Inhaltsverzeichnis

[1 Aufgabenstellung 2](#_Toc367374569)

[1.1Problemstellung 2](#_Toc367374570)

[1.2 Aktuelle Situation 2](#_Toc367374571)

[1.3 Ziel der Bachelorarbeit 2](#_Toc367374572)

[2 Das Modell 3](#_Toc367374573)

[2.1 Mechanische Komponenten 3](#_Toc367374574)

[2.2 Elektronische Komponenten 3](#_Toc367374575)

[3 Entwicklung des Algorythmusses 4](#_Toc367374576)

[3.1Kompensationsansatz 4](#_Toc367374577)

[3.2 Entwicklung mit Blockschaltbildern 5](#_Toc367374578)

[3.3 Entwicklung der Regleralgorythmus 6](#_Toc367374579)

[4 Programmierung 8](#_Toc367374580)

[4.1 Main Funktion 8](#_Toc367374581)

[3.2 Funktion: Dateneinlesen 9](#_Toc367374582)

[4.3 Funktion: LCDansteuern 9](#_Toc367374583)

[4.4 Funktion: Ausgangansteuern 10](#_Toc367374584)

[4.4 Funktion: Test 11](#_Toc367374585)

[5 Zusammenfassung 11](#_Toc367374586)

[6 Literaturverzeichnis 12](#_Toc367374587)

# 1 Aufgabenstellung

## 1.1Problemstellung

Bei der Installation von Offshorewindanlagen ist es wichtig, dass eine an einem Kran hängende Last konstant in der gleichen Höhe über dem Meeresgrund gehalten werden kann. Die Position der Last ändert sich jedoch durch den Seegang konstant. Hierdurch wird eine Kompensation des Seegangs nötig.

Es gibt mehrere Möglichkeiten Seegang zu kompensieren:

- Eigenantrieb

- Rumpfform

- Stabilisatoren

- Position des Krans auf dem Schiff möglichst weit mittschiffs

- regelungstechnische Kompensation

Allen Kompensationsmöglichkeiten haben die Gemeinsamkeit, dass sie den Seegang nicht zu 100% kompensieren können.

## 1.2 Aktuelle Situation

Aktuell werden zur Seegangskompensation die pneumatischen Systeme der Firma ?????? verwendet. Hier ist mir über die regelungstechnische Umsetzung nichts Bekannt.

Des weiteren gibt es einen Ansatz mittels Prognosealgorithmen, siehe: Sebastian Küchler

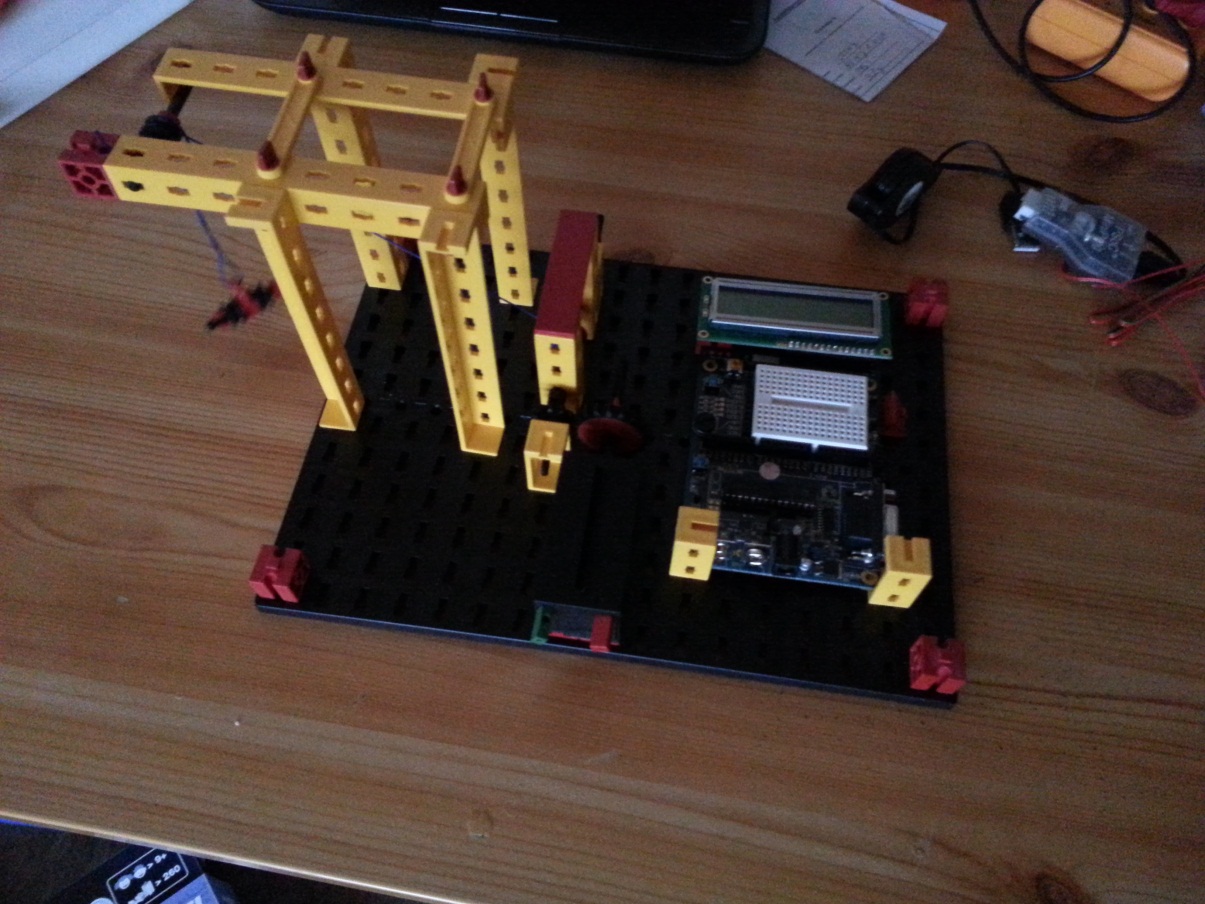
Aktive Seegangskompensation mit beobachtergestützter Prognose der vertikalen Schiffsbewegung

(Ausfürlicher Besser)

## 1.3 Ziel der Bachelorarbeit

Da im Rahmen dieser Bachelorarbeit nur eine reglungstechnische Möglichkeit zur Seegangskompensation gefunden werden soll, wird davon ausgegangen, dass alle durch andere Kompensationsmetoden das Schiff auf einer Position gehalten wird und sich nur noch in der Horizontahlen bewegt. Ziel der Bachelorarbeit ist es also, die durch Seegang verursachte Höhenänderung zu kompensieren. Dies soll ohne pneumatiche Komponenten und ohne komplizierte Prognosealgorithmen geschehen. (Vormulierung).Die Hauptstörgröße in Form der Höhenänderung und der Abstand zum Meeresgrund werden mit einem Sensor erfasst. Es soll nun ein Programm entwickelt werden, welches den Motor so ansteuert, dass die Last möglichst auf einer Position gehalten wird.

# 2 Das Modell

Da der Fachhochschule kein Schiff mit einem Krahn zur Verfügung steht musste ein Modell zu simulationszwecken gebaut werden.

## 2.1 Mechanische Komponenten

Das Modell wurde mithilfe eines Baukastens der Firma Fichertechnik gebaut. Motor und Controller und Sensoren sind in das Modell integriert. Um den Seegang zu simulieren wird das Modell mittels seilen und einer Umlenkrolle an der Decke Befestigt.

## 2.2 Elektronische Komponenten

Zur Regelung des Systems wird ein PSoC1 Evaluation Kit mit einem Prozessor vom Typ CY8C27446-24PXI verwendet. Auf dem mitgelieferten LCH-Display wird die Entfernung der Last zum Entfernungssensor angezeigt. Das System wird durch das Bord mit einer Versorgungsspannung von 5 Volt versorgt.

Zur Ansteuerung des Motors wird ein Treiberbaustein vom Typ L293 D verwendet dieser enthält 2 H-Brücken von denen eine zur Motorsteuerung verwendet wird. Durch eine Pulsweitenmodulation am Enable-Eingang wird die Drehzahl gesteuert.

Der Gleichstrommotor der Firma Fichertechnik benötigt eine Versorgungsspannung von 9 Volt daher muss zusätzlich zu der 5 Volt Spannungsversorgung das PSoCs eine 9V Spannungsquelle am Motortreiber angeschlossen werden.

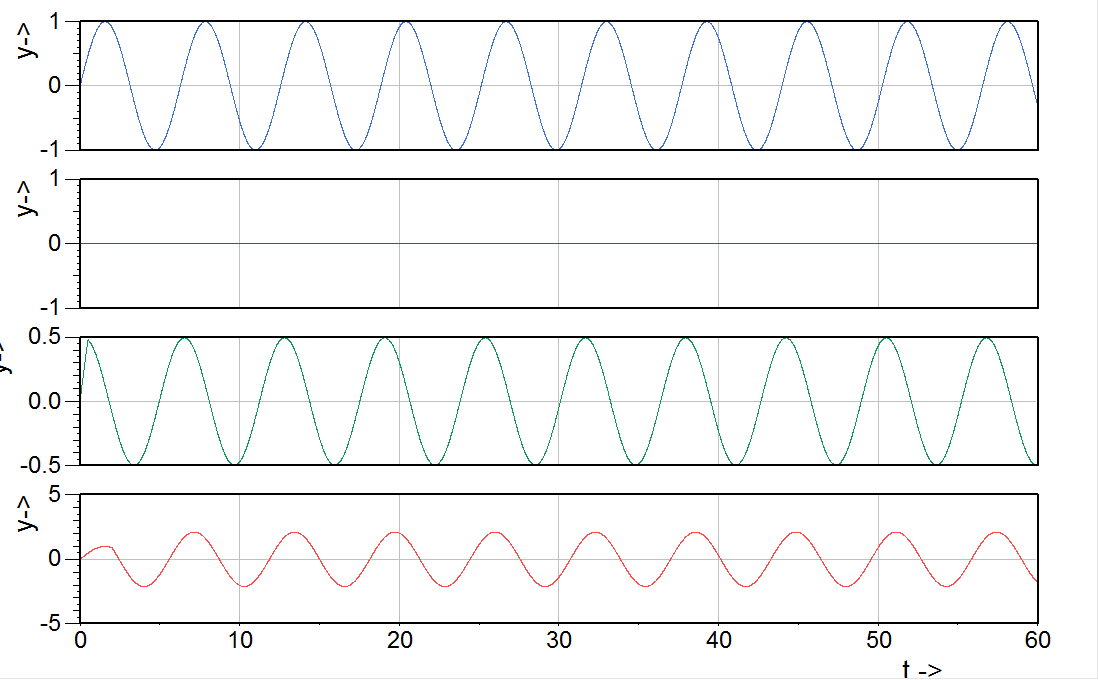
Infrarot und Beschleunigungssensor geben beide eine analoge Spannung aus. (Mehr Schreiben)

Der Schaltplan ist in Anhang ????? beigefügt.

# 3 Entwicklung des Algorithmus

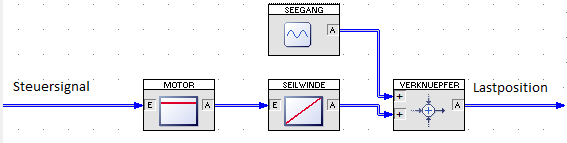
## 3.1 Kompensationsansatz

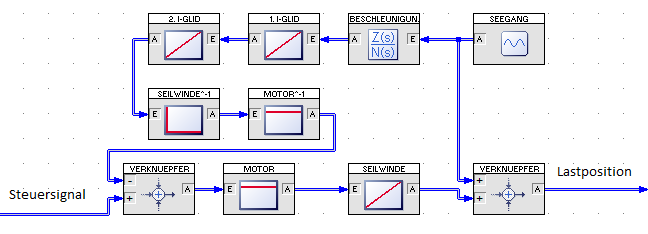
Um die durch den Seegang entstehende Höhenänderung zu kompensieren muss die Last um die gleiche Distanz und in entgegengesetzter Richtung bewegt werden. Zur einfachen Modellbildung wird erst einmal davon ausgegangen, dass die durch die Wellen hervorgerufene Höhenänderung Sinusförmig verläuft. Subtrahiert man nun zu einer sinusförmigen Störgröße (1.Kurve) ein weiteres Sinussignal mit gleicher Periodendauer und Amplitude, heben diese sich auf (2.Kurve). Da die Erfassung und Verarbeitung der Daten des Beschleunigungssensors eine gewisse Zeit dauert, kann der Seegang nicht zu 100% kompensiert werden. Durch diese Zeitverzögerung kommt es nun aber zu einer Phasenverschiebung zwischen dem zweiten Signal und der Störgröße. Dadurch wird die Störgröße nicht vollständig kompensiert (3.Kurve). Im schlimmsten Fall addieren sich beide Größen, sodass die Störgröße noch verstärkt wird (4.Kurve). Die durch den Seegang hervorgerufene Regeldifferenz wird also bei sinkender Verarbeitungszeit immer kleiner. Um eine Verstärkung der Störgröße zu vermeiden, darf die Verzögerungszeit maximal halb so groß sein wie die Periodendauer der Störgröße.



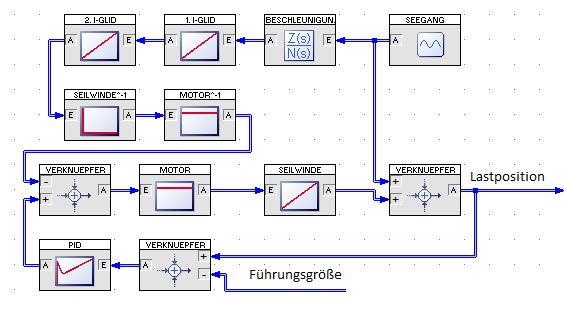
## 3.2 Entwicklung mit Blockschaltbildern

Grundlegend lässt sich das System ohne Regler durch folgendes Blockschaltbild darstellen:



Durch einen Beschleunigungssensor kann nun der Seegang gemessen werden. Um die durch den Seegang entstehende Höhenänderung zu bestimmen müssen die gemessenen Beschleunigungswerte zwei Mal Integriert werden. Würde man nun die Höhenänderung direkt an der Additionsstelle subtrahieren, könnte man den Seegang vollständig ignorieren. Da aber nur das Steuersignal manipuliert werden kann, muss die Subtraktion um zwei Glieder nach links verschoben werden und ein Schaltblock mit den Umkehrfunktionen von Motor und Seilwinde Eingefügt werden:

Zur weiteren Reduzierung der Regeldifferenz und um andere Störgrößen auszugleichen wird nun noch ein Regler hinzugefügt:



Man kann nun das D-Glied mit den beiden I-Gliedern zusammenfassen und Erhält ein einzelnes I-Glied:

Der Regler wird als P-Regler programmiert, da dieser an einer I-Strecke ein optimales Führungsverhalten aufweist (Hückelheim, Tabelle Reglerauswahl). (Begründung)



## 3.3 Entwicklung der Regleralgorithmus

Aus dem Blockschaltbild lässt sich folgende Gleichung bestimmen:

Beziehungsweise

Diese muss zum Einsatz im Programm noch modifiziert werden. (Schlechte Formulierung) Da die Regeldifferenz noch nicht bekannt ist, muss sie erst noch im Microcontroller berechnet werden:

Der Microcontroller arbeitet mit zeitdiskreten Signalen und die Gleichung muss entsprechend angepasst werden. Das Integral kann nun durch eine Summe ersetzt werden.

Da bei der Entfernungsmessung kein linearer Zusammenhang zwischen gemessener Entfernung und ausgegebener Spannung besteht, muss dies im Programm berücksichtigt werden. Der relevante Teil der Kurve zwischen 20 und 60 cm entspricht annähernd Funktion dritter Ordnung deren Parameter mit Hilfe des Programms skidavis bestimmt werden können. Die Funktion hat folgende Funktionsgleichung: (Formulierung)

De diese Gleichung relativ komplex ist kann man statt dessen auch die bestmögliche Gerade ermitteln. Man erhält dann folgende, einfacher zu berechnende Gleichung: (Formulierung erklären warum besser)

Setzt man diese Funktion in die Gleichung ein, so erhält man:

Im Programm wird die Gleichung dann in zwei Schritten berechnet. Zuerst wird die Summe bestimmt:

Anschließend wird die Ausgangs Größe bestimmt.

# 4 Programmierung

Zusetzlich zu den üblichen Konventionen wird folgendes Festgelegt:

-Komentare in Deutch

-Ungaricge Notation nach Charles Simonyi (Apps Hungarian)

Zum Versionsmanegement wird die Internettplattform GitHub benutzt. Dadurch ist jede Programmänderung nachvollziehbar und es kann jederzeit an eine früheren stand zurückgekehrt werden. Desweiteren exestiert immer eine Sicherheitskopie (Vormulierung).Das Projekt ist zu finden unter: <https://github.com/lesky/Batchlorarbeit--Seegangskompensation-bei-Krananlagen>

-Bugtracking in Readme

Das Bugtracking erfolgte direkt in der Readmee.txt des Projektes

-Präprozessor

-Globale Struktur

-Funktionsprototypen

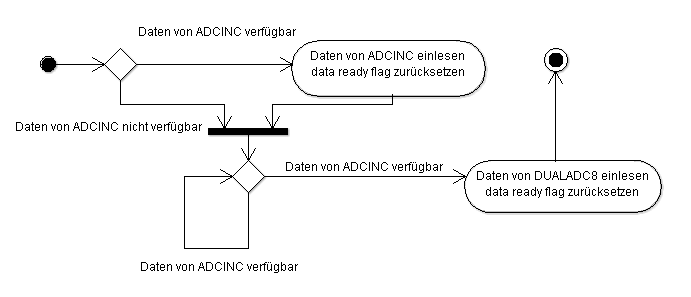
## 4.1 Main Function



Zu Beginn der Main Function werden sämtliche um Chipleveldeseing festgelegten User Modules initialisiert. Anschließend werden in einer Endlosschleife die Daten eingelesen, verarbeitet und ausgegeben.

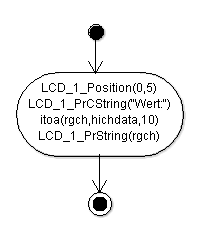
Zu Testzwecken wurde in der Main Function eine weitere Endlosschleife programmiert. Diese wird statt der oben genannten Schleife kompiliert, wenn der Präprozessor TEST definiert ist. Sie hat die Aufgabe, nacheinander die Daten einzulesen und auf dem LCD-Display auszugeben.

## 4.2 Funktion: Dateneinlesen

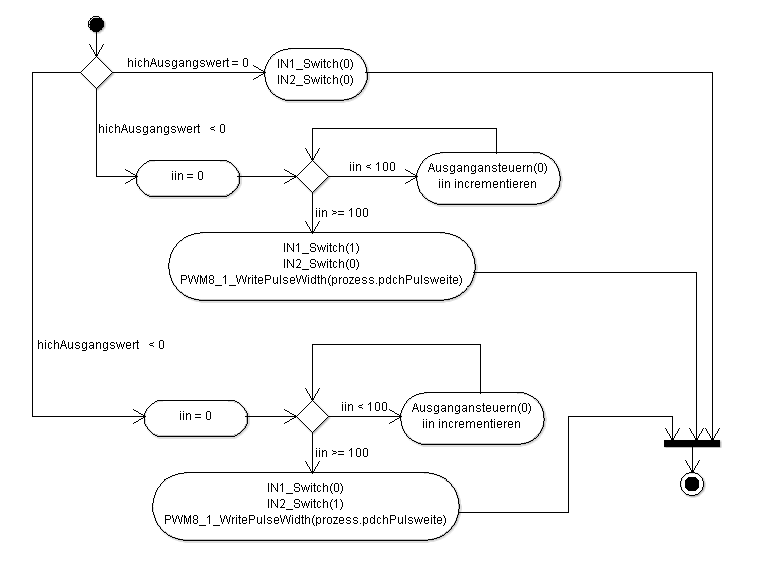


Das einlesend der Daten erfolgt in zwei Schritten. Zuerst wird überprüft ob die Daten des für den Sollwert zuständigen A/D-Wandlers verfügbar sind. Sind diese verfügbar werden sie eingelesen. Anschließend wird das data ready flag des A/S-Wandlers Zurückgesetzt. Da der Sollwert nicht Zeitkritisch ist wird hier nicht auf die Daten gewartet. Als nächstes werden die Beschleunigung und die Entfernung eingelesen. Sind diese Daten nicht verfügbar wird so lange gewartet bis die Daten eingelesen werden können. Anschließend wird auch hier das data ready flag zurückgesetzt und die Funktion verlassen.

## 4.3 Funktion: LCDansteuern

Diese Funktion gibt Eine Zahl vom Datentyp Charakter zusammen mit einem dem Text „Wert:“ auf dem LCD-Display aus. Zuerst wird der Text auf das Display Geschrieben. Da es nicht möglich ist die Zahl wie bei C Üblich direkt mittels %c mit dem Text auszugeben muss hierfür die Funktion iota aus der C Standard-Bibliothek stdlib.h genutzt werden. (Ausfürlicher)

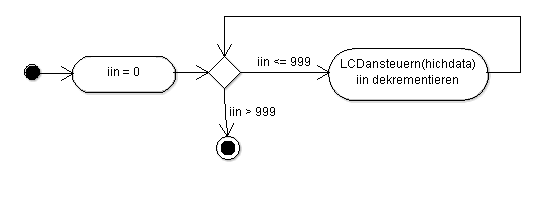
## 4.4 Funktion: Ausgangansteuern



|  |  |  |
| --- | --- | --- |
| IN1 | IN2 | Motor |
| 1 | 0 | Seil Abwickeln |
| 0 | 1 | Seil Aufwickeln |
| 0 | 0 | Bremsen durch Leerlauf |
| 1 | 1 | Bremsen durch Kurzschluss |

Die Funktion Ausgangansteuern Wertet den vom Algorithmus berechnete Ausgangsgröße aus und steuert den Motortreiber an. Durch die beiden Ausgänge IN1 und IN 2 wird die Drehrichtung des Motors bestimmt (siehe Tabelle). Sind beide Ausgänge 0 läuft der Motor im Leerlauf und bremst dadurch ab. Ändert sich die Drehrichtung erkennt die Funktion das und bremst den Motor vorher ab. Hierbei ruft die Funktion sich selbst in einer for-Schleife 100 mal auf. Da der Funktion in der Schleife der Wert 0 übergeben wird ist sichergestellt, dass auch die Ausgänge IN1 und IN2 mit dem Wert 0 angesteuert werden.

## 4.4 Funktion: Test



Die Funktion Test wird nur dann kompiliert, wenn der Präprozessor TEST definiert ist. Sie gibt die ihr bergebenen Daten mit Hilfe der Funktion LCDansteuern() direkt auf dem Bildschirm aus. Damit die Daten lange genug auf dem Display erscheinen um sie lesen zu können wird die Funktion LCDansteuern() in einer Schleife 999 mal ausgeführt bevor die Funktion verlassen wird

# 5 Zusammenfassung

-Ausblick Ideen zur verbesserung

Verbesserungsmöglichkeiten:

-Andere sensoren z.b. I2C

-Schnellerer Prozessor

-Schnellerer Motor

# 6 Literaturverzeichnis

(kein Datum). Abgerufen am 22. 8 2013 von rn-wissen: http://www.rn-wissen.de/index.php/Getriebemotoren\_Ansteuerung

Bosch Rexroth AG. (3 2012). Sicher durch Wellen Sekundärgeregelte Antriebe Kompensieren den Seegang. *drive&control*, S. 2-3.

Chacon, S. (2009). *Pro Git (Expert's Voice in Software Development).* http://git-scm.com/book.

Hückelheim, K. (kein Datum). *Elementare Übertragungsglieder*. Abgerufen am 12. 09 2013 von https://elearning.fh-frankfurt.de

Hückelheim, K. (kein Datum). *Tabelle Reglerauswahl*. Abgerufen am 12. 09 2013 von https://elearning.fh-frankfurt.de

Lehmann, H. L. (2008). *Taschenbuch der Elektrotechnik und Elektronik.* Münchem.

Merz, L. J. (1990). *Grundkurs der Reglungstechnik.* Oldenburg.

Papula, L. (2009). *Mathematiche Formelsammlung für Ingeniere und Naturwissenschaftler.*

Rabe, G. L. (kein Datum). *github*. Abgerufen am 28. 08 2013 von https://github.com/lesky/Weather-Station/