

# Gierregelung

Yawregelung





# Inhalt

**Umfang: ca. 1-4 Zeitstunden**

- Gierregelung
- Sicherheitseinweisung
- Theorie zur Regelung
- Superposition
- Aufgaben & Hinweise





# Gierregelung

## **Definition Gierregelung:**

Das Drehmoment eines jeden Motors bewirkt, dass der Quadrokopter um die eigene Achse (z-Achse) giert. Deshalb drehen stets zwei Motoren links und zwei rechts ´rum. Damit dies mit der Lageregelung nicht kollidiert, drehen stets die gegenüberliegenden Motoren in die selbe Richtung.

Mit Hilfe eines Gierreglers lassen sich dann Giermanöver ausführen bzw. Störungen ausregeln. Verglichen mit der Lageregelung ist diese Regelung meist deutlich einfacher, weil das System nicht gegen die Gravitation ankämpft und somit sich gutwilliger verhält.





# Sicherheitsbelehrung

## Sicherheitsbelehrung:

- Vor dem Start der Motoren stets prüfen, dass alles fest sitzt:  
Dies betrifft alle Schrauben, Motoren, Ausleger, Y-Teile, Rändelschrauben, etc.
- NUR mit Genehmigung des Betreuers die Motoren anschließen und starten.
- Abstand halten. Nachbarn warnen. Vorsicht walten lassen. Ausknopf parat (Finger drüber)!



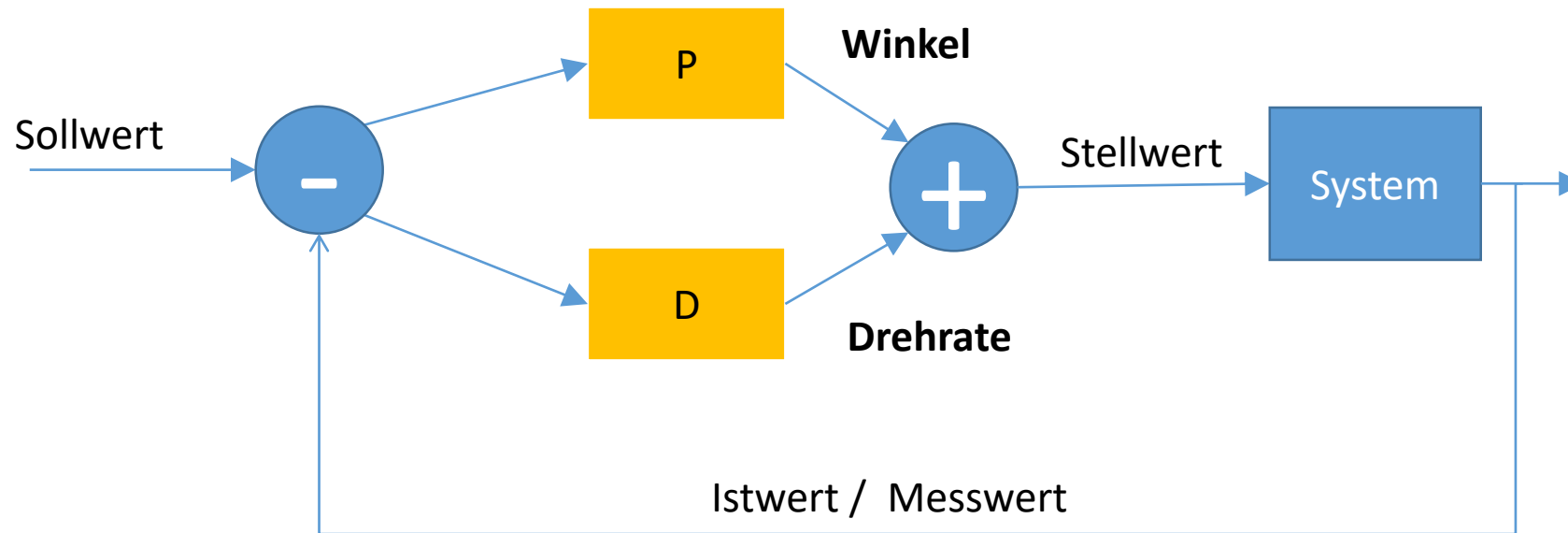
## Not-Aus:

- Spannungsquelle ausschalten bzw. trennen
- I2C Kabel trennen (herausziehen)
- MCU ausschalten



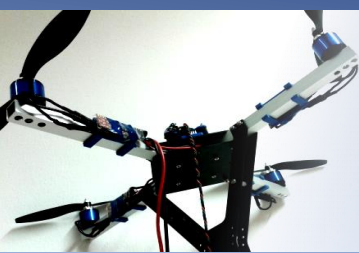
# Theorie zur Regelung

Mit einem PD-Regler und einem Gyrosensor lässt sich das System regeln. Als Regelgröße dient der Neigungswinkel.



Problem: 2 Motoren / Stellwerte , 1 Messwert (Winkel)

Lösung: Vereinfach durch Regelung der Differenz der Stellwerte beider Motoren



# Superposition

Superposition:

Bei der Superposition werden die Regler einfach additiv überlagert. Die Superposition geht dabei davon aus, dass die Regler unabhängig voneinander sind.

Ziel dieser Übung ist die Regelung eines Quadropters. Alle dazu benötigten Regler werden einzeln entwickelt. Zu Regeln sind die Freiheitsgrade RPY, damit der Quadropter in der Luft bleibt. Dies wird durch 2 (3) Regler realisiert:

- Attitude-Controller (x2 = Roll + Pitch)
- Yaw-Controller

Am Ende werden die beiden (drei) Regler einfach superpositioniert. Für den zweidimensionalen Fall (1 Gierregler, 1 Lageregeler) ergibt sich somit:

```
M1 = (unsigned char)(START_GAS + roll + yaw);  
M2 = (unsigned char)(START_GAS - yaw );  
M3 = (unsigned char)(START_GAS - roll + yaw);  
M4 = (unsigned char)(START_GAS - yaw);
```

## Beachte:

Die Superposition ist eine Vereinfachung. In der Realität beeinflussen sich die Regler. Daher ist bei der Realisierung noch mehr zu tun!



# Aufgaben

## Notwendige Hardware:

- EMQ3000
- Mini USB Kabel für Strom und zum Flashen
- Mini USB Kabel für Kommunikation: RS232 auf PC (RS232 oder USB)
- QCS im 1DOF Yaw Control Modus (später in 2 DOF umschaltbar)

## Notwendige Software:

- AVR Studio 32 installiert (mit Tool Chain und FLIP Treiber)
- EMQ Framework (Code)
- Dokumente:
  - EMQ\_Framework.pdf
- Qt SDK Version 4.8.1 oder 4.7.4.
- Quatplay Qt Framework (Code)





# Aufgaben

## **Aufgabe 1:**

Implementiere einen PD-Regler als Gierregler (engl.: Yaw Controller), der das Gieren des Quadropters regelt. Stell die Reglerparameter empirisch durch Ausprobieren ein. Der Regler muss in der Lage sein Stöße zu kompensieren.

Hinweis:

Durch Verwendung einer zusätzlichen Schraube kann der Quadcopter in der Waagerechten fixiert werden, so dass eine Lageregelung nicht erforderlich ist. Verwende diese Schraube und entferne sie für Aufgabe 2.

## **Aufgabe 2:**

Fusioniere nun die beiden Regler. Dabei sollte dir auffallen, dass sich die Regler beeinflussen. Das Gesamtverhalten darf sich trotzdem nicht Wesentlich verschlechtern. Insbesondere starke Yaw-Stöße sollen kompensierbar sein, ohne dass das System umkippt. Es ist wünschenswert, wenn das System weiterhin schnell gieren kann.





# Aufgaben

## Hinweise zur Regler-Dimensionierung:

Folgende Parameter/Vorgaben soll dir helfen: Nur Richtwerte als Größenordnungshilfe, ohne Sample-Time oder Skalierung:

Sample-Time:	10ms		
Bias-Samples	1000 bis 2000		
P	3	6	
D	0.5	2	(bremst P ab)
I (optional)	0.001	0.01	(vs. stationäre Regelabweichung)

## Vorgehen bei der Regler-Parametrisierung:

Fange mit  $D = 0$  und  $I = 0$  an und erhöhe  $P$  solange, bis das System „regelt“. Wenn es instabil wird, ist  $P$  bereits zu hoch. Wenn das System den Sollwert nie erreicht, ist  $P$  zu niedrig. Im Normalfall sollte das System oszillieren, d.h. hin und her schlagen. Als Nächstes erhöhe  $D$  um ein Überschwingen abzdämpfen.  $D$  sollte der Bewegung (Änderung) entgegenwirken.

## Idee:

Der  $P$ -Anteil gibt an, wie schnell/heftig auf einen Fehler reagiert wird. Mit dem  $D$ -Anteil kann dann das System gebremst werden (Gegenregelung), wenn z.B. ein Fehler auftritt aber das System bereits in Bewegung ist.