeM2 = analogRead (joy1Pin2);
int waardeM3 = analogRead (joy2Pin1);
int waardeM4 = analogRead (joy2Pin2);
int waardeM5 = analogRead (joy3Pin1);
int waardeM6 = analogRead (joy3Pin2);

// Aantal stappen die gemaakt worden door te kijken hoeveel de joystick aangestuurd is.
Telkens aansturing per 5°.
float stappenDoorJoyM1 = (abs(512 - waardeM1) / 512) * stepsperRevM1 * (5 / 360);
float stappenDoorJoyM2 = (abs(512 - waardeM2) / 512) * stepsperRevM2 * (5 / 360);
float stappenDoorJoyM3 = (abs(512 - waardeM3) / 512) * stepsperRevM3 * (5 / 360);
float stappenDoorJoyM4 = (abs(512 - waardeM4) / 512) * stepsperRevM4 * (5 / 360);
float stappenDoorJoyM5 = (abs(512 - waardeM5) / 512) * stepsperRevM5 * (5 / 360);
float stappenDoorJoyM6 = (abs(512 - waardeM6) / 512) * stepsperRevM6 * (5 / 360);

while (joystickbediening == 0)

```

```

{
if (waardeM1 < 490 || waardeM1 > 530) // Marge inbouwen, middenstand is +- 512
{

if (waardeM1 < 512) // RECHTSOM DRAAIEN (volgens kinematische diagram)
{
for (int a = 0; a < stappenDoorJoyM1; a++)
{
digitalWrite(dirPin1, LOW);
digitalWrite(stepPin1, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin1, LOW);
delayMicroseconds(200);
}
}
if (waardeM1 > 512)
{
for (int a = 0; a < stappenDoorJoyM1; a++)
{
digitalWrite(dirPin1, HIGH); // LINKSOM DRAAIEN (volgens kinematische diagram)
digitalWrite(stepPin1, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin1, LOW);
delayMicroseconds(200);
}
}

}

if (waardeM2 < 490 || waardeM2 > 530)
{

if (waardeM2 < 512) // RECHTSOM DRAAIEN (volgens kinematische diagram)
{
for (int b = 0; b < stappenDoorJoyM2; b++)
{
digitalWrite(dirPin2, LOW);
digitalWrite(stepPin2, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin2, LOW);
delayMicroseconds(200);
}
}
if (waardeM2 > 512)
{
for (int b = 0; b < stappenDoorJoyM2; b++)
{
digitalWrite(dirPin2, HIGH); // LINKSOM DRAAIEN (volgens kinematische diagram)

```

```

    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
  }
}

}

if (waardeM3 < 490 || waardeM3 > 530)
{

if (waardeM3 < 512) // RECHTSOM DRAAIEN (volgens kinematische diagram)
{
  for (int c = 0; c < stappenDoorJoyM3; c++)
  {
    digitalWrite(dirPin3, LOW);
    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
  }
}
if (waardeM3 > 512)
{
  for (int c = 0; c < stappenDoorJoyM3; c++)
  {
    digitalWrite(dirPin3, HIGH); // LINKSOM DRAAIEN (volgens kinematische diagram)
    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
  }
}

}

if (waardeM4 < 490 || waardeM4 > 530)
{

if (waardeM4 < 512) // RECHTSOM DRAAIEN (volgens kinematische diagram)
{
  for (int d = 0; d < stappenDoorJoyM4; d++)
  {
    digitalWrite(dirPin4, LOW);
    digitalWrite(stepPin4, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin4, LOW);

```

```

    delayMicroseconds(200);
  }
}
if (waardeM4 > 512)
{
  for (int d = 0; d < stappenDoorJoyM4; d++)
  {
    digitalWrite(dirPin4, HIGH); // LINKSOM DRAAIEN (volgens kinematische diagram)
    digitalWrite(stepPin4, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin4, LOW);
    delayMicroseconds(200);
  }
}

}

if (waardeM5 < 490 || waardeM5 > 530)
{

  if (waardeM5 < 512) // RECHTSOM DRAAIEN (volgens kinematische diagram)
  {
    for (int e = 0; e < stappenDoorJoyM5; e++)
    {
      digitalWrite(dirPin5, LOW);
      digitalWrite(stepPin5, HIGH);
      delayMicroseconds(200);
      digitalWrite(stepPin5, LOW);
      delayMicroseconds(200);
    }
  }
  if (waardeM5 > 512)
  {
    for (int e = 0; e < stappenDoorJoyM5; e++)
    {
      digitalWrite(dirPin5, HIGH); // LINKSOM DRAAIEN (volgens kinematische diagram)
      digitalWrite(stepPin5, HIGH);
      delayMicroseconds(200);
      digitalWrite(stepPin5, LOW);
      delayMicroseconds(200);
    }
  }
}

}

if (waardeM6 < 490 || waardeM6 > 530)
{

```

```

if (waardeM6 < 512) // RECHTSOM DRAAIEN (volgens kinematische diagram)
{
  for (int f = 0; f < stappenDoorJoyM6; f++)
  {
    digitalWrite(dirPin6, LOW);
    digitalWrite(stepPin6, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin6, LOW);
    delayMicroseconds(200);
  }
}
if (waardeM6 > 512)
{
  for (int f = 0; f < stappenDoorJoyM6; f++)
  {
    digitalWrite(dirPin6, HIGH); // LINKSOM DRAAIEN (volgens kinematische diagram)
    digitalWrite(stepPin6, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin6, LOW);
    delayMicroseconds(200);
  }
}

}
if(Serial.read() == 1) joystickbediening = 1;
}
}
void positieAlsInVoorbeeld() {

  // Voor de homing wordt er van uitgegaan dat er gestart wordt in uitgestrekte stand: theta 1-
  6 = 0°
  // Gewenste positie: theta 1 = 30°, theta 2 = -45°, theta 3 = 90°, theta 4 = -45, theta 5 = -90,
  theta 6 = -90
  // Vooraf zeker de draairichting controlleren!
  theta1 = 0;
  theta2 = 0;
  theta3 = 0;
  theta4 = 0;
  theta5 = 0;
  theta6 = 0;

  float theta1gewenst = 30;
  float theta2gewenst = -45;
  float theta3gewenst = 90;
  float theta4gewenst = -45;
  float theta5gewenst = -90;
  float theta6gewenst = -90;

```

int aantalDeelBewegingen = 1000; // De beweging zal opgesplitst worden in 1000 deelbewegingen. Hieruit kan het aantal stappen van elke motor per deelbeweging bepaald worden. Voor een meer vloeiende beweging kan het aantal deelbewegingen opgedreven worden.

```
float aantalStappenPerDeelM1 = ((theta1gewenst / 360) * stepsperRevM1) /
aantalDeelBewegingen;
float aantalStappenPerDeelM2 = ((theta2gewenst / 360) * stepsperRevM2) /
aantalDeelBewegingen;
float aantalStappenPerDeelM3 = ((theta3gewenst / 360) * stepsperRevM3) /
aantalDeelBewegingen;
float aantalStappenPerDeelM4 = ((theta4gewenst / 360) * stepsperRevM4) /
aantalDeelBewegingen;
float aantalStappenPerDeelM5 = ((theta5gewenst / 360) * stepsperRevM5) /
aantalDeelBewegingen;
float aantalStappenPerDeelM6 = ((theta6gewenst / 360) * stepsperRevM6) /
aantalDeelBewegingen;
```

```
for (int z = 0; z < aantalDeelBewegingen; z++)
{
  for (int zM1 = 0; zM1 < aantalStappenPerDeelM1; zM1++)
  {
    // Controle draairichting
    if (theta1gewenst > 0)digitalWrite(dirPin1, HIGH);
    if (theta1gewenst < 0)digitalWrite(dirPin1, LOW);
    digitalWrite(stepPin1, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin1, LOW);
    delayMicroseconds(200);
  }

  for (int zM2 = 0; zM2 < aantalStappenPerDeelM2; zM2++)
  {
    // Controle draairichting
    if (theta2gewenst > 0)digitalWrite(dirPin2, HIGH);
    if (theta2gewenst < 0)digitalWrite(dirPin2, LOW);
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
  }

  for (int zM3 = 0; zM3 < aantalStappenPerDeelM3; zM3++)
  {
    // Controle draairichting
    if (theta3gewenst > 0)digitalWrite(dirPin3, HIGH);
    if (theta3gewenst < 0)digitalWrite(dirPin3, LOW);
    digitalWrite(stepPin3, HIGH);
```

```

    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}

for (int zM4 = 0; zM4 < aantalStappenPerDeelM4; zM4++)
{
    // Controle draairichting
    if (theta4gewenst > 0)digitalWrite(dirPin4, HIGH);
    if (theta4gewenst < 0)digitalWrite(dirPin4, LOW);
    digitalWrite(stepPin4, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin4, LOW);
    delayMicroseconds(200);
}

for (int zM5 = 0; zM5 < aantalStappenPerDeelM5; zM5++)
{
    // Controle draairichting
    if (theta5gewenst > 0)digitalWrite(dirPin5, HIGH);
    if (theta5gewenst < 0)digitalWrite(dirPin5, LOW);
    digitalWrite(stepPin5, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin5, LOW);
    delayMicroseconds(200);
}

for (int zM6 = 0; zM6 < aantalStappenPerDeelM6; zM6++)
{
    // Controle draairichting
    if (theta6gewenst > 0)digitalWrite(dirPin6, HIGH);
    if (theta6gewenst < 0)digitalWrite(dirPin6, LOW);
    digitalWrite(stepPin6, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin6, LOW);
    delayMicroseconds(200);
}

Serial.print("Eindpositie bereikt");
}
}

int invKin5( float XTCP, float YTCP, float ZTCP, float gripperHoek_d, int counterM1, int
counterM2, int counterM3, int counterM4, int counterM5, int counterM6 )
{
    // Bij de 5-assige inverse kinematica wordt de zesde as (motor 4) als star beschouwd.
    // Er wordt vanuit de horizontale stand gestart (zoals in het kinematisch diagram).

```



```
// Bepalen van de huidige hoeken (counters zijn tov basispositie zoals in kinematisch diagram)
```

```
float theta1Huidig = (counterM1 / stepsperRevM1) * 360;  
float theta2Huidig = (counterM1 / stepsperRevM1) * 360;  
float theta3Huidig = (counterM1 / stepsperRevM1) * 360;  
float theta4Huidig = (counterM1 / stepsperRevM1) * 360;  
float theta5Huidig = (counterM1 / stepsperRevM1) * 360;  
float theta6Huidig = (counterM1 / stepsperRevM1) * 360;
```

```
float deltaTheta1;  
float deltaTheta2;  
float deltaTheta3;  
float deltaTheta4;  
float deltaTheta5;  
float deltaTheta6;
```

```
// Definiëren verschillende afstanden zoals in tabel 7-1.
```

```
float L1 = 75.00;  
float L2 = 480.00;  
float L3 = 450.75;  
float L4 = 56.00;  
float L5 = 12.00;  
float L45 = 68.00;  
float H1 = 278.00;  
float D1 = 87.25;  
float D2 = 99.00;  
float D3 = 76.00;  
float D4 = 76.00;
```

```
float R;  
float D;  
float costheta3;  
float sintheta3;  
float alfa;  
float beta;
```

```
float pi = 3.141593;
```

```
// Bepaling kwadranten
```

```
int kwZ; // kwadranten volgens het zijaanzicht  
int kwB; // kwadranten volgens het bovenaanzicht  
if ((XTCP > 0) && (ZTCP > 0)) kwZ = 1 ;  
else if ((XTCP < 0) && (ZTCP > 0)) kwZ = 2 ;  
else if ((XTCP < 0) && (ZTCP < 0)) kwZ = 3 ;  
else if ((XTCP > 0) && (ZTCP < 0)) kwZ = 4 ;
```

```
if ((XTCP > 0) && (YTCP > 0)) kwB = 1 ;  
else if ((XTCP < 0) && (YTCP > 0)) kwB = 2 ;
```

```

else if ((XTCP < 0) && (YTCP < 0)) kwB = 3 ;
else if ((XTCP > 0) && (YTCP < 0)) kwB = 4 ;

// Inverse kinematica 5-assig: Deze verloopt grotendeels analoog met de inverse
positiekinematica uit 7.3.5.

/* Basishoek theta1 uit de x,y coördinaten rekening houdend met theta1Offset*/
if ((kwB == 1) or (kwB == 4)) theta1 = (atan(YTCP / XTCP) + abs(atan(-11.75 / sqrt((XTCP *
XTCP + YTCP * YTCP)))) * 180.00 / pi;
else theta1 = (atan(YTCP / XTCP) - abs(atan(-11.75 / sqrt((XTCP * XTCP + YTCP *
YTCP)))) * 180.00 / pi;

// Bepalen coördinaten POLS
float gripperHoek_r = radians( gripperHoek_d ); //gripperhoek in radialen voor
berekeningen
float Z = ZTCP - (( sin( gripperHoek_r )) * L45);
float X = XTCP - (( cos( gripperHoek_r )) * L45);
float Y = YTCP - (( sin (theta1 * pi / 180.00)) * L45);

/* Stap 2: Ellebooghoek theta3 */
if ((kwZ == 1) || (kwZ == 4)) R = sqrt((X * X + Y * Y) ) - 75.00 ;
else R = sqrt((X * X + Y * Y) ) + 75.00;

D = sqrt((R * R + (Z - 278.00) * (Z - 278.00)));
costheta3 = (D * D - 480.00 * 480.00 - 450.75 * 450.75) / (2.00 * 480.00 * 450.75);
sintheta3 = sqrt((1.00 - (costheta3 * costheta3)));
theta3 = atan(sintheta3 / costheta3) * 180.00 / pi;

/* Stap 3: Schouderhoek theta2 */
alfa = atan((Z - 278.00) / R);
beta = atan((450.75 * sintheta3) / (480.00 + 450.75 * costheta3));
if ((kwZ == 1) || (kwZ == 4)) theta2 = (-alfa - beta) * 180.00 / pi;
else theta2 = (-alfa + beta) * 180.00 / pi;

/* Step 4: pols hoek theta 5 */
float theta5 = ( gripperHoek_d - theta3 ) - theta2;

// Controle als de hoeken in het werkingsgebied liggen
if ((kwZ == 2) || (theta1 < -90) || (theta1 > 90) || (theta2 < -90) || (theta2 > 90) || (theta3 < 0) ||
(theta3 > 125) || (theta5 < -135) || (theta5 > 135)) Serial.println("Dit punt in de ruimte valt
buiten het werkingsgebied");

// Sturen van de verschillende motoren, analoog aan methode "positieAlsInVoorbeeld()".
Verschillend is hier dat alle stappen geteld worden. Daarna worden ze teruggestuurd naar de
loop.
else

```

```

{
// Hoekverschil bepalen (teken!)
if (theta1Huidig < theta1) deltaTheta1 = theta1 - theta1Huidig; //deltaTheta1
else deltaTheta1 = -(theta1Huidig - theta1);

if (theta2Huidig < theta2) deltaTheta2 = theta2 - theta2Huidig; //deltaTheta2
else deltaTheta2 = -(theta2Huidig - theta2);

if (theta3Huidig < theta3) deltaTheta3 = theta3 - theta3Huidig; //deltaTheta3
else deltaTheta3 = -(theta3Huidig - theta3);

if (theta4Huidig < theta4) deltaTheta4 = theta4 - theta4Huidig; //deltaTheta4
else deltaTheta4 = -(theta4Huidig - theta4);

if (theta5Huidig < theta5) deltaTheta5 = theta5 - theta5Huidig; //deltaTheta5
else deltaTheta5 = -(theta5Huidig - theta5);

if (theta6Huidig < theta6) deltaTheta6 = theta6 - theta6Huidig; //deltaTheta6
else deltaTheta6 = -(theta6Huidig - theta6);

int aantalDeelBewegingen = 1000; // De beweging zal opgesplitst worden in 1000
deelbewegingen. Hieruit kan het aantal stappen van elke motor per deelbeweging bepaald
worden. Voor een meer vloeiende beweging kan het aantal deelbewegingen opgedreven
worden.

float aantalStappenPerDeelM1 = ((deltaTheta1 / 360) * stepsperRevM1) /
aantalDeelBewegingen;
float aantalStappenPerDeelM2 = ((deltaTheta2 / 360) * stepsperRevM2) /
aantalDeelBewegingen;
float aantalStappenPerDeelM3 = ((deltaTheta3 / 360) * stepsperRevM3) /
aantalDeelBewegingen;
float aantalStappenPerDeelM4 = ((deltaTheta4 / 360) * stepsperRevM4) /
aantalDeelBewegingen;
float aantalStappenPerDeelM5 = ((deltaTheta5 / 360) * stepsperRevM5) /
aantalDeelBewegingen;
float aantalStappenPerDeelM6 = ((deltaTheta6 / 360) * stepsperRevM6) /
aantalDeelBewegingen;

for (int z = 0; z < aantalDeelBewegingen; z++)
{
for (int zM1 = 0; zM1 < aantalStappenPerDeelM1; zM1++)
{
// Controle draairichting
if (deltaTheta1 > 0) {
digitalWrite(dirPin1, HIGH);
counterM1++;
}
if (deltaTheta1 < 0) {

```

```

    digitalWrite(dirPin1, LOW);
    counterM1--;
}
digitalWrite(stepPin1, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin1, LOW);
delayMicroseconds(200);
}

for (int zM2 = 0; zM2 < aantalStappenPerDeelM2; zM2++)
{
    // Controle draairichting
    if (deltaTheta2 > 0) {
        digitalWrite(dirPin2, HIGH);
        counterM2++;
    }
    if (deltaTheta2 < 0) {
        digitalWrite(dirPin2, LOW);
        counterM2--;
    }
    digitalWrite(stepPin2, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin2, LOW);
    delayMicroseconds(200);
}

for (int zM3 = 0; zM3 < aantalStappenPerDeelM3; zM3++)
{
    // Controle draairichting
    if (deltaTheta3 > 0) {
        digitalWrite(dirPin3, HIGH);
        counterM2++;
    }
    if (deltaTheta3 < 0) {
        digitalWrite(dirPin3, LOW);
        counterM3--;
    }
    digitalWrite(stepPin3, HIGH);
    delayMicroseconds(200);
    digitalWrite(stepPin3, LOW);
    delayMicroseconds(200);
}

for (int zM4 = 0; zM4 < aantalStappenPerDeelM4; zM4++)
{
    // Controle draairichting
    if (deltaTheta4 > 0) {
        digitalWrite(dirPin4, HIGH);
        counterM4++;

```

```

}
if (deltaTheta4 < 0) {
  digitalWrite(dirPin4, LOW);
  counterM4--;
}
digitalWrite(stepPin4, HIGH);
delayMicroseconds(200);
digitalWrite(stepPin4, LOW);
delayMicroseconds(200);
}

for (int zM5 = 0; zM5 < aantalStappenPerDeelM5; zM5++)
{

  // Controle draairichting
  if (deltaTheta5 > 0) {
    digitalWrite(dirPin5, HIGH);
    counterM5++;
  }
  if (deltaTheta5 < 0) {
    digitalWrite(dirPin5, LOW);
    counterM5--;
  }
  digitalWrite(stepPin5, HIGH);
  delayMicroseconds(200);
  digitalWrite(stepPin5, LOW);
  delayMicroseconds(200);
}
for (int zM6 = 0; zM6 < aantalStappenPerDeelM6; zM6++)
{
  // Controle draairichting
  if (deltaTheta6 > 0) {
    digitalWrite(dirPin6, HIGH);
    counterM6++;
  }
  if (deltaTheta6 < 0) {
    digitalWrite(dirPin6, LOW);
    counterM6--;
  }
  digitalWrite(stepPin6, HIGH);
  delayMicroseconds(200);
  digitalWrite(stepPin6, LOW);
  delayMicroseconds(200);
}
Serial.print("Positie bereikt");
}
}
}

```

# Bijlage H LIJST MET ALLE GEBRUIKTE MATERIALEN VOOR DE KOSTPRIJSANALYSE

Tabel H-1 Gebruikte materialen

Component	Aantal	Prijs/stuk (€)	Totaalprijs (€)
<b>Lagers</b>			
SKF 30204 J2-Q	6	10.89	65.34
SKF 30206 J2-Q	2	13.25	26.5
SKF 30212 J2-Q	2	41.14	82.28
<b>Tandriemen</b>			
OMEGA 366-3M-9	1	5.69	5.69
OMEGA 582-3M-9	1	6.66	6.66
<b>Tandriemschijven</b>			
OPTIBELT HTD 30-3M-9	3	4.48	13.44
OPTIBELT HTD 40-3M-9	2	5.09	10.18
<b>Kettingwielen</b>			
06B-2Z-12	1	4.00	4.00
06B-2Z-14	1	4.24	4.24
06B-2Z-16	1	4.72	4.72
06B-2Z-20	1	5.87	5.87
<b>Kettingen</b>			
FENNER ISO606 06B-2 5 (BULK)	106	0.3423	36.2838
FENNER VERBINDINSSCHAKEL 06B-2	2	1.21	1.21
<b>Motoren</b>			
STEPPER ONLINE 14HS13-0804S-PG5	1	31.61	31.61
STEPPER ONLINE 17HS15-1684S-HG10	1	43.84	43.84
STEPPER ONLINE 17HS15-1684S-HG15	2	54.42	108.84
STEPPER ONLINE 23HS22-2804S-HG50	1	81.64	81.64
STEPPER ONLINE 34HS31-5504S	1	40.82	40.82
<b>Gearbox</b>			
TOAUTO - 86PX-30	1	61.4	61.40
<b>Bracket</b>			
STEPPER ONLINE ST-M7	1	7.96	7.96
<b>Drivers</b>			
STEPPER ONLINE DM320T	1	18.07	18.07

STEPPER ONLINE DM542T	4	28.39	113.56
STEPPER ONLINE DM860T	1	48.78	48.78
<b>Bevestigingsmateriaal</b>			
DIN 125A M4 (BULK)	12	0.0082	0.0984
DIN 125A M6 (BULK)	18	0.0146	0.2628
DIN 912 M4X30 (BULK)	4	0.0662	0.2648
DIN 912 M5X20 (BULK)	4	0.0500	0.2000
DIN 912 M6X30 (BULK)	8	0.1005	0.8040
DIN 912 M8X25 (BULK)	3	0.1852	0.7408
DIN 914 M3X6 (BULK)	4	0.1249	0.4996
DIN 931 M6x30 (BULK)	4	0.1307	0.5228
DIN 934 M4 (BULK)	8	0.0094	0.0752
DIN 934 M6 (BULK)	12	0.0202	0.2424
DIN 6885A 3X3X16 (BULK)	1	0.7142	0.7142
DIN 6885A 4X4X12 (BULK)	2	0.2880	0.576
DIN 6885A 5X5X18 (BULK)	1	0.4321	0.4321
DIN 6885A 5X5X20 (BULK)	1	0.4832	0.4832
DIN 6885A 6X6X20 (BULK)	1	0.4808	0.4808
DIN 7991 M3X16 (BULK)	30	0.0379	1.1370
DIN 7991 M4X12 (BULK)	8	0.0464	0.3712
DIN 7991 M4X16 (BULK)	24	0.0559	1.5652
DIN 7991 M4x20 (BULK)	62	0.0662	4.1044
DIN 7991 M4X30 (BULK)	12	0.0928	1.1136
DIN 7991 M4X35 (BULK)	6	0.1	0.6
DIN 7991 M6X30 (BULK)	14	0.0816	1.1424
<b>Uitbestedingen</b>			
VOET LASERCUTTING - PLOOI/LASERWERK	1	314.60	314.60
	<b>Totaal (€):</b>	<b>1149.93</b>	

Tabel H-2 Bulkgoederen

Component	Aantal/ verpakking	Prijs/ verpakking (€)	Prijs/ stuk (€)
<b>Kettingen</b>			
FENNER ISO606 06B-2 5	525	179.69	0.3423
<b>Bevestigingsmateriaal</b>			
DIN 125A M4	500	4.11	0.0082
DIN 125A M6	500	7.32	0.0146
DIN 912 M4X30	200	13.24	0.0662
DIN 912 M5X20	200	9.99	0.0500
DIN 912 M6X30	200	20.09	0.1005
DIN 912 M8X25	200	37.03	0.1852
DIN 914 M3X6	500	62.44	0.1249
DIN 931 M6x30	200	26.14	0.1307
DIN 934 M4	500	4.72	0.0094
DIN 934 M6	500	10.10	0.0202
DIN 6885A 3X3X16	50	35.71	0.7142
DIN 6885A 4X4X12	25	7.20	0.2880
DIN 6885A 5X5X18	200	86.42	0.4321
DIN 6885A 5X5X20	25	12.08	0.4832
DIN 6885A 6X6X20	25	12.02	0.4808
DIN 7991 M3X16	200	7.57	0.0379
DIN 7991 M4X12	200	9.27	0.0464
DIN 7991 M4X16	200	11.18	0.0559
DIN 7991 M4x20	200	13.24	0.0662
DIN 7991 M4X30	200	18.56	0.0928
DIN 7991 M4X35	200	19.99	0.1000
DIN 7991 M6X30	200	16.31	0.0816
	<b>Totaalprijs bulk (€):</b>	<b>444.73</b>	



# **Bijlage I      TECHNISCHE DOSSIERS**

In de bijgeleverde groene mappen zijn volgende tekendossiers te vinden:

- AgroBot Bovenbouw: Assembly - stuktekeningen - documentatie
- AgroBot A.R.M.: Assembly
- AgroBot A.R.M.: Stuktekeningen - lasertekeningen - plooittekeningen - documentatie

---

**FACULTEIT INDUSTRIELE INGENIEURSWETENSCHAPPEN**  
TECHNOLOGIECAMPUS GENT  
Gebroeders De Smetstraat 1  
9000 GENT, België  
tel. + 32 92 65 86 10  
iiw.gent@kuleuven.be  
www.iw.kuleuven.be

