



Samuel Bertsch

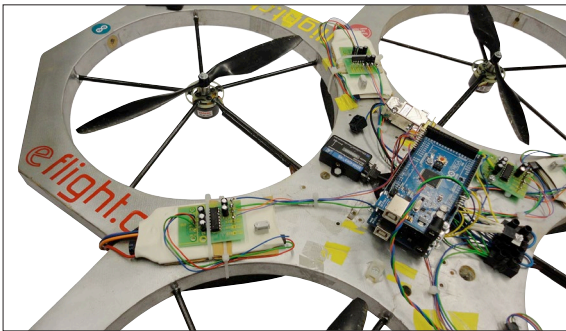


Biagio Cardillo

Diplomanden	Samuel Bertsch, Biagio Cardillo
Examinator	Prof. Dr. Markus Kottmann
Experte	Dr. Markus A. Müller, Frei Patentanwaltsbüro, Zollikon ZH
Themengebiet	Regelungstechnik

Hardware/Software zur Regelung eines Quadrocopters

Untersuchung der Programmierung eines Arduino direkt aus Matlab/Simulink



Der Quadrocopter wurde an der HSR entwickelt

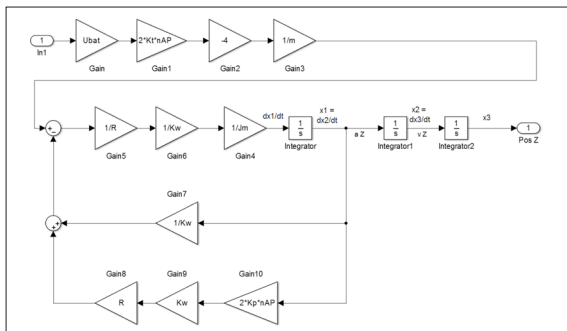
Aufgabenstellung: Ziel der Arbeit war, für die Regelung eines Quadrocopters ein Board zu finden, welches sich idealerweise direkt aus Matlab/Simulink programmieren lässt. Diese Anforderung erfüllen zurzeit folgende Produkte:

- Lego NXT Mindstorms
- BeagleBoard
- PandaBoard
- Arduino

Für die Navigation wurde eine integrierte Sensorlösung eingesetzt. Die Implementierung der Navigationsalgorithmen war nicht Teil der Arbeit. Als Fluggerät stand uns ein an der HSR entwickelter Quadrocopter zur Verfügung.

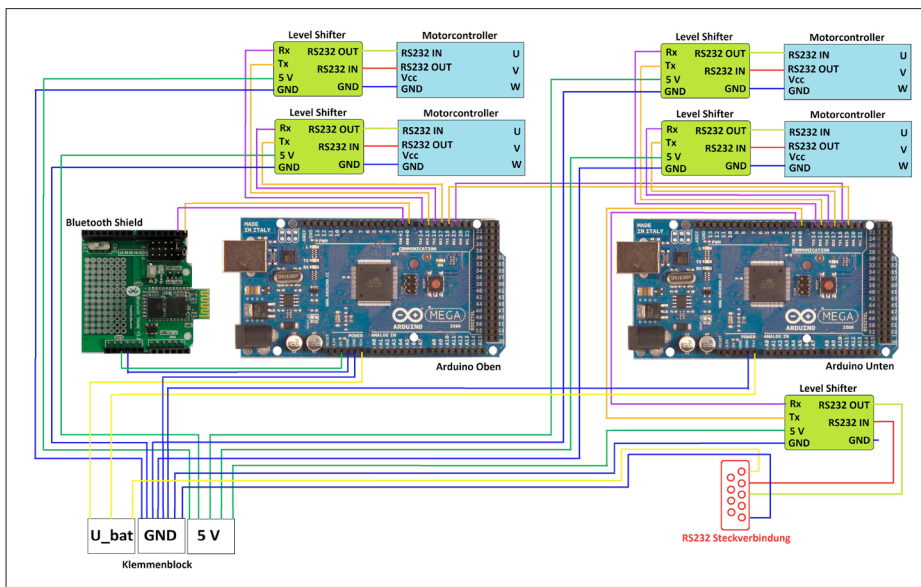
Vorgehen:

- Zu Beginn stand die Auswahl der zu verwendenden Hardware an. Früh wurde entschieden, auf einen Arduino zu setzen. Dies aufgrund seiner Anzahl serieller Schnittstellen, welche im Vergleich zu den anderen Produkten höher ist.
- Nachdem feststand, welche Hardwarekomponenten verwendet werden, unterzogen wir diese verschiedenen Tests. Dem Sensor wurde dabei die meiste Zeit gewidmet. Insbesondere das Kommunikationsprotokoll stellte sich als Herausforderung dar.
- Den Tests folgten der Zusammenbau aller Komponenten und das Ausmessen der physikalischen Parameter. Mit diesen war es möglich, verschiedene Linearisierungsmodelle zu erstellen, aus welchen anschliessend die Systemmatrizen bestimmt wurden.



Linearisierungsmodell des Quadrocopters in z-Richtung

Ergebnis: Eine grosse Schwierigkeit bestand in der Implementierung von seriellen Schnittstellen in Simulink. Ein fehlerfreies Auslesen der Sensordaten, welches sich als problematisch herausstellte, wurde mit einer aufwändig programmierten Zustandsmaschine erreicht. Die serielle Kommunikation zwischen den übrigen Komponenten lief stabil. Aufgrund der fehlenden PWM-Schnittstellen auf dem Arduino wurde anstelle einer üblichen Modellflugzeug-Fernbedienung eine Bluetooth-Schnittstelle implementiert.



Verdrahtung der Hardware. Auf die Darstellung der Motoren wurde in dieser Abbildung verzichtet. Am RS232-Stecker wird der Sensor angeschlossen.