



FACHHOCHSCHUL - BACHELORSTUDIENGANG
AUTOMATISIERUNGSTECHNIK

Implementierung einer Armkinematik am Roboter für die Eurobot 2010

Als Bachelorarbeit eingereicht
zur Erlangung des akademischen Grades
Bachelor of Science in Engineering

von

ROBERT RATHGEBER

Juni 2010

Betreuung der Bachelorarbeit durch
Prof(FH) Dipl. Ing. Walter Rokitansky



Campus **Wels**

Ich erkläre ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benutzt, die den benutzten Quellen entnommenen Stellen als solche kenntlich gemacht habe und dass die Arbeit mit der vom Begutachter beurteilten Arbeit übereinstimmt.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht.

.....
«Vorname, Name»

.....
«Wohnort», «Datum»

Kurzfassung

Die vorliegende Arbeit befasst sich mit der Implementierung eines Greifarmes auf einem autonomen Roboter.

Als Erstes wird auf den mechanischen Aufbau des Greifers eingegangen, welche Komponenten verwendet wurden und auf die Funktionsweise. Im nächsten Schritt werden die theoretischen Grundlagen zur Modellierung eines Roboterarmes vorgestellt. Im theoretischen Teil wird die Denavit-Hartenberg-Konvention erläutert. Ausgehend davon wird die mathematische Modellierung des Armes präsentiert.

Das mathematische Modell dient als Grundlage zur Entwicklung eines inversen Algorithmus. Der Algorithmus und die Funktionsweise des Greifertasks wird anhand von Diagrammen erläutert. Auf Darstellung von Quellcode wird verzichtet.

Im letzten Abschnitt werden Fehler und Probleme analysiert, welche im Verlauf des Projektes aufgetreten sind.

Inhaltsverzeichnis

Inhaltsverzeichnis	III
1 Motivation	1
1.1 Eurobot	1
1.2 Eurobot - Thema 2010[1]	2
1.2.1 Spielelemente	2
1.3 Gesamtes Regelwerk	3
2 Projektbeschreibung	4
2.1 Projektinhalt	4
2.2 Der Greifarm[2]	4
3 Mathematische Modellierung mit Denavit-Hartenberg	6
3.1 Die Denavit-Hartenberg Parameter[3]	6
3.2 Festlegung der Koordinatensysteme[4]	7
3.2.1 Koordinatensystem in einer rotatorisch fluchtenden Achse	7
3.2.2 Festlegung des Koordinatensystems K_i bei nicht-schneidenden nicht-parallelen Achsen	7
3.2.3 Festlegung des Koordinatensystems K_i bei parallelen Achsen	8
3.2.4 Festlegung des Koordinatensystems K_i bei sich schneidenden Achsen	8
4 Mathematische Modellierung des Roboterarmes	10
4.1 Definieren der Koordinatensysteme	10
4.2 Aufstellen der Transformationen	10
4.3 Erläuterung zur Modellierung	12
4.4 Errechnen der Pickframes	12
5 Inverse Kinematik	13
5.1 Berechnen der Armwinkel	13
5.2 Algorithmus zum Berechnen der Winkeln	15

6 Der Greifertask	17
6.1 Programmübersicht	17
7 Lessons Learned	19
Literaturverzeichnis	21

Kapitel 1

Motivation

Die Unmenschlichkeit des Computers beruht darauf, dass er richtig programmiert und einwandfrei funktionierend, so absolut ehrlich ist.

Isaac Asimov (1920 - 1992) , russisch-amerikanischer Biochemiker, Sachbuchautor

Die heutige technische Welt ist geprägt von der Robotertechnik. Das Einsatzspektrum von Robotern ist weit gefächert. Das Anwendungsgebiet reicht von einfachen Positionierungsaufgaben bis hin zu vollautomatisierten Fertigungslinien. Die Robotik vereint unter anderem Ingenieursdisziplinen wie Elektronik, Informatik und Maschinenbau.

Das RoboRacingTeam der Fachhochschule Wels hat es sich zum Ziel gesetzt, Studenten die Thematik der Robotik näher zu bringen und diese dafür zu begeistern. Während des Studiums besteht die Möglichkeit, in Form von Projektarbeiten mit dem RoboRacingTeam zusammenzuarbeiten. Im Zuge dieser Projektarbeiten wird jeweils in einem Team ein Roboter für einen nationalen oder internationalen Wettbewerb konstruiert.

1.1 Eurobot

Die Eurobot ist aus einem nationalen französischen Wettbewerb hervorgegangen. Das RoboRacingTeam nimmt seit 2007 erfolgreich an dem internationalen Wettbewerb Eurobot teil. Die Aufgabe für den Bewerb wird jedes Jahr neu definiert. Einige Grundregeln sind fixe Bestandteile und werden nicht von Jahr zu Jahr geändert. Die Spielfeldgröße beträgt immer 300 cm x 210 cm, die Spieldauer ist auf 90 sec festgelegt und der maximale Umfang eines Roboters darf während des Spieles 1400 mm nicht überschreiten.

1.2 Eurobot - Thema 2010[1]

Das Thema der Eurobot 2010 ist „Feed the World“. Die Roboter befinden sich auf einem Feld mit Getreide, Früchte und Gemüse. Aufgabe der Roboter ist, so viele der genannten Güter wie möglich innerhalb der 90 Sekunden einzusammeln. Jeder Roboter darf alle Elemente aufnehmen. Die Punkte werden nach Gewicht vergeben. Der Roboter, der am Ende das meiste Gewicht in den Sammelbehältern hat, gewinnt das Spiel.

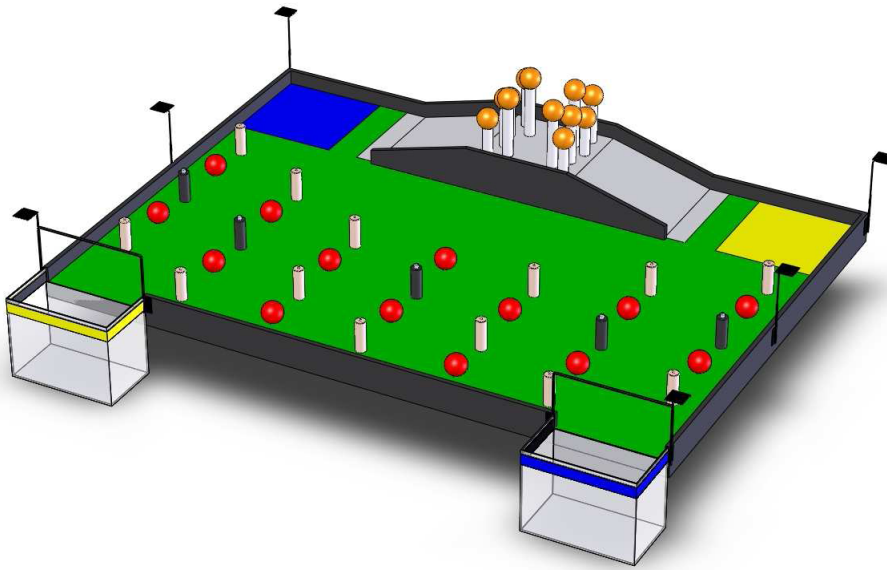


Abbildung 1.2.1: Spielfeld Eurobot 2010

1.2.1 Spielelemente

Wie bereits erwähnt, müssen verschiedene Elemente eingesammelt werden. Diese Elemente unterscheiden sich durch Gewicht, Material und Lage am Spielfeld.

1. Die Orangen sind direkt neben dem Startfeld abzuholen. Es gilt eine Rampe hochzufahren, anschließend müssen die Orangen von einem Baum gepflückt werden. Die Problematik dabei ist, dass die Orangen (sechs pro Seite) in allen drei Dimensionen zueinander versetzt sind. Die Orange ist mit 300g das schwerste Element am Spielfeld und am schwierigsten aufzunehmen. Orangen werden als Vollgummibälle dargestellt, sie haben einen Durchmesser von 100 mm.
2. Die Getreideähren sind am ganzen Spielfeld in fix definierten Abständen verteilt. Diese bestehen aus einem weißen Kunststoffzylinder und wiegen 250g. Sie haben einen Durchmesser von 50 mm und sind 150 mm hoch. Mit einem Zapfen werden die Ähren in Bohrungen gesteckt. Es befinden sich auch schwarze Ähren auf dem Spielfeld, welche nicht aufgenommen werden können, da sie fest im Boden verschraubt sind. Die Position der schwarzen Ähren ist von Spiel zu Spiel verschieden.

3. Die Tomaten wiegen nur 150 g und bringen somit am wenigsten Punkte, dafür sind davon reichlich am Spielfeld vorhanden. Die Tomaten sind zwischen den Kornähren positioniert. Tomaten sind ebenfalls Gummibälle mit 100 mm Durchmesser, jedoch innen hohl.

1.3 Gesamtes Regelwerk

Das gesamte Regelwerk umfasst 44 Seiten, zusätzlich gibt es Ergänzungen zum Regelwerk, die so genannten FAQ. Das Regelwerk für die Eurobot 2010 kann unter der Homepage der Eurobot nachgelesen werden. <http://www.eurobot.org/eng/rules.php>

Kapitel 2

Projektbeschreibung

Beim Eurobotbewerb stellten Tomaten und Orangen wichtige Elemente dar. Um diese aufnehmen zu können wurde ein Roboterarm mit vier Achsen geplant. Der autonome Roboter selbst wurde mit zwei dieser Greifarme bestückt. Ziel des Berufspraktikums an der Fachhochschule war, die Achsensteuerung zu implementieren. Der mechanische Aufbau sowie die Elektronik zur Ansteuerung der Servos wurden zur Verfügung gestellt. Die Achssteuerung wurde mit einer inversen Kinematik realisiert, damit der Roboter, abhängig von seiner aktuellen Position, die Elemente richtig greift.

2.1 Projektinhalt

Folgende Punkte galt es im Berufspraktikum zu realisieren:

1. Aufstellen der inversen Kinematik
2. Errechnen des Pickframes für Tomaten
3. Errechnen des Pickframes für Orangen
4. Umsetzen des mathematischen Modells in ein C-Programm
5. Inbetriebnahme der Arme

In der folgenden Ausarbeitung wird zuerst auf die mathematische Modellierung eingegangen. Im Anschluss werden die Grundzüge des C-Programmes vorgestellt.

2.2 Der Greifarm[2]

Der Greifarm besitzt insgesamt vier rotatorische Achsen. Die Basisachse ist eine rotatorisch fluchtende Achse, die weiteren drei Folgachsen sind rotatorisch nicht fluchtend. Als Endeffektor wird ein Faltenbalgsauger verwendet.

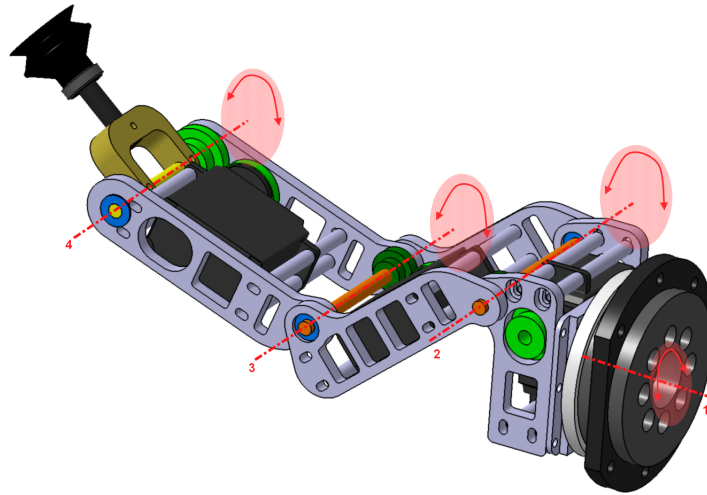


Abbildung 2.2.1: Greifarm [2]

Die Basisachse besitzt einen Zahnkranz, welcher über ein Zahnrad mit einem Faulhaber DC-Motor angesteuert wird. Die Achsen zwei bis vier werden mit digitalen Servos der Firma Futaba positioniert. Beschränkt durch mechanische Anschläge oder elektrische Leitungen ergaben sich folgende maximale Bewegungsräume:

Achse	maximaler Winkel [°]
1	360
2	180
3	180
4	120

Tabelle 2.2.1: Achswinkel

Kapitel 3

Mathematische Modellierung mit Denavit-Hartenberg

Für die mathematische Modellierung eines Roboterarmes bedient man sich Transformationen. Eine Transformation kann eine Rotation um eine Achse oder eine Translation in Richtung einer Achse sein. Zur mathematischen Modellierung muss man grundsätzlich nur in jedes Gelenk des Armes ein Koordinatensystem legen. Die einzelnen Koordinatensysteme werden mittels Transformationen ineinander übergeführt. Um dies sinnvoll zu realisieren gibt es die Denavit- und Hartenberg Konvention.

3.1 Die Denavit-Hartenberg Parameter[3]

Wie bereits erwähnt wird in jedes Gelenk des Roboterarmes ein Koordinatensystem gelegt. Die einzelnen Koordinatensysteme werden einzeln ineinander übergeführt. Wichtig dabei ist, dass die Koordinatensysteme so gelegt werden, dass die Anzahl der Transformationen minimal wird.

Die Denavit- und Hartenberg Konvention definiert vier Transformationen mit denen die einzelnen Koordinatensysteme ineinander übergeführt werden sollen.

1. Rotation um die z_{n-1} Achse mit dem Winkel θ_n
2. Translation in Richtung der z_{n-1} Achse mit der Länge d_n
3. Translation in Richtung der x_n Achse mit der Länge a_n
4. Rotation um die x_n Achse mit dem Winkel α_n

Daraus ergibt sich folgende Teiltransformation:

$${}^{n-1}T = ROT(z_{n-1}; \Theta_n) \cdot TRANS(z_{n-1}; d_n) \cdot TRANS(x_n; a_n) \cdot ROT(x_n; \alpha_n) \quad (3.1.1)$$

Aus den Teiltransformationen kann eine Gesamttransformation von der Basis zum Endeffektor erstellt werden.

$${}^0_kT = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \dots {}^{k-1}_kT \quad (3.1.2)$$

3.2 Festlegung der Koordinatensysteme[4]

Zur Festlegung der Koordinatensysteme gibt es feste Regeln. Wichtig dabei ist, dass die Koordinatensysteme so gelegt werden, dass möglichst viele Denavit-Hartenberg-Parameter zu null werden. Nachfolgend werden einige Grundsätze zur Festlegung der Koordinatensysteme vorgestellt.

3.2.1 Koordinatensystem in einer rotatorisch fluchtenden Achse

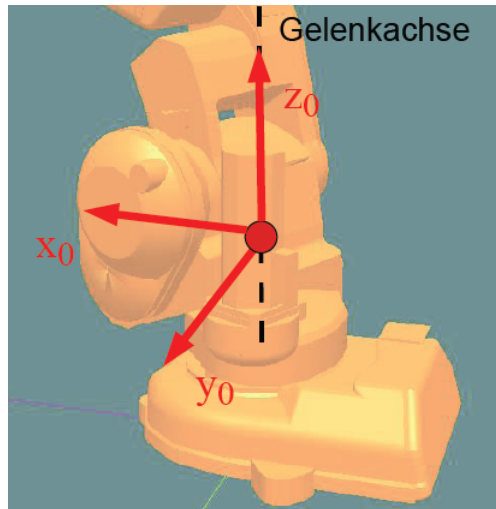


Abbildung 3.2.1: Rotatorisch fluchtende Achse [4]

1. z-Achse in die Rotationsachse des Armteils i legen
2. x- und y- Achse frei wählbar, soll ein kartesisches Rechtssystem ergeben

3.2.2 Festlegung des Koordinatensystems K_i bei nicht-schneidenden nicht-parallelen Achsen

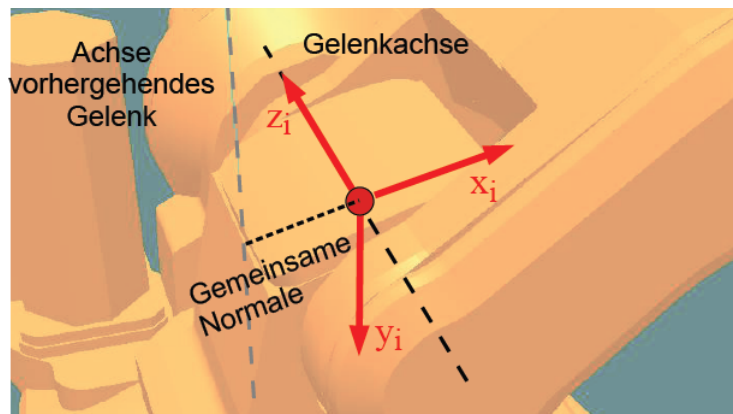


Abbildung 3.2.2: nicht-schneidende nicht-parallele Achse [4]

1. z_i - Achse in die Gelenkachse legen

2. x_i wird in Richtung der gemeinsamen Normale zwischen K_{i-1} und K_i gelegt, von K_{i-1} wegweisend
3. y_i zum Rechtssystem ergänzen

3.2.3 Festlegung des Koordinatensystems K_i bei parallelen Achsen

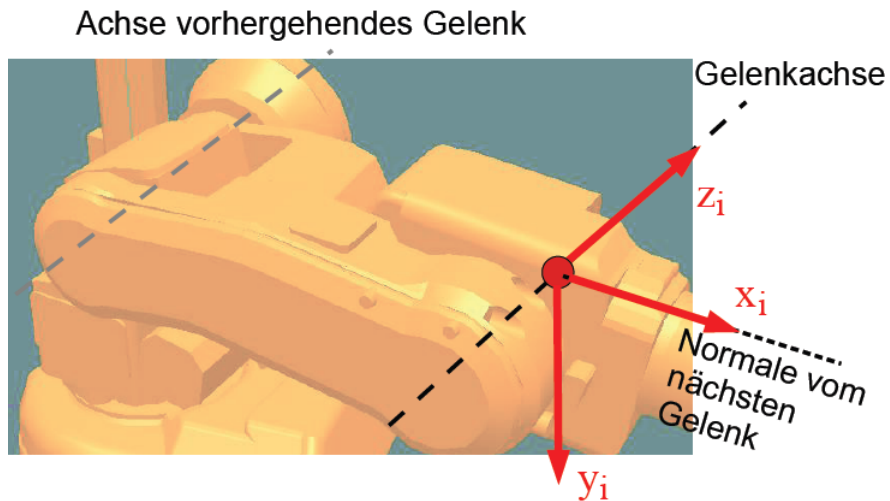


Abbildung 3.2.3: parallele Achsen [4]

1. z_i -Achse in die Gelenkachse des Armteiles
2. x_i -Achse in die Normale vom nächsten Gelenk legen
3. y_i -Achse zum Rechtssystem ergänzen

3.2.4 Festlegung des Koordinatensystems K_i bei sich schneidenden Achsen

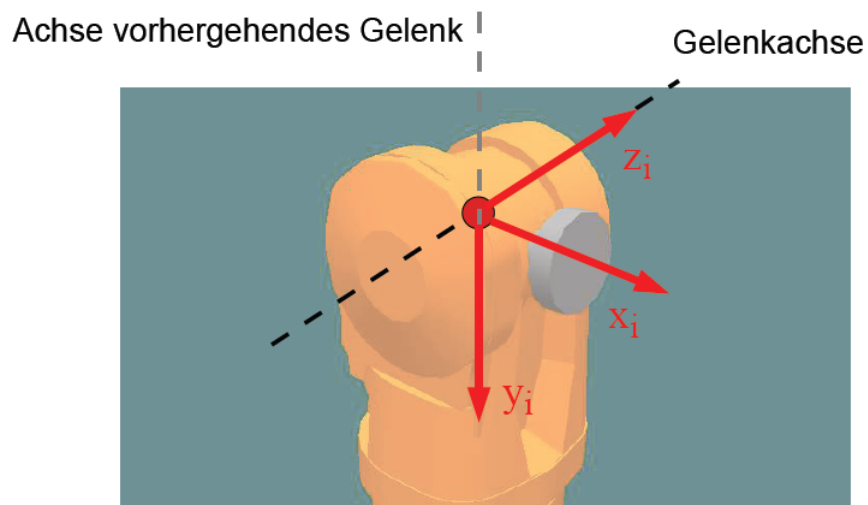


Abbildung 3.2.4: schneidende Achsen [4]

1. Ursprung in den Schnittpunkt der Achsen legen

2. z_i -Achse in die Gelenkachse des Armteiles
3. x_i -Achse in Richtung des nächsten Koordinatensystems
4. y_i -Achse zum Rechtssystem ergänzen

Kapitel 4

Mathematische Modellierung des Roboterarmes

4.1 Definieren der Koordinatensysteme

Für den Greifarm wurden sechs Koordinatensysteme definiert. Das Koordinatensystem $\{0\}$ bildet die feste Basis des Armes. Die rotatorischen Achsen sind definiert durch die Koordinatensysteme $\{1\}$ bis $\{4\}$, $\{5\}$ definiert den Toolcenterpoint.

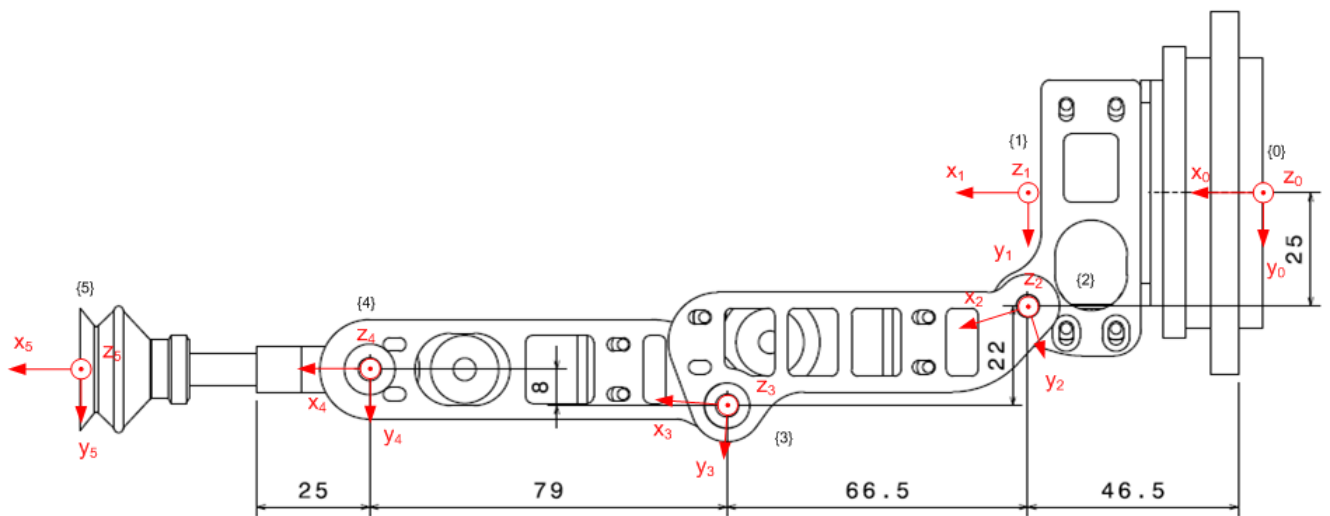


Abbildung 4.1.1: Roboterarm mit Koordinatensysteme

4.2 Aufstellen der Transformationen

Als nächste Schritte werden nun die Teiltransformationen definiert, aus diesen Einzeltransformationen wird anschließend die Gesamttransformation ermittelt. Die einzelnen Achswinkeln werden mit θ_n festgelegt. Die Einzeltransformationen wurden folgendermaßen definiert:

1.

$${}^0_1T = \text{Trans}(x, 46.5) = \begin{bmatrix} 1 & 0 & 0 & 46.5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2.1)$$

2.

$${}^1_2T = \text{Rot}(x, \theta_1) \cdot \text{Trans}(y, 25) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta_1 & -\sin \theta_1 & 0 \\ 0 & \sin \theta_1 & \cos \theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 25 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2.2)$$

3.

$${}^2_3T = \text{Rot}(z, \theta_2) \cdot \text{Trans}(x, 70.05) = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & 0 \\ \sin \theta_2 & \cos \theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 70.05 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2.3)$$

4.

$${}^3_4T = \text{Rot}(z, \theta_3) \cdot \text{Trans}(x, 79.4) = \begin{bmatrix} \cos \theta_3 & -\sin \theta_3 & 0 & 0 \\ \sin \theta_3 & \cos \theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 79.4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2.4)$$

5.

$${}^4_5T = \text{Rot}(z, \theta_4) \cdot \text{Trans}(x, 50) = \begin{bmatrix} \cos \theta_4 & -\sin \theta_4 & 0 & 0 \\ \sin \theta_4 & \cos \theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 50 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2.5)$$

Aus den Einzelmatrizen 4.2.1 bis 4.2.5 kann nun die Gesamttransformation berechnet werden.

$${}^0_5T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T \cdot {}^4_5T \quad (4.2.6)$$

Nach Ausmultiplizieren und Vereinfachen der Einzelmatrizen folgt die Gesamttransformationsmatrix:

$${}^0_5T = \begin{bmatrix} r_{00} & r_{01} & r_{02} & r_{03} \\ r_{10} & r_{11} & r_{12} & r_{13} \\ r_{20} & r_{21} & r_{22} & r_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2.7)$$

$$r_{00} = \cos(\theta_2 + \theta_3 + \theta_4) \quad (4.2.8)$$

$$r_{10} = \cos(\theta_1) \cdot \sin(\theta_2 + \theta_3 + \theta_4) \quad (4.2.9)$$

$$r_{20} = \sin(\theta_1) \cdot \sin(\theta_2 + \theta_3 + \theta_4) \quad (4.2.10)$$

$$r_{01} = -\sin(\theta_2 + \theta_3 + \theta_4) \quad (4.2.11)$$

$$r_{11} = \cos(\theta_1) \cdot \cos(\theta_2 + \theta_3 + \theta_4) \quad (4.2.12)$$

$$r_{21} = \sin(\theta_1) \cdot \cos(\theta_2 + \theta_3 + \theta_4) \quad (4.2.13)$$

$$r_{02} = 0 \quad (4.2.14)$$

$$r_{12} = -\sin(\theta_1) \quad (4.2.15)$$

$$r_{22} = \cos(\theta_1) \quad (4.2.16)$$

$$r_{03} = 46.5 + 79.4 \cos(\theta_2 + \theta_3) + 70.05 \cos(\theta_2) + 50 \cos(\theta_2 + \theta_3 + \theta_4) \quad (4.2.17)$$

$$r_{13} = \cos(\theta_1) [25 + 70.05 \sin(\theta_2) + 79.4 \sin(\theta_2 + \theta_3) + 50 \sin(\theta_2 + \theta_3 + \theta_4)] \quad (4.2.18)$$

$$r_{23} = \sin(\theta_1) [25 + 70.05 \sin(\theta_2) + 79.4 \sin(\theta_2 + \theta_3) + 50 \sin(\theta_2 + \theta_3 + \theta_4)] \quad (4.2.19)$$

4.3 Erläuterung zur Modellierung

Vergleicht man die Vorgehensweise nach Denavit-Hartenberg und das Vorgehen in Kapitel 4 ist zu erkennen, dass die Denavit-Hartenberg Konvention nicht eingehalten wird.

Die Komplexität von Transformationen wird maßgeblich von den Rotationen geprägt. Deshalb wurde bei der Modellierung darauf geachtet, dass die Rotationen im System minimal werden. Translationen sind in alle Richtungen möglich.

4.4 Errechnen der Pickframes

In den vorhergehenden Abschnitten wurde das mathematische Modell des Armes festgelegt. Damit daraus die Armwinkel berechnet werden können, müssen für die Elemente Pickframes errechnet werden. Ein Pickframe definiert den Aufnahmepunkt und legt zusätzlich ein Koordinatensystem im Aufnahmepunkt fest.

Um Tomaten aufnehmen zu können wurde an der Front des Roboters eine Sensorleiste mit Infrarotsensoren montiert. Die Sensorleiste soll den seitlichen Versatz zum Greifer auswerten. Wird vor der Sensorleiste eine Tomate erkannt, stoppt der Roboter und soll die Tomate aufnehmen. Der Versatz nach vorne ist immer annähernd gleich.

Die Orangen liegen am Feld auf einer fix definierten Position. Abhängig von der Roboterposition kann eine Relativposition der Orangen zum Roboter berechnet werden.

Die Berechnung der Pickframes erfolgt aus einfachen geometrischen Zusammenhängen, deshalb wird an dieser Stelle nicht weiter darauf eingegangen.

Kapitel 5

Inverse Kinematik

Das Pickframe gibt die Lage und Orientierung des Endeffektors im Basiskoordinatensystem des Roboterarmes an. Aus dieser Angabe müssen nun die Achswinkel berechnet werden. In der folgenden Ausarbeitung beschreibt der Buchstabe P das Koordinatensystem im Aufnahme punkt.

5.1 Berechnen der Armwinkel

Das Koordinatensystem des Endeffektor muss gleich dem Koordinatensystem des Pickframes sein. Daraus folgt:

$${}^0_5T = {}^0_P T \quad (5.1.1)$$

$${}^0_5T = \begin{bmatrix} r_{00} & r_{01} & r_{02} & r_{03} \\ r_{10} & r_{11} & r_{12} & r_{13} \\ r_{20} & r_{21} & r_{22} & r_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.1.2)$$

$${}^0_P T = \begin{bmatrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \\ p_{20} & p_{21} & p_{22} & p_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.1.3)$$

$$\begin{bmatrix} r_{00} & r_{01} & r_{02} & r_{03} \\ r_{10} & r_{11} & r_{12} & r_{13} \\ r_{20} & r_{21} & r_{22} & r_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \\ p_{20} & p_{21} & p_{22} & p_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.1.4)$$

Für die weiteren Berechnungen werden die Matrixgleichungen aus Kapitel 4 verwendet. θ_1 lässt sich aus r_{12} bzw. r_{22} berechnen:

$$r_{22} = p_{22} = \cos(\theta_1) \quad (5.1.5)$$

$$\theta_1 = \arccos(p_{22}) \quad (5.1.6)$$

bzw. aus r_{12}

$$\theta_1 = \arcsin(-p_{22}) \quad (5.1.7)$$

Die drei Hauptgleichungen aus dem Ortsvektor lauten:

$$I : 46.5 + 79.4 \cos(\theta_2 + \theta_3) + 70.05 \cos(\theta_2) + 50 \cos(\theta_2 + \theta_3 + \theta_4) = p_{03} \quad (5.1.8)$$

$$II : \cos(\theta_1) [25 + 70.05 \sin(\theta_2) + 79.4 \sin(\theta_2 + \theta_3) + 50 \sin(\theta_2 + \theta_3 + \theta_4)] = p_{13} \quad (5.1.9)$$

$$III : \sin(\theta_1) [25 + 70.05 \sin(\theta_2) + 79.4 \sin(\theta_2 + \theta_3) + 50 \sin(\theta_2 + \theta_3 + \theta_4)] = p_{23} \quad (5.1.10)$$

Gleichung zwei und drei sagen im Prinzip das Selbe aus. Diese definieren den seitlichen Versatz und den Höhenunterschied zum Greifer. Würde ein Ball vor dem Greifer liegen, sodass keine Drehung um θ_1 notwendig wäre, wird $\cos(\theta_1) = 1$ und $\sin(\theta_1) = 0$. Nun können einige Terme substituiert werden:

$$\cos(\theta_2 + \theta_3 + \theta_4) = p_{00}$$

$$\cos(\theta_1) = p_{22}$$

$$\sin(\theta_1) = -p_{12}$$

$$\cos(\theta_1) \cdot \sin(\theta_2 + \theta_3 + \theta_4) = p_{10}$$

$$\sin(\theta_1) \cdot \sin(\theta_2 + \theta_3 + \theta_4) = p_{20}$$

Nach Einsetzen in die Gleichungen folgt:

$$I : 46.5 + 79.4 \cos(\theta_2 + \theta_3) + 70.05 \cos(\theta_2) + 50 p_{00} = p_{03} \quad (5.1.11)$$

$$II : 25 p_{22} + 70.05 p_{22} \sin(\theta_2) + 79.4 p_{22} \sin(\theta_2 + \theta_3) + 50 p_{10} = p_{13} \quad (5.1.12)$$

$$III : 25 p_{12} + 70.05 p_{12} \sin(\theta_2) + 79.4 p_{12} \sin(\theta_2 + \theta_3) + 50 p_{20} = p_{23} \quad (5.1.13)$$

Die Gleichungen zwei bzw. drei werden nach θ_3 umgeformt:

$$II : \theta_3 = \arcsin \left(\frac{p_{13} - 50 p_{10} - 25 p_{22} - 70.05 p_{22} \sin(\theta_2)}{79.4 p_{22}} \right) - \theta_2 \quad (5.1.14)$$

bzw.

$$III : \theta_3 = \arcsin\left(\frac{50 p_{20} - p_{23} - 25 p_{12} - 70.05 p_{12} \sin(\theta_2)}{79.4 p_{12}}\right) - \theta_2 \quad (5.1.15)$$

Nun wird in Gleichung eins eingesetzt und es ergibt sich daraus eine Gleichung mit einer Unbekannten.

$$II \rightarrow I : 79.4 \cos\left[\arcsin\left(\frac{p_{13} - 50 p_{10} - 25 p_{22} - 70.05 p_{22} \sin(\theta_2)}{79.4 p_{22}}\right)\right] + 70.05 \cos(\theta_2) = p_{03} - 46.5 - 50 p_{00} \quad (5.1.16)$$

bzw.

$$III \rightarrow I : 79.4 \cos\left[\arcsin\left(\frac{p_{13} - 50 p_{10} - 25 p_{22} - 70.05 p_{22} \sin(\theta_2)}{79.4 p_{12}}\right)\right] + 70.05 \cos(\theta_2) = p_{03} - 46.5 - 50 p_{00} \quad (5.1.17)$$

Der Term p_{22} entspricht $\cos(\theta_1)$, abhängig von Winkel θ_1 kann dieser Term null werden. Somit würde sich eine Division durch null ergeben. Das Gleiche gilt für p_{12} . Aus diesem Grund muss vor der Berechnung von θ_2 überprüft werden ob die Gleichung 5.1.16 oder 5.1.17 verwendet werden muss.

Die Gleichungen sind nun soweit aufgelöst, dass alle Winkel berechnet werden können.

5.2 Algorithmus zum Berechnen der Winkeln

Der Berechnungsalgorithmus wird auf einem Mikrocontroller implementiert. Alle dazu nötigen Funktionen sind in der Bibliothek math.h gegeben.

Der Winkel θ_1 kann direkt berechnet werden. θ_2 wird mit Hilfe der Gleichungen 5.1.16 und 5.1.17 berechnet. Hierfür wird das Ergebnis der rechten Seite berechnet. Im Anschluss wird der Winkelbereich für θ_2 von 0 bis 180 Grad durchlaufen, das Ergebnis für die linke Seite berechnet und mit dem Ergebnis der rechten Seite verglichen. Entspricht das Resultat der rechten Seite dem der linken, natürlich mit geringen Abweichungen, ist ein Ergebnis gefunden. Im Anschluss kann θ_3 und θ_4 berechnet werden.

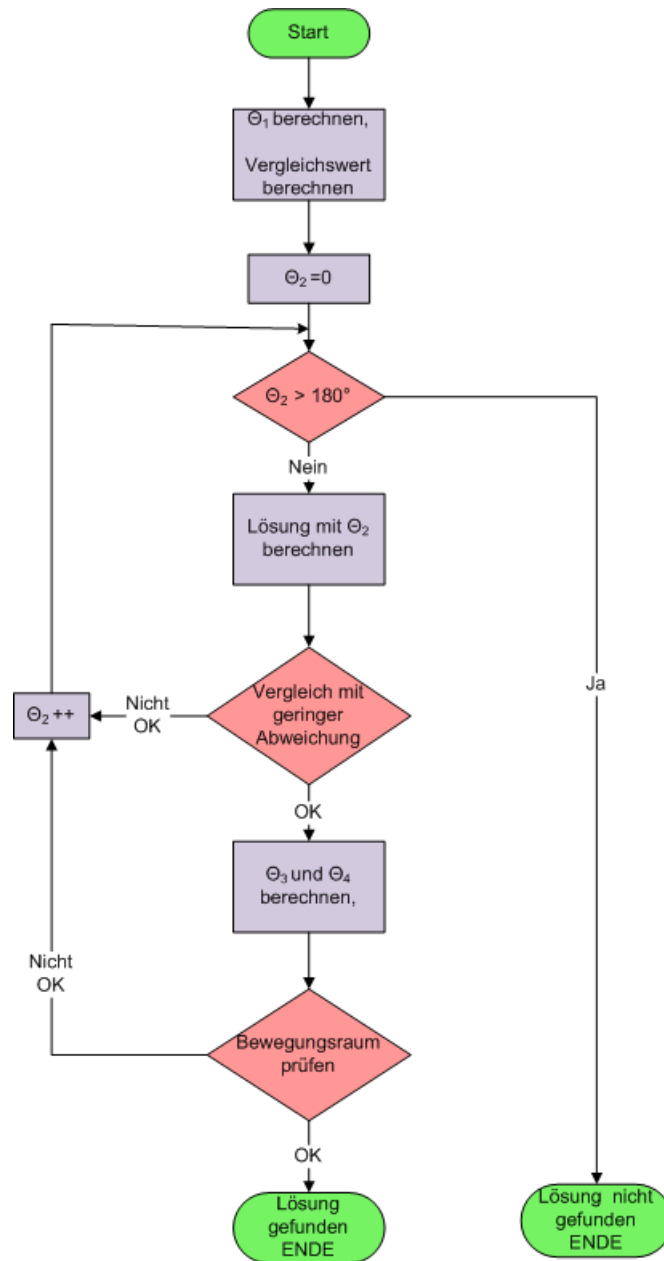


Abbildung 5.2.1: Funktionsweise des inversen Algorithmus

Kapitel 6

Der Greifertask

Die Berechnung der Winkel sowie die Ansteuerung der Achsen übernimmt der Greifertask. Der Task ist ins Multitaskingsystem eingebunden und wird von einem hierarchisch übergeordneten Task aufgerufen. Der Task selbst arbeitet als Statemaschine.

6.1 Programmübersicht

Wird der Task aufgerufen, wird abgefragt welches Element aufzunehmen ist. Die Kommunikation und der Datenaustausch zwischen den einzelnen Tasks erfolgt über eine Mailbox. Für jeden Task sind fix zehn Byte Speicher reserviert, welche beschrieben und ausgelesen werden können. Nachdem die Elementnummer ausgelesen wurde, wird der Tomatenzweig oder der Orangenweig ausgeführt.

Die Grundfunktionen beider Zweige sind ident. Wesentliche Unterschiede gibt es bei Berechnungen und Ansteuerung. Die Hauptfunktionen des Tasks sind:

- Berechnungen durchführen
- Zielpunkte anfahren
- Entleerpositionen anfahren

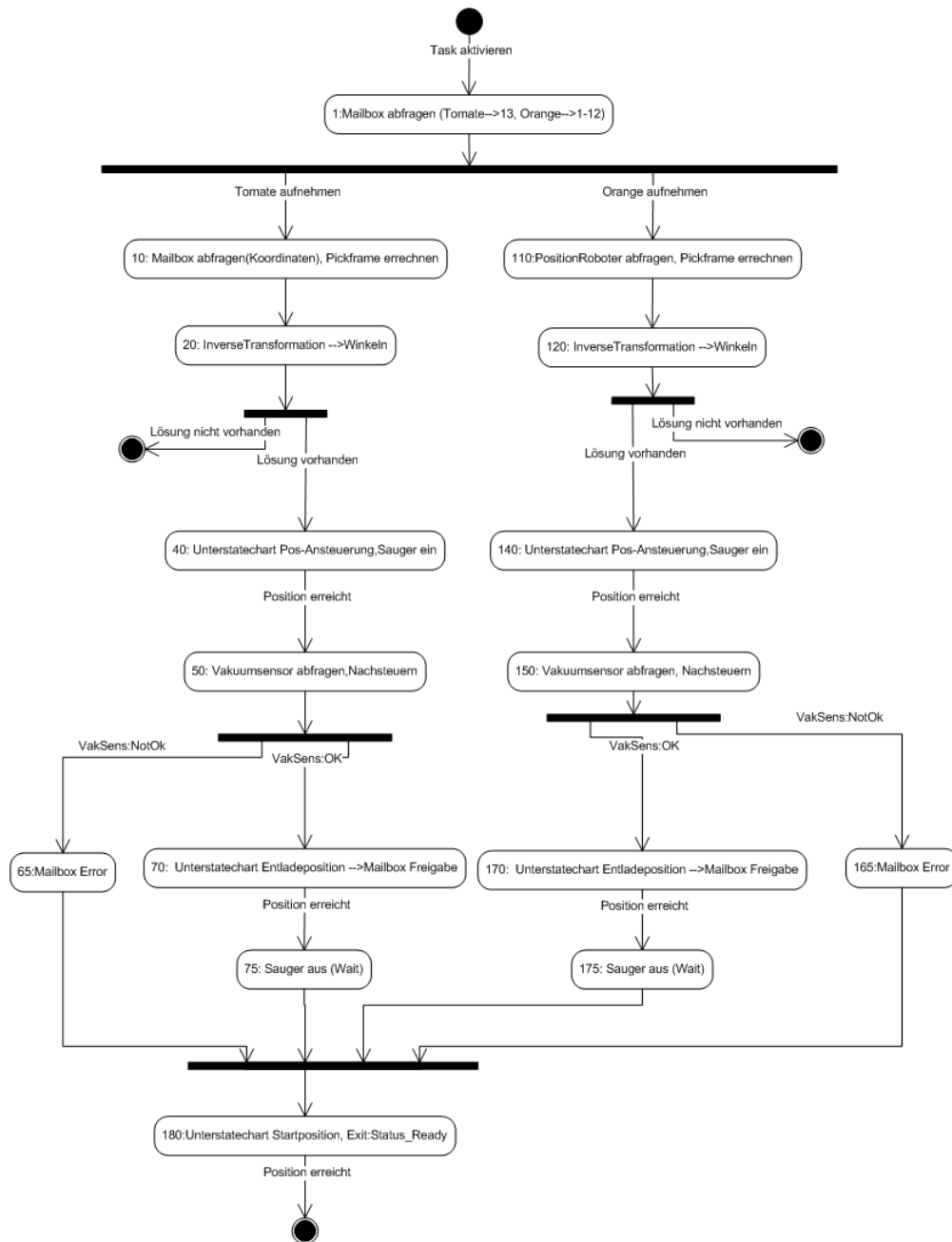


Abbildung 6.1.1: Statechart Greifertask

Nach Abarbeitung eines States wird eine Zykluszeit zum Wiederaufruf des Tasks gesetzt und die Berechtigung abgegeben. Die Zykluszeit variiert zwischen 50 Millisekunden und einer Sekunde. Dies ist abhängig vom aktuellen bzw. folgendem State. Aufeinander folgende Berechnungen können mit sehr geringer Zykluszeit durchgeführt werden. Die Ansteuerung von Servos, Ventilen oder der Unterdruckaufbau kann länger dauern.

Bei den Ausstiegspunkten der Statemaschine werden Statusmeldungen abgegeben, der Task deaktiviert sich selbst.

Kapitel 7

Lessons Learned

Der Verlauf des Projektes lag weit von den Erwartungen entfernt. Im Projektteam wurde über den Verlauf des Projektes gesprochen und die wichtigsten Faktoren ergründet die zum Scheitern des Projektes geführt hatten.

1. Planung - Wirklichkeit - Abweichung

Was war geplant?	Was geschah wirklich?	Warum Abweichung?
Einen konkurrenzfähigen, voll ausgetesteten Roboter zu entwickeln	Roboter war beschränkt funktionsfähig, wenige Funktionen getestet	Fehlerhafte vorhandene Hardware, dadurch Zeitverlust

Tabelle 7.0.1: Planung - Wirklichkeit - Abweichung

2. Punkte bei denen Verbesserungsbedarf erkannt wurde

Stichwort	Beschreibung
Organisation	Es dauerte zu lange bis erste Tests durchgeführt wurden
Modulares System (el.)	Eignet sich sehr gut für schnelle Testaufbauten, verbraucht zu viel Platz
Elektronik	War nicht ausreichend dimensioniert (z.B.: Servoboard),
Antrieb	Schneckenantrieb war eine Fehlplanung, die Motoren waren nicht für die Belastungen ausgelegt.

Tabelle 7.0.2: Verbesserungsbedarf

3. Punkte die gut liefen

Stichwort	Beschreibung
Team	großer Zusammenhalt und angenehmes Arbeitsklima, es wurde niemals aufgegeben

Tabelle 7.0.3: Positive Aspekte

4. Verbesserungsvorschläge

Stichwort	Beschreibung
Organisation	FUP im fünften Semester verstärkt für Tests und Entwicklung nutzen
Modulare System (el.)	Keine Endlösung, nur zu Testzwecken
Elektronik	Anforderungen müssen geprüft werden und Elektronik darauf auslegen; Eine Grundplatine aus den Erfahrungen mit dem modularen System entwickeln
Antrieb	Neuen Antrieb planen: Eine durchgehende Grundplatte; Verstiften aller Elemente (Motoren, Inkrementalgeber,...), damit die Verschraubungsposition fix ist; Übersetzung mittels Zahnriemen oder Zahnräder

Tabelle 7.0.4: Verbesserungsvorschläge

5. Schriftliches Resümee

Der Roboter war nur zum Teil funktionsfähig und viele Module des Roboters wenig bis gar nicht getestet. Dies ist darauf zurückzuführen, dass der fremdentwickelte Antrieb nicht auf die Belastungen ausgelegt war. Einen großen Teil der vorhandenen Zeit wurde für Reparaturen bzw. zur Neuplanung des Antriebs aufgewendet. Da die zur Verfügung stehende Zeit von Beginn des Berufspraktikums bis zum Bewerb sehr knapp war, konnte vieles nicht mehr realisiert werden. Auch die Elektronik muss besser auf die Anforderungen ausgelegt werden.

Um bei den Greifarmen eine korrekte Positionierung zu garantieren, müssen diese mechanisch steif und spielfrei sein. Wenige Grade Abweichung in den einzelnen Achsen können am Endeffektor Abweichungen im Zentimeterbereich hervorrufen.

Das Vertrauen in bereits vorhandene Module war zu groß. Man muss sich selbst im Vorhinein besser mit den zu Verfügung gestellten Komponenten beschäftigen und überprüfen ob diese den Anforderungen entsprechen.

Literaturverzeichnis

- [1] (2009, September) Feed the world - rules 2010. [Online]. Available: <http://www.eurobot.org>
- [2] M. Pautzenberger. (2010) Konzeptionierung, konstruktion und aufbau der mechanik eines mobilen autonomen roboters. [Online]. Available: <http://rrt.fh-wels.at/publications.html>
- [3] G. S. Jörg Bartenschlager, Hans Hebel, *Handhabungstechnik mit Robotertechnik*, Seite 302, 1998.
- [4] (2010, Mai) Denavit-hartenberg.pdf. [Online]. Available: <http://homepages.fh-giessen.de/~hg6458/Robotik/Denavit-Hartenberg.pdf>