



*Dokumentation der praktischen Jahresarbeit*

*„Bau zweier FPV-Drohnen“*

*in der Klasse 9*



## *Inhaltsverzeichnis:*

<i>Einleitung</i> .....	<i>3</i>
<i>Was ist „FPV“ überhaupt?</i> .....	<i>3</i>
<i>Analog- vs. Digital FPV</i> .....	<i>4</i>
<i>Fernsteuerung und Fernsteuerungs-Protokolle</i> .....	<i>5</i>
<i>Verwendete Teile + Erklärung</i> .....	<i>6</i>
<i>Zusammenbau des Frames</i> .....	<i>8</i>
<i>FC, 3D Druck und Akkus</i> .....	<i>8</i>
<i>Motoren</i> .....	<i>10</i>
<i>Analog FPV</i> .....	<i>11</i>
<i>Crossfire-Empfänger</i> .....	<i>13</i>
<i>GPS-Modul</i> .....	<i>13</i>
<i>Digital FPV</i> .....	<i>14</i>
<i>Betaflight (Software)</i> .....	<i>15</i>
<i>Erstflug des analogen Copters und Crossfire</i> .....	<i>20</i>
<i>Aufnahmen und Gyroflow</i> .....	<i>21</i>
<i>Das GPS-Problem</i> .....	<i>22</i>
<i>Drohnen-Gesetze</i> .....	<i>22</i>



## Einleitung

Als meine praktische Jahresarbeit in der neunten Klasse habe ich zwei sogenannte FPV-Drohnen bzw. FPV-Copter gebaut. Einen mit analogem- und einen mit digitalem FPV-Standard. Was genau das bedeutet erkläre ich später. Auf das Projekt bin ich tatsächlich über YouTube gekommen, über den FPV-Kanal „Spinfast“. Bei diesem habe ich ein Video über den 3Style, einen 3 Zoll Freestyle-Copter unter 250g Abfluggewicht gesehen. Ich fliege nun schon seit über 3 Jahren FPV-Drohnen und wollte von Anfang an schon eine selbst bauen. Dazu ist es leider nicht gekommen. Mittlerweile fliegt mein Vater allerdings auch FPV-Drohnen, und als wir besagtes Video geschaut haben, kam die Idee, doch einfach diese Drohne als praktische Jahresarbeit zu bauen. Klar war allerdings, dass mein Vater keinen analogen Copter möchte, da er das digitale DJI-System benutzt. Er kann damit zwar auch analoges Videosignal empfangen, dieses ist aber deutlich schlechter als das digitale (siehe unten). Ich wiederum brauche das analoge System, da ich dieses nutze und kein digitales Signal empfangen kann.

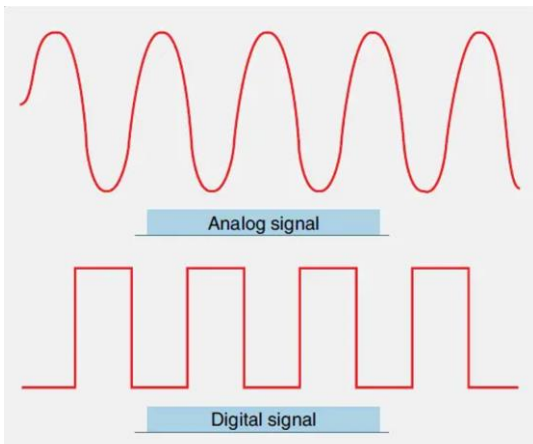
## Was ist „FPV“ überhaupt?

Aber was bedeutet „FPV“ überhaupt? Und was ist der Unterschied zwischen digitalem- und analogem FPV-Signal? FPV ist die Abkürzung für „First Person View“, zu Deutsch: Ich-Perspektive. Im RC Bereich bedeutet das, dass das Modell bzw. die Drohne nicht auf Sicht (Line of Sight, kurz LOS), sondern per Video ferngesteuert wird. Dafür gibt es die sogenannten FPV-Brillen. Diese sind entfernt mit einer VR-Brille vergleichbar. An der Drohne sind eine FPV-Kamera und ein Videosender, kurz VTX, angebracht. Dieser sendet das Kamera Bild mit einer Latenz von ca. 20-30 Millisekunden an die FPV-Brille.

Hier kommt auch der Unterschied zwischen analogem- und digitalem Bild ins Spiel.



## Analog- vs. Digital FPV



<https://www.flight-mechanic.com/fundamentals-of-electronics-analog-electronics-part-one/>

Zuerst einmal zum allgemeinen Unterschied zwischen analogem- und digitalem Signal: Bei analogem Signal wird das Bild mit Hilfe von kleinen Frequenzunterschieden übertragen, es gibt also eine Art Kurve mit unterschiedlichen Höhen. Nachteil daran ist, dass wenn das Signal schlechter wird, wird die Kurve etwas verfälscht, dadurch entstehen Bildfehler. Vorteil daran ist, dass das Signal, nicht wie bei digitalem Signal, einfach abbricht, sondern deutlich

schlechter wird, bis man nur noch Bildrauschen hat.

Digitales Signal wird mit Hilfe von Nullen und Einsen übertragen, welche wie oben zu sehen, nicht als Kurve, sondern als Erhebung oder Vertiefung dargestellt werden. Werden diese durch schlechten Empfang verfälscht, ist immer noch sehr lange eine Höhe oder Tiefe zu erkennen, also eine Eins oder Null. Dadurch gibt es ein insgesamt viel besseres Bild mit deutlich weniger Fehlern. Nachteil ist, dass der Empfang schlagartig abbrechen kann, wenn der Empfang zu schlecht ist.



Wie man sieht, ist die digitale Bildqualität deutlich besser als die analoge. Dafür aber deutlich teurer.



## Fernsteuerung und Fernsteuerungs-Protokolle

Ferngesteuert wird die Drohne mittels einer Fernsteuerung, auch „Funke“ genannt. Diese ist bei mir eine Jumper T16 Pro.



Diese hat ein eingebautes Multiprotokoll-Modul, welches alle gängigen Fernsteuerungs-Protokolle beherrscht. Verschiedene Firmen nutzen nämlich verschiedene Fernsteuerungs-Protokolle. Da gibt es z.B. das Hott-Protokoll von Graupner, das DSMX-Protokoll von Spektrum oder das Frsky ACCST Protokoll. Die Protokolle kann man auch mit verschiedenen Sprachen vergleichen, also ein Protokoll spricht z.B. Englisch, ein anderes Chinesisch usw.

Ich habe mich für das Projekt allerdings für das Crossfire-Protokoll entschieden, da ich dieses auf den meisten meiner anderen Copter im Einsatz habe und damit bis vor diesem Projekt nur gute Erfahrungen gemacht habe. Dieses ist ein Long-Range System von der Firma Team Black Sheep, welches statt auf 2,4Ghz auf 868MHz sendet.



Da dieses wie gesagt auf 868MHz statt auf 2,4 GHz sendet, wird es von dem Multiprotokoll-Modul meiner Funke nicht unterstützt, dafür habe ich das sog. Crossfire Micro TX, welches in den Modulschacht der Funke kommt. Durch ein Lua-Script ist dieses über die Funke, aber auch über WLAN per App konfigurierbar. Je geringer der GHz bzw. MHz Wert, desto weiter reicht der Empfang bei gleicher Sendeleistung, jedoch steigt auch die Latenz. Dies liegt daran, dass längere Wellen Objekte besser durchdringen können als kürzere, dafür brauchen längere Wellen aber auch länger als kurze, daher die höhere Latenz auf 868MHz. In Deutschland und der EU darf man auf besagten 5,8GHz (FPV Signal) und 868MHz nur mit 25MW senden. Damit kommt man, je nach Umgebung, unterschiedlich weit. Mein „Rekord“ waren per GPS gemessene 555 Meter. Ein GPS ist im FPV Bereich, anders als bei den „normalen“ Kamera-Drohnen, kein Standard. Zum Geschwindigkeits- und Entfernungsmessen ist es aber trotzdem praktisch. FPV Drohnen gibt es in verschiedenen „Größen“. Früher waren 5Zoll Copter sehr verbreitet, heute sind, wegen der EU Regularien bzw. Drohnengesetze, Copter unter 250 Gramm Abfluggewicht, meist 3 Zoll, weit verbreitet. Der Zoll Wert gibt dabei den Propeller Durchmesser an. Der 3Style, also mein Selbstbau ist, wie der schon Name sagt, ein 3Zoll Copter

## Verwendete Teile + Erklärung

Aber nun zum eigentlichen Projekt. Als der 3Style als Projekt feststand, habe ich als erstes eine Teile Liste erstellt und die richtigen Komponenten im Internet herausgesucht. Als erstes natürlich den besagte „3Style“ Frame, also der „Rahmen“, auf den alle Komponenten aufgebaut werden. Bei diesem handelt es sich wie gesagt um einen 3Zoll Frame aus Karbon. Die Frames für FPV-Drohnen sind in der Regel alle aus Karbon, da dieses leicht und widerstandsfähig ist. Eine Ausnahme sind da nur die sog. Tiny Whoops, also kleine FPV Copter mit Propellerschutz, z.B. fürs indoor fliegen, welche dann meist TPU artigen, also „weichen“, bzw. etwas verformbaren Kunststoff verwenden. Als FC (Flight Controller) kommt ein BetaFPV F405 F4 FC mit 20 Ampere ESC (Electronic Speed Control) zum Einsatz. Der FC ist also ein AIO (all in one) FC, was bedeutet, dass die ESC´s für die 4 Motoren gleich mit auf dem Board sind und man diese nicht einzeln bzw. als extra Board benötigt. Der FC ist wie eine Art Einplatinencomputer, auf dem die Betaflight Firmware läuft (dazu komme ich später) und auf dem alle Befehle von der Funke ankommen und durch Betaflight verarbeitet werden. Betaflight gibt die berechneten Befehle für die Motoren an die 4 ESC´s weiter, welche dann die Motoren „steuern“. Als Motoren kommen die BetaFPV 1404 4500 KV Brushless Motoren zum



Einsatz. 1404 gibt dabei die Größe des Stators an, also 14 mm in der Breite und 4 in der Höhe. KV steht dabei für Umdrehungen pro Volt, also 4500 Umdrehungen pro Minute bei einem Volt. Brushless bedeutet, dass der Motor keine Kohlebürsten mehr zur Stromübertragung zwischen Rotor und Stator benötigt. Als FPV Kamera habe ich eine Caddx Ant, und als VTX (Videosender) einen Zeus Nano von HGLRC ausgewählt. Als Empfänger nutze ich den TBS Crossfire Nano RX. Bis auf FPV-Kamera und Videosender sind die Komponenten bei beiden Coptern gleich. Bei dem digitalen Copter kommt eine Caddx Vista- Kamera zum Einsatz. Bei dieser ist das digitale Sendemodul gleich mit dabei. Als Propeller habe ich am Anfang die HQProp 3 Zoll 3 Blatt Props benutzt, ab Mitte Mai die deutlich besseren BetaFPV HQ 75 5 Blatt Propeller. Die GPS-Module sind die M8 Glonass von TBS. Dazu kommen noch einige Kleinteile.



## Zusammenbau des Frames

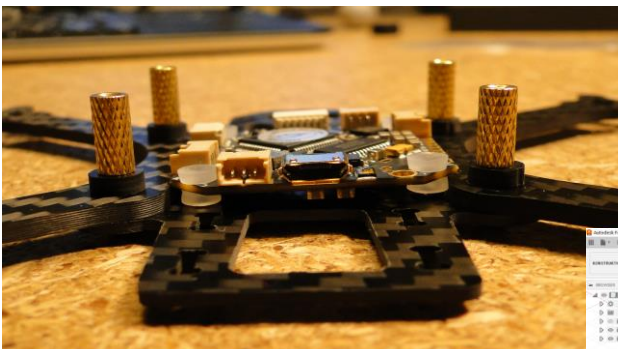
Als ersten Bauschritt habe ich den Frame ausgepackt, die verschiedenen Schrauben, Karbonteile und Standoffs sortiert und den Frame anschließend zusammengebaut. Das Zusammenbauen war nicht weiter kompliziert, außerdem gab es eine Explosionszeichnung im Internet.



## FC, 3D Druck und Akkus

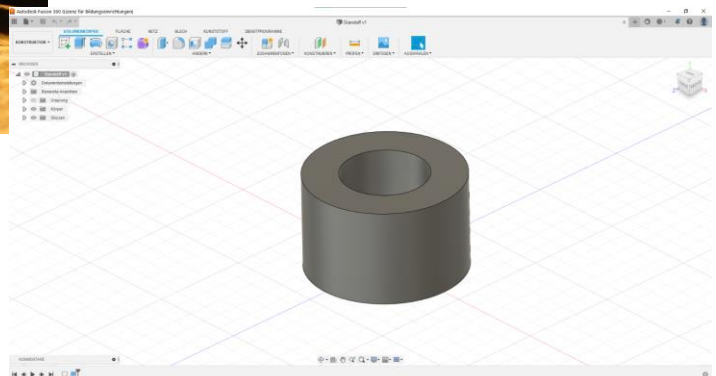
Als nächstes war der FC an der Reihe. Bei diesem musste ich zuerst die mitgelieferten Gummipuffer in die Befestigungslöcher drücken, damit dieser mechanisch wenigstens etwas vom Frame entkoppelt ist. Dafür habe ich die Gummipuffer auf ihre jeweiligen Löcher gelegt und mit einem Schraubenzieher hineingedrückt. Das war die dritt nervigste Arbeit am Copter.

Als die Gummipuffer dran waren, wollte ich den FC auf den Frame aufschrauben. Allerdings war der Abstand zwischen den beiden Armen nicht groß genug, sodass der



FC an den Seiten auflag. Das war natürlich unpraktisch, deshalb habe ich einfach Abstandshalter mit dem 3D Drucker gedruckt. Diese habe ich zuerst in Fusion

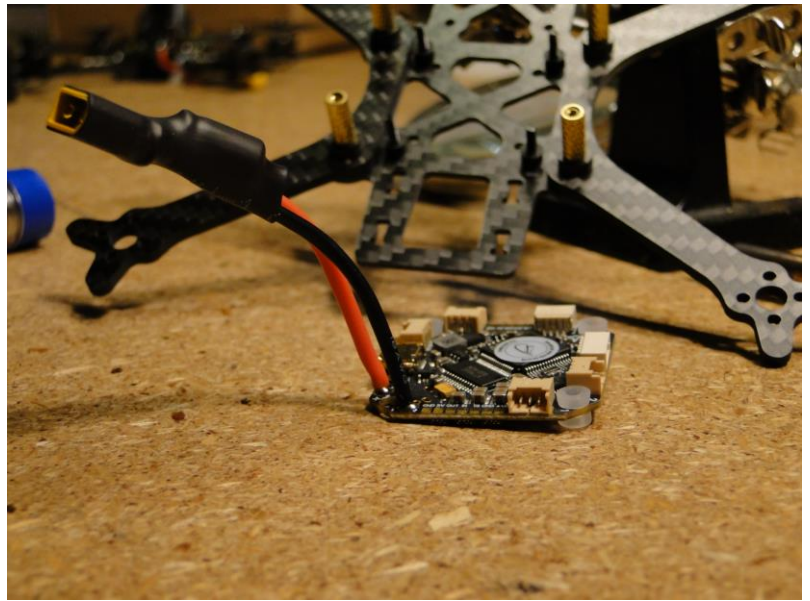
360 konstruiert und dann in der 3D Druck-Software Cura von Ultimaker die richtigen Druck-Einstellungen vorgenommen und die erzeugte





G-code Datei auf die SD- Karte des Druckers kopiert. Anschließend habe ich den Druck gestartet, welcher zum Glück nur 5 Minuten gedauert hat.

Mit Hilfe der Abstandshalter passte der FC dann auch auf den Frame. Bevor ich diesen jedoch einbauen konnte, habe ich noch das Lipo-Kabel angelötet. An diesem ist am XT30 Stecker ein kleiner Kondensator angelötet, welcher kleine Spannungsspitzen herausfiltert. Bei den 3S Akkus ist das noch nicht so wichtig, für 4S und 6S aber schon. Das „S“ gibt dabei die Anzahl der in Reihe geschalteten Zellen im Akku an. Bei Lithium Polymer Akkus (kurz Lipo) sind



das immer 3,7V pro Zelle, also

z.B. 11,1V bei 3S-Akkus. Ist ein Lipo geladen, kann die Zellenspannung auch bis zu 4,2V bzw. auch 4,3V betragen. Tiefer als bis zu 3,7V pro Zelle sollte man Lipos aber nicht entladen, da diese sonst Schaden nehmen können. Auf den Akkus ist außerdem immer die Kapazität in



Milliamperestunden (mAh) angegeben. Der C-Wert steht für den maximalen Entladestrom.

Also 1C = Kapazität des Akkus. Die Akkus für den 3Style haben 750mAh und 95C, können also theoretisch 71,25 Ampere abgeben.

Als das Akku-Kabel angelötet war, habe ich den FC mit den 4 langen Schrauben, die eigentlich für die Caddx Vista gedacht waren, angeschraubt. Dabei war zu beachten, dass die Gummipuffer nicht zusammengequetscht werden, damit diese noch als Vibrationsdämpfer dienen. Als nächstes habe ich den FC kurz mit Strom versorgt, um zu schauen, ob er überhaupt funktioniert. Dazu habe ich einen sog. Smokestopper verwendet. Dieser ist eine Art Sicherung für den Akku, den ich verwende, wenn ich irgendetwas ausprobieren möchte. Wenn der FC nämlich einen Kurzschluss hätte, könnte es

sein, dass mir der Lipo um die Ohren fliegt, und das wäre ungünstig. Der Smokestopper ist also eine Art Sicherung. Es haben zum Glück beide FC's funktioniert.

## Motoren

Als nächstes wollte ich die Motoren montieren. Dabei musste ich leider feststellen, dass bei den Motoren, die natürlich nicht an den FC gelötet, sondern nur mit Steckern angesteckt werden sollten, die Kabel zu kurz waren. Da man die Motoren leider nicht direkt an den FC anlöten konnte, sondern diese wie gesagt stecken musste, musste ich also alle Motorenkabel verlängern. Das war die zweit nervigste Arbeit am Copter. Ein Brushless Motor hat nämlich 3 Kabel, die natürlich alle verlängert werden wollen. Also musste ich an jedem Motor 6-mal Löten, insgesamt hatte ich also 24 Lötstellen alleine für die Motoren, die natürlich auch noch alle mit Schrumpfschlauch isoliert werden mussten. Ich habe also die Motorenkabel in der Mitte durchgeschnitten und abisoliert. Dann habe aus dem Kabel eines defekten Temperaturfühlers die Verlängerungsstücke herausgeschnitten und ebenfalls abisoliert. Anschließend habe ich die „Verlängerungskabel“ auf der Motorseite angelötet und anschließend pro Motor 6 kleine Schrumpfschlauch Stücke abgeschnitten. Diese habe ich dann über die schon verlöteten Kabel geschoben, jeweils natürlich zwei pro Kabel. Danach habe ich den Stecker wieder angelötet und den Schrumpfschlauch mit Hilfe eines GaslötKolbens geschrumpft. Dies habe ich bei allen 4 Motoren gemacht. Zum Glück habe ich immer dran gedacht, den Schrumpfschlauch vor anlöten der Stecker über das Kabel zu schieben, es wäre nämlich nicht das erste Mal gewesen, das ich eine Lötstelle wieder trennen und nochmal Llöten musste, weil ich den Schrumpfschlauch vergessen hatte.

Vor dem anschrauben der Motoren habe ich als Schutz der Kabel vor den Kabelbindern, mit denen ich diese auf den Frame - Armen befestige, einen größeren Schrumpfschlauch über jeweils alle drei Kabel geschoben. Als letztes habe ich die Motoren natürlich noch angeschraubt.





## Analog FPV

Als nächstes ging es um die FPV Komponenten. Hier gibt es natürlich einen großen Unterschied zwischen analogem- und digitalem Copter. Als erstes beschreibe ich deshalb den Aufbau der analogen- und anschließend den der digitalen Komponenten. Zum Einsatz kommt hier, wie am Anfang beschrieben, ein VTX von HGLRC und die FPV-Kamera Caddx Ant zum Einsatz. Als Antenne für den VTX nutze ich eine Foxeer Lollipop mit UFL Connector. Die Komponenten sind, wie auf dem Bild zu sehen, nicht viel größer als eine 1 Cent Münze – entsprechend klein sind auch die Lötstellen.

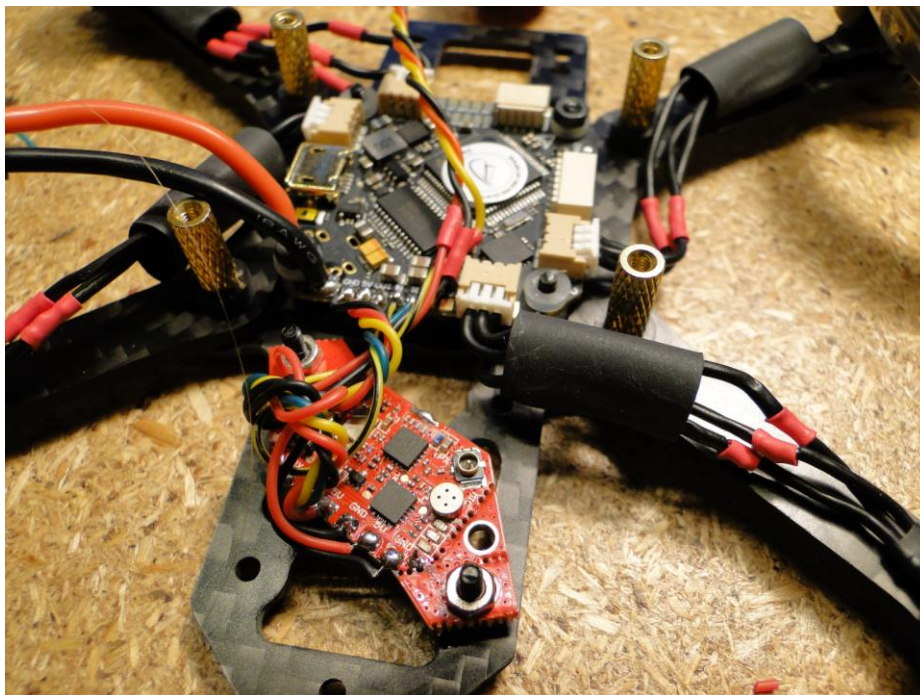


Als erstes habe ich den VTX mit dem FC verkabelt, danach die Kamera mit dem VTX und dem FC. Der VTX benötigt folgende Anschlüsse: 5V, GND (Ground, also Minus), VID, also Video in und RX bzw. am FC dann natürlich TX. RX steht dabei für Receiver, also Empfänger und TX für Transmitter, also Sender. Diese gehören zu den sog. UART's. Ein UART ist wie ein USB-Anschluss am PC, man kann über diesen also „Geräte“ an den FC anschließen, bzw. es „Geräten“ ermöglichen, mit

dem FC zu kommunizieren. Jeder FC hat mindestens einen UART, nämlich den für den Empfänger. Der FC, den ich nutze hat 5 UART's, einer davon hat aber nur einen TX-Anschluss, dieser wird für den VTX verwendet, um diesem theoretisch Befehle zu Kanal und Sendeleistung zu geben, sodass man diese über Betaflight einstellen kann. Praktisch hat das natürlich nicht funktioniert. Es gibt, wie gesagt, meist RX und TX. TX sendet dabei einen Befehl oder Wert, RX empfängt einen Befehl oder Wert. Daher muss man natürlich das RX Pad am VTX mit dem TX Pad am FC verbinden.

Als ich den VTX richtig verkabelt hatte, folgte die Verkabelung der FPV-Kamera. Deren 5V- und GND-Anschluss habe ich mit an den VTX angelötet, das Signal-kabel für das Bild habe ich an das dafür vorgesehene Pad am FC gelötet. Das Video-Bild geht deshalb erst in den FC und dann in den VTX, da ich sonst kein OSD (On-Screen-Display) hätte. Dieses wird vom FC über das Kamera Bild „gelegt“ und zeigt mir alle wichtigen Infos über die Drohne an, also Lipo-Spannung, aktueller Stromverbrauch in Ampere und einiges mehr. Das OSD wird in der Betaflight-Software konfiguriert, dazu komme ich aber später noch.

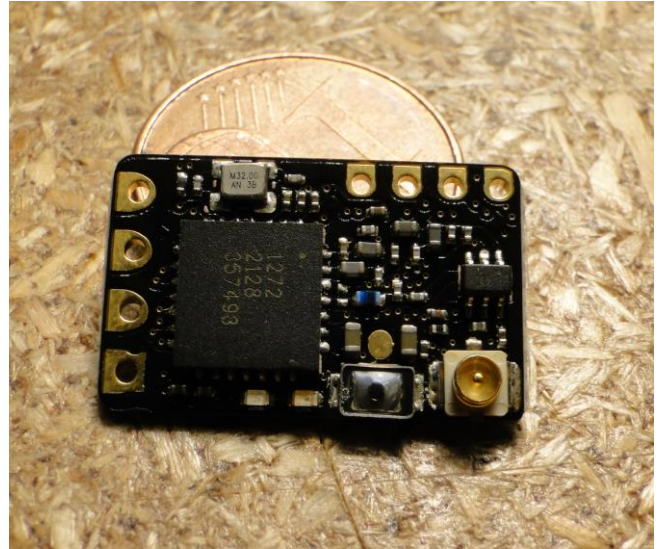
Als alles verkabelt war, habe ich den VTX noch auf den Frame geschraubt.





## Crossfire-Empfänger

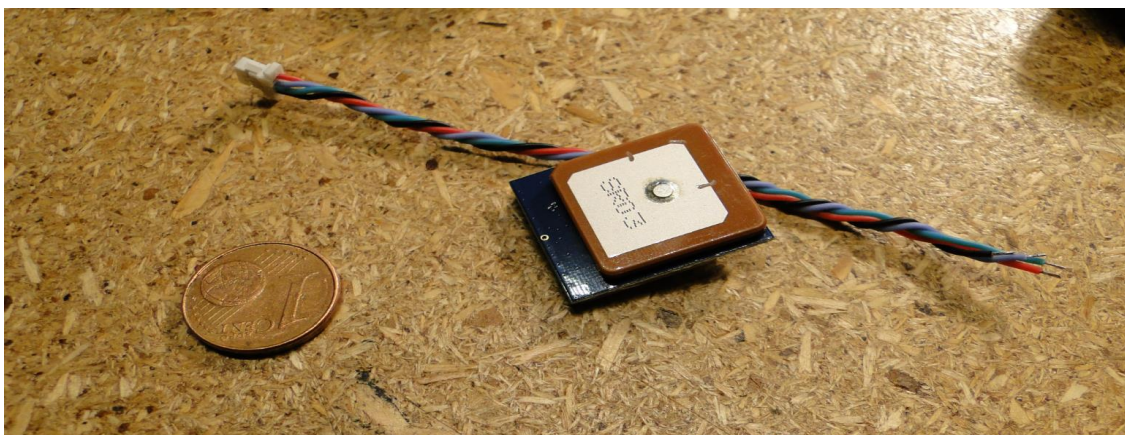
Als nächstes ging es um den Crossfire-Empfänger. Dieser ist, wie rechts zu sehen, auch nicht viel größer als eine 1Cent-Münze. Er benötigt 4 Anschlüsse, nämlich RX und TX, 5V und GND. Außerdem benötigt er natürlich noch eine Antenne mit UFL-Connector, allerdings eine andere als der VTX, da die Antennen auf ihre jeweilige Frequenz optimiert sind.



Zuerst habe ich den Nano an den beim FC mitgelieferten Stecker mit Hilfe des Anschlussplans, auf dem die Bezeichnungen der Löt-Pads dargestellt waren, angelötet und anschließend die Antenne aufgesteckt. Als letztes habe ich das Ganze noch mit Schrumpfschlauch vor Staub und Co. geschützt und den Stecker an den FC angeschlossen.

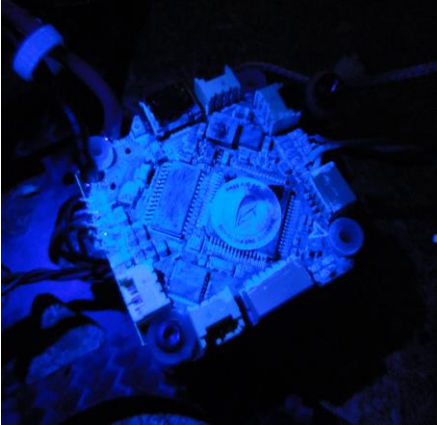
Als vorletzten Bauschritt habe ich noch das GPS-Modul verkabelt, dieses benötigt ebenfalls 5V, GND und RX und TX. Das Modul habe ich vorerst über der Kamera montiert, später habe ich dann aber noch eine Halterung gedruckt, die am hintersten Standoff angebracht wird und das GPS Modul so hinten am Copter befestigt ist.

## GPS-Modul



Als letzten Bauschritt habe ich nun noch die Kamera- „Halterung“ zusammengeschaubt, diese samt Kamera auf den Frame gesetzt und die Top Plate auf den Frame geschraubt. Als alles passte, habe ich den FC nochmal abgenommen und mit sog. Coating eingepinselt. Dieses schützt den FC vor (Spritz)Wasser und ist nur

*unter UV Licht sichtbar*



*Als letztes habe ich dann alles wieder zusammengebaut.*

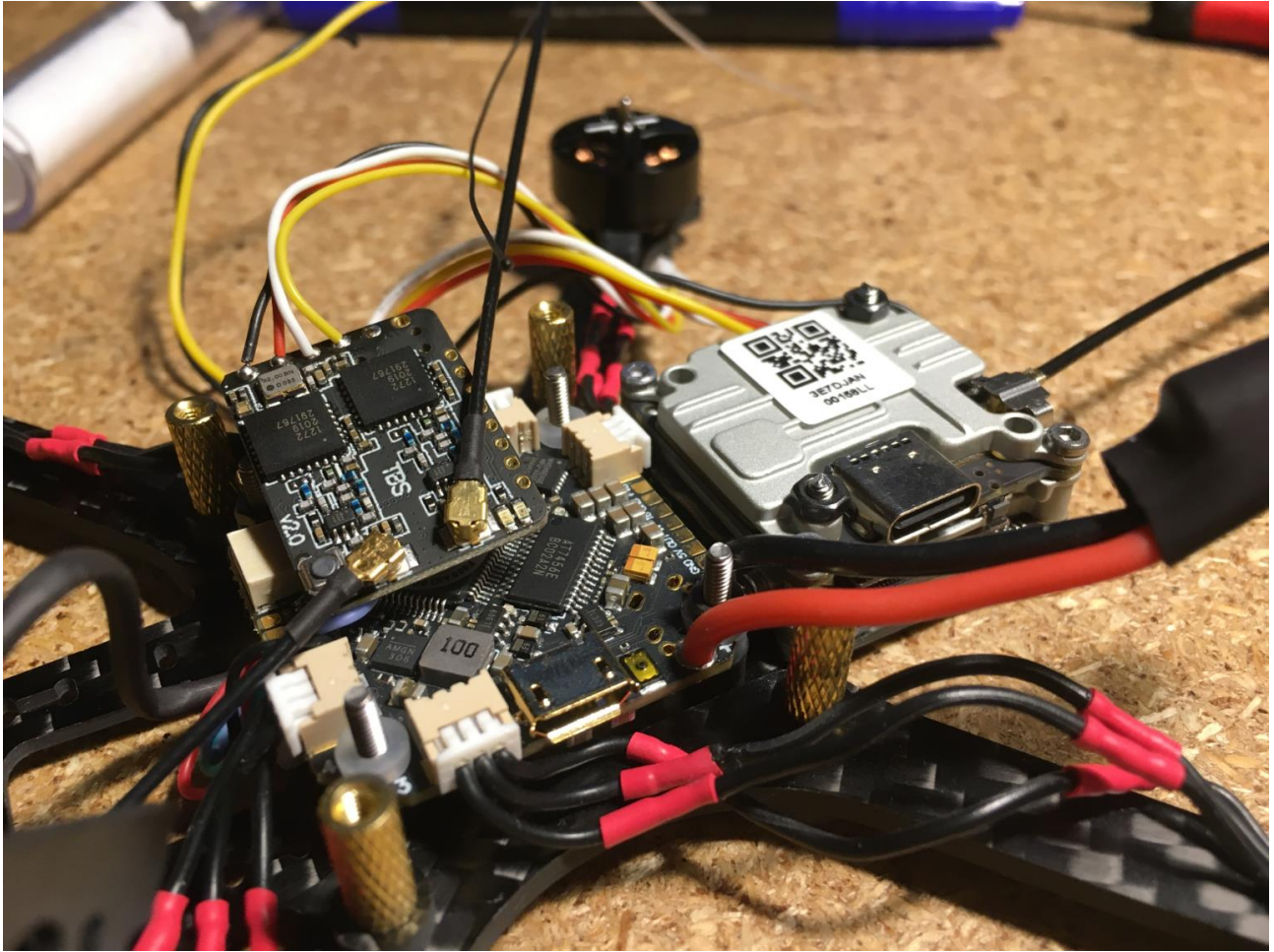
## Digital FPV

*Bevor ich die Software-Konfiguration erkläre, zeige ich noch kurz die Montage der Caddx Vista.*

*Die Kamera kommt natürlich ebenfalls nach vorne in die Halterung und die „Sendeinheit“ natürlich auch nach hinten. Der einzige Unterschied, was die Verkabelung angeht, ist eigentlich, dass die Vista einen „ganzen“ UART, also RX und TX und nicht 5V, sondern Lipo-Spannung benötigt. Dafür gibt es auch ein Pad am FC. Ich habe also an der Vista die vier Kabel des beim FC mitgelieferten Steckers für Batt+, GND und RX und TX angelötet. Als letztes habe ich auch hier noch die UFL-Antenne angesteckt, welche auch eine andere ist, als die, die ich für den analogen VTX verwendet habe, da die Antennen jeweils auf analoges- bzw. digitales Signal optimiert sind.*

*Ein letzter Unterschied zwischen analogem- und digitalem Copter ist, dass ich auf dem Digitalen statt einem Crossfire Nano einen Crossfire Diversity-Empfänger verbaut habe. Ein Diversity-Empfänger hat zwei Antennen, zwischen denen er hin- und her wechselt, je nachdem welche den besseren Empfang hat. Dies ist auch bei den FPV-Brillen der Fall.*





## Betaflight (Software)

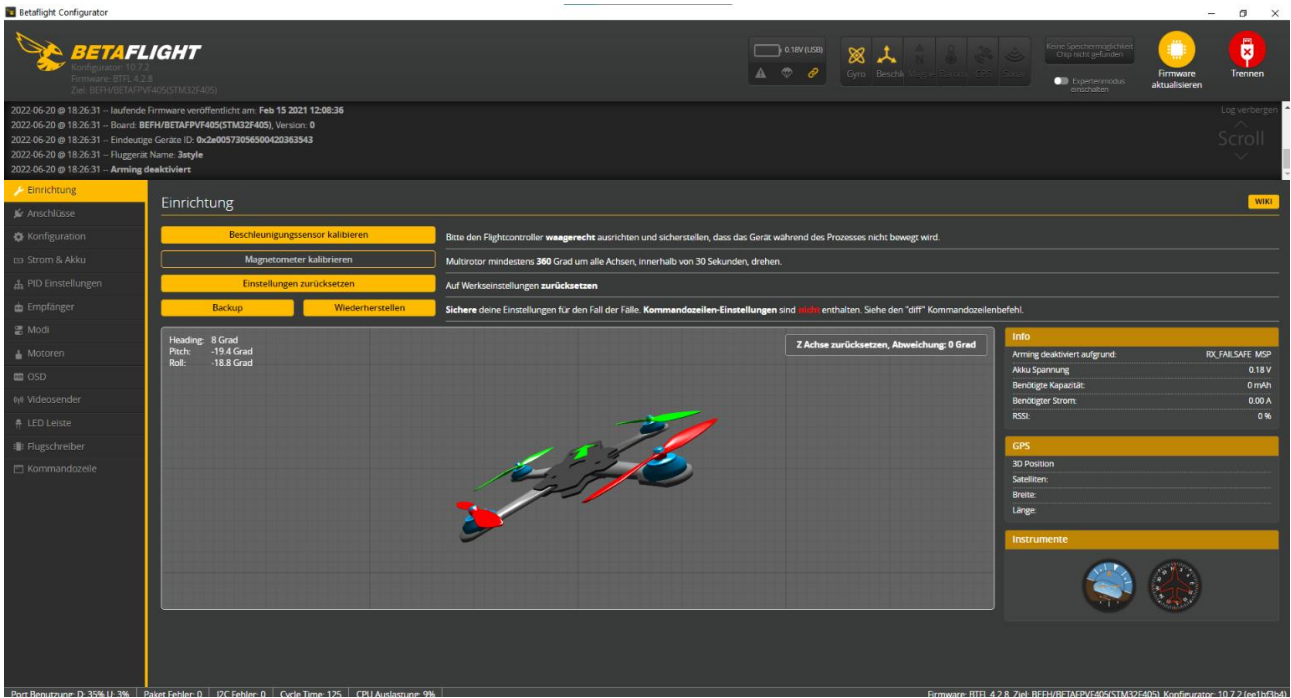
*Als letztes kommt nun nur noch die Software-Konfiguration in Betaflight.*

*Betaflight ist eine open Source Software für Drohnen und fixed Wings (Nurflügler).*

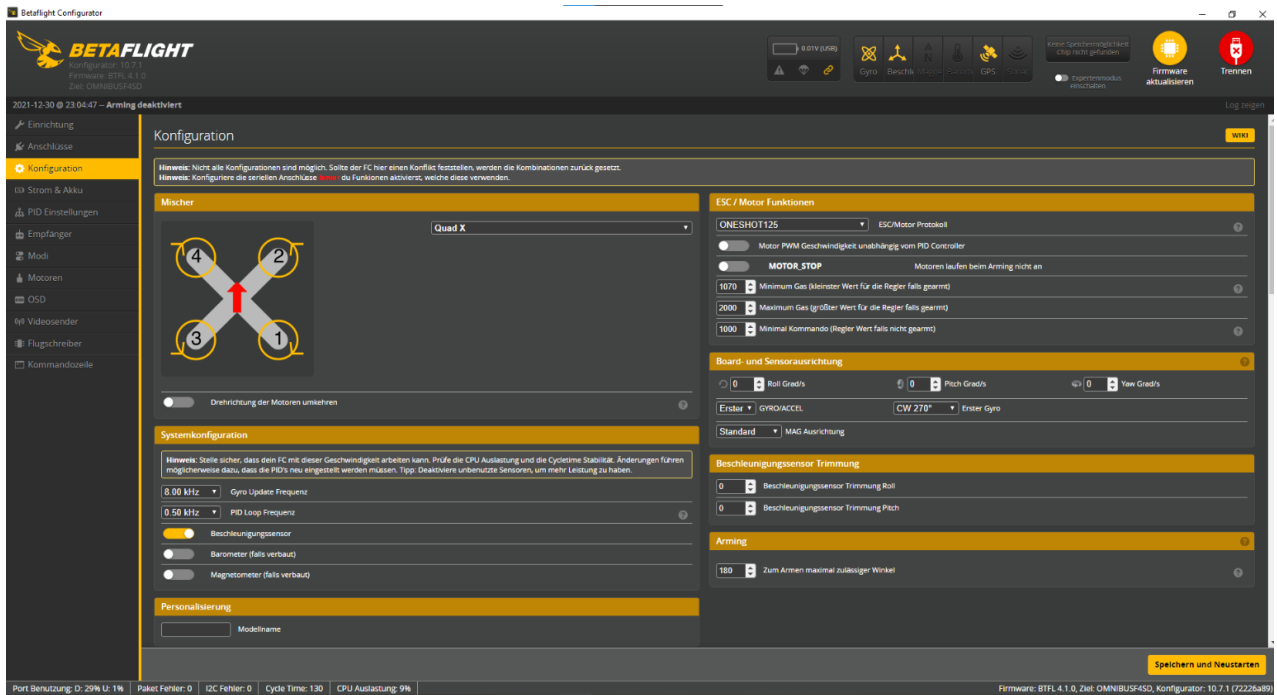
*Es gibt zwar auch noch andere Software für FPV-Drohnen, wie z.B. INAV (eher für Flächenmodelle, aber auch für Copter), Betaflight ist aber die, zumindest im FPV Copter Bereich, verbreitetste.*

*Die Software ist auf dem Copter für eigentlich alles verantwortlich, vom empfangen der Befehle vom Empfänger bis hin zum Ansteuern der Motoren.*

*Die gesamten Einstellungen können über den Betaflight-Configurator am PC, über eine Smartphone-App und einige sogar über das FPV-OSD oder über ein LUA-Script an der Funke vorgenommen werden.*



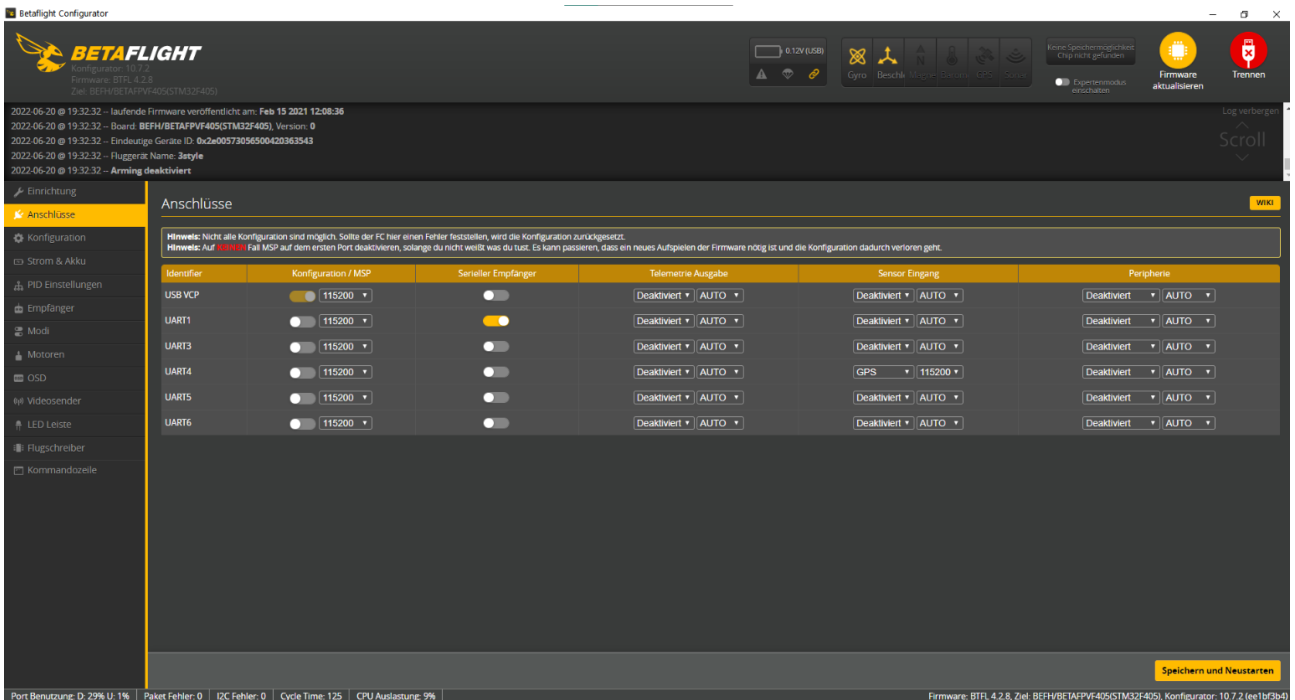
Die Software ist in mehrere Tabs eingeteilt, in denen man jeweils die Einstellungen einer Kategorie vornehmen kann. Als erstes habe ich den Gyro/Beschleunigungssensor kalibriert, dieser erfasst alle Bewegungen der Drohne.



Als nächstes habe ich dann im Konfiguration Tab das richtige Fernsteuerungsprotokoll ausgewählt (CRSF, also Crossfire) und den Arming Winkel auf 180 Grad eingestellt. Arming und disarmen bedeutet, die Motoren an- bzw. aus zu schalten. Bei einer FPV-Drohne sind die Motoren meistens so eingestellt, dass sie, wenn sie gerammt sind, immer mit ca. 3-5% ihrer maximalen Drehzahl drehen, damit der Copter

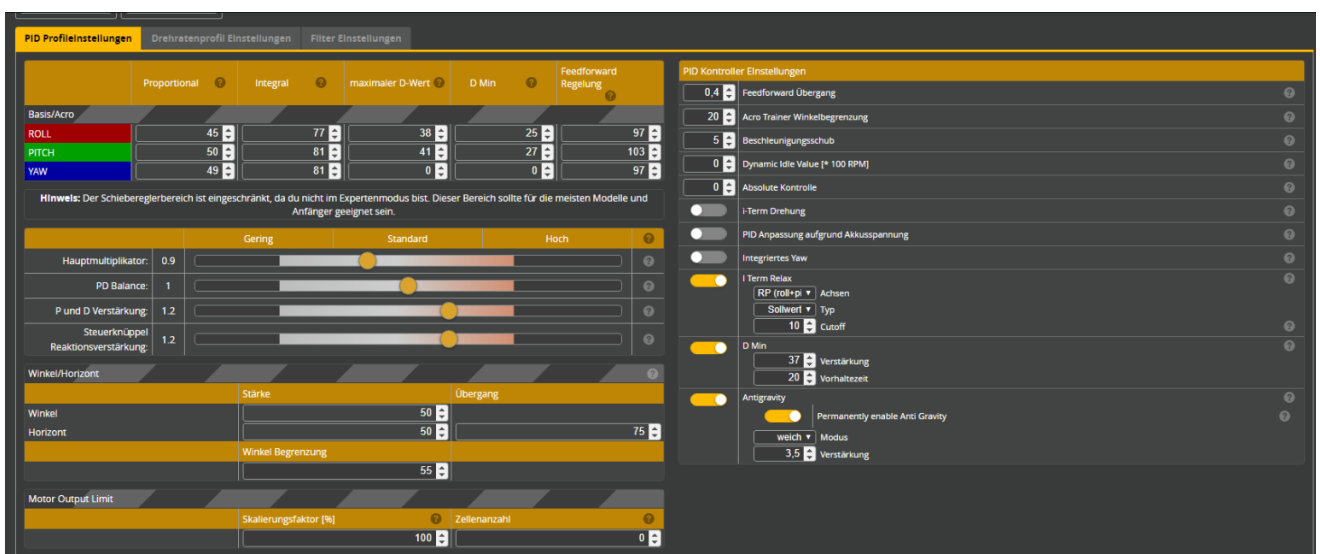


auch noch kontrollierbar ist, wenn man kein Gas gibt. Der Arming-Winkel ist der Winkel, den die Drohne maximal schief stehen darf, um armen zu können. Da es äußerst ungünstig wäre, wenn die Drohne falsch rum im Baum hängt und man sie nicht armen kann, damit sie von selber runterfällt, habe ich den Wert auf 180 Grad gestellt, damit sie immer gearmt werden kann.



Anschließend habe ich im Anschluss-Tab die richtigen UART's den richtigen Funktionen bzw. Protokollen zu geordnet:

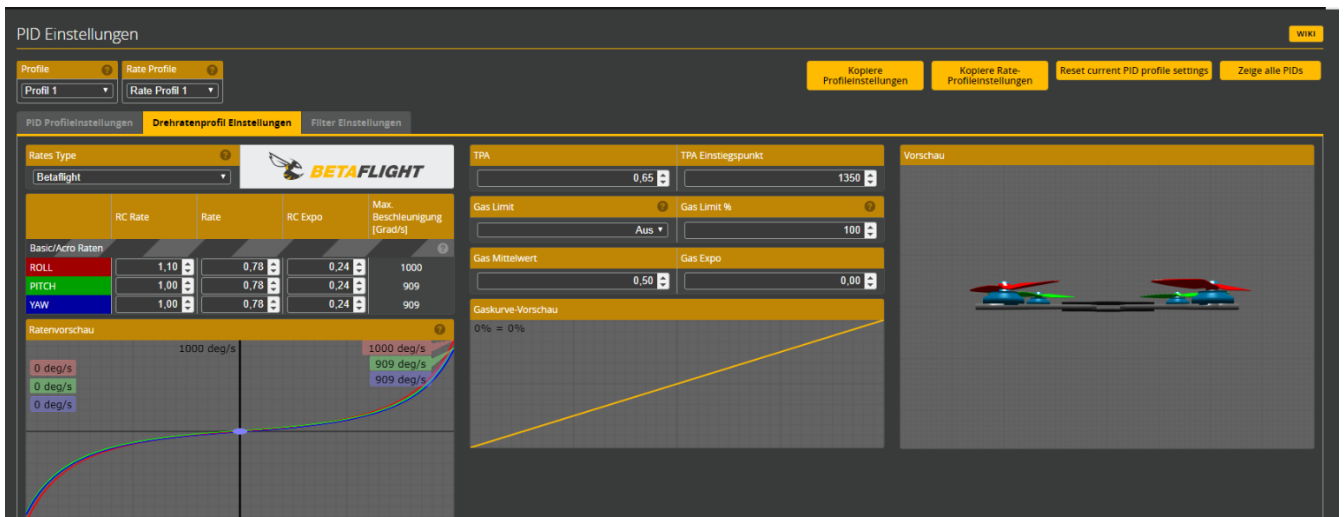
An UART 1 kam der Serielle Empfänger, also der Crossfire Nano bzw. Diversity und an UART 4 das GPS-Modul. Bei dem digitalen Copter habe ich außerdem noch einen UART für die Vista richtig eingestellt.



Dann ging es in den PID und Rates Tab. Die PID's sind vereinfacht gesagt dafür da,

in die Steuerbefehle einzugreifen und diese zu verfeinern/verbessern und, wenn die Drohne z.B. von einem Windstoß erfasst wird, gegen zu steuern, sodass der Copter weiterhin seine Flugbahn behält. Das ganze Thema ist sehr komplex, deshalb erkläre ich es hier nur sehr vereinfacht, ein gutes Video vom YouTube Kanal Spinfast gibt es unter folgendem Link: <https://youtu.be/25TWmtt33wY>. Die Infos zur Flugbahn bzw. Fluglage gibt der Gyro. Die PID's müssen bei jedem Copter individuell eingestellt werden. Ich habe allerdings die Standard-PID's für 3 Zoll-Copter verwendet, da diese relativ gut funktionieren und das ganze PID-Tuning-Thema ziemlich kompliziert bzw. aufwendig ist. Sind die PID's falsch eingestellt, fliegt der Copter schlecht bis gar nicht. Diese Erfahrung durfte ich schon bei dem digitalen Copter machen, da hatte ich nämlich vor dem Erstflug vergessen, die PID's richtig einzustellen. Das hat sich insofern geäußert, dass wenn man abgehoben ist, der Copter angefangen hat zu „wobbeln“ und dann unkontrolliert mit Vollgas nach oben geflogen ist. Disarmen hat ihm dann eine recht unsanfte Landung verschafft. Nach einstellen der richtigen PID's flog der Copter dann richtig.

Die Rates wiederum geben an, wie bzw. wie stark die Steuerbefehle in Bewegung umgesetzt werden. Ich habe meine Rates so eingestellt, das der Copter sich bei vollem Stick-Ausschlag an der Funke mit 1000 Grad pro Sekunde dreht. Dies ist fürs Freestyle fliegen ein guter Wert.



Als nächstes gings an den Empfänger-Tab, in dem ich nach dem binden des Crossfire-Empfängers mit der Funke kontrolliert habe, ob alle Kanäle richtig ankommen bzw. ob die richtige Channel-Map, also Kanalbelegung eingestellt ist. Da dies der Fall war konnte ich im Modes-Tab weitermachen und dort die verschiedenen Flug-Modi auf die Schalter an der Funke legen. Am wichtigsten ist da natürlich der Arming-Schalter, welchen ich auf den linken 2 Wege-Schalter gelegt habe.

Auf den rechten 3 Wege-Schalter habe ich die verschiedenen Flight-Modis gelegt.



Das sind natürlich der Acro-Mode, der Angle- und der Horizon-Mode. Im Acro-Modus fliegt sich die Drohne eigentlich wie ein Flugzeug, behält also die Eingegebene Richtung bei, auch wenn man den Stick loslässt. Im Angle- und im Horizon Modus hingegen kehrt die Drohne hingegen selbstständig mehr oder weniger in die Ausgangs-, also Schweben-Position zurück. Dies ist, wenn man das erste Mal eine (FPV) Drohne fliegt, zwar durchaus praktisch, wenn man aber schöne Filmaufnahmen machen möchte, sieht es einfach ziemlich doof aus, wenn die Drohne immer wieder in ihre „Normal“-Position zurück „kippt“.

Als letztes ging es dann an den OSD-Tab. Dort habe ich alle Infos, die ich im Flug angezeigt bekommen möchte, eingeschaltet und richtig positioniert.



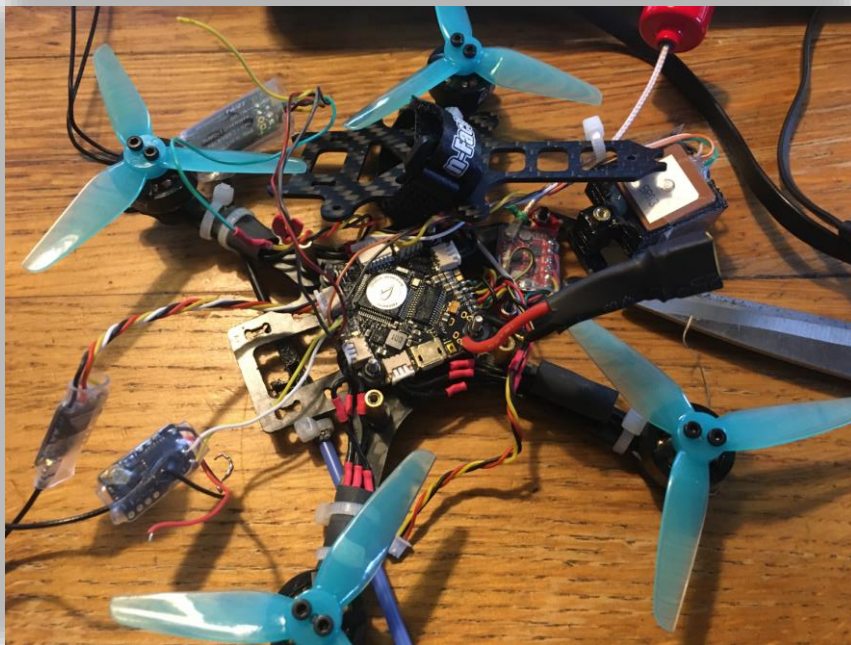
Dazu zählen z.B. Akku-Spannung, aktueller Stromverbrauch in Ampere, der bisher verbrauchte Strom in MAh, die Flugzeit, die Höhe und die Steigrate in Metern, die zurückgelegte Strecke pro Flug sowie die Entfernung zum Home Point, die Daten zur Empfangsstärke und die GPS-Koordinaten und einige weitere Daten.

Jetzt war in Betaflight eigentlich alles eingestellt und es ging an den ersten Probeflug mit dem analogen Copter.

## Erstflug des analogen Copters und Crossfire

Mein Vater und ich sind also an einem Wochenende auf den Modellflugplatz an der Babisnauer Pappel gefahren, da dort genug Platz für eventuelle Fehlfunktionen der Drohne ist. Ich habe dann also den Akku eingesteckt und es hat eigentlich alles funktioniert. Dann ist die Drohne auch normal abgehoben und auch gut geflogen – allerdings nur ca. 10 Meter, nach 10 Metern ist sie nämlich einfach aus der Luft gefallen. Schuld daran war ein sog. Fail Safe. Der Fail Safe ist dafür da, dass wenn der Copter die Verbindung zur Fernsteuerung verliert, nicht „ausrastet“ oder sonst wo hin fliegt, sondern disarmt und so einfach aus der Luft fällt.

Das Problem an der Sache war nur, dass es eigentlich nicht sein kann, dass der Copter mit einem Fernsteuerungssystem, mit dem ich auch schon weit über 500 Meter weit gekommen bin, einfach aus der Luft fällt. Das war das allernervigste am ganzen Copter-Bau. Ich hatte den Verdacht, dass das Problem mit dem Firmware-Update des Micro Tx zu tun haben könnte, zu dem ich durch TBS gezwungen wurde, da die neuen Crossfire-Empfänger nicht mehr mit der alten Firmware-Version kompatibel waren. Wären die neuen Empfänger mit der alten Firmware kompatibel gewesen, hätte ich natürlich nie ein Update gemacht, da es dabei, wie man sieht, oft zu Problemen kommen kann. Dazu gibt es auch einen Merksatz: „Never touch your running System“. Ich habe dann also eine Weile mit verschiedenen Firmware-Versionen experimentiert – keine hat funktioniert. In den Ferien sind wir dann



nach Hamburg, auf die Streuobstwiese meiner Großeltern, gefahren. Dort wollte ich natürlich gerne fliegen und filmen. Natürlich ist mir dort der Copter auch immer nach ca. 20 Metern aus der Luft gefallen. Ich habe dann noch mal eine andere Firmware-Version aufgespielt, was natürlich nichts ge-



bracht hat. Irgendwann ist mir dann der „Kragen geplatzt“ und ich habe einfach einen Graupner-Empfänger eingelötet, den ich zufällig dabei hatte.

## Aufnahmen und Gyroflow

Mit diesem Empfänger konnte ich dann immerhin gut fliegen und schöne Aufnahmen machen. Unter anderem habe ich mit der Drohne die Rehe gefilmt, die auf der Wiese leben. Wieder in Dresden habe ich dann noch ein kurzes 1,5-Minuten-Video aus den Aufnahmen geschnitten.

Leider war das Crossfire Problem immer noch nicht gelöst. Ich habe mich von Anfang an gewundert, warum seit dem Firmware-Update die Regions-Einstellungen nicht mehr angezeigt werden – weder auf der Handy-App noch auf dem PC oder im LUA-Script auf der Funke. Die Regions-Einstellungen sind wichtig, da man mit ihnen die Region, also die Sendefrequenz festlegt sowie die Sendeleistung einstellt. Ist die falsche Frequenz eingestellt, kann es sein, dass diese sich mit der Frequenz von z.B. Funkmasten überlagert, sodass der Empfang, wie bei mir, extrem schlecht wird. Ich habe dann also die Anleitung des Micro TX sowie das Internet nach einer Möglichkeit zum Freischalten der Regions-Einstellungen gesucht, jedoch ohne Erfolg.

Mir kam allerdings komisch vor, dass ich in den letzten Sommerferien schon mal einen Fail Safe nach nur 200 Metern hatte, obwohl ich zuvor am selben Ort genau 555 Meter weit gekommen bin. Damals ist mir die Drohne dann aus 99 Metern Höhe abgestürzt – das war gar nicht gut. Jedenfalls dachte ich dann, dass das Update und Config-Programm – der Agent X von TBS – doch eigentlich meine Region kennen muss und diese, wenn ich diese schon nicht selber einstellen darf, schon richtig eingestellt hat. Da der Fail Safe im Sommer vor dem Firmware-Update war und das Micro TX schon älter und gebraucht gekauft ist, dass es vielleicht einfach defekt ist. Also haben wir das Mico TX V2 gekauft – den Nachfolger des alten Moduls. Bei diesem musste ich natürlich auch das Update machen. Natürlich ging danach auch nichts mehr. Das war ziemlich frustrierend, da TBS eigentlich dafür bekannt ist, zwar teuer aber dafür auch gut zu sein und vor allem immer einfach zu funktionieren.

Irgendwann bin ich dann durch Zufall über ein Video bei YouTube „gestolpert“, in dem jemand gezeigt hat, wie man die Regions-Einstellungen wieder freischaltet. Dazu muss man genau 10 mal hintereinander die Taste hinten am Modul drücken. Das habe ich natürlich gleich ausprobiert und es hat zum Glück funktioniert. Ich habe dann die richtige Frequenz ausgewählt und den Graupner-Empfänger wieder gegen den Crossfire-Empfänger getauscht. Beim Test bei meiner Oma auf dem Feld hat dann zum Glück alles funktioniert. Zwar nicht mit der Reichweite die ich von

Crossfire kenne, das könnte aber auch an den zwei Funkmasten in Sichtweite und den vielen Hochspannungsleitungen, die in der Nähe vorbei führen, liegen, da diese durch ihr Magnetfeld den Empfang stören können. In Hamburg allerdings konnte ich ohne Failsafe einmal um die ganze Wiese fliegen bzw. war das FPV-Bild eher weg als der Crossfire-Empfang.

## Das GPS-Problem

Ein weiteres Problem waren die GPS-Module beider Copter. Diese haben nämlich keine Satelliten gefunden und haben somit nicht funktioniert. Ich habe dann natürlich die Kabel überprüft – diese waren aber in Ordnung. Als nächstes habe ich Betaflight die Baudraten überprüft – diese waren auch richtig eingestellt und nicht das Problem. Als nächstes habe ich dann wieder im Internet recherchiert. Dort habe ich dann in einem englischen FPV-Forum herausgefunden, dass es wohl eine fehlerhafte Produktion gab, bei denen die Module die angegebene Spannung von 5V nicht vertragen haben. Also habe ich meine GPS-Module vermutlich einfach „gegrillt“, also mit zu viel Spannung versorgt, da ich dem offiziellen Datenblatt getraut habe. Die beiden GPS-Module sind übrigens die M8 von TBS. Diese habe ich auch auf anderen Coptern im Einsatz und die haben noch nie Probleme gemacht. Am Ende haben wir dann nochmal 2 neue GPS-Module gekauft, welche ich dann erst nur mit 3,3V versorgt habe. 3,3V waren anscheinend aber zu wenig, also habe ich zwischen 5V Pad und GPS-Modul einfach einen kleinen Widerstand gelötet, das hat dann funktioniert. Als das GPS-Modul dann funktioniert hat, konnte ich dann bei Nieselregen endlich auch mal einen Geschwindigkeitstest machen. Höchstgeschwindigkeit mit den 3-Blatt Propellern ist 96 kmh. Mit den 5-Blatt Props bin ich mangels Zeit und Platz noch nicht zum Geschwindigkeitstest gekommen.

Mit den 5-Blatt Propellern fliegt der Copter allerdings viel besser und schneller als mit den 3-Blatt Propellern.

## Drohnen-Gesetze

Es gibt natürlich auch Gesetze, an die sich ein Drohnenpilot halten muss. Diese sind leider sehr komplex und unübersichtlich und widersprechen sich teilweise sogar. Im letzten Jahr wurden die Drohnen-Gesetze geändert. Um sie zu „vereinfachen“. Diese „Vereinfachung“ hat nur leider dazu geführt, dass man das Thema eigentlich schon fast Jahre lang studieren muss um halbwegs durchzusehen. So habe ich alleine auf der Webseite vom LBA (Luftfahrt Bundesamt) zwei mögliche Kategorien für meine Drohne gefunden. In einer Broschüre, mit der man seine Drohne besser einer



Kategorie zuordnen können soll, bin ich in Kategorie A1 gekommen, auf einer anderen Seite des LBA stand dann wieder was von wegen, dass die Drohne für A1 nicht schneller als 68 kmh fliegen darf – die Drohne fliegt allerdings deutlich schneller als 68 kmh. Nach dieser Einordnung lande ich dann also in der Kategorie A3, was die Orte, an denen ich fliegen darf, schon wieder stark einschränkt.

Am Ende hilft, finde ich zumindest, auch einfach der gesunde Menschenverstand, also das man natürlich nicht einfach über Privatgrundstücke etc. fliegen darf.

Am 9.7.22 durfte ich sogar auf bzw. über der Schulbaustelle fliegen und filmen. Das war auf jeden Fall super. Leider sind die Videos aufgrund des Windes nicht ganz so gut bzw. stabil geworden wie ich es mir gewünscht hätte. Hier aber auch nochmal vielen Dank an die Schulleitung und die beiden Hausmeister, dass ich überhaupt auf der Baustelle fliegen durfte! Vielleicht darf ich das nach den Sommerferien z.B. für die Dokumentation fürs Bautagebuch noch einmal machen.

Zusammenfassend kann man auf jeden Fall sagen, das mir das Projekt trotz nerviger Gesetze und technischer Schwierigkeiten wirklich sehr großen Spaß gemacht hat, dass ich viel gelernt habe und dass mein Vater und ich hoffentlich noch lange und möglichst ohne Abstürze viel Spaß mit den beiden Drohnen haben werden. Ich denke außerdem nicht, dass dies meine letzten Eigenbau-Drohnen sein werden.