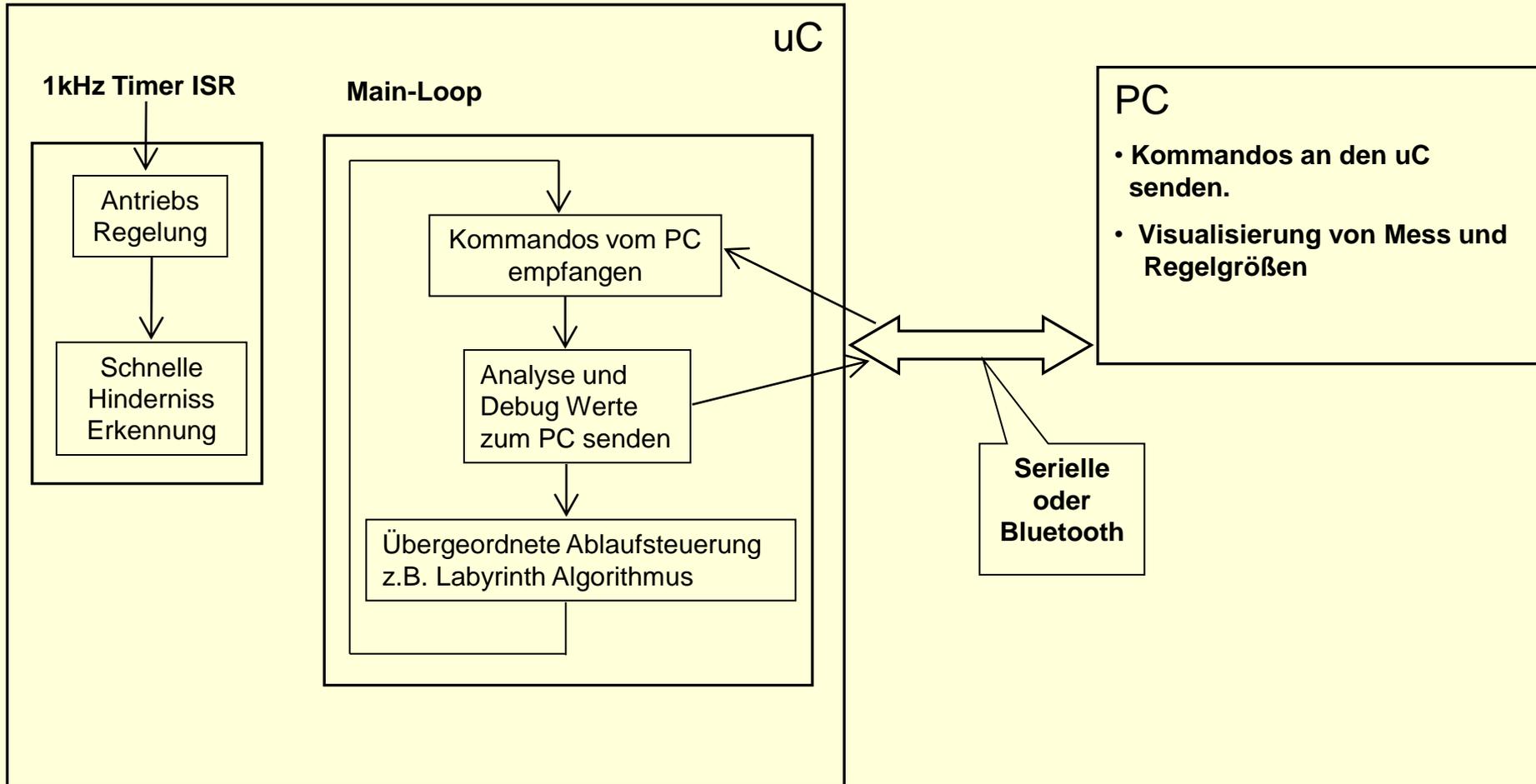


Regelungstechnik
in
Robotik Anwendungen

Vorgestellte Beispiele

- Framework (Library) zur Visualisierung und Analyse von Mess und Regelgrößen mit bis zu 2KHz.
Die Library und die dazugehörige Graphische Oberfläche werden in allen Regelungs und Robotik Anwendungen verwendet.
- Drehzahlregelung eines Gleichstrommotors
- Positionsregelung (Servo-Regelung) eines Gleichstrommotors
- 3 Achsen Lage Regelung/Stabilisierung eines Quadrocopters

SW-Struktur von embedded Robotik Anwendungen



Die Standard Oberfläche zur Visualisierung von Messwerten und zum Senden von Kommandos.

Wird unverändert zur Analyse der verschiedensten Robotik-Anwendungen:
(Bulgati, WallFollower, MicroMouse, Quadrocopter, Liniensensor...) verwendet.
Angepasst wird lediglich ein Config-File mit Messkanal-Namen und Skalierungen

Screenshot von SvVis3

Anforderungen an das Prozessvisualisierungs-Framework (Library)

- Einfaches verständliches Datenübertragungsprotokoll.
Keine komplizierten Überbauten wie Checksum und Retransmission.
- Durchsatz bei 115K-Baud:
1 Kanal mit 2KHZ; oder 10 Kanäle mit 100Hz
- Durchsatz kann mit FTDI auf 2MBit, und mit Ethernet auf 10/100? Mbit gesteigert werden.
- Die Übertragung der Analysedaten kann vom PC aus (remote) Ein und Ausgeschaltet werden.
- Default (wenn der Quadrocopter fliegt) sind alle Analysekanäle ausgeschaltet.
- Erst wenn der Quadrocopter in Bluetooth-Reichweite ist, oder USB angesteckt ist wird von der PC-Oberfläche aus die Datenübertragung eingeschaltet.

Das Übertragungsprotokoll

- ChanId,Wert; ChanId,Wert; ChanId,Wert; ChanId,Wert; . . . SyncPattern
ChanId,Wert; ChanId,Wert; ChanId,Wert;
- **ChanId:** 1Byte
In ChanId ist der Datentyp und die Kanalnummer kodiert.
- Datentypen:
int16: 2Byte
float: 4Byte
string: 0-terminierter C-String
- Der Datentyp string ist vor allem für Meldungen aus dem embedded-System in einer Meldungsbox in SvVis3 wichtig.
Meldungen als Quittierung von Kommandos wie z.B.
“Nullpunkts-Kalibrierung OK“
- Minimale Byteanzahl für einen Messwert sind 3-Byte: 1Byte ChanId, 2 Byte Value

2 einfache Visualisierungsbeispiele

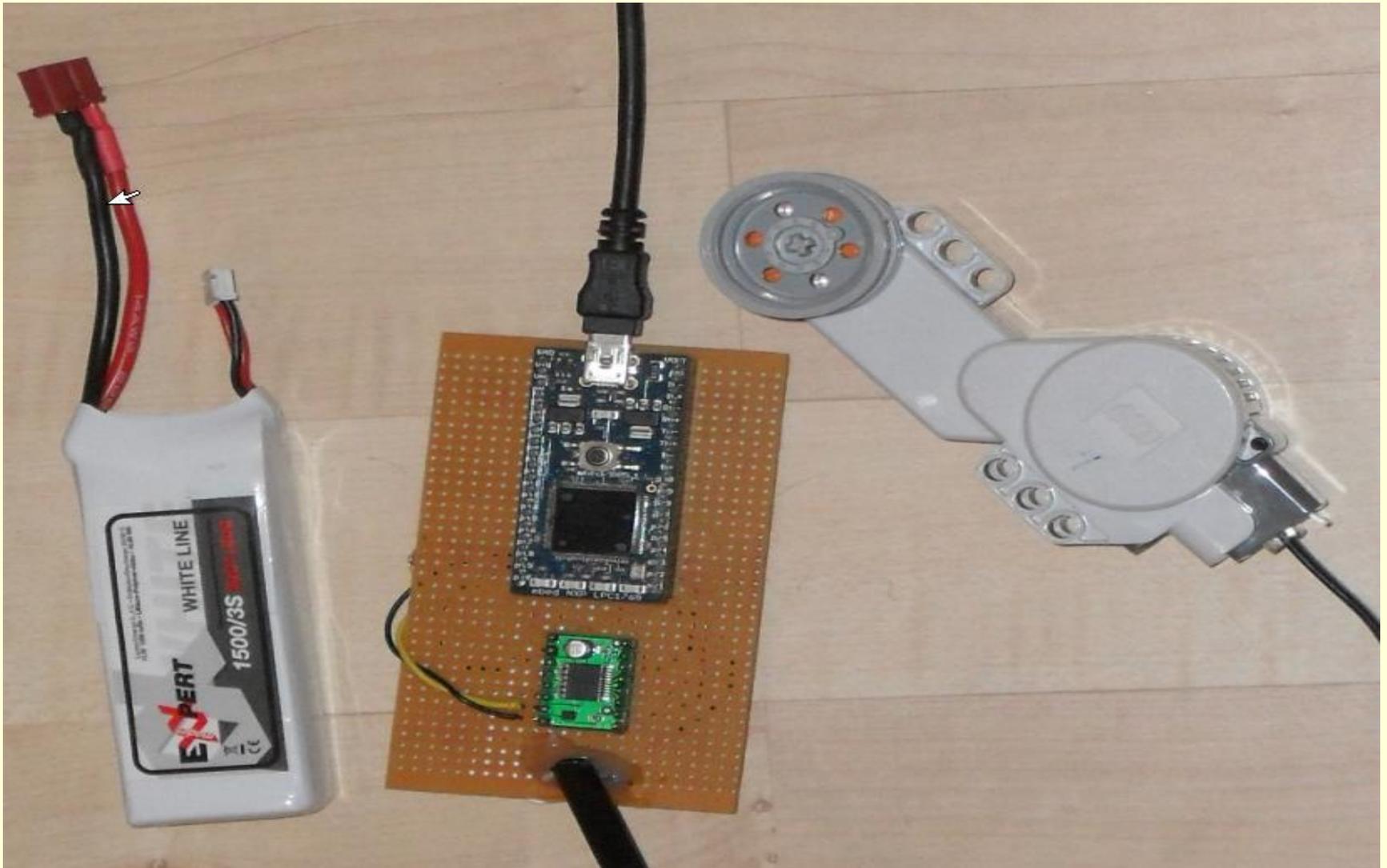
- **SvTest2.cpp:**
Anzeige von 2 Variablen (sv1, sv2).
Empfangen von Kommandos und Daten.
Quittieren des korrekten Empfangs der Daten mit printf()
- **FilterDemo.cpp:**
Rechteck-Funktionsgenerator mit einstellbarer Frequenz.
Parametrisches digitales Filter Fg und Güte (Bandbreite)
können online verstellt werden.
- Die Signalverarbeitungskette *Funktionsgenerator-Filter* wird zu Analysezwecken
(alle Abtastpunkte können angezeigt werden) mit 100Hz durchgerechnet.
- Wird die Signalverarbeitungskette mit 22kHz durchgerechnet kann man sich
die Wirkungsweise des Filters auch anhören.
- class Pulsgen in *FuncGen.h*
class ResoTP1p27 in *Filters.h*
class DachL in *AnalogIO.h*
class SimpleUart in *SimpleUart.h*

Parametrisches digitales Filter

- Strukturbild und Differenzengl. von ResoTP1p12

Regelung eines Gleichstrommotors

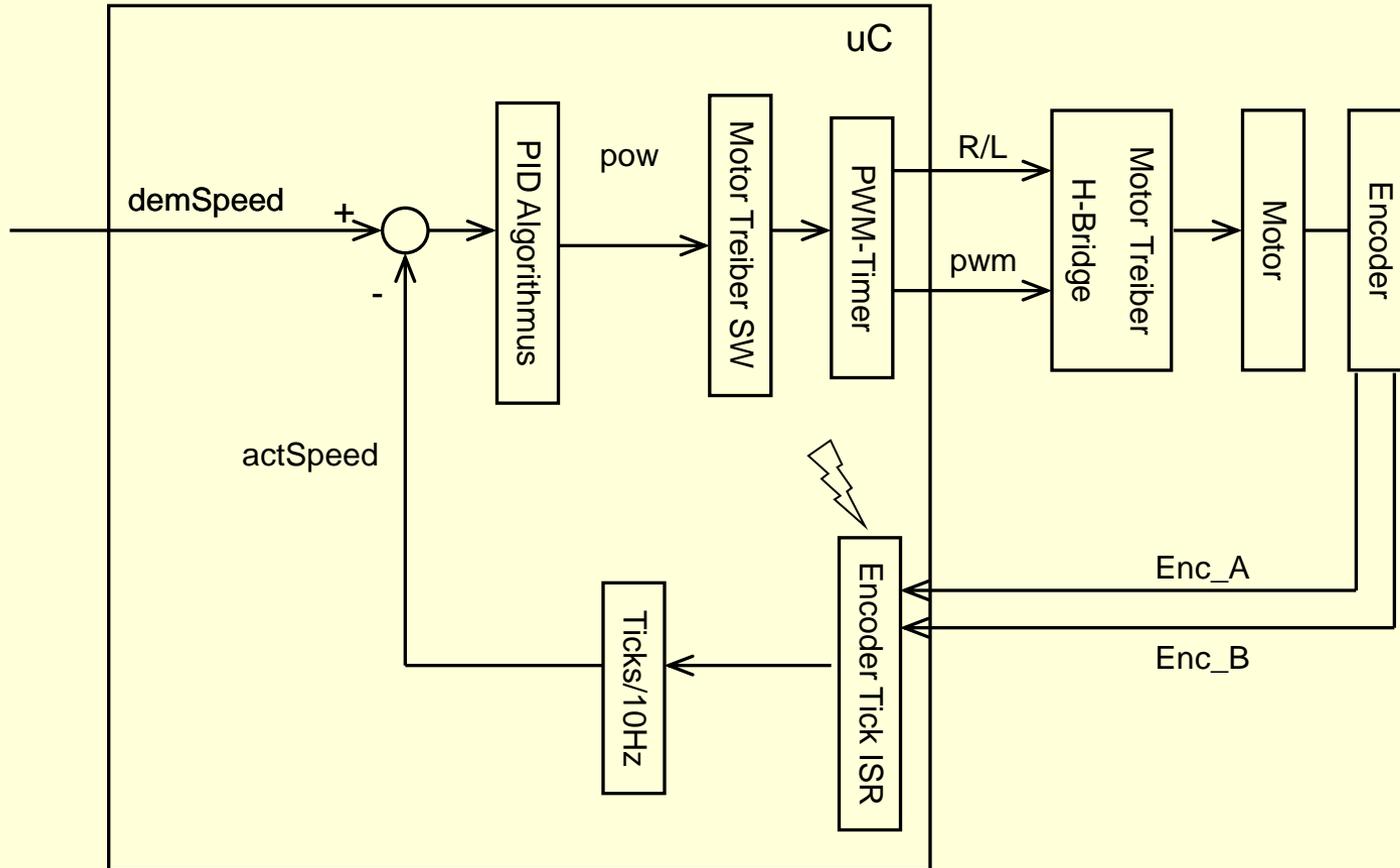
1. Drehzahlregelung (Accu-Schrauber/Bohrmaschine)
2. Positionsregelung (Servo-Regelung, Modellbau-Servo)



Verwendete Hardware

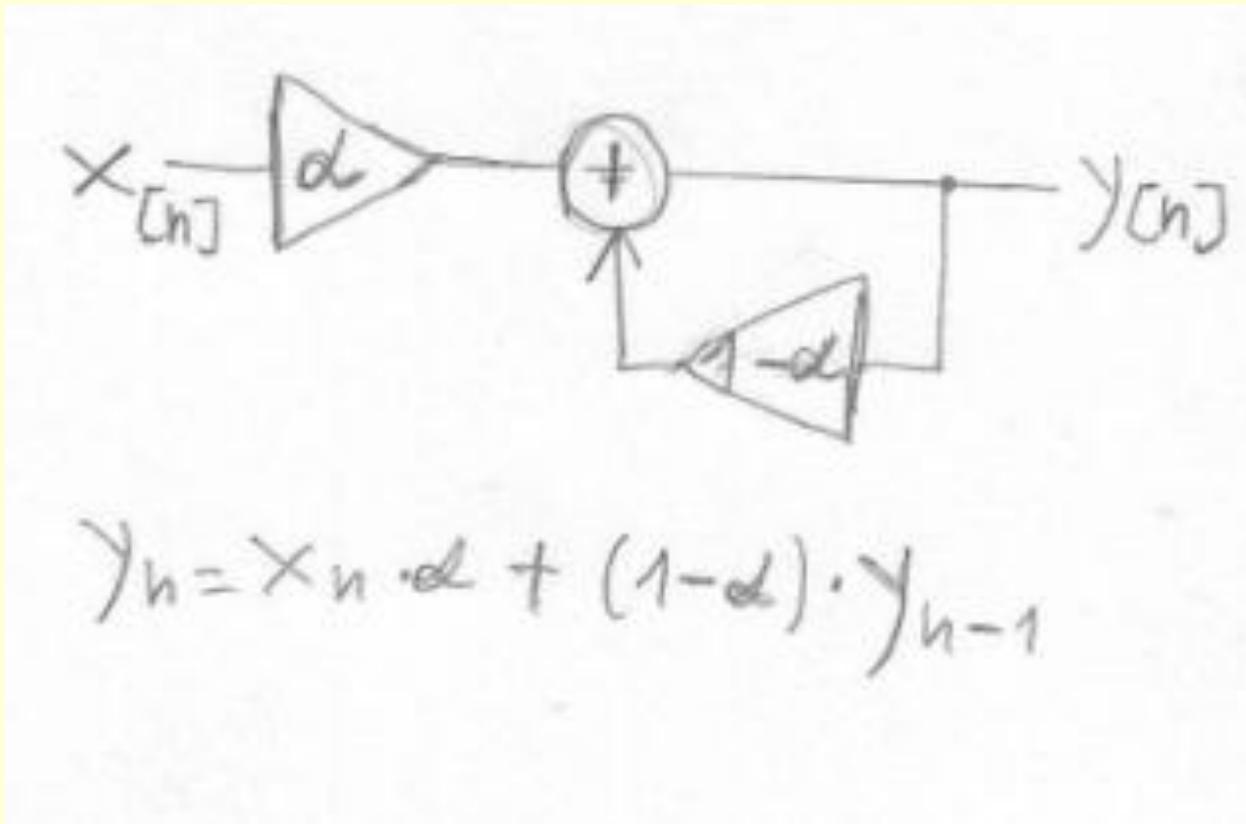
- Motortreiber (H-Brücke) TB66-12FNG von www.pololu.com um 6 Euro
- Lego-Mindstorms Getriebemotor (1:50)
mit Quadratur-Encoder 360 Ticks pro Umdrehung
um 15 Euro einzeln z.B. bei Ebay.
Das Getriebe ist vor allem für die Servo-Experimente wichtig.
- 12V Akku oder Netzgerät.

HW/SW Blockschaltbild



Simple Digitalfilter 1er Ordnung

- Zur Simulation einer größeren Motorzeitkonstanten
- Parameter Alpha (0..1) verhält sich wie eine Grenzfrequenz
- Zeitkonstante 1/Alpha kann durch einen einzigen Parameter verstellt werden
- Vorteil gegenüber Biliniertransformation eines analogen TP 1Ord hier werden alle Koeffizienten in H(z) verändert.



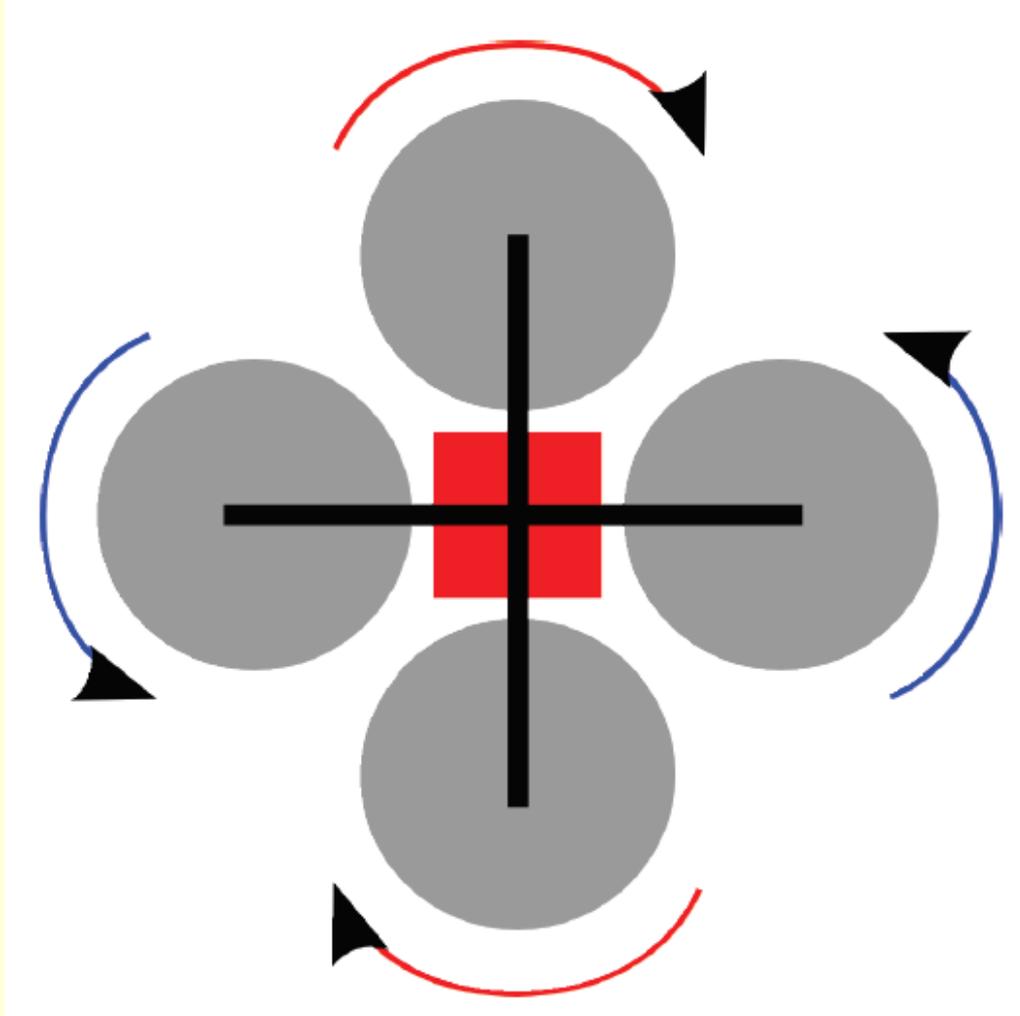
Quadrocopter

Allgemeine Einführung in
Quadrocopter_fuer_Anfaenger.pdf

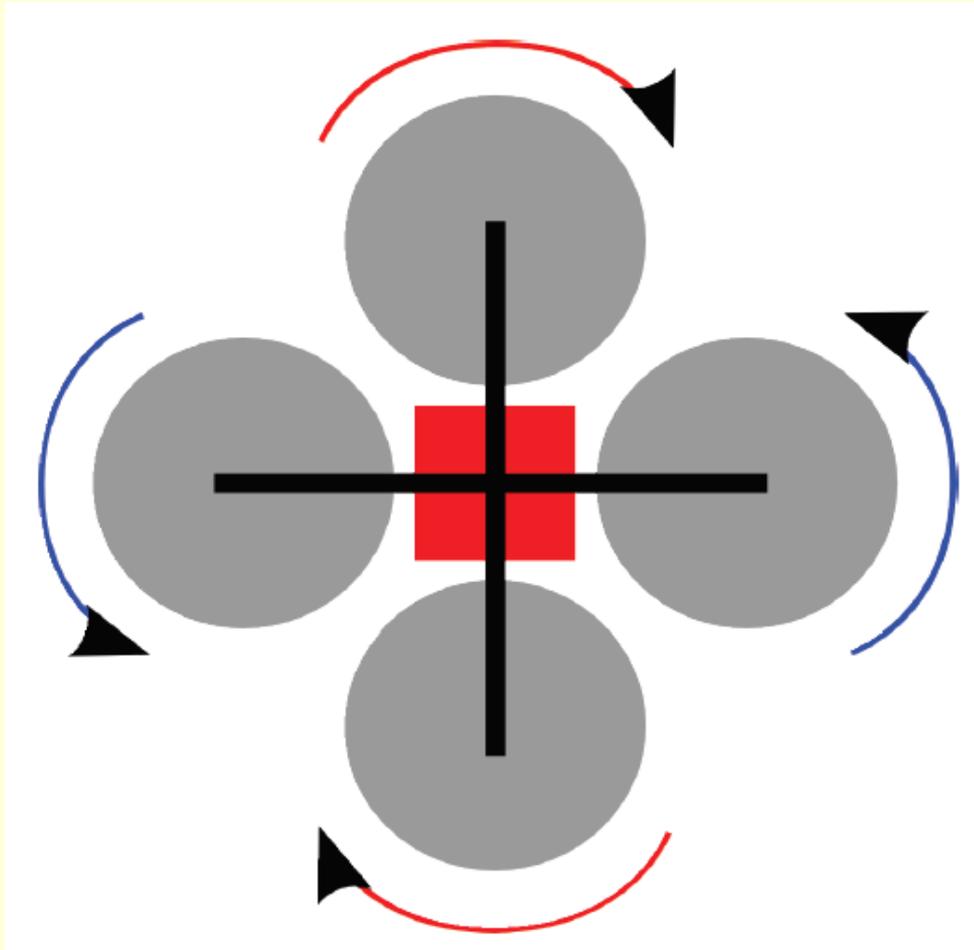
Was wird gesteuert (geregelt)

- Bei einem kunstflugtauglichen Flächenflugzeug werden durch den Steuerknüppel Drehraten um die 3 Achsen Roll, Nick und Yaw vorgegeben.
Ein durch den Piloten eingestellter Lagewinkel sollte eingehalten werden.
Flugzeug steigt mit 10Grad oder steuert nach NW.
Die Lenkung eines Autos verhält sich aus Regelungstechn. Sicht genau gleich.
- => Die Sollwerte für die Lageregelung (Stabilisierung) sind Winkelgeschwindigkeiten.
Die Istwerte der Winkelgeschwindigkeiten werden durch 3 orthogonal angeordnete Gyro-Sensoren erfasst.
- Ein PI-Regler für die Winkelgeschwindigkeit sollte die Winkelgeschwindigkeit ohne Regelabweichung auf 0 bringen können.
Und damit einen durch den Piloten einmal eingestellten Lagewinkel auch halten können.

Motordrehrichtungen und Drehmomentausgleich



Steuerung von: Roll, Nick, Yaw, Steigen, Sinken
durch die richtigen Drehzahländerungen an den
Motoren: Front, Back, Left, Right



Die Quadrocopter Gleichung

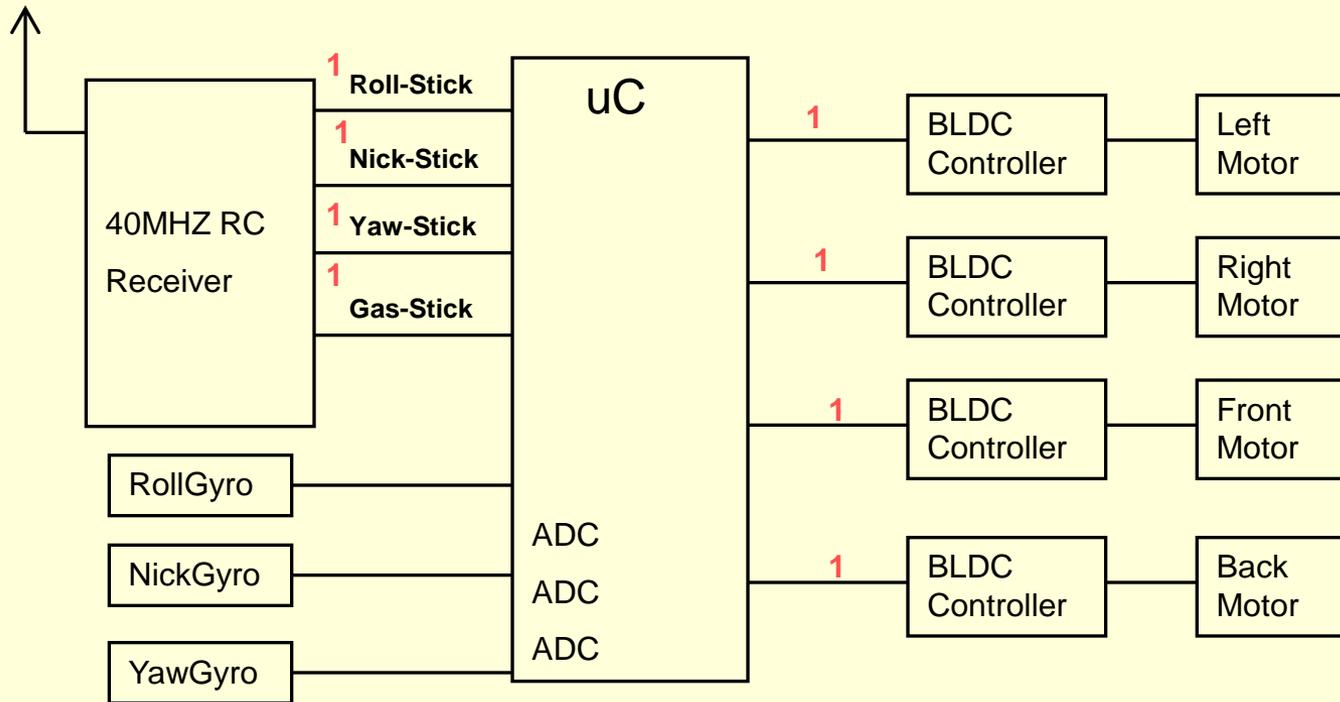
$$\textit{Front} = \textit{Throttle} + \textit{Nick} + \textit{Yaw}$$

$$\textit{Back} = \textit{Throttle} - \textit{Nick} + \textit{Yaw}$$

$$\textit{Right} = \textit{Throttle} + \textit{Roll} - \textit{Yaw}$$

$$\textit{Left} = \textit{Throttle} - \textit{Roll} - \textit{Yaw}$$

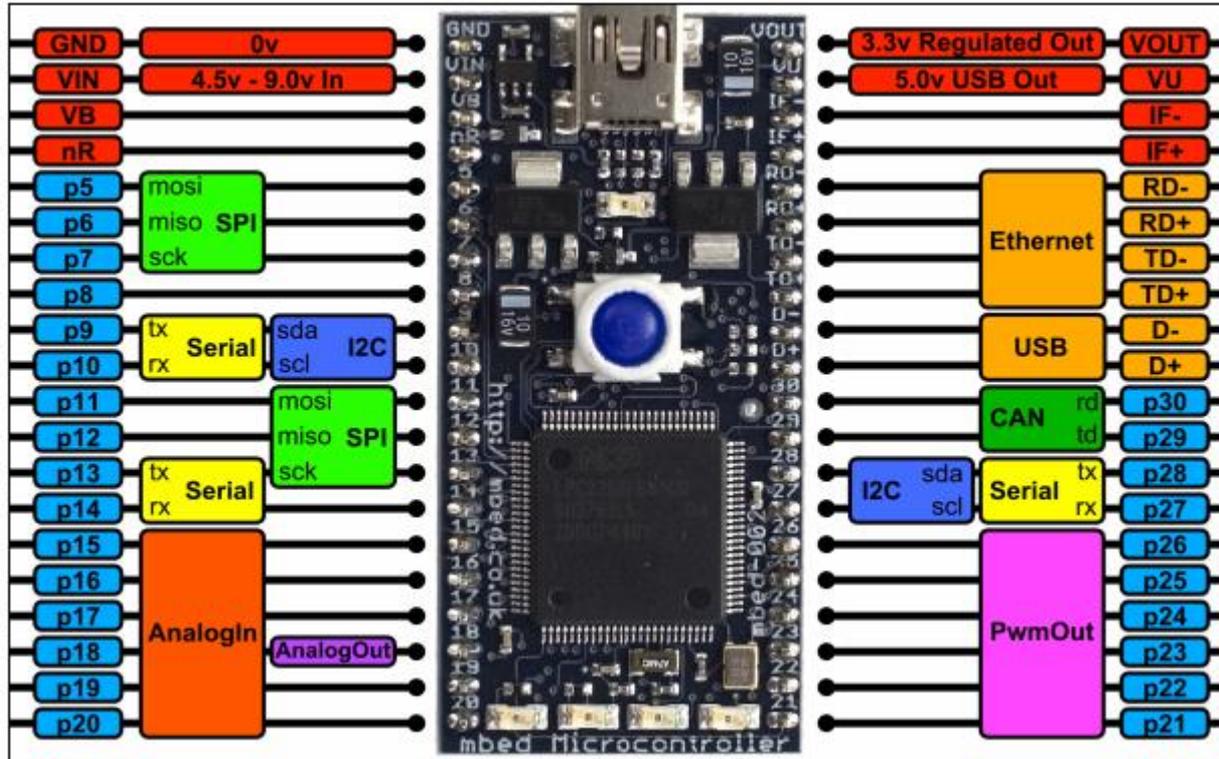
Blockschaltbild 1



1..... Standart RC PPM Signale
1ms Pulsbreite = voll Links
1.5ms = Mitte
2 ms = voll Rechts

BLDC
Brushless DC-Motor Controller

Verfügbare MBed IO/Pins



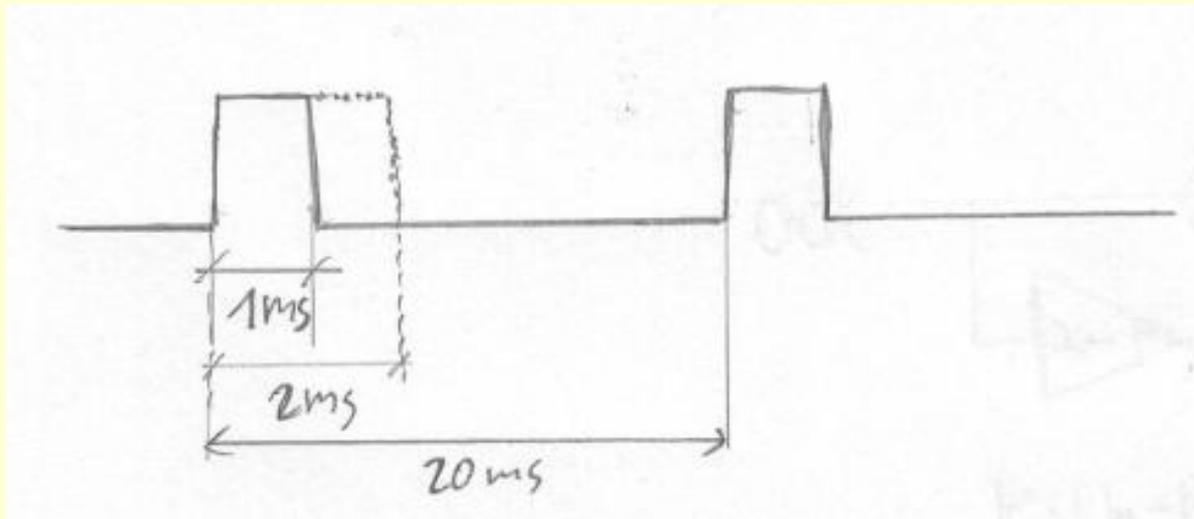
Standart RC PPM-Signal

PPM steht für Pulse Position Modulation.

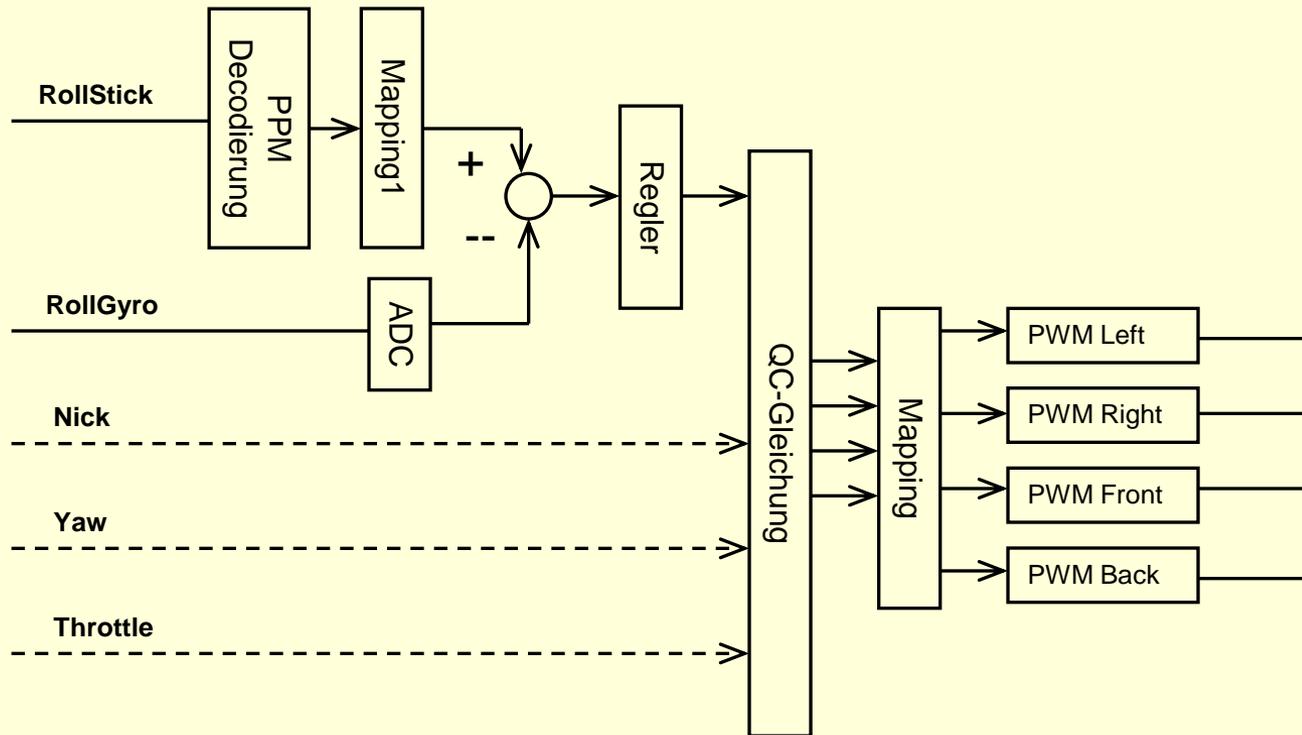
Die Information steckt in der Position der fallenden Flanke bezogen auf die steigende Flanke.

Der RC-Receiver liefert PPM-Signale.

Den BLDC-Controllern wird der Motorleistungs-Sollwert über PPM-Signale mitgeteilt.



Blockschaltbild 2



Mapping1: Der Wertebereich des RollSticks wird auf den Wertebereich gemapped den der RollGyro bei der maximalen gewünschten Rollrate liefert.

Parameter `ROLL_DEMAND_RANGE`

Benötigte Bauteile

- Von www.hobbyking.com und www.pololu.com
- 4x Brushless Motor
- 4x Motor-Controller
- 1x RC-Fernsteuerung
- 1x Lipo-Accu + Ladegerät
- 3 Achsen Gyro-Sensor
- Wie die Bauteile aussehen schauen wir uns im Internet an