Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Masterthesis

Tobias Michaelsen

Lagebestimmung durch Sensorfusion mittels Kalmanfilter

Technik und Informatik
Studiendepartment Informatik

Faculty of Engineering and Computer Science
Department of Computer Science

Tobias Michaelsen

Lagebestimmung durch Sensorfusion mittels Kalmanfilter

Masterthesis eingereicht im Rahmen der Masterprüfung

im Studiengang Automatisierungstechnik
am Department Informations- und Elektrotechnik
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Betreuender Prüfer: Prof. Dr. Andreas Suhl
Zweitgutachter: Prof. Dr. Jochen Maaß

Eingereicht am: 27. Juni 2018

Tobias Michaelsen

Thema der Arbeit

Lagebestimmung durch Sensorfusion mittels Kalmanfilter

Stichworte

Sensorfusion, Kalman Filter, MEMS Sensor, IMU, Inertiale Navigation, Koordinatensysteme, Python

Kurzzusammenfassung

In dieser Arbeit geht es um eine Fusionierung der Daten von Beschleunigungssensoren, Gyroskopen und den Daten eines Magnetometers mittels Kalman Filter. Dies dient der Lagebestimmung eines Bootes.
Mittels eines dynamischen Prüfstands wird der Sensor kalibriert und die Ergebnisse auf ihre Richtigkeit überprüft.

Tobias Michaelsen

Title of the paper

Location determination by sensor fusion using a Kalman filter

Keywords

Sensor fusion, Kalman Filter, MEMs sensor, IMU, inertial navigation, coordinate systems, Python

Abstract

This thesis deals with a fusion of data from acceleration sensors, gyroscopes and data from magnetometer using Kalman filter. This serves the orientation of a boat.
By means of a dynamic test bench, the sensor is calibrated and the results are checked for correctness.

Inhaltsverzeichnis

Abbildungsverzeichnis vi

Tabellenverzeichnis viii

1 Einführung 1

2 Grundlagen 3
2.1 Inertiale Naviation . . . 3
2.1.1 Inertialsensoreinheit . . . 3
2.1.2 Koordinatensysteme . . . 4
2.2 Nick-, Roll- und Gierbestimmung mittels Eulerwinkel . . . 5
2.2.1 Winkelbestimmung durch die Gyroskope . . . 6
2.2.2 Winkelbestimmung mit Hilfe der Beschleunigungssensoren . . . 7
2.2.3 Gierwinkel Bestimmung mit Magnetometer . . . 8

3 MEMS Sensoren 10
3.1 Beschleunigungssensoren . . . 12
3.2 Gyroskop . . . 15
3.3 Magnetometer . . . 16
3.3.1 Sensoren mit Hall-Effekt . . . 16
3.3.2 Sensoren mit anisotropem magnetoresistivem Effekt . . . 17
3.4 Fehlereigenschaften . . . 18
3.4.1 Fehlereigenschaften Gyroskop . . . 20
3.4.2 Fehlereigenschaften Beschleunigungssensor . . . 20
3.4.3 Fehlereigenschaften Magnetometer . . . 21

4 Hardware 22
4.1 Rechnereinheit . . . 22
4.2 Sensorpaket LSM9DS0 . . . 24
4.3 Kommunikation . . . 27
4.4 Spannungswandler . . . 28
4.5 Hardwareaufbau . . . 29

5 Kalibrierung 30
5.1 Verlässlichkeit der Sensoren . . . 30
5.2 Kalibrierung der Beschleunigungssensoren . . . 35

5.3 Kalibrierung der Gyroskope . . . 39
5.3.1 Anpassungen zur Benutzung des Drehtellers . . . 39
5.3.2 Kalibrierung . . . 42

6 Kalman Filter **46**
6.1 Algorithmus des linearen Kalman Filters . . . 48
6.2 Extended Kalman Filter (EKF) . . . 50
6.2.1 Fusionierung . . . 51
6.2.2 Systemmodelle A und H . . . 52
6.2.3 Kovarianzen Q und R . . . 54
6.2.4 Fehler-Kovarianz . . . 55

7 Implementierung **56**
7.1 Betriebsmodi . . . 59
7.2 Auslesen, Umrechnen und Kalibrieren der Sensormesswerte . . . 61
7.3 Berechnen des Roll-, Nick- und Gierwinkels . . . 63
7.4 Funktionen des extended Kalman Filters . . . 64

8 Evaluation **67**
8.1 Testreihe: Nickwinkel θ . . . 68
8.2 Testreihe: Rollwinkel ϕ . . . 72
8.3 Testreihe: Gierwinkel ψ . . . 74

9 Fazit und Ausblick **77**
9.1 Fazit . . . 77
9.2 Ausblick . . . 78

Literaturverzeichnis **79**

Anhang **81**

Abbildungsverzeichnis

2.1 Darstellung der unterschiedlichen Koordinatensysteme (Wendel (2011), S. 29). 5
2.2 Berechnung des Gierwinkels mittels Magnetometer (Sensitec Körber Solutions (2010), S. 6) . . . 8

3.1 Sensorpaket BerryIMU . . . 12
3.2 Grundprinzip eines Beschleunigungssensors (Tränkler; Reindl (2014), S. 545) . 13
3.3 Aufbau eines kapazitiven Beschleunigungssensors (Tränkler; Reindl (2014), S. 555) . . . 14
3.4 Schaltskizze der Signalgenerierung beim kapazitiven Beschleunigungssensor (Tränkler; Reindl (2014), S. 556) . . . 14
3.5 Prinzipskizze des Gyroskops (vgl. Wendel (2011), S. 62) . . . 15
3.6 Magnetfeld messen mit Hilfe des Hall-Effekts (vgl. Fraden (2010), S. 103) . . . 16
3.7 Grungprinzip der Widerstandveränderung (vgl. Sensitec Körber Solutions (2016), S. 6) . . . 17
3.8 zeigt die verschiedenen Fehler Anteile (Kittel (2005),S. 8) . . . 18

4.1 Rechnereinheit Raspberry PI Model B Revision 2.0 . . . 22
4.2 Blockdiagramm des LSM9DS0 . . . 25
4.3 Übersicht vom I^2C Bus . . . 27
4.4 Spannungswandler MP2307 . . . 28
4.5 Hardwareaufbau des Prototypen . . . 29

5.1 Messwerte der unkalibrierten Beschleunigungssensoren . . . 31
5.2 Messwerte der unkalibrierten Gyroskope . . . 32
5.3 Winkeldrift der unkalibrierten Gyroskope . . . 33
5.4 Messwerte des Magnetometers bei 154° . . . 34
5.5 Versuchsaufbau für die Kalibrierung der Beschleunigungssensoren . . . 36
5.6 Beschleunigungswerte X-Achse über Nickwinkel θ . . . 37
5.7 Beschleunigungswerte Y-Achse über Rollwinkel ϕ . . . 37
5.8 Beschleunigungswerte Z-Achse über Nickwinkel ϕ . . . 38
5.9 Montageaufbau des Drehtellers zum Kalibrieren der Gyroskope . . . 40
5.10 Grafische Oberfläche zur Bedienung des Drehtellers . . . 41
5.11 Messwerte bei einer Drehrate von +245°$/s$ . . . 43
5.12 Messwerte bei einer Drehrate von -245°$/s$ . . . 44
5.13 Stillstandmessung der Gyroskope nach Kalibrierung . . . 45
5.14 Winkeldrift der kalibrierten Gyroskope . . . 45

6.1 Algorithmus des linearen Kalman Filters (LKF) (Kim (2011), S. 49) 48
6.2 Algorithmus des extended Kalman Filters (EKF) (vgl. Kim (2011), S. 146) 50
6.3 Schematische Darstellung der Sensorfusion mittels Kalman Filter 51

7.1 Schema Programmablauf 1/2 57
7.2 Schema Programmablauf 2/2 58
7.3 Einstellen der Betriebsmodi der Gyroskope 59
7.4 Einstellen der Betriebsmodi der Beschleunigungssensoren 59
7.5 Einstellen der Betriebsmodi des Magnetometers 60
7.6 Auslesen der Rohdaten der Sensoren (Beispiel Gyroskope) 61
7.7 Umrechnen der Rohdaten der Sensoren 62
7.8 Kalibrierung der Beschleunigungssensoren und der Gyroskope 62
7.9 Ermitteln des Gierwinkels mit Tilt-Kompensation 63
7.10 Roll-und Nickwinkelbestimmung durch Beschleunigungssensoren 63
7.11 Extended Kalman Filter Funktion 64
7.12 Berechnung der Jacobi-Matrix 65
7.13 Berechnung der Jacobi-Matrix 66

8.1 Versuchsaufbau zur Simulation von Wellenbewegungen 67
8.2 Simulation des Nickwinkels θ bei $\pm 30°$ bei 0,75Hz 68
8.3 Ausschnitt der Simulation des Nickwinkels θ bei $\pm 30°$ bei 0,75Hz 69
8.4 Simulation des Nickwinkels θ bei $\pm 30°$ bei 1Hz mit falschen Parametern für Q und R 70
8.5 Simulation des Nickwinkels θ bei $\pm 20°$ bei 0,25Hz mit einem Rollwinkel von 10° 71
8.6 Simulation des Rollwinkels ϕ bei $\pm 30°$ bei 0,75Hz 72
8.7 Ausschnitt der Simulation des Rollwinkels ϕ bei $\pm 30°$ bei 0,75Hz 73
8.8 Schematischer Testaufbau zur Messung des Gierwinkels 74
8.9 Verhalten des Magnetometers bei hoher Dynamik. 76

Tabellenverzeichnis

4.1 Spezifikationen des LSM9DS0 26

5.1 Mittelwert, Varianz und Standardabweichung Beschleunigungssensoren . . . 32
5.2 Mittelwert, Varianz und Standardabweichung Gyroskope 33
5.3 Mittelwert, Varianz und Standardabweichung Magnetometer 34
5.4 Bias und Skalierungsfaktoren der Beschleunigungssensoren 35
5.5 Skalierungsfaktoren der Gyroskope 43
5.6 Mittelwert der Gyroskope nach der Kalibrierung 44

8.1 Testreihe des Gierwinkels 75

1 Einführung

Diese Arbeit wurde im Rahmen des Masterstudiengangs Informations- und Elektrotechnik Vertiefungsrichtung Automatisierung an der Hochschule für Angewandte Wisschenschaften Hamburg erstellt.
Die Idee zu dieser Masterthesis entstand durch das Hobby des Authors, dem Schleppangeln. Durch das Einholen beziehungsweise Rauslassen von sogenannten Sideplanern wird das Boot abgelenkt. Sideplaner ermöglichen das Angeln in einem gewissen Abstand von der Seite des Bootes. Des Weiteren hat die Gewichtsverteilung im Boot einen großen Einfluss auf den Kurs. Bei einer größeren Gewichtsverteilung auf der linken Seite wird das Boot auf eine Kurvenfahrt nach links abgelenkt. Deshalb ist es nötig immer wieder Korrekturen vorzunehmen, um das Boot auf Kurs zu halten.
Die Idee ist es diese Kurskorrekturen automatisch durchführen zu lassen, sodass nicht immer eine Person auf den Kurs des Bootes achten muss. Um dies umzusetzen ist es nötig den Rollwinkel, Nickwinkel und den Gierwinkel des Bootes zu ermitteln. In dieser Masterarbeit geht es um diese Ermittlung der Lagebestimmung des Bootes.
Zur Ermittlung der Winkel wird in dieser Arbeit eine Sensorkombination (inertiale Messeinheit) genutzt. Diese besteht aus einem Gyroskop, einem Beschleunigungssensor und einem Magnetometer (digitaler Kompass).
Es handelt sich hierbei um digitale MEMS-Sensoren. Diese Mikro-elektronischen mechanischen Systeme sind dank ihrer technischen Weiterentwicklung heute die erste Wahl. Da diese Sensoren eine geringe Baugröße aufweisen, können sie in vielen technischen Gerätschaften, wie zum Beispiel Handys, Controllern oder auch Notebooks eingesetzt werden. Der günstige Anschaffungspreis ist ein weiterer, wirtschaftlicher Vorteil dieser Sensoren. Viele Anwendungen wären sonst wirtschaftlich nicht realisierbar.
Ein Nachteil dieser Sensoren ist, dass Sie Messfehler und Probleme haben die ein falsches Messergebnis erzeugen können. Diese Messfehler zu beheben ist ein Teil dieser Arbeit. Um die Fehler zu analysieren und zu beheben, wird ein Dynamischer Prüfstand genutzt, um die Abweichungen durch geeignete Maßnahmen zu beseitigen. Um die Vorteile der Beschleunigungssensoren, der Gyroskope und des Magnetometers zu nutzen um den Roll- Nick- und

Gierwinkel zu bestimmen, werden die Daten der drei Sensoren mit Hilfe des Kalman-Filters fusioniert.
Das Ziel ist es ein kompaktes, mobiles und kostengünstiges System zu entwickeln, welches den Nickwinkel, Rollwinkel und Gierwinkel des Bootes erzeugt.
Die Masterthesis ist folgendermaßen gegliedert: Im Kapitel 2 wird auf die Grundlagen eingegangen, damit auf diese Bezug genommen werden kann. Kapitel 3 beschäftigt sich mit den genutzten MEMS-Sensoren. Das Kapitel 4 gibt einen Überblick über die verwendeten Hardwarekomponenten welche in dieser Arbeit benutzt werden. Des Weiteren wird auf die Kommunikation zwischen dem Sensorpaket und der Rechnereinheit eingegangen. In Kapitel 5 wird auf die Kalibrierung der Sensoren eingegangen. Das Kalman-Filter wird in Kapitel 6 grundsätzlich erklärt und anschließend wird auf den hier verwendeten Extended Kalman Filter eingegangen. Kapitel 7 erläutert die Implementierung der entwickelten Software. Kapitel 8 evaluiert das erstellte System und beinhaltet einige Tests. In Kapitel 9 wird ein Fazit gezogen und ein Ausblick gegeben.

2 Grundlagen

In diesem Kapitel wird auf die später genutzten Grundlagen eingegangen, wie zum Beispiel die unterschiedlichen Koordinatensysteme oder die Berechnung der Winkel mit Hilfe des Sensorpakets.

2.1 Inertiale Naviation

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.

2.1.1 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 (Winkelgeschwindigkeit).
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 Navigationskoordinatensystem durchgeführt. In Verbindung mit einer Recheneinheit werden zum Beispiel die Winkelgeschwindigkeiten erfasst und durch Integration kann die Fahrzeuglage berechnet werden.

2.1.2 Koordinatensysteme

Laut WGS84 (World Deodetic System 1984) werden vier verschiedene Koordinatensysteme definiert. Für eine Strapdown-Rechnung ist es notwendig diese zu kennen, da der Inertialsensor Daten (Beschleunigung und Drehrate) im körperfesten Koordinatensystem aufzeichnet, aber die Lagebestimmung und Navigation im Navigationskoordinatensystem stattfinden. Es wird in dieser Arbeit nicht die Rotationsgeschwindigkeit der Erde betrachtet und die Gravitationskraft wird als Konstante angenommen. Dadurch fallen die Koordinatensysteme des i-frames und des e-frames zusammen. Im folgenden Abschnitt werden die Koordinatensystem definiert (vgl. Wendel (2011), S. 28):

- **Körperfestes Koordinatensystem (b-frame)** Die Achsen dieses Koordinatensystems sind fest mit dem Fahrzeug (Boot) verbunden. Der Ursprung befindet sich im Fahrzeug. Bei genauer Ausrichtung der IMU zum Fahrzeug fallen die Achsen der IMU mit denen des körperfesten Koordinatensystems zusammen.

- **Inertialkoordinatensystem (i-frame)** Der Mittelpunkt dieses Koordinatensystems befindet sich im Mittelpunkt des Rotationsellipsoids, der die Erdgestalt annähert. Die Z-Achse fällt mit der Rotatiotsachse der Erde zusammen. Eine IMU misst Beschleunigungen und Drehraten des körperfesten Koordinatensystems bezüglich des Inertialkoordinatensystems.

- **Erdfestes Koordinatensystem (e-frame)** Das erdfeste Koordninatensystem besitzt den selben Ursprung wie das Inertialkoordinatensystem. Die Z-Achsen fällt mit dem des Inertialkoorinatensystems zusammen. Das Koordinatensystem wird auch als earth centered, earth fixed (ECEF) Koordinatensystem bezeichnet.

- **Navigationskoordinatensystem (n-frame)** 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.

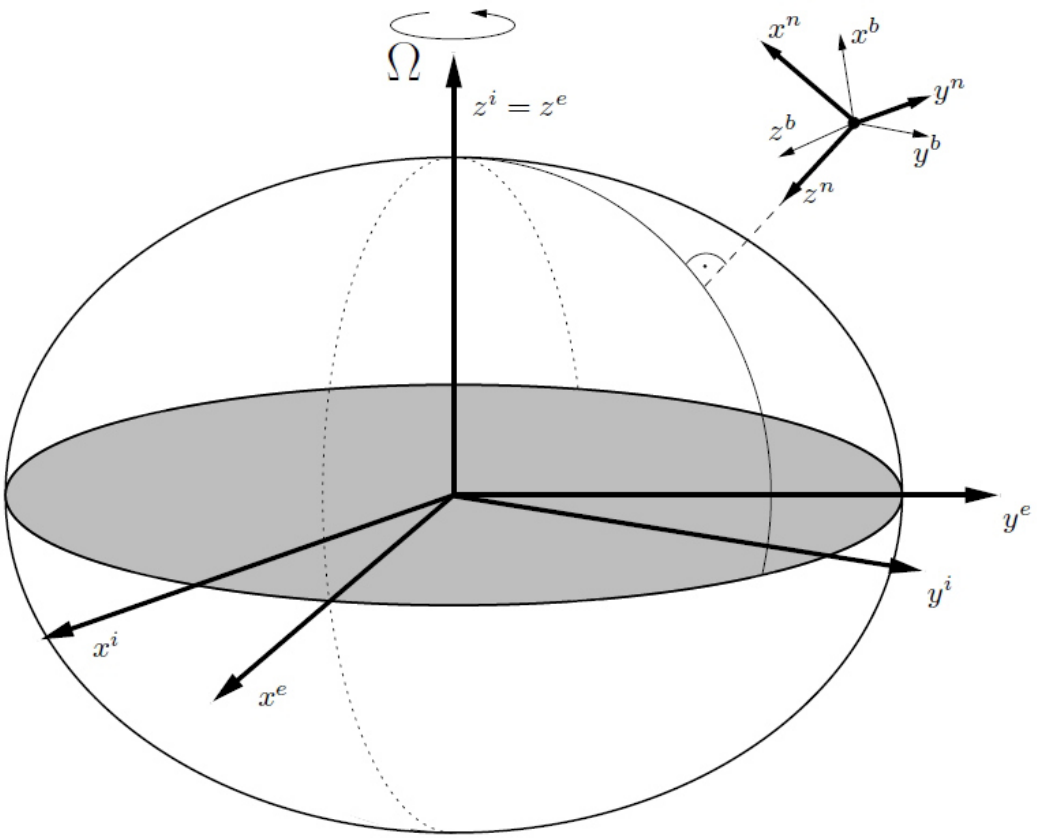

Abbildung 2.1: Darstellung der unterschiedlichen Koordinatensysteme (Wendel (2011), S. 29).

2.2 Nick-, Roll- und Gierbestimmung mittels Eulerwinkel

Um die Nick-, Roll- und Gierwinkel zu bestimmen, ist eine Koordinatentransformation notwendig. In dieser Arbeit wird die Koordinatentransformation durch die Eulerwinkel bestimmt. Die drei Eulerwinkel beschreiben eine Drehung, die nacheinander ausgeführt das Navigationskoordinatensystem in das körperfeste Koordinatensystem überführen. Es gibt mehrere Konventionen, wie die Reihenfolge der Drehungen um die jeweiligen Achsen durchgeführt werden müssen. In dieser Arbeit wird die Konvention (z, y', x") benutzt. Hierbei beschreibt der Gier Winkel ψ die erste Drehung um die z-Achse des Navigantionskoordinatensystems. Anschließend wird die neue y-Achse des rotierenden Koordinatensystems um den Nick Winkel Θ gedreht. Die letzte Drehung erfolgt schließlich um denn Rollwinkel ϕ um die neue x-Achse. (vgl. Wendel (2011), S. 39)

Ein Nachteil der Eulerwinkel ist, dass bei Winkeln von $\pm$ 90 Grad Nickwinkel Singularitäten auftreten. Der Gierwinkel beschreibt die Drehung um die lokale Vertikale. Das Problem ist, dass bei einer Drehung um $\pm$ 90 Grad des Nickwinkels zwei Achsen auf einer Ebene (lokale Vertikale) liegen. Wenn jetzt um den Rollwinkel gedreht wird, ist es ebenfalls eine Drehung um die lokale Vertikale, in diesem Fall sind die Eulerwinkel nicht eindeutig. Dieses Problem nennt sich Gimbal Lock. Da ein Boot nicht auf $\pm$ 90 Grad Nickwinkel kommen wird, verursacht der Gimbal Lock keine Probleme.

2.2.1 Winkelbestimmung durch die Gyroskope

Es ist nicht möglich die drei Drehraten p,q und r der drei Gyroskope zu integrieren um die Eulerwinkel zu erhalten. Dies ist nicht möglich, da die Bezugssysteme nicht übereinstimmen. Es ist notwendig die Drehraten zu transformieren. Durch die Richtungskosinusmatrix C_b^n können die Drehraten in das Bezugssystem tranformiert werden und die Eulerwinkelgeschwindigkeiten $\dot{\Phi}$, $\dot{\theta}$ und $\dot{\psi}$ bestimmt werden. Durch erneutes Integrieren über die Zeit werden die Eulerwinkel berechnet.
Die Richtungskosinusmatrix kann als Funktion der Eulerwinkel angegeben werden. In dieser Arbeit wird nicht auf die Herleitung dieser Transformationsmatrix eingegangen. Dazu wird folgende Literatur empfohlen: Titterton; Weston (2004) und Wendel (2011).

$$C_b^n = \begin{bmatrix} \cos\theta\cos\psi & -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\phi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix} \tag{2.1}$$

Die zeitliche Änderung der Eulerwinkel aufgrund der Drehraten p, q und r wird durch folgende Differentialgleichung dargestellt:

$$\dot{\Phi} = p + \sin\phi * \tan\theta * q + \cos\phi * \tan\theta * r \tag{2.2}$$

$$\dot{\theta} = \cos\phi * q - \sin\phi * r \tag{2.3}$$

$$\dot{\psi} = \sin\phi/\cos\theta * q + \cos\phi/\cos\theta * r \tag{2.4}$$

Ein weiteres Integrieren der Eulerwinkelgeschwindigkeiten ergibt die Eulerwinkel.

$$\Phi = \int \dot{\Phi} dt \tag{2.5}$$

$$\theta = \int \dot{\theta} dt \tag{2.6}$$

$$\Psi = \int \dot{\Psi} dt \tag{2.7}$$

2.2.2 Winkelbestimmung mit Hilfe der Beschleunigungssensoren

Zur Berechnung des Roll- und Nickwinkels mit Hilfe der Beschleunigungssensoren, müssen zunächst die Zustände geklärt werden, in dem sich das Bezugssystem, also das Boot, bewegt. Das Boot wird beim Schleppen mit einer konstanten Geschwindigkeit gefahren. Das bedeutet, dass das Boot keine Beschleunigung erfährt außer der Anziehungskraft. Der Sensor misst also nur die Anziehungskraft der Erde und genau dies kann sich zu Nutzen gemacht werden. Die Beschleunigungen f_x, f_y und f_z ergeben sich laut (Kim (2011) S. 126) wie folgt:

$$\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = g \begin{bmatrix} \sin\theta \\ -\cos\theta\sin\phi \\ -\cos\theta\cos\phi \end{bmatrix} \tag{2.8}$$

Dabei stellt g die Erdanziehungskraft von 9,81m/s dar. Das Ganze ist aber nur gültig, wenn das System in eingeschwungenem Zustand ist, also im Stillstand, oder sich konstant mit einer gewissen Geschwindigkeit bewegt. Nun können wir mit Hilfe von Formel 2.8 den Rollwinkel ϕ und den Nickwinkel θ berechnen. Die Bestimmung des Gierwinkels ist mit dieser Methode nicht möglich, da der Bezugsvektor, wie die Gravitation, fehlt.

$$\phi = \arcsin \frac{-f_y}{g\cos\theta} \tag{2.9}$$

$$\theta = \arcsin \frac{f_x}{g} \tag{2.10}$$

2.2.3 Gierwinkel Bestimmung mit Magnetometer

Mit Hilfe eines Magnetometers wird die Richtung, in die das Boot fährt bestimmt. Hierzu misst das Magnetometer, um genau zu sein sind es wie beim Beschleunigungssensor drei an der Zahl, das Magnetfeld der Erde. Das Magnetfeld der Erde ist zirka 0.5 bis 0.6 Gauss stark und ist immer in Richtung des magnetischen Nordpols ausgerichtet. Da der magnetische Nordpol nicht mit dem geodätischen übereinstimmt, muss der sogenannte Deklinationswinkel bei der Berechnung mit einbezogen werden. Mit der Abbildung 2.2 wird die Berechnung des Gierwinkels, im Englischen auf als Heading bezeichnet, erläutert:

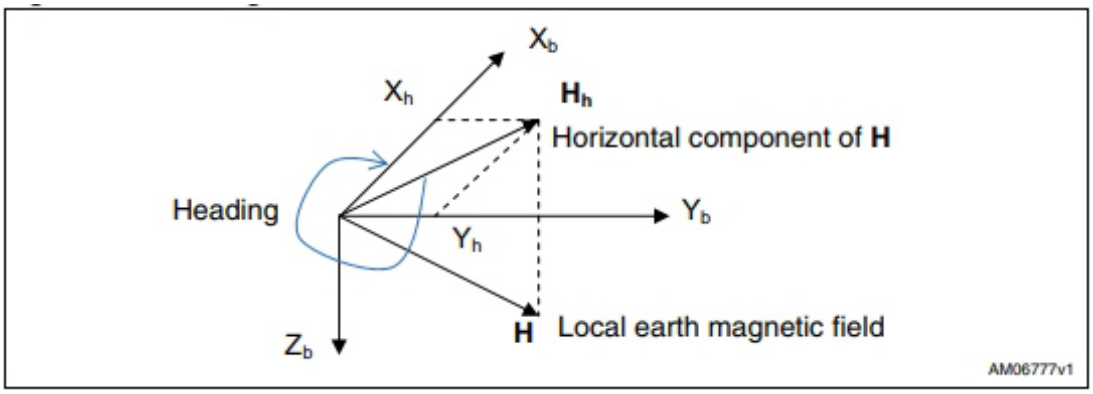

Abbildung 2.2: Berechnung des Gierwinkels mittels Magnetometer (Sensitec Körber Solutions (2010), S. 6)

Das Erdmagnetfeld hat eine fixe Komponente H_h, welche horizontal zum magnetischen Nordpol zeigt. Die Komponenten des Vektors H_h in X und Y Richtung bezogen auf das körperfeste Koordinatensystem sind hier mit X_h und Y_h benannt und können mit dem Magnetometern gemessen werden. Dies funktioniert aber nur wenn der Sensor horizontal zur Erdoberfläche steht. Mit folgender Formel laut (Sensitec Körber Solutions (2010), S. 6) kann der Gierwinkel berechnet werden:

$$\psi = \arctan(Y_h/X_h) + D \tag{2.11}$$

D ist der Deklinationswinkel, welcher für Hamburg +2,5° beträgt.
Wenn die Achsen X_h und H_h parallel zueinander stehen, also das Boot Richtung Norden zeigt, wird ein Gierwinkel von 0° angezeigt, da $Y_h = 0$ ist.
Da das Boot durch Wellen oder Gewichtsverteilung immer einen gewissen Rollwinkel oder Nickwinkel hat, ist die obere Formel nicht anwendbar. Da in dieser Arbeit der Nickwinkel und

der Rollwinkel bekannt sind, lässt sich dies mit folgenden Formeln kompensieren (Sensitec Körber Solutions (2010). S. 7).

$$X_h = X_M \cos\theta + Z_M \sin\theta \tag{2.12}$$

$$Y_h = X_M \sin\phi \sin\theta + Y_M \cos\phi - Z_M \sin\phi \cos\theta \tag{2.13}$$

Dabei sind X_M,Y_M und Z_M hier die Werte, die das Magnetometer misst. Mit den neuen Werten für X_h und Y_h wird nun der Tilt kompensierte Gierwinkel wieder mit der Formel 2.11 berechnet.

3 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 der selben Platine befinden, lassen sich die Messgrößen direkt auslesen, zum Beispiel über einen I^2C Bus. Einige Messeigenschaften von MEMS Sensoren sind:

- **Beschleunigung**
- **Drehrate**
- **Druck**
- **Feuchtigkeit**
- **Temperatur**
- **Magnetismus**
- **Sound**

Des Weiteren sind sie kompakt und können dadurch in zum Beispiel Handys oder kleinen Sensoreinheiten verbaut werden, die in vielen Anwendungen Gebrauch finden. Durch die Produktion großer Stückzahlen sind diese Sensoren kostengünstig in der Anschaffung, was viele Projekte erst realisierbar macht. Der Weltmarkt für MEMS Inertialsensoren, welche aus räumlichen Kombinationen von Beschleunigungssensoren und Drehratensensoren bestehen wird laut Tränkler; Reindl (2014), S. 524 von circa 850 Millionen Dollar in 2009 auf 2,56 Milliarden Dollar in 2015 steigen.

Beispiele für Anwendungen von MEMS Intertialsensoren sind laut (Tränkler; Reindl (2014), S. 542):

- **Flug und Raumfahrt**
  Messen von kleinen Bewegungen im All
  Lagebestimmung von Flugzeugen
  Fliegen von autonomen Drohen
- **Automobilindustrie**
  Auslösen von Airbageinheiten
  Steuerung von aktiven Stoßdämpfern
- **Maschinenbau**
  Messen von Neigungswinkeln
  Maschinensteurerung
- **Seismologie**
  Bodenbewegungen und Erdstöße
- **Medizin**
  Steuerung von Herzschrittmachern

In dieser Arbeit sind im Vorfeld einige Spezifikationen an den Sensoren gestellt worden. Der Sensor muss kostengünstig sein, sowie Beschleunigung, Drehrate und Magnetimus messen können. Das Auslesen der Daten muss in Echtzeit geschehen können. Hin zu kommt ein günstiger Anschaffungspreis, damit das Projekt realisierbar bleibt.

Die Auswahl viel hier auf die BerryIMU. Auf der BerryIMU befinden sich zwei integrierte Schaltkreise, einmal der LSM9DS0 von STMicroelectronics und der BMP180 von Bosch. Der LSM9DS0 beinhaltet Beschleunigungssensoren um die translatorische Bewegung zu messen, Drehratensensoren um die Winkelgeschwindigkeit zu messen und Magnetometer um das Magnetfeld der Erde zu messen. Beschleunigung, Drehrate und das Magnetfeld werden jeweils auf 3-Achsen gemessen. Des Weiteren beinhaltet der LSM9DS0 Sensor einen I^2C Bus, was das Auslesen durch eine Steuereinheit ermöglicht. Der BMP180 von Bosch beinhaltet einen Drucksensor, sowie einen Temperatursensor. In dieser Arbeit wird nur der LSM9DS0 von STMicroelektronics benutzt.

Abbildung 3.1: Sensorpaket BerryIMU

3.1 Beschleunigungssensoren

Für die Erklärung der Funktionsweise eines Beschleunigungssensors empfehle der Autor Tränkler; Reindl (2014). Beschleunigungssensoren messen die durch eine Beschleunigung ausgeübte Kraft F auf eine Masse m. Das zweite Newton'sche Axiom beschreibt den Zusammenhang wie folgt:

$$F = m * a \tag{3.1}$$

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

$$a(t) = \frac{dv(t)}{dt} = \frac{d^2z(t)}{dt^2} \tag{3.2}$$

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 LSM9DS0 gibt die Beschleunigung in der Einheit g aus. Dabei ist g die Schwerebeschleunigung der Erde, 1g entspricht 9,81 $\frac{m}{s^2}$.

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.

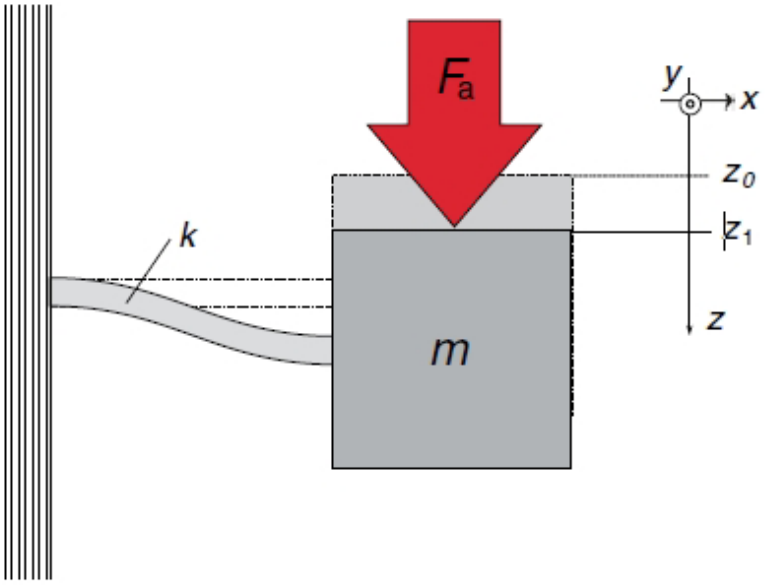

Abbildung 3.2: Grundprinzip eines Beschleunigungssensors (Tränkler; Reindl (2014), S. 545)

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. Dazu gehören 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.

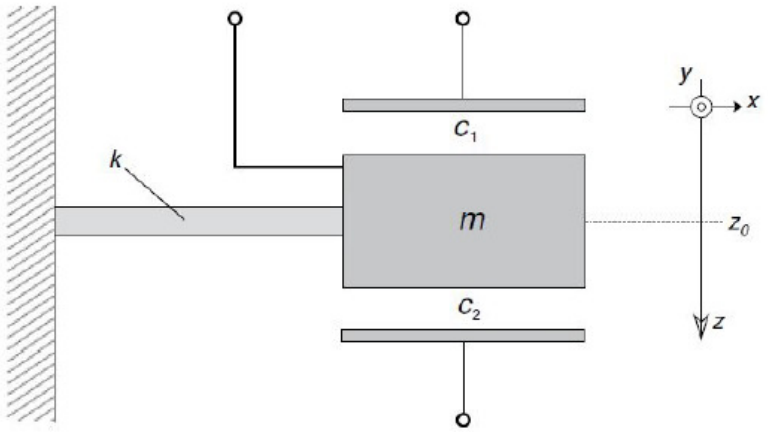

Abbildung 3.3: Aufbau eines kapazitiven Beschleunigungssensors (Tränkler; Reindl (2014), S. 555)

Wie in Abbildung 3.3 zu sehen, bildet die elastisch aufgehängte Masse m die Elektrode der beiden Kondensatoren C_1 und C_2. Die seismische Masse m bildet die mittlere Platte eines Differentialkondensators. Je nach Beschleunigungsrichtung rückt die Mittelplatte entweder näher an die Platte des ersten Kondensators und entfernt sich von der des zweiten oder umgekehrt. Über die in Abbildung 3.4 gezeigte Messbrücke kann nun die Messspannung U_m einen Rückschluss auf die Beschleunigung geben.

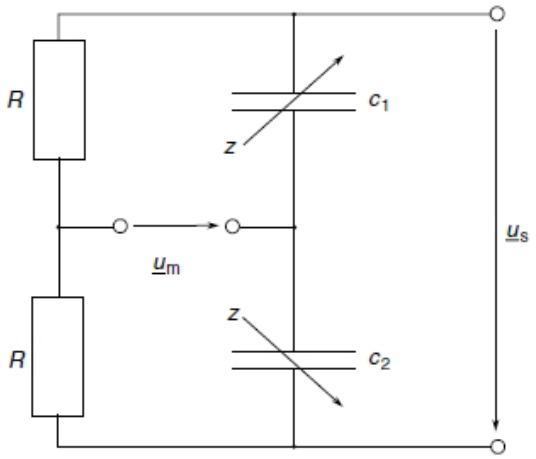

Abbildung 3.4: Schaltskizze der Signalgenerierung beim kapazitiven Beschleunigungssensor (Tränkler; Reindl (2014), S. 556)

3.2 Gyroskop

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. Beim Sagnac-Effekt kann die Drehgeschwindigkeit gemessen werden, indem Lichtstrahlen über Spiegel oder Glasfasern im Kreis geführt werden und je nach Drehrate einen längeren oder kürzeren Weg durchlaufen. Über die Laufzeit kann nun ein Rückschluss auf die Drehrate gezogen werden.
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 Sensors 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. Detaillierte Betrachtungen sind in Tränkler; Reindl (2014) und Wendel (2011) zu finden.
Mit der Relativgeschwindigkeit v_{ref}, der Corioliskraft F_c und der Drehrate Ω ergibt sich die Beschleunigung a_c wie folgt:

$$a_c = \frac{F_c}{m} = 2v_{ref} * \Omega \tag{3.3}$$

Durch Umstellen der Formel kann nun die Drehrate Ω berechnet werden.

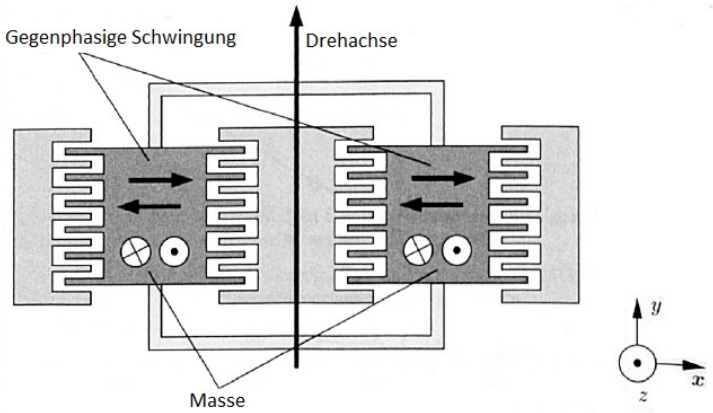

Abbildung 3.5: Prinzipskizze des Gyroskops (vgl. Wendel (2011), S. 62)

3.3 Magnetometer

Ein Magnetometer kann ein Magnetfeld, wie zum Beispiel das der Erde messen. Gemessen wird das Magnetfeld in der Einheit Gauss. Das Erdmagnetfeld hat in Deutschland eine ungefähre Feldstärke von 0,48 Gauss. Durch eine Anordnung von drei orthogonal angeordneten Sensoren kann ein digitaler Kompass erzeugt werden, welcher den Gierwinkel im erdfesten Koordinatensystem erzeugt. Es gibt zwei verschiedene Arten das Magnetfeld zu messen. Einmal mit Hilfe des Hall-Effekts und einmal mit des von William Thomas entdeckten anisotropen magnetoresistiven Effekts. Aus dem Datenblatt des hier verwendeten Sensors ist nicht herauszulesen, welches Prinzip dieser Sensor benutzt, deshalb wird in dieser Arbeit kurz auf beide eingegangen.

3.3.1 Sensoren mit Hall-Effekt

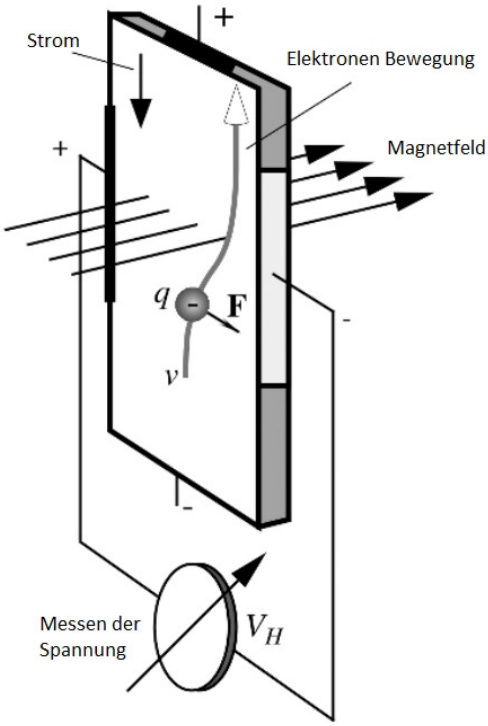

Abbildung 3.6: Magnetfeld messen mit Hilfe des Hall-Effekts (vgl. Fraden (2010), S. 103)

Bei diesem Sensor wird ein elektrischer Leiter an eine Spannungsquelle gelegt, sodass ein Strom fließt. Die Ladungsträger sind Elektronen, welche sich mit einer Geschwindigkeit v durch den Leiter bewegen.

Durch die vom Magnetfeld verursachten Lorentz-Kraft F werden die Elektronen q senkrecht zu einer Bewegungsrichtung abgelenkt. Dadurch entsteht auf der einen Seite ein Elektronenüberschuss und auf der anderen Seite ein Elektronenmangel. Durch Anschließen eines Voltmeters kann nur eine Spannung gemessen werden, die Rückschluss auf die Stärke und Richtung des Magnetfelds geben kann. Genauere Informationen zum Prinzip sind bei (Fraden (2010), S. 103) nachzulesen.

3.3.2 Sensoren mit anisotropem magnetoresistivem Effekt

Die Messung des Magnetfeldes erfolgt durch Sensoren, die mit Hilfe des 1857 durch Thomson entdeckten anisotropen magnetoresistiven Effektes die Flussrichtung des Magnetfeldes messen können. Hier wird die Widerstandsveränderung einer ferromagnetischen Metalllegierung in Abhängigkeit eines magnetischen Feldes gemessen. Die Ausrichtung des Sensors im Bezug auf das Magnetfeld bestimmt den Widerstands eines Permalloy Elements. Bei Permalloy handelt es sich um eine Legierung die zu, 81% aus Nickel und zu 19% Eisen besteht.

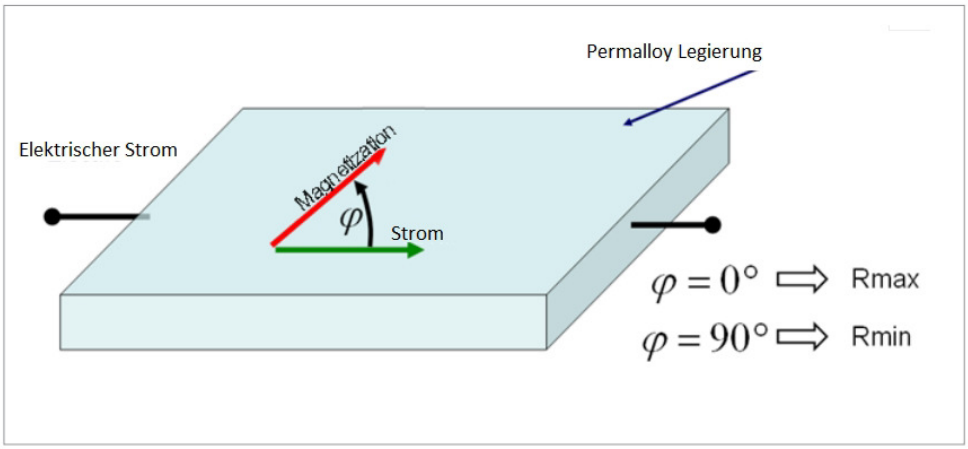

Abbildung 3.7: Grungprinzip der Widerstandveränderung (vgl. Sensitec Körber Solutions (2016), S. 6)

Wenn das Magnetfeld und Stromrichtung parallel zueinander liegen $\varphi = 0°$ (gleiche Richtung oder auch entgegengesetzt) ist der Widerstand der Permalloy Legierung am größten.

3.4 Fehlereigenschaften

MEMS Sensoren sind wie alle anderen Sensoren nicht ideal. Dies bedeutet, dass die gemessenen Werte der MEMS Sensoren nicht den realen Werten entsprechen. Die Differenz zwischen dem Istwert y_i und dem Sollwert y_s nennt sich laut (Tränkler; Reindl (2014),S. 7) absoluter Fehler.

$$F_{abs} = y_i - y_s \tag{3.4}$$

Die folgende Skizze zeigt, wie der gesamte Fehler sich in verschiedene Anteile untergliedert:

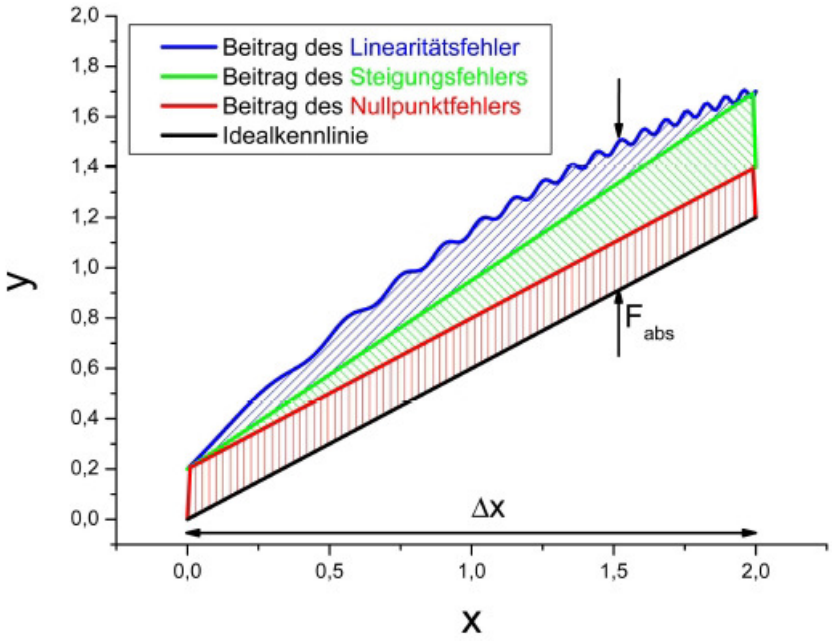

Abbildung 3.8: zeigt die verschiedenen Fehler Anteile (Kittel (2005),S. 8)

Wie in Abbildung 3.8 ergeben sich folgende Gleichungen für die Ist-Kennlinie y_i, Soll-Kennlinie y_s und den absoluten FehlerF_{abs}:

$$y_i(x) = y_{0i} + \frac{\Delta y_i}{\Delta x}(x - x_0) + F_{li}(x) \tag{3.5}$$

$$y_s(x) = y_{0s} + \frac{\Delta y_s}{\Delta x}(x - x_0) \tag{3.6}$$

$$F_{abs} = \underbrace{y_{0i} - y_{0s}}_{F_{abs}} + \underbrace{\frac{\Delta y_i - \Delta y_s}{\Delta x}(x - x_0)}_{F_{st}(x)} + F_{li}(x) \tag{3.7}$$

Nachfolgend werden die einzelnen Fehler genau erklärt, diese Fehler sind temperaturabhängig. Die Kalibrierung des verwendeten Sensors fand im Labor bei einer Temperatur von 21°C statt.

Bias (Nullpunktfehler)

Gibt die Abweichung des Ausgangssignals von Null an, bei einem Eingangssignal von Null. Hierbei wird zwischen einem Konstanten Bias und einem Bias-Drift unterschieden. Der konstante Bias kann sich zwar von Einschalten zu Einschalten verändern, bleibt aber konstant über die Zeit. Der Bias-Drift kann über die Zeit gesehen durch Temperaturschwankungen entstehen.

Steigungsfehler

Hierbei handelt es sich um eine Abweichung von den vom Hersteller angegebenen Skalenfaktoren. Dies führt zu einer falschen Steigung der Geraden. Dieser Fehler kann durch Toleranzen in der Herstellung entstehen oder durch Alterung.

Linearitätsfehler

Der Linearitätsfehler ist die Abweichung der Ist-Kennlinie von einer gewünschten linearen Soll-Kennlinie.

Rauschen

Unter Rauschen versteht man statistische Fluktuationen, die das Ausgangssignal überlagern können. Diese einstehen durch die Brownsche Molekularbewegung oder beim Stromtransport von Elektronen.

3.4.1 Fehlereigenschaften Gyroskop

Das Messen von Drehraten ist immer mit Fehlern behaftet. Das folgende Fehlermodell laut (Wendel (2011),S. 68) zeigt die einzelnen Fehler auf:

$$\tilde{\vec{\omega}}^{b}_{ib} = M_{Gyro} * \vec{\omega}^{b}_{ib} + \vec{b}_{\omega} + \vec{n}_{\omega} \tag{3.8}$$

Dabei ist $\tilde{\vec{\omega}}^{b}_{ib}$ die gemessene Drehrate, $\vec{\omega}^{b}_{ib}$ ist die reale Drehrate. Im idealen Fall liegen die 3 Achsen genau orthogonal aufeinander. Da dies in der Realität nicht der Fall ist, gibt die Misalignment-Matrix M_{Gyro} diesen Fehler an. Das Misaligment wird nach dem Zusammenbau der IMU vermessen und kompensiert. Der restliche Fehler wird in der Misalignment-Matrix angegeben. Auf diese wird in dieser Arbeit nicht eingegangen, da die auftretenden Fehler viel kleiner sind als die anderer Terme und keinen großen Einfluss auf die Genauigkeit haben. Der Term $\vec{b}_{\omega}$ gibt den Bias des Sensors an. Term $\vec{n}_{\omega}$ beschreibt das sensorinherente Rauschen, was hier als weiß, normalverteilt und mittelwertfrei angenommen wird.

3.4.2 Fehlereigenschaften Beschleunigungssensor

Auch die Messung der Beschleunigung ist mit Fehlern behaftet. Hier treten die gleichen Fehler wie beim Drehratensensor auf, so dass die Fehlerformel laut (Wendel (2011), S. 72) wie folgt aussieht:

$$\tilde{\vec{f}}^{b}_{ib} = M_{Acc} * \vec{f}^{b}_{ib} + \vec{b}_{a} + \vec{n}_{a} \tag{3.9}$$

Dabei ist $\tilde{\vec{f}}^{b}_{ib}$ die gemessene Beschleunigung, $\vec{\omega}^{b}_{ib}$ ist die reale Beschleunigung. M_{Acc} gibt hier wieder die Misalignment-Matrix an und wird wie beim Gyroskop vernachlässigt. Beim Bias Term des Beschleunigungsmessers $\vec{b}_{a}$ können als rectification errors bezeichnete Anteile auftauchen, die durch Vibrationen verursacht werden. Dies ist vor allem dann ein Problem, wenn ein Datenfusionsalgorythmus, wie in dieser Arbeit das Kalman Filter, verwendet wird. Der Bias wird im Labor bestimmt, kann aber außerhalb des Labors abweichen, was ein anpassen der Fehler-Kovarianzmatrix Q nötig machen kann. $\vec{n}_{a}$ gibt wieder das Sensor Rauschen an.

3.4.3 Fehlereigenschaften Magnetometer

Magnetometer weisen Fehler wie ein Beschleunigungssensor oder ein Gyroskop auf, jedoch sind die magnetischen Abweichungen, welche den wahren Messwert überlagern, weit höher. Die wichtigsten dieser Störfelder sind laut Phidgets (2017) Überlagerungen des geomagnetischen Feldes durch Hard- und Soft-Iron Effekte.
Der sogenannte **Hard-Iron Störeinfluss** entsteht durch statische Magnetfelder, die vom Fahrzeug selbst ausgehen. Dies können ferromagnetische Metalle oder Magneten sein. Diese Störungen sind abhängig vom Fahrzeug, an dem der Sensor montiert ist. Solange der Sensor nicht umgebaut wird, bleiben die Störungen des Fahrzeugs konstant. Auf den Sensorachsen zeigt sich dieser Effekt als konstanter positiver oder negativer Offset.
Der **Soft-Iron Störeinfluss** entsteht durch nichtmagnetische Metalle, die aber das Erdmagnetfeld stören können. Sie verzerren das Magnetfeld der Erde und können so zu Abweichungen führen.
Ein weiteres Problem können **dynamische Magnetfelder** sein, welche auf dem Fahrzeug selbst erzeugt werden. Die Magnetfelder können durch Elektrogeräte erzeugt werden. Zu den größten Störfaktoren gehören hier die Elektromotoren. Diese Störungen verändern sich kontinuierlich und lassen sich so schwer bis gar nicht kalibrieren.

In dieser Arbeit für diesen speziellen Fall werden die oben genannten Störungen nicht betrachtet, da es sich um ein offenes Konsolenboot aus Glasfaserverstärktem Kunststoff handelt. Des Weiteren wird der Sensor separat in ein Kunststoffgehäuse montiert.

4 Hardware

In diesem Kapitel wird auf die verwendete Hardware eingegangen, welche zur Umsetzung dieser Arbeit benutzt wird. Dabei handelt es sich um die Recheneinheit, die Sensorplatine und einen Spannungswandler. Es wird auf technische Daten und ihre Funktionen eingegangen.

4.1 Rechnereinheit

Bei der Rechnereinheit handelt es sich um einen Rasperry PI Model B Revision 2.0. Diese besteht aus einem Einplatinencomputer, welcher eine kleine Bauform aufweist (in etwa so groß wie eine Kreditkarte). Die Rechnereinheit besitzt ein Ein-Chip-System mit einem ARM-Mikroprozessor von Broadcom.

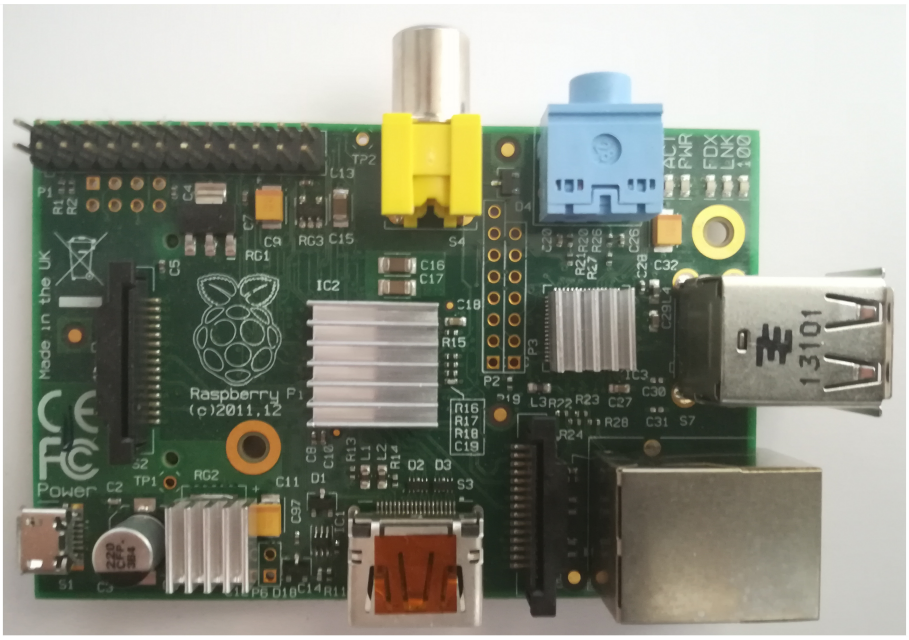

Abbildung 4.1: Rechnereinheit Raspberry PI Model B Revision 2.0

Der Raspberry Pi wurde entwickelt, um den Menschen einen kostengünstigen und einfachen Einstieg in Programmier- und Hardwarekenntnisse zu ermöglichen. Bis Ende 2017 wurden mehr als 17 Millionen Geräte verkauft (Raspberry Pi Foundation, 2018).
In dieser Arbeit wird der Raspberry PI als Zentralrechner verwendet. Er dient zum Auslesen und Bearbeiten der Sensordaten des Sensorpakets und zum Berechnen von Algorithmen (Kalman Filter). Der Rasbperry PI ist in dieser Arbeit gewählt worden, da die Community sehr groß ist und so bei Problemen helfen können. Ein weiterer Grund ist der günstige Preis von unter 30 Euro und dessen Größe von nur 93mm x 63,5mm x 20mm.

Die nachfolgende Aufzählung zeigt die Leistungdaten des Raspberry PI Model B Revision 2.0:

- System on Chip (SoC) BCM2835 von Broadcom mit einem CPU (ARM11), GPU, DSP und SDRAM
- CPU ist ein ARM11 in der ARMv6 Architerture mit 700Mhz. Aufgetaktet auf 900Mhz, dank Kühlkörper
- GPU (Graphics Processing Unit) ist ein Broadcom Viedocore IV mit OpenGL ES 2.0. Ermöglicht eine Auflösung von 1080p
- Einen Arbeitsspeicher von 512MB
- Einen Netzwerkanschluss 10/100Mbits Ethernet RJ45
- 2 x USB 2.0 Anschlüsse
- 26 Pin Anschlüssen. Diese beinhalten zum Beispiel die I^2C und SPI Busse und Spannungsversorgungen (3,3V,5V, GND)
- HDMI-Anschluss vom Typ A HDMI 1.3a
- Einen SD-Kartenslot (max. 32GB) (Hinweis: Es ist darauf zu achten, dass die SD-Karte eine möglichst hohe Lese- und Schreibgeschwindigkeit hat, da dann das System flüssiger läuft)
- Die Leistungsaufnahme beträgt ca. 3,5W bei 5V Spannungsversorgung
- Das Betriebssystem ist Linux Raspbian

4.2 Sensorpaket LSM9DS0

Der Sensor, der auf dem BerryIMU verwendet wird ist der LSM9DS0. Dieser kann, wie schon in Kapitel 3 beschrieben, Beschleunigung, Drehrate und Magnetismus auf 3 orthogonal zueinander angeordneten Achsen messen in, X-, Y- und Z-Richtung.
Der Sensor lässt sich auf folgende Messbereiche einstellen:

- **Beschleunigung**
  ±2g, ±4g, ±6g, ±8g, ±16g
- **Gyroskop**
  ±245dps, ±500dps, ±2000dps
- **Magnetometer**
  ±2gauss, ±4gauss, ±8gauss, ±12gauss

Des Weiteren besitzt der LSM9DS0 ein I^2C Interface, über das es Daten zum Raspberry PI übertragen kann. Es kann bei Temperaturen von -40°C bis +85°C eingesetzt werden.
Genauere Spezifikation sind dem Datenblatt des LSM9DS0 von STMicroelectronis zu entnehmen, welches sich auf der beigelegten CD befindet.

Die Funktion des Sensors lässt sich durch eine Blockdiagramm, wie in Abbildung 4.2 zu sehen ist, am besten erklären. Der Sensor misst in X-, Y- und Z-Richtung die Sensordaten. Dabei handelt es sich um ein analoges Signal. Über einen Multiplexer werden die Rohdaten nun selektiert und über einen Verstärker an ein Analog-Digital-Wandler übertragen. Dieser digitalisiert das Signal, sodass dies über das I^2C Interface an den Raspberry PI oder eine andere Rechnereinheit gesendet werden kann.

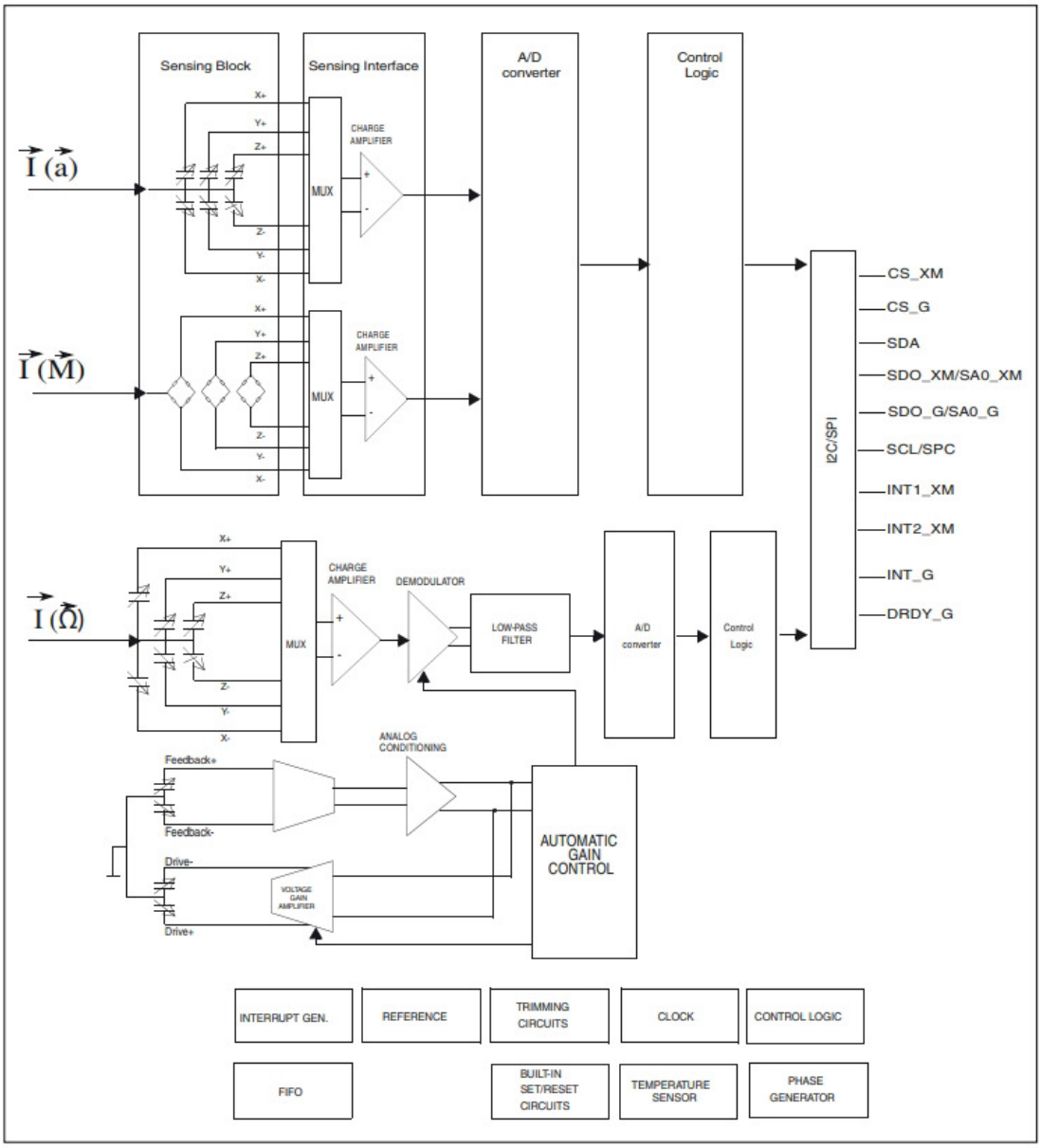

Abbildung 4.2: Blockdiagramm des LSM9DS0

Der folgenden Tabelle sind **Sensorspezifikationen** des LSM9DS0 zu entnehmen:

Parameter	Testbedingungen	Umrechnung	Einheit
Empfindlichkeit Beschleunigungssensor	±2g ±4g ±6g ±8g ±16g	0,061 0,122 0,183 0,244 0,732	mg/LSB
Empfindlichkeit Gyroskop	± 245dps ± 500dps ± 2000dps	8,75 17,5 70	mdps/digit
Empfindlichkeit Magnetometer	± 2gauss ± 4gauss ± 8gauss ± 12gauss	0,08 0,16 0,32 0,48	mgauss/LSB
Empfindlichkeitsveränderung durch Temperatur: Beschleunigunssensor	-40°C bis +85°C	±1,5	%
Empfindlichkeitsveränderung durch Temperatur: Gyroskop	-40°C bis +85°C	±2	%
Empfindlichkeitsveränderung durch Temperatur: Magnetometer	-40°C bis +85°C	±3	%
Null-g Offset Genauigkeit Beschleunigungssensor		±60	mg
Null level Genauigkeit Gyroskop	245dps 500dps 2000dps	±10 ±15 ±25	dps
Null-g Offset Genauigkeit Beschleunigungssensor	Max Delta von 25°C	±0,5	mg/°C
Änderung des Null levels aufgrund von Temperatur: Gyroskop		±0,05	dps/°C

Tabelle 4.1: Spezifikationen des LSM9DS0

4.3 Kommunikation

Die Kommunikation zwischen dem Raspberry PI und dem BerryIMU findet über einen I^2C Bus statt. I^2C steht für Inter-Integrated Circuit (interne integrierte Schaltung). Über diesen Bus können die Daten des Sensors übertragen werden. Bei dem I^2C Bus handelt es sich um einen Master-Slave-Bus. Dabei ist der Raspberry PI der Master und gibt die Taktung auf dem Bus an. Es ist eine Kommunikation von max. 400 kbits/s möglich, da der Raspberry Pi nur den Fast-Mode unterstützt. Der I^2C Bus ist im Ultra-Fast-Mode in der Lage eine Datenübertragung von bis zu 5,0 MBits/s zu ermöglichen.
Für die I^2C Kommunikation werden 2 Leitungen benötigt. Zum einen die Datenleitung SDA, worüber die Datenübertragung abläuft und zum anderen die Taktleitung SCL. Der I^2C Bus arbeitet mit einem Adressbereich von 7Bit, daher ist es möglich 112 Adressen zu vergeben (16 werden für Sonderaufgaben reserviert). Des Weiteren wird am BerryIMU noch eine Spannungsversorgung benötigt, diese besteht aus einer Leitung für die 3,3V Spannungsversorgung und einer Leitung für die Masse.

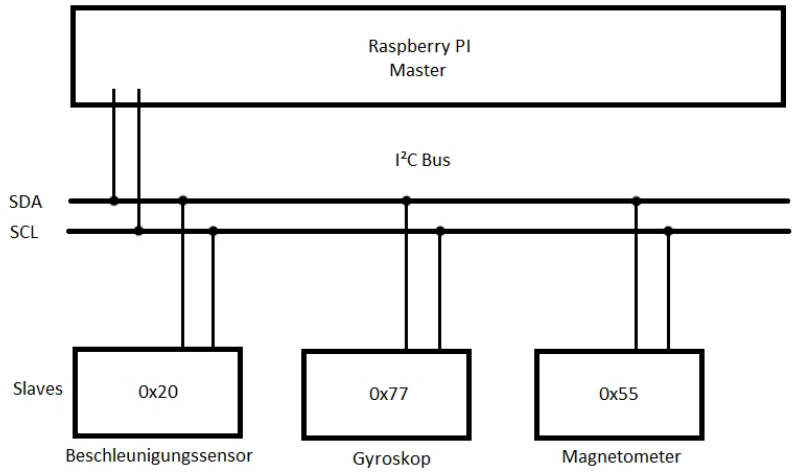

Abbildung 4.3: Übersicht vom I^2C Bus

Genauere Information zur Funktionsweise vom I^2C Bus ist in (Kofler; Kühnast; Scherbeck (2017), S.476) nachzulesen.

4.4 Spannungswandler

Um den Raspberry PI an Bord des Bootes nutzen zu können, wird ein Spannungswandler benötigt. Dieser Spannungswandler versorgt den Raspberry PI mit der benötigten Versorgungsspannung von 5V bei einem 12V Bordspannungsnetz. Der Spannungswandler basiert auf den MP2307 Step down Konverter von MPS (Monolithic Power Systems). Ein Step down Konverter taktet die Eingangsspannung über ein Widerstandsverhältnis auf die gewünschte Ausgangsspannung herunter. Eine genauere Funktionsweise ist dem Datenblatt des MP2307 zu entnehmen, dieses befindet sich auf der zur Arbeit zugehörigen CD.

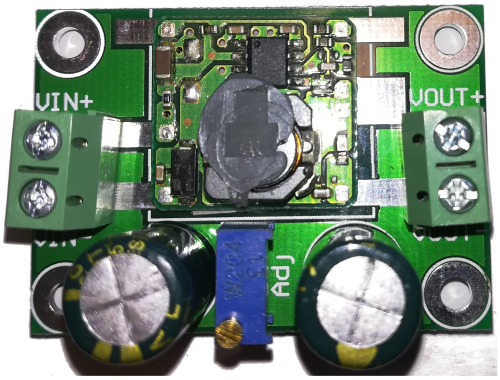

Abbildung 4.4: Spannungswandler MP2307

Der Spannungswandler besitzt folgende Spezifikationen:

- Eingangsspannungsbereich von 4,75V bis 23V
- Ausgangsspannungsbereich von 0,975 bis 20V
- 3A Ausgangsstrom kontinuierlich (der Raspberry PI benötigt nur max. 0,7 Ampere)
- Ausgangsspannung einstellbar über ein Potentiometer
- Kondensator zur Spannungsglättung
- 95% Effizienz

4.5 Hardwareaufbau

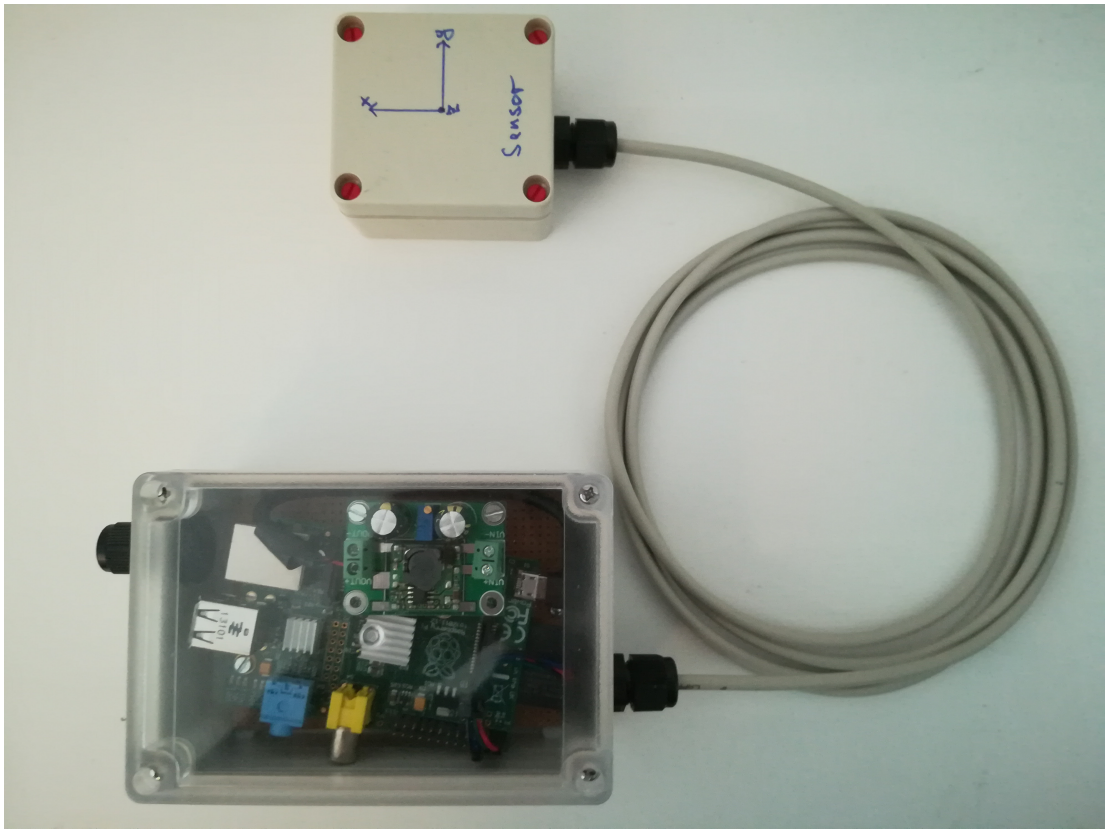

Abbildung 4.5: Hardwareaufbau des Prototypen

Die Abbildung 4.5 zeigt den Aufbau des Hardware-Prototypen. Der Raspberry Pi und der Spannungswandler wurden in ein Gehäuse gebaut und der Sensor in ein separates Gehäuse. Die separate Montage ist wichtig, um Störungen des gemessenen Magnetfeldes zu verhindern. Beide Gehäuse haben die Schutzklasse IP65, was bedeutet, dass sie Staubdicht und gegen Strahlwasser geschützt sind. Dies ist wichtig für den Betrieb auf einem Boot. Über sogenannte PG-Verschraubung kann die Spannungsversorgung, sowie die Sensorleitung in das Gehäuse geführt werden, ohne die Schutzklasse zu minimieren.

5 Kalibrierung

In diesem Kapitel geht es darum die MEMS Sensoren zu kalibrieren. Hierfür wurden im Vorfeld verschiedene Messungen durchgeführt, um die Verlässlichkeit der Sensoren zu bestimmen. Anschließend wird auf die Kalibrierung der Beschleunigungssensoren und der Gyroskope eingegangen.

5.1 Verlässlichkeit der Sensoren

Die Verlässlichkeit der Sensoren ist ein wichtiges Thema. Hierbei geht es darum, wie Zuverlässig die gemessenen Werte der Sensoren sind. Die Standardabweichung S und der arithmetische Mittelwert $\overline{x}_{arithm}$ sind hierfür die Indikatoren. Die Standardabweichung gibt die Streuung der Messwerte um den Erwartungswert (Mittelwert) an. Laut Tränkler; Reindl (2014) sind die Standardabweichung und der arithmetische Mittelwert ein Maß für die Verlässlichkeit der Messergebnisse. Die Messreihen bestehen aus 600 Messwerten, um genug Werte für ein Bestimmung des arithmetischen Mittelwerts und der Standardabweichung zu haben. Mit den folgenden Formeln lassen sich die Standardabweichung S und der arithmetische Mittelwert $\overline{x}_{arithm}$ berechnen:

$$S_x = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2} \tag{5.1}$$

$$\overline{x}_{arithm} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + x_3 + ... + x_n}{?} \tag{5.2}$$

In dieser Arbeit wird die Varianz benötigt, um die Fehler-Kovarianzmatrix zu berechnen, welche beim Kalman Filter erforderlich ist. Die Varianz ist das Quadrat der Standardabweichung und lässt sich wie folgt berechnen:

$$Var(x) = s_x^2 \tag{5.3}$$

Um die Standardabweichung und die Varianz zu ermitteln, wird die Beschleunigung in X-, Y- und Z-Richtung gemessen. Der Sensor befindet sich dabei in Ruhelage. Die Z-Achse zeigt dabei zum Mittelpunkt der Erde. Zu erwarten ist, dass die Beschleunigung in Z-Richtung $9,81m/s^2$ beträgt (Erdanziehungskraft) und in X- und Y-Richtung $0m/s^2$ gemessen werden.

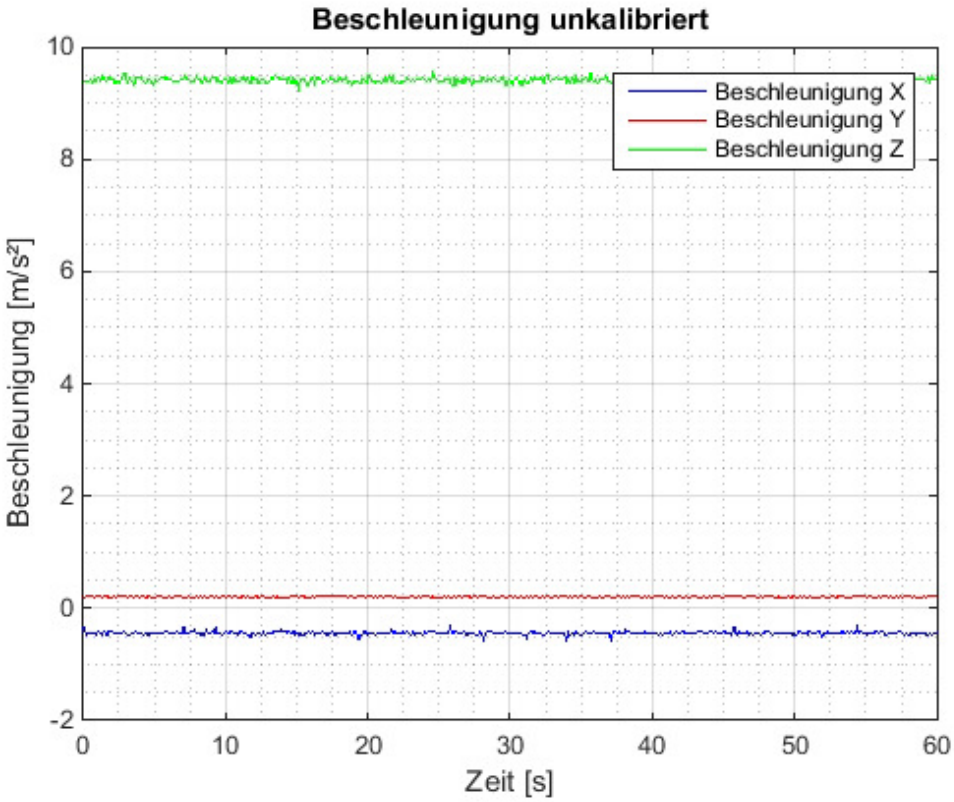

Abbildung 5.1: Messwerte der unkalibrierten Beschleunigungssensoren

Die Abbildung 5.1 zeigt, dass nicht die erwarteten Beschleunigungen gemessen wurden. Es ist klar zu sehen, dass die einzelnen Achsen jeweils einen Offset (Nullpunktfehler) haben. Des Weiteren zeigt sich, dass die Beschleunigungssensoren ein nicht so hohes Rauschen wie die Gyroskope (Siehe Abbildung 5.2) aufweisen.
In der folgenden Tabelle sind die Werte der Varianz, der arithmetischer Mittelwert und die Standardabweichung der Beschleunigungssensoren abzulesen:

Achse	Mittelwert $[m/s^2]$	Varianz $[m^2/s^4]$	Standardabweichung $[m/s^2]$
X	-0,440	0,9*10^{-3}	0,030
Y	0,204	0,144*10^{-3}	0,012
Z	9,406	1,9*10^{-3}	0,044

Tabelle 5.1: Mittelwert, Varianz und Standardabweichung Beschleunigungssensoren

Die Abbildung 5.2 zeigt die Messung der Winkelgeschwindigkeiten der Gyroskope aller drei Achsen. Die Gyroskope befinden sich dabei im Ruhezustand. Es ist zu sehen, dass auch die Gyroskope ein Offset besitzen. Des Weiteren weisen diese ein höheres Rauschen auf als die Beschleunigungssensoren.

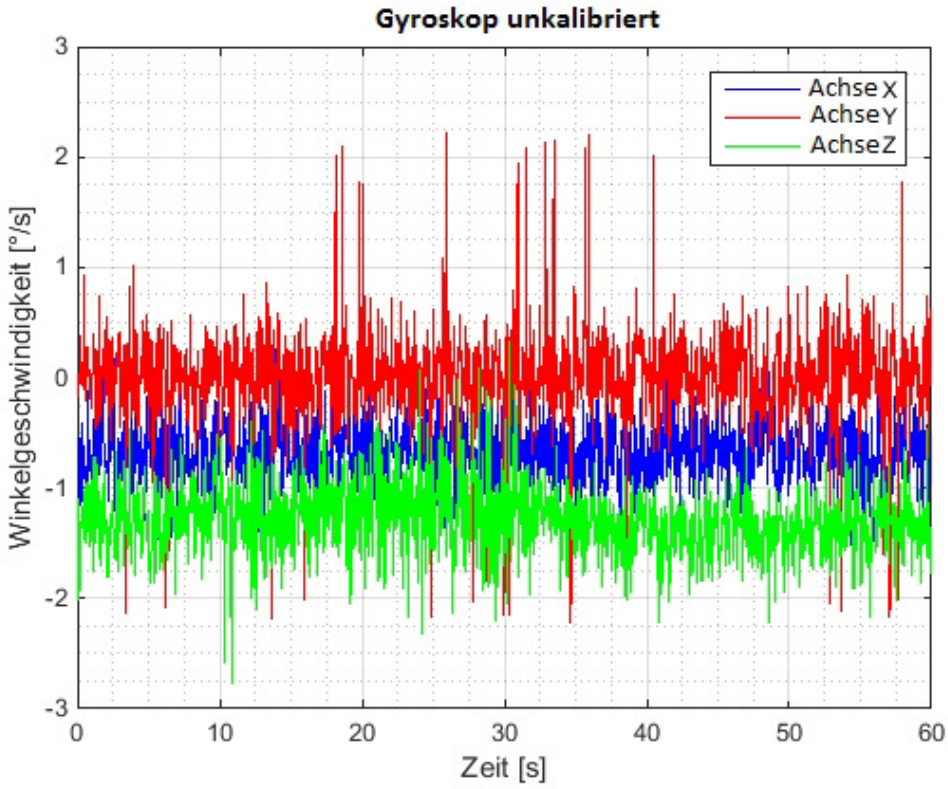

Abbildung 5.2: Messwerte der unkalibrierten Gyroskope

Die gemessenen Werte für den arithmetischen Mittelwert, Varianz und die Standardabweichung zeigt die folgende Tabelle:

Achse	Mittelwert $[°/s]$	Varianz $[°^2/s^2]$	Standardabweichung $[°/s]$
X	-0,668	0,089	0,298
Y	0,021	0,182	0,427
Z	-1,281	0,108	0,329

Tabelle 5.2: Mittelwert, Varianz und Standardabweichung Gyroskope

Um den Winkel zu berechnen, muss die Winkelgeschwindigkeit über die Zeit integriert werden. Im zeitdiskreten Bereich geschieht dies über die Abtastzeit der Winkelgeschwindigkeit. Die Abbildung 5.3 zeigt den Winkeldrift über einen Zeitraum von 180 Sekunden. Durch die Integration des Offsets der Winkelgeschwindigkeit entsteht ein Winkeldrift.

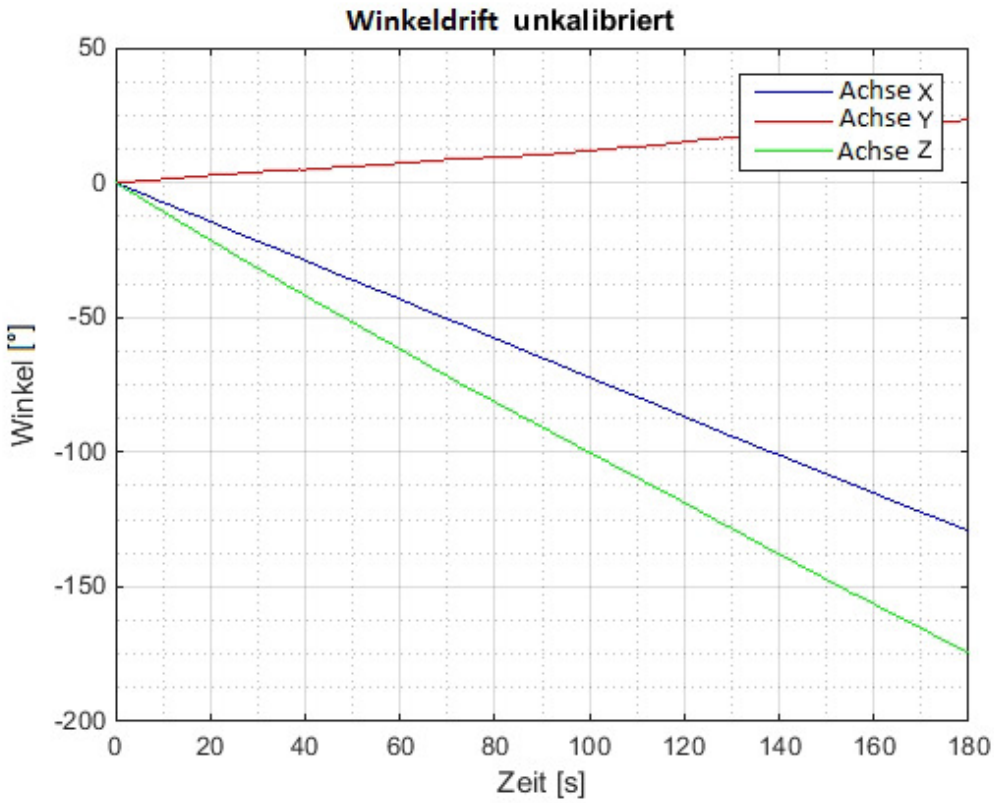

Abbildung 5.3: Winkeldrift der unkalibrierten Gyroskope

Wie in den Abbildungen der unkalibrierten Beschleunigungssensoren und Gyroskope zu sehen, ist es notwendig diese zu kalibrieren um bessere Messwerte zu bekommen. Durch die folgende Kalibrierung lassen sich die konstanten Fehler wie Offsets sowie der Steigungsfehler entfernen. Der Bias-Drift und das Sensorrauschen lassen sich durch diese Kalibrierung nicht vermeiden. Hierfür wird im Kapitel 6 das Kalman Filter besprochen. Dieser versucht die stochastischen Anteile des Messsignals zu minimieren.

Die Abbildung 5.4 zeigt die Messwerte der X-, Y- und Z-Achse bei einem gewählten Gierwinkel von 154°. Der Sensor liegt dabei gerade, sodass auf die Z-Achse die volle Erdbeschleunigung wirkt. Mit dieser Messung können die Varianz und die Standardabweichung für jede Achse bestimmt werden, welche für die Implementierung des Kalman Filters wichtig sind (siehe Tabelle 5.4).

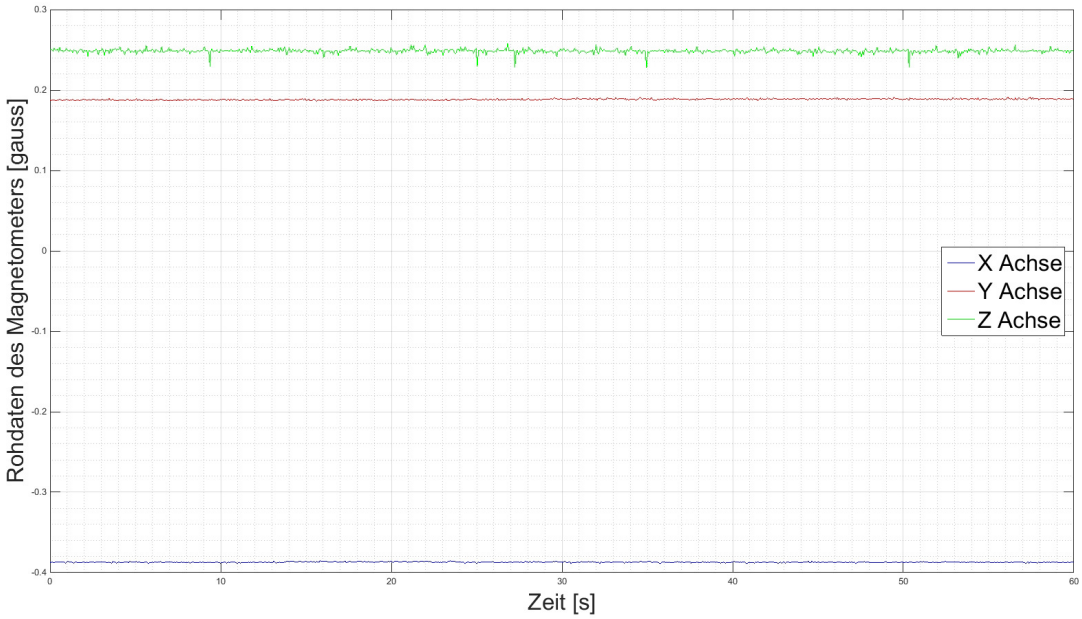

Abbildung 5.4: Messwerte des Magnetometers bei 154°

Achse	Mittelwert [Gauss]	Varianz [$Guass^2$]	Standardabweichung [Gauss]
X	-0,387	3,969*10^{-9}	0,063 *10^{-3}
Y	0,188	0,736*10^{-6}	0,858*10^{-3}
Z	0,248	7*10^{-6}	0,0026

Tabelle 5.3: Mittelwert, Varianz und Standardabweichung Magnetometer

5.2 Kalibrierung der Beschleunigungssensoren

Die in dieser Arbeit benutzte Methode zur Kalibrierung nennt sich Rotationskalibrierung. Diese wird in (Krohn; Beigl; Decker; et al. (2004)) vorgestellt, um Beschleunigungssensor zu kalibrieren. Durch diese Methode lassen sich der Nullpunktfehler (Bias) und der Steigungsfehler jeder Achse korrigieren.
Bei dieser Methode wird jede Achse separat in Richtung der Erdanziehungskraft orientiert, sodass eine Beschleunigung von 1g (Erdbeschleunigung) auf die Achse einwirkt. Nun wird die selbe Achse gemessen, aber der Sensor um 180° gedreht, sodass nun im Idealfall -1g gemessen wird. Durch folgende Formeln lässt sich der Korrigierungsfaktor für den Bias o_x und den Skalierungsfaktor s_x am Beispiel der X-Achse berechnen:

$$o_x = \frac{x_{max} + x_{min}}{2} \tag{5.4}$$

$$s_x = \frac{x_{max} - x_{min}}{2} \tag{5.5}$$

Durch den Skalierungsfaktor wird der Steigungsfehler der Kennlinie der einzelnen Achsen behoben. Durch die folgende Formel lässt sich nun die Kennlinie der Beschleunigungssensoren korrigieren (Beispiel X Achse):

$$ax_{korrigiert} = \frac{ax_{gemessen} - o_x}{s_x} \tag{5.6}$$

Folgenden Tabelle zeigt die gemessenen Werte für den Bias und den Skalierungsfaktor jeder Achse:

Achse	Bias	Skalierungsfaktor
X	-0,073	0,997
Y	0,033	1,002
Z	-0,021	0,996

Tabelle 5.4: Bias und Skalierungsfaktoren der Beschleunigungssensoren

Um die Kalibrierung der Beschleunigungssensoren zu überprüfen, wird ein Versuchsaufbau benutzt, welcher es über eine Gewindestange ermöglicht den Sensor stufenlos in verschiedene Winkel einzustellen. Damit lässt sich die Beschleunigungskennlinie der Erdanziehung abfahren, von einem Winkel von $0°$ bis $90°$. Durch Umdrehen des Sensors wird die Kennlinie von $0°$ bis $-90°$ abgefahren.

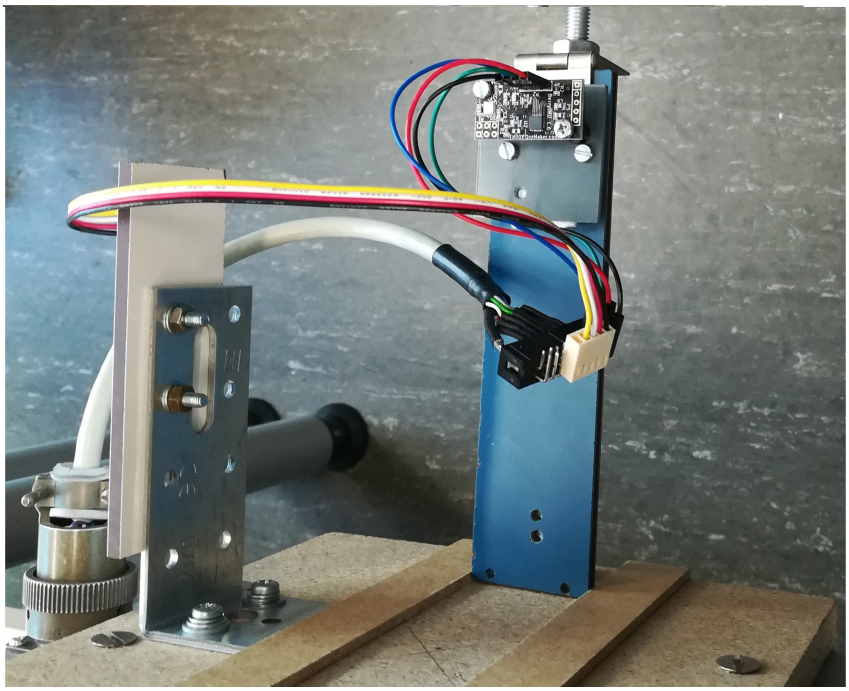

Abbildung 5.5: Versuchsaufbau für die Kalibrierung der Beschleunigungssensoren

Der Winkel wird dabei von $0°$ bis $40°$ um jeweils $5°$ erhöht und ab $40°$ jeweils um $10°$-Schritte. Die kleinere Schrittwahl am Anfang wird gewählt, um eine höhere Genauigkeit im unteren Bereich zu erreichen. Es ist davon auszugehen, dass ein Boot nicht über $40°$ Roll- oder $30°$ Nickwinkel kommt.
Der Winkel wird mit Hilfe eines digitalen Winkelmessers gemessen, dieser besitzt eine Genauigkeit von $0.1°$.
Für jeden gemessenen Beschleunigungswert werden 100 Messwerte aufgenommen und deren arithmetischer Mittelwert bestimmt.
Die Überprüfung der kalibrierten Beschleunigungen der X-Achse und der Z-Achse wird dabei über die Verstellung des Nickwinkels θ überprüft.
Die kalibrierte Beschleunigung der Y-Achse wird über den Rollwinkel ϕ überprüft.

Die kalibrierten und nicht kalibrierten Werte der einzelnen Achsen werden durch Matlab eingelesen und visualisiert. Zum Vergleich wird eine ideale Kennlinie hinzugefügt. Die kalibrierten Werten sollten so gut wie möglich mit der idealen Kennlinie übereinander stimmen. Die Abbildungen 5.6 bis 5.8 zeigen die Überprüfung der einzelnen Achsen in Abhängigkeit des jeweiligen Winkels:

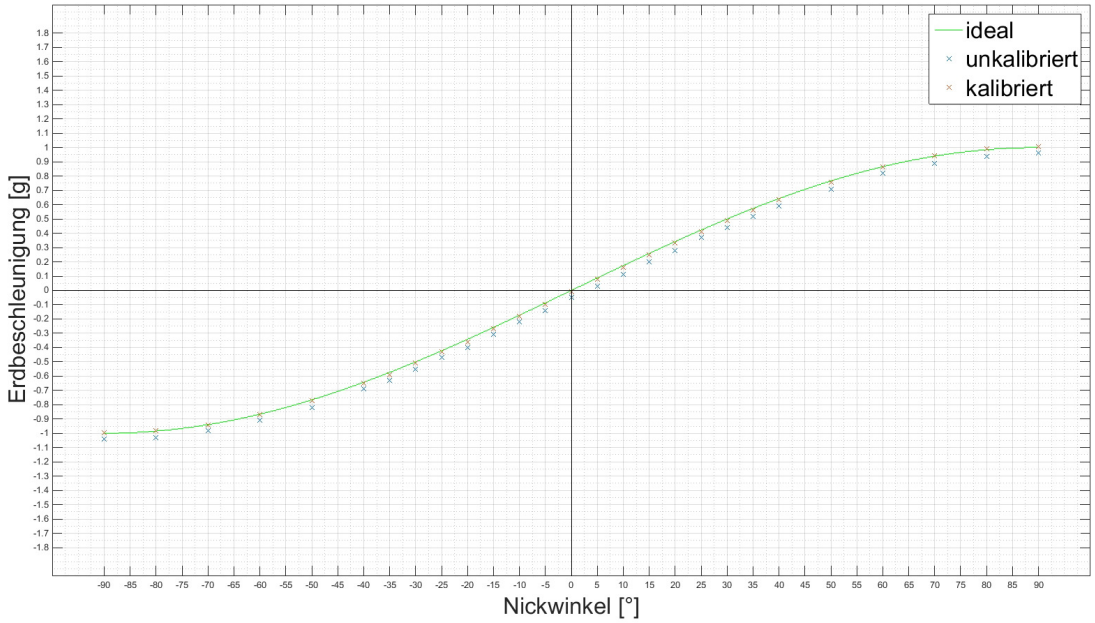

Abbildung 5.6: Beschleunigungswerte X-Achse über Nickwinkel θ

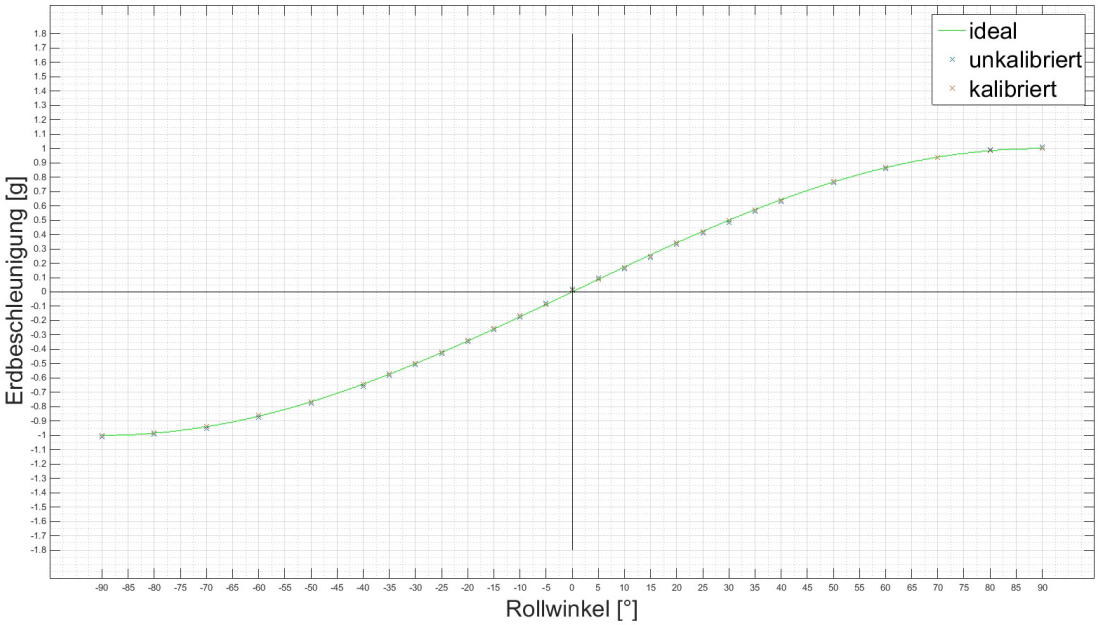

Abbildung 5.7: Beschleunigungswerte Y-Achse über Rollwinkel ϕ

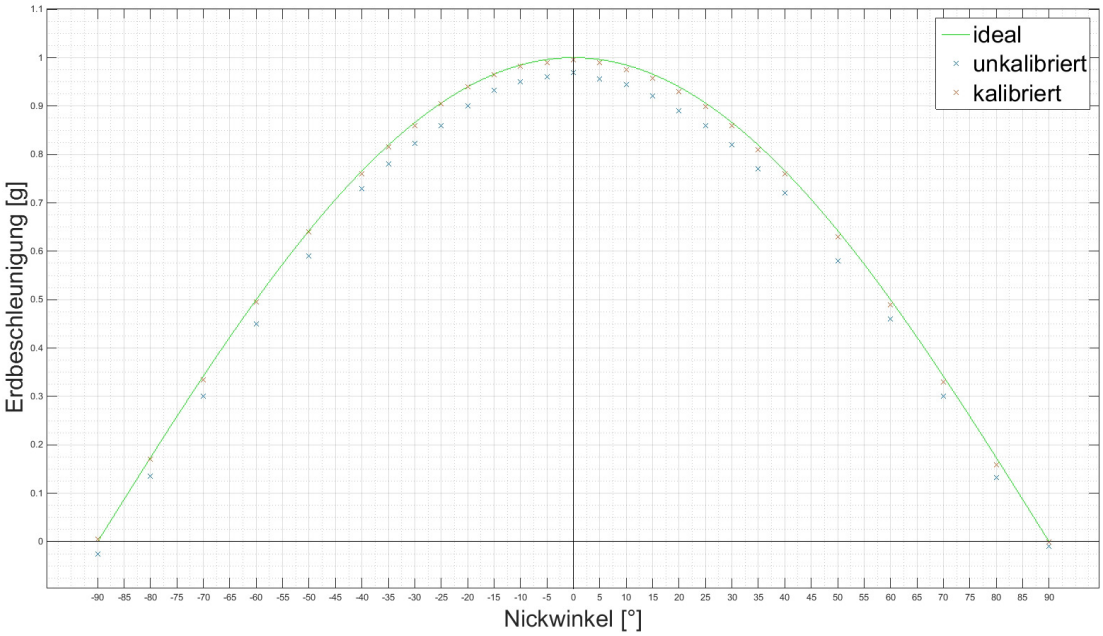

Abbildung 5.8: Beschleunigungswerte Z-Achse über Nickwinkel ϕ

In Abbildung 5.6 ist die Verbesserung der Kalibrierung gut zu sehen. Es ist zu sehen, dass sich die kalibrierte Beschleunigung der X-Achse mit der idealen Kennlinie überlagert, im Gegensatz zur unkalibrierten Kennlinie, welche einen negativen Offset besitzt.

In Abbildung 5.7 ist zu sehen, dass nur eine minimale Verbesserung erreicht werden kann. Das liegt daran, dass sich die Y-Achse unkalibriert schon sehr gut der idealen Kennlinie annähert.

In Abbildung 5.8 ist eine deutliche Verbesserung der Messwerte durch die Kalibrierung zu sehen. Bei 0° Nickwinkel sollte die volle Erdbeschleunigung von 1g gemessen werden. Der kalibrierte Messwert ergab ein Beschleunigung von 0,995g was eine Abweichung von 0,5 % entspricht.

5.3 Kalibrierung der Gyroskope

Um die Gyroskope zu kalibrieren wird in dieser Arbeit ein dynamischer Prüfstand (ACT 1001 PA), auch Drehteller genannt, verwendet. Dieser Prüfstand der Firma Acutronic ermöglicht es präzise Winkelgeschwindigkeiten und Winkel einzustellen. Der Drehteller liefert das Referenzsignal (Winkelgeschwindigkeit), welches als Vergleich zu der gemessenen Winkelgeschwindigkeit der Gyroskope dient. Des Weiteren kann der Drehteller benutzt werden, um spätere Messungen mit dem Kalman Filter durchzuführen. Um die Werte des Drehtellers als Vergleich nutzen zu können, mussten folgende Anpassungen umgesetzt werden:

- Der Aufbau, der schon bei der Kalibrierung des Beschleunigungssensors verwendet wird, muss auf den Drehteller montiert werden.
- Das Auslesen der Messdaten des Drehteller (Winkelgeschwindigkeit mit Zeitstempel)
- Die Synchronisation des Drehtellers mit dem Raspberry PI

5.3.1 Anpassungen zur Benutzung des Drehtellers

In diesem Unterkapitel werden die einzelnen Anpassungen beschrieben, welche zur Kalibrierung der Gyroskope benötigt werden.

Montage des Versuchsaufbaus

Der Bereits verwendete Aufbau, welcher bei der Kalibrierung des Beschleunigungssensors benutzt wurde, wird so angepasst, dass dieser auf dem Drehteller montiert werden kann. Die Verbindung mit dem Rapsberry PI, also die Datenleitungen (I^2CBus) und die Spannungsversorgung des Sensors, werden über Schleifringkontakte im Drehteller realisiert. Es ist darauf zu achten, dass der Drehteller immer ausgerichtet ist, da sonst Messfehler entstehen können.

Abbildung 5.9: Montageaufbau des Drehtellers zum Kalibrieren der Gyroskope

Bedienung und Auslesen der Messdaten des Drehteller

Zur Benutzung des Drehtellers dient eine vorhandene Software von Peters (2013). Diese ermöglicht es über eine grafische Oberfläche, welche in C# programmiert wurde, den Drehteller zu bedienen. Es ist in dieser Grafischen Oberfläche nur möglich die Messwerte vom Drehteller anzuzeigen. Eine Speicherung mit Zeitstempel in eine Datei war nicht vorgesehen. Die originale Software muss also so geändert werden, dass es möglich ist Messreihen mit Zeitstempel zu speichern. Nach dem Einarbeiten in den Programmcode, ist die Datei Betriebsartform.cs so verändert worden, dass bei starten des Drehteller mit Shift F8 (siehe Abbildung 5.10), Listen mit der Winkelgeschwindigkeit und der Zeit erstellt werden. Alte Daten in den Listen werden durch Shift F8 gelöscht. Das Herunterfahren des Drehtellers geschieht mit Shift F10 (siehe

Abbildung 5.10). Nach Betätigen von Shift F10 werden die Listen in einer Test.txt Datei auf dem Laufwerk C gespeichert. Die alte Text Datei wird dabei komplett ersetzt.

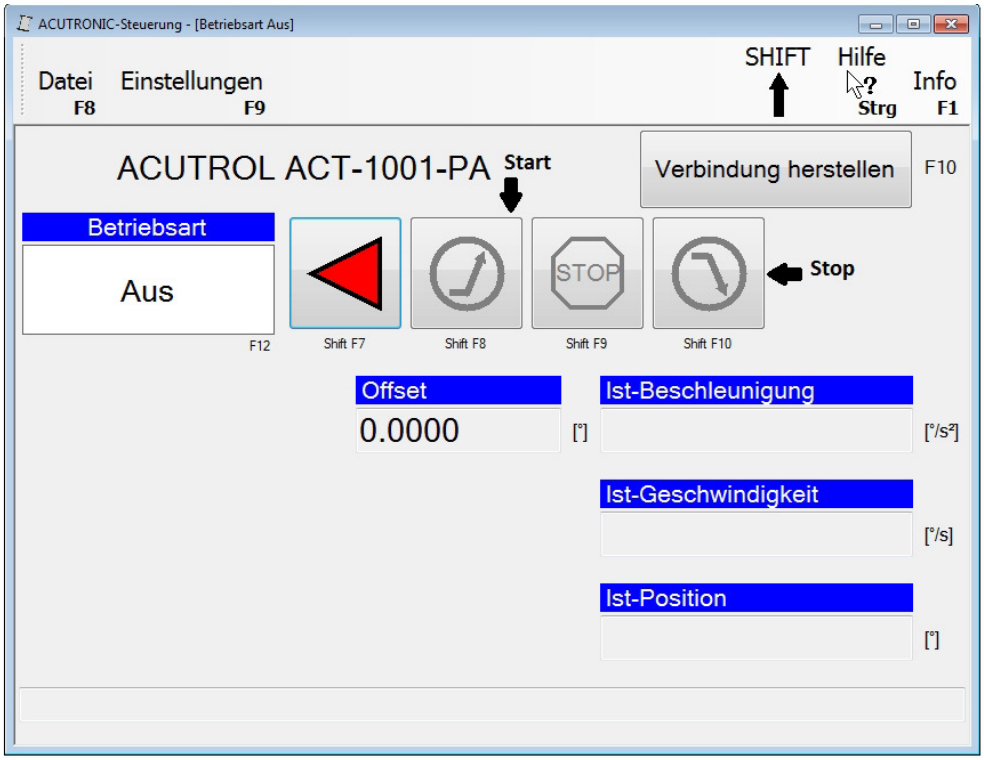

Abbildung 5.10: Grafische Oberfläche zur Bedienung des Drehtellers

Synchronisation zwischen Raspberry PI und Drehteller

Um die Messwerte der Gyroskope mit den Messwerten des Drehtellers vergleichen zu können, muss das Speichern synchronisiert werden. Der Drehteller wird über eine RS232 Schnittstelle durch die Acutronic Software auf dem PC gesteuert. Der Raspberry PI ist über ein Ethernet Kabel mit dem PC verbunden. Damit die Kommunikation zwischen dem Raspberry PI und dem PC funktioniert, muss auf dem PC zunächst ein DHCP Server eingerichtet werden. Die Synchronisation findet mittels einer UDP (User Datagramm Protocol) Übertragung statt. Hierbei handelt es sich um ein ungesichertes und ungeschütztes Übertragungsprotokol (vlg. Elektronik Kompendium (o. J.)), was aber für die Synchronisation nicht von Belang ist. Sobald der Drehteller mit Shift F8 (siehe Abbildung 5.10) gestartet wird, überträgt die veränderte Acutronic Software über UDP eine Eins an den Raspberry PI. Beim Stoppen des Drehtellers über Shift F10 sendet die Acutronic Software eine Zwei an der Raspberry PI.

Um dies zu ermöglichen ist in Betriebsartform.cs der Acutronic Software der Programmcode verändert worden. Der Raspberry PI wartet so lange bis dieser über UDP eine Eins empfängt und startet die Messung mit den Messwerten des Gyroskops. Durch den Empfang einer Zwei wird die Messreihe beendet und die Daten gespeichert. Um die Messwerte zu visualisieren wird die Test.txt Datei mit den Daten des Drehtellers und die Daten, die vom Raspberry PI aufgenommen wurden, durch Matlab eingelesen und visualisiert.

5.3.2 Kalibrierung

Während des Messens der Gyroskop Drehraten im Ruhezustand, wie in Abbildung 5.2 zu sehen ist, ist aufgefallen, dass sich der Bias nach jedem Neustart der Messung verändert. Der Bias der Gyroskope ist also nicht konstant wie bei den Beschleunigungssensoren. Deshalb wird die Bias-Korrektur nicht wie bei den Beschleunigungssensoren einmal durchgeführt, sondern bei jedem Start des Programms. Es werden pro Achse jeweils 500 Messwerte aufgenommen und der Mittelwert berechnet, um die Bias Korrektur o_z, als Beispiel für die Z-Achse des Gyroskops, zu erhalten.

Der Skalierungsfaktor um den Steigungsfehler der Gyroskope zu eliminieren wird laut (Titterton; Weston, 2004) ermittelt indem jede Achse auf ihre maximale Drehrate beschleunigt wird. In dem Datenblatt des Sensors steht, dass die maximale Drehrate der Achsen $\pm 245°/s$ beträgt (vgl. STMicroelectronics (2013), S. 13). Nachdem nun der Bias-Fehler behoben ist, wird mit Hilfe des Drehtellers jede Achse mit +245°$/s$ und -245°$/s$ gemessen. Die Messwerte des Drehtellers dienen dabei als Vergleich. Mit Hilfe der folgenden Formel wird der Skalierungsfaktor für die Z-Achse berechnet:

$$s_z = \frac{z_{max} - z_{min}}{2} \tag{5.7}$$

Durch die folgende Formel lässt sich nun die Kennlinie der Gyroskope korrigieren (Beispiel Z-Achse):

$$\Omega z_{korrigiert} = \frac{\Omega x_{gemessen} - o_z}{s_z} \tag{5.8}$$

Die folgende Tabelle zeigt die gemessenen Werte und den Skalierungsfaktor jeder Achse:

Achse	-245°/s	+245°/s	Skalierungsfaktor
X	-238,86°/s	+238,31 °/s	0,974
Y	-244,32°/s	+243,86°/s	0,996
Z	-250,68°/s	+250,28 °/s	1,0225

Tabelle 5.5: Skalierungsfaktoren der Gyroskope

Die Abbildungen 5.11 und 5.12 zeigen anhand der Z-Achse, dass der Steigungsfehler behoben wurde. Es ist zu sehen, dass die kalibrierte Drehrate sich mit der Referenzdrehzahl des Drehtellers überlagert, sowohl die -245°/s, wie auch die +245°/s Drehrate.

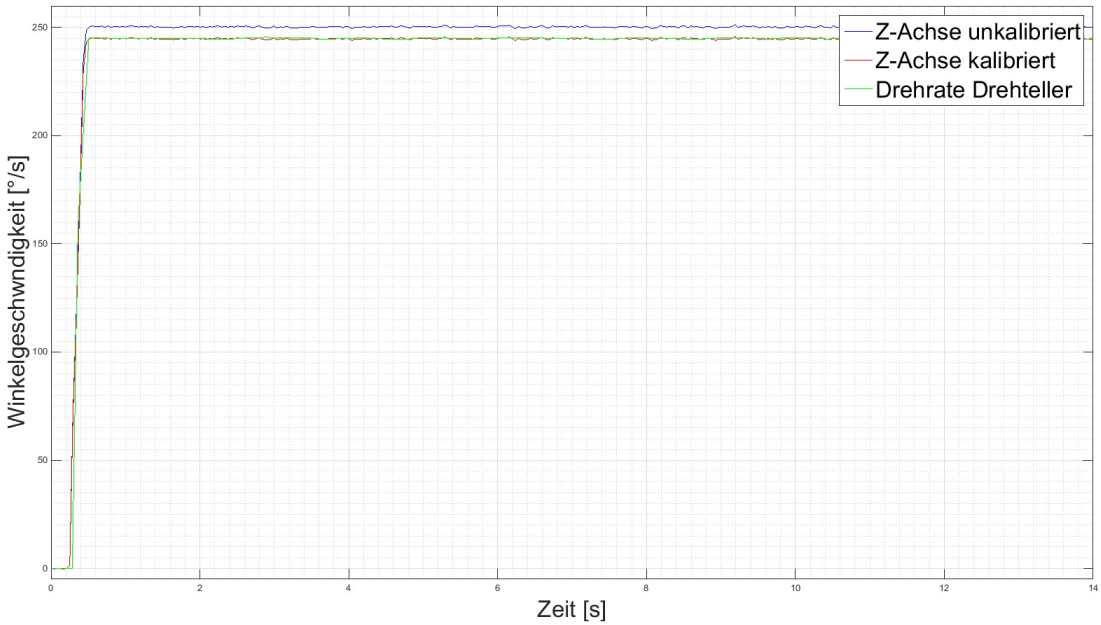

Abbildung 5.11: Messwerte bei einer Drehrate von +245°/s

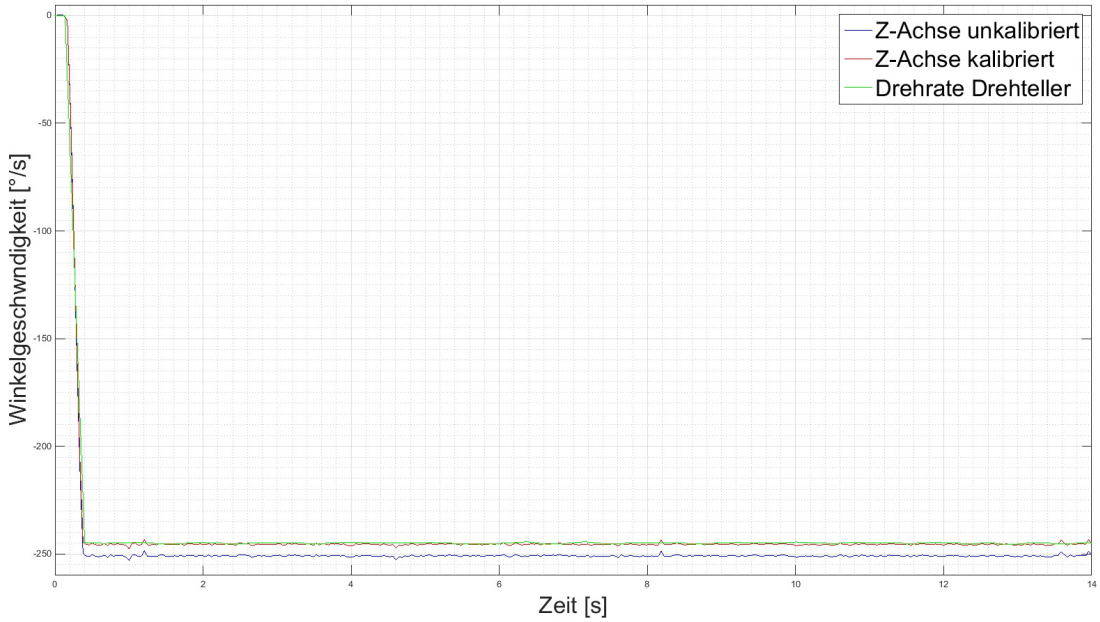

Abbildung 5.12: Messwerte bei einer Drehrate von -245°$/s$

Die Abbildung 5.13 zeigt die Drehraten aller Achsen nach der Bias- und der Steigungsfehlerkalibrierung. Die Gyroskope weisen immer noch ein Rauschen auf. Es ist immer noch ein geringer Bias Offset zu messen, welcher aber sehr gering ausfällt. Die folgende Tabelle zeigt die Mittelwerte der Gyroscope nach der Kalibrierung:

Achse	Mittelwert[°/s]
X	0,0088
Y	0,0173
Z	0,0590

Tabelle 5.6: Mittelwert der Gyroskope nach der Kalibrierung

Die Abbildung 5.14 zeigt, dass die kalibrierten Gyroskope einen wesentlich geringeren Drift aufweisen als unkalibriert. Für eine genau Winkelbestimmung sind diese trotzdem ungeeignet, da über einen längeren Zeitraum die Winkel abweichen. Bei mehreren Messung wurde festgestellt, dass jedes Mal der Drift der Winkel anders verläuft. Hierfür verantwortlich ist der stochastische Anteil des Bias. Zur Lagebestimmung über einen längeren Zeitraum ist eine Stützung durch die Beschleunigungssensoren nötig, dies geschieht mit Hilfe des Kalman Filters.

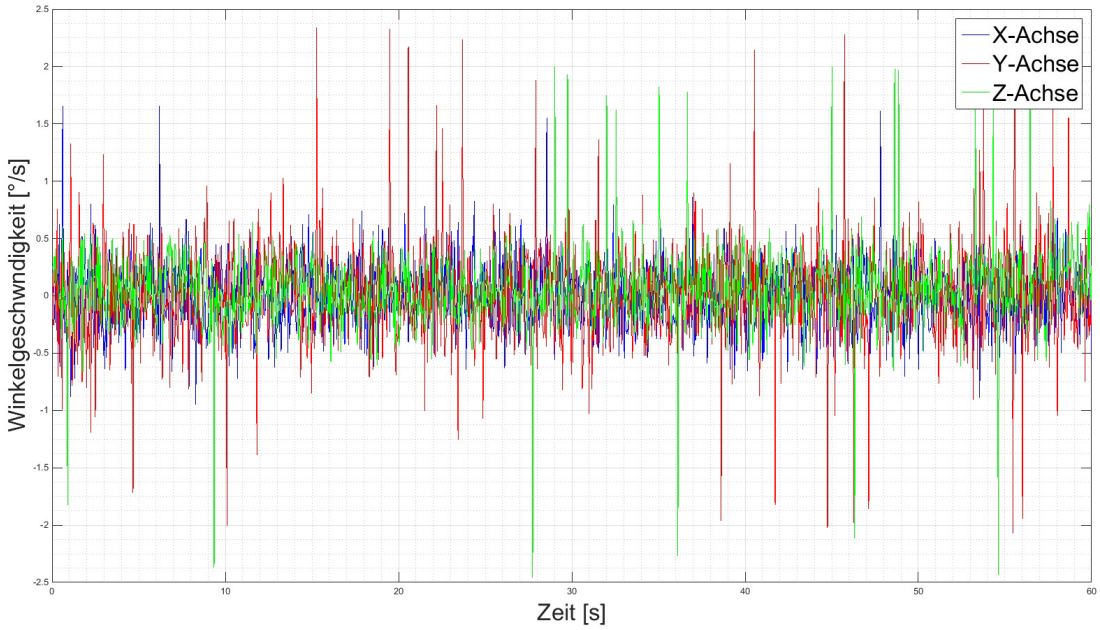

Abbildung 5.13: Stillstandmessung der Gyroskope nach Kalibrierung

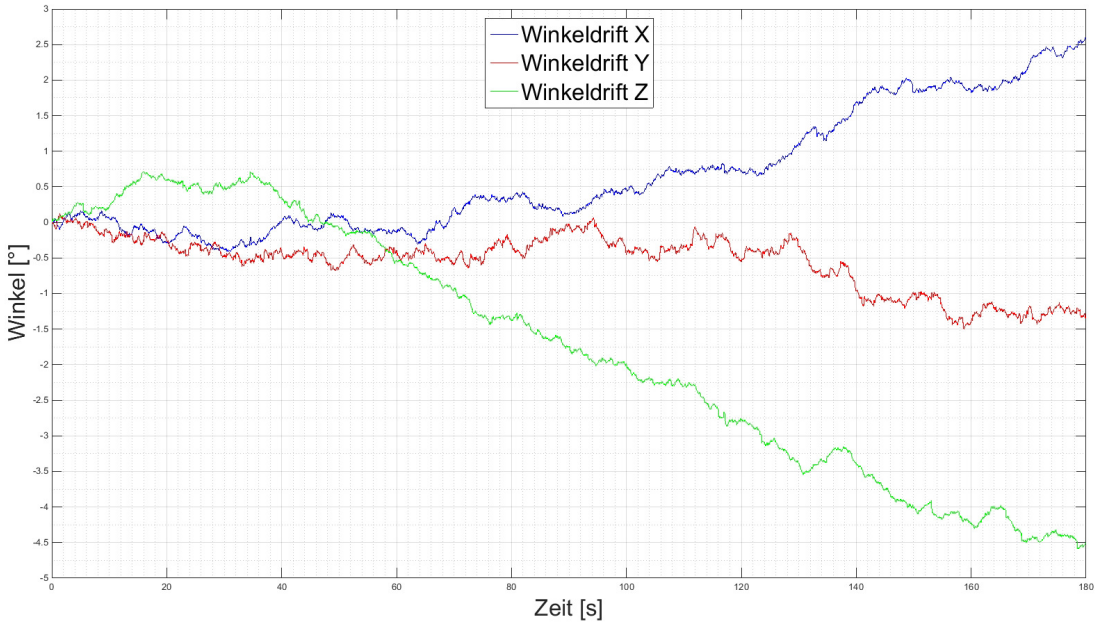

Abbildung 5.14: Winkeldrift der kalibrierten Gyroskope

6 Kalman Filter

Das Kalman Filter wurde von Emil Kalman im Jahre 1960 zuerst vorgestellt. Seit dem wurde das Kalman Filter für zahlreiche Anwendungen benutzt. Durch unterschiedliche Anforderungen gibt es mehrere Varianten des Kalman Filters, wie zum Beispiel den linearisierten Kalman Filter (LKF), den extended Kalman Filter (EKF), den Sigma-Point Kalman Filter oder den Kalman Filter 2. Ordnung.
Das Kalman Filter schätzt anhand eines Systemmodells und den Informationen des vorherigen Zeitintervalls die Zustände dieses Systems (a-proiri). Zusätzlich zur Schätzung erhält das Kalman Filter als Eingabe eine Messung des Zustandes und optimiert damit die Vorhersage (a-posterori). Die a-posterori Schätzung wird im nächsten Zeitintervall die neue a-priori Schätzung.
Das lineare Kalman Filter ist der optimale Schätzalgorithmus für ein lineares System. Kein anderes Filter kann im Bereich der Zustandsschätzung bessere Ergebnisse erzielen. Er liefert im Vergleich zu anderen Schätzern die Schätzwerte mit der kleinsten Varianz (vgl. Wendel (2011), S. 139).
Das Kalman Filter wird für viele Anwendungen benötigt wie zum Beispiel für folgende:

- Zur Vorhersage der Geschwindigkeit eines Objektes bei Vorhandensein der Position und des Systemmodells
- Zur Vorhersage der Position bei Navigationsgeräten
- In der Raumfahrt zur Flugbahnbestimmung
- Echokompensation bei der Sprachübertragung

Um den Kalman Filter zu verstehen hat die Literatur von Kim (2011) und Wendel (2011) eine große Hilfestellung geleistet. In dieser Arbeit wird der Kalman Filter-Algorithmus zur Fusionierung der Messwerte der Gyroskope mit den der Beschleunigungssensoren und des Magnetometer verwendet. Des Weiteren bietet er die Möglichkeit das Rauschen der Sensoren zu mindern. Die Fusionierung ist möglich, da die Sensoren die Winkel des Schiffes messen, dies aber unabhängig voneinander tun.

Der Beschleunigungssensor und das Magnetometer sind beim statischen Zustand des Bootes hervorragend geeignet die Roll-, Nick- und Gierwinkel des Bootes zu bestimmen, aber nicht in der Lage bei hoher Dynamik gute Messwerte zu liefern. Im Gegensatz zum Beschleunigungssensor und Magnetometer ist das Gyroskop in der Lage gute Messwerte bei hoher Dynamik zu liefern, driften aber im Gegensatz zum Beschleunigungssensor und dem Magnetometer über einen längeren Zeitraum ab.
Mit Hilfe des Kalman Filters wird versucht die gute Langzeitstabilität der Beschleunigungssensoren und des Magnetometers zu nutzen (statischer Zustand des Bootes) und bei hohen Winkelgeschwindigkeiten mehr Wert auf die Messwerte der Gyroskope zu legen.

6.1 Algorithmus des linearen Kalman Filters

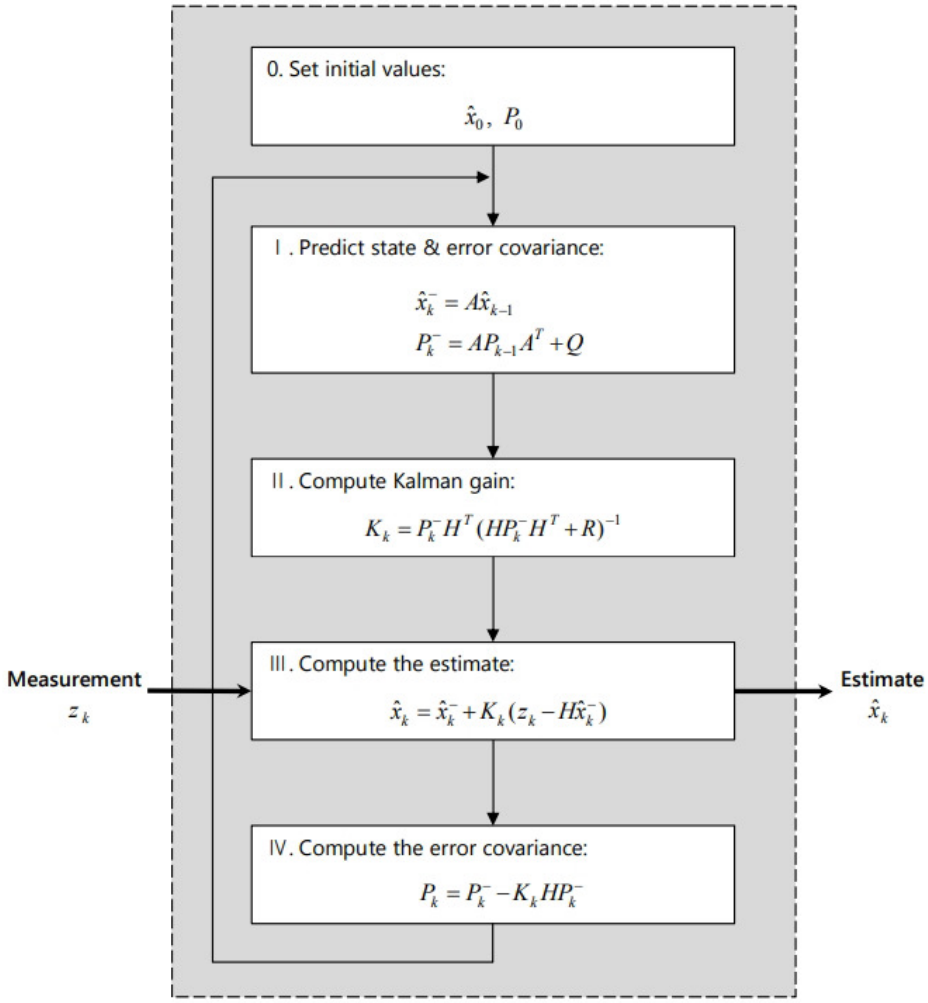

Abbildung 6.1: Algorithmus des linearen Kalman Filters (LKF) (Kim (2011), S. 49)

Die Abbildung 6.1 zeigt den Algorithmus des linearen Kalman Filters. Dazu ist zu sagen, dass in dieser Arbeit das extended Kalman Filter benutzt wird, da das System durch trigonometrische Funktionen beschrieben wird, welche nicht linear sind. Für die Beschreibung des Algorithmus spielt das aber keine Rolle, da der Unterschied im Algorithmus gering ist. Der Unterschied wird in Kapitel 6.2 beschrieben.

Der Kalman Filter-Algorithmus lässt sich in 3 Teile aufspalten:

Initialisierung

In Schritt 0 des Algorithmus werden die Anfangswerte für $\hat{x}_o$ und $\hat{P}_o$ bestimmt. Dabei ist $\hat{x}_o$ der erste Schätzwert des Kalman Filters und wird entweder mit 0 oder mit bekannten Werten für die erste Schätzung initialisiert.
Die Fehler-Kovarianz $\hat{P}_o$ indiziert die Differenz zwischen der Schätzung des Kalman Filters und dem wahren Wert, gibt also den Genauigkeitsgrad an. Falls keine Werte für $\hat{P}_o$ bekannt sind, sollte die Fehler-Kovarianz mit einem relativ hohen Wert wie Zehn initialisiert werden laut Kim (2011).

Vorhersageprozess

In Schritt I wird die Vorhersage für die aktuelle Schätzung $\hat{x}_k^-$ und die aktuelle Fehler-Kovarianz P_k^- berechnet, ausgehend von der letzten Zustandsverteilung $\mathcal{N}(\hat{x}_{k-1}, P_{k-1})$. Dies geschieht mit Hilfe der letzten Schätzung $\hat{x}_{k-1}$ und der letzten Fehler-Kovarianz P_{k-1}. Für die Vorhersage werden die Systemmodelle A und Q benötigt.

Schätzprozess

Zum Schätzprozess gehören die nachfolgend beschriebenen Schritte II, III und IV.

Schritt II zeigt die Berechnung der Kalman-Verstärkung K_K. Die Kalman-Verstärkung gibt an wie stark die Messung gegenüber $\hat{x}_k^-$ gewichtet wird. Zur Bestimmung werden P_k^-, R und H benötigt. Die H-Matrix spiegelt den Zusammenhang der Messwerte und den Zustandsvariablen wieder. Die R-Matrix ist die Kovarianz des Rauschens der Sensoren.

In Schritt III wird die verbesserte Schätzung $\hat{x}_k$ bestimmt. Dies geschieht mit Hilfe der Differenz zwischen den Messwerten und den vorhergesagten Schätzungen, gewichtet mit der Kalman Verstärkung und addiert mit der vorhergesagten Schätzung.

Schritt IV bestimmt die neue Fehler-Kovarianz P_K für die verbesserte Schätzung $\hat{x}_k$.

Zusammengefasst werden zuerst die vorhergesagte Schätzung $\hat{x}_k^-$ und die Fehler-Kovarianz P_k^- für den nächsten Schritt geschätzt, basierend auf den Systemmodellen A und Q. Anschließend wird die Differenz zwischen den Messwerten und der vorhergesagten Schätzung kompensiert und eine neue, verbesserte Schätzung berechnet: $\hat{x}_k$ und P_K. Danach werden diese beiden Schritte in einer Schleife wiederholt.

6.2 Extended Kalman Filter (EKF)

Wie bereits erwähnt, wird in dieser Arbeit das extended Kalman Filter verwendet, da das System nicht linear ist. Dies entsteht aufgrund der trigonometrischen Funktionen, welche bei der Berechnung der Eulerwinkel benutzt werden. Die Algorithmen unterscheiden sich dabei nur an zwei Stellen, wie in Abbildung 6.2 durch die rot markierten Bereiche zu sehen ist. Aus $A\hat{x}_{k-1}$ wird der Ausdruck $f(\hat{x}_{k-1})$. Dabei wird das lineare Systemmodel A durch die sogenannte Jacobi-Matrix ersetzt. Durch die Jacobi-Matrix wird das nichtlineare System um die vorhergesagten Schätzungen des vorherigen Schrittes linearisiert. Der EKF ist kein optimales Filter mehr, da ein nichtlineares System durch ein abgeleitetes lineares System beschrieben wird.

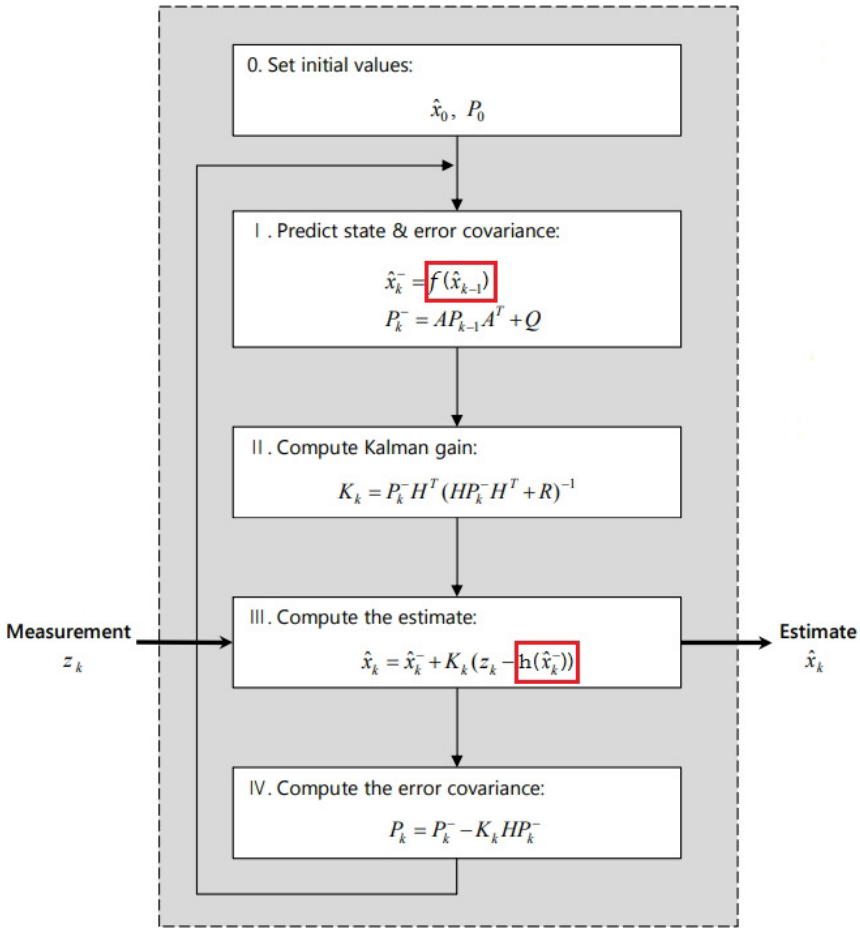

Abbildung 6.2: Algorithmus des extended Kalman Filters (EKF) (vgl. Kim (2011), S. 146)

6.2.1 Fusionierung

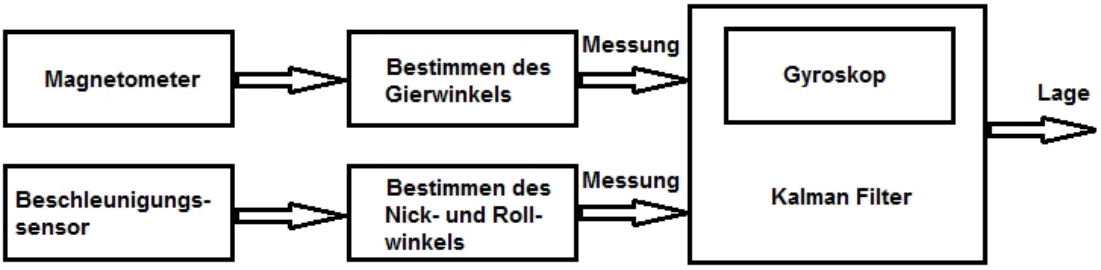

Abbildung 6.3: Schematische Darstellung der Sensorfusion mittels Kalman Filter

In Abbildung 6.3 ist die schematische Darstellung der Fusionierung des extended Kalman Filters zu sehen. Die Roll- und Nickwinkel des Bootes werden aus den Daten der Beschleunigungssensoren bestimmt. Der Gierwinkel hingegen wird aus den Daten des Magnetometers berechnet. Die Roll- und Nickwinkel der Beschleunigungssensoren und der Gierwinkel des Magnetometers dienen dem extended Kalman Filter als neue Messwerte für die Berechnung der verbesserten Schätzung der Eulerwinkel. Im Vorhersageprozess des EKF dienen die Messwerte der Gyroskope zur Bestimmung von Roll-, Nick- und Gierwinkel. Das Kalman Filter ist somit in der Lage die Vorteile aller Sensoren für die Lagebestimmung zu nutzen. Dazu gehören die gute Langzeitstabilität der Beschleunigungssensoren und des Magnetometers sowie die guten Eigenschaften der Gyroskope bei hohen Dynamiken.

6.2.2 Systemmodelle A und H

Für die Berechnung des Kalman Filters ist es notwendig unser Systemmodell, welches aus den Differentialgleichungen 2.2 bis 2.4 besteht, in die Matrix-Form zu bringen und daraus die Jacobi-Matrix zu bilden.

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} p & q\sin\phi * \tan\theta & r\cos\phi * \tan\theta \\ 0 & q\cos\phi & -r\sin\phi \\ 0 & q\sin\phi/\cos\theta & r\cos\phi/\cos\theta \end{bmatrix} + w \tag{6.1}$$

$$\begin{bmatrix} \dot{\Phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = f(x) + w \tag{6.2}$$

In den Formeln 6.1 und 6.2 gibt w das Rauschen des Systems an, was zum Beispiel durch Wind und Wellen verursacht werden kann.

Um einen extended Kalman Filter zu implementieren, wird nun die Jacobi-Matrix aus dem Systemmodell der Gleichung 6.1 berechnet. Um die Jacobi-Matrix zu berechnen, wird aus jedem Term des Systemmodells die partielle erste Ableitung gebildet.

$$A_{Jacobi} = \begin{bmatrix} \frac{\partial f_1}{\partial \phi} & \frac{\partial f_1}{\partial \theta} & \frac{\partial f_1}{\partial \psi} \\ \frac{\partial f_2}{\partial \phi} & \frac{\partial f_2}{\partial \theta} & \frac{\partial f_2}{\partial \psi} \\ \frac{\partial f_3}{\partial \phi} & \frac{\partial f_3}{\partial \theta} & \frac{\partial f_3}{\partial \psi} \end{bmatrix} \tag{6.3}$$

Es entstehen folgende partielle Ableitungen des Systemmodells:

$$\frac{\partial f_1}{\partial \phi} = q \cdot \cos\phi \cdot \tan\theta - r \cdot \sin\phi \cdot \tan\theta \tag{6.4}$$

$$\frac{\partial f_1}{\partial \theta} = q \cdot \frac{\sin\phi}{\cos(\theta)^2} \cdot +r \cdot \cos\phi \cdot \frac{\cos\phi}{\cos(\theta)^2} \tag{6.5}$$

$$\frac{\partial f_1}{\partial \psi} = 0 \tag{6.6}$$

$$\frac{\partial f_2}{\partial \phi} = -q \cdot \sin\phi - r \cdot \cos\phi \tag{6.7}$$

$$\frac{\partial f_2}{\partial \theta} = 0 \tag{6.8}$$

$$\frac{\partial f_2}{\partial \psi} = 0 \tag{6.9}$$

$$\frac{\partial f_3}{\partial \phi} = q \cdot \frac{\cos\phi}{\cos\theta} - r \cdot \frac{\sin\phi}{\cos\theta} \tag{6.10}$$

$$\frac{\partial f_3}{\partial \theta} = q \cdot \sin\phi \cdot \frac{\sin\theta}{2 \cdot \cos(\theta)^2} + r \cdot \cos\theta \cdot \frac{\sin\theta}{2 \cdot \cos(\theta)^2} \tag{6.11}$$

$$\frac{\partial f_3}{\partial \psi} = 0 \tag{6.12}$$

Ein weiteres Systemmodell ist die H-Matrix. Die H-Matrix beschreibt wie die Messwerte mit denen der Zustände zusammenhängen. Die Messwerte hängen in dieser Arbeit linear mit denen der Zustände zusammen, da in der Berechnung direkt die Winkel benutzt werden. Mit dem Rauschen v der Sensoren ergibt sich folgendes Modell:

$$z = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} + v \tag{6.13}$$

$$z = H \cdot x \cdot v \tag{6.14}$$

6.2.3 Kovarianzen Q und R

Die Kovarianzmatrix Q und R beschreiben im Kalman Filter das Rauschverhalten des Systems und der Sensordaten. Sie sind ein wichtiger Faktor damit das Kalman Filter richtig funktioniert, daher werden Sie auch als Tuningparameter bezeichnet.

Kovariazmatrix Q

Die Kovarianzmatrix Q beschreibt das Rauschenverhalten des Systems an sich. Diese entstehen zum Beispiel durch äußere Einflüsse auf das System selbst. Diese äußeren Einflüsse können Wind, Wellen oder bewegende Personen an Bord des Schiffes sein. Da diese Einflüsse nicht exakt beschrieben werden können, werden die besten Parameter für die Matrix Q in mehreren Versuchsreihen ermittelt. Da die Versuchsreihen im Labor geschehen, kann es notwendig sein diese Parameter erneut anzupassen.

Kovariazmatrix R

Die Kovarianzmatrix R beschreibt das Rauschen der Sensoren. Diese Parameter werden vom Hersteller gestellt oder werden, wie es in Kapitel 5 nachzulesen ist, ermittelt. Die in Kapitel 5 ermittelten Varianzen für die Beschleunigungssensoren, Gyroskope und das Magnetometer werden als konstante Parameter für die Kovarianzmatrix R verwendet.

Der Einfluss auf die Funktion und die Gewichtung des EKF kann durch folgende Formel des zweiten Schrittes dargestellt werden:

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \tag{6.15}$$

Dabei handelt es sich um die Berechnung der Kalman-Verstärkung. Werden nun die Matrizen durch Skalare für die richtige Betrachtung getauscht, entsteht eine simple Division von Skalaren (Kim (2011), S. 68):

$$K_k = \frac{P_k^- H^T}{H P_k^- H^T + R} \tag{6.16}$$

Aus der Formel 6.16 ist nun zu sehen, dass bei einer Erhöhung von R die Kalman-Verstärkung sinkt. In der Formel 6.17 des dritten Schrittes des Kalman Filter-Algorithmus ist nun zu sehen, dass durch die Verringerung der Kalman-Verstärkung der Einfluss der Messung z_k auf die verbesserte Schätzung $\hat{x}_k$ weniger Einfluss hat. Das bedeutet, dass bei einer Erhöhung von den Parametern in der Kovarianzmatrix R mehr Wert auf die Messwerte der Gyroskope als auf die der Beschleunigungssensoren und des Magnetometers gelegt wird. Da in dieser

Arbeit die Kovarianzmatrix R durch die Varianzen der Sensoren bestimmt wird, bleibt uns die Kovarianzmatrix Q zum Einstellen des extended Kalman Filters.

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - h(\hat{x}_k^-)) \tag{6.17}$$

Die Kovarianzmatrix Q wird zur Berechnung der aktuellen Fehler-Kovarianz P_k^- verwendet. Die Formel dafür lautet:

$$P_k^- = AP_{k-1}A^T + Q \tag{6.18}$$

In Formel 6.18 ist zu sehen, dass eine Erhöhung der Parameter der Matrix Q automatisch zu einer höheren Fehler-Kovarianz P_k^- führt. Durch eine größeres P_k^- steigt auch der Wert der Kalman-Verstärkung K_k in Formel 6.16. In der Formel 6.16 steht P_k^- im Zähler sowie auch im Nenner. Trotzdem steigt die Kalman-Verstärkung aufgrund von R im Nenner. Das bedeutet also: Bei Erhöhung von Q steigt der Einfluss der aktuellen Messung auf die aktuelle Schätzung. Es wird mehr Wert auf die Messwerte der Beschleunigungssensoren und des Magnetometers gesetzt und weniger auf die Vorhersage durch die Gyroskope.

6.2.4 Fehler-Kovarianz

Die Fehler-Kovarianz im Kalman Filter zeigt die Differenz zwischen der Schätzung und dem wahren, unbekannten Wert an. Sie ist also ein Indikator wie gut die Schätzung des Kalman Filters ist. Wenn P_k groß ist, ist der Fehler der verbesserten Schätzung hoch und bei einem geringen Wert von P_k ist der Fehler der verbesserte Schätzung gering. Um die Fehler-Kovarianz besser zu verstehen, kann sich eine Normalverteilung (Glockenkurve) vorgestellt werden $N = (\hat{x}_k, P_k)$. Dabei gibt der Wert für P_k die Breite der Kurve an. Je größer P_k also ist, desto weiter kann die Schätzung vom wahren Wert abweichen.

In Schritt I des Algorithmus wird die aktuelle Fehler-Kovarianz P_k^- berechnet. Wenn P_k^- gegen unendlich geht, beinhaltet die bisherige Zustandsschätzung wenig brauchbare Informationen. In dem Fall wird die Zustandsschätzung verworfen und der Messwert übernommen. Geht P_k^- jedoch gegen Null, wird die aktuelle Zustandsschätzung als perfekt angesehen und nicht durch den Messwert verändert (vgl. Wendel (2011), S. 138).

7 Implementierung

In diesem Kapitel wird auf die Programmierung der Software und deren einzelnen Programmcodes eingegangen. Um den Ablauf des Programms besser zu verstehen, ist ein Programmablaufplan erstellt worden. Programmiert wurde auf einem Raspberry PI mit dem Betriebssystem Raspbian. Raspbian ist ein Linux-basiertes Betriebssystem und ist das offizel unterstütze Betriebssystem des Raspberry PI. Dabei wurde Geany als integrierte Entwicklungsumgebung benutzt. Als Programmiersprache wurde Python ausgewählt. Python bietet eine gute Strukturierung durch Einrücken und wird oft als Skriptsprache genutzt. Python bietet laut Downey (2014) eine objektorientierte, aspektorientierte und funktionale Programmierung. Die Bücher von Downey (2014), Haji (2008) und Weigend (2016) wurden zur Einarbeitung in die Programmiersprache genutzt.

Die Abbildungen 7.1 und 7.2 zeigen den Ablauf der Software in einem Programmablaufplan.

Die entwickelte Software ist auf der beigelegten CD abgelegt.

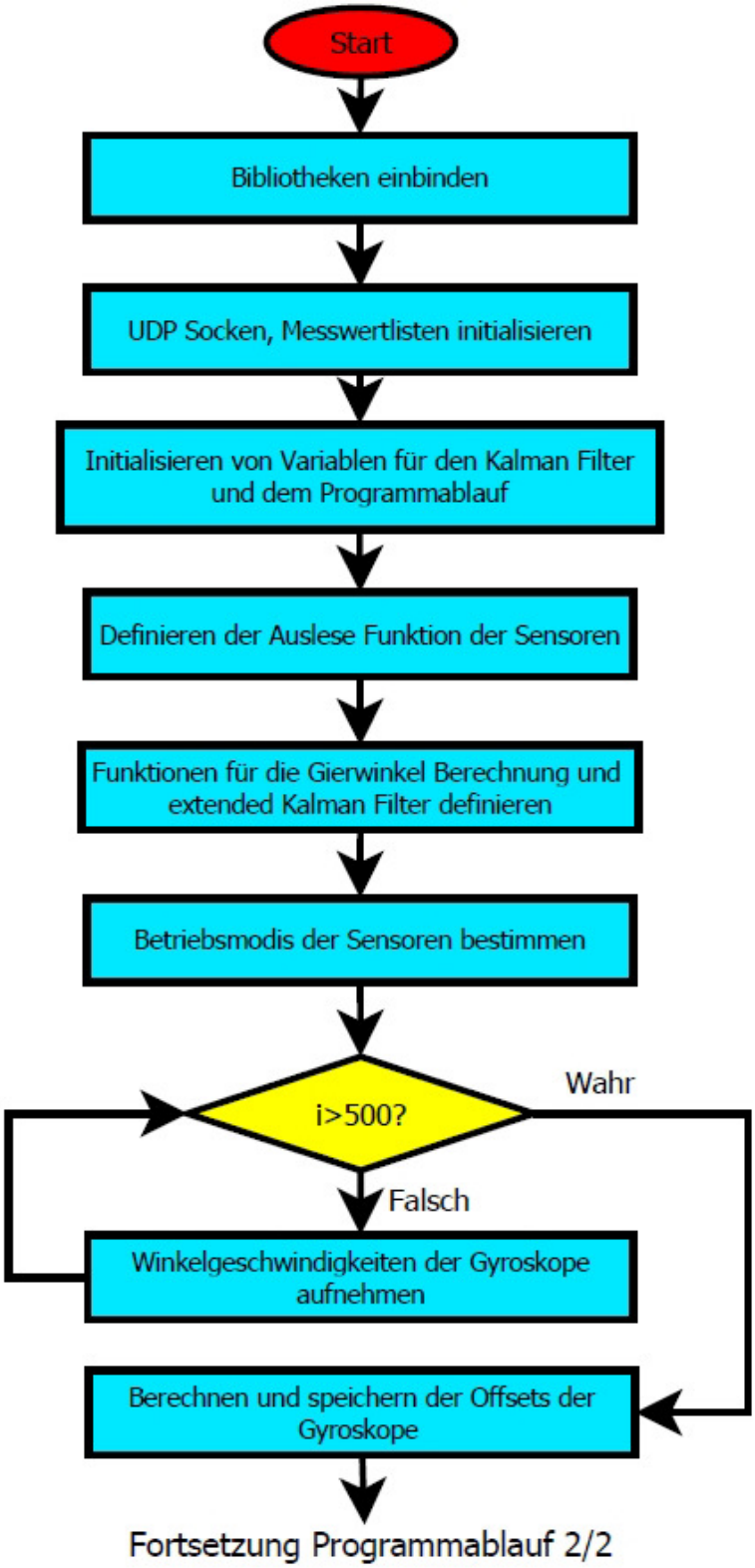

Abbildung 7.1: Schema Programmablauf 1/2

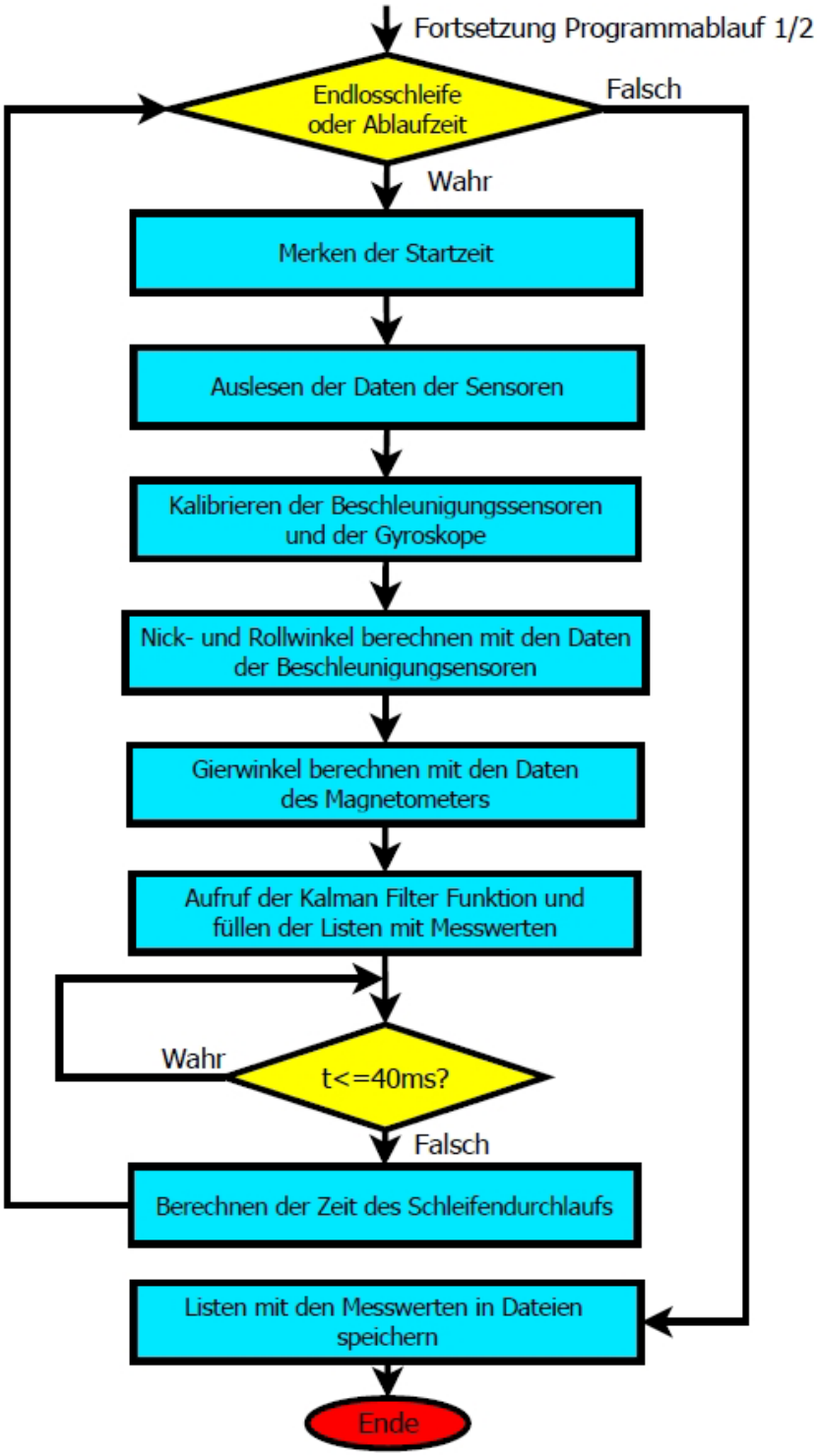

Abbildung 7.2: Schema Programmablauf 2/2

7.1 Betriebsmodi

Nachdem die Variablen des Kalman Filters und die Variablen, die für den Programmablauf benötigt werden, initialisiert sind, werden die Betriebsmodis der Beschleunigungssensoren, Gyrospkope und des Magnetometers bestimmt.

Gyroskope

Die Einstellung der Gyroskope geschieht in den Registern CTRL_ REG1_ G und CTRL_ REG4_ G.

```
#Betriebsmodi Gyroskop einstellen#
writeGYR(CTRL_REG1_G, 0b00001111) #Normaler power mode, alle Achsen aktiviert, 95HZ Output Datenrate, 12.5 Bandbreite
writeGYR(CTRL_REG4_G, 0b00000000) #Continuos update, 245 dps full scale
```

Abbildung 7.3: Einstellen der Betriebsmodi der Gyroskope

Die Messbereiche der Gyroskope werden auf eine Winkelgeschwindigkeit von ±245 Dps eingestellt, um die höchstmögliche Auflösung zu erreichen. Die Ausgabe der Messwerte geschieht mit einer Frequenz von 95Hz mit einer Bandbreite von 12,5Hz. Die Bandbreite gibt an, ab welcher Frequenz das Gyroskop ein Tiefpassfilter-Verhalten hat. Das bedeutet, wenn sich die Winkelgeschwindigkeiten mit einer höheren Frequenz als 12,5Hz ändern, wird das Signal durch einen Tiefpass Filter gedämpft. Alle drei Achsen werden aktiviert.

Beschleunigungssensoren

Die Einstellung der Beschleunigungssensoren geschieht in den Registern CTRL_ REG1_ M und CTRL_ REG2_ M.

```
#Betriebsmodi Beschleunigungssensor einstellen#
writeACC(CTRL_REG1_XM, 0b01100111) # Alle Achsen aktiviert, 100Hz Daten Rate, kontinuierliches Update
writeACC(CTRL_REG2_XM, 0b00000000) # 2G-Full-Scale
```

Abbildung 7.4: Einstellen der Betriebsmodi der Beschleunigungssensoren

Der Messbereich der Beschleunigungssensoren wird auf ±2g eingestellt und die Messwerte mit einer Frequenz von 100Hz ausgegeben. Alle drei Achsen werden aktiviert.

Magnetometer

Die Einstellung der Gyroskope geschieht in den Registern CTRL_ REG5_ XM, CTRL_ REG6_ XM und CTRL_ REG7_ XM.

```
#Betriebsmodi Magneometer einstellen#
writeMAG(CTRL_REG5_XM,0b11110000) #-Set magnetometer to high resolution - 50Hz Output Datenrate
writeMAG(CTRL_REG6_XM,0b00000000) #+/- 2Gauss
writeMAG(CTRL_REG7_XM,0b00000000) #Set the magnetometer to Continuous-conversion Mode
```

Abbildung 7.5: Einstellen der Betriebsmodi des Magnetometers

Die Datenrate des Magnetometers wird auf den höchstmöglichen Werte von 50Hz eingestellt. Die Zeit für einen Schleifendurchlauf schwankt zwischen 55ms und 75ms. Bei einer Taktrate von 50Hz ist somit sichergestellt, dass mindestens ein neuer Wert des Magnetometers ausgelesen werden konnte. Der Messbereich wurde auf ± 2 Gauss eingestellt, um eine hohe Auflösung zu gewährleisten.

Die Einstellungen der Gyroskope, Beschleunigungssensoren und des Magnetometers können in dem Datenblatt LSM9DS0.pdf, welches sich auf der beiliegenden CD befindet, nachgelesen werden.

7.2 Auslesen, Umrechnen und Kalibrieren der Sensormesswerte

Auslesen der Sensordaten

Der folgende Quellcode zeigt das Auslesen der Sensorrohdaten jeder Achse der Gyroskope. Die Sensorrohdaten der Beschleunigungssensoren und des Magnetometers werden auf die selbe Weise ausgelesen, aber aus anderen Registern. GYR_ ADDRESS ist dabei die I^2C Adresse der Gyroskope. Die Rohdaten werden aus zwei Registern mit je 8 Bits ausgelesen und anschließend kombiniert.

```
def readGYRx():
        gyr_l = bus.read_byte_data(GYR_ADDRESS, OUT_X_L_G)
        gyr_h = bus.read_byte_data(GYR_ADDRESS, OUT_X_H_G)
        gyr_combined = (gyr_l | gyr_h <<8)

        return gyr_combined  if gyr_combined < 32768 else gyr_combined - 65536

def readGYRy():
        gyr_l = bus.read_byte_data(GYR_ADDRESS, OUT_Y_L_G)
        gyr_h = bus.read_byte_data(GYR_ADDRESS, OUT_Y_H_G)
        gyr_combined = (gyr_l | gyr_h <<8)

        return gyr_combined  if gyr_combined < 32768 else gyr_combined - 65536

def readGYRz():
        gyr_l = bus.read_byte_data(GYR_ADDRESS, OUT_Z_L_G)
        gyr_h = bus.read_byte_data(GYR_ADDRESS, OUT_Z_H_G)
        gyr_combined = (gyr_l | gyr_h <<8)

        return gyr_combined  if gyr_combined < 32768 else gyr_combined - 65536
```

Abbildung 7.6: Auslesen der Rohdaten der Sensoren (Beispiel Gyroskope)

Rohdatenumrechnung

Bei den ausgelesen Daten handelt es sich um Rohdaten, welche noch je nach Sensor und Messbereich in die entsprechenden Einheiten umgerechnet werden müssen. Die genauen Werte können der Tabelle 4.1 entnommen werden oder im Datenblatt STMicroelectronics (2013), S.13 nachgelesen werden.

```
GYRx = readGYRx()*0.00875 #Datenblatt S. 13 Output in Degrees/s bei FS-Sensitivity= 245dps#
GYRy = readGYRy()*0.00875
GYRz = readGYRz()*0.00875

ACCx= readACCx()*0.000061035156 #Datenblatt S. 13; Output in 2G#
ACCy= readACCy()*0.000061035156
ACCz= readACCz()*0.000061035156

MAGx= readMAGx()*0.00008 #Datenblatt S. 13; Output in Gauss 2 Gauss#
MAGy= readMAGy()*0.00008
MAGz= readMAGz()*0.00008
```

Abbildung 7.7: Umrechnen der Rohdaten der Sensoren

Kalibrieren der Sensoren

Der folgende Code zeigt die Kalibrierung des Offsets und des Steigungsfehlers der Beschleunigungssensoren und der Gyroskope. Die Werte dafür wurden in Kapitel 5 ermittelt und können den Tabellen 5.4 und 5.5 entnommen werden. Der Offset für die Achsen der Gyroskope wird bei jedem Start des Programms neu ermittelt. Des Weiteren erfolgt eine Umrechnung der Daten des Gyroskops in rad/s.

```
#Offset und Skalierungsfaktor korrigierung der Beschleunigungssensor#
ACCx_cal= (ACCx-(-0.073))/0.997
ACCy_cal= (ACCy-(0.033))/1.002
ACCz_cal= (ACCz-(-0.021))/0.996

#Offset und Skalierungsfaktor korrigierung der Gyroskope und Umrechnung rad/s fuer Eulerwinkel #
GYRx_cal = (GYRx - (Offset_X))/0.974 * math.pi/180
GYRy_cal = (GYRy - (Offset_Y))/0.996 * math.pi/180
GYRz_cal = (GYRz - (Offset_Z))/1.0225 * math.pi/180
```

Abbildung 7.8: Kalibrierung der Beschleunigungssensoren und der Gyroskope

7.3 Berechnen des Roll-, Nick- und Gierwinkels

In diesem Kapitel wird die Berechnung der Winkel der Beschleunigungssensoren und des Magnetometers gezeigt. Diese Winkel dienen dem Kalman Filter als neue Messungen für die Berechnung der verbesserten Schätzung.

Gierwinkelberechnung durch Magnetometer

Der folgende Programmcode berechnet den Gierwinkel (Kompass oder im englischen auch heading genannt) des Schiffes mit Tilt-Kompensation. Die Berechnung erfolgt anhand der Formeln aus Kapitel 2.2.3.

```
def Calcheading(X,Y,Z): # Berechnung des Gier Winkels mit Tilt kompensation

        deklination = 39.56/1000 #Deklaration fuer Hamburg

        roll = x[0,0]*(-1)
        pitch = x[1,0] *(-1)

        magXcomp = X*math.cos(pitch) + Z*math.sin(pitch)
        magYcomp = X*math.sin(roll)*math.sin(pitch) + Y*math.cos(roll) - Z*math.sin(roll)* math.cos(pitch)

        heading = 180*math.atan2(magYcomp,magXcomp)/math.pi + deklination*180/math.pi #Berechnung heading mit deklination

        if heading < 0:
            heading+=360

        return heading
```

Abbildung 7.9: Ermitteln des Gierwinkels mit Tilt-Kompensation

Roll-und Nickwinkelbestimmung durch Beschleunigungssensoren

Der folgende Quellcode zeigt den Aufruf der Funktion Calcheading, welche für die Berechnung des Gierwinkels zuständig ist. Des Weiteren wird gezeigt wie die Berechnung des Roll-und Nickwinkels mit Hilfe der Beschleunigungssensoren stattfindet (Siehe Kapitel 2.2.2).

```
#Aufruf zur Berechung des Gierwinkels#
kompass = Calcheading(MAGx,MAGy,MAGz)

#Eulerwinkel Berechnung durch Beschleunigungssensoren#
Theta_Acc = (math.asin(ACCx_cal))
Phi_Acc = (math.asin((-ACCy_cal)/math.cos(Theta_Acc)))
Psi_Acc = 0 #Gierwinkwel nicht Messbar mit den Beschleunigungssensoren#
```

Abbildung 7.10: Roll-und Nickwinkelbestimmung durch Beschleunigungssensoren

7.4 Funktionen des extended Kalman Filters

Im folgenden Unterkapitel wird kurz auf den Quellcode der einzelnen Funktionen des extended Kalman Filters eingegangen.

Extended Kalman Filter

Die Abbildung 7.11 zeigt den Quellcode der Funktion des extended Kalman Filters. In dieser Funktion befindet sich der komplette Algorithmus des extended Kalman Filters, welcher bei jeder Schleife neu aufgerufen wird. Dabei werden bei jedem Aufruf die neue Jacobi-Matrix (AJacobi()) berechnet und eine Schätzung (Prediction()). Die genaue Funktionsweise des extended Kalman Filters ist im Kapitel 6 nachzulesen.

```
def EulerEKF(): Algorythmus des Extended Kalman Filters(EKF)

    global x
    global P

    A = Ajacobi() # Berechnung der Jacobimatrix von A
    Xp = Prediction() #Vorhersage der Winkel

    #Fehler Kovarianz
    Pp = np.mat(A) * np.mat(P) * np.mat(A.transpose()) + np.mat(Q)
    #Berechnung der Kalman Verstärkung
    Kk = np.mat(Pp) * np.mat(H.transpose()) * inv(np.mat(H) * np.mat(Pp) * np.mat(H.transpose()) + np.mat(R))
    #Erstellen des Eingangsvektors mit Hilfe der Winkel vom Beschleunigunssensor und Magnetometer
    z = np.array([[Phi_Acc*(-1)], [Theta_Acc*(-1)],[(kompass*math.pi/180)]])
    #Berechnung der Ausgangswerte des Kalman Filters
    x = np.mat(Xp) + np.mat(Kk)*(np.mat(z) - np.mat(H) * np.mat(Xp))
    #Berechnung der Unsicherheit der Schaetzwerte x
    P = np.mat(Pp) - np.mat(Kk) * np.mat(H) * np.mat(Pp)

    return (x[0,0],x[1,0],x[2,0]) ##Rueckgabe der Ausgangswerte Kalman
```

Abbildung 7.11: Extended Kalman Filter Funktion

Jacobi-Matrix

Die AJacobi() Funktion berechnet die Jacobi-Matrix des Systemmodells und speichert dieses linearisierte Systemmodell in A ab. Im Kapitel 6.2.2 kann die Berechnung der Jacobi-Matrix nochmal nachgelesen werden.

```
def Ajacobi(): ##Zur berechnung der Jacobimatrix von A
    phi = x[0,0]
    theta = x[1,0]
    p = GYRx_cal
    q = GYRy_cal
    r = GYRz_cal

    A = np.zeros(shape=(3,3))
    A[0,0] = q * math.cos(phi) * math.tan(theta) - r * math.sin(phi) * math.tan(theta)
    A[0,1] = q * math.sin(phi) / math.cos(theta)**(2) + r * math.cos(phi) / math.cos(theta)**(2)
    A[0,2] = 0

    A[1,0] = -q * math.sin(phi) - r * math.cos(phi)
    A[1,1] = 0
    A[1,2] = 0

    A[2,0] = q * math.cos(phi) / math.cos(theta) - r * math.sin(phi) / math.cos(theta)
    A[2,1] = q * math.sin(phi) * math.sin(theta)/2 * math.cos(theta)**(2) + r * math.cos(phi)
             * math.sin(theta)/2 * math.cos(theta)**(2)
    A[2,2] = 0

    A = np.array([[1,0,0],[0,1,0],[0,0,1]]) + A * dTs
    return A
```

Abbildung 7.12: Berechnung der Jacobi-Matrix

Vorhersage-Funktion

Die Funktion Prediction() berechnet die Vorhersage der aktuellen Schätzwerte mit Hilfe der Daten der Gyroskope.

```
def Prediction(): ## Berechung der Vorraussage

    p = GYRx_cal
    q = GYRy_cal
    r = GYRz_cal
    phi = x[0,0]
    theta = x[1,0]

    xcast = np.zeros(shape=(3,1))

    #Berechnung der Eulerwinkel fuer die Vorhersage#
    xcast[0,0] = p + q*math.sin(phi)*math.tan(theta) + r*math.cos(phi)*math.tan(theta)
    xcast[1,0] =     q*math.cos(phi)                 - r*math.sin(phi)
    xcast[2,0] =     q*math.sin(phi)/math.cos(theta) + r*math.cos(phi)/math.cos(theta)

    Xp = x + xcast * dTs
    return Xp
```

Abbildung 7.13: Berechnung der Jacobi-Matrix

8 Evaluation

In diesem Kapitel werden einige Tests durchgeführt, um die Funktionsweise und die Ergebnisse des extended Kalman Filters zu überprüfen. Des Weiteren wird auf die Kovarianzmatrizen Q und R eingegangen, die dem extended Kalman Filter als Tuningparameter dienen. Die Abbildung 8.1 zeigt den Versuchsaufbau für die Tests. Dabei wurde der Drehteller auf die Seite gelegt, um eine Wellenbewegung zu simulieren.

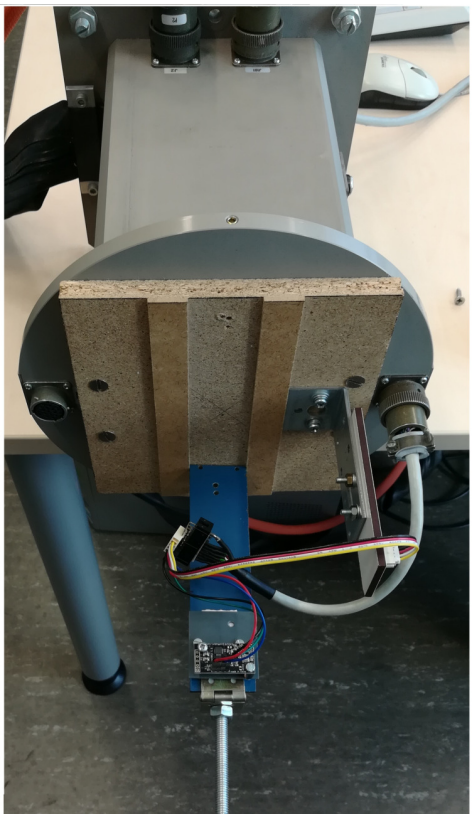

Abbildung 8.1: Versuchsaufbau zur Simulation von Wellenbewegungen

8.1 Testreihe: Nickwinkel θ

In diesem Test wird der Sensor wie in Abbildung 8.1 mit einem Rollwinkel von 0° montiert. Der Drehteller wird über die Softwareoberfläche so eingestellt, dass bei einer Frequenz von 0,75Hz eine Schwingung von +30° bis -30° Nickwinkel θ abgefahren wird. Durch mehrere Versuchsreihen wurden die Parameter für die Kovarianzmatrix Q ermittelt, um das optimale Ergebnisse des EKF zu erzielen. In der Kovarianzmatrix R befinden sich die in Kapitel 5 ermittelten Varianzen der Beschleunigungssensoren, Gyroskope und des Magnetometers.

$$R = \begin{bmatrix} 0,089 & 0 & 0 \\ 0 & 0,182 & 0 \\ 0 & 0 & 0,11 \end{bmatrix} \tag{8.1}$$

$$Q = \begin{bmatrix} 0,001 & 0 & 0 \\ 0 & 0,001 & 0 \\ 0 & 0 & 0,001 \end{bmatrix} \tag{8.2}$$

Die Abbildung 8.2 zeigt die Messwerte der Nickwinkel bei den oben genannten Parametern für Q und R.

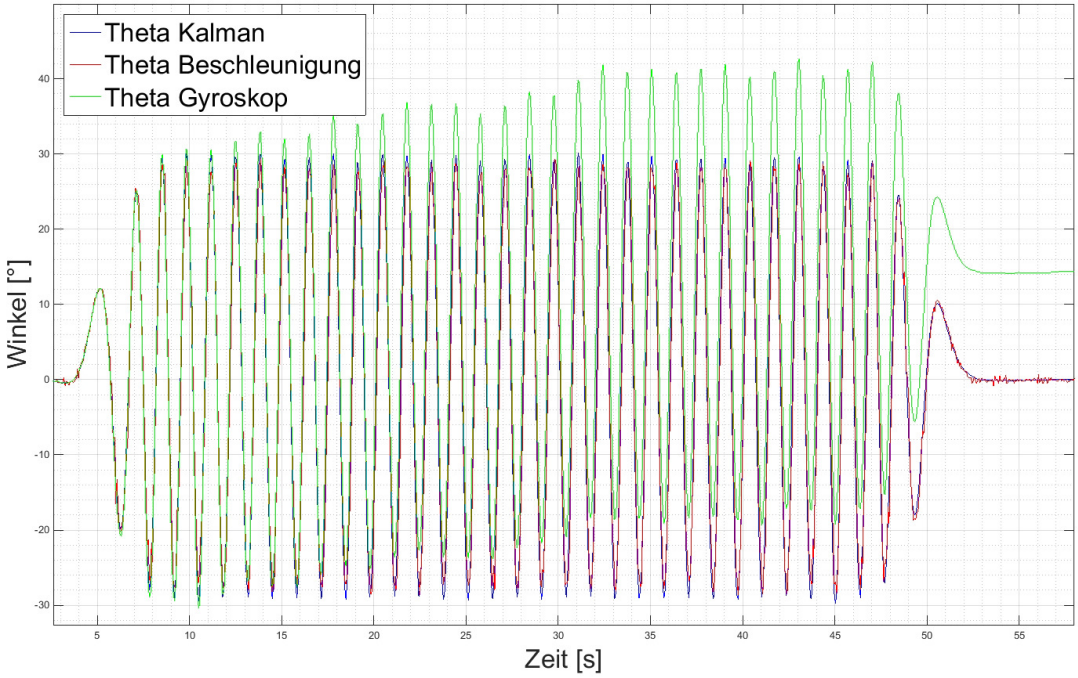

Abbildung 8.2: Simulation des Nickwinkels θ bei $\pm 30°$ bei 0,75Hz

Die Abbildung 8.2 zeigt den Nickwinkelverlauf der Beschleunigungssensoren, Gyroskope und dem des extended Kalman Filters. Es ist zu sehen, dass das Gyroskop einen positiven Drift aufweist und der Beschleunigungssensor bei dieser Dynamik nicht mehr den erwarteten Wert von 30° Nickwinkel erreicht. Am Ende dieser Messreihe ist zu sehen, dass das extended Kalman Filter sich dem Beschleunigungssensor anpasst und die Drift des Gyroskops keine Rolle spielt. Die Abbildung 8.3 zeigt einen hineingezoomten Ausschnitt der Messreihe, wie sie in Abbildung 8.2 zu sehen ist.

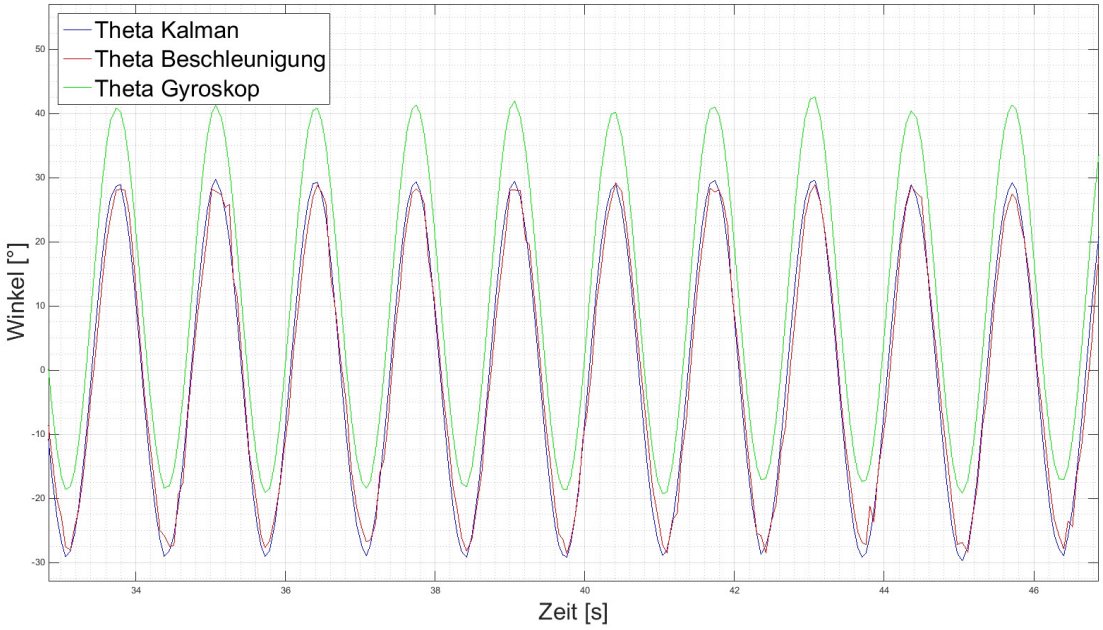

Abbildung 8.3: Ausschnitt der Simulation des Nickwinkels θ bei $\pm 30°$ bei 0,75Hz

In der Abbildung 8.3 ist zu sehen, dass die Beschleunigungssensoren aufgrund der hohen Dynamik nicht in der Lage sind $\pm 30°$ Nickwinkel zu erreichen. In diesem Moment legt das extended Kalman Filter mehr Gewichtung auf die Messwerte des Gyroskops und nährt sich den gewünschten $\pm 30°$ an. Der Nickwinkel θ des extended Kalman Filters übertrifft sowohl bei -30° als auch bei +30° die Werte der Beschleunigungssensoren. In den Sekunden zwischen 42 und 45 ist bei den +30° Werten, zu sehen wie das Gyroskop einen negativen Drift von 2° nach unten hat. Dort schafft es auch der Algorythmus des extended Kalman Filters nicht den Messwert des Beschleunigungssensors zu übertreffen.
Im Mittel erreicht der EKF, bei einem am Drehteller eingestellt Nickwinkel von +30°, einen Wert von 29,57°. Bei eingestellten -30° erreicht der EKF einen Wert von -29,15°, was auf den positiven Drift des Winkels der Gyroskope zurückzuführen ist.

Die Abbildung 8.4 zeigt eine Messreihe mit einem Nickwinkel von ±30° bei einer Frequenz von 1Hz. Es sind die folgenden Parameter für die Kovarianz Matrizen Q und R gewählt worden:

$$R = \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix} \tag{8.3}$$

$$Q = \begin{bmatrix} 0,0001 & 0 & 0 \\ 0 & 0,0001 & 0 \\ 0 & 0 & 0,0001 \end{bmatrix} \tag{8.4}$$

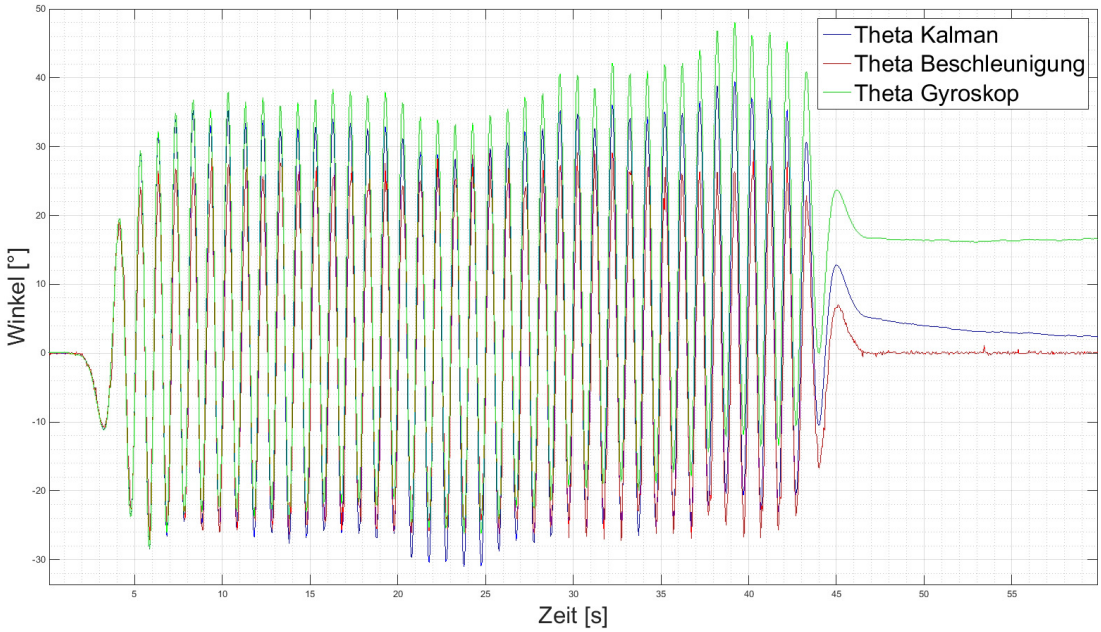

Abbildung 8.4: Simulation des Nickwinkels θ bei ±30° bei 1Hz mit falschen Parametern für Q und R

Die Abbildung 8.4 zeigt, dass in der Messreihe die Parameter falsch gewählt sind. Das extended Kalman Filter ist, durch die Parameter von Q und R so eingestellt, dass die Messwerte der Gyroskope zu hoch gewertet werden, beziehungsweise die aktuelle Messung der Beschleunigungssensoren nicht genug Einfluss hat. Es ist zu sehen, dass Theta des EKF zu sehr den Messwerten der Gyroskope folgt. Dies hat zur Folge, dass der Drift der Gyroskope Fehler erzeugt. Am Ende der Messreihe ist zu sehen, dass der Nickwinkel Theta des EKF sich nicht wieder dem Wert der Beschleunigungssensoren annähert, sondern einen positiven Offset auf-

weist. Die Abbildung 8.5 zeigt die Messreihe für den Nickwinkel bei $\pm$ 20° mit einer Frequenz von 0,25Hz und einem Rollwinkel ϕ von +10°. Diese Simulation zeigt ein Boot welches einen Rollwinkel von +10° besitzt und durch Wellen fährt, welche eine Frequenz von 0,25Hz haben, dies bedeutet, dass der Wellenkamms alle 4 Sekunden überwunden wird. Dabei wird das Boot von den Wellen auf Nickwinkel von $\pm$20° gezwungen.

$$R = \begin{bmatrix} 0,089 & 0 & 0 \\ 0 & 0,182 & 0 \\ 0 & 0 & 0,11 \end{bmatrix} \tag{8.5}$$

$$Q = \begin{bmatrix} 0,001 & 0 & 0 \\ 0 & 0,001 & 0 \\ 0 & 0 & 0,001 \end{bmatrix} \tag{8.6}$$

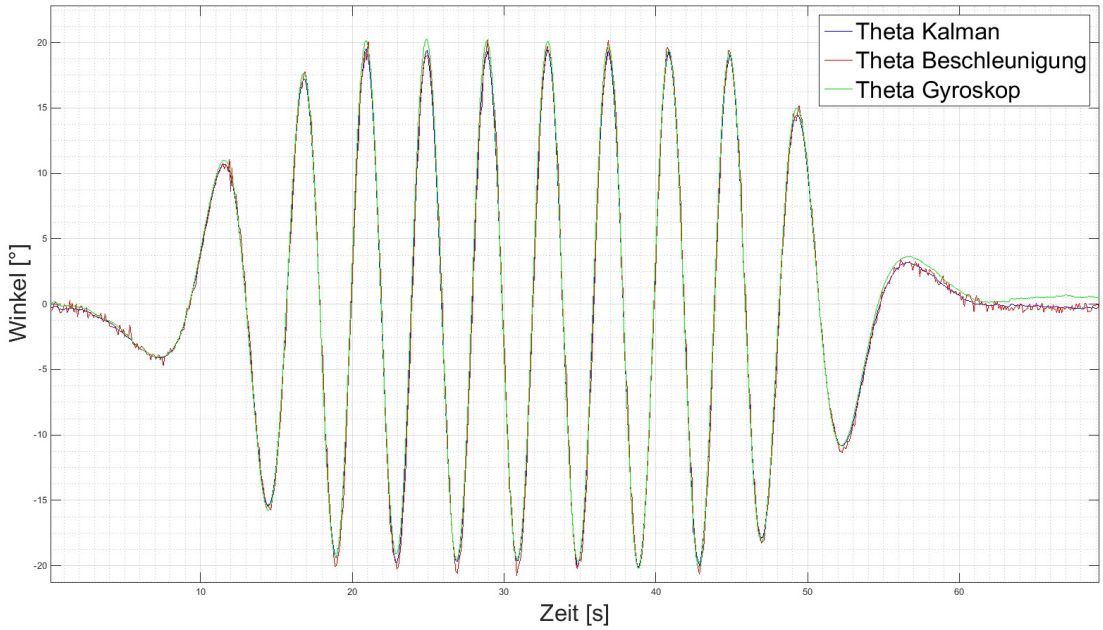

Abbildung 8.5: Simulation des Nickwinkels θ bei $\pm$20° bei 0,25Hz mit einem Rollwinkel von 10°

Die Abbildung 8.5 zeigt, dass ein Rollwinkel der Sensoren keinen Einfluss auf den Nickwinkel hat. Es beweist, dass die Berechnung mit Hilfe von Euler funktioniert. Des Weiteren ist zu sehen, dass das Rauschen des Beschleunigungssensors durch den extended Kalman Filter-Algorithmus gemindert wird.

8.2 Testreihe: Rollwinkel ϕ

Die Testreihe für den Rollwinkel ϕ wird wie die Testreihe für den Nickwinkel θ durchgeführt. Der Unterschied ist, dass der Sensor um 90 Grad gedreht wird, um mit Hilfe des Drehtellers den Rollwinkel des Bootes zu simulieren. Der Drehteller wird, mit einer Frequenz von 0,75Hz, auf $\pm 30°$ eingestellt.

$$R = \begin{bmatrix} 0,089 & 0 & 0 \\ 0 & 0,182 & 0 \\ 0 & 0 & 0,11 \end{bmatrix} \tag{8.7}$$

$$Q = \begin{bmatrix} 0,001 & 0 & 0 \\ 0 & 0,001 & 0 \\ 0 & 0 & 0,001 \end{bmatrix} \tag{8.8}$$

Die Abbildung 8.6 zeigt die Messwerte für den Rollwinkel bei den oben genannten Parametern für Q und R.

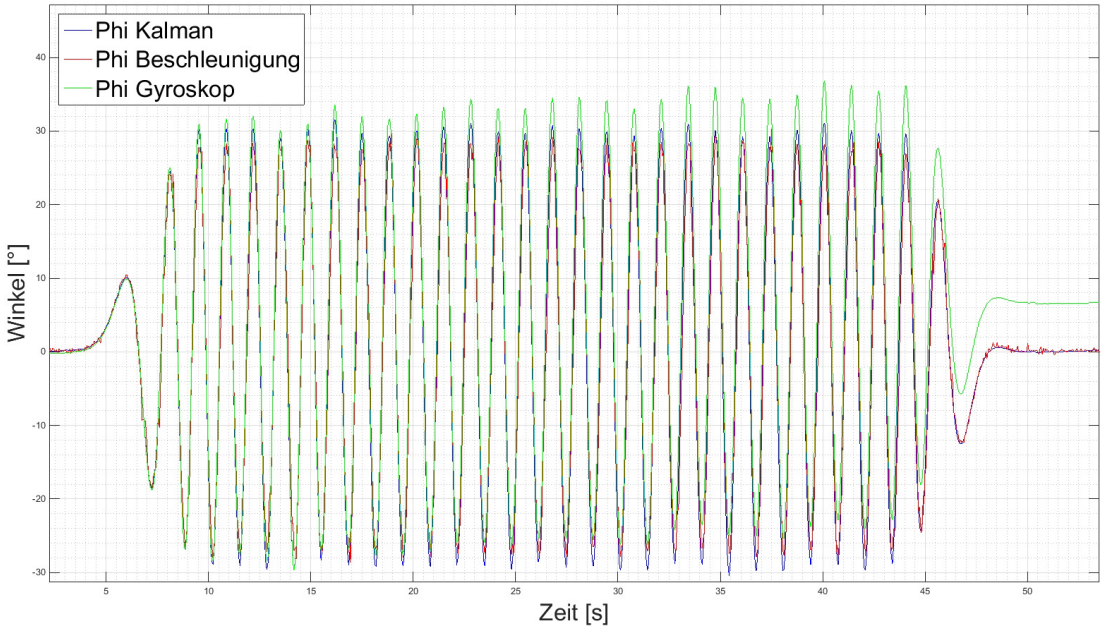

Abbildung 8.6: Simulation des Rollwinkels ϕ bei $\pm 30°$ bei 0,75Hz

In der Abbildung 8.6 ist der Verlauf des Rollwinkels der Beschleunigungssensoren, Gyroskope und des extended Kalman Filters zu sehen. Wie in der Testreihe für den Nickwinkel ist auch bei dem Rollwinkel ein positiver Drift der Gyroskope zu sehen. Dieser positive Drift kann in der nächsten Messung negativ sein, was auf den stochastischen Anteil des Bias zurückzuführen

ist. Die Messung zeigt einen guten Verlauf des Rollwinkels des extended Kalman Filters. Die erwarteten Messwerte von ±30° werden annähernd erreicht. Am Ende der Messung, welches ein Boot in Ruhelage simuliert, verlässt sich das EKF wieder auf die guten Langzeiteigenschaften der Beschleunigungssensoren. In der Abbildung 8.7 ist ein Ausschnitt der Messreihe aus Abbildung 8.6 zu sehen.

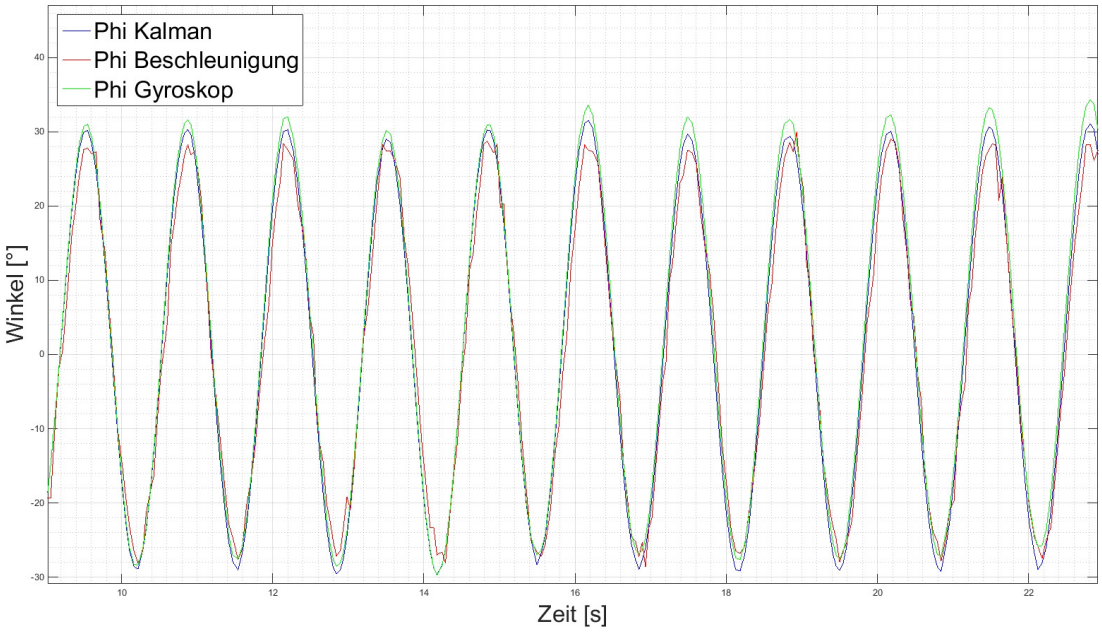

Abbildung 8.7: Ausschnitt der Simulation des Rollwinkels ϕ bei ±30° bei 0,75Hz

In Abbildung 8.7 ist zu sehen, dass die Beschleunigungssensoren nicht in der Lage sind die ±30° zu erreichen. Dies wird durch die guten Eigenschaften bei hohen Dynamiken der Gyroskope ausgeglichen. Es ist ein guter Verlauf des Rollwinkels des extended Kalman Filters zu sehen. Die erwarteten ±30° werden mit geringen Abweichungen erreicht. Zwischen 15 und 17 Sekunden ist zu sehen, wie der Rollwinkel der Gyroskope einen positiven Drift von über 2° hat. Dies hat auch einen negativen Einfluss auf die Messwerte des Kalman Filters. Statt der erwarteten -30° wird ein Wert von -28,32° gemessen. Bei erwarteten +30° wird ein Wert von 31,53° gemessen.

Im Mittel erreicht der extended Kalman Filter, bei einem am Drehteller eingestellt Rollwinkel von +30°, einen Wert von 30,17°. Bei eingestellten -30° ergibt sich ein Wert von -29,37°.

8.3 Testreihe: Gierwinkel ψ

Die Aufnahme einer Messreihe des Gierwinkels mit dem Drehteller ist nicht möglich. Der Grund dafür ist der Standort des Drehtellers, welcher sich in einem Gebäude befindet. Durch die Bestandteile des Gebäudes, kann das Erdmagnetfeld nicht störungsfrei gemessen werden. Des Weiteren erzeugt der Drehteller Störungen für die richtige Funktion des Magnetometers, welche durch den metallischen Aufbau des Drehtellers entstehen sowie durch den Elektromotor selbst.

Testaufbau

Um den Gierwinkel auf dessen Richtigkeit zu prüfen , wird der Sensor auf eine Drehscheibe montiert und ausgerichtet. Außerhalb der Drehscheibe befindet sich eine Schablone mit den Himmelsrichtungen und den Winkeln in 45°-Schritten. Der Test findet im Freien statt, um mögliche Fehlerquellen auszuschließen. Die Drehscheibe selbst besteht aus Acrylglas um eine Störung des Magnetometers auszuschließen. Die Drehscheibe wird dann immer von 0° ausgehend auf den gewünschte Winkel gedreht und die Messwerte aufgenommen. Dadurch wird ein Drift über die Zeit des Gyroskops minimiert.

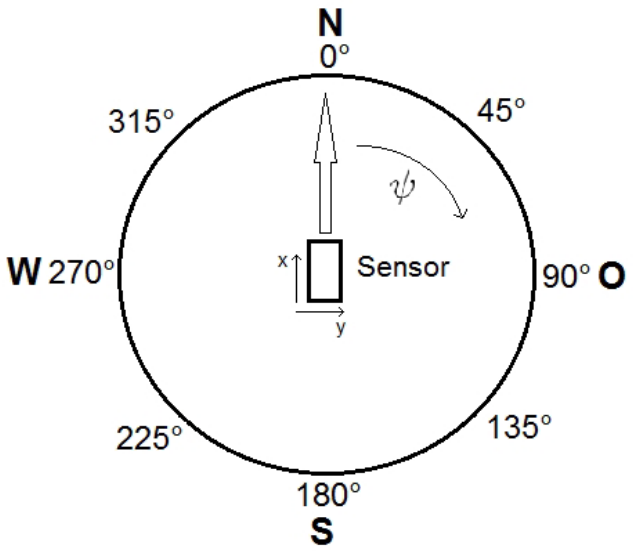

Abbildung 8.8: Schematischer Testaufbau zur Messung des Gierwinkels

Die folgende Tabelle zeigt die Testergebnisse für die Messung der Gierwinkel. Aufgenommen wurden jeweils 50 Messwerte, aus denen der Mittelwert gebildet wurde. Es wurden die selben Parameter für Kovarianz Matrizen Q und R verwendet wie bei den Tests des Nick- und Rollwinkels.

Soll	Ist Kalman [°]	Ist Gyroskop [°]	Ist Magnetometer [°]
0	0.74	0,12	0.86
45	45.73	45.52	45.76
90	91,43	90,84	91,46
135	136,57	135,39	136,76
180	180,89	180,56	181.03
225	227,03	225,97	227,13
270	269,04	271,79	268,74
315	317,22	316,72	317,46

Tabelle 8.1: Testreihe des Gierwinkels

Wie in der Tabelle 8.1 zu sehen ist, befinden sich die Ist-Werte des extended Kalman Filters relativ nah an den Soll-Werten. Des Weiteren ist zu sehen, dass sich der EKF, wie bei einer stationären Messung erwartet, an das Magnetometer anpasst. Für eine Kursnavigation sind die Messwerte des extended Kalman Filters gut geeignet, da eine geringe Abweichung des Kurses keinen großen Einfluss auf das Schleppangeln hat. Für eine Positionsbestimmung, welche mit Hilfe der Beschleunigungswerte jeder Achse erfolgen kann, sind Abweichung nach einer gewissen Zeit zu erwarten. Deshalb ist eine Positionsbestimmung durch den Sensor alleine nicht geeignet.

Nachdem die statischen Werte des Magnetometer aufgenommen sind, ist eine weitere Untersuchung des Magnetometers bei hohen Dynamiken nötig. Getestet wird der Verlauf des Gierwinkels Ψ des Magnetometers bei schnellen Bewegungen im Vergleich zum extended Kalman Filter und den Messwerten der Gyroskope. Für den Test wird der Sensor mit der Hand schnell in verschiedene Richtungen gedreht und die Messwerte aufgenommen.
Die Abbildung 8.9 zeigt den Verlauf der Kennlinien des Magnetomters, des EKF und der Gyroskope.

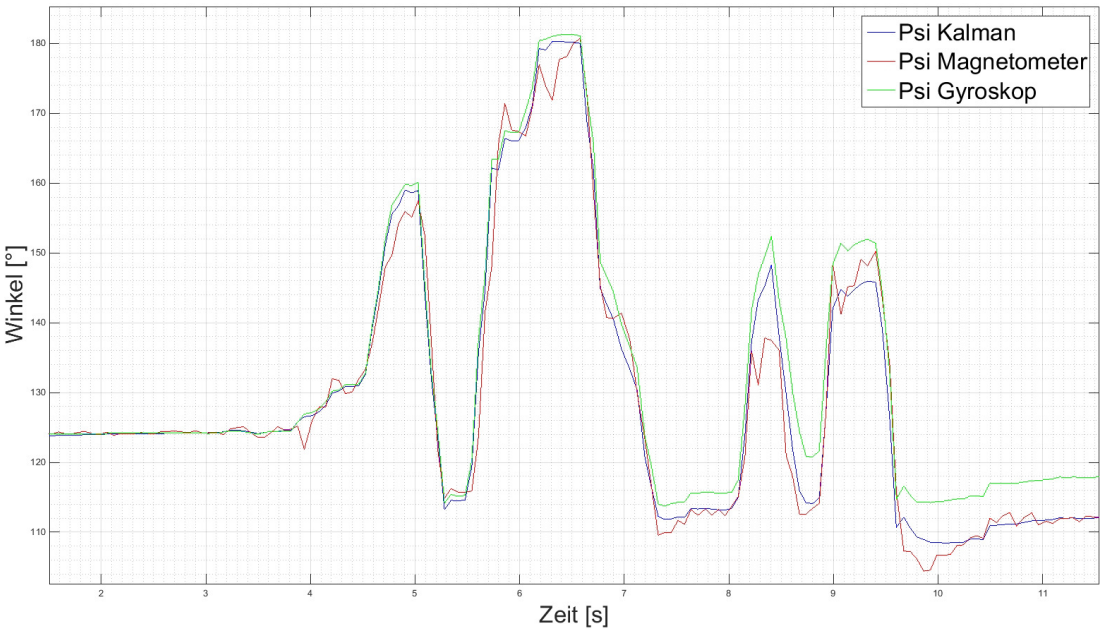

Abbildung 8.9: Verhalten des Magnetometers bei hoher Dynamik.

Abbildung 8.9 zeigt, dass das Magnetometer Fehler bei hohen Dynamiken, welche durch schnelle Bewegungen entstehen, besitzt. Es ist zu sehen, dass nach schnellen Bewegungen Über- und Unterschwinger entstehen die einen Wert von bis zu 5° haben. Das Magnetometer braucht zum Teil 0,5 Sekunden um sich wieder auf den richtigen Wert einzustellen. Die Abbildung zeigt, dass das Magentometer keine Verlässlichen Werte bei hohen Dynamiken liefert. Wie gewünscht, verlässt sich das extended Kalman Filter bei schnellen Bewegungen auf die Messwerte der Gyroskope und über einen längeren Zeitraum, im eingeschwungenen Zustand, wieder auf das Magnetometer.
In der Praxis spielt das schlechte Verhalten des Magnetometers, bei hohen Dynamiken, keine große Rolle. Eine so schnelle Gierwinkeländerung ist Aufgrund der zu bewegenden Masse und dessen Trägheit nicht möglich.

9 Fazit und Ausblick

9.1 Fazit

In dieser Masterarbeit wurde die Lagebestimmung eines Bootes mit Hilfe des BerryIMU Sensors realisiert. Die Bestimmung der einzelnen Winkel geschah mit Hilfe von Beschleunigungssensoren, Gyroskopen und Magnetometern, welche sich auf dem Sensor BerryIMU befinden. Es wurde ein Hardware Prototyp gebaut, welcher an Bord des Bootes montiert und durch die 12V Bordspannung betrieben werden kann. Für die Verarbeitung und Kommunikation mit den Sensoren wurde eine Raspberry PI verwendet, welcher über ein I^2C Bus die Daten des Sensors erhält.
Um eine gute Lagebestimmung zu realisieren, wurden die Fehlereigenschaften der Beschleunigungssensoren, Gyroskope und Magnetometer analysiert. Anschließend wurden die Beschleunigungssensoren und die Gyroskope mit Hilfe eines Aufbaus auf einem Drehteller vermessen und anhand dieser Daten kalibriert.
Da Beschleunigungssensoren und Magnetometer langzeitstabil sind, aber bei hohen Dynamiken nicht die gewünschten Winkel erreichen und Gyroskope zwar bei hohen Dynamiken gute Daten liefern, aber nach kurzer Zeit anfangen zu driften, ist es notwendig die guten Eigenschaften des jeweiligen Sensortyps zu nutzen. Dies geschieht mit Hilfe des extended Kalman Filters. Mit dem extended Kalman Filter wurden die Sensordaten fusioniert, um die Vorteile der einzelnen Sensortypen zu nutzen.
Gemessene Testreihen der Roll-, Nick- und Gierwinkel haben die richtige Funktion des implementierten extended Kalman Filters bestätigt. Die Testreihen zeigen das erwünschte Verhalten des extended Kalman Filters. Es ist zu sehen, dass sich das extended Kalman Filter bei hohen Dynamiken auf die Daten der Gyroskope verlässt und sich im statischen Zustand des Bootes auf die guten Langzeitstabilitäten der Beschleunigungssensoren und Magnetometer bezieht. Die Ergebnisse zeigen, dass eine echtzeit Lagebestimmung mittels Kalmanfusionierung der Messdaten zielführend ist.
Für eine ordnungsgemäße Funktion des extended Kalman Filters wurden die Parameter für die Fehler-Kovarianzen Q und R durch mehrere Testreihen ermittelt. Es wurde gezeigt was passiert

wenn falsche Parameter für die Tuningparameter Q und R gewählt wurden. Die Parameter für Q und R müssen möglicherweise in der praktischen Anwendung angepasst werden.

9.2 Ausblick

Die im Zuge dieser Arbeit gewonnenen Erkenntnisse und Ergebnisse können in späteren Arbeiten oder Projekten verwendet werden.

Eine Möglichkeit besteht darin, einen Kurshalteassistenten auf einem Boot zu realisieren. Mit Hilfe des in dieser Arbeit tiltkompensierten Gierwinkels, kann so ein gewünschter Kurs gewählt und gehalten werden. Durch einen Linearmotor und einer Steuerplatine, welche an den Raspberry PI angeschlossen werden kann, lassen sich Kursabweichungen korrigieren.

Auch eine Positionsbestimmung mit Hilfe der Lagewinkel und der translatorischen Bewegung, welche mit dem Beschleunigungssensor gemessen werden kann, ist möglich. Da diese Positionsbestimmung durch kleine Ungenauigkeiten über die Zeit Abweichungen von der Realität aufweist, kann die Position durch die Daten eines GPS-Empfängers erweitert werden. Mit Hilfe einer Steuerung durch einen Linearmotor kann so eine automatische Navigation realisiert werden.

Da die Sensoren bei einer Temperatur von 21°C vermessen wurden, können die Kennlinien in einer praktischen Anwendung abweichen. Hier könnten die Kennlinien für unterschiedliche Temperaturen aufgenommen und temperaturbedingte Abweichungen durch den internen Temperatursensor in die Implementierung aufgenommen werden.

Literaturverzeichnis

Downey, A. B. (2014). *Programmieren lernen mit Python*. 2. Auflage, Köln: O'Reilly Verlag GmbH & Co. KG.

Elektronik Kompendium(o. J.). *UDP - User Datagram Protocol*. https://www.elektronik-kompendium.de/sites/net/0812281.htm (15.03.2018).

Fraden, J. (2010). *Handbook of Modern Sensors. Physics, Designs, and Applications*. 4. Auflage, Heidelberg: Springer.

Haji, F. (2008). *Das Python-Praxisbuch - Der große Profi-Leitfaden für Programmierer (Open Source Library)*. München: Addison-Wesley Verlag.

Kim, P. (2011). *Kalman Filter for Beginners with Matlab Examples*. Korea: A-JIN Publishing company.

Kittel, A. (2005). *Physikalische Messtechnik*. Oldenburg: Universität Oldenburg. Vorlesungsskript.

Krohn, A.; Beigl, M.; Decker, C.; et al. (2004). *Inexpensive and Automatic Calibration for Acceleration Sensors*. https://www.teco.edu/~krohn/krohn_accl.pdf (12.04.2018).

Kofler, M.; Kühnast, C.; Scherbeck, C. (2017). *Raspberry Pi: Das umfassende Handbuch*. 4 Auflage, Bonn: Rheinwerk Computing.

Peters, R. (2013). *Entwicklung einer grafischen Bedienoberfläche zur Steuerung eines dynamischen Prüfstandes,*. Hamburg: Hochschule für Angewandte Wissenschaften. Bachelorthesis.

Phidgets (2017). *Magnetometer Primer*. https://www.phidgets.com/docs/Magnetometer_Primer (15.04.2018).

Sensitec Körber Solutions (2016). *AA700 Familie. AMR FreePitch Sensoren für Winkel-und Längenmessung*. https://www.sensitec.com/fileadmin/sensitec/

Service_and_Support/Downloads/Product_Information/Application_Notes/SENSITEC_AA700_AND_01.pdf (12.04.2018).

Sensitec Körber Solutions (2010). *An3192 Application Note. Using LSM303DLH for a tilt compensated electronic compass.* https://www.st.com/content/ccc/resource/technical/document/application_note/e6/f0/fa/af/94/5e/43/de/CD00269797.pdf/files/CD00269797.pdf/jcr:content/translations/en.CD00269797.pdf (13.04.2018).

STMicroelectronics (2013). *LSM9DS0. iNEMO inertial module: 3D accelerometer, 3D gyroscope, 3D magnetometer.* https://www.mouser.de/datasheet/2/389/lsm9ds0-955023.pdf (13.02.2018).

Titterton, D.; Weston, J. (2004). *Strapdown inertial Navigation Technology.* 2. Auflage, Herts (UK): The Institution of Electrical Engineers.

Tränkler, H.-R.; Reindl, L. M. (2014). *Sensortechnik. Handbuch für Praxis und Wissenschaft.* 2. Auflage, Berlin Heidelberg: Springer-Verlag.

Weigend, M. (2016). *Python 3: Lernen und professionell anwenden: Das umfassende Praxisbuch (mitp Professional.* 6. Auflage, Frechen: mitp-Verlag.

Wendel, J. (2011). *Integrierte Navigationssysteme. Sensordatenfusion, GPS und Inertiale Navigation.* 2. Auflage, München: Oldenbourg Wissenschaftsverlag GmbH.

Raspberry Pi Foundation (2018). *Raspberry Pi.* https://www.raspberrypi.org/blog/annual-review-2017/ (13.04.2018).

Anhang

Die verwendeten Datenblätter, Messwerte, Quellcodes sowie die angepasste Acutronic-Software befinden sich auf der beiliegenden CD.

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe.

Hamburg, 27. Juni 2018 Tobias Michaelsen