

# myRIO - Balancing Robot - Mathematics - DRAFT

Adrian Schmid

28. Oktober 2013

## Overview

In order to correctly understand the basics underneath myRIO Balancing Robot, several steps towards the final model are presented in this paper.

## Inhaltsverzeichnis

<b>1</b>	<b>myBOT und die Geometrie</b>	<b>3</b>
1.1	myBOT . . . . .	3
1.2	Das absolute Koordinatensystem . . . . .	4
1.3	myBOT im Raum . . . . .	4
1.4	Reduzierung der Freiheitsgrade . . . . .	5
1.4.1	$r$ in Abhängigkeit von $\gamma$ und $\alpha$ . . . . .	5
1.4.2	$\beta$ in Abhängigkeit von $\gamma$ . . . . .	5
1.4.3	$\Theta$ in Abhängigkeit von $\gamma$ . . . . .	6
1.4.4	Vereinfachte Positionsdarstellung . . . . .	6
1.4.5	Vereinfachte Positionsdarstellung im absoluten Koordinatensystem . . . . .	6
<b>2</b>	<b>Kinematik</b>	<b>6</b>
<b>3</b>	<b>Handhabung in der Praxis</b>	<b>7</b>
3.1	Physikalische Parameter des myBOT . . . . .	7
3.2	Verifikation in LabVIEW . . . . .	7
<b>4</b>	<b>Energiegleichungen</b>	<b>8</b>
4.1	Potenzielle Energie des Massepunktes . . . . .	8
4.2	Kinetische translatorische Energie des Massepunktes . . . . .	8
4.3	Kinetische rotative Energie des Massepunktes . . . . .	8
4.4	Kinetische translatorische Energie der Räder . . . . .	8
4.5	Kinetische rotative Energie der Räder . . . . .	8
<b>5</b>	<b>Reglerkonzept</b>	<b>8</b>
5.1	Regelkreis 1: $\alpha$ . . . . .	8
5.1.1	Mögliche Regelkonzepte . . . . .	8
5.1.2	Balance Controller Feder-Dämpfer-System . . . . .	10
5.1.3	Anti-Windup . . . . .	10
5.1.4	Tuning . . . . .	10
5.2	Geschwindigkeitsregelung . . . . .	10
5.3	Positionregelung . . . . .	10

<b>6</b>	<b>Bestimmung von <math>\alpha</math></b>	<b>10</b>
6.1	Accelerometer . . . . .	11
6.1.1	Winkelmessung mit dem Accelerometer . . . . .	11
6.1.2	1-achsige Winkelmessung . . . . .	11
6.2	Gyroscope . . . . .	11
6.3	Komplementärfilter . . . . .	11
<b>7</b>	<b>SW Implementierung</b>	<b>11</b>
7.1	SW Architektur . . . . .	11
7.2	FPGA Architektur . . . . .	11
<b>8</b>	<b>Übertragungsfunktionen des myBOT</b>	<b>12</b>
8.1	1 DOF . . . . .	12
8.1.1	2 DOF . . . . .	15
8.1.2	Vereinfachung aufgrund der Energieanteile . . . . .	15
8.2	Simulation in LabVIEW . . . . .	16
<b>9</b>	<b>Komplette Form</b>	<b>17</b>
9.1	Simulation in LabVIEW . . . . .	17
<b>10</b>	<b>der myBOT Im Zustandsraum</b>	<b>17</b>
10.1	Das allgemeine Zustandsraummodell . . . . .	18
10.2	Zustandsraummodell für den vereinfachten myBOT . . . . .	18
10.2.1	Simulation in LabVIEW . . . . .	19
10.3	Zustandsraummodell des kompletten myBOT . . . . .	19
10.3.1	Simulation in LabVIEW . . . . .	20
10.4	Erweiterung auf 3DOF . . . . .	21
10.5	Erweiterung um Nichtlinearitäten . . . . .	21

# 1 myBOT und die Geometrie

Der myBOT kann im Raum vollständig beschrieben werden durch die Verwendung zweier Koordinatensystem, wie gezeigt in Figur: 5. Diese Darstellung soll nun Schritt um Schritt hergeleitet und beschrieben werden.

## 1.1 myBOT

Der myBOT selbst kann beschrieben werden durch seine wichtigsten Elemente. Zwei Räder mit dem Radius  $R$  sind verbunden durch eine Achse der Länge  $L$ . Die Räder weisen eine Masse auf von je  $\frac{M}{2}$ , die gesamte Masse der Räder ist deshalb  $M$ . Der Körper des myBOT wird durch eine Punktmasse beschrieben. Die Punktmasse hat die Masse  $m$ . Sie ist über einen masselosen Stab der Länge  $l$  mit der Achse verbunden. Der myBOT spannt ein relatives Koordinatensystem  $C_{rel}\{u, v, w\}$  auf. Die  $v$ -Achse liegt dabei in Richtung der Radachse. Die  $u$ -Achse und die  $z$ -Achse spannen eine dazu senkrechte Ebene auf. Der myBOT in seinem Koordinatensystem ist dargestellt in Figur: 1. Der myBOT besitzt in diesem Koordinatensystem nun zwei rotative Freiheitsgrade

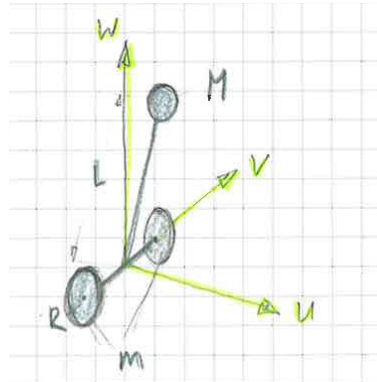


Abbildung 1: myBOT im Raum mit Relativem Koordinatensystem

(2DOF). Zum einen kann sich der Körper, dargestellt durch die Punktmasse, um die  $v$ -Achse drehen. Diese Drehung wird beschrieben durch den Winkel  $\alpha$ . Des weiteren kann ein beliebiger Punkt auf dem Radumfang sich um die  $v$ -Achse drehen. Diese Drehung wird beschrieben durch den Winkel  $\gamma$ . Siehe Figur: 4.

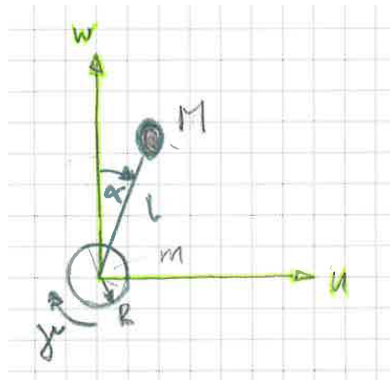


Abbildung 2: myBOT Seitenansicht

## 1.2 Das absolute Koordinatensystem

Um die Position des myBOT im Raum zu beschreiben ist nun ein zweites, absolutes Koordinatensystem notwendig,  $C_{abs}\{x, y, z\}$ . Das relative Koordinatensystem des myBOT  $C_{rel}\{u, v, w\}$  besitzt in diesem Koordinatensystem drei Freiheitsgrade (3DOF). Zum einen kann es sich translatorisch in der  $xy$ -Ebene bewegen (2DOF). Zusätzlich kann es sich rotatorisch um die  $z$ -Achse drehen. Die Position des Ursprungs des relativen Koordinatensystems auf der  $xy$ -Ebene kann dabei durch einen Punkt  $P_i(x_i, y_i)$  beschrieben werden. Dieselbe Position kann auch in Polarkoordinaten angegeben werden  $P_i(r_i, \Theta_i)$ . Die Rotation um die  $z$ -Achse wird mit dem Winkel  $\beta$  beschrieben. Siehe Figur: 3 Die komplette Lage des relativen Koordinatensystems  $C_{rel}\{u, v, w\}$  im absoluten Koordinaten-

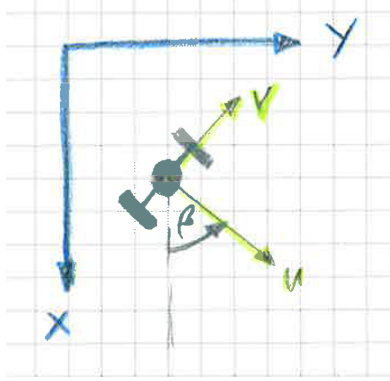


Abbildung 3: myBOT Aufsicht

system  $C_{abs}\{x, y, z\}$  kann dabei durch einen Vektor  $A_i(x_i, y_i, \beta_i)$  beschrieben werden. Siehe Figur: 5.

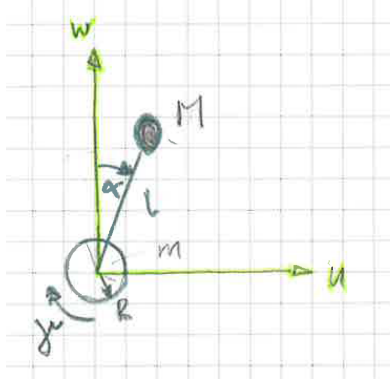


Abbildung 4: myBOT Seitenansicht

## 1.3 myBOT im Raum

Die gesamte Position und Lage des myBOT kann nun angegeben werden, zusätzlich zum Verschiebungsvektor  $A_i(x_i, y_i, \beta_i)$  noch die beiden Winkel  $\alpha$  und  $\gamma$  betrachtet werden. Somit kann die komplette Position und Lage des myBOT im Raum durch den Vektor  $myPOS_i = \{x_i, y_i, \alpha_i, \beta_i, \gamma_i\}$  bzw.  $myPOS_i = \{r_i, \theta_i, \alpha_i, \beta_i, \gamma_i\}$  beschrieben werden. Für die nächsten Betrachtungen wird die Polardarstellung verwendet.

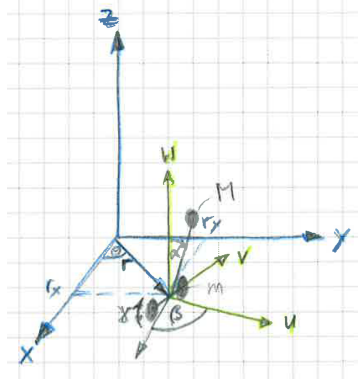


Abbildung 5: myBOT im Raum mit relativem und absolutem Koordinatensystem

## 1.4 Reduzierung der Freiheitsgrade

Es wird angenommen, dass sich die Räder des myBOT ohne Schlupf auf der Ebene bewegen. Dadurch entsteht naturgemäss eine Abhängigkeit zwischen den Freiheitsgraden.

### 1.4.1 $r$ in Abhängigkeit von $\gamma$ und $\alpha$

Die translatorische Positionskomponente  $r$  kann durch  $\gamma$  und  $\alpha$  ausgedrückt werden. So gilt

$$r = \int_0^T \dot{r} dt = R \cdot \int_0^T (\dot{\gamma} + \dot{\alpha}) dt \quad (1)$$

Da  $\gamma$  den Winkel in der Mitte der Radachse bezeichnet, der myBOT jedoch auf zwei Rädern links und rechts der Achse balanciert gilt weiter für  $\gamma$ .

$$\gamma = \frac{\gamma_l + \gamma_r}{2} = \frac{1}{2}\gamma_l + \frac{1}{2}\gamma_r \quad (2)$$

So kann  $\dot{\gamma}$  geschrieben werden als

$$\dot{\gamma} = \frac{1}{2}\dot{\gamma}_l + \frac{1}{2}\dot{\gamma}_r \quad (3)$$

$r$  kann deshalb geschrieben werden als

$$r = R \cdot \int_0^T \left( \frac{1}{2}\dot{\gamma}_l + \frac{1}{2}\dot{\gamma}_r + \dot{\alpha} \right) dt \quad (4)$$

Das heisst, die translatorische Position  $r$  des myBOT im Raum kann durch die drei Variablen  $\alpha, \gamma_l$  und  $\gamma_r$  beschrieben werden.

### 1.4.2 $\beta$ in Abhängigkeit von $\gamma$

Der Drehwinkel  $\beta$  kann geschrieben werden als

$$\beta = \int_0^T \dot{\beta} dt = \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \quad (5)$$

### 1.4.3 $\Theta$ in Abhängigkeit von $\gamma$

Die rotatorische Positionskomponente  $\Theta$  kann nun geschrieben werden als

$$\Theta = \int_0^T \beta \dot{r} dt = \int_0^T \left\{ \left[ \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \right] \cdot \left[ \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} \right] \right\} \cdot dt \quad (6)$$

### 1.4.4 Vereinfachte Positionsdarstellung

Der Vektor  $myPOS$  welcher die gesamte Lage des myBOT im Raum angibt kann nun vereinfacht geschrieben werden, abhängig einzig von  $\alpha$  und  $\gamma$ .

$$myPOS_{polar} = \begin{bmatrix} r \\ \Theta \\ \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} R \cdot \int_0^T \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} dt \\ \int_0^T \left\{ \left[ \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \right] \cdot \left[ \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} \right] \right\} \cdot dt \\ \alpha \\ \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \\ \frac{1}{2} \gamma_l + \frac{1}{2} \gamma_r \end{bmatrix} \quad (7)$$

Dies bedeutet, dass die vollständige Lage des myBOT im Raum einzig durch  $\alpha$ ,  $\gamma_l$ ,  $\gamma_r$  und deren Ableitungen nach der Zeit beschrieben werden kann.

### 1.4.5 Vereinfachte Positionsdarstellung im absoluten Koordinatensystem

Durch geeignete Rücktransformation kann nun  $myPOS$  auch in das absolute Koordinatensystem zurückgerechnet werden, was insbesondere für Positionierungsaufgaben interessant ist.

$$myPOS_{abs} = \begin{bmatrix} x \\ y \\ \alpha \\ \beta \\ \gamma \end{bmatrix} = \begin{bmatrix} r \cdot \cos(\Theta) \\ r \cdot \sin(\Theta) \\ \alpha \\ \beta \\ \gamma \end{bmatrix} = \quad (8)$$

$$\begin{bmatrix} R \cdot \int_0^T \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} dt \cdot \cos\left(\int_0^T \left\{ \left[ \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \right] \cdot \left[ \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} \right] \right\} \cdot dt \right) \\ R \cdot \int_0^T \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} dt \cdot \sin\left(\int_0^T \left\{ \left[ \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \right] \cdot \left[ \left( \frac{1}{2} \dot{\gamma}_l + \frac{1}{2} \dot{\gamma}_r \right) \dot{\alpha} \right] \right\} \cdot dt \right) \\ \alpha \\ \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_r \cdot R}{L}\right) \right) dt - \int_0^T \left( \arctan\left(\frac{\dot{\gamma}_l \cdot R}{L}\right) \right) dt \\ \frac{1}{2} \gamma_l + \frac{1}{2} \gamma_r \end{bmatrix} \quad (9)$$

## 2 Kinematik

Für die Beschreibung der Bewegung des myBOT kann nun der Vektor  $myPOS_i = \{r_i, \theta_i, \alpha_i, \beta_i, \gamma_i\}$  komponentenweise abgeleitet werden. So entsteht ein Geschwindigkeitsvektor  $myVEL_i = \{\dot{r}_i, \dot{\theta}_i, \dot{\alpha}_i, \dot{\beta}_i, \dot{\gamma}_i\}$  welcher die Geschwindigkeit des myBOT im Raum vollständig beschreibt. Des weiteren könnte durch abermalige komponentenweise Ableitung ein Beschleunigungsvektor  $myACC_i = \{\ddot{r}_i, \ddot{\theta}_i, \ddot{\alpha}_i, \ddot{\beta}_i, \ddot{\gamma}_i\}$

gewonnen werden, welcher die Beschleunigung des myBOT im Raum vollständig beschreibt. Diese Darstellung wird jedoch nicht weiterverfolgt.  $myVEL$  kann dabei folgendermassen geschrieben werden

$$myVEL = \begin{bmatrix} \dot{r} \\ \dot{\Theta} \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} R \cdot (\frac{1}{2}\dot{\gamma}_l + \frac{1}{2}\dot{\gamma}_r) \dot{\alpha} \\ \int_0^T \left( \arctan(\frac{\dot{\gamma}_r \cdot R}{L}) \right) dt - \int_0^T \left( \arctan(\frac{\dot{\gamma}_l \cdot R}{L}) \right) dt \cdot [(\frac{1}{2}\dot{\gamma}_l + \frac{1}{2}\dot{\gamma}_r) \dot{\alpha}] \\ \dot{\alpha} \\ \arctan(\frac{\dot{\gamma}_r \cdot R}{L}) - \arctan(\frac{\dot{\gamma}_l \cdot R}{L}) \\ \frac{1}{2}\dot{\gamma}_l + \frac{1}{2}\dot{\gamma}_r \end{bmatrix} \quad (10)$$

### 3 Handhabung in der Praxis

In der Praxis können vier Sensoren hergezogen werden für die Bestimmung der Lage und Geschwindigkeit des myBOT. Es sind dies der Encoder des linken und des rechten Motors, sowie ein 3-achsiger Accelerometer und ein 3-achsiges Gyroscope. Das heisst direkt gemessen werden können

**Winkel**  $\gamma_l$  gemessen durch den Encoder des linken Motors

**Winkel**  $\gamma_r$  gemessen durch den Encoder des rechten Motors

**Winkel**  $\alpha$  gemessen durch eine Kombination aus Gyroscope und Accelerometer, sowie geeignete Filterung

**Winkelgeschwindigkeit**  $\dot{\alpha}$  gemessen durch das Gyroscope

#### 3.1 Physikalische Parameter des myBOT

$$R = 35mm, M = 40g, L = 167mm, m = 650g, l = 50mm, g = 9.81 \frac{m}{s^2}$$

#### 3.2 Verifikation in LabVIEW

Anhand des Prototypen sowie eines geeigneten Programmes sollen die oben hergeleiteten Vektoren  $myPOS_{abs}$ ,  $myPOS_{polar}$  und  $myVEL$  verifiziert werden.

Vorgeschlagen wird dabei das folgende Vorgehen

1. Herleitung von  $myVEL$ . Für die Bildung von  $myVEL$  wird die folgende Reihenfolge vorgeschlagen
  - a) VI für die Bestimmung von  $\dot{\gamma}_l$  und  $\dot{\gamma}_r$ . Aus diesen Werten können  $\dot{\gamma}$  und  $\dot{\beta}$  berechnet werden.
  - b) VI für die Bestimmung von  $\dot{\alpha}$ . Mit diesem zusätzlichen Wert können nun auch  $\dot{r}$  und  $\dot{\Theta}$  berechnet werden. Somit ist  $myVEL$  vollständig bestimmt.
2. Herleitung von  $myPOS_{polar}$ . Für die Bildung von  $myPOS_{polar}$  wird die folgende Reihenfolge vorgeschlagen
  - a) VI für die Bestimmung von  $\alpha$ .  $\alpha$  soll dabei nicht durch numerische Integration aus  $\dot{\alpha}$ , sondern durch eine direkte Bestimmung aus dem Accelerometer und dem Gyroscope, sowie einer geeigneten Filterung gewonnen werden.
  - b) Danach können mit dem so gewonnenen Wert  $\alpha$ , sowie dem Vektor  $myVEL$  alle Werte berechnet werden.
3. Herleitung von  $myPOS_{abs}$ .  $myPOS_{abs}$  kann durch eine einfache Transformation beschrieben in Formel 8 hergeleitet werden.

Alle benötigten Integrale werden dabei iterativ, durch numerische Integration berechnet.

## 4 Energiegleichungen

Aufstellen der Energiegleichungen des myBOT im Raum gemäss oben. Dabei sind folgende Teilenergien vorstellbar.

1. Potenzielle Energie des Massepunktes, abhängig von  $l, \alpha, m$
2. Kinetische translatorische Energie des Massepunktes
3. Kinetische rotative Energie des Massepunktes
4. Kinetische translatorische Energie der Räder
5. Kinetische rotative Energie der Räder

Die Energien müssen dabei im Raum betrachtet werden, ausser man vereinfacht diesbezüglich, und nimmt an, dass die Energien aufgrund der Drehbewegungen sehr klein sind und vernachlässigt werden können.

### 4.1 Potenzielle Energie des Massepunktes

$$P_{Body} = mgl \cos \alpha \quad (11)$$

### 4.2 Kinetische translatorische Energie des Massepunktes

$$K_{T_{Body}} = \frac{1}{2} m \cdot \dot{r}^2 \quad (12)$$

### 4.3 Kinetische rotative Energie des Massepunktes

$$K_{R_{Body}} = \frac{1}{2} J_{Body} \cdot \dot{\alpha}^2 = \frac{1}{2} ml^2 \cdot \dot{\alpha}^2 \quad (13)$$

### 4.4 Kinetische translatorische Energie der Räder

$$K_{T_{Wheels}} = \frac{1}{2} M \cdot \dot{r}^2 \quad (14)$$

### 4.5 Kinetische rotative Energie der Räder

$$K_{R_{Wheels}} = \frac{1}{2} J_{Wheels} \cdot \dot{\gamma}^2 \approx \frac{1}{2} (MR^2) \cdot \left( \frac{\dot{r}}{R} \right)^2 = \frac{1}{2} M \dot{r}^2 \quad (15)$$

## 5 Reglerkonzept

Wie oben dargestellt sind im myBOT verschiedene Regelgrößen zu regeln.

Der innerste Regelkreis balanciert den myBOT.  $\alpha$  ist dabei die Grösse mit der höchsten Priorität, da  $\alpha$  bestimmt, ob der myBOT balanciert oder umfällt.

Darüber geschaltet wird ein zweiter Regelkreis, welcher die Geschwindigkeit des myBOT im Raum regelt. Dabei werden  $\dot{r}$  &  $\dot{\beta}$  geregelt.

Als dritter und äusserster Regelkreis wird die Position des myBOT auf der  $xy$ -Ebene geregelt.

Abbildung 6 stellt dieses Reglerblockschaltbild dar.

### 5.1 Regelkreis 1: $\alpha$

#### 5.1.1 Mögliche Regelkonzepte

P, PI, BalanceController Feder-Dämpfer-System



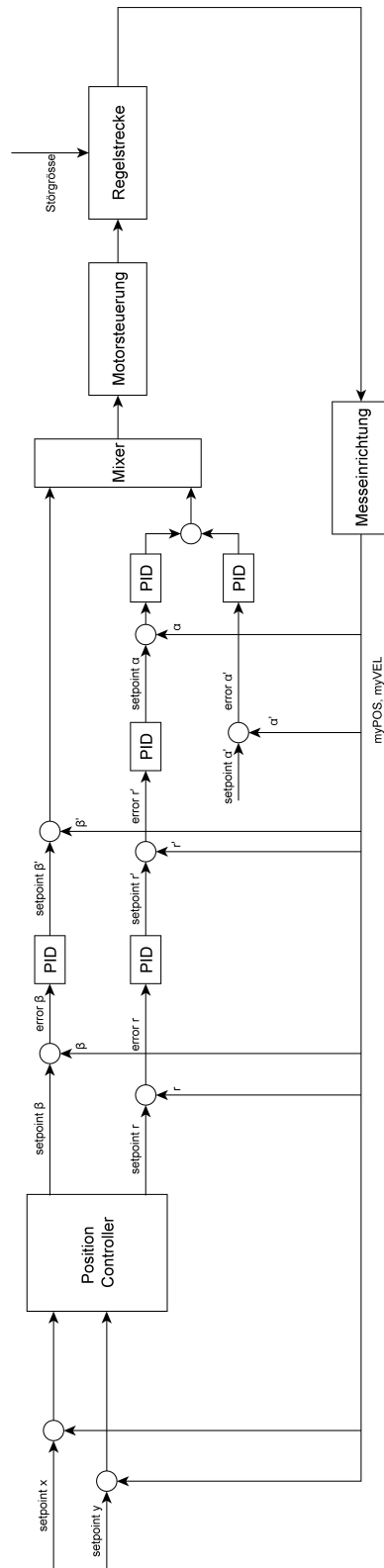


Abbildung 6: Reglerkonzept für den myBOT

### 5.1.2 Balance Controller Feder-Dämpfer-System

Ein Feder-Dämpfer-System wie abgebildet (Abbildung: 7) wird genutzt, um den myBOT in der Vertikalen ( $\alpha = 0$ ) zu halten.

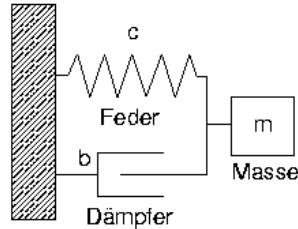


Abbildung 7: Feder-Dämpfer-System in der Mechanik

Dabei wird die Federkonstante als  $K_p$  bezeichnet, die Dämpferkonstante als  $K_d$ . Der Output ergibt sich somit wie folgt. Gleichung für den Output:

$$Output = \int (K_p \cdot \alpha + K_d \cdot \dot{\alpha}) dt \quad (16)$$

### 5.1.3 Anti-Windup

Eine Begrenzung des Outputs ist notwendig, ansonsten kann der Regler sich gegen Unendlich verabschieden.

### 5.1.4 Tuning

**Manuelles Tuning**  $K_p$  erhöhen, bis System oszilliert, dann  $K_i$  erhöhen, bis Overshoot und Steady-state error in akzeptablem Mass.

**Ziegler-Nichols**  $K_p$  erhöhen, bis System oszilliert. dann gemäss Tabelle werte berechnen. für PI Regler:  $K_p = 0.45 \cdot K_u$ ,  $K_i = 1.2 \cdot K_p / P_u$  ( $P_u$ =oscillation period)

## 5.2 Geschwindigkeitsregelung

Die zweite Regelstufe regelt die Geschwindigkeit des myBOT. Insbesondere sind dies die Werte  $\dot{r}$  und  $\dot{\beta}$ . Für beide Werte wird eine PI-Regelung vorgeschlagen. Beide Regler müssen in Sättigung gehen. (Zu grosse Werte für  $\dot{r}$  und  $\dot{\beta}$  führen zu Unstabilität bei Regelkreis 1.) Der I-Anteil soll zusätzlich durch ein Anti-Windup begrenzt werden.

## 5.3 Positionsregelung

Die dritte Regelstufe regelt die Position des myBOT. Insbesondere sind dies die Werte  $r$  und  $\Theta$ , bzw.  $x$  und  $y$ . Für beide Werte wird eine Proportional-Regelung vorgeschlagen. Aus der Differenz zwischen Ist- und Sollposition wird ein Fahrvektor  $\{r, \beta\}$  gebildet.

## 6 Bestimmung von $\alpha$

Zur Bestimmung von  $\alpha$  stehen zwei Sensoren zur Verfügung, der 3-achsige Accelerometer, integriert im myRIO und der externe 3-achsige Gyroscope.

Im Folgenden sollen beide Sensoren kurz vorgestellt werden, und die jeweiligen Vor- und Nachteile aufgelistet werden.

## 6.1 Accelerometer

TBD

### 6.1.1 Winkelmessung mit dem Accelerometer

#### 6.1.2 1-achsige Winkelmessung

grafik, accelerometer In Ruhelage wirken keine Kräfte auf den Accelerometer ausser der Gravitation. So kann über das Verhältnis von der vom Accelerometer gemessenen Kraft im Verhältnis zur Gravitation der Winkel berechnet werden. Dabei gilt:

$$\sin \alpha = \frac{Acc_z}{g}, bzw. \alpha = \arcsin \frac{Acc_z}{g} \quad (17)$$

Für einfache Winkel  $\alpha \leq 30 \text{ deg}$  gilt näherungsweise  $\sin \alpha \approx \alpha$   
Problem am Accelerometer. Nimmt Vibrationen auf. Verfälscht Winkelmessung. Nur Longtime-Average nützlich.

## 6.2 Gyroscope

Misst die Winkelgeschwindigkeit  $\dot{\alpha}$ .  $\alpha$  kann durch numerische Integration gewonnen werden. Problem am Gyro. Durch die numerische Integration bildet sich aufgrund des Offsets (Gyro  $\neq 0$  ohne Bewegung) ein Fehler. Der Gyro läuft auf die lange Distanz davon. Die Änderungen des Gyrowinkels über kurze Zeiten sind jedoch sehr genau.

## 6.3 Komplementärfilter

Deshalb Komplementärfilter.

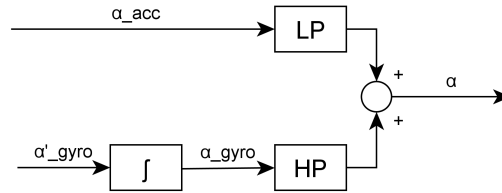


Abbildung 8: Komplementärfilter

Feder- Dämpfer System, Tiefpass auf Accelerometerwinkel, Hochpass auf Gyroscopewinkel.

$$\tau = \frac{A \cdot dt}{1 - A} \Leftrightarrow A = \frac{\tau}{\tau + dt} \quad (18)$$

Übertragungsfunktion des Komplementärfilters

$$\alpha_{n+1} = A \cdot \left( \alpha_n + \int \dot{\alpha} dt \right) + (1 - A) \cdot \alpha_{acc} \quad (19)$$

Erste Tests mit dem Komplementärfilter

## 7 SW Implementierung

### 7.1 SW Architektur

### 7.2 FPGA Architektur

Herleitung von myPOS und myVEL auf dem FPGA.

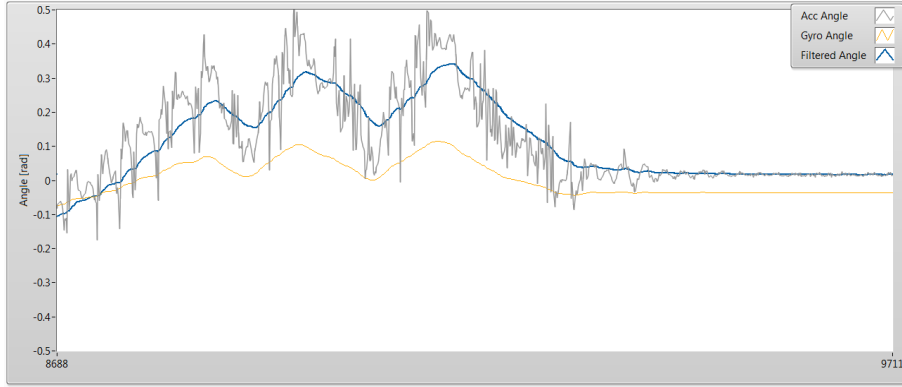


Abbildung 9: Erste Tests mit dem Komplementärfilter

## 8 Übertragungsfunktionen des myBOT

Nach dem Lagrange Verfahren. Der Lagrangian wird wie folgt gebildet.

$$\mathcal{L} = K - P = E_{kin} - E_{pot} \quad (20)$$

### 8.1 1 DOF

$$\mathcal{L} = K - P = K_{Body} - P_{Body} = \frac{1}{2}ml^2 \cdot \dot{\alpha}^2 - mgl \cos \alpha \quad (21)$$

Nicht-konservatives System. Es liegt ein externes Moment sowie Reibung vor.

$$\frac{\partial}{\partial t} \cdot \left( \frac{\partial \mathcal{L}}{\partial \dot{\alpha}} \right) - \frac{\partial \mathcal{L}}{\partial \alpha} = T_M + Q_\alpha \quad (22)$$

Die Reibung  $Q_\alpha$  ist definiert als

$$Q_\alpha = -\mu v = -\mu l \dot{\alpha} \quad (23)$$

Bildung der Terme auf der linken Seite

$$\frac{\partial \mathcal{L}}{\partial \alpha} = mgl \sin \alpha \quad (24)$$

$$\frac{\partial \mathcal{L}}{\partial \dot{\alpha}} = ml^2 \dot{\alpha} \quad (25)$$

$$\frac{\partial}{\partial t} \cdot \left( \frac{\partial \mathcal{L}}{\partial \dot{\alpha}} \right) = ml^2 \ddot{\alpha} \quad (26)$$

D.h. die Übertragungsfunktion wird zu

$$ml^2 \ddot{\alpha} - mgl \sin \alpha = T_M - \mu l \dot{\alpha} \quad (27)$$

(Friction is supposed to be linear with the velocity of the motor.) ( $\sin \alpha \approx \alpha$ )

The last equation is now normalized in order to do the Laplace Transformation

$$\ddot{\alpha} + \frac{\mu}{ml} \cdot \dot{\alpha} - \frac{g}{l} \cdot \alpha = \frac{T_{Mot}}{ml^2} \quad (28)$$

The equation can now be transformed with the Laplace Formalism. This leads to

$$(s^2 \alpha(s) - s\alpha_0 - \dot{\alpha}_0) + \frac{\mu}{ml} (s\alpha(s) - \alpha_0) - \frac{g}{l} \alpha(s) = \frac{T_{Mot}(s)}{ml^2} \quad (29)$$

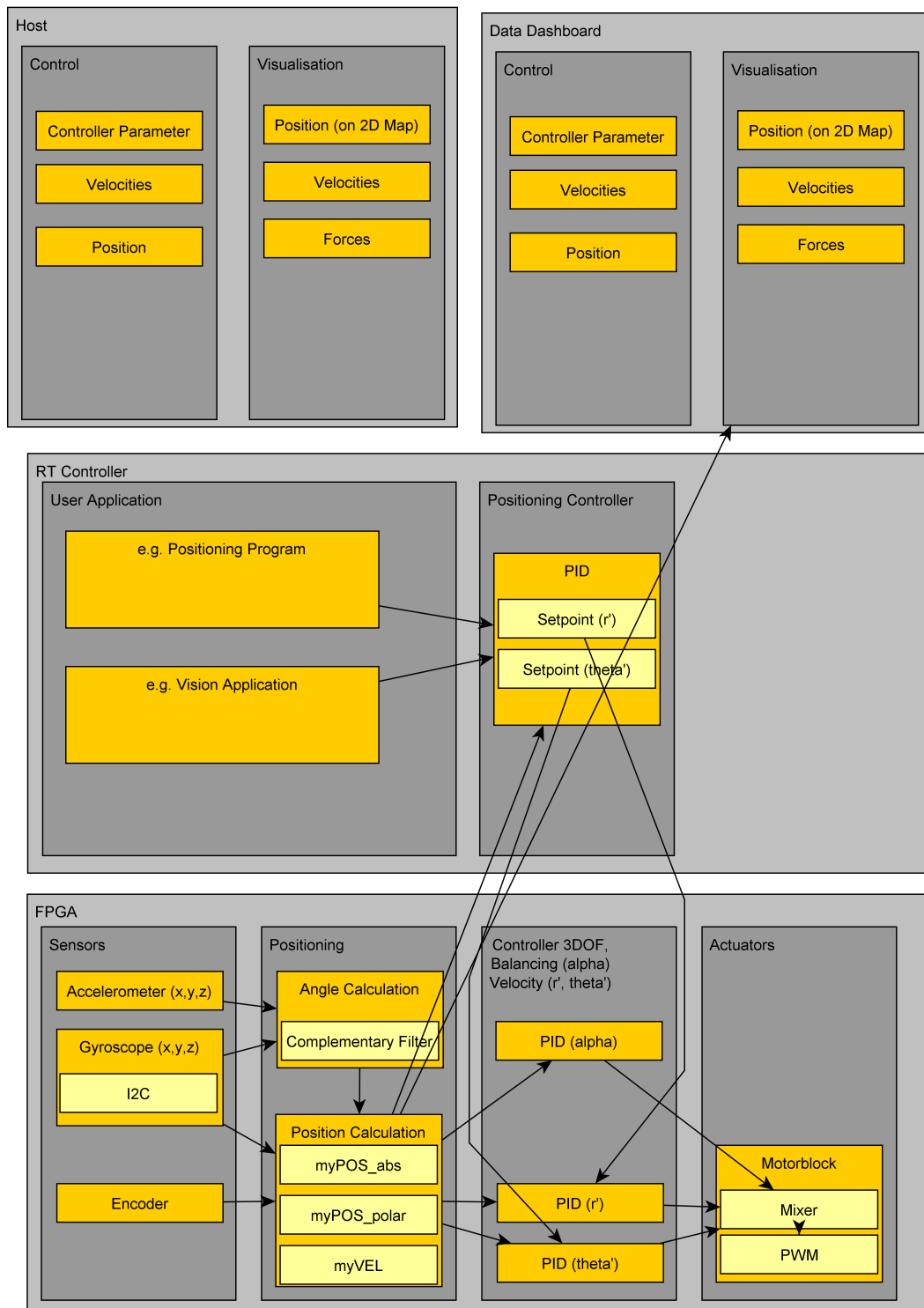


Abbildung 10: SW Architektur des myBOT, 3 Ebenen

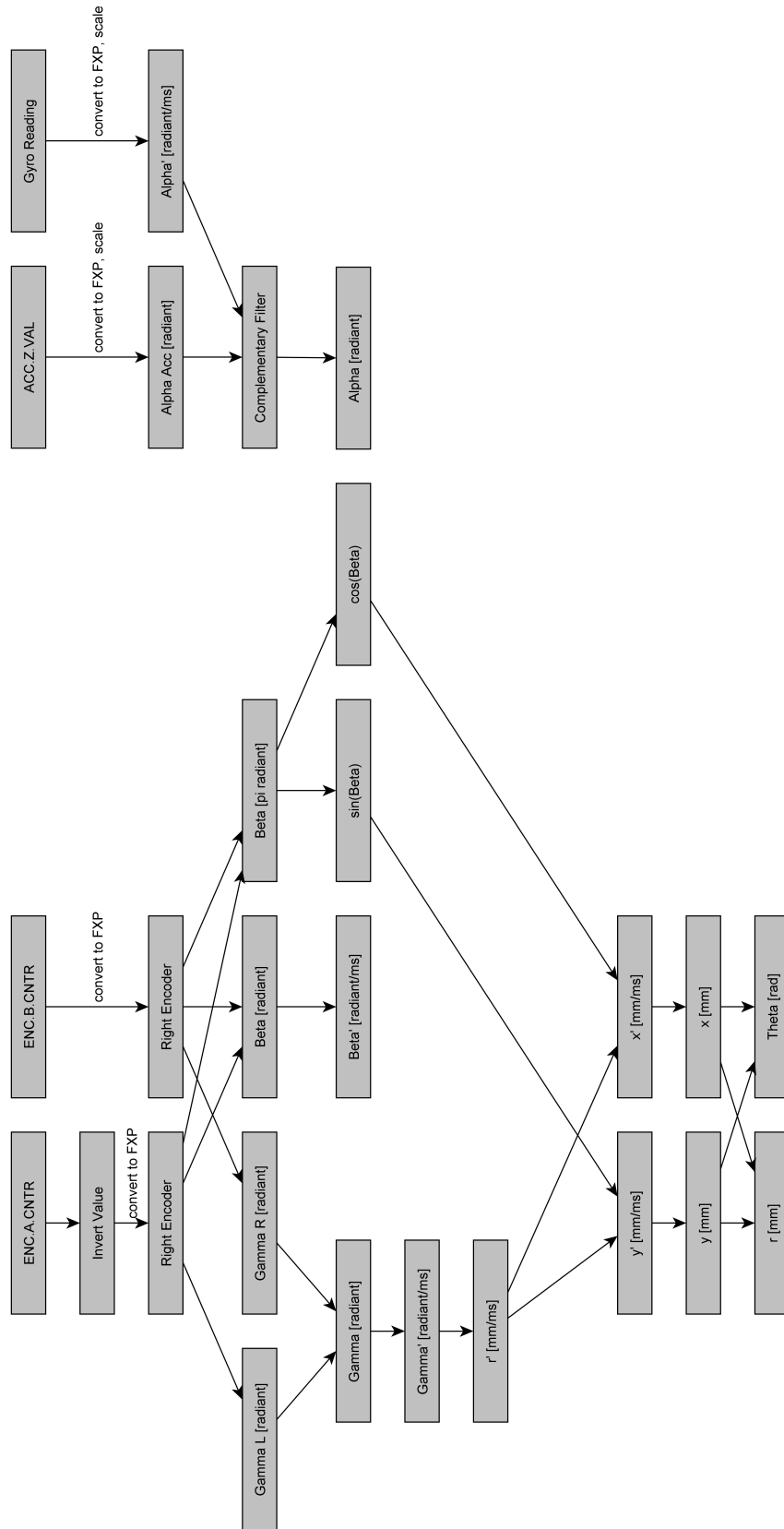


Abbildung 11: Berechnungen zur Gewinnung von myPOS und myVEL auf dem FPGA

The next step is to find the transfer function. The transfer function is defined as

$$G(s) = \frac{\text{Output}}{\text{Input}} = \frac{\alpha(s)}{T_{Mot}(s)} \quad (30)$$

The equation (29) can not be brought to the form of (30). To solve (29) one needs to assume starting values for  $\alpha, \alpha_0$  and  $\dot{\alpha}_0$ . The most simple idea is to set them all to 0. This leads to

$$s^2\alpha(s) + \frac{\mu}{ml}s\alpha(s) - \frac{g}{l}\alpha(s) = \frac{T_{Mot}(s)}{ml^2} \quad (31)$$

This can be written as

$$\frac{\alpha(s)}{T_{Mot}(s)} = \frac{1}{ml^2} \cdot \frac{1}{s^2 + \frac{\mu}{ml}s - \frac{g}{l}} \quad (32)$$

Mit numerischen Werten. (Siehe Abschnitt: 3.1)

$$\frac{\alpha(s)}{T_{Mot}(s)} = \frac{615.385}{s^2 + 15.3846s - 196.2} \quad (33)$$

### 8.1.1 2 DOF

Zusätzliche Bewegung der Räder möglich. Allerdings keine rotative Bewegung. Zielgrößen sind nun somit:  $\alpha$  und  $\gamma$ .

$$\mathcal{L} = K - P = K_{T_{Body}} + K_{R_{Body}} + K_{T_{Wheels}} + K_{R_{Wheels}} - P_{Body} = \frac{1}{2}m\dot{r}^2 + \frac{1}{2}ml^2\dot{\alpha}^2 + \frac{1}{2}M\dot{r}^2 + \frac{1}{2}M\dot{r}^2 - mgl \cos \alpha \quad (34)$$

Ausdrücken von  $r$  mit  $\gamma$  ergibt

$$\mathcal{L} = \frac{1}{2}mR\dot{\gamma}^2 + \frac{1}{2}ml^2\dot{\alpha}^2 + \frac{1}{2}MR\dot{\gamma}^2 + \frac{1}{2}MR\dot{\gamma}^2 - mgl \cos \alpha \quad (35)$$

### 8.1.2 Vereinfachung aufgrund der Energieanteile

Nicht alle Terme liefern im gleichen Anteil zur Gesamtenergie des myBOT bei. Wie Abbildung 12

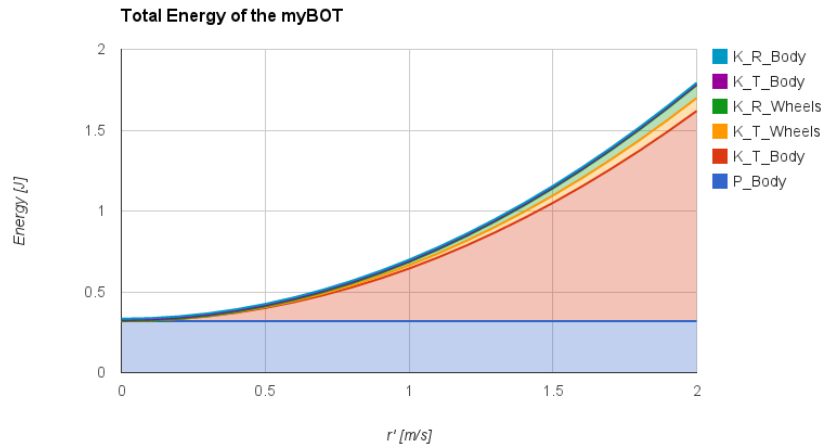


Abbildung 12: Anteile der verschiedenen Energien an der Gesamtenergie des myBOT

zeigt, liefern vor allem zwei Energieanteile viel zum Gesamtergebnis bei. Bei  $\dot{\alpha} = 3 \frac{\text{rad}}{\text{s}}$  und  $\dot{r} = 1 \frac{\text{m}}{\text{s}}$  beträgt der Anteil von  $P_{Body} + K_{T_{Body}}$  92.2 % der Gesamtenergie. Für die weitere Betrachtung

und die Herleitung einer (ersten) Übertragungsfunktion, sollen deshalb die anderen Anteile vernachlässigt werden.

Demnach gilt

$$\mathcal{L} = \frac{1}{2}mR \cdot \dot{\gamma}^2 - mgl \cos \alpha \quad (36)$$

Für 2 DOF demnach

$$\frac{\partial}{\partial t} \cdot \left( \frac{\partial \mathcal{L}}{\partial \dot{\alpha}} \right) - \frac{\partial \mathcal{L}}{\partial \alpha} = T_M + Q_\alpha \quad (37)$$

$$\frac{\partial}{\partial t} \cdot \left( \frac{\partial \mathcal{L}}{\partial \dot{\gamma}} \right) - \frac{\partial \mathcal{L}}{\partial \gamma} = T_M + Q_\gamma \quad (38)$$

Durch bilden aller Terme folgt

$$0 - mgl \sin \alpha = T_{Mot} - \mu \dot{\alpha} \quad (39)$$

$$mR \ddot{\gamma} = T_{Mot} - \mu \dot{\gamma} \quad (40)$$

linearisiert

$$\dot{\alpha} - \frac{mgl}{\mu} \alpha = \frac{T_{Mot}}{\mu} \quad (41)$$

$$\ddot{\gamma} + \frac{\mu}{mR} \dot{\gamma} = \frac{T_{Mot}}{mR} \quad (42)$$

Nach der Laplace-Trafo, Anfangswerte alle = 0 1)

$$\frac{\alpha(s)}{T_{Mot}} = \frac{1}{\mu} \cdot \frac{1}{s - \frac{mgl}{\mu}} = \frac{2}{s - 0.63765} \quad (43)$$

2)

$$\frac{\gamma(s)}{T_{Mot}} = \frac{1}{mR} \cdot \frac{1}{s^2 + s \frac{\mu}{mR}} = \frac{43.956}{s^2 + 21.978s} \quad (44)$$

## 8.2 Simulation in LabVIEW

Die Simulation erfolgt gemäss dem Reglerkonzept (Abschnitt: 5). Demnach wird der Sollwert für den Winkel  $\gamma$  erreicht, indem der Sollwert für  $\alpha$  verändert wird, und der myBOT sich deshalb in die jeweilige Richtung bewegen muss. Der myBOT verhält sich in dieser Simulation wie erwartet. Eine

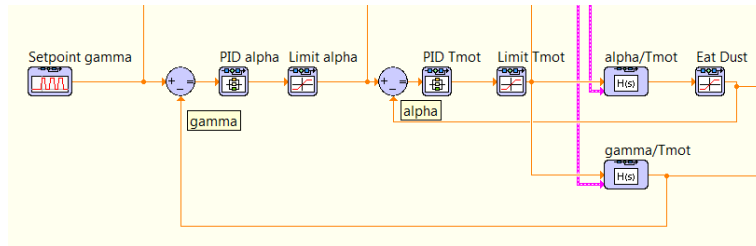


Abbildung 13: Controlloop (Ausschnitt) für den 2DOF Regler

Veränderung des Setpoints für  $\gamma$  führt zu einer Neigung von  $\alpha$ . Die Bewegung des myBOT geht dabei zu Beginn zuerst kurz in die entgegengesetzte Richtung, auch das gemäss den Erwartungen.



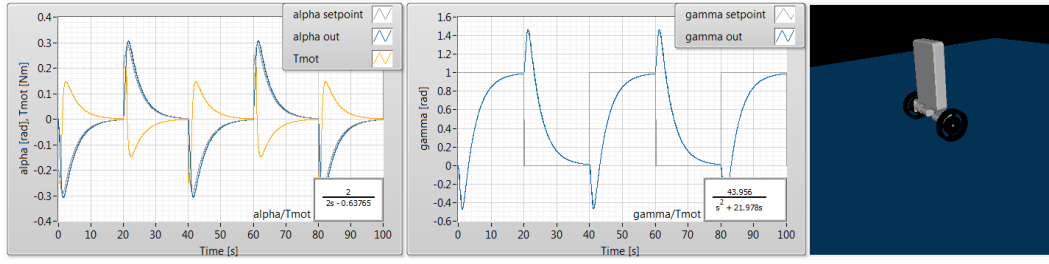


Abbildung 14: Resultate der 2DOF Simulation

## 9 Komplette Form

$$\mathcal{L} = \frac{1}{2}mR \cdot \dot{\gamma}^2 + \frac{1}{2}ml^2 \cdot \dot{\alpha}^2 + \frac{1}{2}MR \cdot \dot{\gamma}^2 + \frac{1}{2}MR\dot{\gamma}^2 - mgl \cos \alpha \quad (45)$$

Für 2 DOF demnach

$$\frac{\partial}{\partial t} \cdot \left( \frac{\partial \mathcal{L}}{\partial \dot{\alpha}} \right) - \frac{\partial \mathcal{L}}{\partial \alpha} = T_M + Q_\alpha \quad (46)$$

$$\frac{\partial}{\partial t} \cdot \left( \frac{\partial \mathcal{L}}{\partial \dot{\gamma}} \right) - \frac{\partial \mathcal{L}}{\partial \gamma} = T_M + Q_\gamma \quad (47)$$

Durch bilden aller Terme folgt

$$ml^2 \ddot{\alpha} - mgl \sin \alpha = T_{Mot} - \mu \dot{\alpha} \quad (48)$$

$$R(m + 2M) \ddot{\gamma} = T_{Mot} - \mu \dot{\gamma} \quad (49)$$

linearisiert und aufgeräumt

$$\ddot{\alpha} + \frac{\mu}{ml^2} \dot{\alpha} - \frac{g}{l} \alpha = \frac{T_{Mot}}{ml^2} \quad (50)$$

$$\ddot{\gamma} + \frac{\mu}{R(m + 2M)} \dot{\gamma} = \frac{T_{Mot}}{R(m + 2M)} \quad (51)$$

Nach der Laplace-Trafo, Anfangswerte alle = 0 1)

$$\frac{\alpha(s)}{T_{Mot}} = \frac{1}{ml^2} \cdot \frac{1}{s^2 + s \frac{\mu}{ml^2} - \frac{g}{l}} = \frac{615.385}{s^2 + 307.692s - 196.2} \quad (52)$$

2)

$$\frac{\gamma(s)}{T_{Mot}} = \frac{1}{R(m + 2M)} \cdot \frac{1}{s^2 + s \frac{\mu}{R(m+2M)}} = \frac{39.1389}{s^2 + 19.5695s} \quad (53)$$

### 9.1 Simulation in LabVIEW

Die Simulation erfolgt gemäss dem Reglerkonzept (Abschnitt: 5). Demnach wird der Sollwert für den Winkel  $\gamma$  erreicht, indem der Sollwert für  $\alpha$  verändert wird, und der myBOT sich deshalb in die jeweilige Richtung bewegen muss. Der myBOT verhält sich in der Simulation viel unruhiger als der myBOT in der vereinfachten Form. Die Bewegungen in  $\gamma$  Richtung sind viel langsamer, bei zu schnellem Kippen von  $\alpha$  fällt der myBOT sofort hin.

## 10 der myBOT Im Zustandsraum

Der myBOT kann ebenso gut im Zustandsraum beschrieben werden. Dazu müssen die bereits hergeleiteten Bewegungsgleichungen lediglich in ein Zustandsraummodell überführt werden.

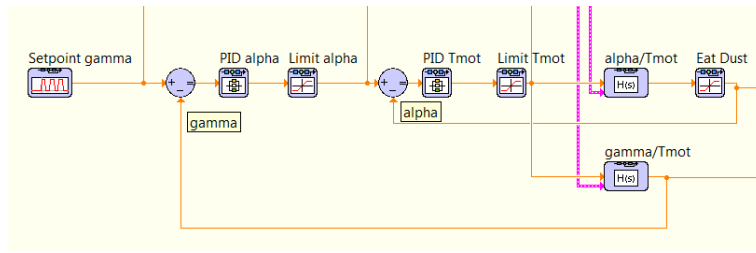


Abbildung 15: Controlloop (Ausschnitt) für den 2DOF Regler

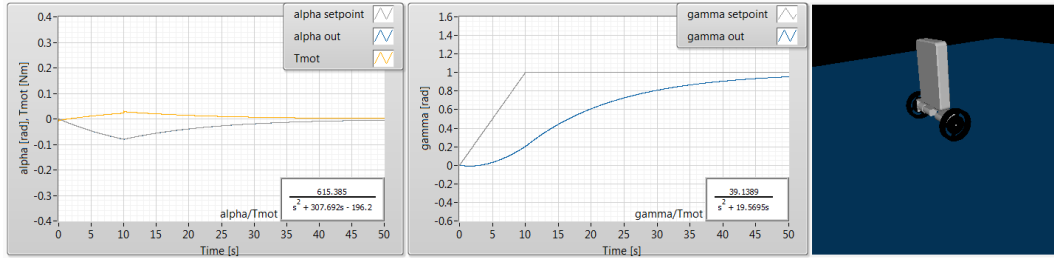


Abbildung 16: Resultate der 2DOF Simulation in der komplexen Form

## 10.1 Das allgemeine Zustandsraummodell

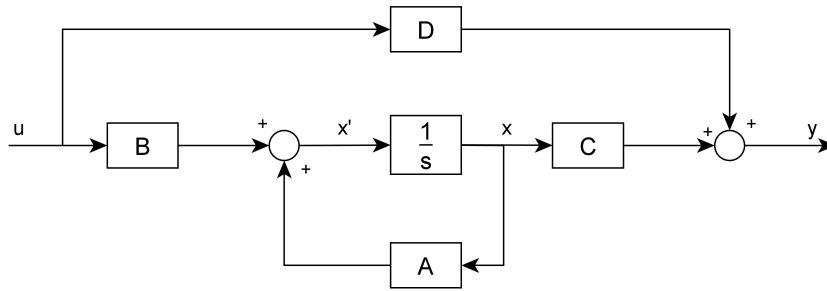


Abbildung 17: Abbildung des allgemeinen Zustandsraummodells

## 10.2 Zustandsraummodell für den vereinfachten myBOT

Dieses beruht auf den Gleichungen gemäß dem vereinfachten 2DOF Modell aus Abschnitt: 8.1.1. Das Modell wird hierbei zur folgenden Matrizengleichung.

$$\begin{bmatrix} \dot{\alpha} \\ \dot{\gamma} \\ \ddot{\gamma} \end{bmatrix} = \begin{bmatrix} \frac{mgl}{\mu} & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{\mu}{mR} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \gamma \\ \dot{\gamma} \end{bmatrix} + \begin{bmatrix} \frac{1}{\mu} \\ 0 \\ \frac{1}{mR} \end{bmatrix} \cdot T_{Mot} \quad (54)$$

Die Ausgangsmatrix C ist dabei definiert als

$$y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \gamma \\ \dot{\gamma} \end{bmatrix} \quad (55)$$

So ergeben sich die vier Matrizen  $A, B, C$  &  $D$  wie folgt:

$$A = \begin{bmatrix} \frac{mgl}{\mu} & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{\mu}{mR} \end{bmatrix}, B = \begin{bmatrix} \frac{1}{\mu} \\ 0 \\ \frac{1}{mR} \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (56)$$

Damit ist das System vollständig bestimmt und kann simuliert werden.

### 10.2.1 Simulation in LabVIEW

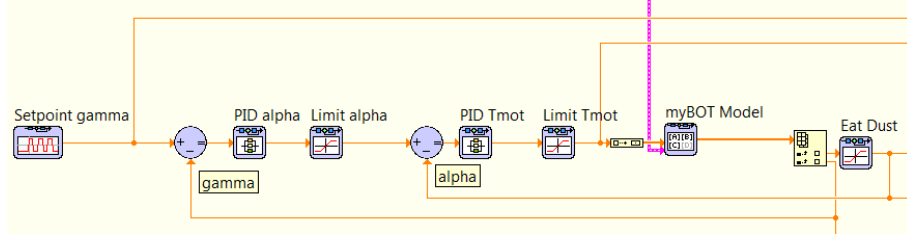


Abbildung 18: Controlloop für das vereinfachte Zustandsraummodell mit 2DOF

Die Simulation verhält sich dabei wie erwartet.

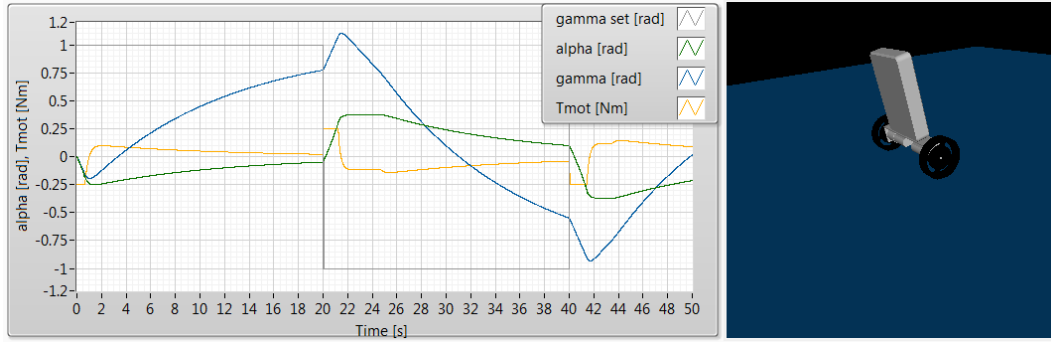


Abbildung 19: Erste Resultate für die Simulation des vereinfachte 2DOF Zustandsraummodells

### 10.3 Zustandsraummodell des kompletten myBOT

Dieses beruht auf den Gleichungen gemäss dem vereinfachten 2DOF Modell aus Abschnitt: 9. Die Matrizen sind dabei in der Form

$$x = \begin{bmatrix} \alpha \\ \dot{\alpha} \\ \gamma \\ \dot{\gamma} \end{bmatrix}, b = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \cdot T_{Mot}, A = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \\ c_1 & c_2 & c_3 & c_4 \\ d_1 & d_2 & d_3 & d_4 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, D = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Durch Auflösen der Gleichung aus 9 kann somit das folgende Modell gewonnen werden.

$$\begin{bmatrix} \dot{\alpha} \\ \ddot{\alpha} \\ \dot{\gamma} \\ \ddot{\gamma} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{g}{l} & -\frac{\mu}{ml^2} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & \frac{-\mu}{R(m+2M)} \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \dot{\alpha} \\ \gamma \\ \dot{\gamma} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{ml^2} \\ 0 \\ \frac{1}{R(m+2M)} \end{bmatrix} \cdot T_{Mot} \quad (57)$$

Die Ausgangsmatrize C ist dabei definiert als

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \alpha \\ \dot{\alpha} \\ \gamma \\ \dot{\gamma} \end{bmatrix} \quad (58)$$

### 10.3.1 Simulation in LabVIEW

Simulation mit Controlloop aus Abbildung 18. Verwendet wurden dabei die Parameter:

**Simulation Parameter**  $T_{simulation} = 50s, dt = 0.005s$

**myBOT Parameter**  $\mu = 0.05$

**PID alpha**  $K_p = -0.2, K_i = 0, K_d = 0$

**PID Tmot**  $K_p = 50, K_i = 0, K_d = 0$

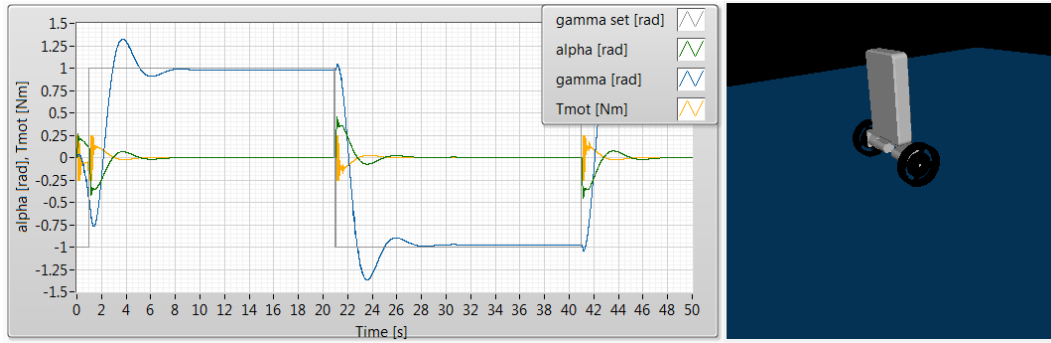


Abbildung 20: Simulationsergebnisse mit P-Regler

In einem zweiten Versuch wurde der Regler aus Abschnitt: 5.1.2 implementiert. Abbildung 21 zeigt dabei den Controlloop. Dabei wurden die folgenden Parameter verwendet:

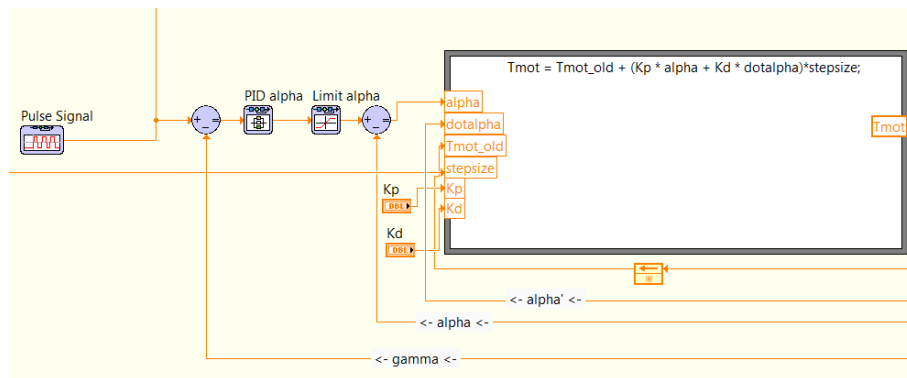


Abbildung 21: Balance Controller, Rückführung von  $\alpha, \dot{\alpha}$  &  $\gamma$

**Simulation Parameter**  $T_{simulation} = 50s, dt = 0.005s$

**myBOT Parameter**  $\mu = 0.05$

**PID alpha**  $K_p = -0.1, K_i = 0, K_d = 0$

**PID BalanceController**  $K_p = 10, K_i = 0, K_d = -1$

Dies führte zu den Resultaten in Abbildung 22 Zu sehen ist insbesondere, dass mit dem Balance

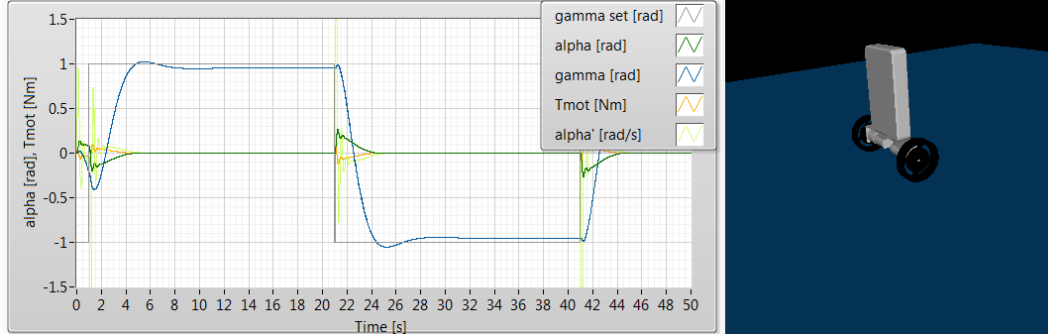


Abbildung 22: Erste Simulationsergebnisse mit dem Balance Controller

Controller in diesem ersten Anlauf die Werte von  $T_{Mot}$  massiv viel kleiner blieben als in der vorangehenden Simulation mit dem P-Regler. Dies ist sicherlich begrüßenswert und soll genauer verfolgt werden.

## 10.4 Erweiterung auf 3DOF

Für die Erweiterung um den 3. Freiheitsgrad  $\beta$  müsste das Gleichungssystem erweitert werden. Der

Zustandsvektor  $x$  bekäme dabei die Form  $x = \begin{bmatrix} \alpha \\ \dot{\alpha} \\ \beta \\ \gamma_l \\ \dot{\gamma}_l \\ \gamma_r \\ \dot{\gamma}_r \end{bmatrix}$  A wird eine 7x7 Matrix. Das ganze System

wird einiges unübersichtlicher, und die Erkenntnisse für die Regelung des realen Systems dürften nicht massiv grösser werden.. Für den Moment wird deshalb auf diese Erweiterung verzichtet.

## 10.5 Erweiterung um Nichtlinearitäten

Berücksichtigung von Nichtlinearitäten. Deadzone bei der  $\frac{T_{Mot}}{Spannung}$ -Kennlinie des Motors, Backlash (Spiel des Getriebes)

**Simulation Parameter**  $T_{simulation} = 50s, dt = 0.005s$

**myBOT Parameter**  $\mu = 0.05$

**PID alpha**  $K_p = -0.1, K_i = 0, K_d = 0$

**PID BalanceController**  $K_p = 8, K_i = 0, K_d = -2$

Dies führte zu den Resultaten in Abbildung 23 Zu sehen ist, das auch beim Prototypen beobachtete Schwingverhalten im Stillstandsbereich.

Tuning der Parameter zu

**Simulation Parameter**  $T_{simulation} = 50s, dt = 0.005s$

**myBOT Parameter**  $\mu = 0.05$

**PID alpha**  $K_p = -0.1, K_i = 0, K_d = 0$

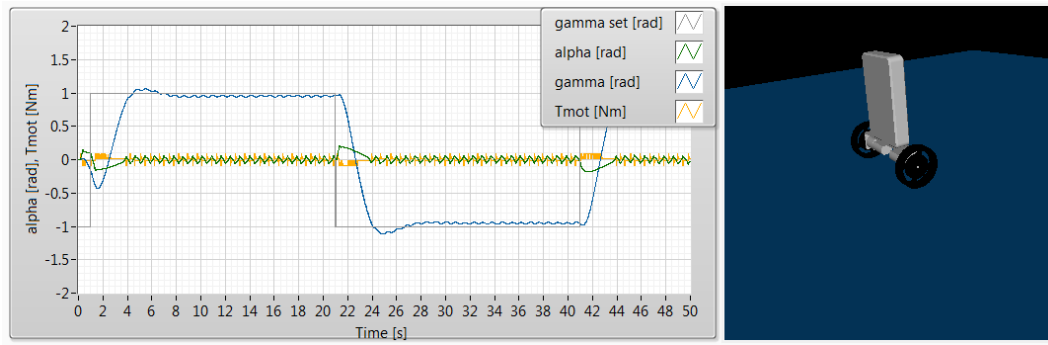


Abbildung 23: Erste Simulationsergebnisse mit dem Balance Controller und Nichtlinearitäten

**PID BalanceController**  $K_p = 12, K_i = 0, K_d = -6$

führte zu einer Eliminierung dieses Phänomens Dies führte zu den Resultaten in Abbildung 24

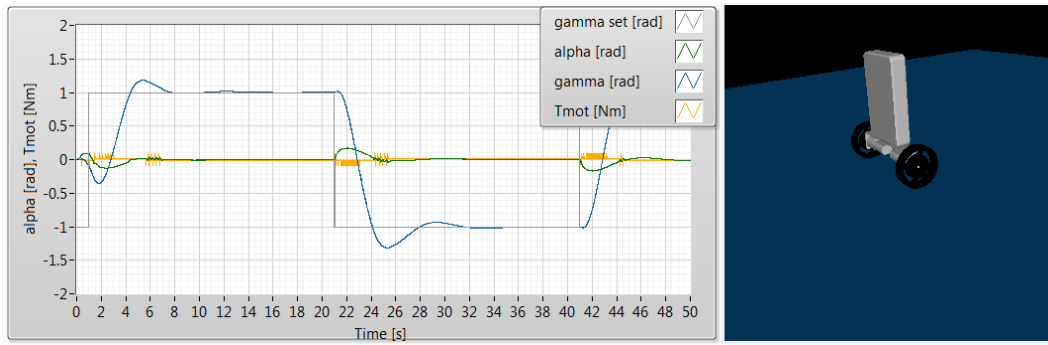


Abbildung 24: Verbesserte Simulationsergebnisse mit dem Balance Controller und Nichtlinearitäten

Insgesamt wird das System durch die Nichtlinearitäten schwieriger zu kontrollieren. Zu sehen ist in den Resultaten die Ausgangsbeschränkung der Motoren (gelbe Linie) sowie die Deadzone der Motoren.