Hochschule Karlsruhe
Technik und Wirtschaft
UNIVERSITY OF APPLIED SCIENCES

Fakultät für Maschinenbau und Mechatronik

Studiengang Maschinenbau

Projektarbeit

Berechnung des Lagewinkels eines Pendels mit Hilfe eines Kalman-Filter

Projektcode: 19ws_SC_KALMAN_FILTER_PENDEL

von

Santiago Ramos

Matrikel-Nr. 66280

Pilar Samaniego

Matrikel-Nr. 66217

Betreuer: Prof. Dr.-Ing. Helmut Sherf

Zeitraum: 05.11.2019 – 20.02.2019

Eidesstattliche Erklärung

Hiermit versichern wir, die vorliegende Projektarbeit ohne unzulässige fremde Hilfe selbständig verfasst, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie alle wörtlich oder sinngemäß übernommenen Stellen in der Arbeit gekennzeichnet zu haben.

Karlsruhe, den (Unterschrift)

Kurzfassung

Der Kalman-Filter wird in großem Umfang verwendet, um die Zustände von zeitdiskreten dynamischen Systemen in einem großen Anwendungsbereich wie Beispielweise bei Steuerungssystemen, Signalverarbeitungs- und Kommunikationssystemen abzuschätzen. Eine korrekte und genaue Zustandsabschätzung von linearen oder nicht linearen Systemen kann durch Auswahl der richtigen Schätztechnik verbessert werden. Kalman-Filteralgorithmen werden häufig verwendet, um lineare, unverzerrte und minimale Varianzschätzungen eines unbekannten Zustandsvektors für nicht linearen Systeme bereitzustellen. Daher untersuchen wir den Einsatz des Kalman-Filter zu Bestimmung des Pendelwinkels im Vergleich zu anderen Methoden. Dabei ist das Ziel den Pendelwinkel so genau wie möglich zu, indem ein Arduino-Board in Echtzeit mit Matlab / Simulink kommuniziert und eine Trägheitsmesseinheit MPU 6050 mit einem Beschleunigungsmesser und einem Gyroskop verwendet wird.

Abstract

The Kalman filter is used extensively to estimate the states of discrete-time dynamic systems in a wide range of applications such as control systems, signal processing and communication systems. Correct and accurate state estimation of linear or non-linear systems can be improved by choosing the right estimation technique. Kalman filter algorithms are widely used to provide linear, undistorted, and minimal variance estimates of an unknown state vector for non-linear systems. We are therefore investigating the use of the Kalman filter to determine the pendulum angle compared to other methods. The goal is to determine the pendulum angle as precisely as possible, with an Arduino board communicated in real time with Matlab/Simulink and using a MPU 6050 Inertial Measurement Unit, which has an Accelerometer and a Gyroscope.

Nomenklatur

Lateinische Formelzeichen

w_k		Rauschen des Systems
v_k		Rauschen der Messung
z_k		Messungsvektor oder Beobachtungsvektor
H_k		Messungsmatrix oder Beobachtungsmatrix
$\hat{\boldsymbol{x}}_{k\|k}$		Der geschätzte Zustand im Zeitpunkt k
$\hat{\boldsymbol{x}}_{k+1\|k}$		Der geschätzte Zustand im Zeitpunkt k+1
Q_{k+1}		Kovarianzmatrix des Rauschens des Systems
R_{k+1}		Kovarianzmatrix des Rauschens der Messung
$P_{k\|k}$		Fehler- Kovarianzmatrix
K_{k+1}		Kalman Verstärkung
$\boldsymbol{x}_k$		Der echte Zustandsvektor
A_p		Beschleunigungsmesserwert
$R_x(\phi)$		Roll Rotationsmatrizen,
$R_y(\theta)$		Nick Rotationsmatrix
$R_z(\psi)$		Gier Rotationsmatrix
R_{xyz}		Rotationsmatrix in Ordnung x,y,z
g	m/s^2	Schwerkraft
A_p		Beschleunigungsmesserausgabe
a	m/s^2	Lineare Beschleunigung
F	N	Kraft
v	m/s	Geschwindigkeit
z	m	Position
m	Kg	Masse
Ω		Rotationsgeschwindigkeit
F_c		Corioliskraft

Griechische Formelzeichen

Δ	-	Differenz
π	-	Pi
φ	Grad	Neigungswinkel zur Horizontalen
Φ	Grad	Sprühwinkel
$\Phi_{k+1,k}$		Die Zustandsübergangsmatrix
$\Gamma_{,k}$		Die Koeffizienten Matrix für das Rauschen

Indizes

0		Bezugsgröße zur geraden Länge
1,2,3,4,5	Oder k	Zustände
ref		Referenz
k		Zustand im Zeitpunkt k
$k+1$		Zustand im Zeitpunkt k+1

Differenz

(1)	-	Gleichungsnummer
[12]	-	Nummer im Quellenverzeichnis

Abkürzungen

Gl.	Gleichung
HsKA	Hochschule Karlsruhe – Technik und Wirtschaft

Inhaltsverzeichnis

1 Einleitung

Rudolf Emil Kalman wurde am 19. Mai 1930 in Budapest, Ungarn, geboren. 1958 hatte er erstmals die Idee des Kalman-Filters. 1960 und 1961 veröffentlichte er seine Arbeiten zum Kalman-Filter und revolutionierte damit das Feld der Schätzung.

Das Kalman-Filter ist ein rekursives Vorhersagefilter, das auf der Verwendung von Zustandsraumtechniken und rekursiven Algorithmen basiert. Es schätzt den Zustand eines dynamischen Systems. Dieses dynamische System kann durch etwas Rauschen gestört werden, das meist als weißes Rauschen angenommen wird. Um den geschätzten Zustand zu verbessern, verwendet der Kalman-Filter Messungen, die sich auf den Zustand beziehen, aber auch gestört sind.

Somit besteht der Kalman-Filter aus zwei Schritten: die Vorhersage und die Korrektur. Im ersten Schritt wird der Zustand mit dem dynamischen Modell vorhergesagt. Im zweiten Schritt wird es mit dem Beobachtungsmodell korrigiert, so dass die Fehlerkovarianz des Schätzers minimiert wird. [1]. Dieser Vorgang wird für jeden Zeit-schritt mit dem Zustand des vorherigen Zeitschritts als Anfangswert wiederholt. Daher wird der Kalman-Filter als rekursiver Filter bezeichnet.

1.1 Aufgabenstellung und Zielsetzung

- Verarbeitung der Messungen des inertial Measurement Unit
- Modellierung und Umsetzung der Winkelberechnung mit MATLAB/SIMULINK
- Modellierung und Implementieren eines zeitdiskreten linearen Kalman-Filters für die Zustandsschätzung des Pendelwinkels mittels MATLAB/SIMULNK
- Vergleich der Ergebnisse zwischen Kalman-Filter, Komplementär-Filter und Potenziometers.

1.2 Vorausgegangene Arbeiten und Stand der Erkenntnisse

In früheren Arbeiten wie in [1] und [2] ist die derzeitige Verwendung von Kalman-Filtern in verschiedenen Bereichen zu sehen, wobei der erste verwendet wurde, um den Gleichgewichtswinkel, die Neigung und das Gieren anhand der Daten der drei Sensoren zu bestimmen Beschleunigungsmesser, Gyroskope und Magnetometer, die mit dem Kalman-Filter verschmolzen werden. In der zweiten Arbeit wurde ein Kalman-Filter für Unterrichtszwecke mit MATLAB implementiert. Dies sind jedoch nur zwei Beispiele für die

Vielfalt der Bereiche, in denen die Kalman-Filterung eine wichtige Rolle spielt. Die Anwendungsbereiche reichen von der Luft- und Raumfahrt über die Seefahrt bis hin zur demografischen Modellierung, Klimawissenschaft und Fertigung. Da der Kalman-Filter für eine so große Klasse von Problemen sehr effektiv und nützlich ist, wurde er ausgiebig untersucht.

Daher gibt es in diesem Projekt sowohl eine Implementierung des Kalman-Filters als auch eine praktische Anwendung bei der Bestimmung des Winkels eines Pendels.

2 Literaturrecherche

2.1 Inertiale Navigation

Bei der Inertialen Navigation handelt es sich um eine sogenannte Koppelnavigation. Dabei werden fortlaufend die Bewegungsrichtung, Geschwindigkeit und die seit der letzten Positionsbestimmung vergangene Zeit bestimmt. Durch diese Informationen kann die aktuelle Position berechnet werden, indem die zurückgelegte Strecke für einen bestimmten Zeitintervall berechnet wird. Für die Navigation ist es nötig, die Position im Navigationskoordinatensystem zu haben. Dies bedeutet, dass die Achsen in Nord, Ost und zum Erdmittelpunkt zeigen.

Die Grundidee ist es, die benötigten Informationen durch Messung von Beschleunigung, der Drehrate und dem Magnetfeld zu gewinnen. Hier kommt die Inertialsensoreinheit ins Spiel [2].

2.1.1 Inertialsensoreinheit

Bei einer Inertialsensoreinheit (englisch: inertial measurment unit IMU), handelt es sich um ein Sensorpaket aus drei Beschleunigungssensoren und drei Gyroskopen, jeweils angeordnet in X, Y und Z Richtung, sowie einem Magnetometer. Durch die Beschleunigungssensoren lässt sich die translatorische Bewegung, in Form von Beschleunigung messen und mit den Gyroskopen die rotatorische Bewegung in Form einer Drehrate.

Der Vorteil der IMU ist es, dass diese fest im Fahrzeug verbaut werden kann und nicht mehr aufwendig entkoppelt werden muss. Durch eine Transformationsmatrix wird eine Koordinatentransformation vom körperfesten Koordinatensystem ins Navigationskoor-

dinatensystem durchgeführt. In Verbindung mit einer Recheneinheit werden zum Beispiel die Winkelgeschwindigkeiten erfasst und durch Integration kann die Fahrzeuglage berechnet werden [2].

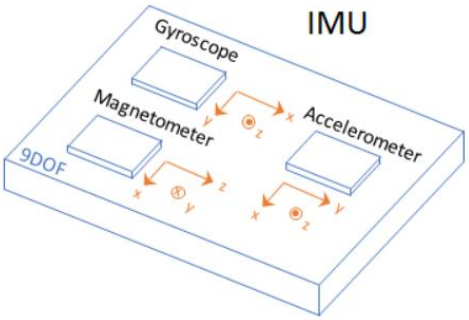


Abbildung 1: Inertialsensoreinheit **[3]**

2.1.2 Koordinatensysteme

Im folgenden Abschnitt werden das Koordinatensysteme definiert.

- **Körperfestes Koordinatensystem (b)**

  Die Achsen dieses Koordinatensystems sind fest mit dem Körper verbunden. Der Ursprung befindet sich im Körper. Bei genauer Ausrichtung der IMU zum Körper fallen die Achsen der IMU mit denen des körperfesten Koordinatensystems zusammen.

- **Inertial Koordinatensystem (i)**
  Der Mittelpunkt dieses Koordinatensystems befindet sich im Mittelpunkt des Rotationsellipsoids, der die Erdgestalt annähert. Eine IMU misst Beschleunigungen und Drehraten des körperfesten Koordinatensystems bezüglich des Inertial Koordinatensystems

- **Erdfestes Koordinatensystem (e)**

  Das erdfeste Koordinatensystem besitzt denselben Ursprung wie das Inertial Koordinatensystem. Das Koordinatensystem wird auch als earth centered, earth fixed (ECEF) Koordinatensystem bezeichnet.

- **Navigationskoordinatensystem (n)**

Der Ursprung dieses Koordinatensystems fällt mit dem des körperfesten Systems zusammen. Jedoch sind die X und Y-Achsen in Nord und Ost Richtung in der Tangentialebene des Erdellipsoid ausgerichtet. Die Z-Achse zeigt in Richtung der Schwerebeschleunigung.

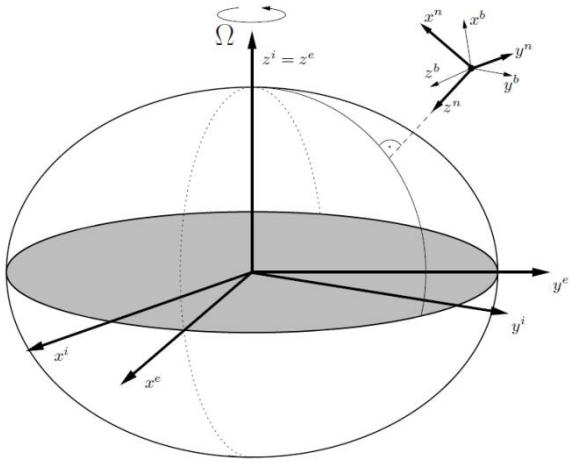


Abbildung 2: Darstellung der unterschiedlichen Koordinatensysteme **[2]**

2.1.3 Winkelbestimmung mit Hilfe der Beschleunigungssensoren

Die Ausrichtung eines Körpers kann durch seine Roll-, Nick- und Gierdrehungen (Abbildung 3) von einer Anfangsposition aus definiert werden.

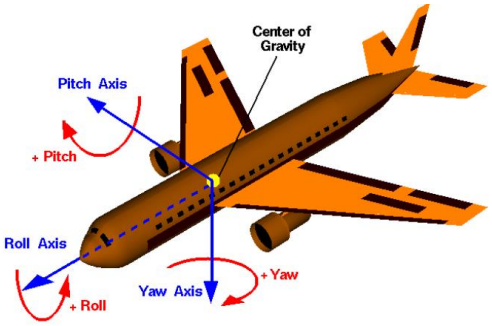


Abbildung 3: Rotationen eines Körpers **[4]**

Die Roll-, Nick- und Gier-Rotationsmatrizen, die einen Vektor (wie den Erdschwerefeldvektor g) bei einer Drehung des Koordinatensystems von Abbildung 2 um die Winkel ϕ in Roll, θ in Nick und ψ in Gier um das x, y und z-Achse - bzw transformieren sind:

$$R_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{pmatrix} \quad (1)$$

$$R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \quad (2)$$

$$R_z(\psi) = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (3)$$

Es gibt sechs mögliche Ordnungen dieser drei Rotationsmatrizen und im Prinzip sind alle gleich gültig. Die Rotationsmatrix R hängt von der Reihenfolge ab, in der die Roll-, Nick- und Gier Rotationen angewendet werden.

$$R_{xyz} = R_x(\phi)R_y(\theta)R_z(\psi) \quad (4)$$

$$= \begin{pmatrix} \cos\theta\ \cos\psi & \cos\theta\ \sin\psi & -\sin\theta \\ \cos\psi\sin\theta\sin\phi - \cos\phi\sin\psi & \cos\psi\cos\phi + \sin\phi\sin\theta\sin\psi & \cos\theta\sin\phi \\ \cos\psi\sin\theta\cos\phi + \sin\phi\sin\psi & -\cos\psi\sin\phi + \cos\phi\sin\theta\sin\psi & \cos\theta\cos\phi \end{pmatrix} \quad (5)$$

Es ist daher üblich, entweder die Rotationssequenz R_{xyz} der Gleichungen oder die Sequenz R_{yxz} zu wählen, um die Gier Drehung Rotation zu eliminieren und eine Lösung für die Roll- und Nickwinkel zu ermöglichen.

Diese Arbeit befasst sich mit der Messung der Orientierung im Gravitationsfeld der Erde. Es wird die Konvention angenommen, dass die Beschleunigungsmesserausgabe auf einen Wert von + 1 g in jeder Achse negiert wird, die mit dem Gravitationsfeld nach unten ausgerichtet ist.

Unter dieser Annahme hat ein dreiachsiger Beschleunigungsmesser, der in unserem Pendel montiert ist und im Erdschwerefeld g ausgerichtet ist und eine lineare Beschleunigung a_r erfährt, die im Erdbezugsrahmen *e* gemessen wird, eine Ausgabe A_p, die durch die folgende Gleichung gegeben ist:

$$A_p = \begin{pmatrix} A_{px} \\ A_{py} \\ A_{pz} \end{pmatrix} = R(g - a_r) \quad (6)$$

Dabei ist R die Rotationsmatrix, die die Ausrichtung des Pendels relativ zum Koordinatenrahmen der Erde beschreibt.

Es wird auch angenommen, dass:

• Der Beschleunigungsmesser hat keine lineare Beschleunigung. Diese Annahme ist erforderlich, um Gl. (6) für die Rotationsmatrix R zu lösen, und folglich führt jede lineare Beschleunigung durch Handshake oder andere Quellen zu Fehlern bei der Orientierungsschätzung.

• Die anfängliche Ausrichtung des Pendels liegt flach, wobei das Gravitationsfeld der Erde auf die Pendel-Z-Achse ausgerichtet ist:

Mit diesen zusätzlichen Annahmen, die Pendelbeschleunigungsmesser-Ausgabe (IMU MPU6050) A_p (gemessen in den nativen Beschleunigungsmessereinheiten von g) ist:

$$A_p = \begin{pmatrix} A_{px} \\ A_{py} \\ A_{pz} \end{pmatrix} = R_{xyz} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{7}$$

Basierend auf den Gleichungen (1) bis (7) ist es möglich, die folgende Gleichheit vorzuschlagen:

$$A_p = \begin{pmatrix} A_{px} \\ A_{py} \\ A_{pz} \end{pmatrix} = R_{xyz} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -\sin\theta \\ \cos\theta \sin\phi \\ \cos\theta \cos\phi \end{pmatrix} \tag{8}$$

Gl. (8) kann in Form von Gl. (9) umgeschrieben werden, die die Roll- und Nickwinkel zu dem normalisierten Beschleunigungsmesserwert A_p in Beziehung setzt:

$$\frac{A_p}{\|A_p\|} = \frac{1}{\sqrt{{A_{px}}^2 + {A_{py}}^2 + {A_{pz}}^2}} \begin{pmatrix} A_{px} \\ A_{py} \\ A_{pz} \end{pmatrix} = \begin{pmatrix} -\sin\theta \\ \cos\theta \sin\phi \\ \cos\theta \cos\phi \end{pmatrix} \tag{9}$$

Das Auflösen der Roll- und Nickwinkel aus Gl. (9) und die Verwendung des Index xyz, um anzuzeigen, dass die Roll- und Nickwinkel gemäß der Rotationssequenz R_{xyz} berechnet werden, ergibt:

$$\tan\phi_{xyz} = \frac{A_{py}}{A_{pz}} \tag{10}$$

$$\tan \theta_{xyz} = \frac{-A_{px}}{A_{py} \sin \phi + A_{pz} \cos \phi} \tag{11}$$

2.1.4 Eliminierung doppelter Lösungen durch Begrenzung der Roll- und Nick-Bereiche

Die nächste zu behandelnder Komplikation besteht darin, dass die Ausdrücke für die Roll- und Nickwinkel in den Gl. (10) und (11) eine unendliche Anzahl von Lösungen bei Vielfachen von 360 ° haben.

Die Einschränkung des Bereichs der Roll- und Nickwinkel auf -180 ° bis 180 ° hilft etwas, aber der nächste Absatz zeigt, dass dies immer noch zu zwei einzigartigen Lösungen für die Roll- und Nickwinkel führt.

Die Auswertung von Gl. (8) für den Nickwinkel $\pi - \theta$ und den Rollwinkel $\phi + \pi$ und die Anwendung von trigonometrischen Standardidentitäten zeigt, dass die Beschleunigungsmesser-Messung dieselbe ist wie die, die sich aus den Drehungen θ und ϕ ergibt.

$$\begin{pmatrix} -\sin(\pi - \theta) \\ \cos(\pi - \theta) \sin(\phi + \pi) \\ \cos(\pi - \theta) \cos(\phi + \pi) \end{pmatrix} = \begin{pmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{pmatrix} \tag{12}$$

Die Lösung besteht darin, entweder den Roll- oder den Nickwinkel (aber nicht beide) auf einen Wert zwischen -90 ° und + 90 ° zu beschränken. Unabhängig von der verwendeten Konvention ist das Nettoergebnis, eine der beiden doppelten Lösungen zu eliminieren und eine einzige Lösung zu ergeben, mit Ausnahme der sehr spezifischen Ausrichtungen, die später diskutiert werden.

2.2 MEMS Sensoren

MEMS Sensoren (Micro Elektronische Mechanische Systeme) sind weit verbreitet, da diese viele Vorteile haben. Sie können physikalische sowie chemische Messgrößen erfassen und diese Messgrößen in ein elektrisches Signal umsetzen. Über elektronische Auswerteschaltungen, welche sich oft auf derselben Platine befinden, lassen sich die Messgrößen direkt auslesen, zum Beispiel über einen I2C Bus. In dieser Arbeit wird der MPU6050 benutz.

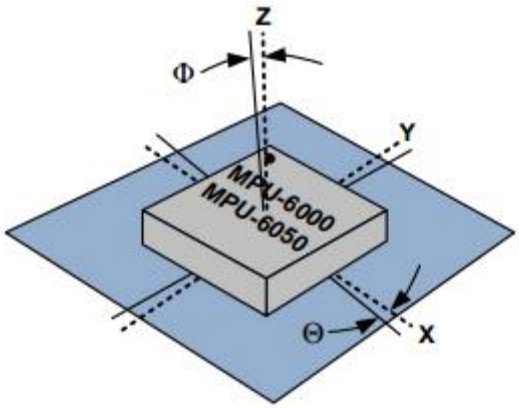


Abbildung 4: Sensor Paket IMU – MPU6050 **[5]**

2.2.1 Beschleunigungssensoren [6]

Beschleunigungssensoren messen die durch eine Beschleunigung ausgeübte Kraft F auf eine Masse m. Das zweite Newton'sche Axiom beschreibt den Zusammenhang wie folgt:

$$F = m * a \tag{13}$$

Den Zusammenhang zwischen dem Weg z und der Geschwindigkeit v sowie der Beschleunigung a zeigt folgende Formel:

$$a(t) = \frac{dv(t)}{dt} = \frac{d^2z(t)}{dt^2} \tag{14}$$

Aufgrund dieser Zusammenhänge lässt sich die Beschleunigung a messtechnisch aus F, v oder z bestimmen oder auch umgekehrt. Mit Hilfe der Erdbeschleunigung g, die zur Kalibrierung und Rekalibrierung von Beschleunigungsaufnehmern dient, sowie zur Bestimmung der Neigung zur Erdoberfläche dient. Der Beschleunigungssensor des MPU6050 gibt die Beschleunigung in der Einheit g aus. Dabei ist g die Schwerebeschleunigung der Erde, 1g entspricht $9{,}81 m/s^2$.

Das Grundprinzip aller Sensoren besteht darin, die Wirkung auf ein gedämpftes Feder-Masse System zu messen. Das Prinzip besteht darin, dass sich eine Masse, welche an das Gehäuse gekoppelt ist, bei einer Beschleunigung des Gehäuses, dem Gehäuse gegenüber verschiebt.

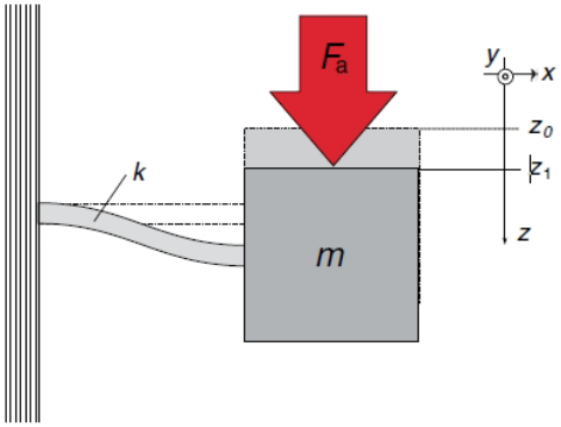


Abbildung 5: Beschleunigungssensors **[6]**

Hierbei ist eine Masse über eine Kopplung mit einer Federkonstanten k an dem Gehäuse befestigt. Durch eine Beschleunigung F_a wird die Masse in Richtung z verschoben.

Es gibt viele unterschiedliche Möglichkeiten einen Beschleunigungssensor aufzubauen. Dazugehören kapazitive, magnetische und induktive, piezoresistive, piezoelektrische, thermische, optische und solche, die den Tunneleffekt benutzen. Die am häufigsten und auch in dieser Arbeit benutzen Beschleunigungssensoren sind die kapazitiven Beschleunigungssensoren.

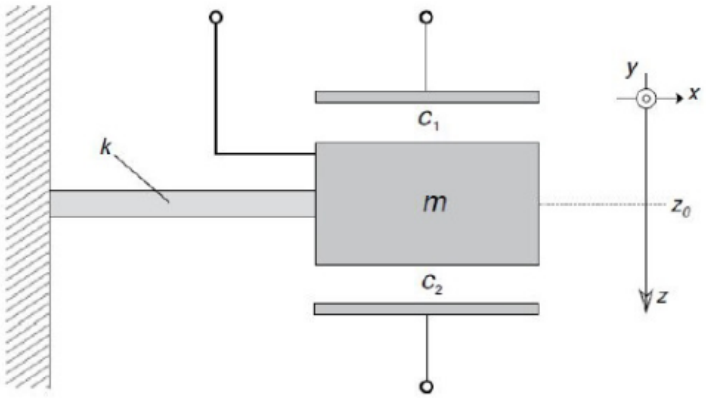


Abbildung 6: Kapazitive Beschleunigungssensors **[6]**

2.2.2 Gyroskop [6]

Ein Gyroskop misst die Rotationsgeschwindigkeit Ω um eine definierte Achse in °/s. Um die Drehrate zu messen gibt es verschiedene physikalische Messprinzipien. Es gibt Sensoren, welche die Drehrate über die Coriolis-Kraft messen, welche auf eine schwingende Masse einwirkt. Andere Sensoren nutzen den Sagnac-Effekt. Wobei zuletzt genannte Sensoren eine höhere Genauigkeit aufweisen. In dieser Arbeit wird ein MEMS Sensor benutzt, der die Corioliskraft nutzt, da diese günstiger und in ihrer Baugröße kleiner sind.

Das Prinzip bei dem hier benutzten Sensor ist, dass zwei Massen durch elektrostatische Anregungen gegenphasig in X-Richtung schwingen. Wird der Sensor in Rotation versetzt, werden die Massen aufgrund der Coriolis-Kraft in Richtung der Z-Achse beschleunigt und eine Coriolisbeschleunigung a_c wird gemessen werden. Diese Auslenkung kann kapazitiv gemessen werden und erlaubt ein Rückschluss auf die Drehrate.

Mit der Relativgeschwindigkeit v_{ref} , der Corioliskraft F_c und der Drehrate Ω ergibt sich die Beschleunigung a_c wie folgt:

$$a_c = \frac{F_c}{m} = 2v_{ref} * \Omega \tag{15}$$

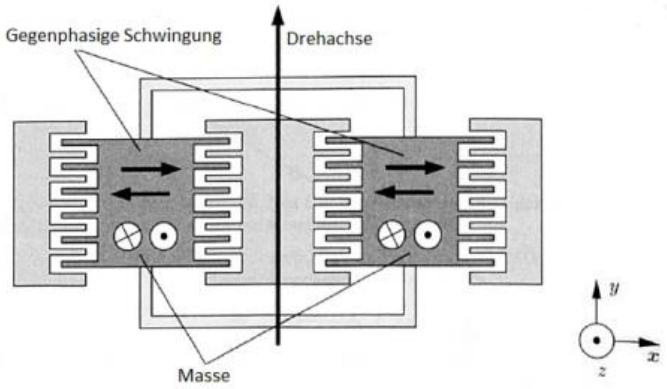


Abbildung 7: Prinzipsskizze des Gyroskops **[6]**

2.3 Kalman Filter [7]

Es ist nötig in abgetasteten Systemen, dass der Kalman Filter diskretisiert wird, denn der Algorithmus soll in einem Rechner implementiert werden. Aufgrund der binären Architektur eines Rechners ist es nicht möglich Ableitung oder Integration Operationen zu realisieren. Deswegen, eine Darstellung mittels Differenz-Gleichung kann benutzt werden, um die kontinuierlichen Kalman-Filter in diskreter Zeit umzuwandeln. Dazu wird die “Forward Euler” Methode ausgewählt, die eine Approximation der Ableitung erlaubt.

$$\frac{x(t_{k+1}) - x(t_k)}{t_{k+1} - t_k} \approx F(t_k)x(t_k) + G(t_k)u(t_k)$$

Wenn h = $t_{k+1} - t_k = \Delta t$ ersetzt wird, das diskrete Modell kann in der folgenden Form geschrieben werden

$$x(t_{k+1}) \approx [1 + hF(t_k)]x(t_k) + hG(t_k)u(t_k)$$
$$x(k+1) = F(k)x(k) + G(k)u(k)$$
$$y(k+1) = H(k)x(k) + D(k)u(k)$$

Für ein Zeitinvariant System wird die rekursive Darstellung eines Kalman-Filters als folgende präsentiert

$$\boldsymbol{x}_{k+1} = \Phi_{k+1,k}\boldsymbol{x}_k + \boldsymbol{\Gamma}_k\boldsymbol{w}_k$$
$$\boldsymbol{w}_k \sim N(0, \boldsymbol{Q}_k)\ (weißes\ Rauschen) \tag{16}$$

$$\boldsymbol{z}_k = \boldsymbol{H}_k\boldsymbol{x}_k + \boldsymbol{v}_k,$$
$$\boldsymbol{v}_k \sim N(0, \boldsymbol{R}_k)(weißes\ Rauschen) \tag{17}$$

$$\hat{\boldsymbol{x}}_{k+1|k} = \Phi_{k+1,k}\hat{\boldsymbol{x}}_{k|k}, \tag{18}$$

$$P_{k+1|k} = \Phi_{k+1,k}P_{k|k}{\Phi^T}_{k+1,k} + \boldsymbol{\Gamma}_k Q_{k+1}{\Gamma^T}_k \tag{19}$$

$$\hat{\boldsymbol{x}}_{k|k} = \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k[\boldsymbol{z}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{k|k-1}] \tag{20}$$

$$P_{k|k} = [\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k]\boldsymbol{P}_{k|k-1} \tag{21}$$

$$\boldsymbol{K}_k = \boldsymbol{P}_{k|k-1}{\boldsymbol{H}^T}_k[\boldsymbol{H}_k\boldsymbol{P}_{k|k-1}{\boldsymbol{H}^T}_k + \boldsymbol{R}_k]^{-1} \tag{22}$$

Gl. (16) ist die Zustandsgleichung genannt. Es ist eine erste Ordnung Gleichung und der Vektor $\boldsymbol{x}_k$ stellt die Parameter oder Zustände dar, die abgeschätzt werden. Gl.(17) ist die sogenannte Beobachtungsgleichung und modelliert die Messung des Prozesses. $\boldsymbol{z}_k$ ist der Beobachtungsvektor und enthält die vermessenen Komponenten. Außerdem, diese Gleichung hängt die Messungen mit dem Zustandsvektor mittels der Beobachtungsmatrix $\boldsymbol{H}_k$ zusammen. Das zufällige messende Rauschen wird mit $\boldsymbol{v}_k$ dargestellt. Gl. (19) stellt die Kovarianz Propagation.

Andererseits, Gl. (20) zeigt, dass die beste Schätzung in $t = k$ mittels des vorhergesagten Zustands plus des Fehlers zwischen dem beobachteten und dem erwarteten Zustand mal eine Konstant ($\boldsymbol{K}_k$) berechnet werden kann. Die wichtigste von dieser Gleichung ist die Verstärkung $\boldsymbol{K}_k$, denn wird bestimmt, wie viel wird die Schätzung des Zustandes in Bezug auf die neue Beobachtung gewechselt. Das heißt, wenn die Elemente von $\boldsymbol{K}_k$ zu klein sind, wird an das Modell mehr vertraut, aber wenn $\boldsymbol{K}_k$ groß ist, wird mehr an die Beobachtung vertraut. $\boldsymbol{K}_k$ ist zeitabhängig und es wird versucht, ihn zu optimieren.

Zusätzlich hängt $\boldsymbol{K}_k$ von $\boldsymbol{R}_k$ ab Gl. (22), wenn eine Messung durchgeführt wird, dessen gemessenes Rauschen klein im Vergleich zum Prior Fehler ist, wird die Verstärkung größer, aber wenn die Messung rauschend bezüglich des Prior Fehlers ist, wird die Verstärkung kleiner und der Filter hat weniger Einfluss der Messung.

Es ist wichtig für die Anwendung des Filters, dass das dynamische Modell, statistisches Rauschen und a priori Daten $[\hat{x}(0), P(0)]$ zur Verfügung stehen. In diesem Fall wird die vorherige Statistik des Rauschens als Weiß und mittelwertfrei dargestellt.

$$\begin{aligned} \xi\{\boldsymbol{w}_k\} = \ \xi\{\boldsymbol{v}_k\} = \boldsymbol{0}, \qquad \xi\{\boldsymbol{x}(0)\} = \ \mu_{\boldsymbol{x}}(0), \\ \xi\{\boldsymbol{w}_k \boldsymbol{w}_j^{\boldsymbol{T}}\} = \boldsymbol{Q}_k \delta_{kj}, \\ \xi\{\boldsymbol{v}_k \boldsymbol{v}_j^{\boldsymbol{T}}\} = R_k \delta_{kj}, \\ \xi\{\boldsymbol{w}_k \boldsymbol{v}_j^{\boldsymbol{T}}\} = 0, \qquad \forall\, j, k, \end{aligned} \tag{23}$$

mit $\mu_{\boldsymbol{x}}(0)$ als Mittelwert und δ_{kj} ist die Kronecker Delta Funktion

$$\delta_{kj} = \begin{cases} 0 \ f\ddot{u}r\ k \neq j \\ 1\ f\ddot{u}r\ k = j \end{cases} \tag{24}$$

Die Matrizen für das Rauschen des Systems und die Kovarianz des Rauschens der Messung können als symmetrische Matrizen dargestellt werden

$$Q_k = \begin{bmatrix} q_{11} & \cdots & q_{1s} \\ \vdots & \ddots & \vdots \\ q_{1s} & \cdots & q_{js} \end{bmatrix} = \begin{bmatrix} \sigma_1^2 & \cdots & \rho_{1s}\sigma_1\sigma_s \\ \vdots & \ddots & \vdots \\ \rho_{1s}\sigma_1\sigma_s & \cdots & \sigma_s^2 \end{bmatrix} \tag{25}$$

Wo ρ_{ij} die Kreuzkorrelation Koeffizient ist. Die Matrix für R_k hat die gleiche Struktur, aber mit der Ausnahme, dass die Ausdrücke außer die Diagonal Null sind. Die Matrix Q kann ausgelegt werden, so dass eine Erhöhung von Q zwei verschiedene Tatsachen hindeutet. Entweder zeigt die Erhöhung eine größere Abhängigkeit der Dynamik von Rauschen oder sie trägt zur Unsicherheit der Fähigkeit des Systems bei, die genaue Dynamik zu beschreiben. Demzufolge wird die Steigerungsrate der Elemente der Matrix P auch zugenommen und somit die Verstärkung des Filters. Auf diese Weise wird die Messung

stärker bewertet. Dabei ist zu beachten, wenn Q erhöht wird, dass der Ausgang des Filters und sein eigenes dynamisches Modell weniger berücksichtigt wird. Entsprechend gilt, dass eine Erhöhung von R die Bedeutung hat, dass die Messungen starkes Rauschen ausgesetzt sind und deswegen sie sollen durch den Filter weniger bewertet werden.

Eine grafische Darstellung des Filters wird im Bild (Abbildung 8) gezeigt,

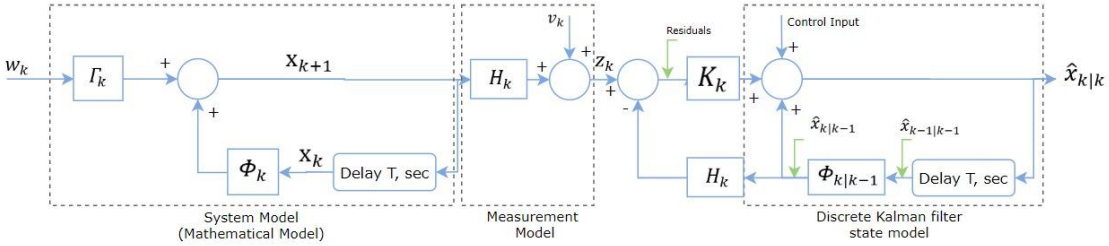


Abbildung 8: Diskret Kalman Filter

Der Kalman-Filter betreibt auf eine prädiktive Weise. Es wird die Gleichung betrachtet

$$\hat{\boldsymbol{x}}_k = \phi\hat{\boldsymbol{x}}_{k-1} + \mathbf{K}_k[\mathbf{z}_k - \boldsymbol{H}\phi\hat{\boldsymbol{x}}_{k-1}] \tag{26}$$

Mit $\hat{x}_k$ als beste Schätzung des Signals x_k für k Zeit. Der erste Ausdruck sagt die Signalschätzung zum Zeitpunkt k vorher, die durch Projektierung der Signalschätzung zum Zeitpunkt $k-1$ in der Zukunft mittels Zustandsübergangsmatrix dargestellt werden kann. Dann wird das Ergebnis dieser Vorhersage beim Verglich zwischen die projektierte Schätzung und die Beobachtung korrigiert, die mit der Kalman Verstärkung bewertet sind. Das Ziel besteht darin, einstufige Schätzer zu finden. Dazu wird die vorhersagende Stufe berücksichtigt

$$\hat{\boldsymbol{x}}_{k|k-1} = \phi\hat{\boldsymbol{x}}_{k-1|k-1} \tag{27}$$

$$\boldsymbol{P}_{k|k-1} = \phi\boldsymbol{P}_{k-1|k-1}\phi^T + \boldsymbol{Q}_{k-1}$$

Dann werden die Ergebnisse der Vorhersage benutzt, um den Schätzprozess durch Gleichungen (20), (21) und (22) zu aktualisieren oder zu korrigieren. Das Schätzproblem fängt mit Gleichung an

$$\hat{\boldsymbol{x}}_{k+1|k} = \phi\hat{\boldsymbol{x}}_k \tag{28}$$

Wo $\hat{x}_k = \hat{x}_{k|k}$ das aktuelle geschätzte Signal ist und $\hat{x}_{k+1|k}$ ist die projektierte Vorhersage, die eine Abtastperiode verschoben wird. Das heißt, für das gegebene gefilterte Signal wird das projektierte Signal, das mittels Zustandsübergangmatrix ermittelt wird,

die beste Vorhersage des zukünftigen Wertes des Signales ist. Wenn das Rauschen für die Vorhersage vernachlässigt wird, und die Gl. (20) wird in der Gl. (28) aufgenommen, dann wird es sich Gl. (29) ergeben

$$\hat{\boldsymbol{x}}_{k+1|k} = \Phi\hat{\boldsymbol{x}}_{k|k-1} + \mathrm{G}_k\left[\mathbf{z}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{k|k-1}\right] \tag{29}$$

$$\mathrm{G}_k = \Phi\mathrm{K}_k$$

Andererseits wird für das Schätzproblem die Fehler-Kovarianzmatrix ($\mathrm{P}_{k+1|k}$) erfordert. Die Fehler-Kovarianzmatrix wird von der Gl. (27) abgeleitet, wenn die Zeit durch eine Abtastperiode verschoben wird.

$$\boldsymbol{P}_{k+1|k} = \Phi\boldsymbol{P}_{k|k}\Phi^T + \boldsymbol{Q}_k$$

Ausschließlich wird die einstufige Vorhersage Fehler-Kovarianzmatrix ermittelt, wenn die Gl. (21) in der obigen Gleichung aufgenommen wird.

$$\boldsymbol{P}_{k+1|k} = [\Phi - G_k H_k]\, P_{k|k-1}\Phi^T + \boldsymbol{Q}_k \tag{30}$$

Der konventionelle zeitdiskrete Kalman-Filter Algorithmus (Gln. (16)-(22)) kann einfacher als folgend geschrieben werden

Vorhergesagter Zustand:

$$\hat{\boldsymbol{x}}_{k|k-1} = \Phi_{k|k-1}\hat{\boldsymbol{x}}_{k-1|k-1} \tag{31}$$

$$\hat{\boldsymbol{x}}(0|0) = \hat{\boldsymbol{x}}(0)$$

Vorhergesagter Beobachtung:

$$\hat{\boldsymbol{z}}_{k|k-1} = \boldsymbol{H}\hat{\boldsymbol{x}}_{k|k-1} \tag{32}$$

Innovation:

$$\mathbf{v}_k = \mathbf{z}_k - \hat{\boldsymbol{z}}_{k|k-1} \tag{33}$$

Vorhergesagte Kovarianz:

$$\boldsymbol{P}_{k|k-1} = \Phi\boldsymbol{P}_{k-1|k-1}\Phi^T + GQ_kG^T \tag{34}$$

$$P(0|0) = P(0)$$

Innovierte Kovarianz:

$$\mathrm{S}_k = H\mathrm{P}_{k|k-1}H^T + R \tag{35}$$

Kalman Verstärkung:

$$\mathrm{K}_k = \mathrm{P}_{k|k-1} H^T S_k^{-1} \tag{36}$$

Zustandsaktualisierung:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + \mathrm{K}_k \mathrm{v}_k \tag{37}$$

Aktualisierung der Kovarianz:

$$P_{k|k} = \mathrm{P}_{k|k-1} - \mathrm{K}_k \mathrm{S}_k K_k^T \tag{38}$$

3 Hardware und Software

Für die Ausführung dieses Projekts wird ein Mega-Arduino verwendet, dessen Eigenschaften später beschrieben und in Echtzeit über die Matlab-Simulink Software kommuniziert werden, in der die Blockprogrammierung durchgeführt wurde. Zur Messung des Winkels wird ein Modul mit Beschleunigungsmesser und ein Gyroskop mit 6 Freiheitsgraden MPU6050 verwendet.

3.1 Mikrocontroller – Arduino Mega 2560

Arduino Mega 2560 ist eine auf dem ATMega2560 basierende Mikrocontroller-Platine. Sie hat 54 digitale I/O-Pins (von denen 14 als PWM-Ausgänge verwendet werden können), 16 analoge Eingänge, 4 UARTs (serielle Hardware-Schnittstellen), einen 16-MHz-Kristalloszillator, eine USB-Schnittstelle, einen Stromanschluss, einen ICSP-Header und einen Rückstellknopf [8].

Abbildung 9: Darstellung der unterschiedlichen Koordinatensysteme **[8]**

Hauptsächlich von der Arduino-Platine wurden die AD-Eingänge und die I2C-Kommunikationsstifte verwendet

3.2 Sensoreinheit – MPU6050

Die MPU-60X0 ist ein 6-Achs-MotionTracking-Gerät, das ein 3-Achs-Gyroskop, einen 3-Achs-Beschleunigungsmesser und einen Digital Motion Processor ™ (DMP) in einem kleinen Paket von 4x4x0,9 mm2 kombiniert. Es verfügt über einen dedizierten I^2C-Sensorbus und akzeptiert direkt Eingaben von einem externen 3-Achsen-Kompass, um einen vollständigen 9-Achsen-MotionFusion-Ausgang bereitzustellen. Das Motion Tracking-Gerät MPU-60X0 enthält eine 6-Achsen-Integration, integrierte MotionFusion - und Laufzeitkalibrierungs-Firmware. Ist auch für die Verbindung mit mehreren digitalen Sensoren ohne Trägheitsmoment ausgelegt, z. B. Drucksensoren, und zwar an seinem zusätzlichen I^2C-Anschluss.

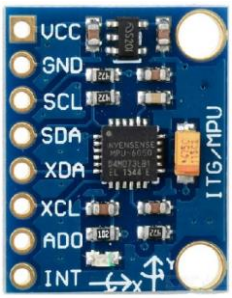


Abbildung 10: Darstellung der unterschiedlichen Koordinatensysteme **[8]**

- **Beschleunigungsmesser**

Die Drei-Achsen-MEMS-Beschleunigungsmesser mit 16-Bit-ADCs und Signalkonditionierung. Der 3-Achsen-Beschleunigungsmesser der MPU-6050 verwendet separate Prüfmassen für jede Achse. Die Beschleunigung entlang einer bestimmten Achse bewirkt eine Verschiebung der entsprechenden Prüfmasse, und kapazitive Sensoren erfassen die Verschiebung differentiell Abbildung 6. Die Architektur der MPU-6050 verringert die Anfälligkeit der Beschleunigungsmesser für Fertigungsschwankungen sowie für thermische Abweichungen. Wenn das Gerät auf einer ebenen Fläche aufgestellt wird, misst es 0 g auf der X- und Y-Achse und + 1 g auf der Z-Achse. Der Skalierungsfaktor des Beschleunigungsmessers ist werkseitig kalibriert und nominell unabhängig von der Versorgungsspannung. Jeder Sensor verfügt über einen dedizierten Sigma-Delta-ADC zur Bereitstellung digitaler Ausgänge. Der Skalenendwert des digitalen Ausgangs kann auf ± 2 g, ± 4 g, ± 8 g oder ± 16 g eingestellt werden. In diesem Project benutzen wir ± 2 g.

- **Gyroskopen**

Das Modul MPU-6050 besteht aus drei unabhängigen Vibrations-MEMS-Frequenzgyroskopen, die die Drehung um die X-, Y- und Z-Achse erfassen. Wenn die Gyros um eine der Erfassungsachsen gedreht werden, verursacht der Coriolis-Effekt eine Vibration, die von einem kapazitiven Abnehmer erfasst wird. Das resultierende Signal wird verstärkt, demoduliert und gefiltert, um eine Spannung zu erzeugen, die proportional zur Winkelrate ist. Diese Spannung wird unter Verwendung einzelner On-Chip-16-Bit-Analog-Digital-Wandler (ADCs) digitalisiert, um jede Achse abzutasten. Der Messbereich der Kreiselsensoren kann digital auf ± 250, ± 500, ± 1000 oder ± 2000 Grad pro Sekunde (dps) programmiert werden. Die ADC-Abtastrate kann von 8.000 Abtastungen pro Sekunde bis zu 3,9 Abtastungen pro Sekunde programmiert werden, und vom Benutzer wählbare Tiefpassfilter ermöglichen einen weiten Bereich von Grenzfrequenzen

3.3 Pendel - Aufbau

Abbildung 11: Aufbau des Pendels

Für die Umsetzung dieses Projekts wurde ein Prototyp verwendet (siehe Abbildung 11), Der eine Aluminium-Metallbasis und ein Profil für die Pendelstütze aufweist. Das Pendel besteht aus einer Kupferplatte, auf der das Messmodul MPU6050 montiert ist. Im oberen Teil der Konstruktion befindet sich ein Potentiometer, das zur Überprüfung im Implementierungs- und Testabschnitt verwendet wird.

3.4 Matlab – Arduino-Support von Simulink

Das Simulink Support Package für Arduino-Hardware ermöglicht das Erstellen und Ausführen von Simulink-Modellen auf Arduino-Platinen. Das Ziel umfasst eine Bibliothek

von Simulink-Blöcken zum Konfigurieren und Zugreifen auf Arduino-Sensoren, -Aktoren und -Kommunikationsschnittstellen. Darüber hinaus können mit dem Ziel auf dem Arduino-Board aus denselben Simulink-Modellen, aus denen Sie die Algorithmen entwickelt haben, ausgeführte Algorithmen überwacht und optimiert werden.

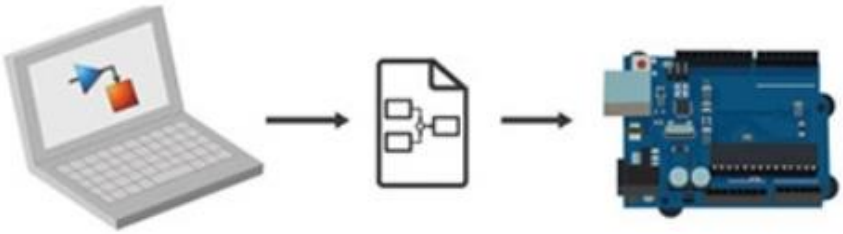


Abbildung 12: – Arduino-Support von Simulink **[3]**

Das Simulink-Unterstützungspaket für Arduino ermöglicht die Entwicklung von Algorithmen in Simulink, einer Blockdiagrammumgebung zur Modellierung dynamischer Systeme und zur Entwicklung von Algorithmen, sowie deren autonome Ausführung auf dem Arduino-Gerät. Das Support-Paket erweitert die Funktionalität von Simulink um Blöcke zur Konfiguration von Arduino-Sensoren sowie zum Lesen und Schreiben von Daten. Nachdem das Simulink-Modell erstellt wurde, kann man eine Simulation durchführen, die Parameter des Algorithmus anpassen, bis das gewünschte Ergebnis erreicht ist, und den fertigen Algorithmus zur eigenständigen Ausführung auf das Gerät herunterladen. Der MATLAB-Funktionsblock ermöglicht die Einbindung von MATLAB-Code in das Simulink-Modell.

4 Implementierung des Kalman-Filter [9]

4.1 Kalman-Filter mit Matlab

Die Implementierung des Filters wird auf die theoretische Beschreibung im Abschnitt 2.3 und auf die Messungen der Beschleunigungssensor und Gyroskop basiert. Der erste Schritt besteht darin, die Darstellung des Modells zu beschreiben. Dazu wird eine modifizierte Version der Gl. (16):

$$x_k = \phi x_{k-1} + \boldsymbol{B} u_k + \boldsymbol{\Gamma}_k w_k \tag{39}$$

Für den Fall des Pendels wird die Zustandsvektor wie in der folgenden Gleichung ausgewählt:

$$x_k = \begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_k \tag{40}$$

Das heißt, der Ausgang ist der Winkel des Pendels aber auch der Bias-Drift $\dot{\theta}_b$, die auf die Messungen der Beschleunigungssensor und Gyroskop basiert sind. Dann kann auch die Winkelgeschwindigkeit abgeschätzt werden, wenn man den Bias von der Messung des Gyroskops subtrahiert. Der Zustandsübertragungsmatrix wird als folgend definiert:

$$\Phi = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \tag{41}$$

Der Stellgröße u_k ist auf diesem Fall die Messung des Gyroskops mit der Einheit Grad per Sekunde zum Zeitpunkt k, die sogenannte Drehwinkel $\dot{\theta}$. Die Zustandsdarstellung kann wie folgt umformuliert werden:

$$\boldsymbol{x}_k = \Phi x_{k-1} + \boldsymbol{B}\dot{\theta}_k + w_k \quad mit\ \Gamma(k) = \boldsymbol{I} \tag{42}$$

Wo $\boldsymbol{B}$ der Eingangsvektor ist und seine zweite Komponente Null gewählt wird, denn der Bias darf nicht direkt aus der Drehwinkel berechnet werden.

$$\boldsymbol{B} = \begin{bmatrix} \Delta t \\ 0 \end{bmatrix} \tag{43}$$

Die Kovarianzmatrix wird denn modelliert, sodass die Kovarianz der geschätzten Zustände der Beschleunigungssensor und Bias berücksichtigt wird. Zusätzlich wird die Schätzung des Bias und der Beschleunigungssensor als unabhängig modelliert, damit die Matrix in der Gl. (25) nur die Varianz der geschätzten Zustände enthält.

$$\Phi = \begin{bmatrix} Q_\theta & 0 \\ 0 & Q_{\dot{\theta}_b} \end{bmatrix} \Delta t \tag{44}$$

Es ist sinnvoll, dass die Kovarianzmatrix von der Abtastzeit abhängt, denn je länger die Zeit für die Aktualisierung, desto mehr Rauschen das System enthält und das Gyroskop könnte driften.

Andererseits wird das Modell der Beobachtung wie in der Gl. (17) definiert. Auf diesem Fall wird nur die Messung der Beschleunigungssensor berücksichtigt, sodass $\boldsymbol{H}$ wie folgt, beschreibt wird:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{45}$$

Das Rauschen der Messung wird wie in Gl. (17) dargestellt. Für diese Anwendung ist $\boldsymbol{R}$ keine Matrix, sondern die Varianz der Messung, denn die Kovarianz derselbe Variable ist gleich als die Varianz der Variable $\boldsymbol{R} = \xi\{\boldsymbol{v}_k \boldsymbol{v}_k^T\} = \sigma_v^2$. Nach der Beschreibung des Modells für diesen spezifischen Fall wird die Erklärung der Implementierung der Gleichungen des Kalman-Filter (Gln. (31)-(38)) durchgeführt.

Vorhergesagter Zustand:

Es wird wie in Gl.(31) [9] mit einem zusätzlichen Stellgröße beschreibt

$$\hat{x}_{k|k-1} = \Phi_{k|k-1}\hat{x}_{k-1|k-1} + \boldsymbol{B}\dot{\theta}_k \tag{46}$$

$$\begin{bmatrix}\theta \\ \dot{\theta}_b\end{bmatrix}_{k|k-1} = \begin{bmatrix}1 & -\Delta t \\ 0 & 1\end{bmatrix}\begin{bmatrix}\theta \\ \dot{\theta}_b\end{bmatrix}_{k-1|k-1} + \begin{bmatrix}\Delta t \\ 0\end{bmatrix}\dot{\theta}_k$$

$$\begin{bmatrix}\theta \\ \dot{\theta}_b\end{bmatrix}_{k|k-1} = \begin{bmatrix}\theta + \Delta t(\dot{\theta} - \dot{\theta}_b) \\ \dot{\theta}_b\end{bmatrix}$$

Vorhergesagte Kovarianz:

Es wird die Gl. (33) angewendet, wo $(G = I)$ und P eine $2x2$ Matrix ist

$$P_{k|k-1} = \Phi \mathrm{P}_{k-1|k-1}\Phi^T + \mathrm{Q}_k \tag{47}$$

$$\begin{bmatrix}P_{00} & P_{01} \\ P_{10} & P_{11}\end{bmatrix} = \begin{bmatrix}1 & -\Delta t \\ 0 & 1\end{bmatrix}\begin{bmatrix}P_{00} & P_{01} \\ P_{10} & P_{11}\end{bmatrix}_{k-1|k-1}\begin{bmatrix}1 & 0 \\ -\Delta t & 1\end{bmatrix} + \begin{bmatrix}Q_\theta & 0 \\ 0 & Q_{\dot{\theta}_b}\end{bmatrix}\Delta t$$

$$\begin{bmatrix}P_{00} & P_{01} \\ P_{10} & P_{11}\end{bmatrix} = \begin{bmatrix}P_{00} + \Delta t(P_{01} - \Delta t P_{11}) + Q_\theta \Delta t & P_{01} - \Delta t P_{11} \\ P_{10} - \Delta t P_{11} & P_{11} + Q_{\dot{\theta}_b}\Delta t\end{bmatrix}$$

Innovation:

Hier werden Gln. (32)-(33) benutzt, um die Differenz zwischen dem gemessenen Winkel und dem apriorischen Zustand zu berechnen

$$\mathbf{v}_k = \mathbf{z}_k - [1 \quad 0]\begin{bmatrix}\theta \\ \dot{\theta}_b\end{bmatrix}_{k|k-1} \tag{48}$$

$$\mathrm{v}_k = \mathrm{z}_k - \theta_{k|k-1}$$

Innovierte Kovarianz:

$$S_k = H\mathrm{P}_{k|k-1}H^T + R \tag{49}$$

$$S_k = [1 \quad 0]\begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k-1}\begin{bmatrix} 1 \\ 0 \end{bmatrix} + R$$

$$S_k = P_{00_{k|k-1}} + \sigma_v^2$$

Die innovierte Kovarianz versucht zu vorhersagen, inwieweit auf die Messung vertraut werden soll, die auf der Fehler-Kovarianzmatrix und der Kovarianzmatrix der Messung basiert ist. Je größer der Wert des Rauschens der Messung, desto größer der Wert von S_k, das heißt, dass es nicht auf die kommende Messung vertraut wird

Kalman Verstärkung:

Aus der Gl. (36) wird die Kalman Verstärkung wie folgt berechnen:

$$\begin{bmatrix} K_0 \\ K_1 \end{bmatrix}_k = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k-1}\begin{bmatrix} 1 \\ 0 \end{bmatrix} + S_k^{-1} \tag{50}$$

$$\begin{bmatrix} K_0 \\ K_1 \end{bmatrix}_k = \frac{\begin{bmatrix} P_{00} \\ P_{10} \end{bmatrix}_{k|k-1}}{S_k}$$

Es ist wichtig zu betonen, dass in diesem Fall S_k keine Matrix ist, und deswegen darf man direkt durch S_k dividieren.

Zustandsaktualisierung:

Unter Anwendung von Gl. (37) wird der Aktualisierung der Zustand durchgeführt

$$\begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_{k|k} = \begin{bmatrix} \theta \\ \dot{\theta}_b \end{bmatrix}_{k|k-1} + \begin{bmatrix} K_0 \\ K_1 \end{bmatrix}_k \mathrm{v}_k \tag{51}$$

Aktualisierung der Kovarianz:

Zuletzt wird für die Aktualisierung der Kovarianzmatrix Gl. (21) verwendet

$$\begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k} = \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} K_0 \\ K_1 \end{bmatrix}_k [1 \quad 0]\right)\begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k-1} \tag{52}$$

$$\begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k} = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}_{k|k-1} - \begin{bmatrix} K_0P_{00} & K_0P_{01} \\ K_1P_{00} & K_1P_{01} \end{bmatrix}$$

Die Implementierung des Filters wird mit MATLAB/SIMULINK durchgeführt. Dazu wird eine Modellbildung des Systems aufgebaut, das aus einer Datenerfassung, einer MATLAB Funktion und einem Ausgang besteht. Die Datenerfassung des Sensors wird durch

I^2C abgelesen. Der Ausgang enthält den geschätzten Winkel. Abbildung 13 zeigt den allgemeinen Aufbau des SIMULINK Modells

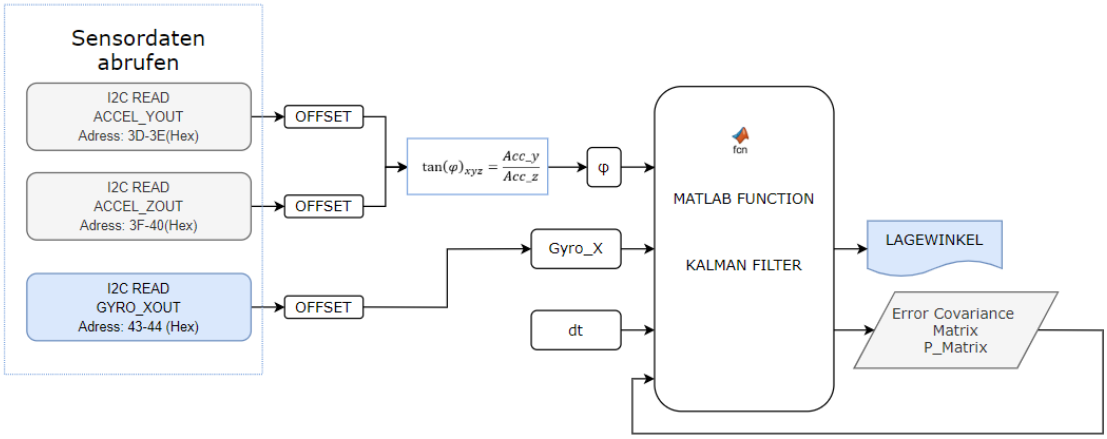


Abbildung 13: – Modellbildung des KALMAN Filter mit MATLAB/SIMULINK

Im Abbildung 13 kann man die Variablen $\varphi, Gyro_x, dt\ und\ P_{matrix}$ erkennen, den $z_k, \dot{\theta}_k, \Delta t\ und\ P_{k|k-1}$ zu entsprechen.

In der MATLAB Funktion KALMAN FILTER werden die Gleichungen (46)-(52) angewendet, so dass der Lagewinkel als Ausgang des Filters nehmen kann, um mit den Ausgängen des komplementären Filters und dem Potenziometer vergleichen zu können. Das Auflaufdiagramm des KALMAN FILTER Funktion wird in der Abbildung 14 zusammengefasst. Der komplette Code in MATLAB der Funktion kann im Anhang A.1 gefunden werden.

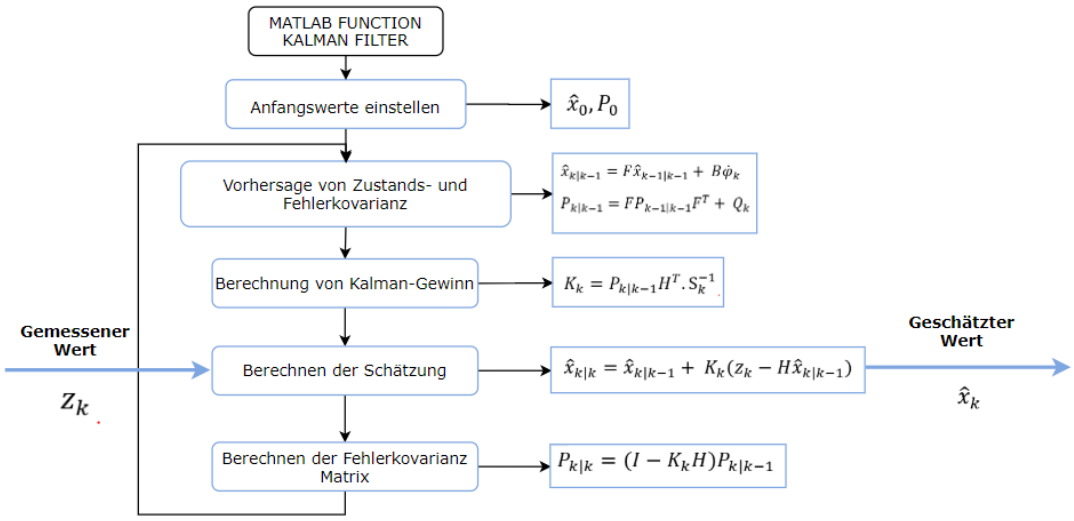


Abbildung 14: – Auflaufdiagramm der KALMAN FILTER Funktion

4.2 Komplementärer Filter [10]

Als Alternative für die Ergänzung der Lagewinkel des Pendels wird der sogenannte komplementäre Filter benutzt. Der Filter besteht aus einem Tiefpassfilter und einem Hochpassfilter, die in diesem Fall auf die gemessenen Signale der Beschleunigungssensor und des Gyroskops angewendet werden. Die Strukturen eines Tiefpassfilters und eines Hochpassfilters können durch Gl. (53) beschrieben werden

$$G(s) = \frac{1}{1+\tau s}\ (Tiefpass) \tag{53}$$

$$G(s) = \frac{\tau s}{1+\tau s}\ (Hochpass)$$

Einerseits wird der Tiefpassfilter dafür benutzt, die Hochfrequenzschwingungen der Messung der Beschleunigungssensor zu dämpfen. Andererseits wird der Hochpassfilter dafür verwendet, die Niederfrequenzsignale wie die Drift des Gyroskops zu beseitigen. Bevor das Signal des Gyroskops filtert wird, wird zuerst es integriert. Dann wird dieses Signal mit dem gefilterten Signal der Beschleunigungssensor addiert und das Ergebnis wird als die Lagerwinkel dargestellt, wie in der Abbildung 15 gezeigt wird.

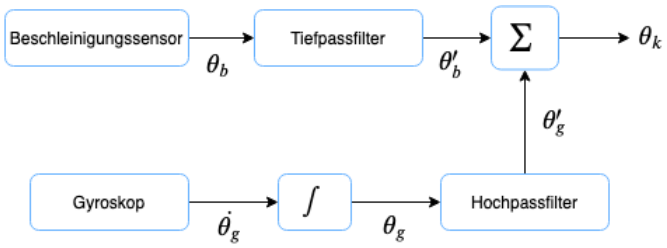


Abbildung 15: Komplementärer Filter für die Lagerwinkel

4.3 Initialisierung des MPU6050-Moduls und der Register

Für die Konfiguration des Sensors über I^2C werden folgende Daten geladen:

Tabelle 1: MPU6050-Moduls und der Register - Einstellen

Register (Hex)	Beschreibung	Parameter	Clock Source
6B	Power Management 1	Bit[2:0] = 0	Internal 8MHz Oszillator
1B	Gyroskope Konfiguration	Bit[2:0] = 0	± 250 °/s oder 131 LSB/°/s
2C	Beschleunigungssensor Konfiguration	Bit[2:0] = 0	± 2g

Für die Implementierung des Filters und die Schätzung des Pendelwinkels müssen die Daten aus dem folgenden Register gelesen werden:

Tabelle 2: MPU6050-Moduls und der Register - Lesen

Register (Hex)	Beschreibung	Bits
3D	ACCEL_YOUT_H	Bit [15:8]
3F	ACCEL_ZOUT_H	Bit [15:8]
43	GYRO_XOUT_H	Bit [15:8]

Die vollständige Implementierung des MATLAB-Simulink-Filters ist in Abbildung 16 zu sehen. Dort wird auch ein AD-Wert gemessen, um den Vergleich mit der vom Potentiometer gelieferten Messung vorzunehmen.

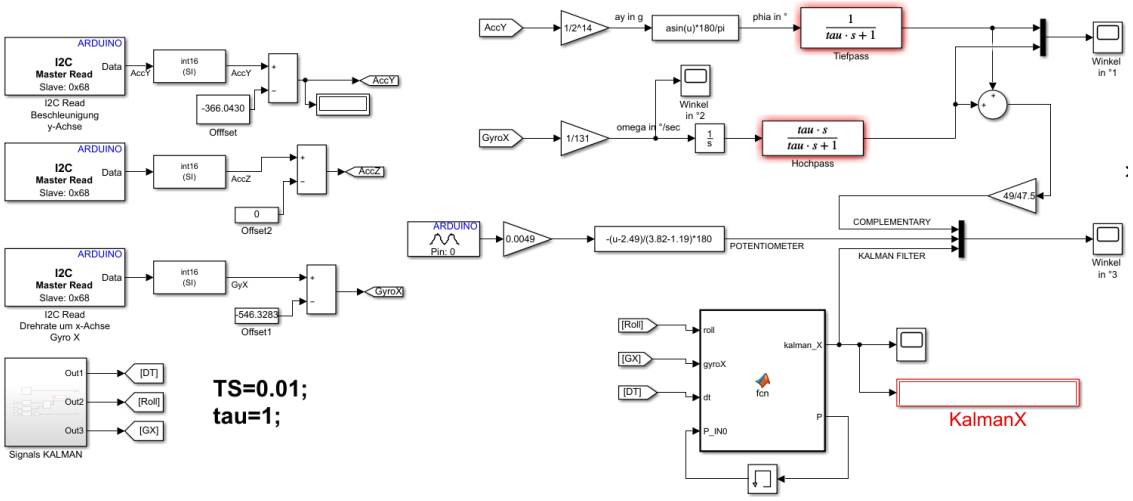


Abbildung 16: –KALMAN Filter mit MATLAB/SIMULINK

4.4 Diskussion und Resultate

Die Ergebnisse, die aus der Schätzung des Winkels unter Verwendung des Kalman-Filters erhalten wurden, sind nachstehend gezeigt. Die Leistung des Filters hängt von den Parametern ($Q_\theta, Q_{\dot{\theta}_b}, \sigma_v^2$) stark ab, wie in der Abbildung 17 und Abbildung 18 gemerkt werden kann. Die ausgewählten Werten der Parameter nach verschieden Versuchen sind:

Tabelle 3: Einstellung des Filter Parameters

Parameter	1	2
Q_θ	0.008	0.0007
$Q_{\dot{\theta}_b}$	0.03	0.003
σ_v^2	0.03	0.15

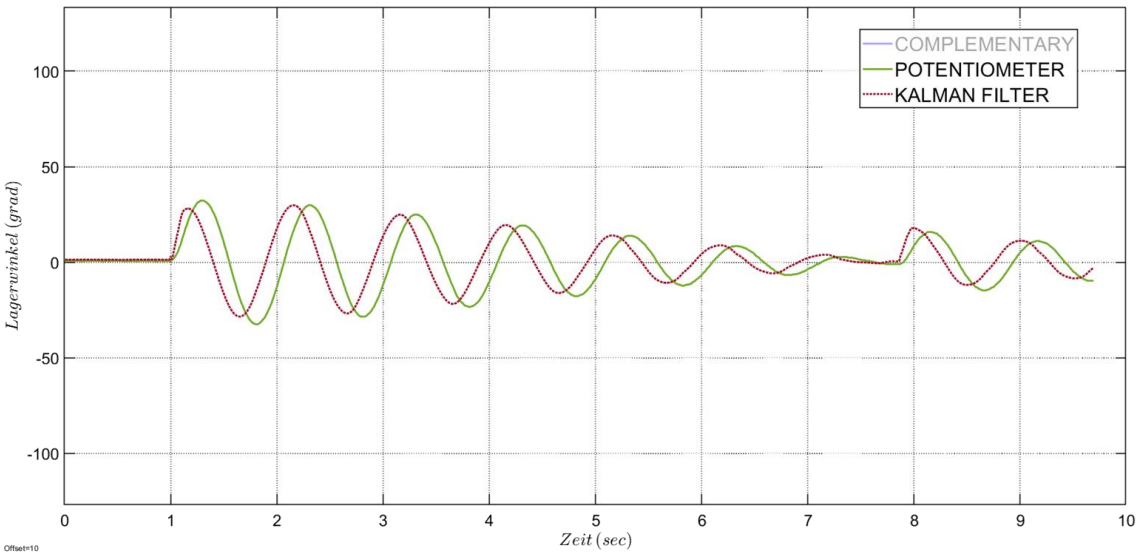


Abbildung 17: –KALMAN Filter mit Einstellung 1

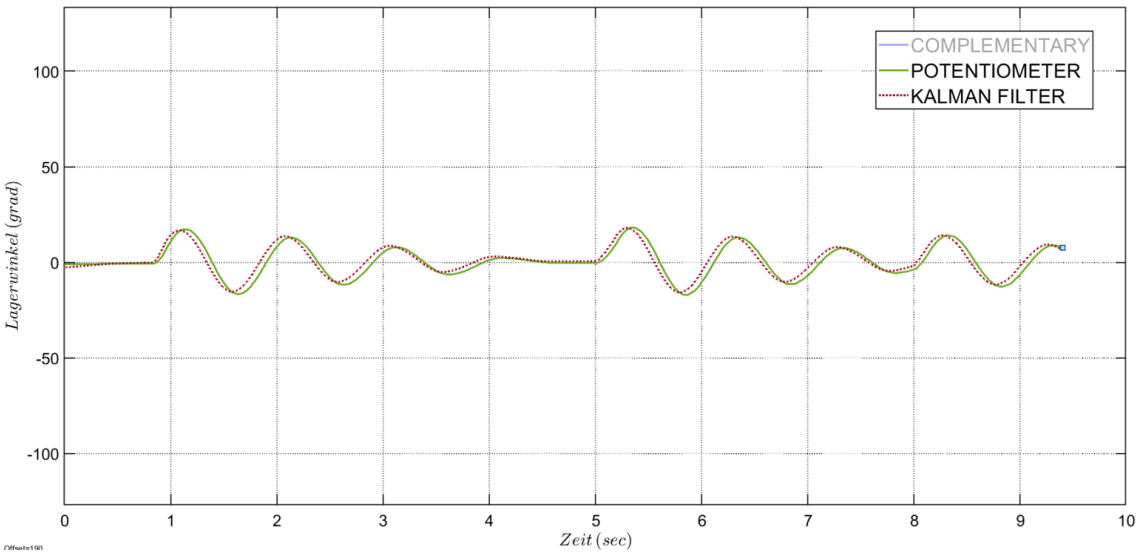


Abbildung 18: –KALMAN Filter mit Einstellung 2

Nach der Einstellung der Parameter wird der Kalman Filter mit dem Potenziometer und komplementärem Filter verglichen. Abbildung 19 und Abbildung 20 zeigen die Lagerwinkel in Grad für die drei verschieden Verfahren gegen die Zeit in Sekunden, wo die blaue, grüne und rote Kurven, dem komplementären Filter, dem Potenziometer und dem Kalman Filter entsprechen. Es kann von Abbildung 19 abgelesen werden, dass der Kalman Filter ein gutes Verhältnis im Vergleich mit dem komplementären Filter zeigt, denn beide folgen dem Potenziometer richtig. Es ist auch lesbar, dass sowohl der komplementäre Filter als auch der Kalman Filter für große Werte der Lagerwinkel fehlerbehaftet sind.

In der Abbildung 20 kann auch die Vergleichung für kleine Werte der Lagerwinkel beobachtet werden. In diesem Fall beide Verfahren folgen dem Potenziometer genauer als mit großen Werten der Lagerwinkel. Das heißt, die Reaktionsgeschwindigkeit der beiden Filter wird negativ durch große Werte der Lagerwinkel beeinflusst. Um die Reaktionsgeschwindigkeit des Kalman Filters muss die Parameter ($Q_\theta, Q_{\dot{\theta}_b}, \sigma_v^2$) gewechselt werden. Es ist aber sinnvoll zu berücksichtigen, dass ein schneller Filter auch mehr Rauschen der Messung enthält wird, und deswegen muss es eine Bilanz gefunden werden.

Andererseits wird die Winkelgeschwindigkeit von beiden Filtern verglichen. In diesem Fall zeigt der Kalman Filter ein besseres Verhalten im Vergleich mit dem komplementären Filter. Das kann erklärt werden, wenn man die Architektur des komplementären Filters analysiert, denn der Filter wurde für die Lagerwinkel entworfen und nicht für die Winkelgeschwindigkeit, die einfach aus dem Gyroskop genommen wird.

Obwohl der Kalman Filter ähnliche Leistung wie der komplementäre Filter präsentiert, hat der Vorteil, dass für Mehrgrößensysteme angewendet werden kann. In diesem Fall hat den Filter zwei verscheiden Variable ($\theta, \dot{\theta}$) abgeschätzt und deswegen ist ein flexibleres Verfahren als der komplementäre Filter. Für die Messung der Geschwindigkeit wird das Gyroskop wegen seiner Natur driftet, kann aber dieser Effekt mit dem Kalman Filter vermeiden werden.

Schließlich kann es geäußert werden, dass der Kalman Filter ein robustes Werkzeug ist, um Signale abzuschätzen. Auf diesen Grund ist der Filter für Anwendungen mit geringer Speicherkapazität nicht geeignet, denn seine Implementierung anfordert mehr rechnerische Ressourcen. In diesem spezifischen Fall wurde keine große Verbesserung in der Bestimmung der Lagerwinkel im Vergleich mit dem komplementären Filter gemerkt. Das heißt, der komplementäre Filter ist auch eine effektive Alternative, denn er verbraucht weniger Ressourcen und präsentiert eine gute Leistung. Die Auswahl von einem Verfahren hängt stark von der Natur der Anwendung ab, denn beide Verfahren haben vor und Nachteile.

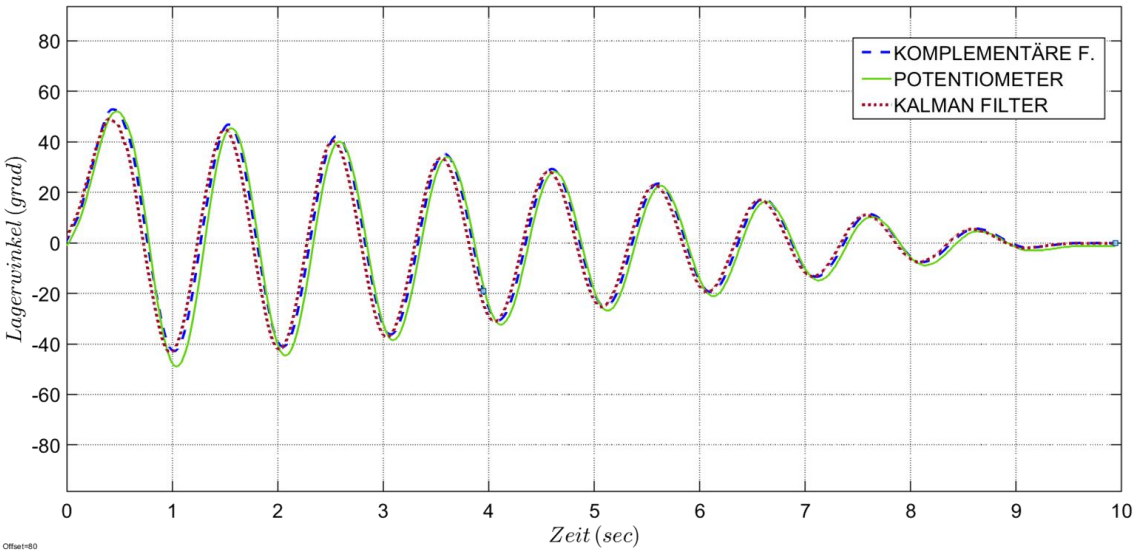


Abbildung 19: – Lagerwinkel gegen Zeit für mittlere Winkelwerte

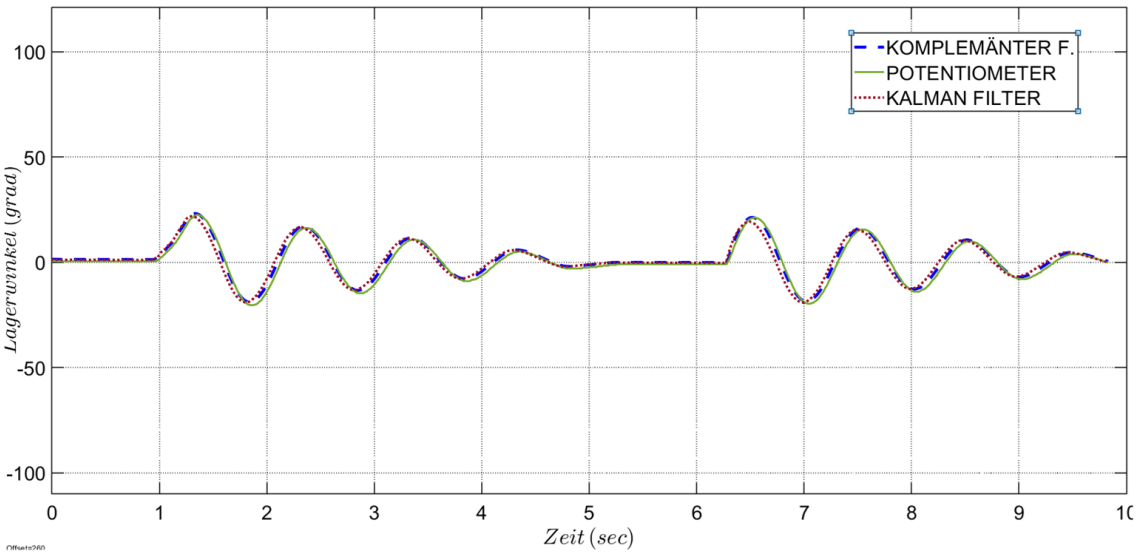


Abbildung 20: – Lagerwinkel gegen Zeit für kleine Winkelwerte

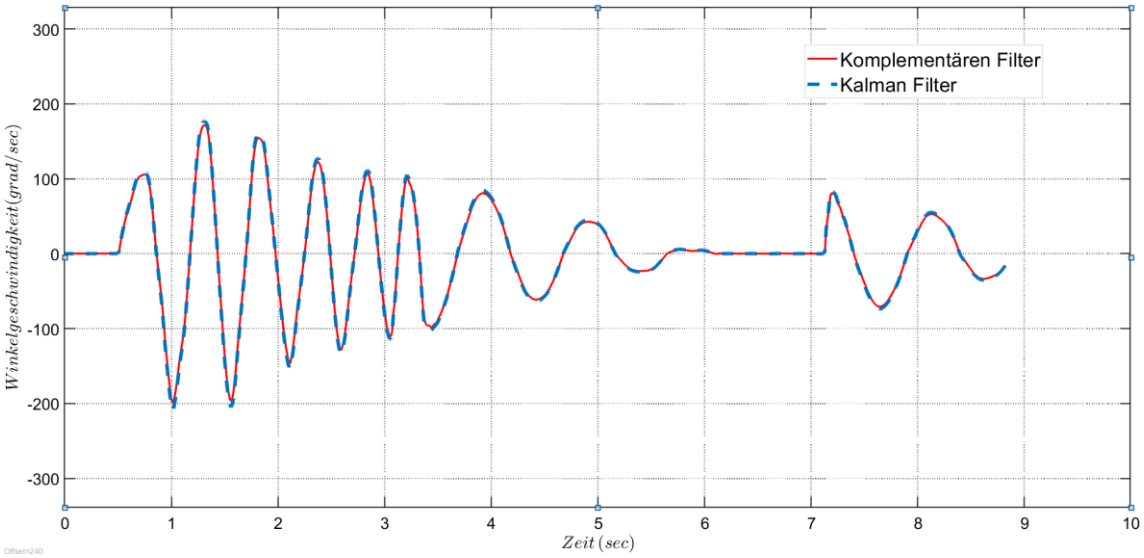


Abbildung 21: – Winkelgeschwindigkeit gegen Zeit für komplementären Filter und Kalman Filter

5 Zusammenfassung und Ausblick

In diesem Projekt wurde eine Implementierung des Kalman-Filters für eine praktische Anwendung bei der Messung der Position eines Pendels durchgeführt, und im Gegenzug wurde ein Vergleich mit einem Kommentarfilter und dem Wert des Winkels durchgeführt, der mit einem Potentiometer gemessen wurde. Für die Ausführung dieses Projekts wurde ein Arduino Mega verwendet, der in Echtzeit über die MATLAB-Simulink-Software kommuniziert wird, in der die Blockprogrammierung durchgeführt wurde. Ein MEMS-Modul mit einem Beschleunigungsmesser und einem Gyroskop mit 6 Freiheitsgraden MPU6050 wurde zur Winkelmessung verwendet.

Für die Implementierung des Kalman-Filters wurde dies schrittweise durchgeführt. Die Werte der Zustandsmatrix und der Kovarianzmatrix wurden zuerst initialisiert. Diese Matrix gibt die Differenz zwischen der Kalman-Filterschätzung und dem wahren Wert an, das heißt, sie gibt den Genauigkeitsgrad an. Dann folgt der Vorhersageprozess, bei dem die Vorhersage für die aktuelle Schätzung die Fehlerkovarianz aus dem letzten Zustand berechnet. Später im Schätzungsprozess wurde der Gewinn von Kalman berechnet, der angibt, wie stark die Messung gewichtet ist. Als nächstes wird die verbesserte Schätzung mit Hilfe der Differenz zwischen den gemessenen Werten und den vorhergesagten Schätzungen bestimmt, mit der Kalman-Verstärkung gewichtet und mit der vorhergesagten Schätzung aggregiert. Dies bestimmt die neue Fehler-Kovarianzmatrix für die verbesserte Schätzung.

Der Filter wurde mit MATLAB / SIMULINK implementiert. Zu diesem Zweck wurde ein Systemmodell erstellt, das aus einer Datenerfassung, einer MATLAB-Funktion und einer Ausgabe besteht. Die Erfassung der Sensordaten wurde von I^2C gelesen und die Ausgabe enthält den geschätzten Winkel. Die geschätzte Bewertung des Kalman-Filters wird schließlich mit dem Komplementärfilter verglichen, der aus einem Tiefpassfilter und einem Hochpassfilter besteht, die in diesem Fall auf die gemessenen Signale des Beschleunigungssensors und des Gyroskops angewendet werden. Der Tiefpass dient zum einen dazu, die hochfrequenten Schwingungen der Beschleunigungssensormessung zu dämpfen. Andererseits wird das Hochpassfilter verwendet, um niederfrequente Signale zu eliminieren.

In der Ergebnisphase zeigt das Kalman-Filter ein angemessenes Verhalten gegenüber dem Komplementärfilter, da beide dem vom Potentiometer gemessenen Wert folgen, jedoch für große Werte der Orientierungswinkel einen Fehler aufweisen.

Schließlich wurde die Winkelgeschwindigkeit beider Filter verglichen und der Kalman-Filter zeigt eine bessere Leistung im Vergleich zum Komplementärfilter.

Literaturverzeichnis

[1] R. Kleinbauer, Kalman Filtering Implementation with Matlab, Stuttgart Deutschland: Universität Stuttgart, 2004.

[2] T. Michaelsen, Hamburg, Deutschland: Hochschule für Angewandte Wissenschaften Hamburg, 2018.

[3] Matlab, „Matlab," Mathworks, Februar 2020. [Online]. Available: https://es.mathworks.com/discovery/arduino-programming-matlab-simulink.html.

[4] N. A. a. S. Administration, „NASA," [Online]. Available: https://www.grc.nasa.gov/www/k-12/airplane/rotations.html.

[5] https://www.makerlab-electronics.com/product/triple-axis-accelerometer-gyro-breakout-mpu6050/, Triple Axis Accelerometer and Gyro Breakout – MPU6050.

[6] H.-R. Tränkler und L. M. Reindl, Handbuch für Praxis und Wissenschaft. 2., Berlin Heidelberg: Springer-Verlag, 2014.

[7] G. M. Siouris, „An engineering approach to Optimal Control and Estimation Theory," AFB, Ohio, Wiley-Interscience, 1996, p. 407.

[8] R. Elektronik, „www.reichelt.de," Reichelt Elektronik, [Online]. Available: https://www.reichelt.de/arduino-mega-2560-atmega-2560-usb-arduino-mega-p119696.html.

[9] K. S. Lauszus, „Kalman filteret," 2011.

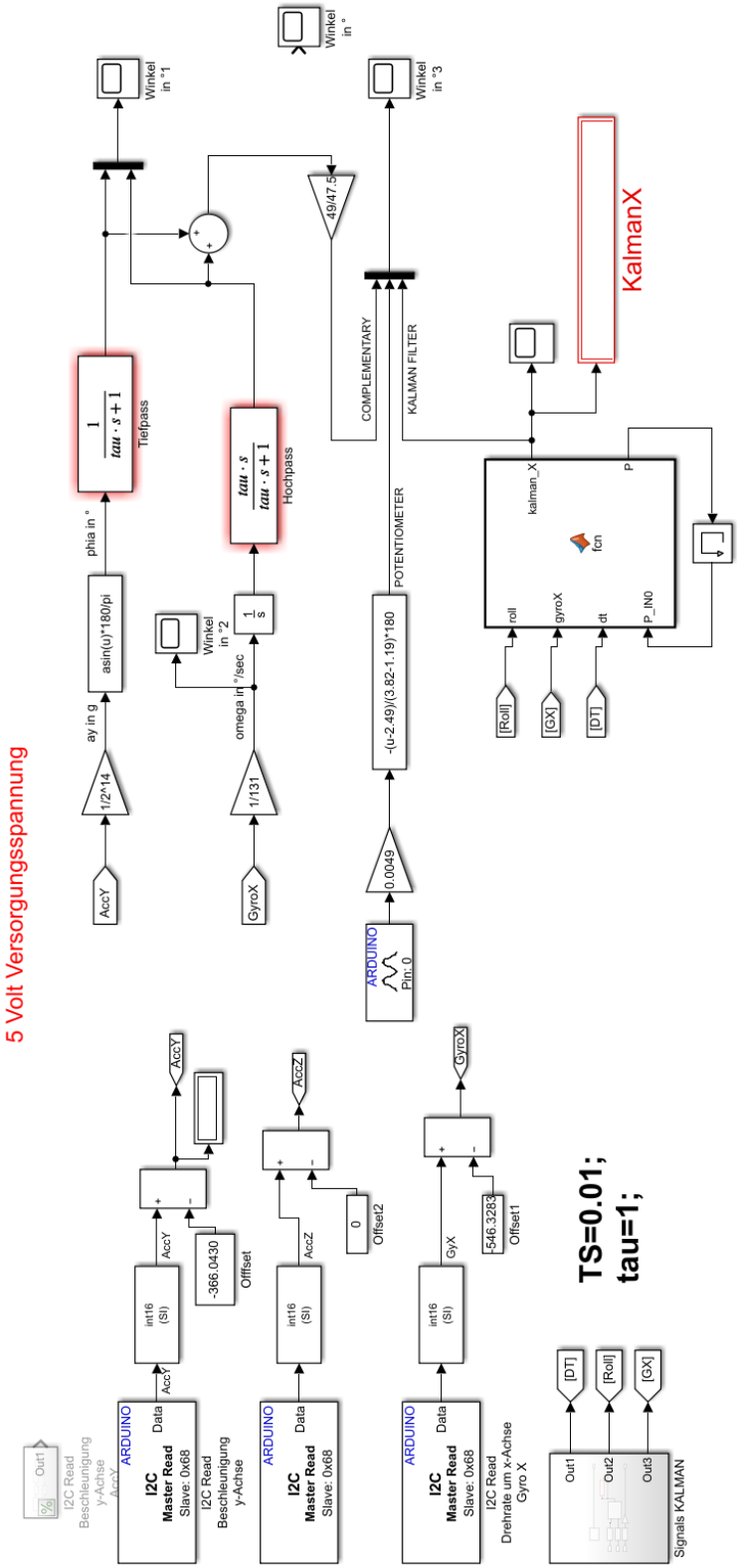
[10] Q. Q. C. L. a. C. H. D. Cao, „Research of Attitude Estimation of UAV Based on Information Fusion of Complementary Filter," Fourth International Conference on Computer Sciences and Convergence Information Technology, pp. 1290-1293., 2009.

[11] A. I. t. t. K. Filter, Greg Welch and Gary Bishop, Chapel Hill, NC: University of North Carolina at Chapel Hill, 2006.

Tabellenverzeichnis

Abbildungsverzeichnis

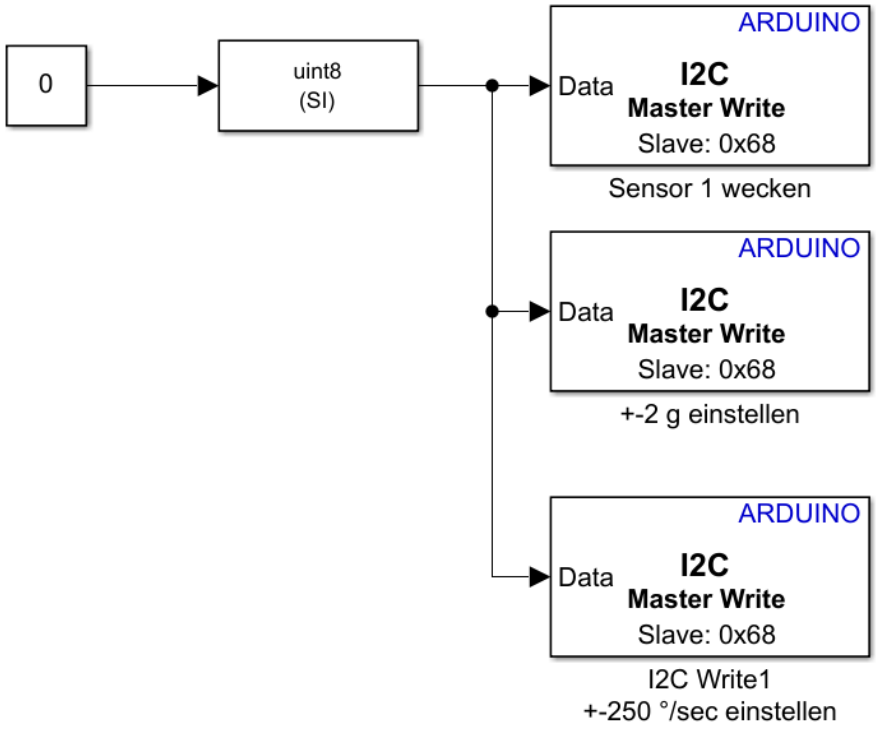
A Anhang


5 Volt Versorgungsspannung
Winkel in °1
Winkel in °
Winkel in °3
49/47.5
COMPLEMENTARY
KALMAN FILTER
KalmanX
Tiefpass
Hochpass
phia in °
asin(u)*180/pi
ay in g
1/2^14
AccY
Winkel in °2
1/s
omega in °/sec
1/131
GyroX
POTENTIOMETER
-(u-2.49)/(3.82-1.19)*180
0.0049
ARDUINO
Pin: 0
kalman_X
fcn
P
roll
gyroX
dt
P_IND
[Roll]
[GX]
[DT]
I2C Read Beschleunigung y-Achse AccY
I2C Master Read
Slave: 0x68
Data
int16 (SI)
AccY
-366.0430
Offset
I2C Read Beschleunigung y-Achse
AccZ
0
Offset2
I2C Read Drehrate um x-Achse Gyro X
GyroX
GyX
546.3283
Offset1
TS=0.01;
tau=1;
Out1
Out2
Out3
Signals KALMAN

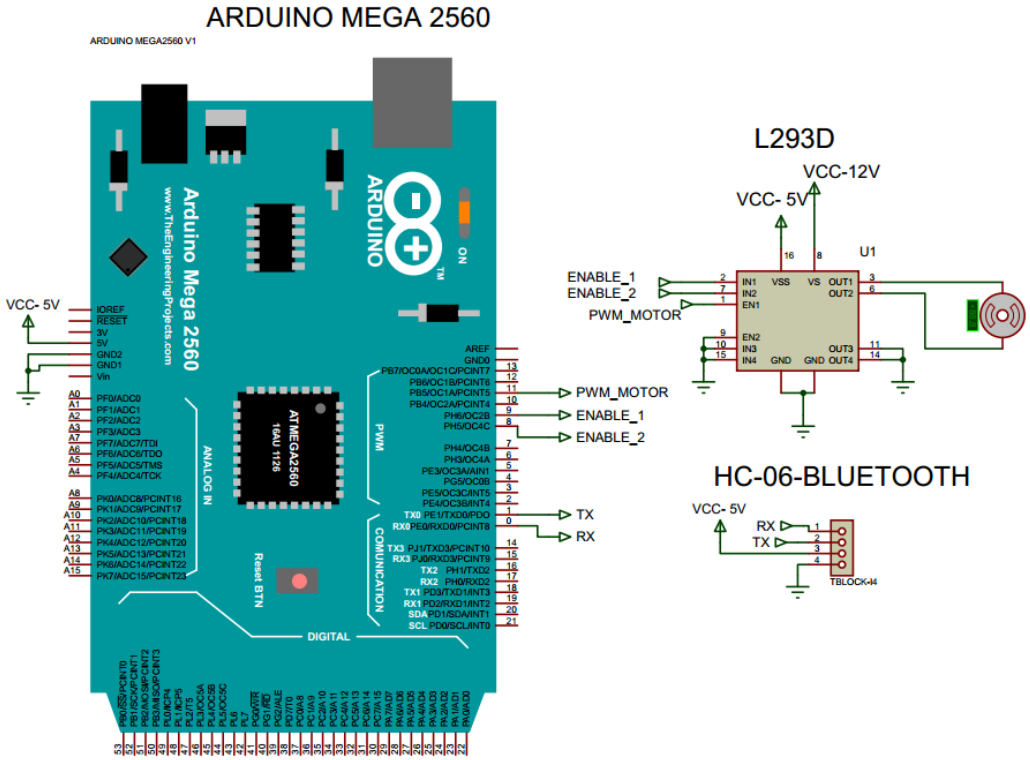
A.1 Kalman Filter Funktionscode

```
function [kalman_X,P] = fcn(roll,gyroX,dt,P_IN0)
%

P_IN=[0 0 0 0 0 0];
P_IN(1:length(P_IN))=P_IN0;

P=[0 0 0 0 0 0];

Q_angle = 0.00007;
 Q_bias = 0.003;
 R_measure = 0.15;


P00=P_IN(1);P01=P_IN(2);
P10=P_IN(3);P11=P_IN(4);
bias = P_IN(5); angle = P_IN(6);

% roll = atan2(accY,accZ)*(180/pi);
rate = gyroX-bias;
angle = angle + dt*rate;
P00 = P00 + dt*(dt*P11 - P01 - P10 + Q_angle);
P01 = P01 - dt*P11;
P10 = P10- dt*P11;
P11 = P11 + Q_bias*dt;
S = P00 + R_measure;
K0 = P00 / S;
K1 = P10 / S;
y = roll-angle;
angle = angle + K0*y;
bias = bias + K1*y;
P00 = P00- K0 * P00;
P01 = P01-K0 * P01;
P10 = P10-K1 * P00;
P11 = P11-K1 * P01;

%if ((roll < -90 && angle > 90) || (roll > 90 && angle < -90))
%angle = roll;
%end
kalman_X = angle; P=[P00,P01,P10,P11,bias,angle];
```

A.2 Sensorbetriebseinstellungen



A.3 Schematische Darstellung zur Steuerung der Drehrichtung eines Motors



A.4 App-Web zur Kommunikation mit dem Bluetooth-Modul(App-Inventor)