

SITREP #1

Aktuelle Situation

- Drohne wurde aus Bausatz aufgebaut.
- Konfiguration der Sensoren via QGroundControl wurde durchgeführt.
- Raspberry Pie 4 wurde eingerichtet; Netzwerkverbindung ist möglich.
- Lastenheft bzw. Aufgabenliste wurde mit Herr Müller abgestimmt.
- Konzept für Regelungsarchitektur wurde erarbeitet
 - Architekturentwurf aktualisiert
- Auslesen von Sensordaten möglich
 - /mavros/state (allgemeiner Zustand; „armed“)
 - /rangefinder/range (Lasermessung Bodenabstand)
 - /mavros/imu/data (Beschleunigungs- und Lagedaten)
- Dokumentation
 - LaTeX Gerüst erstellt
 - Ausarbeitung ausstehend
- Problem der Fernsteuerung behoben (in issue #2 dokumentiert)

Probleme

Probleme sollen als *Issue* (Label **Bug**) in git geführt werden.

- Spannungversorgung für LED-Streifen verpolt
- Netzwerk-Einstellung der genutzten VM umständlich.
 - Host OS = Windows 10, betriebliche Vorgabe
 - Verbindung zu RosMaster benötigt NAT, Kommunikation zu RosTopics benötigt Bridge

Anstehende Aufgaben

Aufgaben sollen als *Issue* in git geführt werden.

- Git Issues entsprechend den hier genannten Aufgaben erstellen.
- ROS-gesteuerte Flugversuche
 - RosNodes (Publisher & Subscriber) implementieren
 - Keyboard-Subscriber (50%)
 - MavLink-Subscriber
 - Rangefinder Reader
 - Acceleration Reader
 - MavLink-Publisher
 - Bridge zur Application-Schicht
- Validieren der entworfenen Architektur mit Herr Strand und/oder Herr Müller
- Implementieren der Architektur
 - Abstraction Schicht (100%)
 - Domain- Schicht (95%)
 - Application- Schicht (95%)
 - Adapter- Schicht (30%)
 - Plugin- Schicht (5%)
 - Einbindung externer .cpp und .h Dateien in catkin_make bzw. CMakeLists.txt ermitteln
 - Einlesen in Verfahren zur Signalglättung (ggf. ist ein Ringspeicher für Daten ausreichend)
- Implementierung von Software-Tests
- Dokumentation
 - Problemstellung aus Projektanmeldung übernehmen und ggf. ausformulieren
 - Coex Clover beschreiben
 - ROS beschreiben
 - Architektur beschreiben
- Kommunikation zwischen PX4-FlightController und Raspberry Pie 4 via UART (optional)

