

AI Drones

**A Bilingual Handbook on the Design, Construction, and Use of
AI-Enabled Drones for Scientific and Educational Projects**

KI-Drohnen

**Ein zweisprachiges Handbuch über Entwurf, Bau und Einsatz
von KI-fähigen Drohnen für wissenschaftliche Projekte und Lehre**

Christian Baun, Theodor Bloch, Matthias Deegener, Oliver Hahm, Martin Kappes, Nur Uddin Syeed

Frankfurt University of Applied Sciences

Preface

This document provides an introduction to the complex topic of drones with artificial intelligence. Its focus is on the development (including the selection of suitable hardware and software components), construction, and operation of drones for teaching and research purposes.

The insights presented in this document stem from the research project *AI-Assisted Drone Platform*, funded by the Connectom Networking and Innovation Fund of hessian.AI, as well as from the course *Drones with Artificial Intelligence* at the Frankfurt University of Applied Sciences.

Key criteria for selecting the components presented in this document include adaptability to different application scenarios, cost, robustness, long-term market availability, and the quality of documentation and manufacturer support.

The goal of this document is not to provide an exhaustive treatment of drones and AI, but rather to focus on technologies and solutions that were current at the time of its creation and for which practical experience was gained in the Computer Science program of Faculty 2 (Computer Science and Engineering) at the Frankfurt University of Applied Sciences.

Chapters 1 and 2 provide a concise overview of the hardware and software components required for building and operating FPV drones. For the AI projects presented here, drones primarily serve as tools and transport platforms. Chapter 3 contains helpful information for before and after the first flight.

From Chapter 4 onward, various AI applications are described that use FPV drones for

Vorwort

Dieses Dokument bietet einen Einstieg in das komplexe Thema Drohnen mit künstlicher Intelligenz. Schwerpunkte sind die Entwicklung (einschließlich der Auswahl geeigneter Hard- und Softwarekomponenten), der Bau und der Betrieb von Drohnen in der Lehre und für Forschungsprojekte.

Die in diesem Dokument dargestellten Erkenntnisse stammen aus dem vom Connectom Vernetzungs- und Innovationsfonds des hessian.AI geförderten Forschungsprojekt *KI-gestützte Drohnenplattform* sowie aus der Lehrveranstaltung *Drohnen mit Künstlicher Intelligenz* an der Frankfurt University of Applied Sciences.

Maßgebliche Kriterien bei der Auswahl der in diesem Dokument vorgestellten Komponenten sind unter anderem die Anpassbarkeit an unterschiedliche Einsatzszenarien, die Anschaffungskosten, die Robustheit, die langfristige Marktverfügbarkeit sowie die Qualität der Dokumentation und des Herstellersupports.

Eine vollständige Abhandlung der Themen Drohnen und KI ist nicht Ziel dieses Dokuments. Der Fokus liegt auf den Technologien und Lösungen, die zum Zeitpunkt der Erstellung aktuell waren und mit denen im Studienfeld Informatik des Fachbereichs 2 (Informatik und Ingenieurwissenschaften) der Frankfurt University of Applied Sciences praktische Erfahrungen gesammelt wurden.

Kapitel 1 und 2 geben einen kompakten Überblick über die Hard- und Softwarekomponenten, die für den Bau und den Betrieb von FPV-Drohnen erforderlich sind. In den hier vorgestellten KI-Projekten dienen die Drohnen in erster Linie als Werkzeug und als Transportplattform. Kapitel 3 enthält hilfreiche Hinweise vor und nach dem ersten Flug.

Ab Kapitel 4 werden verschiedene Anwendungen künstlicher Intelligenz vorgestellt, die FPV-

data collection and as transport platforms. The chapter presents applications developed, implemented, and evaluated in research projects, courses, and theses within the Computer Science unit at the Frankfurt University of Applied Sciences. Each project description includes the required additional hardware and software components, implementation steps, and associated costs.

The components typically used to build FPV drones are not capable of providing AI functionality on their own. Flight controllers lack the computing and memory resources required to run AI applications locally. Additional hardware and software are therefore necessary. These components may be integrated into the drone or remain on the ground to enable data processing and control. The respective chapters discuss different implementation approaches, their advantages and disadvantages, and the associated costs.

We greatly appreciate your comments and suggestions for improvement.

Drohnen zur Datenerfassung oder als Transportplattform nutzen. Vorgestellt werden Anwendungen aus Forschungsprojekten, Lehrveranstaltungen und Abschlussarbeiten der Lehreinheit Informatik der Frankfurt University of Applied Sciences. Jede Projektbeschreibung enthält Angaben zu den zusätzlich benötigten Hard- und Softwarekomponenten, zu den Umsetzungsschritten und zu den entstehenden Kosten.

Die üblicherweise zum Bau von FPV-Drohnen verwendeten Komponenten können keine Funktionalität im Bereich künstlicher Intelligenz bereitstellen. Flugcontroller verfügen nicht über die erforderlichen Rechen- und Speicherressourcen, um Anwendungen künstlicher Intelligenz lokal auszuführen. Zusätzliche Hard- und Software ist daher notwendig. Diese kann entweder in die Drohne integriert werden oder am Boden verbleiben, um Daten zu verarbeiten und die Drohne zu steuern. In den folgenden Kapiteln werden verschiedene Möglichkeiten der Realisierung mit ihren Vorteilen, Nachteilen und Kosten beschrieben.

Über Ihre Kommentare und Verbesserungsvorschläge freuen wir uns sehr.

Frankfurt am Main

*Christian Baun,
Theodor Bloch,
Matthias Deegener,
Oliver Hahm,
Martin Kappes,
Nur Uddin Syeed*

Contents

1 Hardware Components for Building FPV Drones	1
1.1 Frames	2
1.2 Flight Controller and Electronic Speed Controller	4
1.2.1 Stack or AIO	5
1.3 Motors	8
1.3.1 Size of the Motors	8
1.3.2 Electrical Voltage	9
1.3.3 KV Value	9
1.3.4 Propeller Mounting	10
1.3.5 Frame Attachment	11
1.4 Propeller	11
1.4.1 Propeller Size (Diameter)	12
1.4.2 Number of Blades	12
1.4.3 Pitch	13
1.5 Batteries	13
1.5.1 Charge and Discharge Rate (C-Rate)	14
1.5.2 Number of Cells (S4/S6)	15
1.5.3 Plugs (XT30/XT60/XT90) for Discharge and Charge	17
1.6 GPS	18
1.7 Receiver	19
1.8 Remote Control (Sender)	21
1.9 Video Transmitter	21
1.10 Camera	23
1.11 FPV Goggles	25
2 Software for Using FPV Drones	29
2.1 Flight Controller Firmware	29
2.1.1 Betaflight	30
2.1.2 INAV	41
2.1.3 ArduPilot	48
2.1.4 PX4	62
2.2 Remote Control Firmware: EdgeTX	63
2.3 Transmitter and Receiver Firmware: ExpressLRS	66
3 Important Things before the first Flight	71
3.1 Security during Flight	71
3.2 Security before Flight	71
3.3 Secure use of LiPo Batteries	71
3.4 Loading of Batteries	72
3.5 Secure Attachment of loose Cables and other Components	73
3.6 Use the Appropriate Flight Mode	74
3.7 Reduce the Power of the Video Transmitter or Switch It Off	75
3.8 Practice with a Simulator	76

3.9 Automate the Pairing	76
3.10 Monitoring telemetry data with suitable tools	77
4 Object Detection	79
4.1 Object Detection by additional Hardware on the Drone	80
4.1.1 Comparison of AI Accelerators	81
4.1.2 Construction and Implementation	84
4.1.3 Total Weight and Purchase Cost	85
4.2 Object Detection by using the Live Image on the Ground	86
4.2.1 Total Weight and Purchase Cost	90
5 Autopilot	93
5.1 Autopilot with Betaflight	93
5.2 Autopilot with iNAV	94
5.2.1 Navigation Modes in INAV	94
5.2.2 Configure Waypoint Missions with INAV	97
5.2.3 Loading Missions and INAV Stick Commands	99
5.3 Autopilot with ArduPilot	100
5.4 Autopilot with PX4	101
6 Follow-Me Function	103
6.1 Follow-Me with Betaflight	103
6.2 Follow-Me with iNAV	103
6.3 Follow-Me with ArduPilot	104
6.4 Follow-Me with PX4	104
7 Drop Mechanisms for FPV Drones	107
7.1 Servo-based release mechanisms	107
7.2 Electromagnetic mechanisms	108
7.3 Control and Integration	108
8 Stereo Camera Setup with a Raspberry Pi 5	111
8.1 Kamerakalibrierung	112
8.2 Stereo Calibration	119
8.3 Tiefenerkennung	121

Inhaltsverzeichnis

1 Hardware-Komponenten zum Bau von FPV-Drohnen	1
1.1 Rahmen	2
1.2 Flugcontroller und Motorsteuerung	4
1.2.1 Stack oder AIO	5
1.3 Motoren	8
1.3.1 Motoraufbau und -größe	8
1.3.2 Elektrische Spannung	9
1.3.3 KV-Wert	9
1.3.4 Propelleraufnahme	10
1.3.5 Rahmenbefestigung	11
1.4 Propeller	11
1.4.1 Propellergröße (Durchmesser)	12
1.4.2 Anzahl der Blätter	12
1.4.3 Steigung (Pitch)	13
1.5 Akkus	13
1.5.1 Lade- oder Entladerate (C-Wert)	14
1.5.2 Anzahl der Zellen (S4/S6)	15
1.5.3 Stecker (XT30/XT60/XT90) zum Entladen und Laden	17
1.6 GPS	18
1.7 Empfänger	19
1.8 Fernbedienung (Sender)	21
1.9 Videosender	21
1.10 Kamera	23
1.11 Videobrille (FPV-Brille)	25
2 Software zum Betrieb von FPV-Drohnen	29
2.1 Flight Controller Firmware	29
2.1.1 Betaflight	30
2.1.2 INAV	41
2.1.3 ArduPilot	48
2.1.4 PX4	62
2.2 Fernbedienung Firmware: EdgeTX	63
2.3 Sendemodul- und Empfänger-Firmware: ExpressLRS	66
3 Wichtige Punkte vor dem ersten Flug	71
3.1 Sicherheit beim Flug	71
3.2 Sicherheit vor dem Flug	71
3.3 Sichere Verwendung von LiPo-Akkus	71
3.4 Laden von Akkus	72
3.5 Sichere Befestigung loser Kabel und sonstiger Komponenten	73
3.6 Den geeigneten Flugmodus verwenden	74
3.7 Die Leistung des Videosenders reduzieren oder diesen abschalten	75
3.8 Mit einem Simulator üben	76

3.9 Die Kopplung automatisieren	76
3.10 Überwachung der Telemetriedaten mit geeigneten Werkzeugen	77
4 Objekterkennung	79
4.1 Objekterkennung mit zusätzlicher Hardware an der Drohne	80
4.1.1 Vergleich von KI-Beschleunigern	81
4.1.2 Aufbau und Implementierung	84
4.1.3 Gesamtgewicht und Anschaffungskosten	85
4.2 Objekterkennung durch Auswertung des Livebilds am Boden	86
4.2.1 Gesamtgewicht und Anschaffungskosten	90
5 Autopilot	93
5.1 Autopilot mit Betaflight	93
5.2 Autopilot mit iNAV	94
5.2.1 Navigationsflugmodi in INAV	94
5.2.2 Wegpunktmissionen mit INAV konfigurieren	97
5.2.3 Laden von Missionen und INAV Stickbefehle	99
5.3 Autopilot mit ArduPilot	100
5.4 Autopilot mit PX4	101
6 Follow-Me-Funktion	103
6.1 Follow-Me mit Betaflight	103
6.2 Follow-Me mit iNAV	103
6.3 Follow-Me mit ArduPilot	104
6.4 Follow-Me mit PX4	104
7 Drop-Mechanismen für FPV-Drohnen	107
7.1 Servo-basierte Abwurfmechanismen	107
7.2 Elektromagnetische Mechanismen	108
7.3 Ansteuerung und Integration	108
8 Stereokameras mit einem Raspberry Pi 5	111
8.1 Camera Calibration	112
8.2 Stereo-Kalibrierung	119
8.3 Depth Estimation	121

1

Hardware Components for Building FPV Drones

This chapter presents the most important hardware components required to build FPV drones. It does not aim to provide a comprehensive overview of the current state of technology. Likewise, recent developments in FPV drone technology are intentionally not addressed.

The goal of this chapter is to provide beginners with an easily understandable introduction to FPV drones. It aims to enable them to identify the components needed to design and build drones for their own projects. This provides the foundation for selecting and configuring the required software (see Chapter 2) and implementing AI functionality using (see Chapter 4, 5, and 6).

The typical components of an FPV drone without AI extensions are shown in Figure 1.1.

Hardware- Komponenten zum Bau von FPV-Drohnen

Dieses Kapitel stellt die wichtigsten Hardware-Komponenten zum Bau von FPV-Drohnen vor. Es erhebt nicht den Anspruch, einen vollständigen Überblick über den Stand der Technik zu geben. Auch die Entwicklung der FPV-Drohnen in den letzten Jahren wird bewusst ausgeklammert.

Ziel dieses Kapitels ist es, Einsteigern einen leicht verständlichen Zugang zum Thema FPV-Drohnen zu ermöglichen. Sie sollen in der Lage sein, die für ihre Projekte benötigten Komponenten zu identifizieren, um eigene Drohnen zu entwerfen und zu bauen. Dies ist die Voraussetzung für die Auswahl und Installation der notwendigen Software (siehe Kapitel 2) sowie die Realisierung KI-basierter Funktionen mit (siehe Kapitel 4, 5 und 6).

Die typischen Komponenten einer FPV-Drohne ohne KI-Erweiterung sind in Abbildung 1.1 dargestellt.

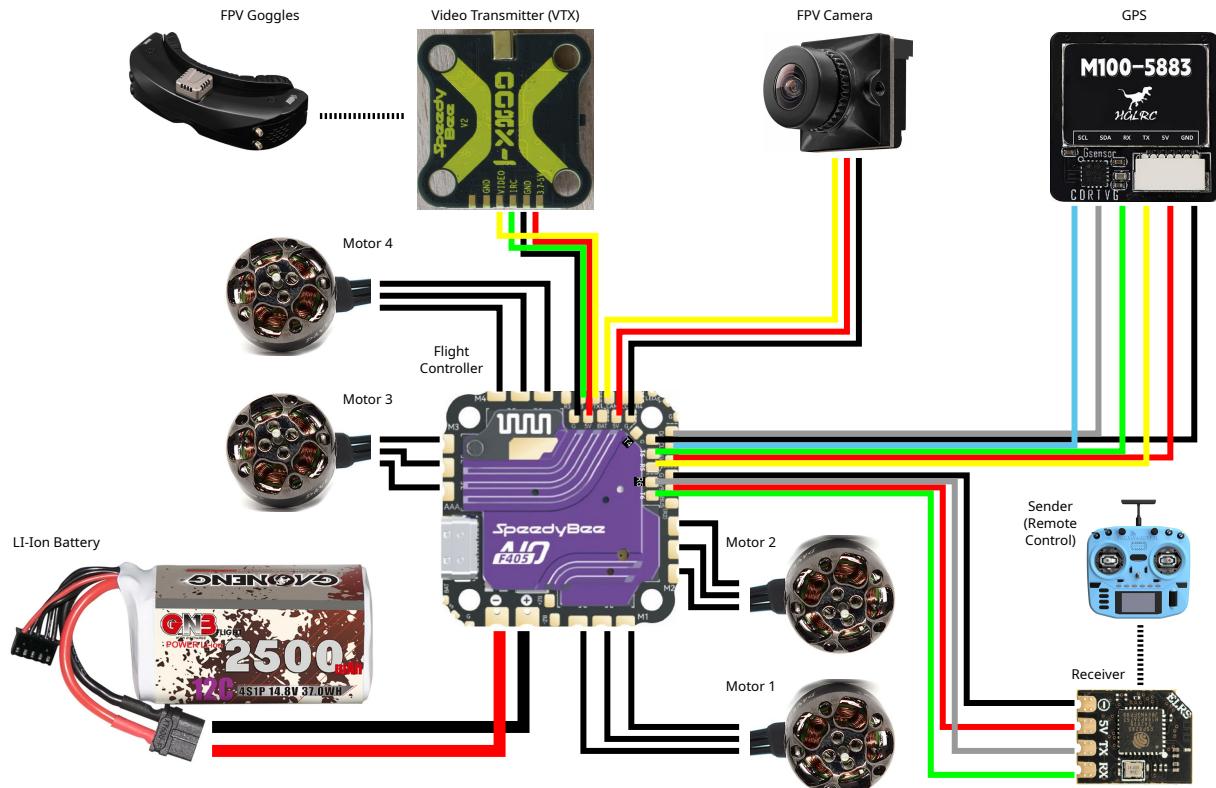


Figure 1.1: Components of a FPV Drone)

1.1

Frames

Der Rahmen verbindet alle Komponenten der Drohne miteinander. Das verwendete Material ist üblicherweise Carbon, ein leichtes, aber äußerst stabiles und verwindungssteifes Verbundmaterial aus Kohlenstofffasern. Seltener kommen Rahmen aus Kunststoff zum Einsatz. Der Rahmen bestimmt die Propellergröße (siehe Abschnitt 1.4).

Die wichtigsten elektronischen Komponenten wie Flugcontroller, Videosender, Empfänger und Kamera werden zentral auf dem Rahmen montiert, um sie zu schützen. Der Akku befindet sich in den meisten Fällen oben auf der Drohne, um Beschädigungen beim Landen zu vermeiden.

Größere Rahmen bieten mehr Platz für Komponenten und ermöglichen die Nutzung leistungsstärkerer Motoren (siehe Abschnitt 1.3) und größerer Propeller. Mit zunehmender Rahmengröße steigt jedoch auch das Gesamtgewicht. Gängige Rahmengrößen und deren typische Anwendungen sind in Tabelle 1.1 aufgeführt.

Rahmen

The frame connects all components of the drone. It is usually made of carbon fiber, a lightweight yet highly rigid and strong composite material. Frames made of plastic are less common. The frame determines the propeller size (see Section 1.4).

The main electronic components such as the flight controller, video transmitter, receiver, and camera are mounted centrally on the frame for protection. The battery is typically placed on top of the drone to prevent damage during landings.

Larger frames provide more space for components and allow the use of more powerful motors (see Section 1.3) and larger propellers. However, frame size also increases the overall weight. Common frame sizes and their typical applications are listed in Table 1.1.

Table 1.1: Overview of different Frame Sizes and Drone Categories

Drone Category	Typical Frame Size	Typical Propeller Diameter
TinyWhoop	1.2 - 2.5 Inch	31 - 64 mm
Toothpick	2 - 3 Inch	51 - 76 mm
CineWhoop	2.5 - 3.5 Inch	64 - 90 mm
Freestyle	3 - 6 Inch	76 - 152 mm
Racing	5 Inch	127 mm
Long-Range, Cinelifter	4 - 10 Inch	100 - 254 mm
Heavy-Lift	10 - 12 Inch	254 - 304 mm

Important distinguishing criteria when selecting an appropriate frame include the hole spacing for mounting the flight controller and the video transmitter. Common dimensions are:

- 30.5×30.5 mm
- 25.5×25.5 mm
- 20×20 mm

If a frame does not provide matching mounting holes for the selected flight controller and video transmitter, a 3D-printed adapter can help — provided there is sufficient space within the frame.

Another distinguishing feature of frames is their geometry. The most straightforward design is the True-X shape, where all arms are of equal length. Another variant is the Squashed-X shape, in which the arms are slightly angled to improve flight characteristics. The asymmetrical Deadcat design positions the front arms farther outward and slightly backward so that the propellers do not enter the camera's field of view. Other variants, such as the H-frame and box frame, aim to increase frame stability through additional reinforcements.

If drones are to be flown indoors or near people, propeller guards are highly recommended. They protect not only the propellers but also the surroundings. Drones intended for racing, freestyle, or long-range applications typically have no propeller guards, as they increase the overall weight. For TinyWhoops and CineWhoops, however, propeller protection is standard.

Wichtige Unterscheidungskriterien bei der Auswahl des passenden Rahmens sind auch die Abstände der Bohrlöcher zur Befestigung des Flugcontrollers und des Videosenders. Gängige Maße sind:

- $30,5 \times 30,5$ mm
- $25,5 \times 25,5$ mm
- 20×20 mm

Verfügt ein Rahmen nicht über passende Bohrlöcher für den ausgewählten Flugcontroller und den Videosender, kann ein per 3D-Drucker hergestellter Adapter helfen, sofern im Rahmen genügend Platz dafür vorhanden ist.

Ein weiteres Unterscheidungsmerkmal von Rahmen ist deren Geometrie. Die naheliegendste Form ist die True-X-Form, bei der alle Arme gleich lang sind. Eine weitere Variante ist die Squashed-X-Form (gequetschtes X), bei der die Arme leicht abgewinkelt sind, was sich positiv auf die Flugeigenschaften auswirken soll. Bei der asymmetrischen Form Deadcat sind die vorderen Arme weiter außen und nach hinten versetzt, damit die Propeller nicht in das Sichtfeld der Kamera hineinragen. Weitere Varianten sind unter anderem H-Frame und Box, die die Stabilität des Rahmens durch zusätzliche Verstrebungen verbessern sollen.

Wenn Drohnen in Innenräumen oder in der Nähe von Personen geflogen werden sollen, ist ein Propellerschutz sehr empfehlenswert. Er schützt nicht nur die Propeller, sondern auch die Umgebung. Drohnen für Racing-, Freestyle- und Long-Range-Anwendungen haben normalerweise keinen Propellerschutz, da er das Gesamtgewicht der Drohne erhöht. Bei TinyWhoops und CineWhoops ist ein Propellerschutz hingegen Standard.

1.2

Flight Controller and Electronic Speed Controller

The flight controller (FC) is the central component of every drone. It processes sensor data from the gyroscope, accelerometer, barometer, and GPS module. Using control algorithms, it stabilizes the drone's attitude and receives pilot commands through the receiver. In addition, the FC controls the motors via the electronic speed controllers (ESC).

The FPV camera and video transmitter are also connected to the FC, which supplies them with power from the battery and controls their operation. The FC additionally transmits telemetry data as an on-screen display (OSD) to the video transmitter.

Common flight controllers use an STM32 microcontroller, which also stores the firmware. Modern MCUs include the F4, F7, and H7 series. These differ in clock speed and memory capacity (see Table 1.2). The available processing power and memory limit which firmware versions can be used effectively.

Flugcontroller und Motorsteuerung

Der Flight Controller (FC) ist die zentrale Komponente jeder Drohne. Der FC verarbeitet die Sensordaten von Gyroskop, Beschleunigungssensor, Barometer und GPS-Modul. Er stabilisiert die Fluglage durch Regelalgorithmen und empfängt über den Empfänger die Steuerbefehle des Piloten. Zudem steuert der FC die Motoren über die Electronic Speed Controller (ESC).

Auch die FPV-Kamera und der Videosender sind mit dem FC verbunden. Er versorgt sie mit Strom aus dem Akku und steuert ihre Funktionen. Zudem überträgt der FC Telemetriedaten als On-Screen-Display (OSD) an den Videosender.

Gängige Flight Controller verwenden einen STM32-Mikrocontroller, auf dem auch die Firmware gespeichert ist. Moderne MCUs sind die Serien F4, F7 und H7. Sie unterscheiden sich in ihrer Taktfrequenz und Speicherkapazität (siehe Tabelle 1.2). Die vorhandenen Rechen- und Speicherressourcen begrenzen, welche Firmware-Versionen sinnvoll eingesetzt werden können.

Table 1.2: Overview of modern Flight Controller STM32 MCUs

CPU (MCU)	Clock	Flash Memory	RAM
F405	168 MHz	1 MB	192 kB
F411	100 MHz	512 kB	128 kB
F745	216 MHz	1 MB	320 kB
F722	216 MHz	512 kB	256 kB
H743	480 MHz	2 MB	1 kB

Betaflight (see Section 2.1.1) runs on all, and INAV (see Section 2.1.2) on almost all of the microcontrollers listed in Table 1.2. ArduPilot (see Section 2.1.3) requires at least 1 MB of flash memory for limited operation. The full range of functions is available only with 2 MB of flash memory.

The electronic speed controllers (ESCs) control the motors. The ESC is one of the most heavily stressed components of a drone, as it often handles continuous currents of 10 to 20 amperes.

Separate ESCs mounted as individual boards are now uncommon. In modern drones, four

Betaflight (siehe Abschnitt 2.1.1) funktioniert auf allen, und INAV (siehe Abschnitt 2.1.2) auf fast allen in Tabelle 1.2 genannten Mikrocontrollern. Der Betrieb von ArduPilot (siehe Abschnitt 2.1.3) erfordert für einen eingeschränkten Betrieb mindestens 1 MB Flash-Speicher. Erst mit 2 MB Flash-Speicher steht der vollständige Funktionsumfang zur Verfügung.

Die Electronic Speed Controller (ESCs) steuern die Motoren. Der ESC ist eines der am stärksten belasteten Bauteile der Drohne, da durch ihn häufig Dauerströme von 10 bis 20 Ampere fließen.

Separate ESCs als einzelne Platinen sind heute unüblich. Bei modernen Drohnen sind die

or more ESCs are combined on a single board—known as a 4-in-1 ESC—for motor control. The motors and the power supply connection to the battery (see Section 1.5) are soldered directly to this board.

1.2.1

Stack or AIO

The combination of a separate board for motor control and a flight controller is called a stack.

If the ESCs are located on the same board as the flight controller, the unit is referred to as an All-In-One Flight Controller (AIO FC).

In smaller drones (2 to 4 inches), AIO FCs are commonly used due to limited space. Larger drones typically use stacks. One advantage of stacks is that separate motor controllers with higher continuous current ratings are available on the market—typically between 45 and 70 A. In addition, FCs used in stacks generally offer more soldering pads, larger pads, often additional connectors, and more usable UART interfaces for sensors and actuators compared to AIO FCs.



vier oder mehr ESCs auf einer einzigen Platine – einem sogenannten 4-in-1-ESC – zur Motorsteuerung integriert. An dieser Platine werden die Motoren sowie die Verbindung zum Akku (siehe Abschnitt 1.5) angelötet.

Stack oder AIO

Die Kombination einer separaten Platine zur Motorsteuerung und dem Flugcontroller heißt Stack.

Befinden sich die ESCs auf der gleichen Platine wie der Flugcontroller, spricht man von einem All-In-One Flight Controller (AIO FC).

Bei kleineren Drohnen (2 bis 4 Zoll) sind schon aus Platzgründen AIO FCs üblich. Bei größeren Drohnen kommen meist Stacks zum Einsatz. Ein Vorteil von Stacks ist, dass separate Motorsteuerungen mit höheren Dauerströmen am Markt verfügbar sind – typischerweise zwischen 45 und 70 A. Zudem bieten die FCs von Stacks tendenziell mehr Platz für Lötpads, größere Lötpads, häufig zusätzliche Steckverbindungen und mehr nutzbare UART-Schnittstellen für Sensoren und Aktoren als AIO FCs.



Figure 1.2: Flight Controller SpeedyBee F405 AIO 40A Bluejay 25.5x25.5 (Front and Back)

Figure 1.2 shows an AIO flight controller. Figures 1.3 and 1.4 show the front and back sides of a stack consisting of a flight controller and a separate ESC board.

Abbildung 1.2 zeigt einen AIO-Flight-Controller. Die Abbildungen 1.3 und 1.4 zeigen die Vorder- und Rückseite eines Stacks, bestehend aus Flight-Controller und separater ESC-Platine.

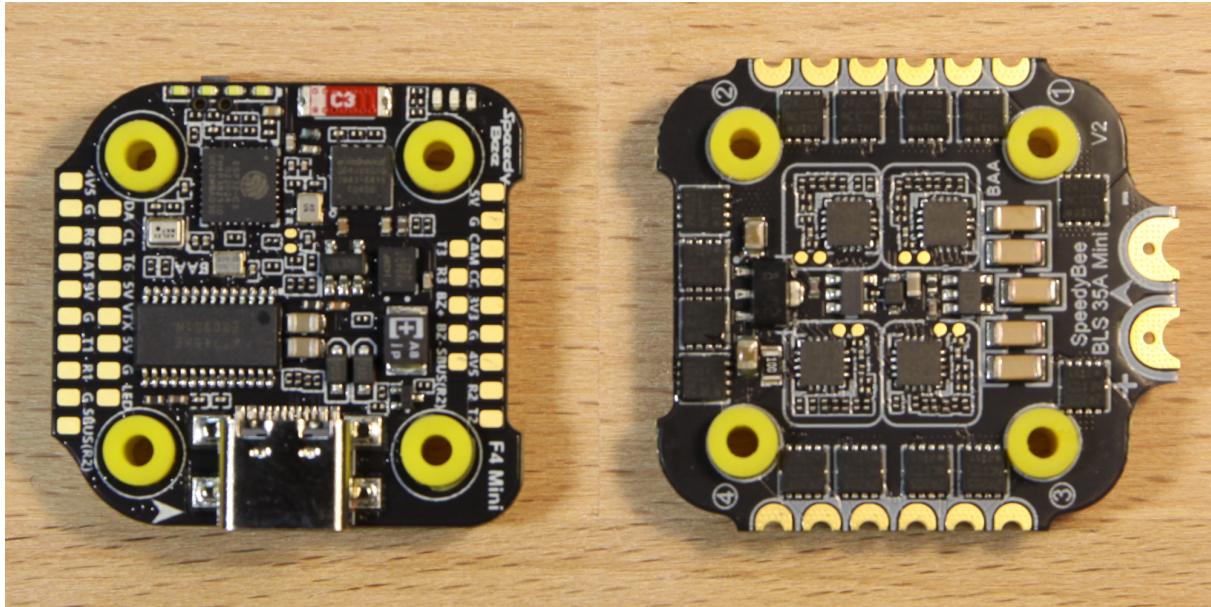


Figure 1.3: Flight Controller SpeedyBee F405 Mini and BSC 35A 20x20 Stack (Front)

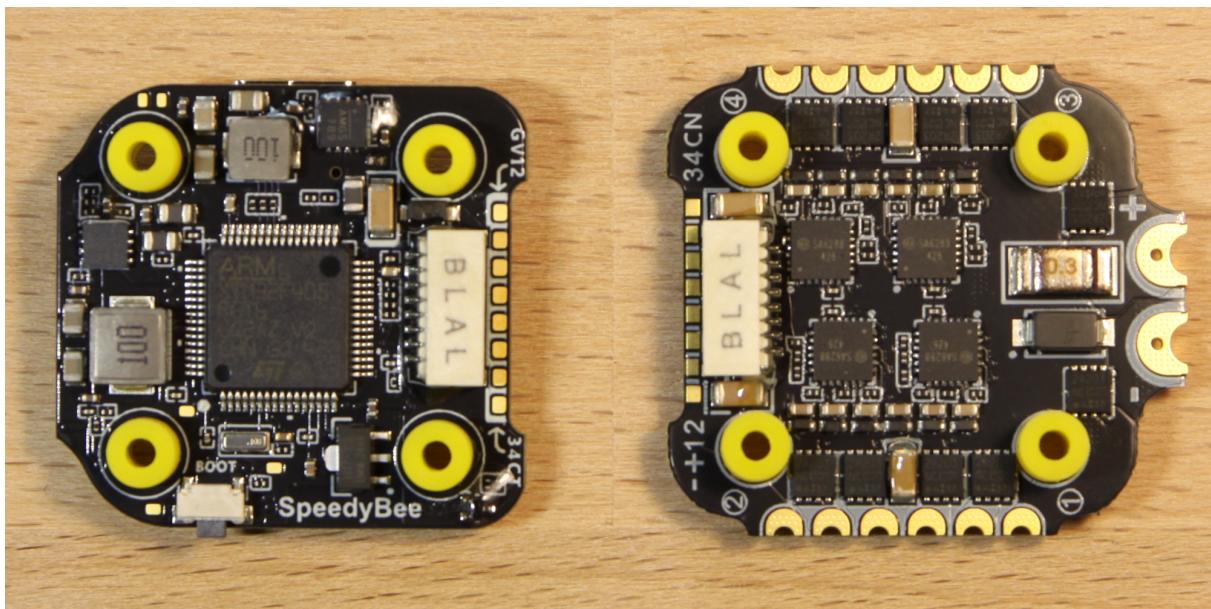


Figure 1.4: Flight Controller SpeedyBee F405 Mini and BSC 35A 20x20 Stack (Back)

An important selection criterion for flight controllers is the number of usable UART interfaces for connecting sensors and actuators. The flight controller shown in Figure 1.2, the SpeedyBee F405 AIO, has six UART interfaces. However, one is used for Wi-Fi and USB administration, and another operates only unidirectionally (simplex), meaning it can only receive data. This leaves only four fully functional (bidirectional) UART interfaces, of which three are typically occupied by the video transmitter, receiver, and GPS module. Therefore, only one free UART interface usually remains.

Ein wichtiges Auswahlkriterium für Flugcontroller ist die Anzahl der nutzbaren UART-Schnittstellen zum Anschluss von Sensoren und Aktoren. Der in Abbildung 1.2 gezeigte Flugcontroller SpeedyBee F405 AIO verfügt über sechs UART-Schnittstellen. Allerdings wird eine für die WLAN- und USB-Schnittstellen zur Administration benötigt und eine weitere funktioniert nur unidirektional (Simplex), das heißt, sie kann nur Daten empfangen. Damit stehen lediglich vier vollwertige (bidirektionale) UART-Schnittstellen zur Verfügung, von denen üblicherweise drei durch die Komponenten Video-

The flight controller shown in Figures 1.3 and 1.4, the SpeedyBee F405 Mini 35A Stack, also has six UART interfaces. Here, too, after connecting the commonly used components — video transmitter, receiver, and GPS module — only one interface remains available. This is because one UART is used by the Bluetooth interface for administration, and another is used by the ESC for telemetry data transmission.

For comparison, the flight controller shown in Figure 1.5, the Flywoo GOKU GN745 AIO 45A, offers seven UART interfaces. None of them is used for USB, but one is reserved for telemetry data transmission. After connecting the common components — video transmitter, receiver, and GPS module — three UART interfaces remain available for additional sensors and actuators.

sender, Empfänger und GPS-Modul belegt sind. Somit bleibt meist nur eine einzige freie UART-Schnittstelle.

Auch der in den Abbildungen 1.3 und 1.4 gezeigte Flugcontroller SpeedyBee F405 Mini 35A Stack verfügt über sechs UART-Schnittstellen. Auch hier bleibt nach dem Anschluss der üblichen Komponenten — Videosender, Empfänger und GPS-Modul — nur eine Schnittstelle frei verfügbar, da die Bluetooth-Schnittstelle zur Administration eine UART-Schnittstelle belegt und die Motorsteuerung eine weitere UART-Schnittstelle zur Übertragung der Telemetriedaten nutzt.

Zum Vergleich: Der in Abbildung 1.5 gezeigte Flugcontroller Flywoo GOKU GN745 AIO 45A verfügt über sieben UART-Schnittstellen. Keine davon ist für die USB-Schnittstelle belegt, jedoch wird eine für die Übertragung von Telemetriedaten verwendet. Nach dem Anschluss der üblichen Komponenten — Videosender, Empfänger und GPS-Modul — bleiben somit drei UART-Schnittstellen für weitere Sensoren und Aktoren verfügbar.

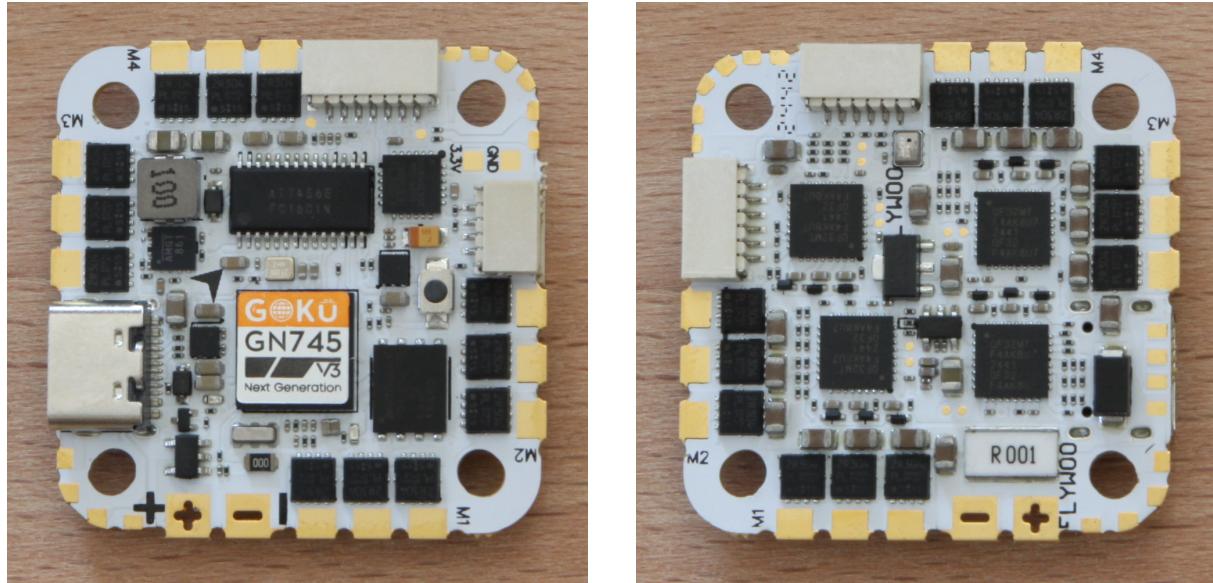


Figure 1.5: Flight Controller Flywoo GOKU GN745 AIO 45A 25.5x25.5 (Front and Back)

Table 1.3 shows the typical use of the available UART interfaces of selected flight controllers and the resulting limitations regarding their suitability for AI projects, which often require connecting additional sensors and actuators.

Tabelle 1.3 zeigt die typische Nutzung der verfügbaren UART-Schnittstellen einiger ausgewählter Flugcontroller und die damit einhergehenden Einschränkungen hinsichtlich ihrer Eignung für KI-Projekte, die häufig den Anschluss weiterer Sensoren und Aktoren erfordern.

Table 1.3: Recommended use and availability of UART Interfaces of some Flight Controllers

Flight Controller	UART1	UART2	UART3	UART4	UART5	UART6	UART7	UART8
SpeedyBee F405 AIO	MSP ^a	available ^b	VTX	available	GPS	Receiver	—	—
SpeedyBee F405 Mini Stack	VTX	Receiver	available	MSP ^a	Telem. ^c	GPS	—	—
SpeedyBee F4V3 / F4V4 Stack	VTX	Receiver	available	MSP ^a	Telem. ^c	GPS	—	—
Flywoo F722 PRO MINI V2	available ^b	Receiver	VTX	available	GPS	Telem. ^c	—	—
Flywoo F722 PRO V2	available ^b	Receiver	VTX	available	GPS	Telem. ^c	—	—
GEPRC F722 35A AIO	VTX	Receiver	GPS	available	available	—	—	—
SpeedyBee F7V3 Stack	VTX	Receiver	available	Telem. ^c	—	GPS	—	—
Flywoo GN745 AIO V3	available	Receiver	VTX	available	available	GPS	Telem. ^c	—
Axisflying F745 40A AIO AM32	VTX	Receiver	GPS	?	available	available	available	available
MicoAir H743 V2 45A AIO	available	VTX	GPS	available	available	Receiver	Telem. ^c	available
T-Motor PACER H743 Stack	available	GPS	available	Telem. ^c	Receiver	VTX	MSP ^a	available

^a MSP = Port for administration via the MultiWii Serial Protocol using Bluetooth and WIFI if available.

^b Offers unidirectional (simplex) communication. It can only receive data.

^c Receives telemetry data from the ESC.

1.3

Motors

Motors differ in their design (construction and size), the operating voltage (in volts), rotational speed (KV rating), propeller mount, and the method of attachment to the frame.

Motoren

Motoren unterscheiden sich in ihrem Aufbau (Konstruktion und Größe), der elektrischen Spannung (in Volt), mit der sie betrieben werden, der Drehzahl (KV-Wert), der Aufnahme des Propellers und der Befestigung am Rahmen.

1.3.1

Size of the Motors

The motors used in FPV drones are three-phase, brushless DC motors. Each motor has three connection wires that are soldered to the electronic speed controller (ESC) or directly to an AIO flight controller. The ESC sends time-shifted alternating currents through these three wires. By switching the current phases, a rotating magnetic field is created, which drives the rotor equipped with permanent magnets. The three wires are equivalent, so the order in which they are soldered to the ESC or flight controller does not matter. The motor's rotation direction can be changed by swapping any two of the three wires. Alternatively, the direction can be checked and reversed in the firmware's graphical interface for each motor individually.

The motors consist of a stator and a rotor. The stator contains the windings and the bearing, while the rotor—the motor bell—carries

Motoraufbau und -größe

Die Motoren von FPV-Drohnen sind dreiphasige, bürstenlose Gleichstrommotoren. Jeder Motor besitzt drei Anschlusskabel, die an die Motorsteuerung (ESC) oder direkt an einen AIO-Flugcontroller angelötet werden. Die Motorsteuerung schickt zeitlich versetzte Wechselströme in diese drei Leitungen. Durch das Umschalten der Ströme entsteht ein rotierendes Magnetfeld, das den Rotor mit Permanentmagneten in Bewegung versetzt. Die Leitungen sind gleichwertig – die Reihenfolge, in der sie angelötet werden, spielt keine Rolle. Die Drehrichtung des Motors kann durch Vertauschen von zwei beliebigen der drei Kabel geändert werden. Alternativ lässt sich die Drehrichtung in der grafischen Benutzeroberfläche der Firmware für jeden Motor separat prüfen und anpassen.

Die Motoren bestehen aus einem Stator und einem Rotor. Der Stator enthält die Wicklungen und das Kugellager, während der Rotor –

the magnets and performs the actual rotation. Each motor is identified by a numerical code that specifies the stator's diameter and height in millimeters. For example, a motor labeled *2306* has a stator with a diameter of 23 mm and a height of 6 mm.

The larger the stator volume, the higher the motor's torque and overall performance. Larger motors also have better thermal robustness. Smaller motors, by contrast, are lighter and more efficient under low loads.

Propellers (see Section 1.3.4) are typically mounted on an M5 shaft with a 5 mm diameter using a matching nut, or alternatively on a 1.5 mm shaft secured with two small M2 screws.

die Motorglocke – die Magnete trägt und die eigentliche Drehbewegung ausführt. Jeder Motor ist mit einer Zahlenkombination gekennzeichnet, die den Durchmesser und die Höhe des Stators in Millimetern angibt. Ein Motor mit der Bezeichnung *2306* besitzt beispielsweise einen Stator mit 23 mm Durchmesser und 6 mm Höhe.

Je größer das Statorvolumen, desto höher sind Drehmoment und Gesamtleistung des Motors. Größere Motoren sind zudem thermisch robuster, während kleinere Motoren leichter und bei geringerer Last effizienter arbeiten.

Die Propelleraufnahme (siehe Abschnitt 1.3.4) erfolgt in der Regel auf einer M5-Welle mit 5 mm Durchmesser und einer passenden Mutter oder alternativ auf einer 1,5 mm-Welle, die mit zwei kleinen M2-Schrauben am Motor befestigt wird.

1.3.2

Electrical Voltage

Motors and batteries (see Section 1.5) must be compatible. 4S motors require 4S batteries with a maximum voltage of 16.8 V, while 6S motors require 6S batteries with a maximum voltage of 25.2 V.

4S components are usually cheaper and lighter but provide less power. 6S components offer more power or longer flight time but are generally heavier – especially the batteries. For lightweight and cost-effective drones, 4S is often the better choice. Some motors can handle both 4S and 6S voltages, making them more versatile in use.

Elektrische Spannung

Motoren und Akkus (siehe Abschnitt 1.5) müssen zusammenpassen. 4S-Motoren benötigen 4S-Akkus mit einer maximalen Spannung von 16,8 V, während 6S-Motoren 6S-Akkus mit maximal 25,2 V benötigen.

Komponenten für 4S sind meist günstiger und leichter, bieten dafür aber weniger Leistung. 6S-Komponenten liefern mehr Leistung oder längere Flugzeiten, sind jedoch in der Regel schwerer – insbesondere die Akkus. Für leichte und kostengünstige Drohnen ist 4S häufig die bessere Wahl. Es gibt auch Motoren, die sowohl mit 4S- als auch mit 6S-Akkus betrieben werden können und dadurch flexibler einsetzbar sind.

1.3.3

KV Value

The KV value describes how fast a motor spins per volt under no load, measured in revolutions per minute (RPM). For example, a 3000 KV motor spins at 3000 RPM per volt. When powered by a 4S battery (see Section 1.5) with approximately 16 V, this motor would have a no-load speed of about $3000 \times 16 \approx 48000$ RPM.

Motors with high KV values (2800–7000 KV) are more agile and provide faster response times

KV-Wert

Der KV-Wert beschreibt, wie schnell sich ein Motor im Leerlauf pro Volt dreht, gemessen in Umdrehungen pro Minute (U/min). Ein Motor mit beispielsweise 3000 KV dreht also bei 1 V Versorgungsspannung 3000 U/min. An einem 4S-Akku (siehe Abschnitt 1.5) mit etwa 16 V Spannung hat dieser Motor somit eine Leerlaufdrehzahl von ungefähr $3000 \times 16 \approx 48000$ U/min.

Motoren mit hohen KV-Werten (2800–7000 KV) sind agiler und ermöglichen schnellere Re-

but generate less torque per ampere. They are well suited for smaller propellers and lighter drones. Motors with lower KV values (1500–2450 KV) deliver higher torque per ampere and are better suited for larger propellers and heavier drones.

A higher KV value does not mean that a motor is more powerful than one with a lower KV value. It simply means the motor spins faster, but also draws more current to produce thrust.

1.3.4

Propeller Mounting

Larger motors (e.g., 22xx, 23xx, 24xx) for drones of 5 inches and above typically feature an M5 shaft with a 5 mm diameter for propeller mounting (see Figure 1.6). The propellers are placed directly onto the motor shaft and secured with a nut.

Smaller motors (e.g., 13xx, 14xx, 18xx) for drones up to 3.5 inches usually have much smaller shafts of 1.5 mm in diameter, where the propellers are attached with two screws (see Figure 1.7). Motors for medium-sized drones (e.g., 20xx, 21xx) with 3.5 or 4-inch propellers are available in both M5 and 1.5 mm shaft variants.



aktionen, erzeugen jedoch ein geringeres Drehmoment pro Ampere. Sie eignen sich gut für kleinere Propeller und leichtere Drohnen. Motoren mit niedrigen KV-Werten (1500–2450 KV) liefern ein höheres Drehmoment pro Ampere und sind für größere Propeller und schwerere Drohnen besser geeignet.

Ein hoher KV-Wert bedeutet nicht, dass ein Motor stärker ist als ein Motor mit einem niedrigeren KV-Wert. Er dreht lediglich schneller, benötigt dafür aber auch mehr Strom, um denselben Schub zu erzeugen.

Propelleraufnahme

Größere Motoren (z. B. 22xx, 23xx, 24xx) für Drohnen ab 5 Zoll besitzen zur Propelleraufnahme meist eine M5-Welle mit 5 mm Durchmesser (siehe Abbildung 1.6). Die Propeller werden direkt auf die Motorwelle gesteckt und mit einer Mutter gesichert.

Kleinere Motoren (z. B. 13xx, 14xx, 18xx) für Drohnen bis 3,5 Zoll haben in der Regel eine deutlich kleinere Welle mit 1,5 mm Durchmesser, an der die Propeller mit zwei Schrauben befestigt werden (siehe Abbildung 1.7). Motoren (z. B. 20xx, 21xx) für mittelgroße Drohnen (3,5 oder 4 Zoll) sind sowohl mit M5-Wellen als auch mit 1,5 mm-Wellen erhältlich.



Figure 1.6: T-Motor 2306 KV2400 Motor with an M5 Shaft



Figure 1.7: T-Motor 1604 KV2850 4-6S Motor with a 1.5 mm Shaft

1.3.5

Frame Attachment

When selecting motors, it is important to ensure that the mounting holes on the frame match in number, spacing, and diameter. Motors are typically attached to the frame using three or four screws (M1.4, M2, or M3). On smaller frames, the holes are arranged in an equilateral triangle or a square pattern. Common mounting hole distances are 7 mm, 9 mm, 12 mm, 16 mm, or 19 mm.

The motor shown in Figure 1.6 has a hole spacing of 12 mm, while the motor shown in Figure 1.7 has a hole spacing of 7 mm.

Rahmenbefestigung

Bei der Auswahl der Motoren ist darauf zu achten, dass die Bohrlöcher des Rahmens in Anzahl, Abstand und Durchmesser übereinstimmen. Üblicherweise werden die Motoren mit drei oder vier Schrauben (M1.4, M2 oder M3) am Rahmen befestigt. Bei kleinen Rahmen sind die Schraubenlöcher in Form eines gleichseitigen Dreiecks oder als Quadrat angeordnet. Gängige Lochabstände sind 7 mm, 9 mm, 12 mm, 16 mm oder 19 mm.

Der in Abbildung 1.6 gezeigte Motor weist einen Lochabstand von 12 mm auf, während der in Abbildung 1.7 dargestellte Motor einen Lochabstand von 7 mm besitzt.

1.4

Propeller

Propellers convert the motor's rotational movement into thrust (lift) and control forces. Their diameter, pitch, and number of blades influence thrust, efficiency, noise level, and flight behavior. This document considers only drones with four propellers, although other configurations such as hexacopters and octocopters are also possible.

Propeller

Die Propeller wandeln die Drehbewegung des Motors in Schub (Lift) und Steuerkräfte um. Propellergröße (Durchmesser), Steigung (Pitch) und Blätterzahl beeinflussen Schub, Effizienz, Geräuschentwicklung und Flugverhalten. Dieses Dokument berücksichtigt ausschließlich Drohnen mit vier Propellern, auch wenn andere Konfigurationen wie Hexacopter und Octocopter ebenfalls möglich sind.

1.4.1

Propeller Size (Diameter)

The propeller size determines, in simple terms, how much air is moved. The specified propeller size always refers to the diameter in inches. The propellers must fit the frame used (see Section 1.1).

Smaller propellers make the drone more agile, allowing quicker responses to control inputs. This is because smaller propellers have less air resistance and lower rotational mass (moment of inertia). As a result, the motor can increase or decrease its rotational speed faster. Since smaller propellers are typically used with smaller motors that have a higher KV rating, they also spin faster, which further enhances agility and responsiveness.

Larger propellers generate more thrust and are more efficient. They also provide smoother flight characteristics but at the cost of agility. As propeller size increases, so does the load on the motors and ESCs, since the larger blade area produces more air resistance. In addition, larger propellers are heavier.

Propellergröße (Durchmesser)

Die Propellergröße bestimmt vereinfacht gesagt, wie viel Luft bewegt wird. Die angegebene Propellergröße entspricht immer dem Durchmesser in Zoll. Die Propeller müssen zum verwendeten Rahmen passen (siehe Abschnitt 1.1).

Kleinere Propeller machen die Drohne wendiger und ermöglichen schnellere Reaktionen auf Steuerbefehle. Der Grund dafür ist, dass kleinere Propeller weniger Luftwiderstand haben und insgesamt leichter sind. Sie besitzen eine geringere Rotationsmasse (Trägheitsmoment), wodurch der Motor die Drehzahl schneller erhöhen oder verringern kann. Da kleinere Propeller typischerweise mit kleineren Motoren mit höherem KV-Wert kombiniert werden, drehen sie zusätzlich schneller, was die Agilität und das Ansprechverhalten weiter steigert.

Größere Propeller erzeugen mehr Schub und arbeiten effizienter. Sie sorgen zudem für ein ruhigeres Flugverhalten, allerdings auf Kosten der Agilität. Mit zunehmender Propellergröße steigt auch die Belastung für die Motoren und die Motorsteuerung (ESC), da die größere Blattfläche einen höheren Luftwiderstand erzeugt. Außerdem haben größere Propeller ein höheres Eigengewicht.

1.4.2

Number of Blades

Propellers can have between two and eight blades. The fewer blades a propeller has, the higher the possible speed and efficiency (thrust per watt or per ampere). With an increasing number of blades, control, smoothness, and acceleration improve – but power consumption also increases.

For long-range applications where efficiency is the top priority, two-blade propellers are typically used. Racing and freestyle drones often use three-blade propellers. CineWhoops commonly use propellers with five or more blades, as these provide maximum flight stability and precise control.

Anzahl der Blätter

Es gibt Propeller mit zwei bis acht Blättern. Je weniger Blätter ein Propeller hat, desto höher sind die erreichbare Geschwindigkeit und Effizienz (Schub pro Watt oder pro Ampere). Mit steigender Anzahl an Blättern verbessern sich Kontrolle, Laufruhe und Beschleunigung, gleichzeitig steigt jedoch auch der Stromverbrauch.

Für Long-Range-Anwendungen, bei denen Effizienz oberste Priorität hat, werden in der Regel Zweiblattpropeller verwendet. Racing- und Freestyle-Drohnen sind häufig mit Dreiblattpropellern ausgestattet. CineWhoops nutzen oft Propeller mit fünf oder mehr Blättern, da sie ein besonders stabiles Flugverhalten und präzise Steuerung ermöglichen.

1.4.3

Pitch

In addition to the number of blades, the pitch of the propeller also affects speed and efficiency (power consumption). A low pitch provides greater efficiency when hovering due to lower air resistance and results in smoother flight behavior. A high pitch allows for greater speed and more aggressive flight maneuvers but increases power consumption.

The pitch is specified in inches and describes the theoretical distance the propeller would move forward during one full rotation if there were no slippage.

For example, with a pitch of 4.3 inches per rotation and a motor speed of 3000 RPM, the theoretical distance traveled per minute is $3000 \times 4.3 = 12,900$ inches per minute. Since one inch equals 25.4 mm, the theoretical speed is $12,900 \times 0.0254 \approx 327.7$ m/min, or approximately $327.7/60 \approx 5.46$ m/s.

In practice, these theoretical values are reduced by about 20–30% due to slippage. This occurs because air is not a solid medium but a fluid one. As air is displaced and swirled, part of it *slips away*, meaning the propeller generates less thrust in reality than predicted theoretically.

Steigung (Pitch)

Neben der Anzahl der Blätter beeinflusst auch die Steigung (Pitch) der Propeller die Geschwindigkeit und Effizienz (Stromverbrauch). Ein niedriger Pitch ist aufgrund des geringeren Luftwiderstands beim Schweben effizienter und führt zu einem ruhigeren Flugverhalten. Ein hoher Pitch ermöglicht höhere Geschwindigkeiten und aggressivere Flugmanöver, steigert jedoch den Stromverbrauch.

Der Pitch wird in Zoll angegeben und beschreibt den theoretischen Vortrieb pro Umdrehung, also wie weit sich der Propeller bei einer vollständigen Umdrehung durch die Luft schrauben würde, wenn es keine Schlupfverluste gäbe.

Bei einem Pitch von 4,3 Zoll pro Umdrehung und einer Motordrehzahl von 3000 U/min ergibt sich ein theoretischer Weg von $3000 \times 4,3 = 12,900$ Zoll/min. Ein Zoll entspricht 25,4 mm. Daraus ergibt sich eine theoretische Geschwindigkeit von $12,900 \times 0,0254 \approx 327,7$ m/min bzw. $327,7/60 \approx 5,46$ m/s.

In der Praxis werden diese theoretischen Werte durch Schlupfverluste um etwa 20–30% verringert. Der Grund dafür ist, dass Luft kein festes, sondern ein bewegliches Medium ist. Durch das Ausweichen und Verwirbeln rutscht immer ein Teil der Luft weg, wodurch der Propeller in der Realität weniger Vortrieb erzeugt als in der Theorie.

1.5

Batteries

FPV drones traditionally use lithium-polymer (LiPo) or lithium-ion (Li-Ion) batteries. Batteries are also classified by their capacity (mAh), voltage (number of cells), discharge rate (C rating, or capacity rate), and connector type. Table 1.4 provides an overview of the characteristic properties of LiPo and Li-Ion batteries.

Akkus

Klassischerweise verwenden FPV-Drohnen Lithium-Polymer- (LiPo) oder Lithium-Ionen-Akkus (Li-Ion). Akkus werden zudem nach ihrer Kapazität (mAh), Spannung (Zellenzahl), Entladerate bzw. C-Wert (*Capacity Rate*) und dem Steckertyp unterschieden. Eine Übersicht der charakteristischen Eigenschaften von LiPo- und Li-Ion-Akkus zeigt Tabelle 1.4.

Table 1.4: Comparison of Lithium-Polymer (LiPo) and Lithium-Ion (Li-Ion) Characteristics

Feature	Li-Ion	LiPo
Energy Density (Wh/kg)	Higher	Lower
Max. Discharge Current (C-Rate)	Lower (10-12 C)	Higher (75-120 C)
Weight (per stored energy)	Lighter	Heavier
Lifespan (number of charge cycles)	Longer	Shorter
Safety	More stable, more robust	More sensitive to over-/undercharge
Use Cases	Long-range, smooth flying, cruising	Racing, freestyle, aggressive maneuvers

1.5.1

Charge and Discharge Rate (C-Rate)

Batteries consist of one or more cells connected in series. FPV drones typically use LiPo batteries because they offer very high C-rates, ranging from 75 to 120 C. The C-rate indicates how quickly a battery can be discharged. In contrast, Li-Ion batteries usually have C-rates of only 10–12 C.

For example, a 1500 mAh LiPo battery (see Figure 1.8) with a C-rate of 100 can deliver $1.5 \text{ Ah} \times 100 = 150 \text{ A}$ of continuous current.



Lade- oder Entladerate (C-Wert)

Akkus bestehen aus einer oder mehreren in Reihe geschalteten Zellen. FPV-Drohnen verwenden meist LiPo-Akkus, da diese sehr hohe C-Werte zwischen 75 und 120 C erreichen. Der C-Wert gibt an, wie schnell ein Akku entladen werden kann. Li-Ion-Akkus bieten im Gegensatz dazu nur 10–12 C.

Ein 1500 mAh LiPo-Akku (siehe Abbildung 1.8) mit einem C-Wert von 100 kann somit $1.5 \text{ Ah} \times 100 = 150 \text{ A}$ Dauerstrom liefern.

**Figure 1.8:** 4S Lithium-Polymer Battery with 1500 mAh Capacity and 100 C Capacity Rate

Such high current levels are only needed for very dynamic flight maneuvers. For typical use in education and research projects, this is not relevant. Patrol flights or hovering for data collection do not require high C-rates.

LiPo batteries have several disadvantages. They can be easily over-discharged, which permanently damages or destroys them. Crashes often lead to physical damage, and improper handling — such as charging too quickly or using a damaged battery — carries a latent risk of fire. A much more robust alternative to LiPo batteries are lithium-ion batteries, although they provide much lower C-rates.

Solch hohe Ströme werden nur für sehr dynamische Flugmanöver benötigt. Für typische Anwendungen in Lehre und Forschungsprojekten ist das jedoch uninteressant. Patrouillenflüge oder das Verharren an einer Position zur Datensammlung erfordern keine hohen C-Werte.

LiPo-Akkus haben mehrere Nachteile. Sie können leicht tiefentladen werden, was sie dauerhaft beschädigt oder zerstört. Abstürze führen häufig zu Beschädigungen, und bei unsachgemäßer Handhabung – etwa durch zu schnelles Laden oder den Betrieb trotz beschädigter Außenhülle – besteht ein latentes Brandrisiko. Eine deutlich robustere Alternative zu LiPo-Akkus sind

For example, a 2500 mAh Li-Ion battery (see Figure 1.9) with a C-rate of 12 can deliver $2.5 \text{ Ah} \times 12 = 30 \text{ A}$ of continuous current.



Figure 1.9: 4S Lithium-Ion Battery with 2500 mAh Capacity and 12 C Capacity Rate

Li-Ion batteries are ideal for scenarios where long flight times, smooth flight behavior, and extended range are desired. LiPo batteries, on the other hand, are better suited for fast flight maneuvers and racing applications.

The C-rate also indicates the recommended charging rate. Unless the manufacturer explicitly specifies a higher value, the standard charging rate is 1 C. For the 1500 mAh LiPo battery mentioned earlier, a 1 C charging rate corresponds to a current of 1.5 A, meaning that a full charge theoretically takes one hour. However, this is a hypothetical value, as a deeply discharged battery can no longer be recharged.

Lithium-Ionen-Akkus, die allerdings wesentlich geringere C-Werte bieten.

Ein 2500 mAh Li-Ion-Akku (siehe Abbildung 1.9) mit einem C-Wert von 12 kann somit $2.5 \text{ Ah} \times 12 = 30 \text{ A}$ Dauerstrom liefern.

Li-Ion-Akkus sind ideal für Szenarien, in denen lange Flugzeiten, ruhiges Flugverhalten und gegebenenfalls große Reichweiten (Long-Range) angestrebt werden. LiPo-Akkus hingegen eignen sich besser für schnelle Flugmanöver und Rennanwendungen.

Der C-Wert gibt auch die empfohlene Laderate an. Diese beträgt 1 C, sofern der Hersteller nicht ausdrücklich eine höhere Laderate, wie etwa 2 C, freigegeben hat. Beim eingangs erwähnten 1500 mAh LiPo-Akku entspricht eine Laderate von 1 C einem Strom von 1,5 A, wodurch das vollständige Laden theoretisch eine Stunde dauern würde. Dies ist jedoch ein hypothetischer Wert, da ein derart tiefentladener Akku nicht mehr geladen werden kann.

1.5.2

Number of Cells (S4/S6)

LiPo and Li-Ion batteries contain one or more cells connected in series. The so-called S rating of a battery indicates the number of cells it contains. Each LiPo cell has a nominal voltage of approximately 3.7 V (fully charged around 4.2 V, discharged around 3.0 V). When multiple cells are connected in series, their voltages add up.

The most common battery types are 4S and 6S. A 4S battery has four cells connected in series, resulting in a nominal voltage of $4 \times 3.7 \text{ V} = 14.8 \text{ V}$. When fully charged, its voltage

Anzahl der Zellen (S4/S6)

LiPo-Akkus und Li-Ion-Akkus enthalten eine oder mehrere in Reihe geschaltete Zellen. Der sogenannte S-Wert eines Akkus gibt die Anzahl der enthaltenen Zellen an. Jede LiPo-Zelle hat eine Nennspannung von etwa 3,7 V (voll geladen etwa 4,2 V, entladen etwa 3,0 V). Werden mehrere Zellen in Reihe geschaltet, addieren sich die Spannungen.

Die gängigsten Akkutypen sind 4S und 6S. Bei einem 4S-Akku sind vier Zellen in Reihe geschaltet, wodurch sich eine Nennspannung von $4 \times 3,7 \text{ V} = 14,8 \text{ V}$ ergibt. Ist ein solcher Akku

is 16.8 V. Table 1.5 provides an overview of different LiPo battery configurations.

voll geladen, beträgt seine Spannung 16,8 V. Eine Übersicht über verschiedene LiPo-Akkus zeigt Tabelle 1.5.

Table 1.5: Overview of Lithium-Polymer (LiPo) and Lithium-Ion (Li-Ion) Batteries

Battery Type	Cells	Nominal Voltage	Fully Charged	Discharged
S1	1	3.7 V	4,2,V	3.0 V
S2	2	7.4 V	8.4,V	6.0 V
S3	3	11.1 V	12.6,V	9.0 V
S4	4	14.8 V	16.8,V	12.0 V
S5	5	18.5 V	21.0,V	15.0 V
S6	6	22.2 V	25.2,V	18.0 V

At higher voltages, a motor requires less current for the same power output and operates more efficiently. 6S components are more powerful but also heavier and more expensive. Larger or performance-oriented drones usually use 6S batteries, while smaller and medium-sized drones typically use 4S batteries and correspondingly lightweight components.

As battery capacity increases, not only does flight time improve, but total weight also increases (see Table 1.6). Very small 4S batteries with a capacity of 450 mAh weigh around 60 g, while the battery shown in Figure 1.8 with a capacity of 1500 mAh weighs about 185 g. Higher total weight also affects flight characteristics.

Bei höherer Spannung benötigt ein Motor bei gleicher Leistung weniger Strom und arbeitet effizienter. 6S-Komponenten sind leistungsstärker, jedoch auch schwerer und teurer. Größere oder leistungsorientierte Drohnen verwenden meist 6S-Akkus, während kleinere und mittlere Drohnen in der Regel 4S-Akkus und entsprechend leichte Komponenten nutzen.

Mit steigender Kapazität eines Akkus erhöht sich nicht nur die Flugdauer, sondern auch das Gesamtgewicht (siehe Tabelle 1.6). Sehr kleine 4S-Akkus mit 450 mAh Kapazität wiegen etwa 60 g, während der in Abbildung 1.8 gezeigte Akku mit 1500 mAh rund 185 g wiegt. Ein höheres Gesamtgewicht beeinflusst auch das Flugverhalten.

Table 1.6: Weight of some Lithium-Polymer (LiPo) and Lithium-Ion (Li-Ion) Batteries

Technology	Type	Vendor	Capacity	C-rate	Voltage	Connector	weight
LiPo	4S	Tattu	450mAh	75C	14.8V	XT30	~ 55 g
LiPo	4S	Gaoneng	650mAh	120C	15.2V	XT30	~ 70 g
LiPo	4S	Tattu	650mAh	95C	14.8V	XT30	~ 80 g
LiPo	4S	Tattu	750mAh	95C	14.8V	XT30	~ 85 g
LiPo	4S	Gaoneng	850mAh	120C	15.2V	XT30	~ 90 g
LiPo	4S	Tattu	850mAh	95C	14.8V	XT30	~ 110 g
LiPo	4S	Tattu Funfly	1300mAh	100C	14.8V	XT60	~ 150 g
LiPo	4S	CNHL	1500mAh	100C	14.8V	XT60	~ 185 g
LiPo	4S	CNHL	2000mAh	100C	14.8V	XT60	~ 235 g
LiPo	6S	Tattu R-Line	650mAh	95C	22.2V	XT30	~ 120 g
LiPo	6S	CNHL	1200mAh	120C	22.2V	XT60	~ 210 g
LiPo	6S	Tattu Funfly	1300mAh	100C	22.2V	XT60	~ 220 g
LiPo	6S	CNHL	1300mAh	130C	22.2V	XT60	~ 230 g
LiPo	6S	CNHL	1500mAh	130C	22.2V	XT60	~ 265 g
Li-Ion	4S (4S1P)	Gaoneng	2500mAh	12C	14.8V	XT60	~ 200 g
Li-Ion	4S (4S1P)	GEPRC	3000mAh	15C	14.8V	XT60	~ 200 g
Li-Ion	4S (4S1P)	RACEpow	4000mAh	10C	14.8V	XT60	~ 290 g
Li-Ion	4S (4S2P)	RACEpow	10000mAh	10C	14.8V	XT60	~ 600 g
Li-Ion	6S (6S1P)	RACEpow	4000mAh	10C	22.2V	XT60	~ 430 g
Li-Ion	6S (6S1P)	GEPRC	4000mAh	15C	22.2V	XT60	~ 445 g
Li-Ion	6S (6S2P)	RACEpow	8000mAh	10C	22.2V	XT60	~ 860 g
Li-Ion	6S (6S2P)	GEPRC	8000mAh	15C	22.2V	XT60	~ 890 g

Li-Ion batteries typically consist of 18650 cells connected in series. The capacity of individual cells depends on the manufacturer and cell quality, typically ranging from 2500 mAh to 3500 mAh. To achieve higher capacity, additional cells must be connected in parallel. The number of cells and their configuration are indicated on the Li-Ion battery using the letters S (series) and P (parallel). Common Li-Ion battery configurations on the market include 3S1P, 4S1P, 4S2P, 6S1P, and 6S2P.

Li-Ion-Akkus bestehen üblicherweise aus in Reihe geschalteten 18650-Zellen. Die Kapazität der einzelnen Zellen hängt vom Hersteller und von der Zellqualität ab und liegt typischerweise zwischen 2500 mAh und 3500 mAh. Soll eine höhere Kapazität erreicht werden, müssen zusätzliche Zellen parallel geschaltet werden. Die Anzahl der Zellen und deren Verschaltung sind auf den Li-Ion-Akkus mit den Buchstaben S (Serie) und P (Parallel) angegeben. Häufig verfügbare Li-Ion-Akkus sind beispielsweise 3S1P, 4S1P, 4S2P, 6S1P und 6S2P.

1.5.3

Plugs (XT30/XT60/XT90) for Discharge and Charge

The battery is usually connected to the flight controller via an XT30 or alternatively an XT60 DC connector. Some larger batteries use an XT90 plug. Two wires with a connector are soldered either to the electronic speed controller (in a stack) or directly to the flight controller in an AIO configuration (see Section 1.2).

Stecker (XT30/XT60/XT90) zum Entladen und Laden

Der Anschluss der Akkus an den Flugcontroller erfolgt in der Regel über einen XT30- oder alternativ einen XT60-Gleichstromstecker. Einige größere Akkus verwenden einen XT90-Stecker. Zwei Kabel mit einer Buchse sind an die Motorsteuerung (bei einem Stack) oder am Flug-

For small and lightweight drones up to 4 inches, XT30 connectors are commonly used. 5- and 6-inch drones typically use XT60 connectors, while larger drones often rely on XT90 connectors. The higher the number, the higher the current the connector can safely handle. However, larger connectors are also heavier and bulkier. The wire gauge must also match the connector size.

XT30 connectors and sockets are designed for a maximum continuous current of 30 A and peak currents of 40-60 A. XT60 connectors are rated for a maximum continuous current of 60 A and peak currents of 80-120 A, depending on the manufacturer, the wires used, the quality of the solder joints, and the thermal load. XT90 connectors are designed for high-current applications, supporting continuous currents of up to 90 A and peak currents exceeding 150 A.

In addition to the main DC connector, every battery with two or more cells has a balance connector for safe charging. This connector, which in modern batteries typically follows the JST-XHR standard, allows the charger to measure the voltage of each individual cell and control the charging process accordingly. The number of pins equals the number of cells plus one (for the common ground). Thus, a 4S battery has five wires, and a 6S battery has seven. Using the data from the balance connector, the charger balances the voltage across all cells, preventing overcharging of individual cells and reducing the risk of damage or fire, while also avoiding deep discharge of single cells during operation.

controller eines AIO-Systems angelötet (siehe Abschnitt 1.2).

Bei kleinen und leichten Drohnen bis zu einer Größe von 4 Zoll sind XT30-Buchsen üblich. 5- und 6-Zoll-Drohnen verwenden üblicherweise XT60-Buchsen, während größere Drohnen häufig XT90-Buchsen nutzen. Je größer die Zahl, desto höhere Ströme kann der Stecker sicher übertragen. Größere Buchsen und Stecker sind allerdings auch schwerer und sperriger. Auch der Leitungsquerschnitt muss zur jeweiligen Buchse passen.

XT30-Buchsen und -Stecker sind für einen maximalen Dauerstrom von 30 A und einen Spitzenstrom von 40-60 A ausgelegt. XT60-Stecker sind für einen maximalen Dauerstrom von 60 A und einen Spitzenstrom von 80-120 A ausgelegt, abhängig vom Hersteller, den verwendeten Leitungen, der Qualität der Lötverbindungen und der thermischen Belastung. XT90-Stecker sind für Hochstromanwendungen konzipiert und für einen Dauerstrom von bis zu 90 A sowie für über 150 A Spitzenstrom ausgelegt.

Neben dem Gleichstromstecker verfügt jeder Akku mit zwei oder mehr Zellen über einen Balancerstecker zum sicheren Laden. Dieser Stecker, der bei modernen Akkus meist dem JST-XHR-Standard entspricht, ermöglicht es dem Ladegerät, die Spannung jeder einzelnen Zelle zu messen und den Ladevorgang entsprechend zu steuern. Die Anzahl der Pins entspricht der Anzahl der Zellen plus eins (für die gemeinsame Masse). Ein 4S-Akku hat somit fünf Leitungen, ein 6S-Akku sieben. Mithilfe der über den Balancerstecker ausgelesenen Werte gleicht das Ladegerät die Spannung der einzelnen Zellen an. Dies verhindert die Überladung einzelner Zellen während des Ladevorgangs und damit das Risiko einer Beschädigung des Akkus oder gar eines Brandes und beugt einer Tiefentladung einzelner Zellen im späteren Betrieb vor.

1.6

GPS

A GPS module (Global Positioning System) is essential for many useful functions such as Return-to-Home (automatic return flight), speed measurement, altitude determination, position hold, autonomous flight, and more.

GPS

Ein GPS-Modul (Global Positioning System) ist für viele nützliche Funktionen wie Return-to-Home (automatisches Heimfliegen), Geschwindigkeitsmessung, Höhenbestimmung, Unterstützung der Positionshaltung oder autonomen Flug unverzichtbar.

Not every GPS module is compatible with all common flight controller firmwares. While Betaflight supports almost any GPS module, this is not the case for INAV and ArduPilot.

A GPS module with six connections (pads or wires) typically includes a magnetic compass. If the module has only four connections, it lacks this compass. Without a magnetic compass, a drone cannot determine its heading when stationary—meaning it does not know which direction it is facing. Consequently, several useful functions, such as GPS position hold in ArduPilot, are unavailable.

Common GPS modules (see Figure 1.10) with an integrated magnetic compass are typically between 20×20 mm and 25×25 mm in size.

Nicht jedes GPS-Modul ist mit allen gängigen Firmwares für Flugcontroller kompatibel. Während Betaflight nahezu jedes GPS-Modul unterstützt, ist das bei INAV und ArduPilot nicht immer der Fall.

Ein GPS-Modul mit sechs Anschlüssen (Löt-pads oder Kabeln) verfügt in der Regel über einen integrierten magnetischen Kompass. Hat das Modul hingegen nur vier Anschlüsse, fehlt der Kompass. Ohne magnetischen Kompass kann eine Drohne im Stillstand nicht erkennen, in welche Richtung sie ausgerichtet ist. Dadurch sind mehrere sinnvolle Funktionen, wie etwa GPS-Position-Hold bei ArduPilot, nicht nutzbar.

Gängige GPS-Module (siehe Abbildung 1.10) mit integriertem magnetischem Kompass sind in der Regel zwischen 20×20 mm und 25×25 mm groß.



Figure 1.10: HGLRC M100-5883 M10 GPS Module 21x21 mm with built-in magnetic Compass (Front and Back)

1.7

Receiver

The receiver receives the control signals (various channel values) from the remote controller and passes them on to the flight controller. It demodulates the incoming signals and converts them into digital channel values.

The receiver and transmitter must be compatible, meaning they must use the same protocol

Empfänger

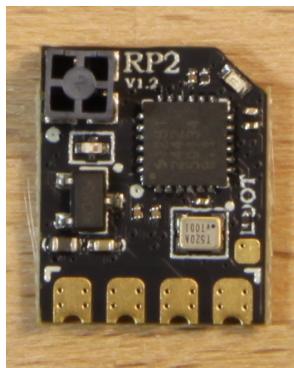
Der Empfänger empfängt die Steuersignale (verschiedene Kanalwerte) der Fernsteuerung und gibt sie an den Flugcontroller weiter. Er demoduliert die empfangenen Signale und wandelt sie in digitale Kanalwerte um.

Empfänger und Fernsteuerung müssen zueinander kompatibel sein, das heißt, sie müssen das-

and operate in the same frequency band. Common frequency bands are 2.4 GHz and, for long-range applications, 868 MHz in the EU or 915 MHz in the USA. In Germany, the use of these frequency ranges is permitted at a maximum transmission power of 25 mW.

Two popular protocols are the open-source system ExpressLRS (ELRS) and the proprietary TBS Crossfire. Because ELRS is open source, there are many hardware providers, and development progresses rapidly. For this reason, this document focuses entirely on the ELRS standard. ELRS devices (transmitters and receivers) run the ExpressLRS firmware, and ideally, both should use the same firmware version.

There are receivers with an integrated ceramic antenna (see Figure 1.11) mounted directly on the circuit board, as well as receivers with one (see Figure 1.12) or two detachable antennas. The commonly used connector is the delicate U.FL (also called *pigtail*). Receivers with fixed ceramic antennas are more compact, as no antenna cable needs to be routed, but the small antenna height of only a few millimeters significantly reduces range.



selbe Protokoll und denselben Frequenzbereich verwenden. Gängige Systeme nutzen die Frequenzbereiche 2,4 GHz und – für Long-Range-Anwendungen – 868 MHz in der EU bzw. 915 MHz in den USA. In Deutschland ist die Nutzung dieser Frequenzbereiche bei einer maximalen Sendeleistung von 25 mW zulässig.

Zwei populäre Protokolle sind das offene System ExpressLRS (ELRS) und das proprietäre TBS Crossfire. Da ELRS ein Open-Source-Protokoll ist, gibt es zahlreiche Anbieter von Hardwarekomponenten, und die Weiterentwicklung schreitet schnell voran. Aus diesem Grund konzentriert sich dieses Dokument vollständig auf den Standard ELRS. ELRS-Geräte (Sender und Empfänger) verwenden die ExpressLRS-Firmware, und idealerweise sollten beide die gleiche Firmware-Version nutzen.

Es gibt Empfänger mit einer integrierten Keramikantenne (siehe Abbildung 1.11), die direkt auf der Platine montiert ist, sowie Empfänger mit einer (siehe Abbildung 1.12) oder zwei abnehmbaren Antennen. Der üblicherweise verwendete Steckverbinder ist auch hier der filigrane U.FL-Stecker (*Pigtail*). Empfänger mit fester Keramikantenne sind kompakter, da kein Antennenkabel verlegt werden muss; durch die nur wenige Millimeter hohe Antenne reduziert sich jedoch die Reichweite deutlich.

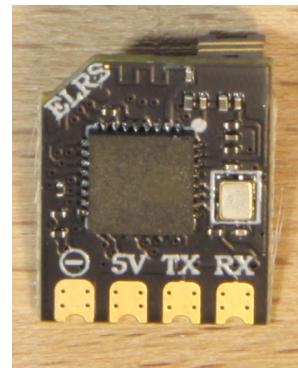


Figure 1.11: RadioMaster RP2 ELRS 2.4GHz Nano Receiver 13x11 mm with integrated Antenna (Front and Back)

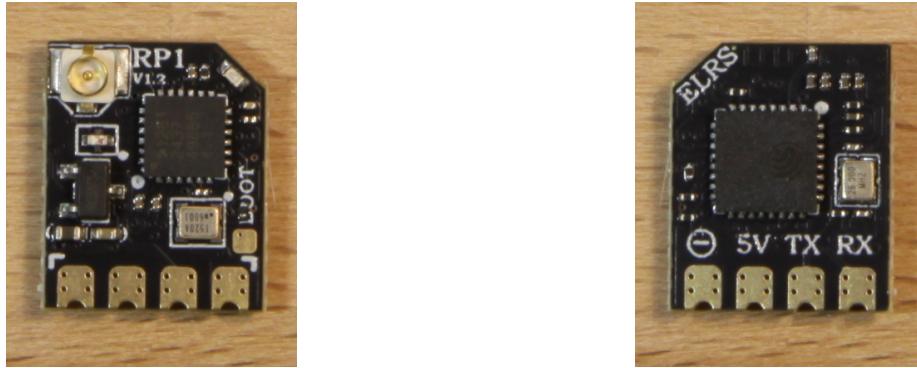


Figure 1.12: RadioMaster RP1 ELRS 2.4GHz Nano Receiver 13x11 mm with U.FL Antenna Connector (Front and Back)

1.8

Remote Control (Sender)

The remote control must implement the same protocol as the receiver in use and operate in the same frequency band. Manufacturers that produce remote controllers compatible with the ELRS protocol include RadioMaster, Axisflying, and Jumper.

ELRS remote controls (transmitters) use ExpressLRS as the firmware for the radio protocol and transmitter module, and EdgeTX as the operating system – a modernized version of OpenTX. EdgeTX generates the control signals, while ExpressLRS transmits them.

Fernbedienung (Sender)

Die verwendete Fernbedienung muss dasselbe Protokoll implementieren wie der eingesetzte Empfänger und denselben Frequenzbereich nutzen. Zu den Herstellern von mit dem ELRS-Protokoll kompatiblen Fernbedienungen gehören RadioMaster, Axisflying und Jumper.

ELRS-Fernbedienungen (Sender) verwenden ExpressLRS als Firmware für das Funkprotokoll und das Sendermodul sowie EdgeTX als Betriebssystem, eine modernisierte Variante von OpenTX. EdgeTX erzeugt die Steuersignale, und ExpressLRS überträgt sie.

1.9

Video Transmitter

The video transmitter (VTX) transmits the live image from the FPV camera to the FPV goggles using the selected channel within the defined frequency band. In addition to analog video transmitters, there are several digital systems that are incompatible with one another. Video transmitters are sold either as standalone devices or as kits that include a camera and antenna(s). Figure 1.13 shows an example of an analog video transmitter.

Videosender

Der Videosender (VTX) überträgt das Livebild der FPV-Kamera über den eingestellten Kanal im festgelegten Frequenzband an die FPV-Brille. Neben analogen Videosendern existieren auch verschiedene digitale Systeme, die untereinander nicht kompatibel sind. Videosender werden entweder als Einzelgerät oder als Kit mit Kamera und Antenne(n) verkauft. Abbildung 1.13 zeigt einen analogen Videosender.

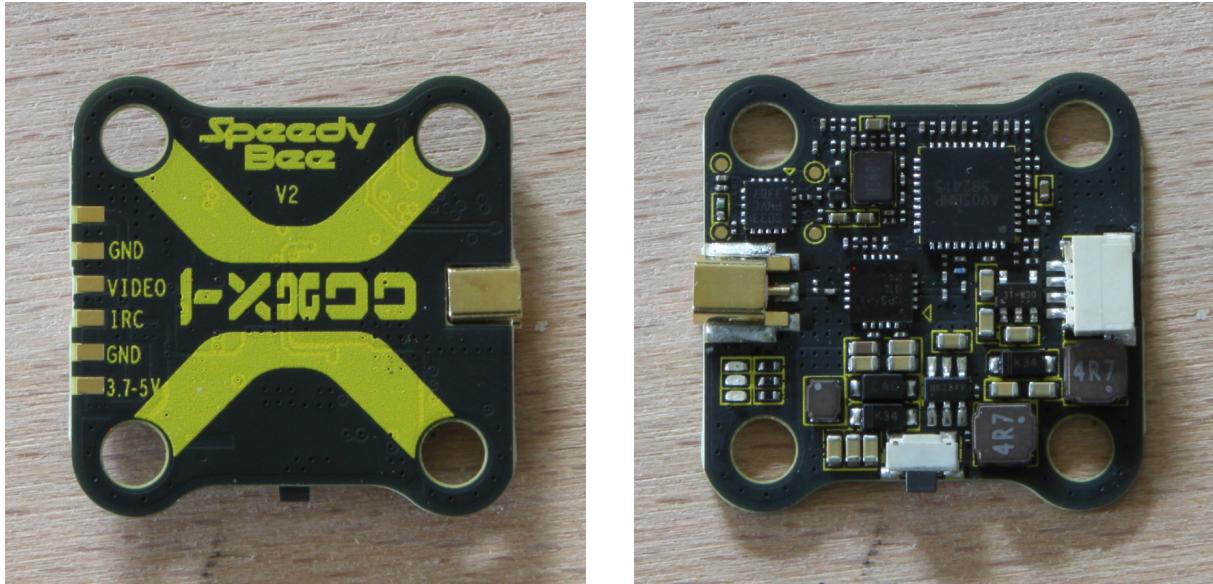


Figure 1.13: SpeedyBee TX800 FPV VTX Video Transmitter (Front and Back)

In Germany, the frequency range from 5725 MHz to 5875 MHz may be used with a maximum transmission power of 25 mW. The channels permitted for analog video transmission within the respective bands are shown in Table 1.7. Notably, the Race Band has wide channel spacing of 37 MHz to prevent channel overlap.

In Deutschland ist die Nutzung des Frequenzbereichs von 5725 MHz bis 5875 MHz bei einer maximalen Sendeleistung von 25 mW zulässig. Die für die analoge Bildübertragung in den jeweiligen Bändern zugelassenen Kanäle zeigt Tabelle 1.7. Auffällig sind beim Race Band die großen Kanalabstände von 37 MHz, die Kanalüberlagerungen verhindern.

Table 1.7: Overview about permitted VTX Channels in Germany

CPU (MCU)	Channel Number (Frequency MHz)
Band A	7 (5745), 6 (5765), 5 (5785), 4 (5805), 3 (5825), 2 (5845), 1 (5865)
Band B	1 (5733), 2 (5752), 3 (5771), 4 (5790), 5 (5809), 6 (5828), 7 (5847), 8 (5866)
Band E	none Frequency permitted in Germany
Band F	1 (5749), 2 (5760), 3 (5780), 4 (5800), 5 (5820), 6 (5840), 7 (5860)
Race Band	3 (5732), 4 (5769), 5 (5806), 6 (5843)

The channel and band used by the video transmitter for image transmission are configured in the graphical interface of the flight controller firmware.

Analog video transmitters are cross-compatible among manufacturers, while this is not the case for digital systems. For digital video transmission, the transmitter and receiver must be from the same manufacturer. Digital systems for live video transmission are available from DJI, HDZero, and CaddxFPV under the brand name Walksnail.

The advantages of digital live video transmission include excellent image quality and the

Der vom Videosender zur Übertragung verwendete Kanal und das Band werden in der grafischen Benutzeroberfläche der Firmware des Flugcontrollers festgelegt.

Analoge Videosender sind herstellerübergreifend mit analogen Videobrillen kompatibel. Bei digitalen Systemen ist das nicht der Fall – hier müssen Sender und Empfänger vom selben Hersteller stammen. Digitale Systeme zur Übertragung des Livebilds werden von DJI, HDZero und CaddxFPV unter dem Namen Walksnail angeboten.

Vorteile der digitalen Livebildübertragung sind die hervorragende Bildqualität und die

ability to record HD footage directly without the need for an additional action camera.

Analog transmission, on the other hand, offers advantages such as cross-manufacturer compatibility, low latency, a gradual rather than abrupt degradation of image quality under interference, and significantly lower hardware costs.

Antennas have a major impact on both image quality and transmission range. Operating a video transmitter without an antenna is not recommended, as it can overheat and become damaged. Analog systems typically use delicate high-frequency connectors following the U.FL standard, commonly known as *pigtail* connectors (also used in Wi-Fi devices). Digital systems, such as those from DJI, generally use MMCX connectors. Antennas with larger and more robust SMA screw connectors can be attached using adapter cables, which are often included with the video transmitter.

The antennas on both the transmitter and receiver sides should match. Available video transmitter antennas differ by their polarization. There are two types:

- linearly polarized antennas (LP)
- circularly polarized antennas (CP)

LP antennas are the simplest and most cost-effective type. They are often included with video transmitters and FPV goggles and typically have a rod-like shape.

CP antennas provide better image quality and greater range. With circular polarization, the signal's polarization rotates as it propagates. There are two variants: right-hand circular polarization (RHCP) and left-hand circular polarization (LHCP).

RHCP antennas are typically used with analog FPV drones, while LHCP antennas are commonly used with digital FPV systems.

Möglichkeit, Videos in HD-Qualität direkt ohne zusätzliche Action-Cam aufzuzeichnen.

Analoge Übertragung bietet hingegen Vorteile wie herstellerübergreifende Kompatibilität, geringe Latenz, ein schrittweises statt abruptes Verschlechtern des Bildes bei Störungen und deutlich geringere Anschaffungskosten der Komponenten.

Die verwendeten Antennen haben großen Einfluss auf Bildqualität und Reichweite. Der Betrieb eines Videosenders ohne Antenne ist nicht empfehlenswert, da dieser überhitzen und beschädigt werden kann. Analoge Systeme verwenden in der Regel sehr filigrane Hochfrequenzsteckverbinder nach dem U.FL-Standard, umgangssprachlich auch *Pigtail* genannt, wie sie auch bei WLAN-Geräten üblich sind. Digitale Systeme, z. B. von DJI, nutzen meist MMCX-Steckverbinder. Antennen mit größerem und robusterem SMA-Schraubgewinde können über Adapterkabel angeschlossen werden, die häufig dem Videosender beiliegen.

Die Antennen auf Sender- und Empfängerseite sollten zueinander passen. Die verfügbaren Antennen für Videosender unterscheiden sich in ihrer Polarisation. Es gibt:

- Antennen mit linearer Polarisation (LP)
- Antennen mit zirkularer Polarisation (CP)

LP-Antennen sind die einfachste und kostengünstigste Variante. Sie werden häufig mit dem Videosender oder der Videobrille mitgeliefert und besitzen meist eine stabförmige Bauweise.

CP-Antennen ermöglichen eine bessere Bildqualität und größere Reichweite. Bei zirkularer Polarisation dreht sich die Polarisation des Signals während der Ausbreitung. Es gibt zwei Varianten: rechtsdrehende (RHCP) und linksdrehende (LHCP) Polarisation.

RHCP-Antennen werden in der Regel bei analogen FPV-Drohnen eingesetzt, während LHCP-Antennen häufig bei FPV-Drohnen mit digitaler Bildübertragung verwendet werden.

1.10

Camera

This section focuses exclusively on analog FPV cameras. Digital systems for live video transmis-

Kamera

Dieser Abschnitt befasst sich ausschließlich mit analogen FPV-Kameras. Digitale Systeme zur

sion have not been used in the projects carried out so far.

The camera generates a video signal (NTSC or PAL), which is passed through the flight controller to the video transmitter. In digital systems, the cameras are directly connected to the video transmitter.

The image quality of analog cameras appears outdated due to noise and blurriness, far from HD standards. The perceived quality ranges from VHS-level at worst to DVD-level at best. However, analog video transmission offers several advantages, such as lower cost, minimal latency, lightweight design, and broad cross-manufacturer compatibility of available components.

Recording high-quality video is not possible with an analog camera. If this requirement exists, an additional action camera or a similar device must be mounted on the drone.

Cameras available on the market differ primarily in resolution, lens focal length (which determines the field of view), and physical size (width).

The camera must not be too wide for the frame used (see Section 1.1). Modern cameras typically have widths of 14 mm (nano cameras) or 19 mm (micro cameras). Figure 1.14 shows a nano camera, and Figure 1.15 shows a micro camera.



Übertragung des Livebilds kamen in den bisher durchgeführten Projekten nicht zum Einsatz.

Die Kamera erzeugt ein Videosignal (NTSC oder PAL), das über den Flugcontroller an den Videosender weitergeleitet wird. Bei digitalen Systemen sind die Kameras direkt mit dem Videosender verbunden.

Die Bildqualität analoger Kameras wirkt durch Bildfehler und Unschärfen nicht mehr zeitgemäß und liegt weit unter HD-Qualität. Der Bildeindruck ist subjektiv im schlechtesten Fall mit VHS-Kassetten und im besten Fall mit DVDs vergleichbar. Die analoge Bildübertragung hat jedoch mehrere Vorteile: geringere Kosten, minimale Latenz, geringes Gewicht und eine herstellerübergreifende Kompatibilität der Komponenten.

Die Aufnahme qualitativ hochwertiger Videos ist mit einer analogen Kamera nicht möglich. Wenn diese Anforderung besteht, muss eine zusätzliche Actioncam oder ein vergleichbares Gerät auf der Drohne mitgeführt werden.

Die am Markt verfügbaren Kameras unterscheiden sich in erster Linie in der Auflösung, der Brennweite der Linse – die das Sichtfeld beeinflusst – und der Bauform (Breite).

Die Kamera darf für den verwendeten Rahmen (siehe Abschnitt 1.1) nicht zu breit sein. Moderne Kameras sind in der Regel 14 mm (Nano-Kamera) oder 19 mm (Micro-Kamera) breit. Abbildung 1.14 zeigt eine Nano-Kamera und Abbildung 1.15 eine Micro-Kamera.

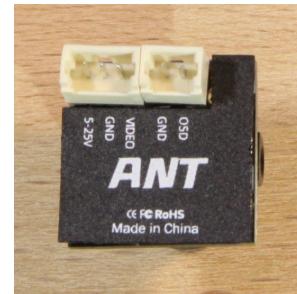


Figure 1.14: Caddx ANT Nano Analog Camera 1200 TVL 14x14 mm (Top and Back)



Figure 1.15: RunCam Phoenix 2 Analog Camera 1000 TVL 19x19 mm (Top and Back)

The horizontal resolution is specified in TV lines (TVL), typically 1000, 1200, or 1500 TVL. The higher this value, the better the potential image quality.

The focal length determines the field of view. The smaller the value, the wider the field of view, which is advantageous for flying. However, a low focal length (e.g., 1.8 mm or 2.1 mm) creates a fisheye effect, resulting in image distortion that can cause eye fatigue and make small objects harder to recognize.

Higher focal lengths (e.g., 2.5 mm or 2.8 mm) provide a more natural image but narrow the field of view, which can make orientation during flight more difficult.

Die horizontale Auflösung wird in TV-Lines (TVL) angegeben und beträgt meist 1000, 1200 oder 1500 TVL. Je höher dieser Wert ist, desto besser ist die mögliche Bildqualität.

Die Brennweite definiert das Sichtfeld. Je kleiner der Wert ist, desto größer ist das Sichtfeld, was beim Fliegen vorteilhaft ist. Eine niedrige Brennweite (z. B. 1.8 mm oder 2.1 mm) führt jedoch zu einem Fischaugen-Effekt und damit zu Verzerrungen, die die Augen schneller ermüden lassen und besonders kleine Objekte schwerer erkennbar machen.

Hohe Brennweiten (z. B. 2.5 mm oder 2.8 mm) bieten ein angenehmeres Bildgefühl, verengen jedoch das Sichtfeld und erschweren dadurch möglicherweise die Orientierung während des Fluges.

1.11

FPV Goggles

The FPV goggles used must be compatible with the video transmitter. Components for digital live video transmission are generally only compatible if they come from the same manufacturer. Analog FPV goggles, on the other hand, are cross-compatible with all analog video transmitters.

FPV goggles available on the market differ primarily in resolution, display technology, and the available interfaces (inputs and outputs). Table 1.8 compares several key characteristics of some analog FPV goggles currently available on the market.

Videobrille (FPV-Brille)

Die verwendete Videobrille muss zum Videosender passen. Komponenten für die digitale Livebildübertragung sind in der Regel nur dann zueinander kompatibel, wenn sie vom gleichen Hersteller stammen. Analoge Videobrillen sind hingegen herstellerübergreifend mit allen analogen Videosendern kompatibel.

Die am Markt verfügbaren Videobrillen unterscheiden sich in erster Linie hinsichtlich der Auflösung, der verwendeten Display-Technologie und der vorhandenen Schnittstellen (Ein- und Ausgänge). Tabelle 1.8 zeigt eine Gegenüberstellung einiger relevanter Merkmale ausgewählter analoger Videobrillen.

Table 1.8: Overview about analog FPV Goggles

Goggle (Product)	Resolution	Display	Video Output
Skyzone Cobra X V4	1280x720 px	LCD	analog A/V
Skyzone SKY04X Pro	1920x1080 px	OLED	analog A/V
Skyzone SKY04O Pro	1280x720 px	OLED	analog A/V
Skyzone SKY02O	640x400 px	LCD	analog A/V
FatShark Dominator	1920x1080 px	OLED	USB-C video output
FatShark Dominator HDO+	1920x1080 px	OLED	analog A/V
FatShark Recon HD	1920x1080 px	LCD	USB-C video output
Fat Shark Echo	800x480 px	LCD	no video output
Eachine EV300O	1024x768 px	OLED	analog A/V
Eachine EV300D	1280x960 px	LCD	analog A/V
HDZero Goggle 2 ^a	1920x1080 px	OLED	analog A/V + HDMI output
HDZero BoxPro ^a	1280x720 px	LCD	analog A/V + HDMI output
Rotorama 800D	800x480 px	LCD	no video output

^a HDZero goggles are designed for the digital HDZero video system, but also include an analog receiver

The ability to output the video signal through an interface and process it in real time on another device is particularly interesting for many potential AI projects. Although numerous FPV goggles feature an HDMI port, it is typically used only for playing back video from external sources. Only few FPV goggles currently support video output via HDMI.

Most FPV goggles include a 3.5 mm AV port (see Figure 1.16) that allows analog video and audio input and output. Using a video capture device, the output signal can be transmitted to a computer for further processing.

Besonders die Möglichkeit, das Videosignal über eine Schnittstelle auszugeben und in einem anderen Gerät in Echtzeit zu verarbeiten, ist für viele denkbare KI-Projekte eine interessante Option. Zahlreiche Videobrillen verfügen zwar über einen HDMI-Port, dieser dient jedoch ausschließlich zum Abspielen von Videos von externen Quellen. Eine Ausgabe des Videosignals über HDMI unterstützen nur wenige gängigen FPV-Brillen.

Die meisten Videobrillen verfügen über eine 3,5 mm AV-Schnittstelle (siehe Abbildung 1.16), über die Video und Audio analog ein- und ausgegeben werden können. Mithilfe eines Videograbbers kann das Ausgabesignal an einen Computer weitergeleitet und dort verarbeitet werden.



Figure 1.16: A/V Port a Skyzone SKY04X Pro FPV Goggle)

If FPV goggles have a USB-C port for video output, it depends on whether the signal is exported as USB Video Class (UVC), like a webcam, or as HDMI. In the case of a UVC device, the video stream can be processed directly. If the signal is HDMI, a suitable video capture device is also required to forward and process the signal on a computer.

Verfügt eine Videobrille über eine USB-C-Schnittstelle zur Ausgabe des Videosignals, hängt es davon ab, ob das Videosignal – wie bei einer Webcam – als USB Video Class (UVC) exportiert wird oder als HDMI-Signal. Bei einem UVC-Gerät kann der Videostream direkt weiterverarbeitet werden. Handelt es sich hingegen um ein HDMI-Signal, ist ein geeigneter Videograbber erforderlich, um das Ausgabesignal an einen Computer weiterzuleiten und zu verarbeiten.

2

Software for Using FPV Software zum Betrieb von FPV-Drohnen

This chapter introduces the essential software components required for operating FPV drones. Similar to Chapter 1, it does not aim to provide a complete overview of all available software. Instead, it presents a selection of popular firmware options for flight controllers, transmitters, and receivers, as well as several useful tools. The chapter provides an overview of their core functionalities and suitable use cases, offering guidance for installation and initial configuration.

Dieses Kapitel stellt die wichtigsten Software-Komponenten zum Betrieb von FPV-Drohnen vor. Genau wie Kapitel 1 hat auch dieses Kapitel nicht den Anspruch, einen vollständigen Überblick über die verfügbare Software zu geben. Vorgestellt werden eine Auswahl populärer Firmwares für Flugcontroller, Sender und Empfänger sowie einige nützliche Werkzeuge. Das Kapitel soll einen Überblick über den grundlegenden Funktionsumfang und sinnvolle Einsatzbereiche geben und bei den ersten Schritten der Installation und Administration unterstützen.

2.1

Flight Controller Firmware

Every flight controller requires an operating system that primarily processes sensor data (including gyroscope, accelerometer, GPS, and magnetic compass) and user control inputs into motor commands. In addition, flight controller operating systems implement various flight modes (e.g., Acro, Angle, Horizon), route video data from the camera to the video transmitter, provide an on-screen display (OSD), generate and store flight logs (Blackbox logs) for post-flight analysis, allow fine-tuning of flight behavior using filters and mixers, and offer multiple interfaces for communication and configuration.

The operating system of the flight controller is implemented as firmware stored in the flash memory. Since all common flight controller firmwares are open-source projects, it occasionally happens that developer groups split off and create forks that, through new features

Flight Controller Firmware

Jeder Flugcontroller benötigt ein Betriebssystem, das in erster Linie die Sensordaten (u. a. Gyroskop, Beschleunigungssensor, GPS, magnetischer Kompass) und die Steuerbefehle des Benutzers in Motorbefehle umsetzt. Darüber hinaus implementieren die Betriebssysteme der Flugcontroller verschiedene Flugmodi (z. B. Acro, Angle, Horizon), leiten Videodaten der Kamera an den Videosender weiter, bieten ein On-Screen-Display (OSD), erzeugen und speichern Log-Daten (Blackbox-Logs) zur nachträglichen Analyse des Flugverhaltens, ermöglichen die Feineinstellung des Flugverhaltens mithilfe von Filtern und Mixern und stellen verschiedene Schnittstellen zur Kommunikation und Konfiguration bereit.

Das Betriebssystem des Flugcontrollers wird als Firmware implementiert, die im Flash-Speicher abgelegt ist. Da alle gängigen Flugcontroller-Firmwares Open-Source-Projekte sind, kommt es immer wieder vor, dass sich Entwicklergruppen abspalten und Forks

and more active development, may eventually surpass the original project. Likewise, projects may become obsolete over time due to developer inactivity or shifting priorities.

This document introduces the flight controller firmwares Betaflight, INAV, ArduPilot, and PX4.

2.1.1

Betaflight

Betaflight is a firmware for flight controllers used in FPV drones. Its focus lies on supporting manual flight modes such as freestyle and racing. The main goals of Betaflight are fast response to user input and extensive configuration options for optimizing flight behavior (tuning).

Unlike alternative firmwares such as INAV or ArduPilot, Betaflight provides almost no support for autonomous flight. The only exceptions are the *Rescue Mode*, which allows the drone to return automatically in case of connection loss using GPS, and an experimental mode for maintaining altitude and position automatically.

Betaflight is compatible with almost all flight controllers and receivers available on the market and provides an intuitive graphical user interface.

Installation and Configuration

The installation and configuration of the firmware are carried out using the locally installed Betaflight Configurator or the web application available at app.betacflight.com. Figure 2.1 shows the option for installing Betaflight via the web application. The model of the flight controller is usually detected automatically. This page can be accessed via the *Update Firmware* button at the top of the website. Existing configuration backups can also be restored to the flight controller on this page. A backup of the current settings is automatically created when a new version is installed.

gründen, die durch neue Funktionen und aktiver Weiterentwicklung mittelfristig das ursprüngliche Projekt verdrängen. Ebenso kann es passieren, dass Projekte aufgrund fehlender Entwickleraktivität oder veränderter Lebensumstände nicht weiterentwickelt werden und mit der Zeit obsolet werden.

Dieses Dokument stellt die Flugcontroller-Firmwares Betaflight, INAV, ArduPilot und PX4 vor.

Betaflight

Betaflight ist eine Firmware für Flugcontroller von FPV-Drohnen. Der Fokus von Betaflight liegt auf der Unterstützung des manuellen Fliegens (Freestyle, Racing usw.). Ziele von Betaflight sind eine schnelle Reaktion auf Benutzereingaben und umfangreiche Einstellungsmöglichkeiten zur Optimierung des Flugverhaltens (Tuning).

Im Gegensatz zu alternativen Firmwares wie INAV oder ArduPilot unterstützt Betaflight kaum autonomes Fliegen. Ausnahmen sind der *Rescue Mode*, der die Drohne bei Verbindungsproblemen mithilfe eines GPS-Moduls automatisch zurückkehren lässt, sowie ein experimenteller Modus zum automatischen Halten von Höhe und Position.

Betaflight ist mit nahezu allen am Markt verfügbaren Flugcontrollern und Empfängern kompatibel und bietet eine intuitive grafische Benutzeroberfläche.

Installation und Konfiguration

Die Installation und Konfiguration der Firmware erfolgt entweder mit der lokal installierten Software Betaflight Configurator oder über die Webanwendung app.betacflight.com. Abbildung 2.1 zeigt die Möglichkeit, Betaflight über die Webanwendung zu installieren. Das Modell des Flugcontrollers wird in der Regel automatisch erkannt. Diese Seite erreicht man über den Button *Update Firmware* am oberen Rand der Webseite. Ein vorhandenes Backup von Einstellungen kann auf dieser Seite ebenfalls in den Flugcontroller eingespielt werden. Ein Backup der aktuellen Einstellungen wird bei der Installation einer neuen Version automatisch angelegt.

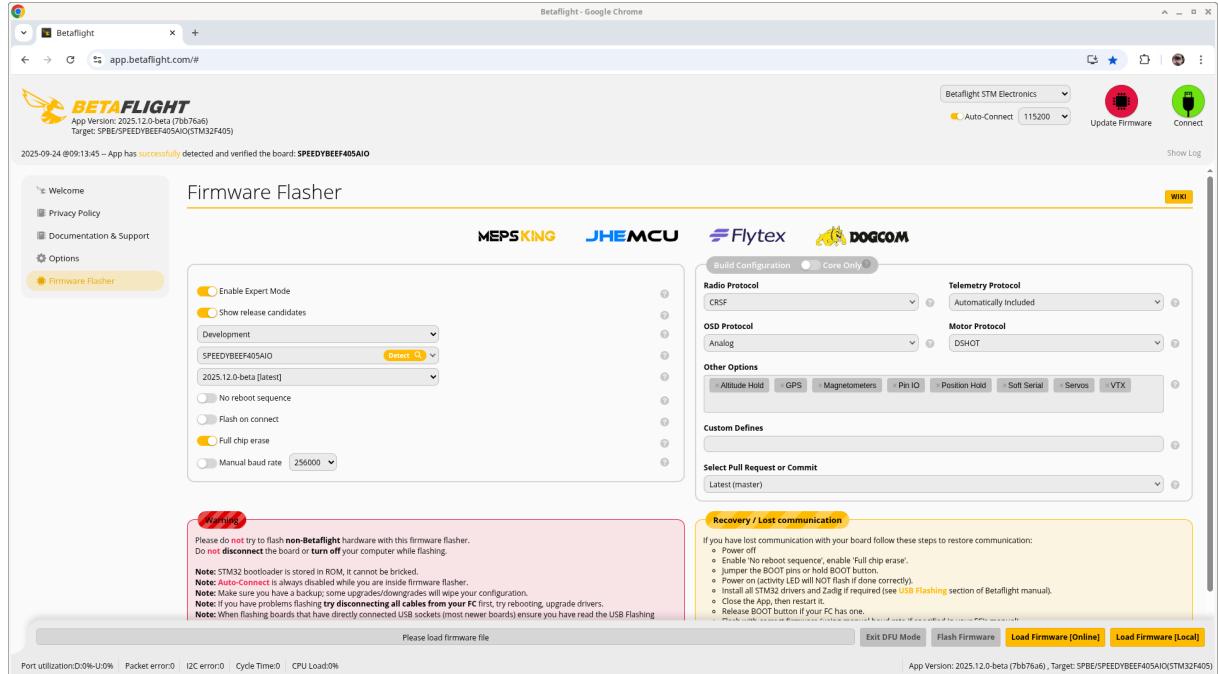


Figure 2.1: Installation of Betaflight and Option to Backup / Restore of Configuration Settings

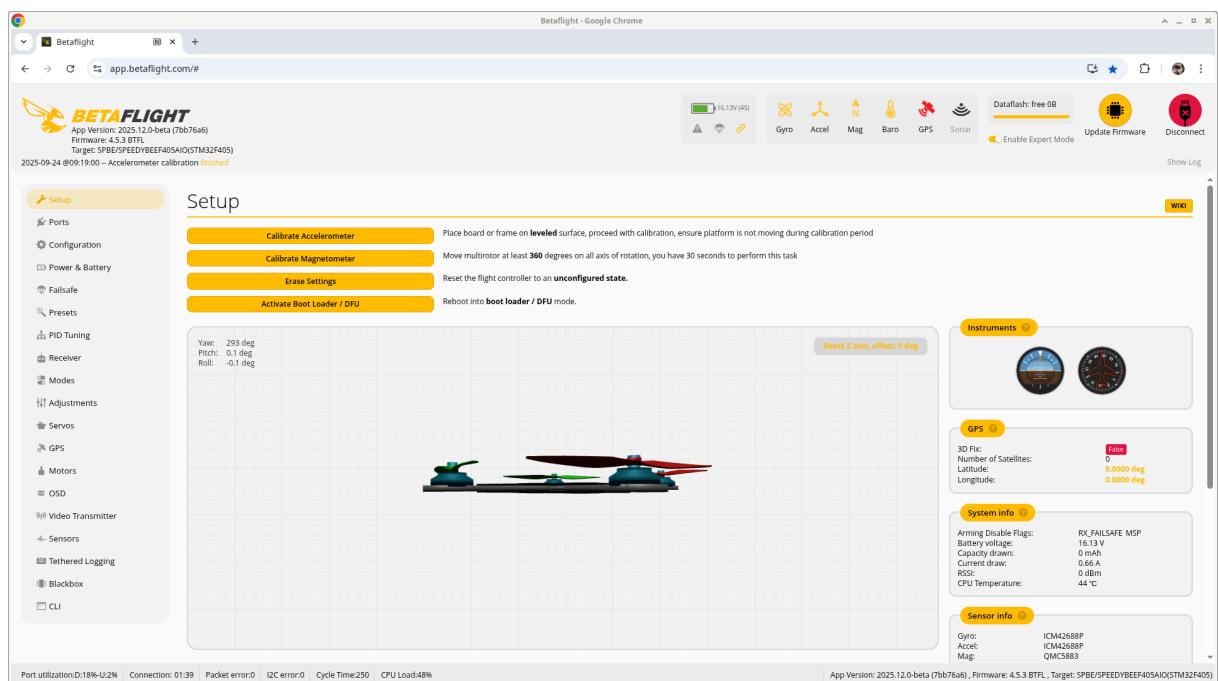


Figure 2.2: Main Page of Betaflight

Figure 2.2 shows the main page (*Setup*) of Betaflight when accessed through the web application. On the main page, the accelerometer and magnetic compass can be calibrated. It also provides general information about the drone's current status and its components. The main page is a valuable tool for verifying the correct orientation configuration of the flight controller within the frame. Especially in flight controller

Abbildung 2.2 zeigt die Startseite (*Setup*) von Betaflight beim Zugriff über die Webanwendung. Auf der Startseite können der Beschleunigungssensor und der magnetische Kompass kalibriert werden. Sie enthält zudem allgemeine Informationen über den Zustand der Drohne und ihrer Komponenten. Ein wertvolles Werkzeug ist die Startseite bei der Kontrolle der korrekten Konfiguration der Ausrichtung des Flugcontrollers

stacks and small drones, it often happens that the flight controller is mounted rotated or upside down. The main page displays the drone's position as a live 3D model corresponding to the orientation configuration set on the *Configuration* page.

Figure 2.3 shows the *Ports* page, where the available UART interfaces are configured. Correctly setting up the UART functions is essential for using many components connected to the drone, such as the video transmitter (in Figure 2.3 on UART3), GPS module (on UART5), and receiver (on UART6).

im Rahmen. Besonders bei Flight-Controller-Stacks und kleineren Drohnen kommt es häufig vor, dass der Flugcontroller gedreht oder auf dem Kopf stehend verbaut ist. Die Startseite zeigt die Position der Drohne als Live-Bild entsprechend der in der Seite *Configuration* eingesetzten Ausrichtung des Flugcontrollers.

Abbildung 2.3 zeigt die Seite *Ports*. Hier wird die Konfiguration der verfügbaren UART-Schnittstellen vorgenommen. Die korrekte Einstellung der Funktionsweise der UARTs ist Voraussetzung für die Nutzung vieler an die Drohne angeschlossener Komponenten, wie beispielsweise des Videosenders (in Abbildung 2.3 an UART3), des GPS-Moduls (an UART5) und des Empfängers (an UART6).

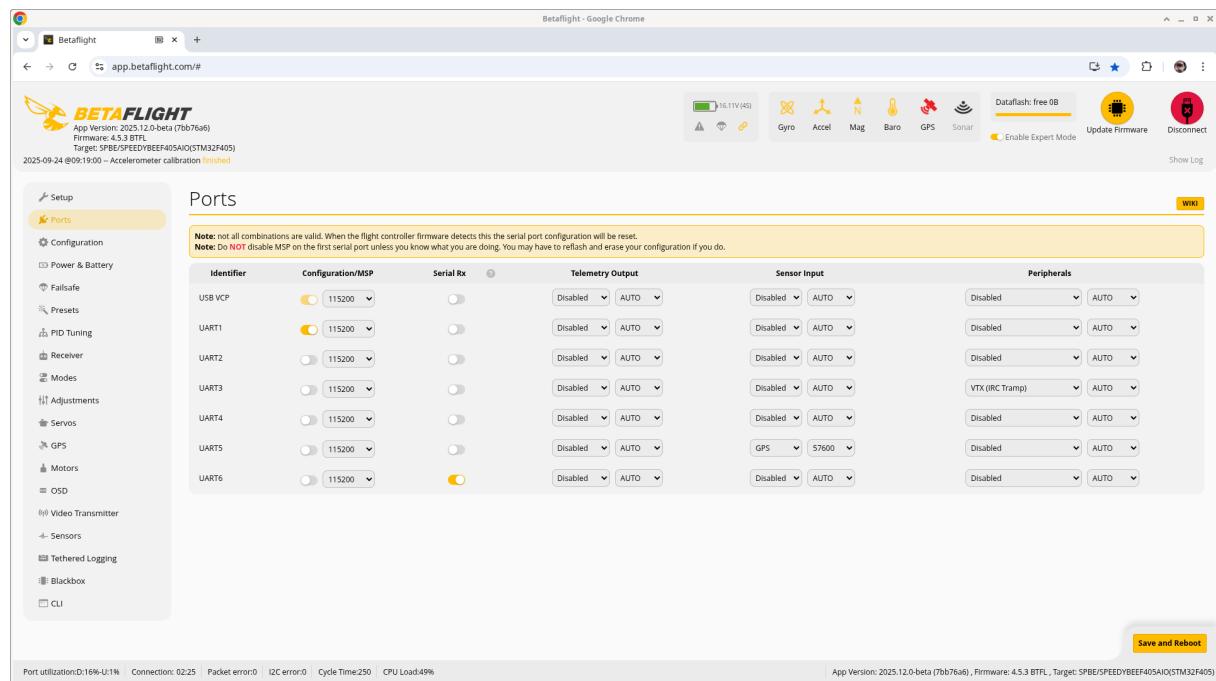


Figure 2.3: Ports Page of Betaflight

On the *Configuration* page (see Figures 2.4 and 2.5), settings are made for, among other things, the orientation of the flight controller and the gyro, which is essential for every drone as it serves as the core instrument for attitude determination.

Auf der Seite *Configuration* (siehe Abbildungen 2.4 und 2.5) werden unter anderem Einstellungen zur Ausrichtung des Flugcontrollers und des für jede Drohne unverzichtbaren Gyroskops (Kreiselinstrument zur Lagebestimmung) vorgenommen.

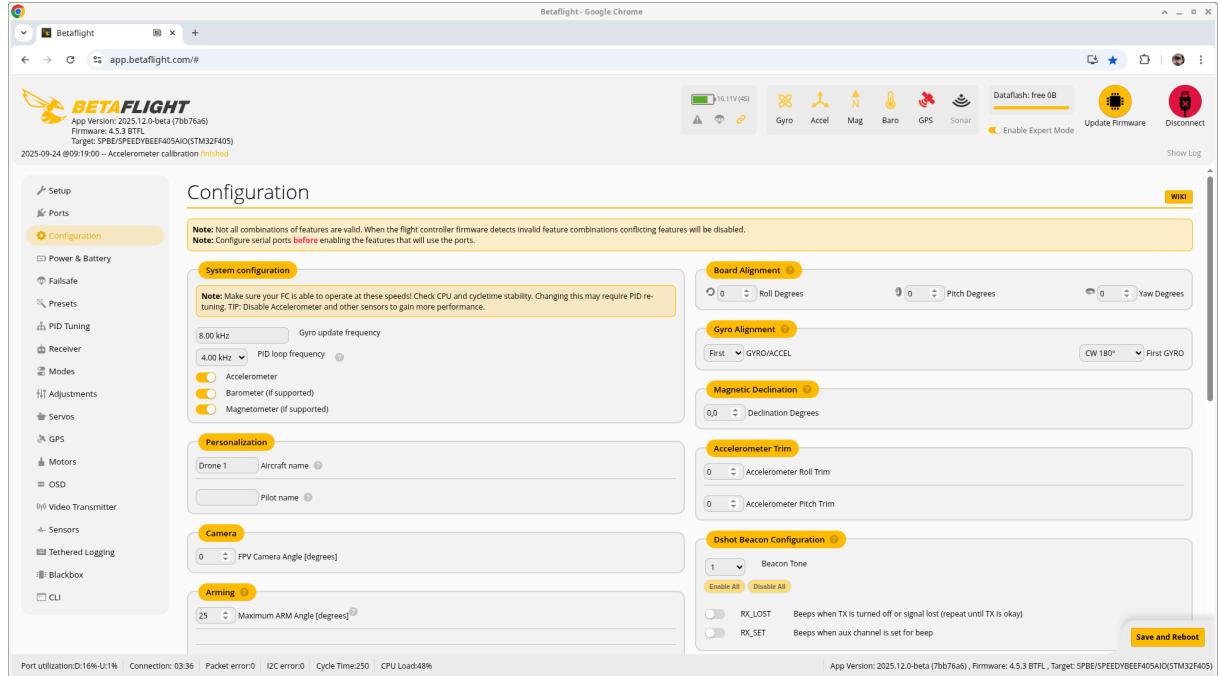


Figure 2.4: Configuration Page of Betaflight (Part 1/2)

Safety-relevant settings, such as the maximum tilt angle of the drone at which arming (activating the motors to prepare for flight) is allowed, are also configured on the *Configuration* page. If an on-screen display (OSD) is to be shown in the FPV live video by the flight controller, it must be enabled here. The *Airmode* option, shown as active in Figure 2.5, ensures that the motors continue to spin slightly even at zero throttle, which improves both the stability and controllability of the drone.

Auch sicherheitsrelevante Einstellungen, wie der maximale Neigungswinkel der Drohne, bei dem das Arming (das Aktivieren der Motoren zur Flugvorbereitung) überhaupt möglich ist, werden auf der Seite *Configuration* festgelegt. Soll im Livebild der FPV-Drohne durch den Flugcontroller ein On-Screen-Display (OSD) eingeblendet werden, muss es hier grundsätzlich aktiviert werden. Die in Abbildung 2.5 aktivierte Einstellung *Airmode* sorgt dafür, dass die Motoren auch bei null Throttle leicht weiterlaufen, was die Stabilität und Kontrollierbarkeit der Drohne verbessert.

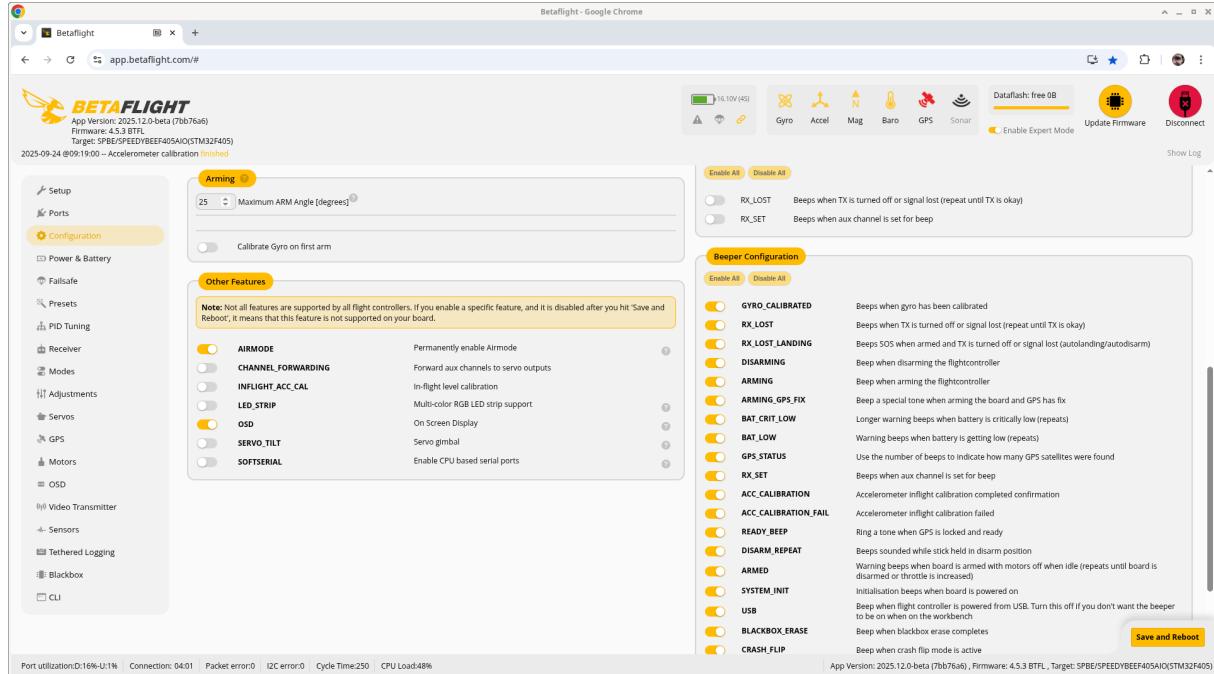


Figure 2.5: Configuration Page of Betaflight (Part 2/2)

The *Power & Battery* page allows you to define the voltage levels at which the battery is considered fully charged or discharged, as well as the thresholds at which the flight controller should issue warnings.

On the *Failsafe* page, you can specify in great detail how the flight controller should respond if the pilot triggers a failsafe condition or if signal loss occurs. Useful settings include having the drone perform a controlled landing or, alternatively, return to its starting point using GPS.

The *Presets* page allows you to save and restore your own Betaflight configuration. It also provides access to a public database of shared configurations, which can be searched and applied based on various parameters.

On the *PID Tuning* (Proportional-Integral-Derivative) page, fine adjustments to the motor control can be made through the flight controller. The goal is typically to achieve more precise flight behavior and to counteract undesirable drone reactions such as oscillations or overly slow or aggressive responses to control inputs.

The correct operation of the receiver can be verified on the *Receiver* page (see Figure 2.6). Movements of the transmitter controls are displayed here when received by the receiver. The

Die Seite *Power & Battery* ermöglicht es, die Spannungswerte zu definieren, bei denen der Akku als vollständig geladen oder entladen gilt, sowie die Schwellen, bei denen der Flugcontroller entsprechende Warnungen ausgeben soll.

Auf der Seite *Failsafe* kann sehr detailliert festgelegt werden, wie der Flugcontroller reagieren soll, wenn der Pilot ein Problem meldet oder ein Signalverlust auftritt. Sinnvolle Einstellungen sind beispielsweise, dass die Drohne kontrolliert landen oder mithilfe des GPS-Moduls selbstständig zum Startpunkt zurückkehren soll.

Die Seite *Presets* ermöglicht es, die eigene Betaflight-Konfiguration zu speichern und wiederherzustellen. Zudem bietet sie Zugriff auf eine öffentliche Datenbank veröffentlichter Konfigurationen, die nach verschiedenen Parametern durchsucht und angewendet werden können.

Auf der Seite *PID Tuning* (Proportional-Integral-Derivative) können Feineinstellungen in der Motorsteuerung durch den Flugcontroller vorgenommen werden. Ziel ist es meist, ein präziseres Flugverhalten zu erreichen und unerwünschte Reaktionen der Drohne – etwa Zittern oder ein zu langsames bzw. zu aggressives Reagieren auf Steuerbefehle – zu vermeiden.

Die korrekte Funktionsweise des Empfängers kann auf der Seite *Receiver* (siehe Abbildung 2.6) überprüft werden. Bewegungen an den Bedienelementen des Senders werden, sobald sie

connection type and communication protocol are also defined on this page. For the receiver to operate correctly, it must be assigned to the correct UART on the *Ports* page.

If telemetry data is to be transmitted from the receiver to the transmitter, the corresponding option must be enabled on the *Receiver* page. Telemetry data typically include information such as battery voltage, current consumption, signal quality, GPS data (latitude, longitude, speed, number of satellites), altitude, and temperature.

vom Empfänger empfangen werden, hier angezeigt. Auch die Anschlussart des Empfängers und das verwendete Protokoll sind auf dieser Seite definiert. Damit der Empfängers korrekt funktioniert, muss er auf der Seite *Ports* dem richtigen UART zugewiesen sein.

Soll der Empfänger Telemetriedaten an den Sender übertragen, muss die entsprechende Option auf der Seite *Receiver* aktiviert werden. Zu den Telemetriedaten gehören in der Regel Informationen wie Akkuspannung, Stromverbrauch, Signalqualität, GPS-Daten (Breitengrad, Längengrad, Geschwindigkeit, Anzahl der Satelliten), Höhe und Temperatur.

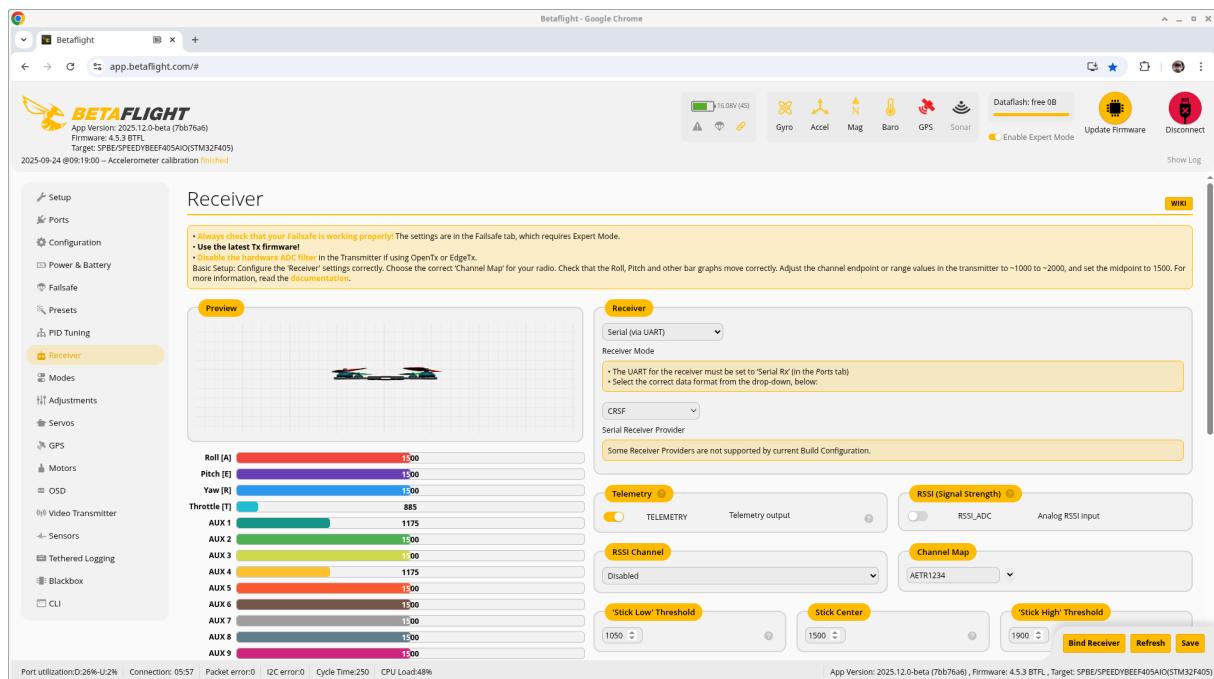


Figure 2.6: Receiver Page of Betaflight

The assignment of important drone functions to individual switches on the transmitter is configured on the *Modes* page (see Figure 2.7). The most important function here is arming the motors to prepare them for flight. Other functions that are sensibly assigned to transmitter switches include flight modes (Angle, Horizon, and Acro), the beeper (buzzer) for locating a crashed drone more easily, and the automatic return-to-home (GPS Rescue) function.

Die Verknüpfung wichtiger Funktionen der Drohne mit einzelnen Schaltern des Senders erfolgt auf der Seite *Modes* (siehe Abbildung 2.7). Die wichtigste Funktion ist hier das Anlaufen („scharf schalten“) der Motoren, um sie für den Flug vorzubereiten. Weitere Funktionen, die sinnvollerweise Schaltern des Senders zugeordnet werden, sind unter anderem die Flugmodi (Angle, Horizon und Acro), der Pieper (Buzzer) zum leichteren Wiederfinden abgestürzter Drohnen sowie die automatische Rückkehr (GPS Rescue) zum Startpunkt.

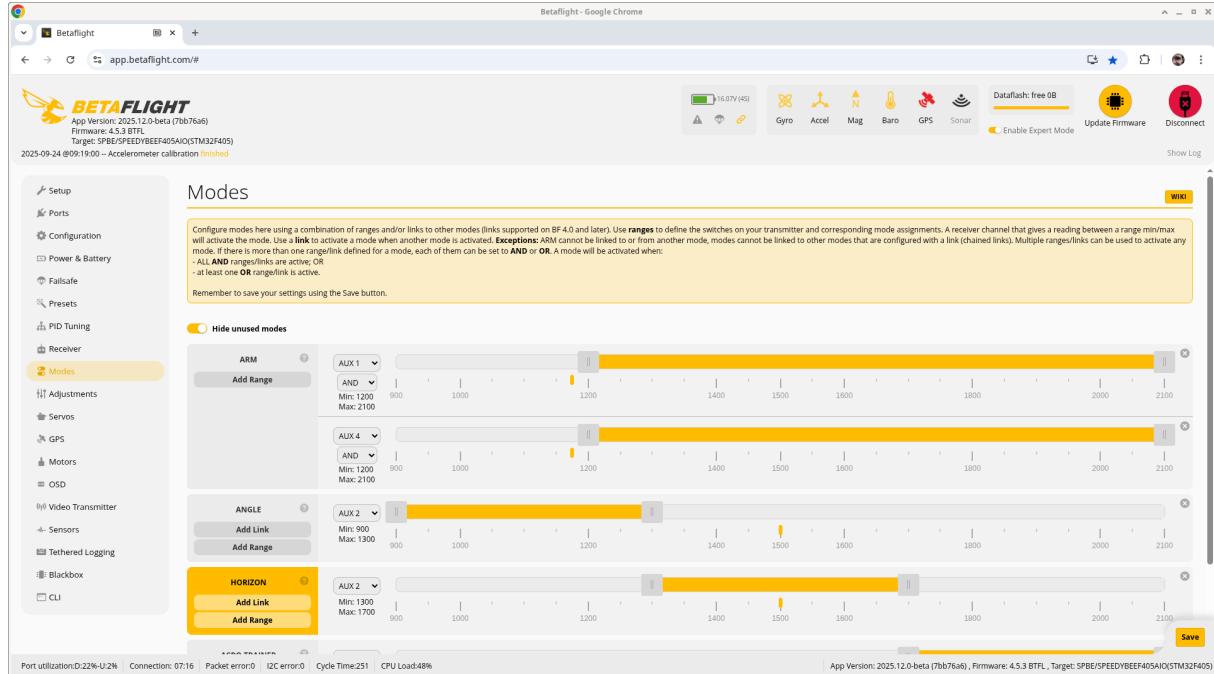


Figure 2.7: Modes Page of Betaflight

The SA switch (AUX1) is very often assigned to the arming function. However, using only one switch increases the risk of accidental arming, which can lead to injuries, such as to hands or fingers. To improve safety when holding the drone, it is therefore recommended to link the arming process to two switches – for example, SA (AUX1) + SB (AUX4) – that are not located next to each other. This exact configuration is shown in Figure 2.7.

Another common configuration is to assign a three-position switch for selecting the flight mode. As shown in Figure 2.7, the SB switch (AUX2) can be configured so that the lower position activates the Angle mode, the middle position activates Horizon mode, and the upper position activates Acro mode.

On the *Adjustments* page, it is possible to define which drone parameters can be adjusted live during flight using specific channels (switches) or potentiometers (dials). This feature allows advanced pilots to fine-tune flight behavior in real time.

The *Servos* page defines control options for servo motors. Here, transmitter channels (switches) are assigned to individual servos, and fine adjustments such as midpoint (neutral posi-

Sehr häufig wird der Schalter SA (AUX1) mit dem Arming belegt. Da bei nur einem Schalter auch ein versehentliches Arming vorkommen kann, treten Verletzungen, z. B. an Händen und Fingern, häufiger auf. Zur Verbesserung der Sicherheit beim Halten der Drohne empfiehlt es sich daher, das Arming mit zwei Schaltern – z. B. SA (AUX1) + SB (AUX4) – zu verknüpfen, die nicht direkt nebeneinander liegen. Genau diese Einstellung ist auch in Abbildung 2.7 zu sehen.

Eine weitere etablierte Konfiguration ist die Belegung eines Schalters mit drei möglichen Positionen, über den der Flugmodus eingestellt wird. So kann beispielsweise, wie in Abbildung 2.7 dargestellt, der Schalter SB (AUX2) so konfiguriert werden, dass bei der unteren Position der Flugmodus Angle, bei der mittleren Position Horizon und bei der oberen Position Acro verwendet wird.

Auf der Seite *Adjustments* ist es möglich zu definieren, welche Parameter der Drohne im Flug mit bestimmten Kanälen (Schaltern) oder Potentiometern (Drehreglern) live angepasst werden können. Diese Seite eröffnet Möglichkeiten zur Feineinstellung für fortgeschrittene Drohnenpiloten während des Fluges.

Die Definition von Steuermöglichkeiten für Servomotoren erfolgt auf der Seite *Servos*. Hier werden Kanäle (Schalter des Senders) einzelnen Servomotoren zugeordnet und nötig

tion) and endpoints (Min/Max) can be set. Typical applications include camera gimbal control or building a gripping mechanism for delivering or collecting small objects.

If a GPS module is connected to the drone's flight controller, configuration and monitoring of the GPS functionality can be performed on the *GPS* page. For the GPS module to operate correctly, it must be assigned to the correct UART on the *Ports* page. When a sufficient number of satellites are visible, this page displays the current coordinates (latitude and longitude), speed, direction of movement, and altitude. The distance from the home position is also shown here. During flight, this information is useful when it is transmitted to the transmitter via telemetry data.

Die Feineinstellungen wie Mittelstellung (Nullposition) und Endpunkte (Min/Max) vorgenommen. Anwendungsszenarien sind zum Beispiel eine Gimbal-Steuerung der Kamera oder der Bau eines Greifmechanismus, um Objekte mit einer Drohne auszuliefern oder aufzusammeln.

Wenn an den Flugcontroller der Drohne ein GPS-Modul angeschlossen ist, können Kontrolle und Konfiguration der GPS-Funktionalität auf der Seite *GPS* erfolgen. Damit das GPS-Modul korrekt funktioniert, muss es auf der Seite *Ports* dem richtigen UART zugewiesen sein. Besteht Sichtkontakt zu ausreichend vielen Satelliten, können hier unter anderem die aktuellen Koordinaten (Breiten- und Längengrad), Geschwindigkeit, Bewegungsrichtung und Höhe kontrolliert werden. Auch die Entfernung zum Startplatz ist hier sichtbar. Während des Fluges sind diese Informationen hilfreich, wenn sie über die Telemetrydaten an den Sender übertragen werden.

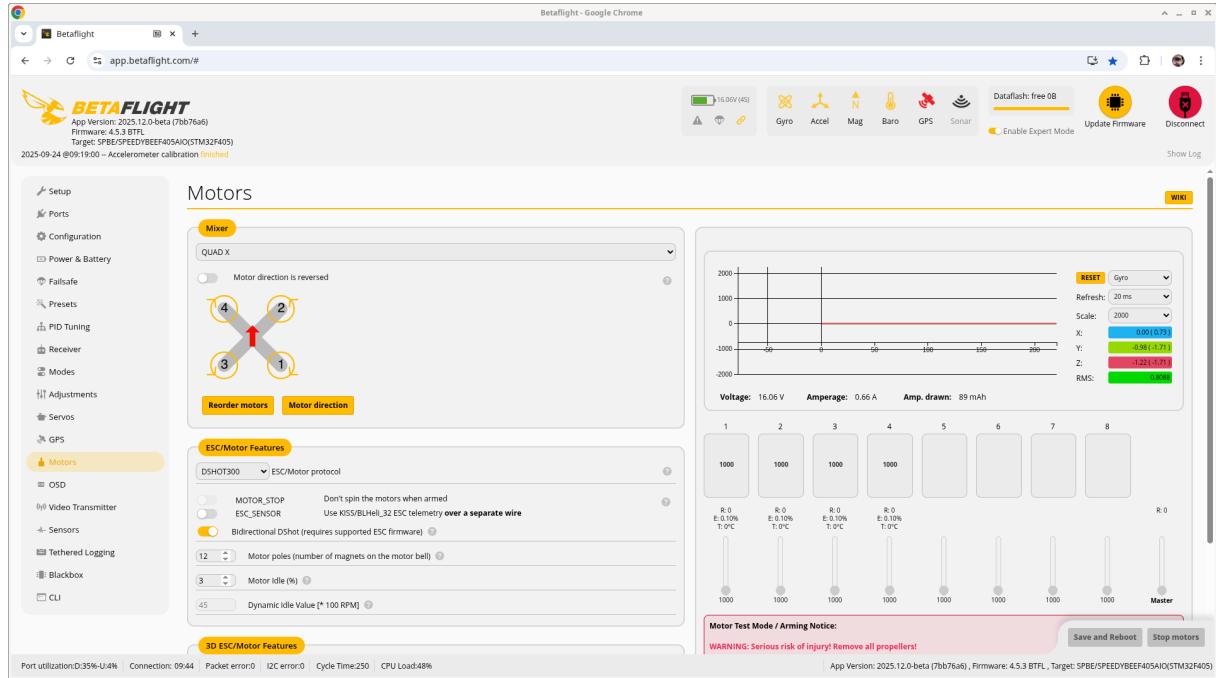


Figure 2.8: Motors Page of Betaflight

All motor-related settings and the testing of motor function, mapping, and rotation direction are performed on the *Motors* page. These tests must always be carried out without propellers, as otherwise damage or even injury can easily occur.

The precise configuration of the on-screen display (OSD) is done on the *OSD* page. Here, the positions and layout of the desired display ele-

Alle Einstellungen bezüglich der Motoren sowie das Testen der korrekten Funktion, Zuordnung und Drehrichtung der einzelnen Motoren erfolgen auf der Seite *Motors*. Diese Tests müssen immer ohne Propeller durchgeführt werden, da es sonst leicht zu Beschädigungen oder gar Verletzungen kommen kann.

Die präzise Konfiguration des On-Screen-Displays erfolgt auf der Seite *OSD*. Hier können die Positionen und das Layout der gewünschten

ments can be freely arranged, and multiple configurations can be saved as profiles—for example, for racing, freestyle, or long-range flying. For OSD configuration to be possible at all, the flight controller must include an OSD chip. If it does not, an analog video transmitter cannot provide an OSD. Digital video transmitters (e.g., from DJI, HDZero, and Walksnail) implement their own OSD systems.

Settings for the video transmitter, such as the selected frequency (see Table 1.7) and transmission power, can be configured on the *Video Transmitter* page (see Figure 2.9). For the video transmitter to function correctly, it must be assigned to the appropriate UART on the *Ports* page.

Anzeigen frei platziert und mehrere Konfigurationen als Profile – zum Beispiel für Racing, Freestyle oder Longrange – gespeichert werden. Damit die Konfiguration des OSD überhaupt möglich ist, muss der Flugcontroller über einen OSD-Chip verfügen. Ist das nicht der Fall, kann mit einem analogen Videosender kein OSD realisiert werden. Digitale Videosender (z. B. von DJI, HDZero und Walksnail) implementieren ein eigenes OSD.

Einstellungen zum Videosender, wie zum Beispiel die verwendete Frequenz (siehe Tabelle 1.7) und die Sendeleistung, können auf der Seite *Video Transmitter* (siehe Abbildung 2.9) vorgenommen werden. Damit der Videosender korrekt funktioniert, muss er auf der Seite *Ports* dem korrekten UART zugewiesen sein.

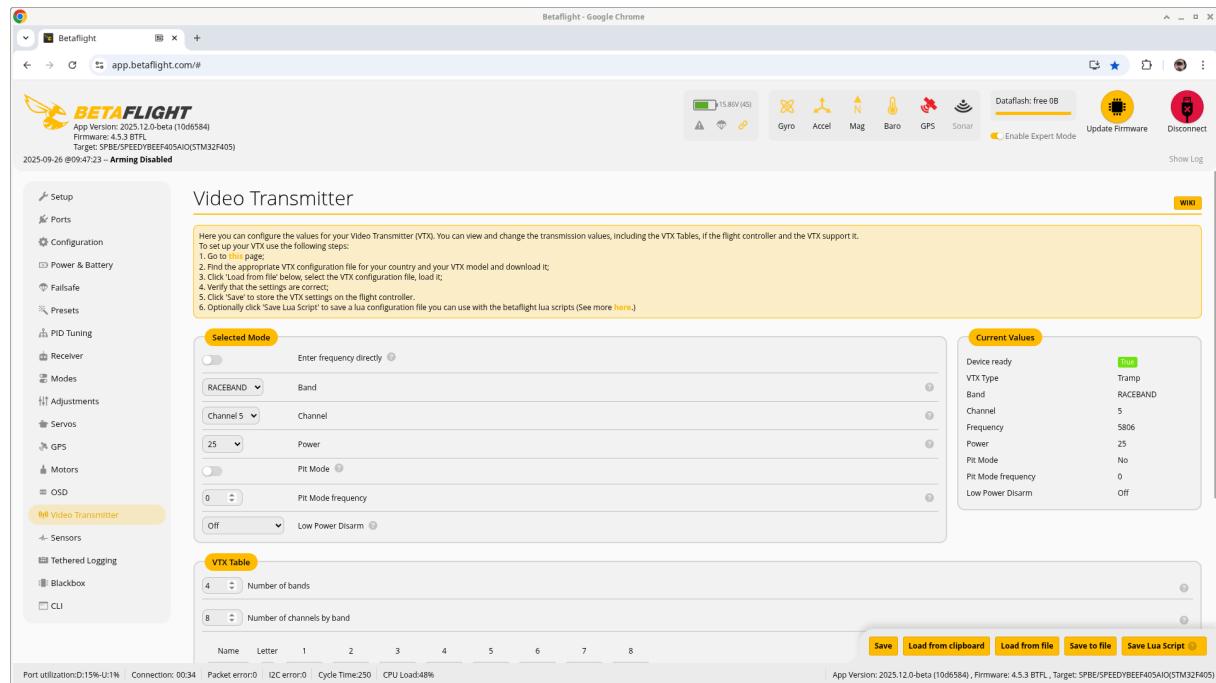


Figure 2.9: Video Transmitter Page of Betaflight

The *Sensors* page (see Figure 2.10) displays the sensor data both graphically and numerically – for example, values from the gyroscope, accelerometer, barometer, and magnetometer. This allows users to verify that all sensors are working correctly and delivering plausible data. The page is particularly useful for troubleshooting sensor orientation or calibration issues.

Die Seite *Sensors* (siehe Abbildung 2.10) zeigt Sensorwerte sowohl grafisch als auch numerisch an – zum Beispiel Werte des Gyroskops, Beschleunigungssensors, Höhenmessers (Barometer) und magnetischen Kompasses. Dadurch lässt sich überprüfen, ob alle Sensoren korrekt funktionieren und plausible Werte liefern. Die Seite ist insbesondere hilfreich zur Fehlersuche bei Problemen mit der Sensororientierung oder Kalibrierung.

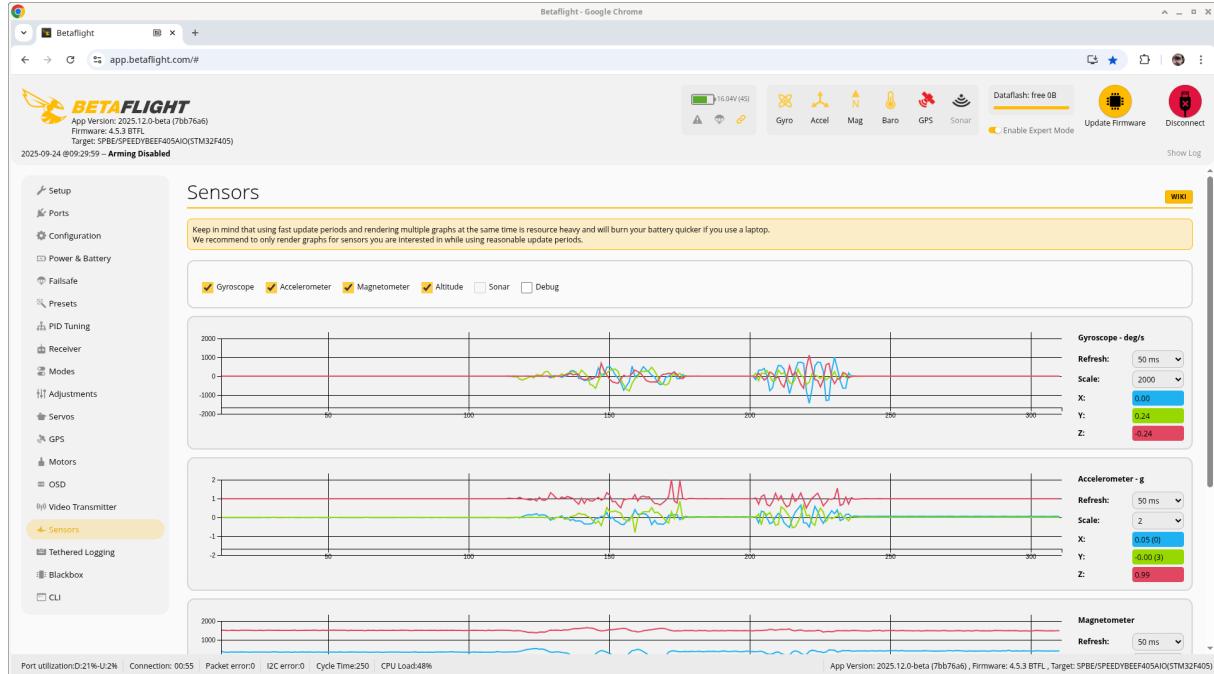


Figure 2.10: Sensors Page of Betaflight

The live recording of sensor data (e.g. gyroscope, motor values, GPS data, altitude) while the flight controller is connected to a computer is available on the *Tethered Logging* page. In contrast to classic Blackbox logging, where data is stored locally on the flight controller, flash memory, or an SD card, Tethered Logging allows real-time data transmission and monitoring on the computer. It only works as long as a connection (e.g. via USB) between the flight controller and the computer is active. If the connection is lost, the live recording stops immediately.

The flight data recorder for logging various sensor data is configured on the *Blackbox* page (see Figure 2.11). The collected data is recorded during flight and analyzed afterwards. The resulting insights can be used to analyze flight behavior, optimize filters, and diagnose issues. Recording typically starts automatically when arming and stops when disarming. Modern flight controllers store Blackbox data on an on-board flash chip or a microSD card. This page allows the selection of which sensors to log, the storage device to use, and the log rate, which defines how frequently data points are captured. The page also offers options to export or erase Blackbox data.

Die Live-Aufzeichnung von Sensordaten (z. B. Gyroskop, Motorwerte, GPS-Daten, Höhenmeter) während der Verbindung mit dem Computer ist auf der Seite *Tethered Logging* möglich. Im Gegensatz zum klassischen Blackbox-Logging, bei dem die Daten lokal im Flugcontroller, auf Flash oder SD-Karte gespeichert werden, erlaubt Tethered Logging die Übertragung und Überwachung der Daten in Echtzeit auf dem Computer. Das Tethered Logging funktioniert nur, solange eine Verbindung (z. B. via USB) zwischen Flugcontroller und Computer besteht. Wenn die Verbindung abreißt, bricht auch die Live-Aufzeichnung ab.

Der Flugdatenschreiber zur Aufzeichnung verschiedenster Sensordaten wird auf der Seite *Blackbox* (siehe Abbildung 2.11) konfiguriert. Die Aufzeichnung der gesammelten Daten geschieht während des Flugs und wird nachträglich ausgewertet. Mit den gewonnenen Erkenntnissen können das Flugverhalten analysiert, Filter optimiert und Probleme diagnostiziert werden. Die Aufzeichnung beginnt üblicherweise automatisch beim Arming und stoppt beim Disarming. Moderne Flugcontroller speichern die Blackbox-Daten auf einem Onboard-Flashchip oder einer microSD-Karte. Die Seite erlaubt neben der Auswahl der zu erfassenden Sensordaten auch die Auswahl des Speichergeräts und die Einstellung der Log-Rate, die definiert, wie häufig Datenpunkte aufgezeichnet werden sollen.

Auch der Export der Blackbox und das Löschen des Speichers sind auf dieser Seite möglich.

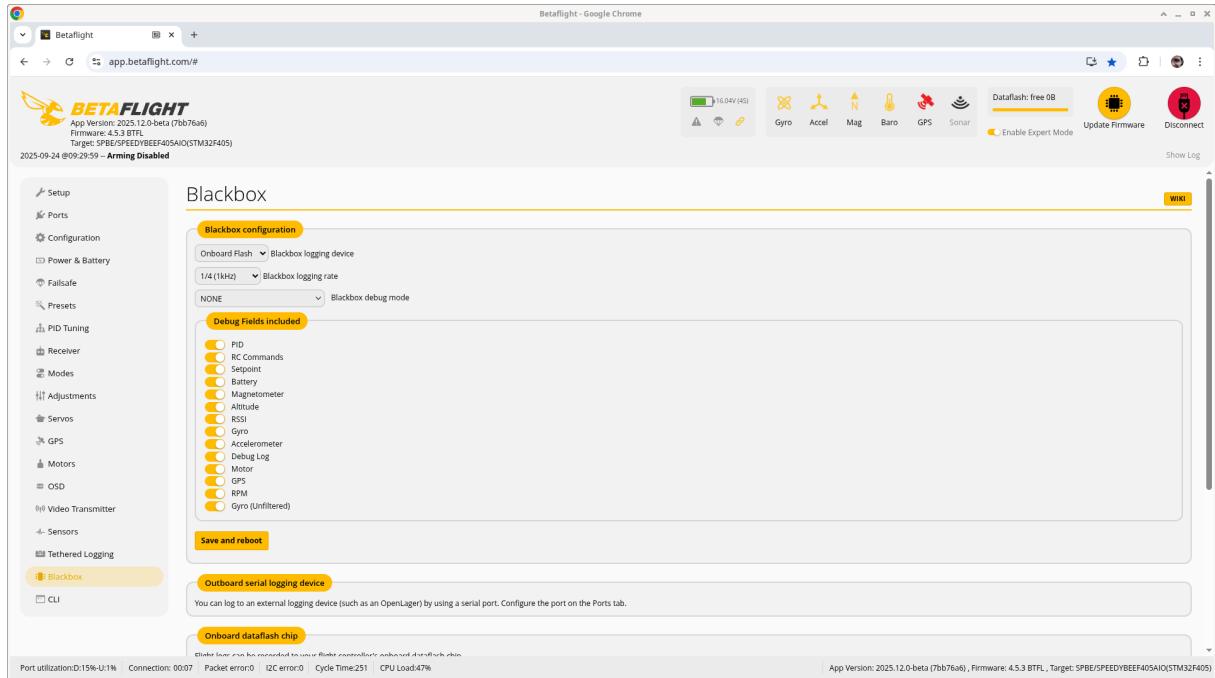


Figure 2.11: Blackbox Page of Betaflight

The *CLI* page in Betaflight (see Figure 2.12) provides access to the integrated command-line interface of the flight controller. This interface allows users to inspect and configure the controller directly. Individual parameters can be displayed and changed using the commands `get` and `set`. The `save` command saves all changes and restarts the flight controller.

Betaflight bietet auf der Seite *CLI* (siehe Abbildung 2.12) Zugriff auf die integrierte Kommandozeilenumgebung des Flugcontrollers. Hierüber kann der Flugcontroller direkt untersucht und konfiguriert werden. Einzelne Parameter können mit den Kommandos `get` und `set` angezeigt und geändert werden. Das Kommando `save` speichert alle Änderungen und startet den Flugcontroller neu.

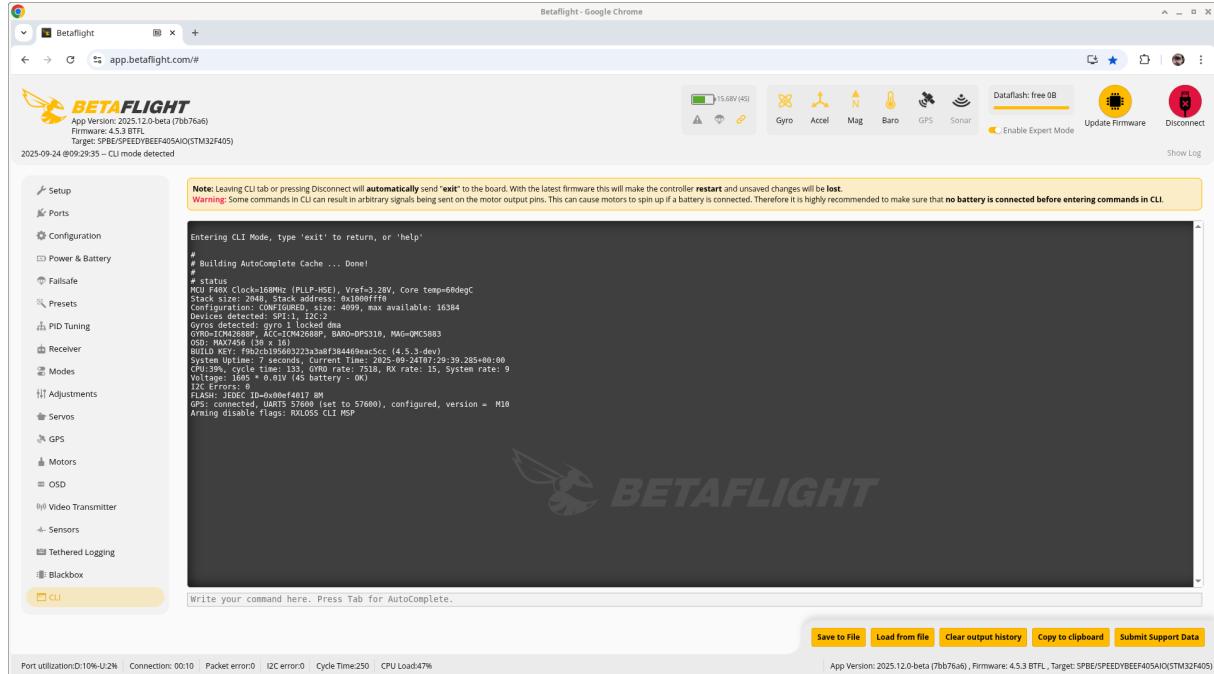


Figure 2.12: CLI Page of Betaflight

The current configuration of the flight controller can be displayed completely with the command **dump**. Only the differences from the default configuration are shown using the command **diff**. These outputs can be copied into text files and re-imported later using copy and paste. This makes it easy to create backups of all settings and restore them when needed.

Information about the flight controller and the installed firmware version can be retrieved with the command **version**. The **status** command provides current sensor readings and information about sensor configuration. The command **tasks** lists all currently running processes on the flight controller, and **resource** displays the pin and UART assignments.

2.1.2

INAV

INAV consists of two main components: the firmware that runs on the flight controller and the configuration tool that is used for installation and administration. There is no web app, as is the case with current Betaflight versions.

In addition to the Return-To-Home (RTH) and rescue functionality, INAV offers additional autopilot features such as waypoint missions,

Die aktuelle Konfiguration des Flugcontrollers kann vollständig mit dem Kommando **dump** ausgegeben werden. Nur die Unterschiede zur Standardkonfiguration zeigt das Kommando **diff**. Diese Ausgaben können in Textdateien gespeichert und bei Bedarf per Copy-and-Paste wieder importiert werden. So lassen sich einfach Backups der Einstellungen erstellen und wiederherstellen.

Informationen zum Flugcontroller und zur installierten Firmwareversion liefert das Kommando **version**. Das Kommando **status** zeigt aktuelle Sensordaten und Informationen zur Konfiguration der Sensoren. Das Kommando **tasks** listet alle aktuell laufenden Prozesse des Flugcontrollers auf, und **resource** zeigt die Belegung der Pins und UART-Schnittstellen.

INAV

INAV besteht aus zwei Hauptkomponenten: Die Firmware, welche auf dem Flugcontroller läuft und dem Konfigurationstool, welches für die Installation und Administration genutzt werden kann. Es gibt keine Web-App, wie bei aktuellen Betaflight-Versionen.

Neben der Return-To-Home- (RTH) bzw. Rescuerfunktionalität bietet INAV weitere Autopilotenfunktionalitäten, wie Wegpunktmissionen,

hold position, hold altitude, and more. These can also be used without a compass (since version 7.1), albeit with lower accuracy.

The software receives a major release every year and minor support updates as needed. The current version 8 supports the following STM flight controller chips: F405, F722, F745, F765, and H743. AT-F435 chips are also supported.

Installation

The installation of INAV is done using the *INAV Configurator*. A certain similarity to Betaflight is very apparent here, as the structure is quite similar to the Betaflight Configurator, which is used for versions up to 4.5 (Figure 2.13). This guide assumes that the flight controller is connected via USB to the computer used for flashing.

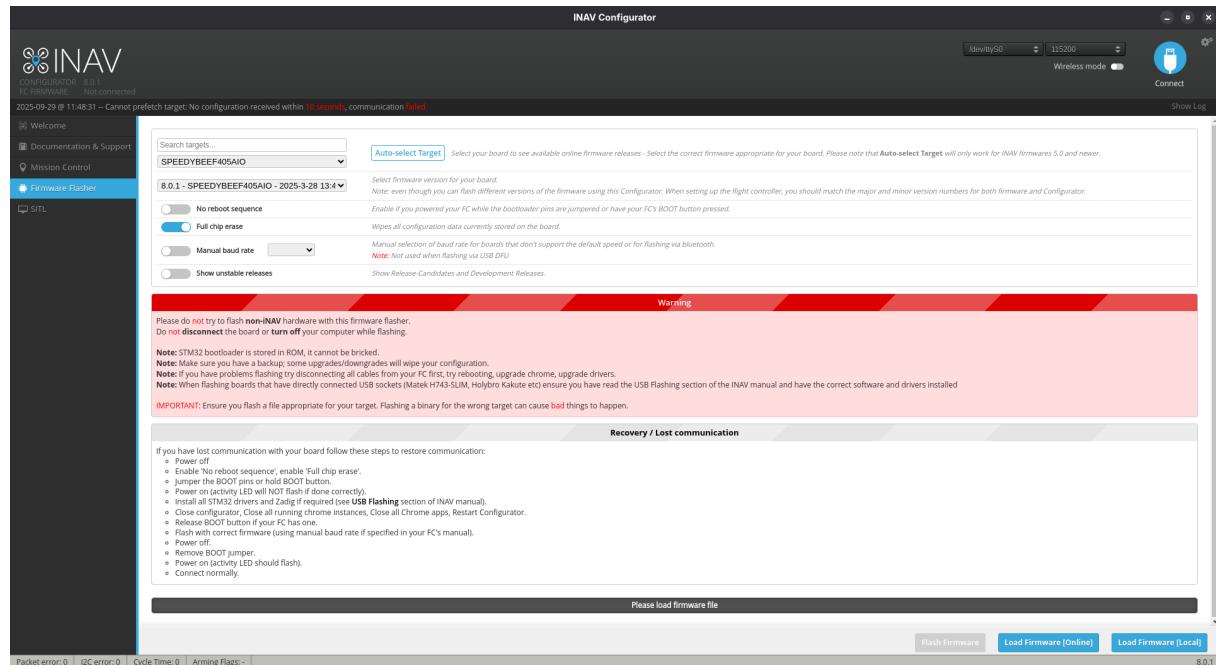


Figure 2.13: Firmware Flasher tab in the INAV Configurator

If the flight controller is not being flashed for the first time, it is good practice to create a backup of the configuration, as some upgrades reset it, or to be safe in case of a failure. This is done via the *CLI* tab. The command `diff all` outputs the corresponding parameters. These can then be saved to a text file using the *Save to File* button. The flashing process itself consists of the following steps:

Position halten, Flughöhe halten und mehr. Diese können auch ohne Kompass genutzt werden (seit Version 7.1), jedoch mit geringerer Genauigkeit.

Die Software erhält jedes Jahr einen großen Release und kleinere Support-Updates, wie sie benötigt werden. Die aktuelle Version 8 unterstützt die folgenden STM-Flugcontrollerchips: F405, F722, F745, F765 und H743. Außerdem werden AT-F435 Chips unterstützt.

Installation

Die Installation erfolgt über den *INAV Configurator*. Der Aufbau dieses Prozesses ist sehr ähnlich zum Betaflight Configurator, welcher bis Version 4.5 verwendet wurde. (Abbildung 2.13). Dieser Guide geht davon aus, dass der Flugcontroller via USB an den Computer angeschlossen ist, mit welchem das Flashen durchgeführt wird.

Wird die Software nicht zum ersten Mal auf dem Flugcontroller installiert, sollte ein Backup der Konfiguration gemacht werden, da manche Upgrades die diese zurücksetzen. Dies geschieht über den *CLI*-Tab. Der Befehl `diff all` gibt die Parameter aus. Mit dem Button *Save to File* können diese in eine Textdatei gespeichert werden. Der Flashprozess besteht dann aus den folgenden Schritten:

1. Putting the flight controller into *Device Firmware Upgrade (DFU) mode* is the first step. This can usually be done using a button on the flight controller board or with the `dfu` command in the *CLI* tab.
2. Once the board is in DFU mode, the *Firmware Flasher* tab in the INAV Configurator must be opened. First, the target must be selected. If *Auto-select Target* does not work, the model name must be searched for manually in the list.
3. The available (stable) releases are then displayed in the drop-down menu below the board. The *Show unstable releases* option allows you to select even more recent versions. However, these may not work as expected.
4. INAV offers three options that can be set for flashing:
 - *No reboot sequence* - Only required if the boot pins are bridged or the boot button remains pressed during the flashing process (depending on the model).
 - *Full chip erase* - Deletes all configuration options (backup available?). This should always be set if a different software was previously installed.
 - *Manual baud rate* - Set a fixed baud rate for Bluetooth or USB if the model in question does not support the standard speed.
5. The flashing process consists of two steps: First, the firmware is loaded (online or file), then the software is flashed onto the board via a separate button. The actual flashing should not be interrupted, as this will brick the device. However, the bootloader required for the flashing itself is stored in ROM and *cannot be bricked*. This means the device can always be re-flashed.
1. Den *Device Firmware Upgrade (DFU)-Modus* aktivieren ist der erste notwendige Schritt. Dies kann in der Regel über einen Button am Flugcontrollerboard realisiert werden oder mit dem Befehl `dfu` im *CLI*-Tab.
2. Befindet sich das Board im DFU-Modus, so muss der *Firmware Flasher* Tab im INAV Configurator aufgerufen werden. In diesem muss zunächst das Target ausgewählt werden, sollte *Auto-select Target* nicht funktionieren, muss der Modellname händisch in der Liste gesucht werden.
3. Im Ausklappmenü unter dem Board werden dann die verfügbaren (stable) Releases angezeigt. Mit der Option *Show unstable releases* können noch aktuellere Versionen gewählt werden. Diese funktionieren aber unter Umständen nicht wie erwartet.
4. INAV bietet drei Optionen die für das Flashen gesetzt werden können:
 - *No reboot sequence* - Wird nur benötigt, wenn die Bootpins überbrückt sind oder der Bootbutton während des Flashens gedrückt wird (modell-abhängig).
 - *Full chip erase* - Löscht alle Konfigurationsoptionen (Backup Vorhanden?). Sollte immer gesetzt werden, wenn vorher eine andere Software installiert war.
 - *Manual baud rate* - Festlegen der Baudrate für Bluetooth oder falls das zu flashende Modell die Standardgeschwindigkeit nicht unterstützt.
5. Der Prozess des Flashens besteht aus zwei Schritten: Zunächst wird die Firmware geladen (online oder Datei), dann wird das eigentliche Flashen über einen separaten Button gestartet. Dies sollte nicht unterbrochen werden, da hierdurch das Board gebrickt wird. Der zum Flashen benötigte Bootloader ist allerdings im ROM gespeichert und kann *nicht gebrickt* werden. Das

6. If necessary, the backup can now be restored via the *CLI* tab. This is done using the *Load from File* button. This allows the corresponding text file to be selected and executed using the *Execute* button.
6. Gegebenenfalls kann nun das Backup über den *CLI*-Tab wieder eingespielt werden. Dies geschieht über den Button *Load from File*. Damit kann die entsprechende Textdatei herausgesucht werden, um sie mit dem *Execute*-Button auszuführen.

Administration

The administration of INAV also occurs in the *INAV Configurator*. When the software is installed for the first time, a few things need to be configured to enable initial flight. These are described in this section.

When the Configurator is launched for the first time after installation, windows appear for setting the *Default Values* (presets). Here, you need to select the preset that is closest to the model you have built. After that, you can configure the UART ports (for GPS, VTX, etc.). However, this can also be done via the Ports tab. The further configuration steps are as follows:

1. The *Status* tab provides a basic overview. Among other things, the *Pre-armng checks* overview shows whether the drone is ready for takeoff. It may be necessary to adjust the board orientation beforehand. To do this, place the drone upright with the camera facing the screen of the configuring computer and then press the *Reset Z axis* button.
2. The *Calibration* tab (Figure 2.14) is used to calibrate the accelerometer and compass. To do this, the drone must be placed in different positions.

bedeutet, dass das Board immer wieder neu geflasht werden kann.

6. Gegebenenfalls kann nun das Backup über den *CLI*-Tab wieder eingespielt werden. Dies geschieht über den Button *Load from File*. Damit kann die entsprechende Textdatei herausgesucht werden, um sie mit dem *Execute*-Button auszuführen.

Administration

Die Administration von INAV erfolgt ebenfalls über den *INAV Configurator*. Wenn die Software das erste Mal installiert wurde, müssen einige Punkte konfiguriert werden, um das Fliegen initial zu ermöglichen.

Wird der Configurator zum ersten Mal nach der Installation aufgerufen, erscheinen Fenster zum Setzen der *Default-Values* (Presets). Hier gilt es das Preset auszuwählen, was am nächsten an dem gebauten Modell dran ist. Danach können die UART-Ports (für GPS, VTX, etc.) konfiguriert werden. Dies kann allerdings auch über den Ports-Tab durchgeführt werden. Die weiteren Konfigurationsschritte gestalten sich wie folgt:

1. Der *Status*-Tab stellt einen Überblick zur Verfügung. Unter anderem ist in der Übersicht *Pre-armng checks* zu sehen, ob die Drohne abheben kann. Eventuell muss hier vorher die Boardorientierung angepasst werden. Dazu wird die Drohne aufrecht mit Kamera Richtung Bildschirm des konfigurierenden Computers gestellt, um dann den *Reset Z axis* Button zu drücken.
2. Im *Calibration*-Tab (Abbildung 2.14) werden der Beschleunigungsmesser und der Kompass kalibriert. Dazu muss die Drohne jeweils in unterschiedliche Posen gebracht werden.

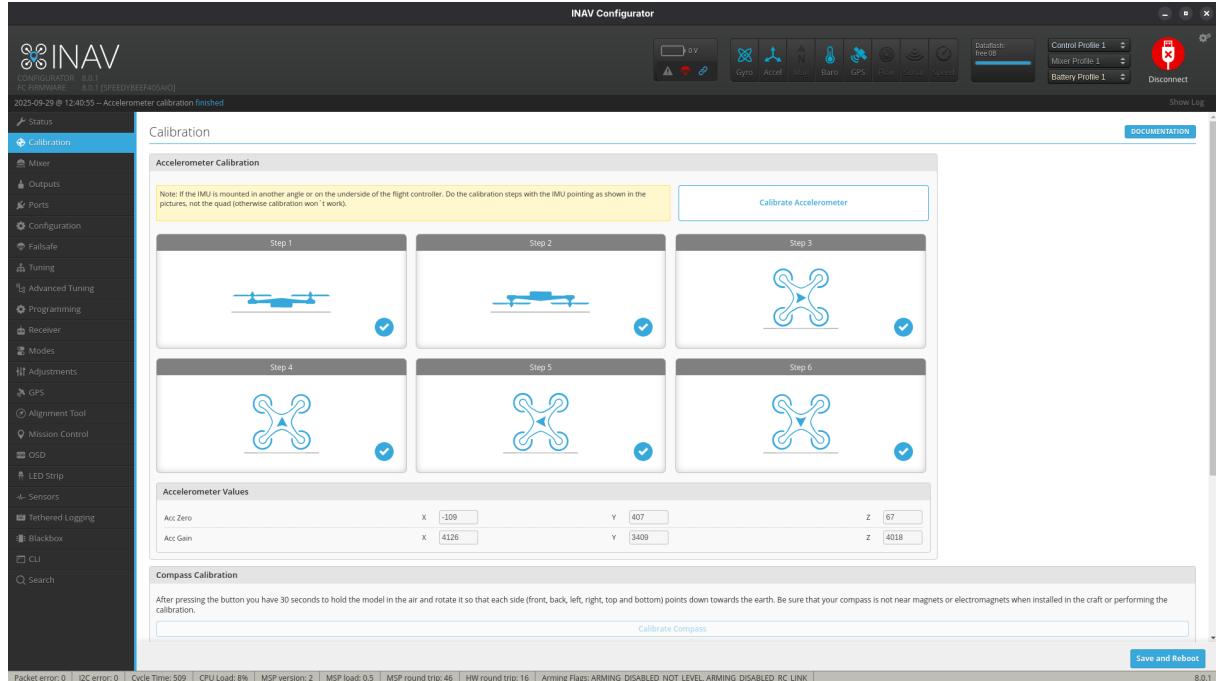


Figure 2.14: INAV Calibration Tab for Board Orientation, Compass and Flow Sensor

3. The *Mixer tab*, together with the *Outputs tab*, configures the drone's motors. It is crucial that the correct *Platform type* and *Mixer preset* (**Multirotor** and **QUAD X** for a quadcopter) are configured according to the type of drone used. If the motors do not rotate in the correct direction, this can be adjusted via the *Motor direction* and the assignment via the *Motor Mixer Wizard*.
4. In the *Outputs tab*, the motors are activated with the *Enable motor and servo output* option (see Figure 2.15). You should also ensure that **DSHOT300** is used as the *ESC protocol*. In addition, the *Motors IDLE power* can be set to 5% if a multicopter drone is used.
3. Der *Mixer-Tab* konfiguriert zusammen mit dem *Outputs-Tab* die Motoren der Drohne. Hier ist entscheidend, dass der richtige *Platform type* und das richtige *Mixer preset* (**Multirotor** und **QUAD X** für einen Quadcopter) entsprechend des verwendeten Drohnen-Typs konfiguriert sind. Sollten die Motoren nicht in die richtige Richtung drehen, kann die diese über die *Motor direction* angepasst werden und die Zuordnung über den *Motor Mixer Wizard*.
4. Im *Outputs-Tab* werden die Motoren aktiviert mit der Option *Enable motor and servo output* (siehe Abbildung 2.15). Außerdem sollte sichergestellt werden, dass **DSHOT300** als *ESC protocol* verwendet wird. Außerdem kann die *Motors IDLE power* auf 5% gesetzt werden, wenn eine Multicopter-Drohne verwendet wird.

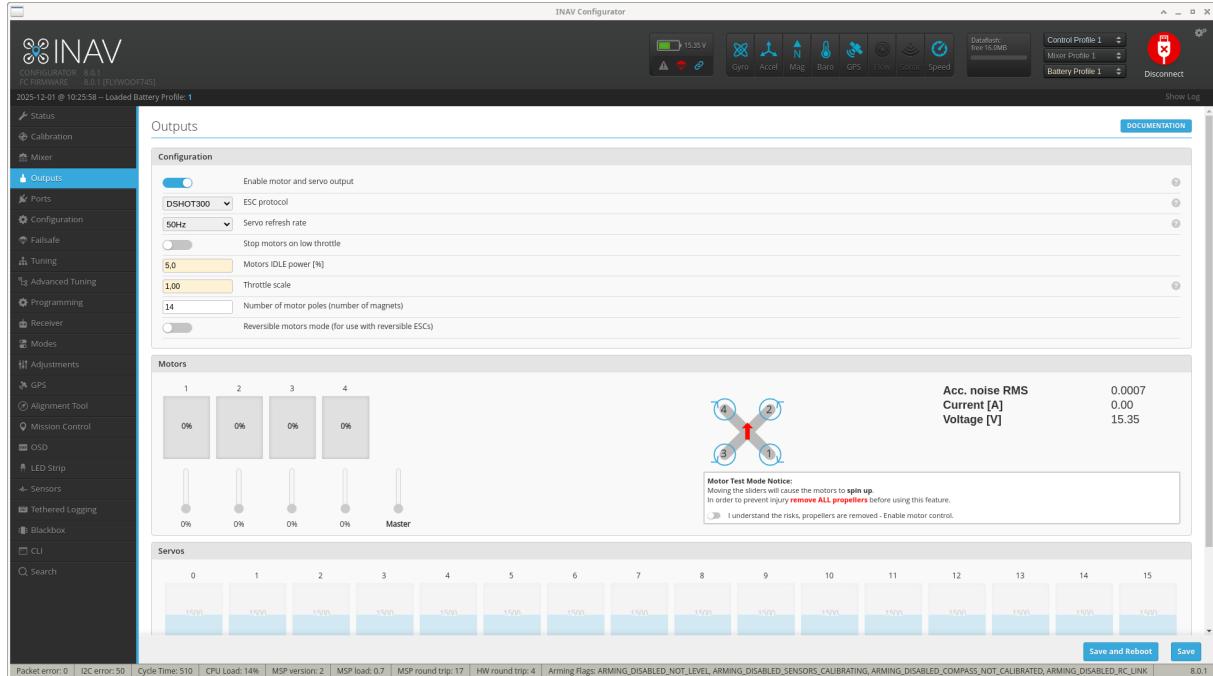


Figure 2.15: INAV Outputs Tab

5. In the *Ports tab*, the peripheral devices are configured, unless this was done in the pop-up menu at the beginning. This looks very similar to the Ports tab of the Betaflight Configurator, with the difference that *USB VCP* is hidden instead of partially grayed out (see Figure 2.3). How these are configured depends on the connected devices and the flight controller chip. In most cases, the configuration options for Betaflight (usually found in the respective device manuals) are transferable to INAV.
6. The sensors can be activated in the *Configuration tab* if this has not been done automatically after configuring the ports. If possible, the *I2C Speed* should be set to **800KHZ** here, if possible.
7. The *Failsafe tab* (Figure 2.16) is very important, as it defines the behavior should the remote control signal be lost. Here, you should ensure that either *Drop*, *Land*, or *Return-to-Home (RTH)* are configured so that the drone does not continue to fly uncontrollably.
5. Im *Ports-Tab* werden die Peripheriegeräte konfiguriert, insofern dies nicht im Pop-Up-Menü am Anfang gemacht wurde. Dieser sieht dem Ports-Tab des Betaflight-Configurators sehr ähnlich mit dem Unterschied, dass *USB VCP* ausgeblendet statt teilweise ausgegraut ist (Abbildung 2.3). Wie diese konfiguriert werden, hängt von den angeschlossenen Geräten und vom Flugcontrollerchip ab. Meist kann die Konfiguration für Betaflight aus den jeweiligen Handbüchern entnommen und übertragen werden.
6. Im *Configuration-Tab* können die Sensoren aktiviert werden, insofern dies nicht nach Konfiguration der Ports automatisch geschehen ist. Wenn möglich, sollte hier der *I2C Speed* auf **800KHZ** gestellt werden.
7. Der *Failsafe-Tab* (Abbildung 2.16) ist sehr wichtig, da in diesem das Verhalten definiert ist, sollte das Fernsteuerungssignal abreißen. Hier sollte sichergestellt werden, dass entweder *textslDrop*, *Land* oder *Return-to-Home (RTH)* konfiguriert sind, damit die Drohne nicht unkontrolliert weiterfliegt.

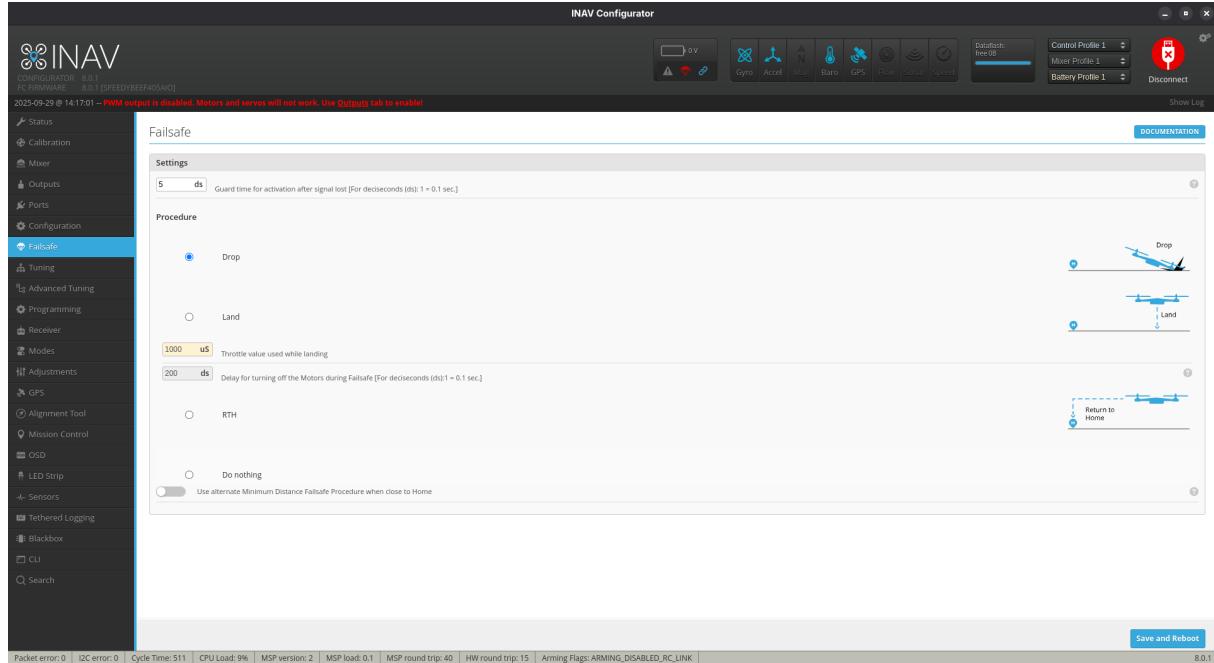


Figure 2.16: INAV Failsafe Tab

8. In the *Receiver* tab, the connection to the remote control from the side of the flight controller is configured. If an ExpressLRS remote control is used, **SERIAL** must be configured as the *Receiver type* and **CRSF** as the *Serial Receiver Provider*. In addition, you can check whether the channel mappings are correct. This tab is very similar to the Receiver tab in the Betaflight Configurator (Figure 2.6).
9. In the *GPS tab*, the *GPS for navigation and telemetry* option must be activated in order to use the GPS. The *Protocol* option must be set according to the specifications of the model used.
10. In the *Modes tab*, which is also very similar to the Betaflight version (Figure 2.7), only **ARM** needs to be configured (CH5 recommended). In addition, it is also worth configuring **ANGLE**, especially for beginner pilots, as it limits the maximum amount of pitch and roll, allowing for a smoother flight experience. **FAILSAFE** should also be configured, so it can be activated in case the drone behaves unex-
8. Im *Receiver-Tab* wird Flugcontrollerseitig die Verbindung zur Fernbedienung konfiguriert. Wird eine ExpressLRS-Fernbedienung verwendet, muss **SERIAL** als *Receiver type* und **CRSF** als *Serial Receiver Provider* konfiguriert werden. Darüber hinaus kann kontrolliert werden, ob die Channel-Mappings korrekt sind. Dieser Tab ist sehr ähnlich zum Receiver-Tab im Betaflight Configurator (Abbildung 2.6).
9. Im *GPS-Tab* muss die Option *GPS for navigation and telemetry* aktiviert werden, um das GPS nutzen zu können. Die Option *Protocol* ist entsprechend der Spezifikation des verwendeten Modells zu setzen.
10. Im *Modes-Tab*, der der Betaflight-Version sehr ähnlich ist (siehe Abbildung 2.7), muss grundsätzlich nur **ARM** konfiguriert werden. Dafür wird Kanal 5 empfohlen. Es lohnt sich außerdem, **ANGLE** einzustellen. Das ist besonders für Fluganfänger wichtig, weil dadurch die Neigung in Pitch und Roll begrenzt wird. Das macht das Fliegen einfacher. Auch **FAILSAFE** sollte aktiviert werden. So kann man reagieren, falls sich die Drohne unerwartet verhält.

pectedly. Modes for autopilot functions can be configured at a later stage.

- The *on-screen display (OSD)* can be configured according to the pilot's preferences. Useful elements that are not displayed by default include *Remaining Flight Time* and *Battery Remaining Percentage*. Depending on the VTX model, the *Video Format* must also be configured in this tab. The relevant info can usually be found in the manual of the VTX.

Modi für Autopilotfunktionalitäten können zu einem späteren Zeitpunkt konfiguriert werden.

- Das *on-screen display (OSD)* kann so konfiguriert werden, wie es vom Piloten bevorzugt wird. Nützliche Elemente, welche nicht standardmäßig eingeblendet werden sind zum Beispiel *Remaining Flight Time* und *Battery Remaining Percentage*. Je nach VTX-Modell muss in diesem Tab auch das *Video Format* konfiguriert werden. Relevante Informationen dazu können in der Regel im Handbuch des VTX gefunden werden.

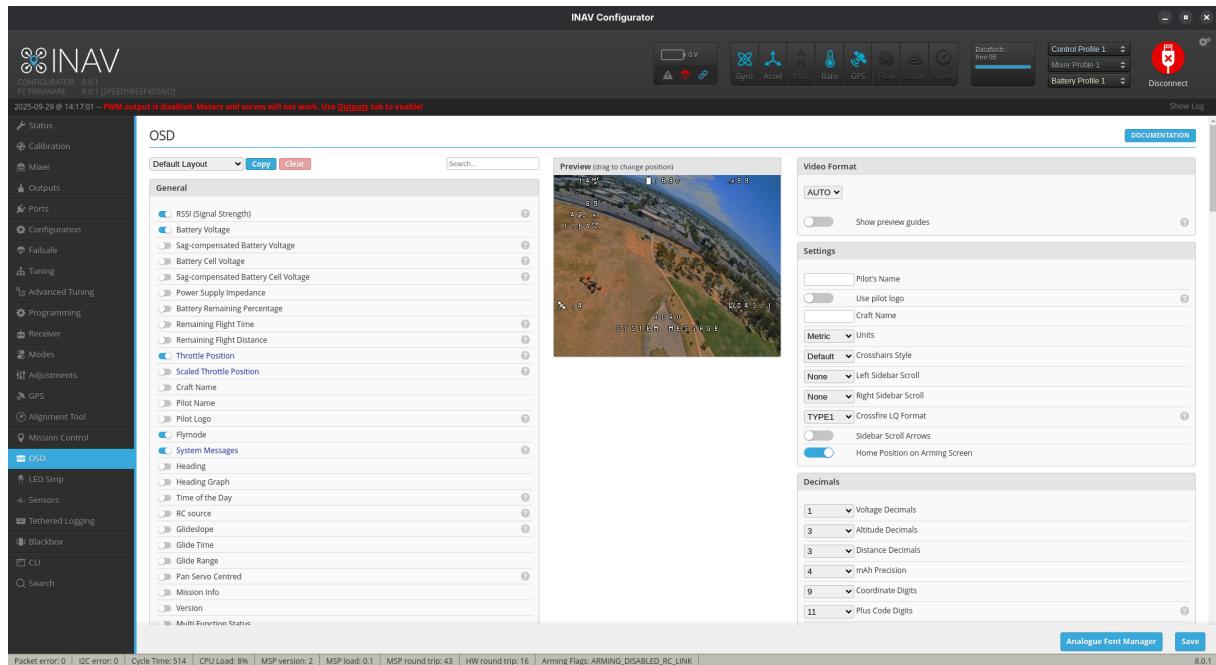


Figure 2.17: INAV OSD Tab

2.1.3

ArduPilot

As a flight controller firmware, ArduPilot focuses on fully autonomous flight and complex mission profiles. It enables mission planning with waypoints, automatic takeoff and landing, obstacle avoidance, follow-me functionality, and more. To achieve this, ArduPilot uses not only standard sensors such as GPS, magnetic compass, and velocity sensors but can also integrate LiDAR sensors for distance measurement and cameras for visual processing.

ArduPilot

ArduPilot legt als Flugcontroller-Firmware den Schwerpunkt auf vollautonome Flüge und komplexe Missionsprofile. Es ermöglicht die Missionsplanung mit Wegpunkten, automatischen Starts und Landungen, Hindernisvermeidung, Follow-Me-Funktion und vieles mehr. Hierfür nutzt ArduPilot nicht nur die üblichen Sensoren wie GPS, magnetischen Kompass und Geschwindigkeitsmessung, sondern kann auch LiDAR-

Of the flight controller firmwares presented in this document, ArduPilot requires the most memory. Ideally, 2 MB of flash memory should be available for running ArduPilot. Only flight controllers equipped with an H743 microcontroller provide this. With 1 MB of flash memory (F405 and F745 microcontrollers), operation is possible but with a reduced feature set. With only 512 kB (F411 and F722 microcontrollers), ArduPilot cannot be used at all.

Sensoren zur Distanzmessung und Kameras zur visuellen Erfassung einbeziehen.

Von den in diesem Dokument vorgestellten Flugcontroller-Firmwares benötigt ArduPilot den meisten Speicher. Idealerweise stehen zum Betrieb von ArduPilot 2 MB Flash-Speicher zur Verfügung. Dies bieten nur Flugcontroller mit einem H743-Mikrocontroller. Mit 1 MB Flash-Speicher (F405- und F745-Mikrocontroller) ist der Betrieb mit reduziertem Funktionsumfang möglich. Mit lediglich 512 kB (F411- und F722-Mikrocontroller) kann ArduPilot nicht verwendet werden.

Installation

Typically, FPV flight controllers are delivered with Betaflight pre-installed. If ArduPilot is not present on the device, the installation process consists of three steps: (1) installing the required drivers and flashing tools, (2) downloading the appropriate ArduPilot firmware, and (3) flashing the firmware to the flight controller.

To install ArduPilot on a compatible flight controller, the necessary drivers and tools must be installed first. The STM32CubeProgrammer, provided by STMicroelectronics, includes the required DFU drivers. The software is available for Windows, Linux, and macOS.

To obtain the ArduCopter firmware, the official ArduPilot firmware server at firmware.ardupilot.org/Copter/stable/ must be accessed. From the list of supported flight controller boards, the folder corresponding to the specific hardware (for example, FlywooF745, Pixhawk1, or MatekH743) is selected. Within that folder, the file ending in `_with_bl.hex` must be downloaded, as it includes the bootloader. If a board is supported by ArduCopter, it is listed in this directory and an appropriate firmware file is available. It is recommended to use the latest stable release to ensure compatibility and operational reliability.

Installation

Typischerweise werden FPV-Flugcontroller mit vorinstalliertem Betaflight ausgeliefert. Ist ArduPilot nicht auf dem Gerät vorhanden, besteht der Installationsvorgang aus drei Schritten: (1) Installation der erforderlichen Treiber und Flash-Werkzeuge, (2) Herunterladen der passenden ArduPilot-Firmware und (3) Flashen der Firmware auf den Flugcontroller.

Für die Installation von ArduPilot auf einem kompatiblen Flugcontroller müssen zunächst die erforderlichen Treiber und Werkzeuge installiert werden. Der STM32CubeProgrammer von STMicroelectronics enthält die benötigten DFU-Treiber. Die Software steht für Windows, Linux und macOS zur Verfügung.

Zum Herunterladen der ArduCopter-Firmware wird der offizielle ArduPilot-Firmware-Server unter firmware.ardupilot.org/Copter/stable/ aufgerufen. Aus der Liste der unterstützten Flugcontroller wird das Verzeichnis ausgewählt, das der verwendeten Hardware entspricht (z. B. FlywooF745, Pixhawk1 oder MatekH743). In diesem Verzeichnis wird die Datei mit der Endung `_with_bl.hex` heruntergeladen, da diese den Bootloader enthält. Ist ein Flugcontroller mit ArduCopter kompatibel, erscheint er in dieser Liste und die passende Firmwaredatei ist verfügbar. Die Verwendung der aktuellen stabilen Version wird empfohlen, um Kompatibilität und Betriebssicherheit sicherzustellen.

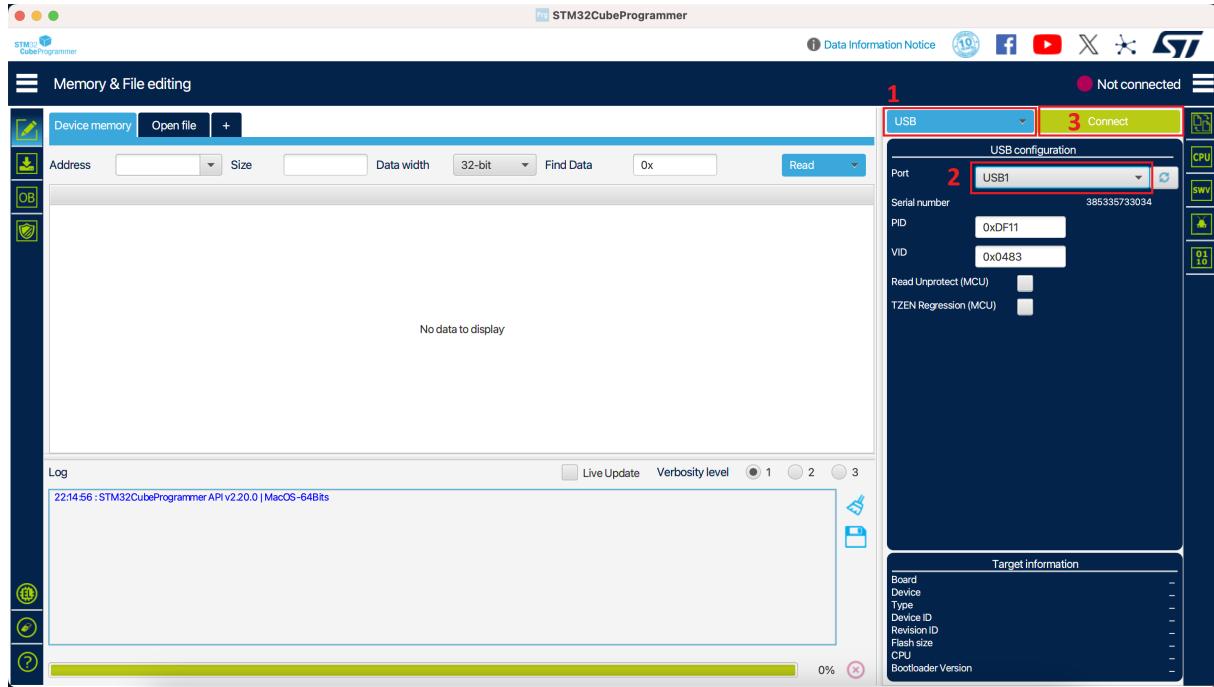


Figure 2.18: Flashing ArduPilot Firmware using STM32CubeProgrammer (Part 1/3)

Before starting the firmware installation process, DFU (Device Firmware Upgrade) mode must be activated on the flight controller. This mode enables the board to accept new firmware via USB. DFU mode is typically enabled by holding down the boot button (or bridging the BOOT pads) while the board is connected to a computer using a USB cable. In STM32CubeProgrammer, the connection method USB is selected (see Figure 2.18), after which the Connect function is used to establish communication with the flight controller. If the device is not detected, it should be verified that the STM32 drivers are correctly installed and that the USB cable supports data transfer.

Bevor der Installationsvorgang der Firmware begonnen wird, muss der DFU-Modus (Device Firmware Upgrade) auf dem Flugcontroller aktiviert werden. In diesem Modus kann das Gerät neue Firmware über die USB-Schnittstelle entgegennehmen. Der DFU-Modus wird in der Regel aktiviert, indem die Boot-Taste gedrückt gehalten oder die entsprechenden BOOT-Pads überbrückt werden, während der Flugcontroller per USB mit einem Computer verbunden wird. Im STM32CubeProgrammer wird anschließend die Verbindungsart USB ausgewählt (siehe Abbildung 2.18) und über die Schaltfläche Connect eine Verbindung zum Flugcontroller hergestellt. Wird das Gerät nicht erkannt, sollte überprüft werden, ob die STM32-Treiber korrekt installiert sind und ob das verwendete USB-Kabel Datenübertragung unterstützt.

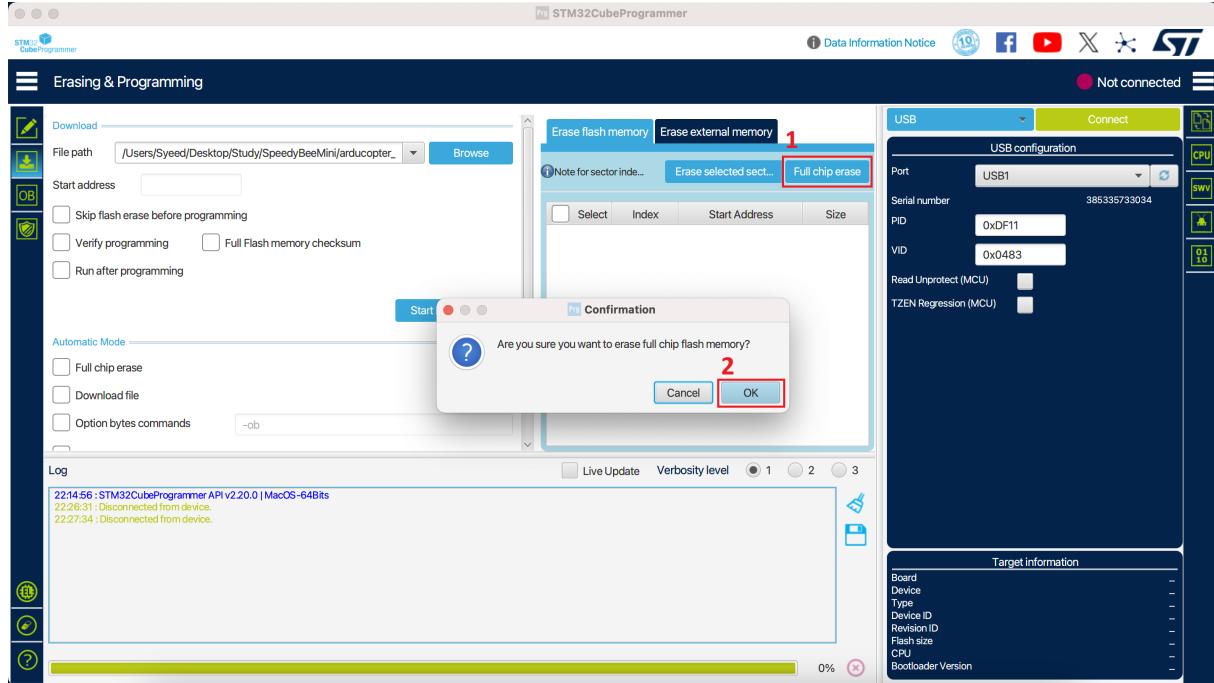


Figure 2.19: Flashing Ardupilot Frimware using STM32CubeProgrammer (Part 2/3)

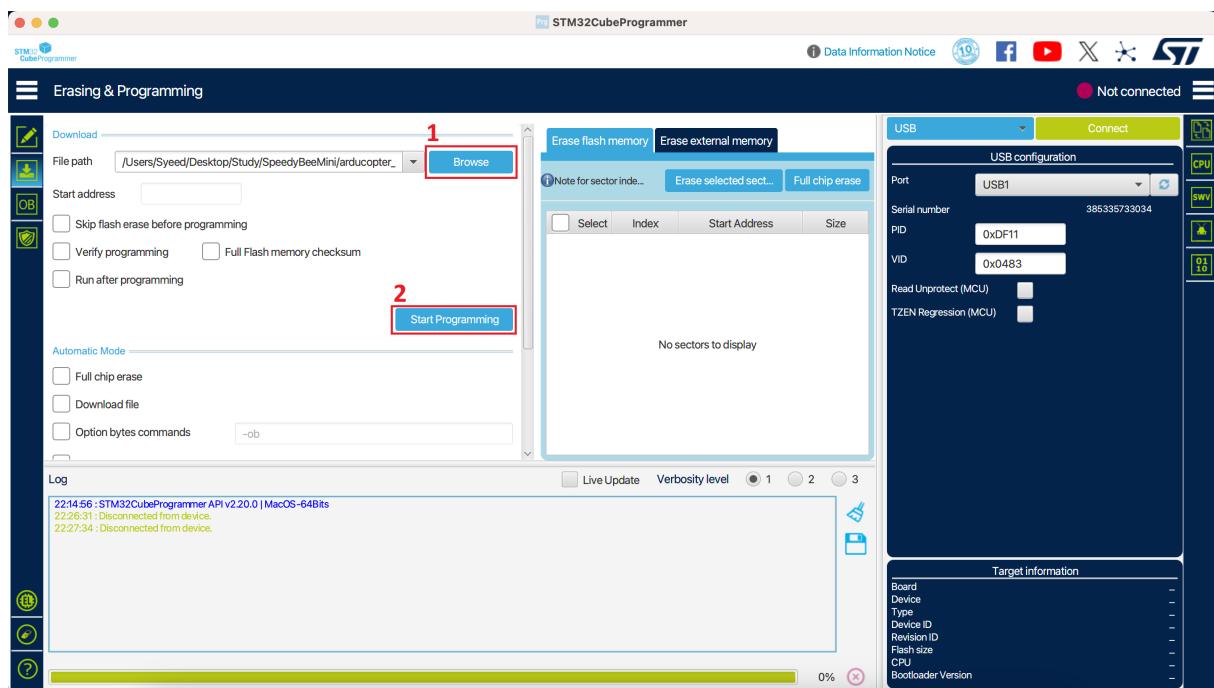


Figure 2.20: Flashing Ardupilot Frimware using STM32CubeProgrammer (Part 3/3)

After a successful connection has been established, the option Erase Full Chip must be selected (see Figure 2.19) to remove any existing firmware and prepare the board for a clean installation. Afterward, the Browse function is used to select the previously downloaded firmware file ending in `_with_bl.hex`. The flashing process is then started by clicking Start Programming (as shown in Figure 2.20). Once

Nach erfolgreicher Herstellung der Verbindung muss die Option Erase Full Chip ausgewählt werden (siehe Abbildung 2.19), um vorhandene Firmware vollständig zu entfernen und den Flugcontroller für eine saubere Installation vorzubereiten. Anschließend wird über die Funktion Browse die zuvor heruntergeladene Firmwaredatei mit der Endung `_with_bl.hex` ausgewählt. Der Flash-Vorgang wird danach

the programming operation has completed, the board must be disconnected and power-cycled to finalize the installation.

über die Schaltfläche Start Programming gestartet (siehe Abbildung 2.20). Nachdem der Programmierprozess abgeschlossen ist, muss der Flugcontroller getrennt und anschließend neu gestartet werden, um die Installation abzuschließen.

Mandatory Configuration

After the ArduPilot firmware has been flashed onto the flight controller, it is essential to perform a series of mandatory hardware configuration steps using the Mission Planner ground control software. These configurations align the firmware with the physical characteristics of the aircraft – such as its frame type, motor orientation, and sensor calibrations – ensuring that the autopilot interprets sensor data correctly and controls the motors appropriately. Completing these steps is a prerequisite for safe and stable flight performance.

Obligatorische Konfiguration

Nachdem die ArduPilot-Firmware auf den Flugcontroller aufgespielt wurde, müssen im Mission Planner eine Reihe verpflichtender Hardwarekonfigurationen vorgenommen werden. Diese Einstellungen passen die Firmware an die physikalischen Eigenschaften des Fluggeräts an – beispielsweise an den Rahmentyp, die Motorausrichtung und die Kalibrierung der Sensoren – und stellen sicher, dass der Autopilot Sensordaten korrekt interpretiert und die Motoren ordnungsgemäß ansteuert. Die Durchführung dieser Schritte ist eine Voraussetzung für einen sicheren und stabilen Flugbetrieb.



Figure 2.21: Main Page of Mission Planner

The Mission Planner software is officially available only for the operating systems Windows and Android. An alternative solution that also works well with ArduPilot is QGroundControl. In addition to Windows and Android, this software is also available for macOS and Linux.

Typically, the drone can be connected to the Mission Planner software by selecting Auto for the communication port, as shown in

Die Software Mission Planner ist offiziell nur für die Betriebssysteme Windows und Android verfügbar. Eine alternative Lösung, die ebenfalls gut mit ArduPilot zusammenarbeitet, ist QGroundControl. Diese Software ist neben Windows und Android auch für macOS und Linux verfügbar.

In der Regel kann die Drohne mit der Mission-Planner-Software verbunden werden, indem für den Kommunikationsport die Option Auto aus-

Figure 2.21. Once the connection has been established, the interface displays general sensor data from the drone. The following sections describe the mandatory hardware configuration steps required for operating a drone with ArduPilot.

gewählt wird, wie in Abbildung 2.21 dargestellt. Nach erfolgreicher Verbindung zeigt die Oberfläche allgemeine Sensordaten der Drohne an. Die folgenden Abschnitte beschreiben die verpflichtenden Hardwarekonfigurationsschritte, die für den Betrieb einer Drohne mit ArduPilot erforderlich sind.

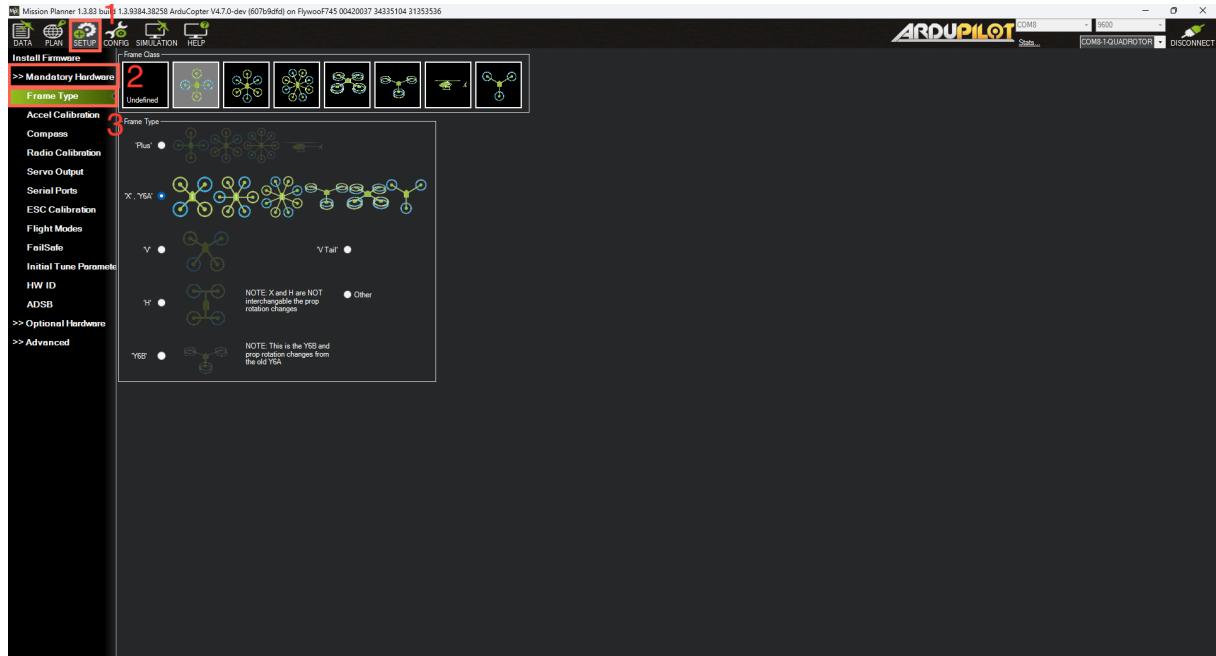


Figure 2.22: Mission Planner: Frame type configuration

Figure 2.22 shows the *Setup* page of Mission Planner. Selecting a configuration option writes the corresponding setting directly to the flight controller. No additional save action is required, as the changes are applied immediately.

In the *Frame Type* section, the appropriate frame class must be selected (e.g., Quad, Hexa, Heli). The specific frame type must also be chosen, such as X for an X-shaped frame or V-type if the aircraft has shorter front arms and longer rear arms.

Abbildung 2.22 zeigt die *Setup*-Seite des Mission Planner. Durch die Auswahl einer Konfigurationsoption wird die entsprechende Einstellung direkt auf den Flugcontroller geschrieben. Eine zusätzliche Speicheraktion ist nicht erforderlich, da die Änderungen automatisch übernommen werden.

Im Abschnitt *Frame Type* wird zunächst die passende Rahmenklasse ausgewählt (z. B. Quad, Hexa, Heli). Anschließend wird der konkrete Rahmentyp festgelegt, etwa X für einen X-förmigen Rahmen oder V-Type, wenn das Fluggerät kürzere vordere und längere hintere Ausleger besitzt.

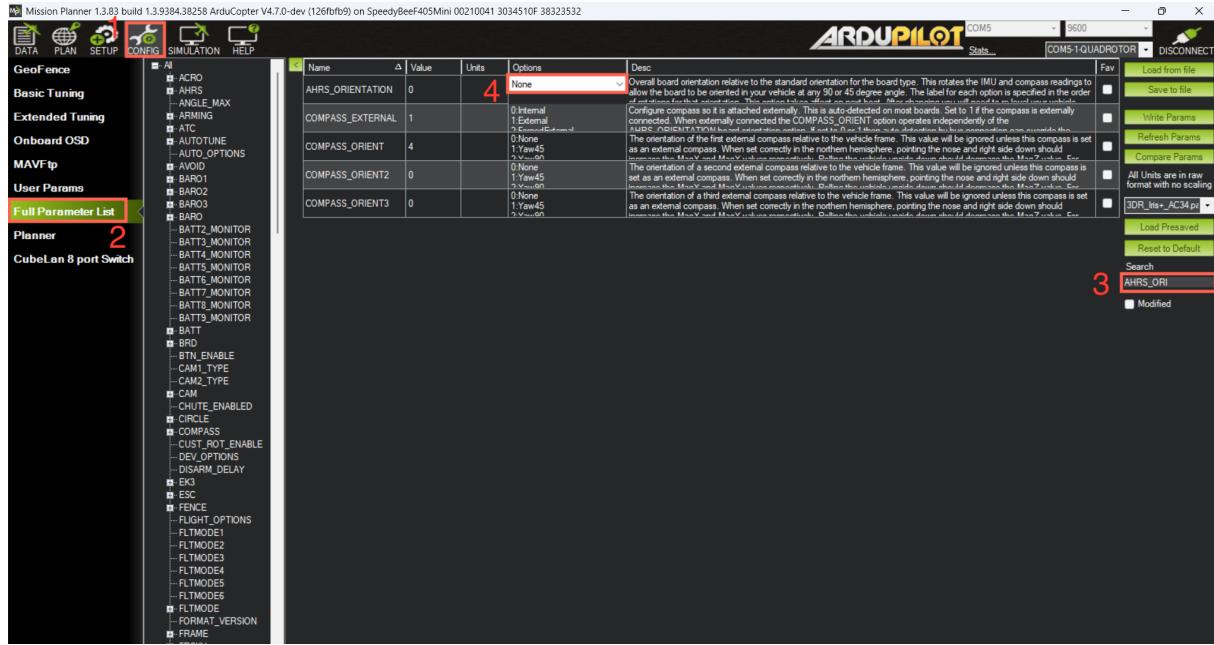


Figure 2.23: Mission Planner: Board orientation

Before performing the accelerometer calibration, it is essential to ensure that the flight controller is mounted with the correct orientation. An incorrect board orientation can lead to significant flight instability or loss of control. As shown in Figure 2.23, the orientation of the flight controller can be configured in Mission Planner by navigating to: Config tab → Full Parameter List → AHRS_ORIENTATION.

After selecting the appropriate orientation, the setting must be saved on the flight controller by clicking the Write Params button.

To verify that the orientation has been configured correctly, open the DATA tab in the top menu of Mission Planner. Then move the drone carefully in all directions (pitch, roll, and yaw) and observe the artificial horizon or attitude indicator. If the displayed movement matches the actual movement of the drone, the orientation has been set correctly.

Bevor die Kalibrierung des Beschleunigungssensors durchgeführt wird, muss sichergestellt werden, dass der Flugcontroller korrekt ausgerichtet montiert ist. Eine fehlerhafte Ausrichtung kann zu erheblicher Fluginstabilität oder sogar zum Kontrollverlust führen. Wie in Abbildung 2.23 dargestellt, kann die Ausrichtung des Flugcontrollers im Mission Planner über folgenden Pfad konfiguriert werden: Registerkarte Config → Full Parameter List → AHRS_ORIENTATION.

Nachdem die korrekte Ausrichtung ausgewählt wurde, muss die Einstellung über die Schaltfläche Write Params auf den Flugcontroller gespeichert werden.

Zur Überprüfung der korrekten Konfiguration wird im Mission Planner die Registerkarte DATA geöffnet. Anschließend wird die Drohne vorsichtig in alle Richtungen bewegt (Nick, Roll und Gier), während der künstliche Horizont bzw. der Lageindikator beobachtet wird. Entspricht die Anzeige den tatsächlichen Bewegungen der Drohne, ist die Ausrichtung korrekt eingestellt.

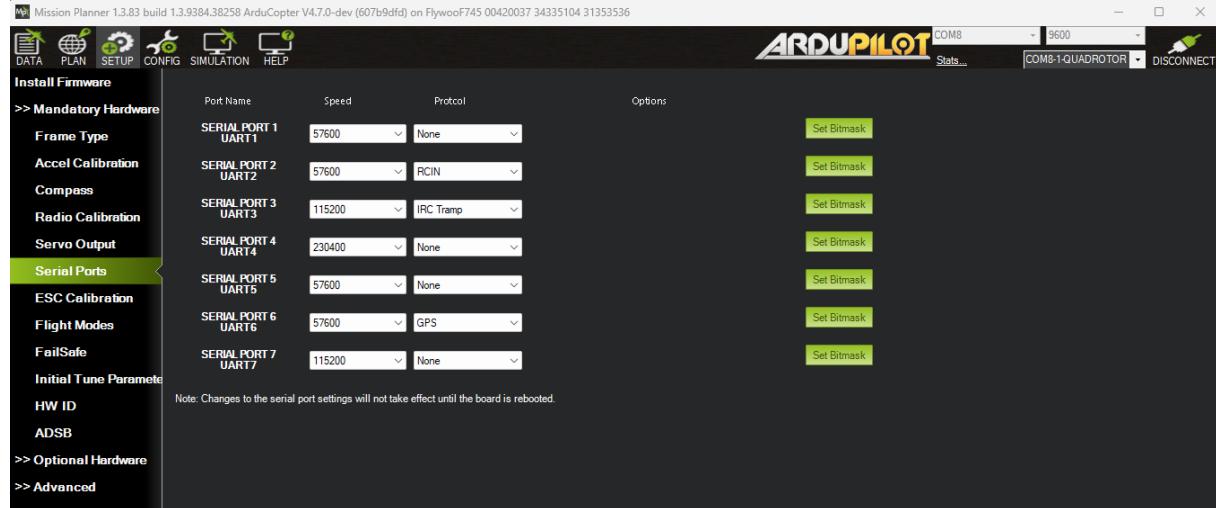


Figure 2.24: Mission Planner: Serial Port Configuration

Figure 2.24 shows the *Serial Ports* page within the *Mandatory Hardware* section in Mission Planner, where the available UART interfaces (UART1, UART2, ...) must be configured according to the peripherals connected to the flight controller. This configuration defines how external devices—such as the *GPS*-module, the receiver, telemetry units, or the *VTX*—communicate with the autopilot. In the example shown in Figure 2.24, UART2 is assigned the *RCIN* protocol for the radio receiver, UART3 is configured with the *IRC Tramp* protocol for VTX control, and UART6 is set to the *GPS* protocol for the connected GPS module.

Abbildung 2.24 zeigt die Seite *Serial Ports* im Bereich *Mandatory Hardware* des Mission Planner, auf der die verfügbaren UART-Schnittstellen (UART1, UART2, ...) entsprechend den an den Flugcontroller angeschlossenen Peripheriegeräten konfiguriert werden müssen. Diese Konfiguration legt fest, wie externe Geräte – wie das *GPS*-Modul, der Empfänger, Telemetrieinheiten oder der *VTX* – mit dem Autopiloten kommunizieren. Im in Abbildung 2.24 dargestellten Beispiel ist UART2 mit dem *RCIN*-Protokoll für den Empfänger belegt, UART3 mit dem *IRC Tramp*-Protokoll für die VTX-Steuerung konfiguriert und UART6 für das *GPS*-Protokoll des angeschlossenen GPS-Moduls eingestellt.

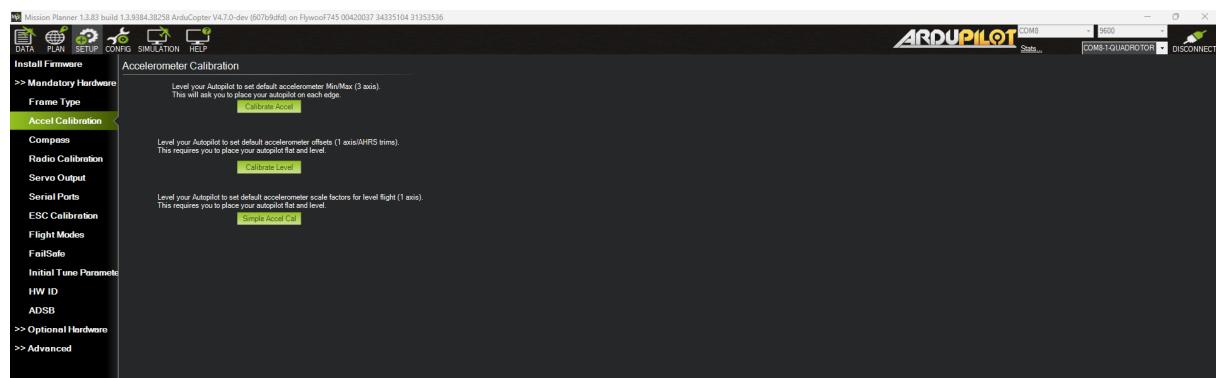


Figure 2.25: Mission Planner: Accelerometer calibration

The Accelerometer Calibration page, accessible from the Setup tab in Mission Planner, provides the interface for calibrating the on-board accelerometer (see Figure 2.25). To begin the process, click *Calibrate Accel* and follow the on-screen instructions to rotate the vehicle slowly around all three axes—yaw, pitch, and

Die Seite zur Kalibrierung des Beschleunigungssensors, die im Mission Planner über die Registerkarte *Setup* aufgerufen werden kann, stellt die Oberfläche zur Kalibrierung des im Flugcontroller integrierten Beschleunigungssensors bereit (siehe Abbildung 2.25). Zu Beginn des Vorgangs ist *Calibrate Accel* auszuwählen

roll. This procedure allows the flight controller to record reference data for each orientation and compute the necessary offsets.

After completing the calibration, restart the drone by reconnecting the flight controller. Once reconnected, place the drone on a level surface and select Calibrate Level to define the horizontal reference plane. The resulting calibration parameters allow the flight controller to maintain stable flight behavior, particularly in Stabilize mode.

und den Anweisungen auf dem Bildschirm zu folgen, um das Fluggerät langsam um alle drei Achsen – Gier, Nick und Roll – zu drehen. Dieser Prozess ermöglicht es dem Flugcontroller, Referenzdaten für jede Lage zu erfassen und die erforderlichen Offset-Werte zu berechnen.

Nach Abschluss der Kalibrierung wird das Fluggerät neu gestartet, indem der Flugcontroller erneut verbunden wird. Anschließend wird die Drohne auf eine ebene Oberfläche gestellt und die Option Calibrate Level ausgewählt, um die horizontale Referenzebene festzulegen. Die so ermittelten Kalibrierungsparameter ermöglichen es dem Flugcontroller, insbesondere im Stabilize-Modus, ein stabiles Flugverhalten sicherzustellen.

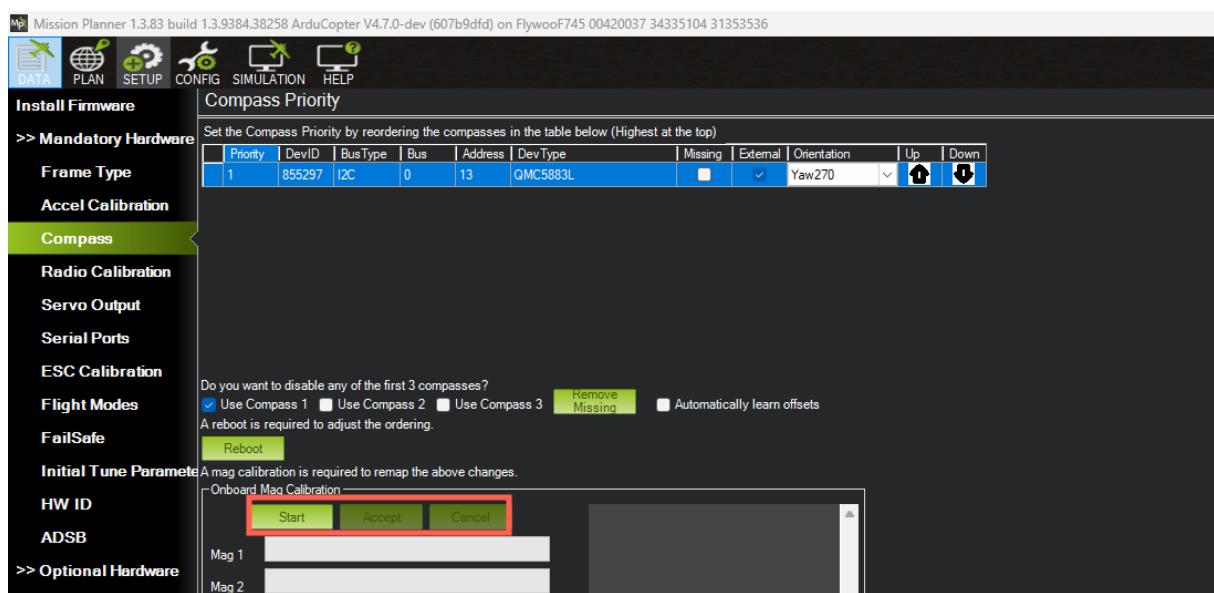


Figure 2.26: Mission Planner: Compass calibration

The Compass Calibration page in Mission Planner provides the interface for calibrating the onboard magnetometer, which is responsible for providing accurate heading information essential for directional navigation and yaw control.

Before beginning the calibration, ensure that the compass sensor is mounted in its final position and located away from strong magnetic sources such as motors, ESCs, or power wiring. In Mission Planner, navigate to the compass calibration page under the Setup tab and click Start (see Figure 2.26). During the calibration process, slowly rotate the vehicle around all three axes—yaw, pitch, and roll—as instructed

Die Seite zur Kompasskalibrierung im Mission Planner stellt die Oberfläche zur Kalibrierung des integrierten Magnetometers bereit. Dieses ist für die präzise Bestimmung der Flugrichtung verantwortlich und damit wesentlich für die Richtungsnavigation sowie für die Gierregelung.

Vor Beginn der Kalibrierung ist sicherzustellen, dass der Kompasssensor in seiner endgültigen Position montiert ist und sich nicht in der Nähe starker Magnetfelder befindet, beispielsweise in der Nähe von Motoren, ESCs oder stromführenden Leitungen. Im Mission Planner wird die Kompasskalibrierung über die Registerkarte Setup aufgerufen. Dort wird die Schaltfläche Start ausgewählt (siehe Abbildung 2.26). Während der Kalibrierung wird das Fluggerät

on screen, ensuring smooth and complete movements.

If the calibration fails, repeat the process several times until a successful calibration is achieved. Persistent failure to calibrate may indicate that the magnetometer is experiencing magnetic interference. In such cases, the compass should be repositioned farther from high-current components or wiring to minimize magnetic disturbances.

Once calibration is successfully completed, Mission Planner will indicate an acceptable calibration result. Finally, save the configuration and reboot the flight controller to apply the updated compass offsets.

entsprechend den Anweisungen langsam um alle drei Achsen – Gier, Nick und Roll – bewegt, wobei auf gleichmäßige und vollständige Bewegungen zu achten ist.

Sollte die Kalibrierung fehlschlagen, ist der Vorgang mehrmals zu wiederholen, bis eine erfolgreiche Kalibrierung erzielt wird. Treten weiterhin Fehlkalibrierungen auf, deutet dies möglicherweise auf magnetische Störungen hin. In diesem Fall sollte der Kompass weiter entfernt von stromstarken Komponenten oder Leitungen montiert werden, um magnetische Einflüsse zu minimieren.

Nach erfolgreichem Abschluss der Kalibrierung zeigt der Mission Planner ein entsprechendes Ergebnis an. Abschließend wird die Konfiguration gespeichert und der Flugcontroller neu gestartet, damit die ermittelten Kompass-Offsetwerte wirksam werden.



Figure 2.27: Mission Planner: Radio Calibration

Figure 2.27 shows the Radio Calibration page in Mission Planner. With the RC transmitter bound and connected to the drone, select Calibrate Radio and move all control sticks (gimbals) and all relevant switches through their full range while observing the bar indicators. Here, one can ensure that every channel – whether it's a stick or a switch – is correctly detected and mapped by the flight controller.

Abbildung 2.27 zeigt die Seite zur Funkfernsteuerungskalibrierung im Mission Planner. Nachdem die Fernsteuerung mit der Drohne gekoppelt und verbunden wurde, wird die Schaltfläche Calibrate Radio ausgewählt. Anschließend werden alle Steuerknüppel (Gimbals) sowie alle relevanten Schalter über ihren gesamten Bewegungsbereich betätigt, während die angezeigten Balken beobachtet werden. Auf diese Weise lässt sich sicherstellen, dass jeder Kanal – unabhängig davon, ob es sich um einen Steuerknüppel oder einen Schalter handelt – korrekt erkannt und dem Flugcontroller zugeordnet wird.

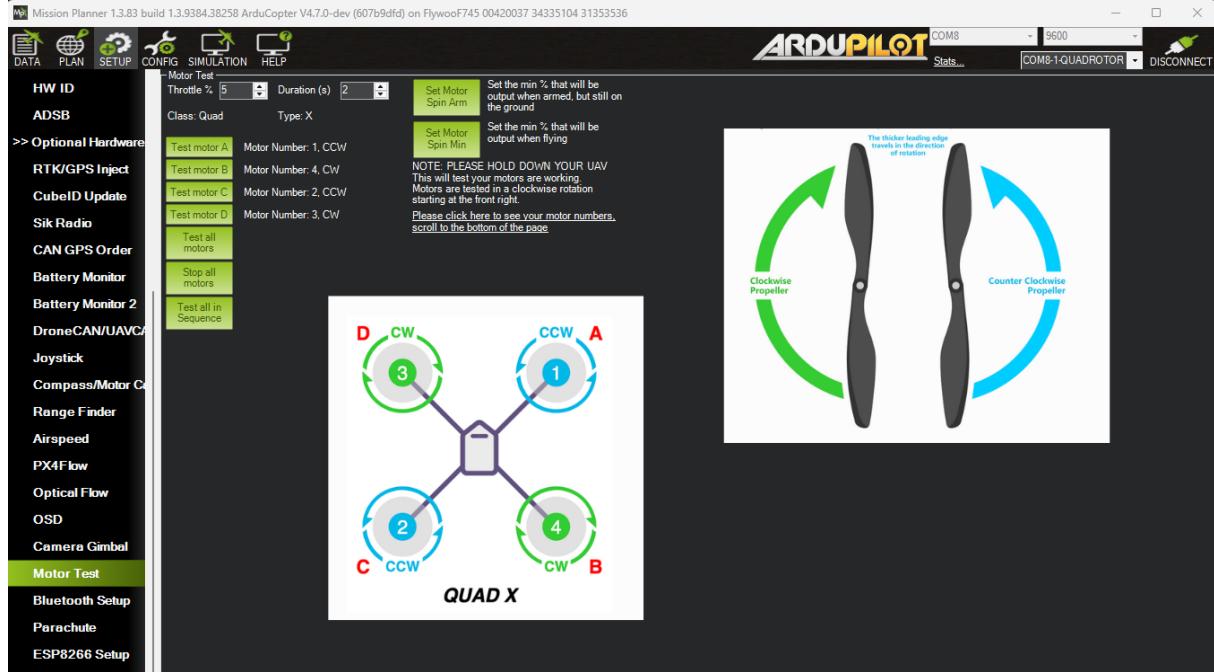


Figure 2.28: Mission Planner: Motor Test

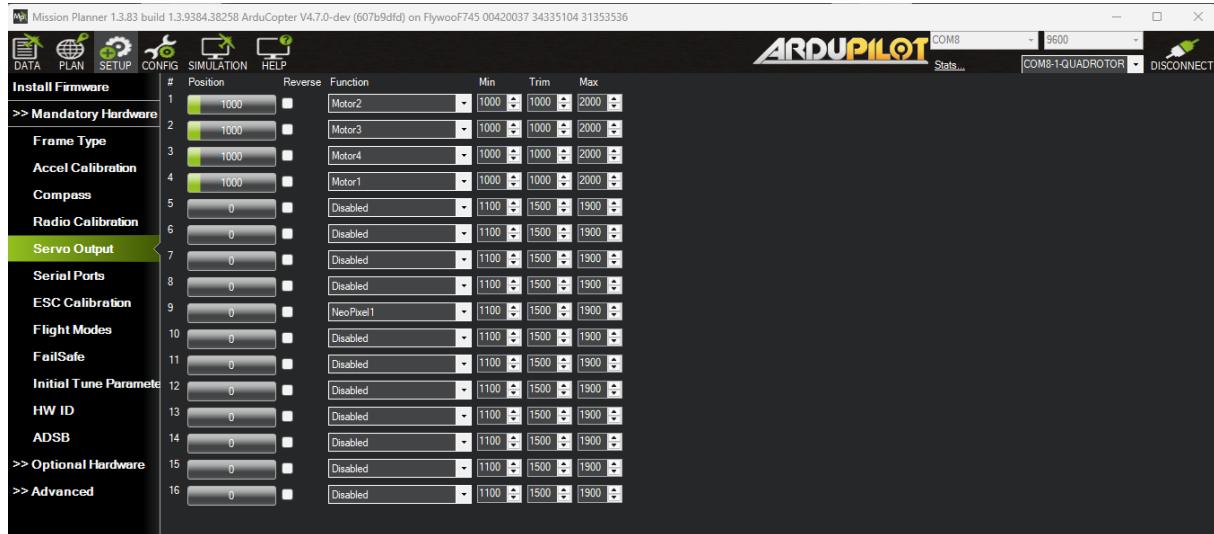


Figure 2.29: Mission Planner: Servo Output

Figure 2.28 shows the Motor Test page in Mission Planner, which is used to confirm that each motor on a quadcopter is correctly assigned and rotating in the proper direction. With the battery connected and the propellers removed, you can test motors A, B, C, and D individually and verify that they match the Quad-X layout shown in the figure and spin in their required CW/CCW directions. If a motor responds incorrectly or rotates the wrong way, the Servo Output page (Figure 2.29) allows you to re-map motor outputs or reverse a motor direction when needed. While this documenta-

Abbildung 2.28 zeigt die Motor-Test-Seite im Mission Planner, die zur Überprüfung dient, ob jeder Motor eines Quadcopters korrekt zugeordnet ist und in die richtige Richtung rotiert. Mit angeschlossener Batterie und entfernten Propellern können die Motoren A, B, C und D einzeln getestet werden. Dabei wird überprüft, ob sie dem in der Abbildung dargestellten Quad-X-Layout entsprechen und jeweils in der erforderlichen Uhrzeiger- bzw. Gegenuhrzeigerrichtung laufen. Reagiert ein Motor nicht wie erwartet oder dreht er in die falsche Richtung, können auf der Seite Servo Output (Abbildung 2.29)

tion focuses on quadcopters, users working with other frame types should refer to the ArduPilot motor connection guide for additional configurations: <https://ardupilot.org/copter/docs/connect-escs-and-motors.html>.

Motorzuordnungen angepasst oder Drehrichtungen geändert werden. Während sich diese Dokumentation auf Quadrokopter konzentriert, sollten Anwender anderer Rahmenkonfigurationen die ArduPilot-Anleitung zu Motoranschlüssen für weitere Konfigurationsoptionen konsultieren: <https://ardupilot.org/copter/docs/connect-escs-and-motors.html>.

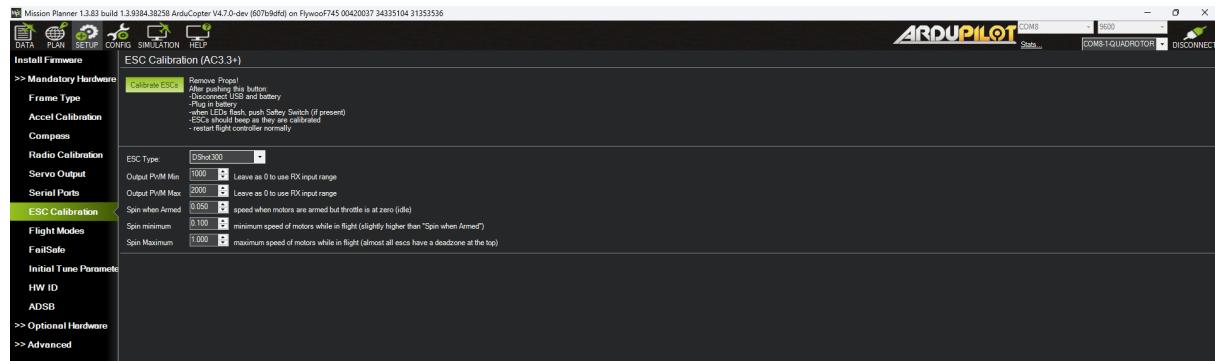


Figure 2.30: Mission Planner: ESC Calibration

Figure 2.30 shows the ESC Calibration page in Mission Planner, which is used to standardize the throttle range of analog ESCs so that all motors respond uniformly. This page also provides configuration options for ESC-related parameters, such as the ESC Type (e.g., DShot300 or DShot600) and various Spin settings that control motor behavior at idle and across the throttle range. It is important to note that digital 4-in-1 ESCs using DShot protocols do not require calibration, as their throttle interpretation is digital rather than PWM-based. For most modern flight controllers and ESCs, selecting the appropriate DShot protocol in the ESC Type field is sufficient to ensure correct operation.

Abbildung 2.30 zeigt die ESC-Kalibrierungsseite im Mission Planner, die zur Vereinheitlichung des Gasbereichs analoger ESCs dient, sodass alle Motoren ein einheitliches Ansprechverhalten aufweisen. Auf dieser Seite können außerdem ESC-bezogene Parameter konfiguriert werden, einschließlich des ESC-Typs (z. B. DShot300 oder DShot600) sowie verschiedener Spin-Einstellungen, welche das Motorverhalten im Leerlauf und über den gesamten Gasbereich definieren. Es ist zu beachten, dass digitale 4-in-1-ESCs, die DShot-Protokolle verwenden, keine Kalibrierung benötigen, da die Gasinterpretation digital und nicht PWM-basiert erfolgt. Für die meisten modernen Flugcontroller und ESCs genügt es, im Feld ESC Type das passende DShot-Protokoll auszuwählen, um einen korrekten Betrieb sicherzustellen.

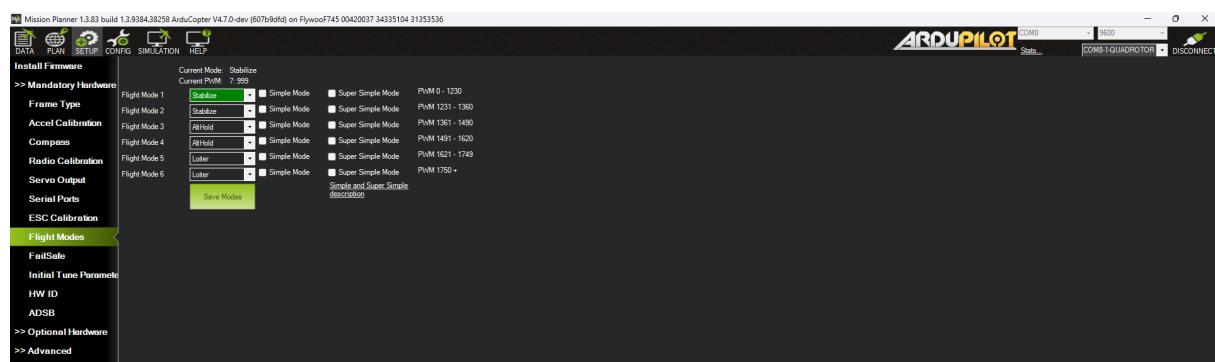


Figure 2.31: Mission Planner: Flight Modes

Figure 2.31 shows the Flight Modes page in Mission Planner, where the user assigns the drone's primary operating modes such as Stabilize, AltHold, or Loiter. Normally these modes are controlled by a multi-position switch on the radio transmitter, and each switch position corresponds to a specific PWM range sent from the RC receiver. Mission Planner detects these PWM ranges and lets the user map each one to a desired flight mode. If a different RC channel needs to be used for flight-mode selection, the user can go Config tab → FullParameter List and search **FLTMODE_CH** to set the appropriate channel parameter (e.g., Channel7 or another chosen channel).

Abbildung 2.31 zeigt die Seite Flight Modes im Mission Planner, auf der die primären Betriebsmodi der Drohne – wie Stabilize, AltHold oder Loiter – zugewiesen werden. In der Regel werden diese Flugmodi über einen Mehrpositionsschalter der Fernsteuerung gewählt, wobei jede Schalterposition einem bestimmten PWM-Bereich entspricht, der vom RC-Empfänger übertragen wird. Mission Planner erkennt diese PWM-Bereiche und ermöglicht deren Zuordnung zu den gewünschten Flugmodi. Soll ein anderer RC-Kanal für die Auswahl des Flugmodus verwendet werden, kann dies über die folgende Navigation konfiguriert werden: Registerkarte Config → Full Parameter List und anschließend Suche nach **FLTMODE_CH**, um den entsprechenden Kanalparameter festzulegen (z. B. Kanal 7 oder ein anderer gewünschter Kanal).

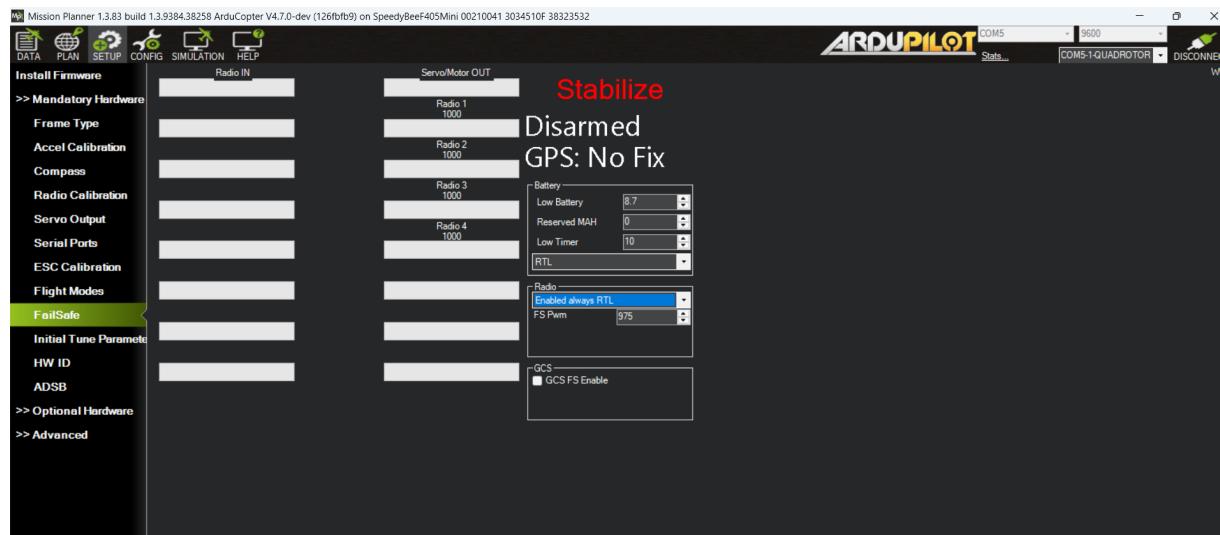


Figure 2.32: Mission Planner: Failsafe

On the Failsafe page (Figure 2.32), the user can configure how the flight controller should respond to critical conditions such as battery too low or loss of RC signal. This page allows detailed control over the drone's behavior during such events, including options like executing a controlled landing or returning to the launch point using GPS. For most applications, it is recommended to select Return-to-Launch (RTL) as the primary failsafe action, as this provides a safe and predictable response by guiding the drone back to its takeoff position.

Auf der Seite Failsafe (Abbildung 2.32) kann konfiguriert werden, wie der Flugcontroller auf kritische Situationen wie eine zu niedrige Batteriespannung oder den Verlust des RC-Signals reagieren soll. Diese Seite ermöglicht eine detaillierte Festlegung des Verhaltens der Drohne in solchen Fällen, einschließlich Optionen wie einer kontrollierten Landung oder der Rückkehr zum Startpunkt mittels GPS. Für die meisten Anwendungen wird empfohlen, Return-to-Launch (RTL) als primäre Failsafe-Aktion zu wählen, da diese Option eine sichere und vorhersehbare Reaktion ermöglicht, indem die Drohne zuverlässig zu ihrer Startposition zurückgeführt wird.

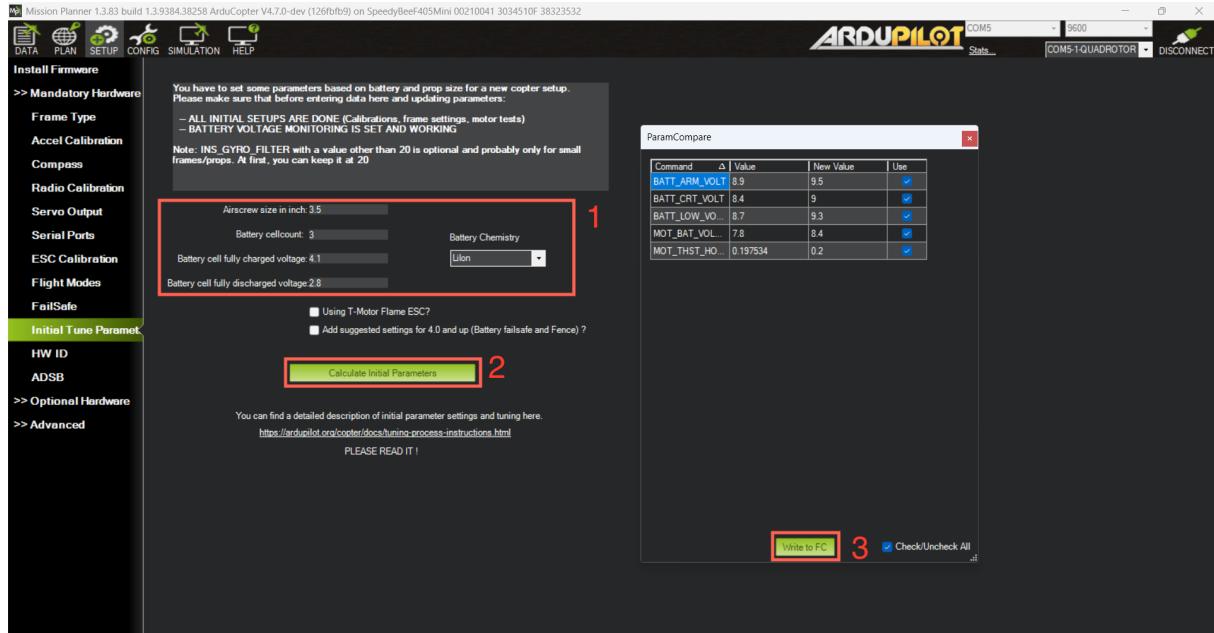


Figure 2.33: Mission Planner: Initial tune parameter

Figure 2.33 shows the Initial Tune Parameters page in Mission Planner, which provides automatically generated starting values for the drone's PID tuning based on its basic physical characteristics. Correctly establishing these initial parameters is essential for achieving stable flight before performing any manual adjustments. To use this tool, the user must enter key information such as the propeller size (Airscrew size in inch), the battery cell count, and the battery chemistry (e.g., Li-ion). Accurate input is important, as these values directly affect the tuning parameters calculated by Mission Planner.

The Onboard OSD page (2.34), accessible from Config Tab → Onboard OSD. From this interface, various telemetry elements – such as battery voltage, current, flight mode, GPS status, or altitude – can be enabled, repositioned, and arranged on the OSD screen. These settings are written directly to the OSD chip on the flight controller, ensuring that essential in-flight data is clearly visible during FPV operations.

Abbildung 2.33 zeigt die Seite Initial Tune Parameters im Mission Planner. Diese Funktion stellt automatisch generierte Ausgangswerte für das PID-Tuning der Drohne bereit, basierend auf ihren grundlegenden physikalischen Eigenschaften. Die korrekte Festlegung dieser Anfangsparameter ist entscheidend, um vor manuellen Anpassungen einen stabilen Flug zu gewährleisten. Für die Nutzung dieses Werkzeugs müssen wesentliche Angaben wie die Propellergröße (Airscrew size in inch), die Anzahl der Batteriezellen sowie die Batterietechnologie (z. B. Li-Ion) eingetragen werden. Eine präzise Eingabe ist wichtig, da diese Werte unmittelbar die vom Mission Planner berechneten Tuningparameter beeinflussen.

Die Seite Onboard OSD (Abbildung 2.34), erreichbar über Config Tab → Onboard OSD, ermöglicht die Konfiguration verschiedener Telemetrieelemente wie Batteriespannung, Strom, Flugmodus, GPS-Status oder Höhe. Diese Elemente können auf dem OSD-Bildschirm aktiviert, verschoben und angeordnet werden. Die Einstellungen werden direkt auf den OSD-Chip des Flugcontrollers geschrieben und stellen sicher, dass während des FPV-Flugs alle relevanten Informationen gut sichtbar dargestellt werden.

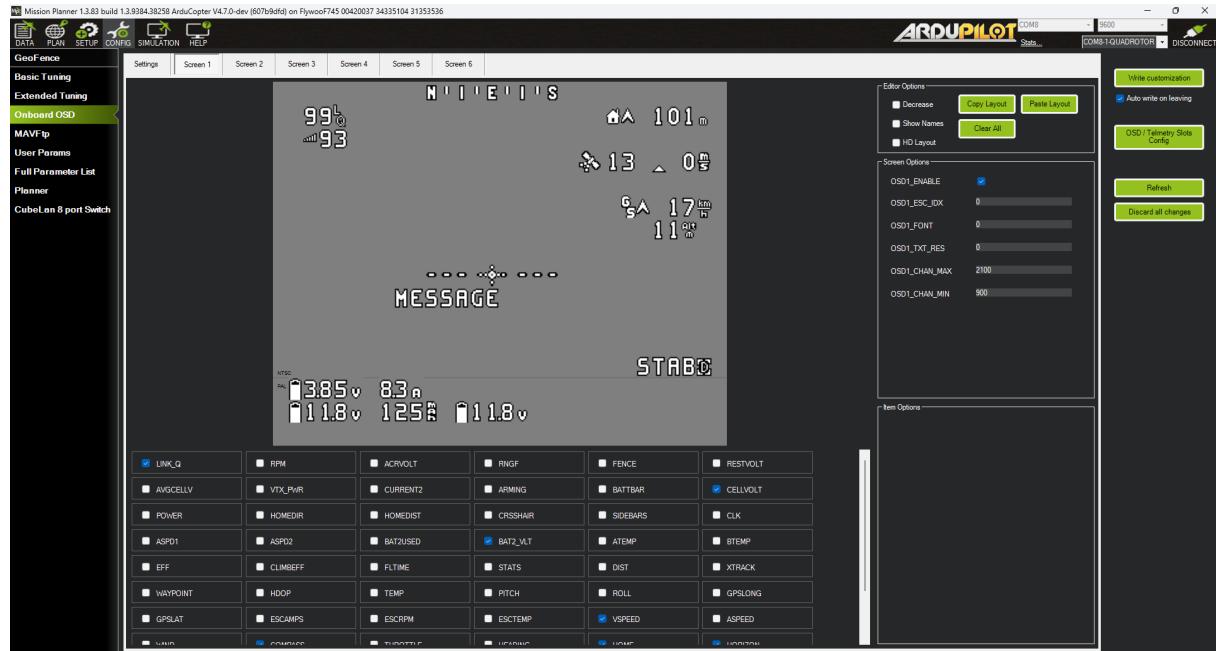


Figure 2.34: Mission Planner: Onboard OSD

Before First Flight

Note for Betaflight users: If you are transitioning from Betaflight, be aware that the Pitch axis of the RC input is typically reversed in ArduPilot compared to Betaflight's default. You may need to set the `RC2_REVERSED` parameter to 1 in the Full Parameter List of Config tab to correct this, ensuring your pitch stick inputs correspond to the drone's expected pitch movements.

It is recommended to configure an arming switch before your first flight. To do this, open the Config → Full Parameter List and search for `RC<channel number>_OPTION` (for example, `RC5_OPTION`). Set its value to 153, which designates that channel as the arming channel.

2.1.4

PX4

PX4 is an open-source autopilot firmware widely used in research, industry, and robotics. Unlike Betaflight and iNAV, which primarily focus on FPV and hobbyist use cases, PX4 provides an advanced and modular architecture designed for

Vor dem ersten Flug

Hinweis für Betaflight-Nutzer: Beim Wechsel von Betaflight zu ArduPilot ist zu beachten, dass die Pitch-Achse der RC-Eingaben in ArduPilot in der Regel entgegengesetzt zu den Betaflight-Standardeinstellungen arbeitet. Gegebenenfalls muss der Parameter `RC2_REVERSED` im Bereich Config → Full Parameter List auf den Wert 1 gesetzt werden, damit die Steuerbewegungen des Pitch-Sticks den erwarteten Bewegungen der Drohne entsprechen.

Es wird empfohlen, vor dem ersten Flug einen separaten Arming-Schalter zu konfigurieren. Hierzu wird im Mission Planner der Bereich Config → Full Parameter List geöffnet und nach `RC<channel number>_OPTION` gesucht (beispielsweise `RC5_OPTION`). Der Wert dieses Parameters wird anschließend auf 153 gesetzt, wodurch dieser RC-Kanal als Arming-Kanal definiert wird.

PX4

PX4 ist eine Open-Source-Autopilot-Firmware, die häufig in Forschung, Industrie und Robotik eingesetzt wird. Im Gegensatz zu Betaflight und iNAV, die primär auf FPV- und Hobbyanwendungen ausgerichtet sind, bietet PX4 ei-

highly autonomous flight operations. Typical application domains include mapping, inspection, multi-robot cooperation, and integration with companion computers such as the Raspberry Pi or NVIDIA Jetson platforms.

PX4 supports a broad range of flight controller boards, including the Pixhawk series and STM32-based systems. The firmware offers extensive functionality such as mission planning, position and altitude hold, obstacle avoidance (when paired with appropriate sensors), geofencing, and fully autonomous waypoint navigation. Configuration and calibration are performed using the ground control software QGroundControl, which provides tools for sensor calibration, flight mode configuration, and monitoring of telemetry data. PX4 communicates using the MAVLink protocol, enabling seamless integration with ground stations, companion computers, and external tools.

Its modular design allows developers to extend or replace subsystems, making PX4 particularly suitable for research projects and complex multi-sensor setups. Compared to the other systems discussed in this chapter, PX4 offers the highest flexibility and scalability but requires more powerful hardware and a more extensive setup effort. It is therefore best suited for applications requiring autonomy, sensor fusion, or advanced mission capabilities rather than purely manual FPV flight.

ne erweiterbare und modulare Architektur für hochgradig autonome Flugoperationen. Typische Einsatzgebiete sind Kartierung, Inspektionen, Multi-Roboter-Kooperation und die Integration von Companion-Computern wie dem Raspberry Pi oder NVIDIA Jetson.

PX4 unterstützt eine große Bandbreite an Flugcontroller-Platinen, darunter die Pixhawk-Serie sowie andere STM32-basierte Systeme. Die Firmware stellt umfangreiche Funktionen bereit, darunter Missionsplanung, Position- und Höhenhaltung, Hinderniserkennung (mit geeigneter Sensorik), Geofencing und vollständig autonome Wegpunktnavigation. Die Konfiguration und Kalibrierung erfolgt über die Bodenstationssoftware QGroundControl, die Werkzeuge zur Sensorkalibrierung, Flugmoduskonfiguration und Überwachung von Telemetriedaten bietet. PX4 nutzt das MAVLink-Protokoll und ermöglicht damit eine nahtlose Anbindung an Bodenstationen, Companion-Computer und externe Werkzeuge.

Durch das modulare Design lassen sich einzelne Subsysteme erweitern oder austauschen, was PX4 besonders für Forschungsprojekte und komplexe Multisensorsysteme geeignet macht. Im Vergleich zu den anderen in diesem Kapitel beschriebenen Systemen bietet PX4 die größte Flexibilität und Skalierbarkeit, erfordert jedoch leistungsfähigere Hardware und mehr Konfigurationsaufwand. Es eignet sich daher insbesondere für Anwendungen, die Autonomie, Sensorfusion oder erweiterte Missionfähigkeiten benötigen und weniger für rein manuell geflogene FPV-Drohnen.

2.2

Remote Control Firmware: EdgeTX

EdgeTX is an operating system for radio transmitters used in model aircraft and drones. It is open-source software and runs on transmitters from many well-known manufacturers. EdgeTX generates the control signals, which are then transmitted by the ExpressLRS firmware (see Section 2.3) running on the transmitter module.

Installation is performed via the web application buddy.edgetx.org. This section describes the installation process on a Radiomaster Boxer using a PC.

Fernbedienung Firmware: EdgeTX

EdgeTX ist ein Betriebssystem für Fernsteuerungen im Modellbau. Es handelt sich um freie Software, die auf Fernbedienungen vieler bekannter Hersteller läuft. EdgeTX erzeugt die Steuersignale, welche anschließend von der Firmware ExpressLRS (siehe Abschnitt 2.3) des Sendemoduls übertragen werden.

Die Installation erfolgt über die Webanwendung buddy.edgetx.org. Dieser Abschnitt beschreibt die Installation auf einer Radiomaster Boxer mithilfe eines PCs.

With the transmitter turned off, connect it to the PC using its USB-C port on the top. In the web application, select the desired version and the correct device (see Figure 2.35). Clicking the *Flash via USB* button starts the next step.

Die Fernbedienung wird im ausgeschalteten Zustand über die obere USB-C-Schnittstelle mit dem PC verbunden. In der Webanwendung wird die gewünschte Version sowie das passende Gerät ausgewählt (siehe Abbildung 2.35). Die Schaltfläche *Flash via USB* führt zum nächsten Schritt.

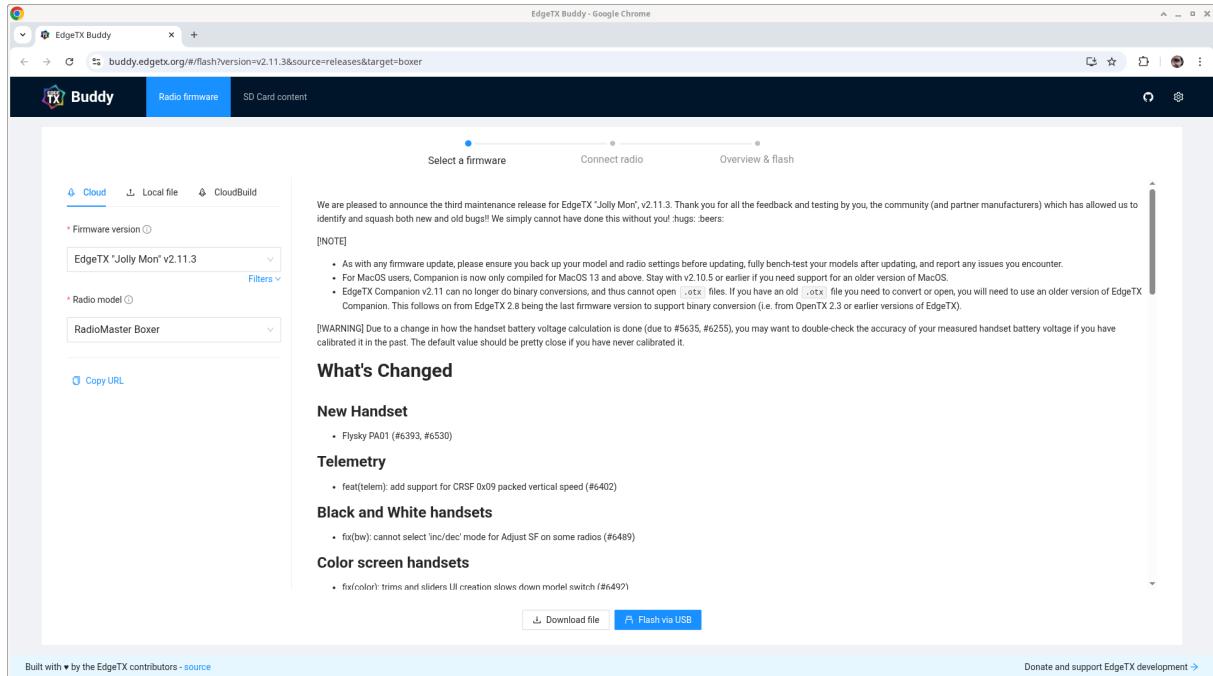


Figure 2.35: EdgeTX Installation Step 1

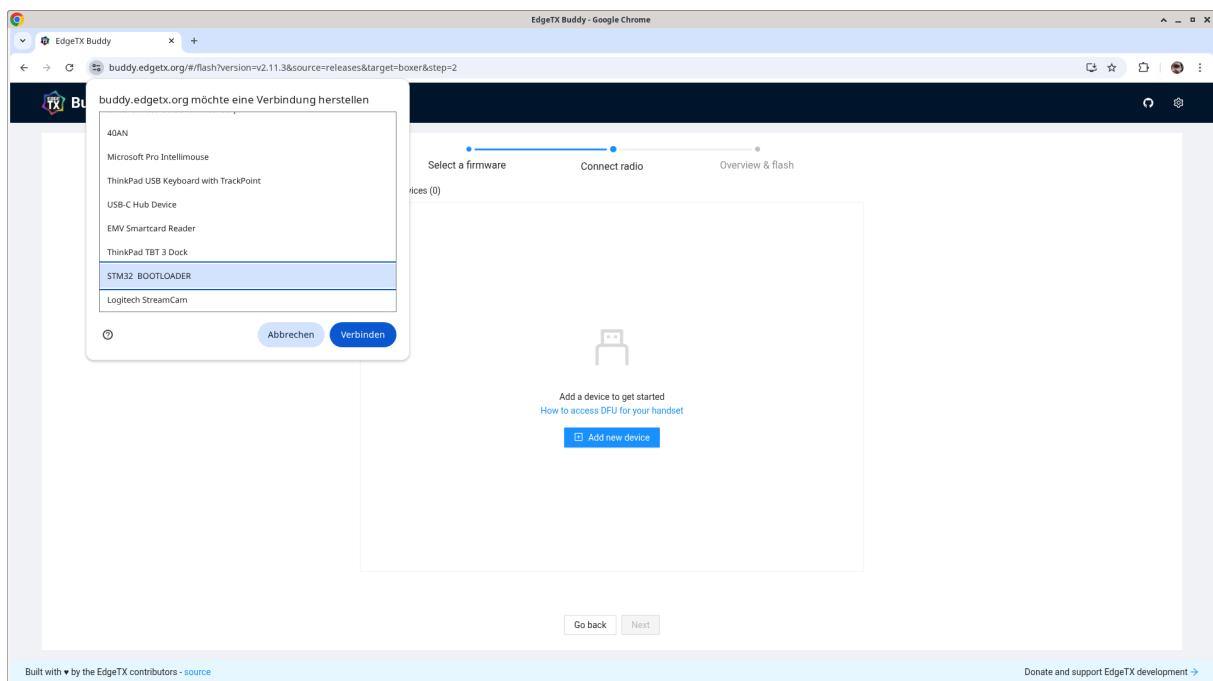


Figure 2.36: EdgeTX Installation Step 2

In the second step, the correct device must be selected. It appears under the name STM32 BOOTLOADER (see Figures 2.36 and 2.37). If the entry STM32 BOOTLOADER is not present and debugging with command-line tools such as `lsusb` does not help, it is likely that the USB-C cable does not transfer data, since many USB-C cables can only be used for battery charging.

The final step is to start the installation by clicking the *Start flashing* button (see Figure 2.38).

Im zweiten Schritt wird das korrekte Gerät ausgewählt. Dieses erscheint unter dem Namen STM32 BOOTLOADER (siehe Abbildung 2.36 und Abbildung 2.37). Wenn der Eintrag STM32 BOOTLOADER nicht erscheint und auch die Fehlersuche mit Kommandozeilenwerkzeugen wie `lsusb` nicht weiterhilft, ist es wahrscheinlich, dass das USB-C-Kabel keine Daten überträgt, da viele USB-C-Kabel nur zum Laden eines Akkus verwendet werden können.

Der letzte Schritt ist der Start der Installation über die Schaltfläche *Start flashing* (siehe Abbildung 2.38).

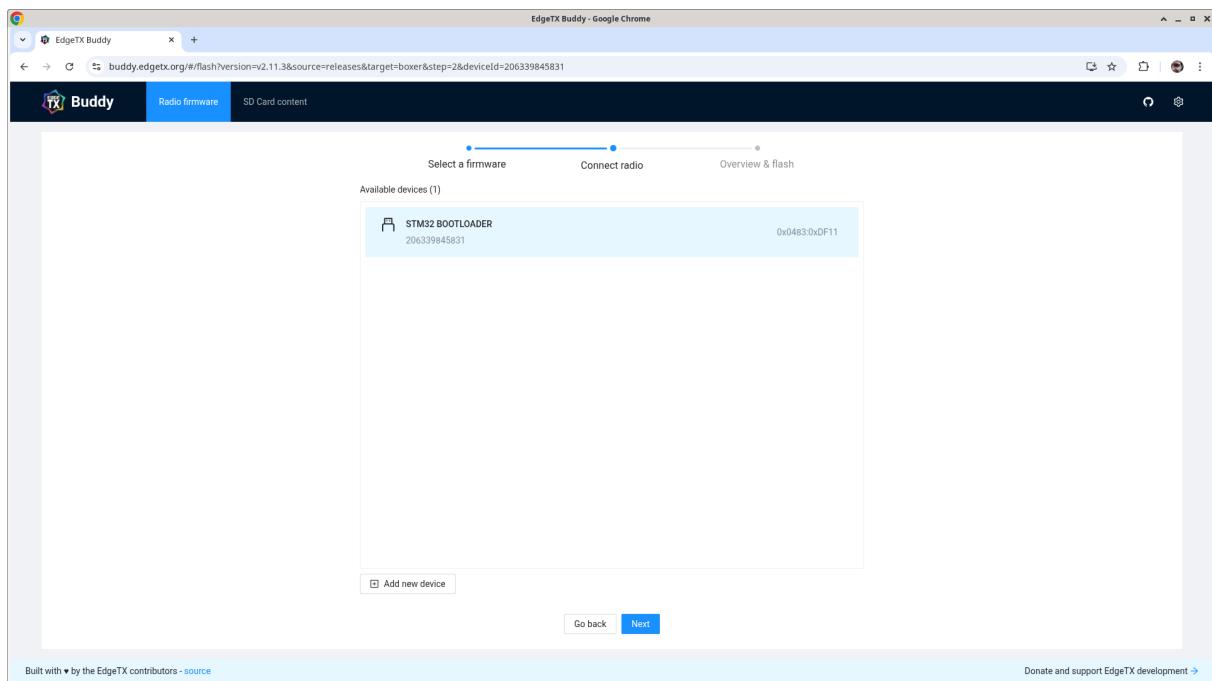


Figure 2.37: EdgeTX Installation Step 3

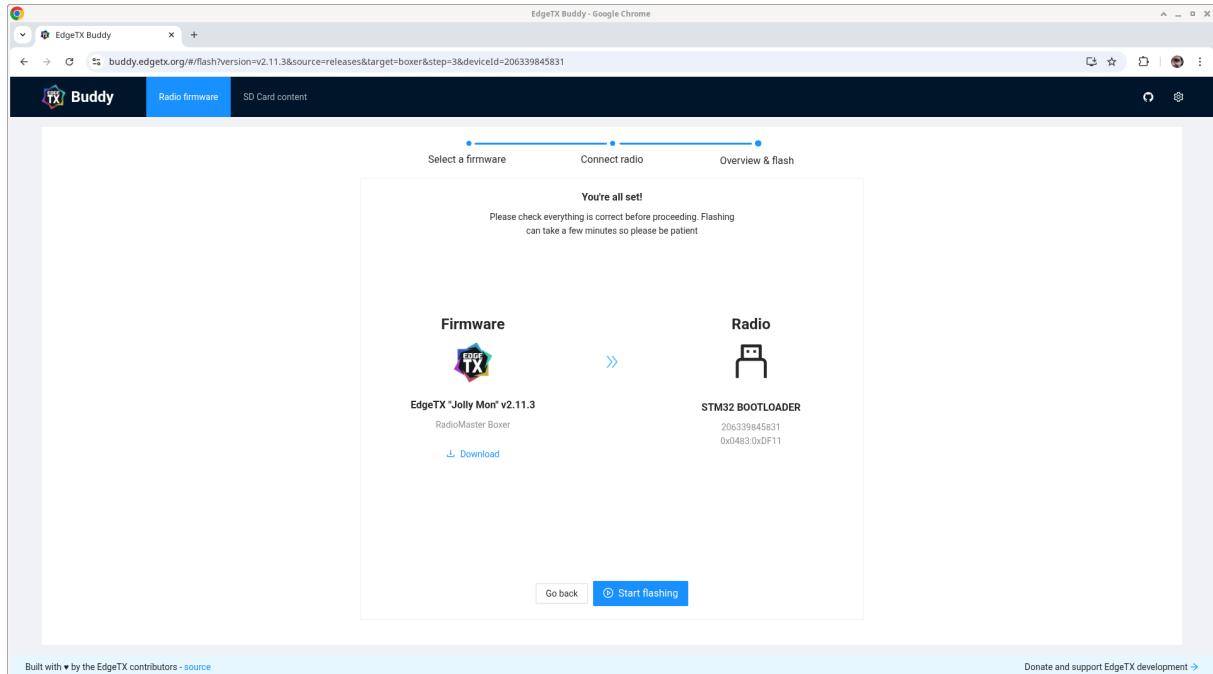


Figure 2.38: EdgeTX Installation Step 4

2.3

Transmitter and Receiver Firmware: ExpressLRS

The firmware of the transmitter module in the remote control and the receiver in the drone must both use compatible firmware versions. Ideally, the firmware versions of the transmitter and receiver are identical. This section focuses exclusively on the ExpressLRS firmware, including its installation and configuration.

Installation is performed using the locally installed application ExpressLRS Configurator. Installation packages and the source code can be found here:

<https://github.com/ExpressLRS/ExpressLRS-Configurator/releases>

In the ExpressLRS Configurator, the desired version, device type, connection method (EdgeTX Passthrough, UART, or Wi-Fi), frequency band usage permissions (Listen Before Talk), and the binding phrase are defined. The binding phrase must match the one later used on the receiver (see Figure 2.39).

Sendemodul- und Empfänger-Firmware: ExpressLRS

Die Firmware des Sendemoduls in der Fernbedienung und des Empfängers in der Drohne müssen beide mit kompatibler Firmware ausgestattet sein. Idealerweise sind bei Sender und Empfänger die Versionen identisch. Dieser Abschnitt befasst sich ausschließlich mit der Firmware ExpressLRS sowie deren Installation und Konfiguration.

Die Installation erfolgt über die lokal installierte Anwendung ExpressLRS Configurator. Installationspakete und der Quellcode befinden sich hier:

Im ExpressLRS Configurator werden die gewünschte Version, das Gerät, die Zugriffsart (EdgeTX Passthrough, UART oder Wi-Fi), die erlaubte Nutzung des Frequenzbands (Listen Before Talk) sowie die Binding Phrase festgelegt. Diese muss mit der später beim Empfänger verwendeten Binding Phrase übereinstimmen (siehe Abbildung 2.39).

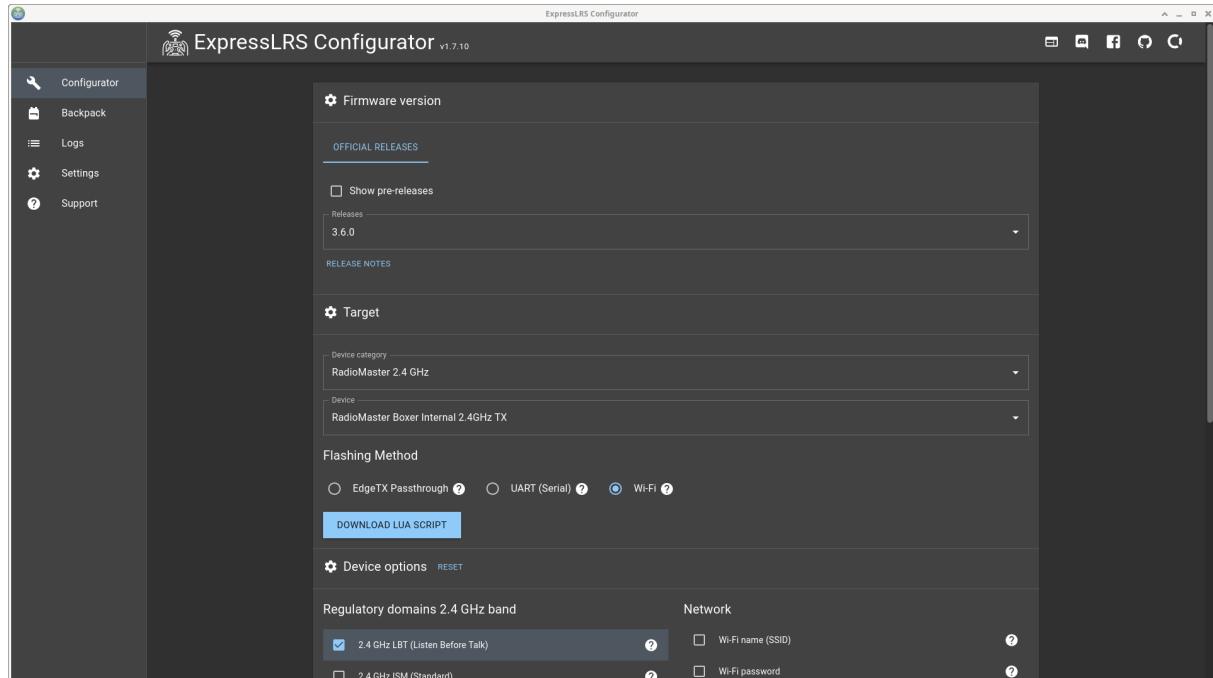


Figure 2.39: ExpressLRS Installation on Sender: Step 1

The default Wi-Fi parameters can also be adjusted here, although this option will not be discussed further in this section. The firmware file is created by clicking the *BUILD* button (see Figure 2.40).

Auch die standardmäßig verwendeten WLAN-Parameter können hier angepasst werden, auf diese Option wird an dieser Stelle jedoch nicht weiter eingegangen. Die Erstellung der Firmware-Datei erfolgt über die Schaltfläche *BUILD* (siehe Abbildung 2.40).

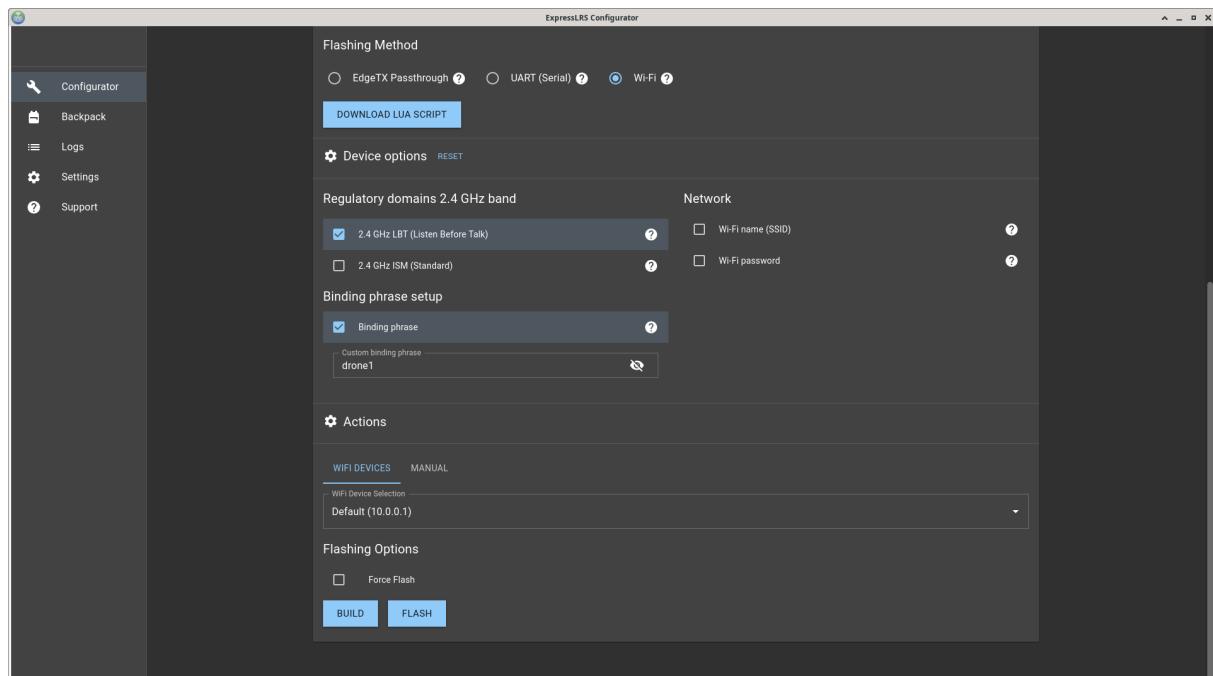


Figure 2.40: ExpressLRS Installation on Sender: Step 2

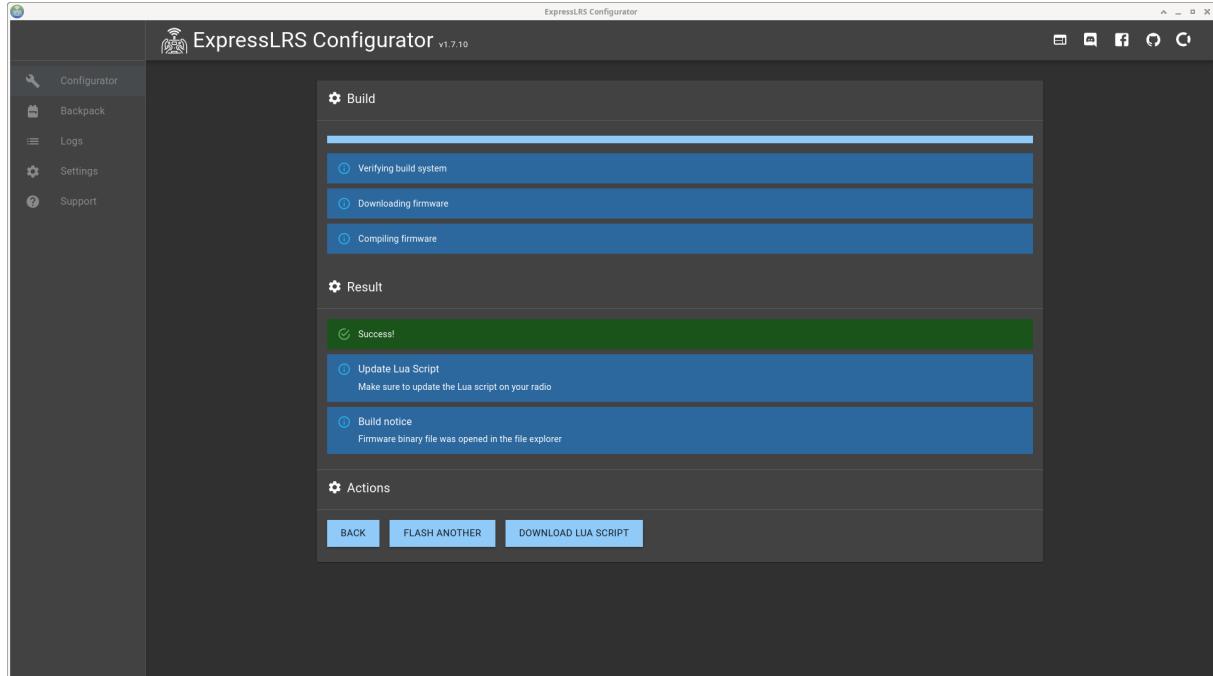


Figure 2.41: ExpressLRS Installation on Sender: Step 3

Figure 2.41 shows that the firmware file has been successfully created. Typically, a file manager window opens at this point, displaying the directory where the firmware file is located.

In the next step, the firmware file must be uploaded to the transmitter module of the remote control. One option is installation via the transmitter's Wi-Fi module. This is activated through SYS→Tools→ExpressLRS→Wi-Fi Connectivity→Enable Wi-Fi. Once the network **ExpressLRS TX** (password: **expresslrs**) becomes available, the PC must be connected to it. The transmitter module uses the IP address **10.0.0.1**. When this address is opened in a web browser, a simple interface allows, among other things, firmware updates (see Figure 2.42).

Abbildung 2.41 zeigt, dass die Firmware-Datei erstellt wurde. Üblicherweise öffnet sich an dieser Stelle ein Dateimanager-Fenster, das den Pfad anzeigt, in dem sich die Firmware-Datei befindet.

Im nächsten Schritt muss die Firmware-Datei auf das Sendemodul der Fernbedienung aufgespielt werden. Eine Möglichkeit hierzu ist die Installation über das WLAN-Modul des Senders. Dieses wird über SYS→Tools→ExpressLRS→Wi-Fi Connectivity→Enable Wi-Fi aktiviert. Sobald das Funknetz **ExpressLRS TX** (Passwort: **expresslrs**) verfügbar ist, muss der PC damit verbunden werden. Das Sendemodul hat die IP-Adresse **10.0.0.1**. Wird diese Adresse über einen Browser aufgerufen, kann über eine einfache Webseite unter anderem die Firmware aktualisiert werden (siehe Abbildung 2.42).

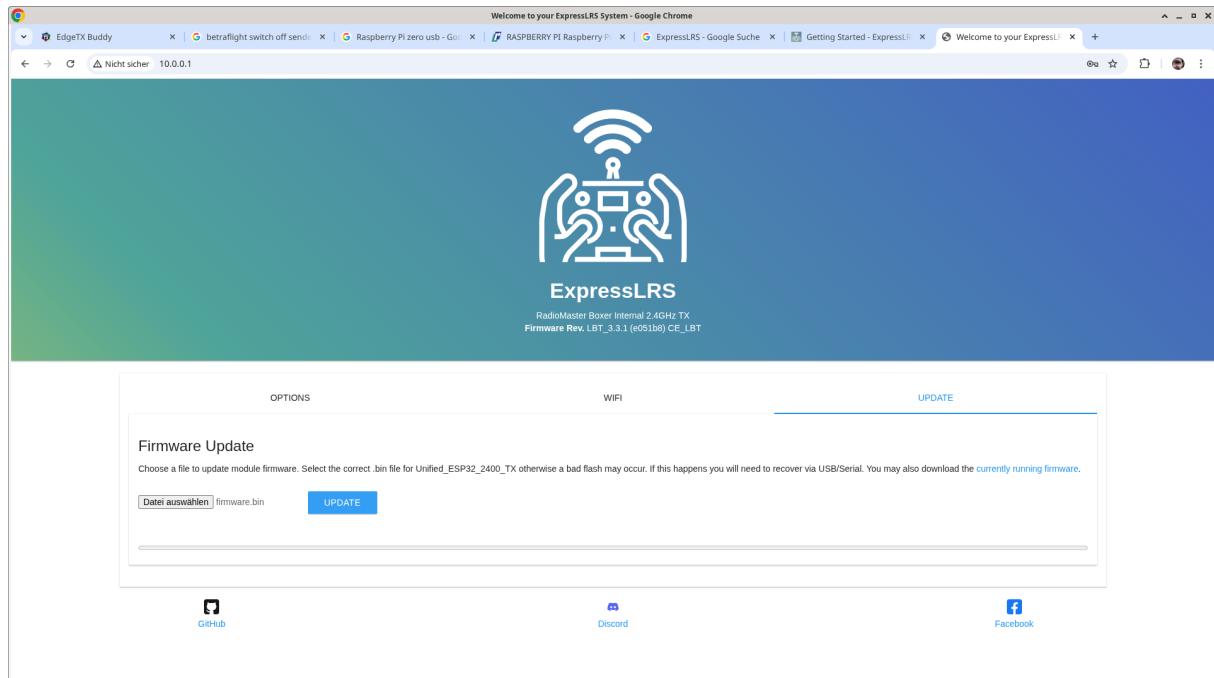


Figure 2.42: ExpressLRS Installation on Sender: Step 4

The installation of ExpressLRS on the receiver is almost identical. Here, too, a firmware file matching the receiver is created using the ExpressLRS Configurator (see Figures 2.43 and 2.44).

Die Installation von ExpressLRS auf dem Empfänger erfolgt nahezu identisch. Auch hier wird mit dem ExpressLRS Configurator eine zur Empfängerhardware passende Firmware-Datei erstellt (siehe Abbildung 2.43 und Abbildung 2.44).

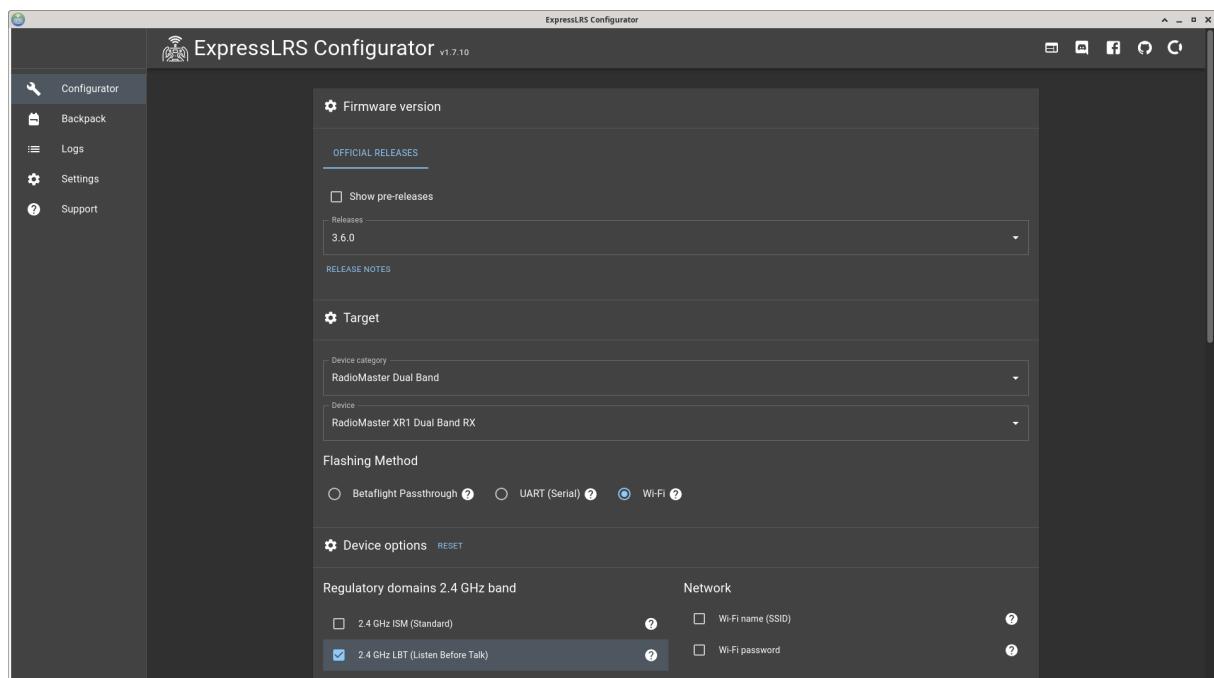


Figure 2.43: ExpressLRS Installation on Receiver: Step 1

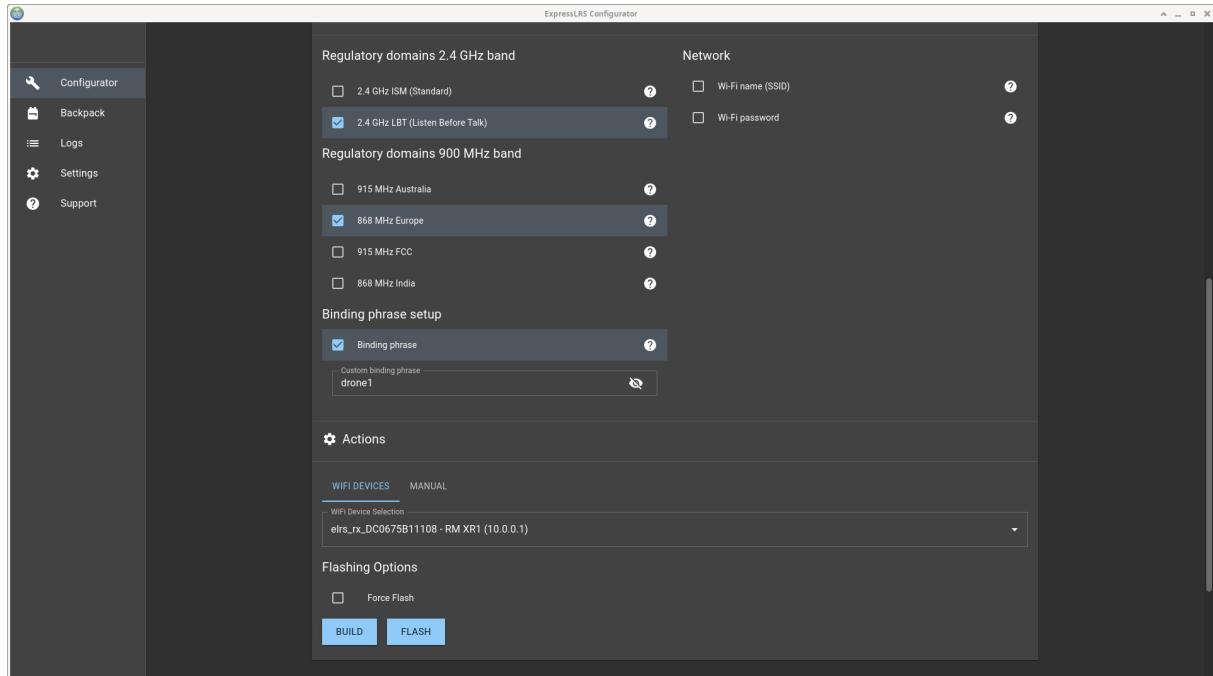


Figure 2.44: ExpressLRS Installation on Receiver: Step 2

Receivers from Radiomaster provide a Wi-Fi network named **ExpressLRS RX** (password: **expresslrs**) after approximately 30 seconds without a connection to a transmitter. The PC must be connected to this network. The receiver module uses the IP address **10.0.0.1**. When this address is opened in a web browser, a simple interface allows, among other things, firmware updates (see Figure 2.45).

Empfänger von Radiomaster stellen nach etwa 30 Sekunden ohne Verbindung zu einer Fernbedienung ein Funknetz mit dem Namen **ExpressLRS RX** (Passwort: **expresslrs**) bereit. Der PC muss mit diesem Netzwerk verbunden werden. Das Empfangsmodul hat die IP-Adresse **10.0.0.1**. Wird diese Adresse über einen Browser aufgerufen, kann über eine einfache Webseite unter anderem die Firmware aktualisiert werden (siehe Abbildung 2.45).

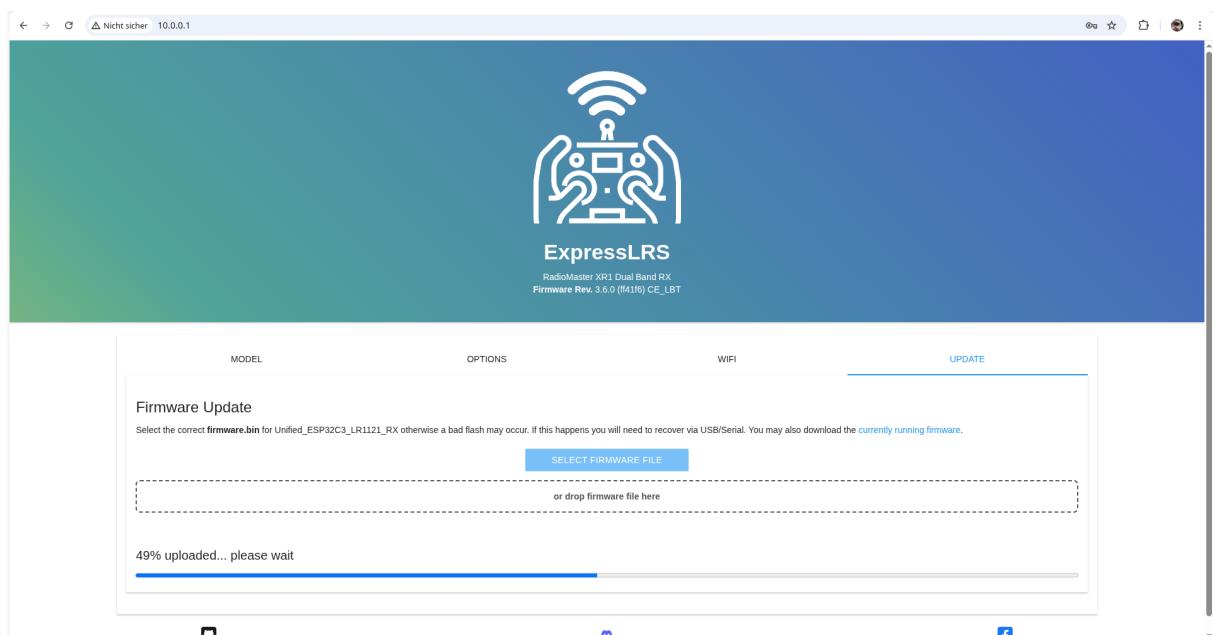


Figure 2.45: ExpressLRS Installation on Receiver: Step 3

3

Important Things before the first Flight

This chapter provides useful advice for both the preparations before the first flight and the checks afterwards.

3.1

Security during Flight

Always keep as much distance as possible from other people during flight. Do not fly over other people's buildings or private property without prior permission.

3.2

Security before Flight

Before each flight, the arming procedure takes place. This means activating the motors in preparation for flight. Although the motors run at a relatively low speed during this process, even at this speed injuries to arms or hands can easily occur. To prevent accidental arming, it is advisable to configure the flight controller so that two switches—ideally not positioned directly next to each other—must be activated simultaneously. A configuration in Betaflight that ensures exactly this is shown in Figure 2.7.

3.3

Secure use of LiPo Batteries

LiPo batteries should never be charged unattended, and they must not be charged while

Wichtige Punkte vor dem ersten Flug

Dieses Kapitel enthält hilfreiche Hinweise für die Vorbereitung vor dem ersten Flug sowie für die Überprüfungen danach.

Sicherheit beim Flug

Während des Fluges ist stets ein möglichst großer Abstand zu anderen Personen einzuhalten. Fremde Gebäude und private Grundstücke dürfen nicht ohne vorherige Zustimmung überflogen werden.

Sicherheit vor dem Flug

Vor dem Flug findet das Arming statt. Dabei handelt es sich um das Aktivieren der Motoren zur Flugvorbereitung. Die Motoren laufen dabei zwar nur mit relativ niedriger Geschwindigkeit, aber selbst bei dieser Drehzahl kann es leicht zu Verletzungen, beispielsweise an Armen und Händen, kommen. Um ein unbeabsichtigtes Arming zu vermeiden, ist es ratsam, den Flugcontroller so zu konfigurieren, dass zum Arming zwei Schalter betätigt werden müssen, die idealerweise nicht direkt nebeneinander liegen. Eine Konfiguration in Betaflight, die genau dies sicherstellt, zeigt Abbildung 2.7.

Sichere Verwendung von LiPo-Akkus

LiPo-Akkus dürfen niemals unbeaufsichtigt geladen werden, insbesondere nicht während man

sleeping. Damaged batteries must no longer be used, especially if they have become swollen due to deep discharge, overcharging, or physical damage. Swollen LiPo batteries pose a significant fire hazard. A swollen battery cannot be repaired and must be properly disposed of. Disposal must not be done through household waste but through designated recycling or collection points provided by the local municipality.

schläft. Beschädigte Akkus dürfen nicht weiterverwendet werden, insbesondere wenn sie durch Tiefentladung, Überladung oder physische Einwirkungen aufgebläht sind. Aufgeblähte LiPo-Akkus stellen ein erhebliches Brandrisiko dar. Ein solcher Akku kann nicht repariert werden und muss ordnungsgemäß entsorgt werden. Die Entsorgung darf nicht über den Hausmüll erfolgen, sondern über die entsprechenden Sammelstellen der örtlichen Gemeinde.

3.4

Loading of Batteries

Batteries (LiPo or Li-Ion) with more than one cell must always be charged in balance mode using the voltage values read via the balancer connector (see Figure 3.1). This prevents individual cells from being overcharged during the charging process and helps to avoid deep discharge of individual cells during later use. The maximum recommended charge rate (see Section 1.5.1) is typically 1 C. If this limit is exceeded, the cells will age more quickly, and in the worst case, a fire may occur. For safety reasons, Li-Ion batteries should only be stored in specially designed fireproof containers.

Laden von Akkus

Akkus (LiPo oder Li-Ion) mit mehr als einer Zelle müssen immer unter Verwendung der über den Balancerstecker ausgelesenen Spannungswerte im Balanciermodus geladen werden (siehe Abbildung 3.1). Dies verhindert eine Überladung einzelner Zellen während des Ladevorgangs und beugt einer Tiefentladung einzelner Zellen im späteren Betrieb vor. Die maximal empfohlene Laderate (siehe Abschnitt 1.5.1) beträgt üblicherweise 1 C. Wird diese Rate überschritten, altern die Zellen schneller, und es kann im schlimmsten Fall zu einem Brand kommen. Zur Sicherheit sollten Li-Ion-Akkus nur in dafür vorgesehenen, feuerfesten Behältnissen gelagert werden.



Figure 3.1: Balanced Loading of a Li-Ion Battery

3.5

Secure Attachment of loose Cables and other Components

If cables or other drone components are not properly secured and can reach the propellers, they eventually will. This can damage either the loose components or the propellers, potentially leading to a crash or costly damage. A common example of components that frequently and unintentionally come into contact with the propellers are the battery balancer connectors (see Figure 3.2) , which are not always easy to replace. Therefore, all components must be securely fastened, and cables must be fixed using screws, cable ties, Velcro straps, or at least electrical tape.

Sichere Befestigung loser Kabel und sonstiger Komponenten

Wenn Kabel oder andere Bauteile der Drohne nicht zuverlässig befestigt sind und die Rotoren erreichen können, werden sie diese früher oder später tatsächlich erreichen. Dabei werden entweder die losen Komponenten oder die Rotoren beschädigt, was zu einem Absturz oder zu teuren Beschädigungen führen kann. Ein typisches Beispiel für Bauteile, die häufig ungewollt mit den Rotoren in Kontakt kommen, sind die Balancerstecker (siehe Abbildung 3.2) der Akkus, deren Austausch nicht immer einfach ist. Daher ist die Befestigung aller Bauteile und die Sicherung aller Kabel mit Schrauben, Kabelbindern, Klettböndern oder zumindest Isolierband zwingend erforderlich.



Figure 3.2: A Balancer Connector got destroyed because it touched a Propeller during Flight

3.6

Use the Appropriate Flight Mode

Den geeigneten Flugmodus verwenden

There are three flight modes: Angle Mode (stabilized), Horizon Mode (semi-stabilized), and Acro Mode (manual).

In Angle Mode, the drone stabilizes itself and remains level without explicit control input from the pilot. This flight mode is the most beginner-friendly and ideal for indoor flights and smooth, slow cinematic shots. Because the maximum tilt angle is limited, spectacular flight maneuvers such as flips or rolls are not possible.

In Horizon Mode, flips and rolls are possible, but the drone automatically stabilizes itself when no control input is applied.

In Acro Mode, which is actually the default mode, there is no automatic stabilization. The drone maintains its current rotational speed until the pilot counteracts it. This mode provides full control on all axes and allows spectacular flight maneuvers at any time. It is the least beginner-friendly mode but ideal for freestyle flying and racing.

Since a drone is set to Acro Mode by default, it is advisable to configure a three-position switch on the transmitter for selecting flight modes so that Angle Mode can be activated automatically when desired. A configuration

Es gibt die Flugmodi Angle Mode (stabilisiert), Horizon Mode (halb-stabilisiert) und Acro Mode (manuell).

Im Angle Mode stabilisiert sich die Drohne selbst und hält sich ohne explizite Steueranweisungen des Piloten automatisch waagrecht. Dieser Flugmodus ist besonders anfängerfreundlich und ideal für Indoor-Flüge sowie für langsame, ruhige Filmaufnahmen. Da die maximale Neigung (Tilt) begrenzt ist, sind spektakuläre Flugmanöver wie Flips oder Rollen nicht möglich.

Im Horizon Mode sind Flips und Rollen möglich, jedoch stabilisiert sich die Drohne automatisch, sobald keine Steuerkommandos mehr gegeben werden.

Im Acro Mode, der eigentlich der Standardmodus ist, erfolgt keine automatische Stabilisierung. Die Drohne hält stets die aktuelle Drehgeschwindigkeit bei, bis der Pilot aktiv gegensteuert. In diesem Modus hat man die volle Kontrolle über alle Achsen, und spektakuläre Flugmanöver sind jederzeit möglich. Dieser Flugmodus ist am wenigsten anfängerfreundlich und ideal für Freestyle-Flüge und Rennen.

Da Drohnen standardmäßig im Acro Mode betrieben werden, ist es sinnvoll, einen Schalter des Senders mit drei möglichen Positionen zur Auswahl des Flugmodus so zu konfigurieren, dass bei Bedarf automatisch der Angle Mode aktiviert werden kann. Eine entsprechende

in Betaflight that enables this is shown in Figure 2.7.

3.7

Reduce the Power of the Video Transmitter or Switch It Off

It is highly recommended to switch off the VTX or significantly reduce its output power during ground or laboratory tests, as it can overheat within a few minutes without the airflow present during flight. Even if it does not overheat, it will still become very hot quickly and draw considerable current from the batteries.

In Betaflight, the VTX can be completely switched off via the CLI:

```
vtx_power = 0
save
```

These commands switch the VTX to the reduced 25 mW power mode:

```
vtx_power = 1
save
```

Another option is to greatly reduce the VTX's output power by switching it to so-called PIT mode. PIT mode lowers the transmit power to approximately 0.1 mW – just enough to receive the signal on the bench or within the same room, while preventing overheating. The following command switches the VTX to PIT mode:

```
set vtx坑模式 = ON
save
```

To deactivate PIT mode again:

```
set vtx坑模式 = OFF
save
```

The PIT mode can also be enabled and disabled on the *Video Transmitter* page in the Betaflight Configurator or via the Betaflight web application. It is useful to assign PIT mode to a switch (AUX channel) on the transmitter. This can be done in the Betaflight Configurator or the Betaflight web application on the *Modes* page using the *VTX Pit Mode* entry.

Additionally, it is possible to operate the VTX in *low-power mode* before arming and after disarming – i.e., during flight preparation and after

Konfiguration in Betaflight, die dies ermöglicht, zeigt Abbildung 2.7.

Die Leistung des Videosenders reduzieren oder diesen abschalten

Es ist sehr empfehlenswert, bei Boden- oder Labortests den VTX abzuschalten oder seine Leistung stark zu reduzieren, da er ohne die Luftkühlung während des Flugs innerhalb weniger Minuten überhitzen kann. Selbst wenn er nicht überhitzt, wird er sehr schnell heiß und zieht viel Strom aus den Akkus.

In Betaflight kann man den VTX über die CLI komplett ausschalten:

Diese Kommandos schalten den VTX in den reduzierten 25-mW-Leistungsbetrieb:

Eine andere Möglichkeit ist, die Leistung des VTX stark zu reduzieren, indem man ihn in den sogenannten PIT-Mode schaltet. Der PIT-Mode reduziert die Sendeleistung auf ungefähr 0,1 mW. Das ist gerade genug, um das Signal auf dem Tisch oder im Raum zu sehen und eine Überhitzung auszuschließen. Das folgende Kommando schaltet den VTX in den PIT-Mode:

Um den PIT-Mode wieder zu deaktivieren:

Der PIT-Mode kann auch auf der Seite *Video Transmitter* im Betaflight Configurator oder über die Betaflight Webanwendung ein- und ausgeschaltet werden. Sinnvoll ist es, dem PIT-Mode einen Schalter (AUX-Kanal) auf der Fernbedienung zuzuweisen. Dies ist im Betaflight Configurator oder in der Betaflight Webanwendung auf der Seite *Modes* über den Eintrag *VTX Pit Mode* möglich.

Zudem ist es möglich, den VTX vor dem Arming und nach dem Disarming im *Low-Power-Modus* zu betreiben, also während der Flug-

the flight. The following command defines that the VTX transmission power will only be increased to the configured level after arming. Before arming and after disarming, the transmission power is reduced to the value defined by `vtx_power=1`.

```
set vtx_low_power_disarm = ON
save
```

vorbereitung und nach dem Flug. Das folgende Kommando definiert, dass die Sendeleistung des VTX erst nach dem Arming auf den eingestellten Wert erhöht wird. Vor dem Arming und nach dem Disarming ist die Sendeleistung auf den Wert von `vtx_power=1` reduziert.

3.8

Practice with a Simulator

Flying a drone is not easy, and crashes are almost inevitable at the beginning, even in Angle Mode. Therefore, it is highly advisable to practice with a simulator. Suitable simulators are available for all operating systems. Some well-known examples include Liftoff, VelociDrone, Uncrashed, FPV Freerider, Quadsim FPV, and Freerider Lite.

The transmitter has a USB interface and should be used as the input device for training. The USB mode must be set to *Joystick*. It is recommended to create a new model in OpenTX for this purpose, for example named *Simulator*. This model should have no active internal or external RF module, since the module consumes a significant amount of power and produces heat. This is unnecessary when using a simulator because no radio signals are transmitted.

Mit einem Simulator üben

Das Fliegen einer Drohne ist nicht einfach, und Abstürze sind insbesondere zu Beginn selbst im Angle Mode kaum zu vermeiden. Daher ist es sehr empfehlenswert, mit einem Simulator zu üben. Geeignete Simulatoren sind für alle Betriebssysteme verfügbar. Einige bekannte Produkte sind Liftoff, VelociDrone, Uncrashed, FPV Freerider, Quadsim FPV und Freerider Lite.

Der Sender verfügt über eine USB-Schnittstelle und sollte als Eingabegerät zum Üben verwendet werden. Als USB-Modus muss *Joystick* ausgewählt werden. Es ist sinnvoll, hierfür in OpenTX ein neues Modell anzulegen, beispielsweise mit dem Namen *Simulator*. Dieses Modell sollte über kein aktives internes oder externes Sendemodul verfügen, da das Sendemodul signifikant Strom verbraucht und Wärme erzeugt. Dies ist beim Einsatz mit einem Simulator unnötig, da keine Funksignale gesendet werden.

3.9

Automate the Pairing

To speed up the pairing process between transmitter and receiver and to prevent accidental pairing with other devices, a binding phrase can be defined in the ExpressLRS firmware (see Section 2.3). This phrase must be identical on both communication partners and cannot be changed without reinstalling (flashing) the firmware.

Die Kopplung automatisieren

Um die Kopplung zwischen Sender und Empfänger zu beschleunigen und ein versehentliches Koppeln mit anderen Geräten zu verhindern, kann in der ExpressLRS-Firmware eine Binding Phrase definiert werden (siehe Abschnitt 2.3). Diese muss bei beiden Kommunikationspartnern identisch sein und kann nur durch eine Neuinstallation (Flashen) der Firmware geändert werden.

3.10

Monitoring telemetry data with suitable tools

For convenient real time monitoring of telemetry data various tools are available. One example is the FM2M ToolBox (Fly Me 2 the Moon) by Robert Janiszewski which can be installed on radio transmitters running EdgeTX. The software aggregates sensor data that is relevant for flight preparation and for monitoring the aircraft during flight. This includes flight mode GPS position satellite count speed altitude battery voltage current draw signal quality and additional sensor data such as barometer and magnetic compass readings. Real time monitoring of telemetry data increases flight safety and enables error detection and analysis.

To use FM2M or similar scripts the flight controller must provide telemetry data via serial interfaces such as UART and must run compatible firmware. FM2M supports the Betaflight and iNAV firmwares and is available for radio transmitters with color or monochrome displays.

Überwachung der Telemetriedaten mit geeigneten Werkzeugen

Zur komfortablen Überwachung der Telemetriedaten in Echtzeit existieren verschiedene Werkzeuge. Ein Beispiel ist die FM2M ToolBox (Fly Me 2 the Moon) von Robert Janiszewski die auf Fernbedienungen mit EdgeTX installiert werden kann. Die Software ermöglicht die Aggregation der Sensordaten die für die Flugvorbereitung und die Überwachung während des Fluges relevant sind. Dazu gehören Flugmodus GPS Position Anzahl der Satelliten Geschwindigkeit Höhe Batteriespannung Stromaufnahme Signalqualität sowie zusätzliche Sensordaten wie Barometer und magnetischer Kompass. Die Echtzeitüberwachung der Telemetriedaten erhöht die Flugsicherheit und ermöglicht Fehlererkennung und Fehleranalyse.

Voraussetzung für die Nutzung von FM2M oder vergleichbarer Skripte ist dass der Flugcontroller Telemetriedaten über serielle Schnittstellen wie UART bereitstellt und kompatible Firmware verwendet. FM2M unterstützt die Firmwares Betaflight und iNAV und ist für Fernbedienungen mit Farb und Schwarz Weiß Display erhältlich.

4

Object Detection

Object detection is one of the most well-known applications of artificial intelligence. Among the best-known open-source solutions enabling object detection are the TensorFlow (Lite) machine learning framework combined with the OpenCV (*Open Computer Vision*) library for image processing and object recognition. An alternative approach is the object detection framework *You Only Look Once* (YOLO).

Object detection with FPV drones can be implemented in two basic ways. Both design concepts have advantages and disadvantages depending on factors such as available budget, hardware, the distance between drone and operator, the number of drones, and personal preferences.

- **Edge Node Concept:** The AI functionality is performed by a computer that flies with the drone. This approach is described in Section 4.1. The additional hardware increases cost, weight, and power consumption, which reduces flight time. Using a higher-capacity battery to compensate further increases the total weight. However, this design offers excellent scalability because the sensor data is processed directly on the drone. Only the results (aggregated data) need to be transmitted during or after the flight, drastically reducing the data volume. This approach follows the edge computing paradigm, which is especially advantageous in multi-drone scenarios where simultaneous data collection would otherwise overload communication channels.

Objekterkennung

Objekterkennung ist eine der bekanntesten KI-Anwendungen. Zu den bekanntesten quelloffenen Softwarelösungen, die Objekterkennung ermöglichen, gehören das Framework TensorFlow (Lite) für maschinelles Lernen in Zusammenarbeit mit der Bibliothek OpenCV (*Open Computer Vision*) zur Bildverarbeitung und Objekterkennung. Eine alternative Lösung ist das Objekterkennungs-Framework *You Only Look Once* (YOLO).

Objekterkennung mit FPV-Drohnen kann auf zwei grundsätzliche Arten implementiert werden. Beide Realisierungskonzepte haben Vorteile und Nachteile, deren Gewichtung unter anderem von den finanziellen Möglichkeiten, den verfügbaren Hardwarekomponenten, der Entfernung zwischen Drohne und Benutzer, der Anzahl der Drohnen und nicht zuletzt den persönlichen Präferenzen abhängt.

- **Realisierungskonzept Edge-Knoten:** Die KI-Funktionalität wird von einem Computer übernommen, der mit der Drohne mitfliegt. Diese Alternative beschreibt Abschnitt 4.1. Die Anschaffung zusätzlicher Hardware verursacht Kosten und erhöht das Gewicht der Drohne. Zudem benötigt die zusätzliche Hardware Strom, was die Flugzeit reduziert. Wird ein Akku mit höherer Kapazität eingesetzt, steigt das Gewicht weiter. Vorteile dieses Konzepts sind die sehr gute Skalierbarkeit, da die Verarbeitung der Sensordaten direkt auf der Drohne erfolgt. Nur die Ergebnisse (aggregierten Daten) müssen während des Flugs oder danach übertragen werden, wodurch die Datenmenge drastisch reduziert wird. Es handelt sich somit um eine Variante des Edge-Computing. Besonders wichtig ist diese Datenreduktion in Szenarien mit mehreren Drohnen, die gleichzeitig fliegen und Daten erfassen.

- **Full Data Transmission Concept:**

The AI processing is performed by a ground-based computer that receives the drone's live video feed. This approach is described in Section 4.2. A simple implementation uses FPV goggles with an AV output and a video capture device. Advantages include the low additional cost (a video grabber costs about 15–20 €) and the fact that the drone's weight remains unchanged. No extra data transmission capacity is required since the drone already transmits its live video feed. However, in multi-drone scenarios, the central AI processing can demand significant computational resources, potentially requiring additional ground-based AI hardware.

- **Realisierungskonzept Übertragung aller Live-Daten:** Die KI-Funktionalität wird von einem Computer am Boden übernommen, der Zugriff auf das Livebild der Drohne hat. Diese Alternative beschreibt Abschnitt 4.2. Eine einfache Implementierungsmöglichkeit bietet eine Videobrille mit AV-Ausgang und ein Videograbber. Vorteile dieses Konzepts sind die geringen zusätzlichen Anschaffungskosten (ein Videograbber kostet etwa 15–20 €) und das unveränderte Gewicht der Drohne. Zusätzliche Übertragungskapazitäten sind nicht erforderlich, da das Livebild ohnehin übertragen wird. In Szenarien mit mehreren Drohnen kann der zentrale Rechenaufwand der Objekterkennung jedoch stark ansteigen und zusätzliche KI-Hardware am Boden erforderlich machen.

4.1

Object Detection by additional Hardware on the Drone

It is possible to integrate the hardware and software required for object detection as additional components directly into the drone, allowing them to fly onboard. Ideally, these components consist of a compact and energy-efficient single-board computer. Table 4.1 shows the different sizes and power requirements of various generations of the Raspberry Pi single-board computer.

Objekterkennung mit zusätzlicher Hardware an der Drohne

Es ist möglich, die zur Objekterkennung nötige Hard- und Software als zusätzliche Komponenten in die Drohne zu integrieren und diese Komponenten mitfliegen zu lassen. Sinnvollerweise handelt es sich dabei um einen platz- und stromsparenden Einplatinencomputer. Tabelle 4.1 zeigt die unterschiedlichen Dimensionen und Strombedarfe verschiedener Generationen des Einplatinencomputers Raspberry Pi.

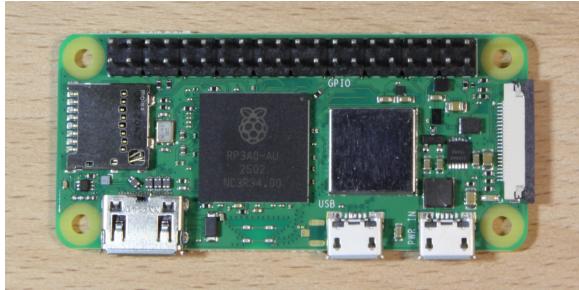
Table 4.1: Generations of the Raspberry Pi Single-Board Computer

Generation	CPU Cores	RAM	Size	Weight	Power Usage (idle)	Power Usage (peak)
Pi Zero W	1 @ 1000 MHz	512 MB	65x30x5 mm	~ 9 g	~ 120 mAh	~ 350 mAh
Pi Zero 2 W	4 @ 1000 MHz	512 MB	65x30x5 mm	~ 10 g	~ 140 mAh	~ 600 mAh
Pi 3 B+	4 @ 1200 MHz	1 GB	85x56x16 mm	~ 50 g	~ 500 mAh	~ 1400 mAh
Pi 4 B	4 @ 1500 MHz	1-8 GB	85x56x16 mm	~ 50 g	~ 600 mAh	~ 1500 mAh
Pi 5	4 @ 2400 MHz	2-16 GB	85x56x16 mm	~ 70 g	~ 700 mAh	~ 2500 mAh

Compared to the approximately credit card-sized Raspberry Pi models, the Raspberry Pi Zero series is significantly smaller, lighter, and consumes less power. The processing performance of the Raspberry Pi 2 W is sufficient to

Im Vergleich zu den etwa scheckkarten-großen Modellreihen sind die Raspberry Pi Zero Modelle deutlich kleiner, leichter und benötigen weniger Strom. Die Prozessorleistung des Raspberry Pi 2 W ist ausreichend, um gängige

run common AI frameworks alongside the operating system. Figure 4.1 shows a Raspberry Pi 2 WH.



KI-Frameworks neben dem Betriebssystem zu betreiben. Abbildung 4.1 zeigt einen Raspberry Pi 2 WH.

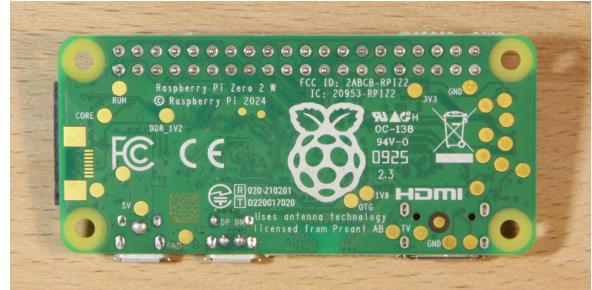


Figure 4.1: Raspberry Pi 2 WH (Front and Back)

Since it is not possible to forward the live FPV camera feed from the flight controller or VTX to the Raspberry Pi, the drone must carry an additional camera that is compatible with the single-board computer. A simple and cost-effective solution is to use the Raspberry Pi Camera Modules v1, v2, or v3, which primarily differ in resolution. All of them weigh only about 3-4 g and have similar dimensions (approximately $25 \times 24 \times 10$ mm). Alternatively, the Raspberry Pi AI Camera (see Section 4.1.1) can be used.

Since a Raspberry Pi, regardless of model, does not have sufficient processing power to perform real-time object detection on video streams, additional hardware is required to accelerate AI applications. In the solution presented here, a Google Coral TPU Accelerator is used for this purpose. The following section describes this and alternative implementation concepts.

Da es nicht möglich ist, über den Flugcontroller oder über den VTX das Livebild der FPV-Kamera auch an den Raspberry Pi weiterzuleiten, muss die Drohne eine weitere, zum Einplatinencomputer kompatible Kamera transportieren. Eine einfache und kostengünstige Lösung sind die Raspberry Pi Camera Module v1, v2 und v3. Diese unterscheiden sich primär in der Auflösung. Sie wiegen alle nur 3-4 g und sind ähnlich groß (ca. $25 \times 24 \times 10$ mm). Alternativ kann auch die Raspberry Pi AI Camera (siehe Abschnitt 4.1.1) verwendet werden.

Da ein Raspberry Pi Einplatinencomputer, egal welcher Baureihe, nicht über die Prozessorleistung verfügt, um Objekterkennung in einem Videostream zu realisieren, ist zusätzliche Hardware zur Beschleunigung der KI-Anwendung nötig. Beim hier vorgestellten Lösungsweg kommt zur Beschleunigung ein Google Coral TPU Accelerator zum Einsatz. Der folgende Abschnitt beschreibt diesen und alternative Implementierungskonzepte.

4.1.1

Comparison of AI Accelerators

This section presents the advantages and disadvantages of several AI accelerators: the Google Coral TPU Accelerator, the Intel Neural Compute Stick 2, the Raspberry Pi AI Camera, and the Raspberry Pi AI Hat. Table 4.2 summarizes key specifications and performance characteristics.

Released in 2019, the Google Coral TPU Accelerator (see Figure 4.2) provides 4 trillion operations per second (Tera-operations per second – TOPS), consumes approximately 500-900 mA

Vergleich von KI-Beschleunigern

Dieser Abschnitt stellt die Vor- und Nachteile der KI-Beschleuniger Google Coral TPU Accelerator, Intel Neural Compute Stick 2, Raspberry Pi AI Camera und Raspberry Pi AI Hat vor. Eine Übersicht relevanter Informationen dazu enthält Tabelle 4.2.

Der 2019 erschienene Google Coral TPU Accelerator (siehe Abbildung 4.2) bietet 4 Billionen Operationen pro Sekunde (Tera-operations per second – TOPS), verbraucht je nach Ge-

depending on performance mode, and measures only $65 \times 30 \times 8$ mm with a weight of around 20 g. An additional USB cable for connection adds 10-15 g. The retail price is about €80-90. A notable drawback is that Google has invested little effort in maintaining or updating drivers and libraries since 2022, making it increasingly difficult to operate the device on modern and future operating systems.

schwindigkeitseinstellung ca. 500-900 mA und ist nur $65 \times 30 \times 8$ mm groß. Das Gewicht dieses KI-Beschleunigers beträgt ca. 20 g. Zusätzlich ist ein USB-Kabel zum Anschluss nötig, das weitere 10-15 g wiegt. Der Kaufpreis liegt bei etwa 80-90 €. Ein Nachteil des Google Coral TPU Accelerator ist, dass der Hersteller seit 2022 keine signifikanten Ressourcen in die Weiterentwicklung der Treiber und Bibliotheken investiert hat und die Entwicklung seitdem vollständig zum Stillstand gekommen ist, was den Betrieb auf aktuellen und zukünftigen Betriebssystemen erschwert.

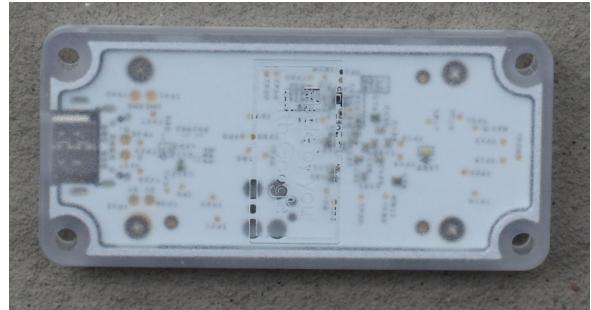


Figure 4.2: Google Coral TPU Accelerator (Front and Back)

An alternative AI accelerator that also connects via USB is the Intel Neural Compute Stick 2 (NCS2), released in 2019 (see Figure 4.3). It offers 4 TOPS, consumes approximately 200-500 mA depending on performance mode, and measures $73 \times 28 \times 14$ mm while weighing about 30 g (without an extension cable). The purchase price is around €100. The NCS2 requires the OpenVINO Toolkit framework, which is free software actively developed by Intel. It converts models from frameworks such as TensorFlow (Lite), PyTorch, ONNX, Keras, and Caffe into a format optimized for Intel hardware and manages execution on compatible devices. It also supports integration with various versions of YOLO. However, starting new projects with the NCS2 is not recommended, as Intel discontinued hardware support with OpenVINO 2022.3 and has not invested significant resources in maintaining the necessary software since 2022, making operation on current and future operating systems increasingly difficult.

Ein alternativer KI-Beschleuniger, der ebenfalls über die USB-Schnittstelle angeschlossen wird, ist der 2019 erschienene Intel Neural Compute Stick 2 – NCS2 (siehe Abbildung 4.3). Dieser bietet ebenfalls 4 TOPS, verbraucht je nach Geschwindigkeitseinstellung ca. 200-500 mA, ist $73 \times 28 \times 14$ mm groß und wiegt (ohne Verlängerungskabel) etwa 30 g. Der Kaufpreis liegt bei rund 100 €. Der NCS2 benötigt das Framework OpenVINO Toolkit, das freie Software ist und von Intel entwickelt wird. Es konvertiert Modelle aus Frameworks wie TensorFlow (Lite), PyTorch, ONNX, Keras und Caffe in ein für Intel-Hardware optimiertes Format und steuert die Ausführung auf kompatibler Hardware. Auch eine Zusammenarbeit mit verschiedenen Versionen von YOLO ist möglich. Neue Projekte mit dem NCS2 zu realisieren ist jedoch nicht empfehlenswert, da Intel die Unterstützung der Hardware mit OpenVINO 2022.3 eingestellt und seit 2022 keine signifikanten Ressourcen mehr in die Weiterentwicklung der zum Betrieb nötigen Software investiert hat, was den Betrieb auf aktuellen und zukünftigen Betriebssystemen erschwert.



Figure 4.3: Intel Neuronal Compute Stick 2 (Front and Back)

A more modern alternative to connecting an AI accelerator via the USB interface is the Raspberry Pi AI Camera, released in 2024. It already includes an integrated AI accelerator whose exact performance in TOPS is not publicly specified, but it is capable of processing up to 30 frames per second for object recognition. The power consumption is approximately 300-600 mA, depending on the operating mode. The purchase price of the Raspberry Pi AI Camera is around €80, which is higher than that of other camera modules. However, it can fully replace an external AI accelerator such as the Google Coral TPU Accelerator. With a weight of only about 6 g, this results in significant weight savings and lower overall system costs.

In addition to the Raspberry Pi AI Camera, which is ideal for use with the Raspberry Pi Zero 2 W single-board computer, the Raspberry Pi Foundation also offers the AI Kit and AI HAT+ in two different performance variants. All three options are HAT (Hardware Attached on Top) expansion boards for the Raspberry Pi 5. The AI Kit can host an M.2 AI accelerator module and comes with a Hailo-8L chip capable of 13 TOPS. The two AI HAT+ variants feature soldered Hailo accelerators with performance levels of 13 TOPS and 26 TOPS, respectively. These HATs cost around €80-120 and offer high AI performance. However, carrying and powering a Raspberry Pi 5 together with such a HAT is not practical in many drone scenarios.

Eine modernere Alternative zum Anschluss eines KI-Beschleunigers über die USB-Schnittstelle ist die 2024 erschienene Raspberry Pi AI Camera. Diese enthält bereits einen integrierten KI-Beschleuniger, dessen Leistungsfähigkeit in TOPS nicht bekannt ist, der jedoch bis zu 30 Bilder pro Sekunde zur Objekterkennung verarbeiten kann. Der Stromverbrauch beträgt je nach Betriebsart etwa 300-600 mA. Der Kaufpreis der Raspberry Pi AI Camera liegt mit rund 80 € zwar höher als der anderer Kamera-Module, sie kann jedoch einen externen KI-Beschleuniger wie den Google Coral TPU Accelerator vollständig ersetzen. Bei einem Gewicht von nur etwa 6 g führt dies zu einer deutlichen Gewichtersparnis und geringeren Gesamtkosten.

Neben der Raspberry Pi AI Camera, die sich ideal für den Einsatz mit dem Raspberry Pi Zero 2 W Einplatinencomputer eignet, bietet die Raspberry Pi Foundation auch die KI-Erweiterungen AI Kit und AI HAT+ in zwei unterschiedlichen Geschwindigkeitsvarianten an. Alle drei Varianten sind Erweiterungsplatinen (sogenannte HATs) für den Raspberry Pi 5. Das AI Kit kann einen KI-Beschleuniger als M.2-Erweiterungsmodul aufnehmen und wird mit einem Hailo-8L-Beschleuniger ausgeliefert, der 13 TOPS erreicht. Die beiden Versionen des AI HAT+ enthalten fest verlötete Hailo-Beschleuniger mit 13 TOPS bzw. 26 TOPS. Diese drei HATs kosten etwa 80-120 € und bieten eine starke KI-Leistung. Der Transport und die Stromversorgung eines Raspberry Pi 5 mit einem solchen HAT ist in vielen Anwendungsszenarien jedoch nicht praktikabel.

Table 4.2: Options for additional Hardware on the Drone that accelerates AI Operations (Edge Computing Scenario)

AI Hardware	Interface	TOPS	Weight	Size [mm]	Power Usage	Support
Google Coral TPU	USB 3.0	4	~ 20 g	65x30x8	500-900 mAh	Discontinued
Intel NCS 2	USB 3.0	4	~ 30 g	73x28x14	200-500 mAh	Discontinued
RPi AI Camera	CSI	?	~ 6 g	25x24x10	300-600 mAh	Active
RPi AI Kit/HAT+	HAT (PCIe)	13	~ 40 g	65x57x6	400-800 mAh	Active

For various reasons, such as short-term hardware availability, it has not yet been possible to integrate the Raspberry Pi AI Camera into a drone project. Therefore, the implementation approach presented in the next section uses the Google Coral TPU Accelerator.

Aus verschiedenen Gründen, wie der kurzfristigen Hardwareverfügbarkeit, war es bislang nicht möglich, die Raspberry Pi AI Camera in ein Drohnenprojekt zu integrieren. Aus diesem Grund verwendet der im nächsten Abschnitt vorgestellte Lösungsweg den Google Coral TPU Accelerator.

4.1.2

Construction and Implementation

Figure 4.4 shows the components of the FPV drone from Figure 1.1, expanded by the components required for onboard image recognition: the Raspberry Pi single-board computer, a Raspberry Pi Camera Module v2, the Google Coral TPU Accelerator, and a BEC (Battery Eliminator Circuit) for converting the battery voltage to 5 V.

The SpeedyBee F405 AIO flight controller used here provides only six UART interfaces, leaving little flexibility, as shown in Figure 4.4. The video transmitter (UART3), GPS module (UART5), and receiver (UART6) each require one UART interface. In addition, one UART (UART1) is reserved for administration via Wi-Fi and USB. Of the remaining interfaces, UART2 supports only simplex mode (receive-only). This leaves a single fully functional (bidirectional) UART interface, which is used to connect the Raspberry Pi Zero 2 W. This connection allows command exchange between the flight controller and the single-board computer.

One disadvantage of the Google Coral TPU Accelerator is that the manufacturer has not invested further resources in driver or library development. The required Python library PyCoral – an extension of TensorFlow Lite enabling integration with the Coral TPU – sup-

Aufbau und Implementierung

Abbildung 4.4 zeigt die Komponenten der FPV-Drohne aus Abbildung 1.1, erweitert um die für die lokale Bilderkennung erforderlichen Komponenten: den Raspberry Pi Einplatinencomputer, ein Raspberry Pi Camera Module v2, den Google Coral TPU Accelerator sowie einen BEC zur Umwandlung der elektrischen Spannung des Akkus in 5 V.

Der verwendete SpeedyBee F405 AIO Flugcontroller verfügt nur über sechs UART-Schnittstellen, was – wie in Abbildung 4.4 gezeigt – nur wenig Spielraum lässt. Videosender (UART3), GPS-Modul (UART5) und Empfänger (UART6) benötigen jeweils eine UART-Schnittstelle. Zusätzlich wird für die Administration über die WLAN- und USB-Schnittstellen (UART1) eine weitere Schnittstelle belegt. Von den verbleibenden Schnittstellen unterstützt UART2 nur den Simplex-Betrieb (nur Lesen). Damit verbleibt lediglich eine einzige vollwertige (bidirektionale) UART-Schnittstelle, über die der Raspberry Pi Zero 2 W angeschlossen wird. Darüber können Kommandos zwischen Flugcontroller und Einplatinencomputer ausgetauscht werden.

Ein Nachteil des Google Coral TPU Accelerator ist, dass der Hersteller keine weiteren Ressourcen in die Weiterentwicklung der Treiber und Bibliotheken investiert. Die benötigte Python-Bibliothek PyCoral, die TensorFlow Lite um die Unterstützung der Google Coral

ports Python 3.9 at most. No further updates are expected. Installation on Raspberry Pi OS (32-bit) Bullseye (Debian 11) is recommended, as Coral packages and PyCoral work reliably there. Newer operating systems based on Debian 12 or 13 are difficult to use with this hardware.

TPU erweitert, unterstützt maximal Python 3.9. Neue Versionen sind nicht mehr zu erwarten. Die Installation auf Raspberry Pi OS (32-bit) Bullseye (Debian 11) ist empfehlenswert, da dort die Coral-Pakete mit PyCoral zuverlässig funktionieren. Neuere Betriebssysteme auf Basis von Debian 12 oder 13 sind hingegen sehr schwierig zu verwenden.

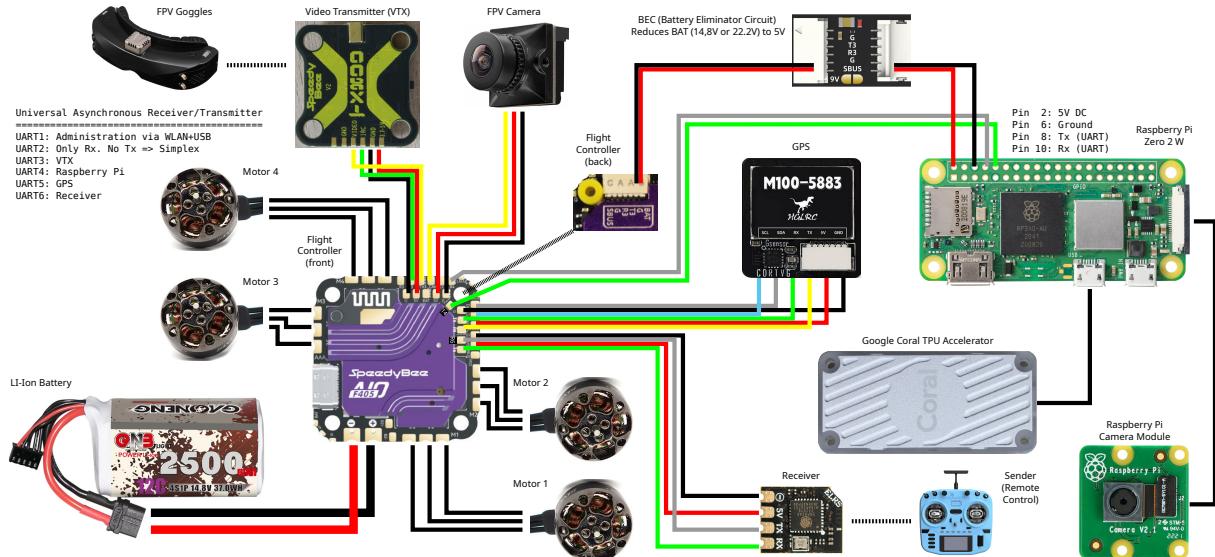


Figure 4.4: Components of a FPV Drone with additional Hardware Components for Object Detection by the Drone itself

4.1.3

Total Weight and Purchase Cost

For building the drone, a frame size between 3.5 and 5 inches is recommended. If the drone is also intended to be flown indoors, a 3.5-inch CineWhoop frame is advisable. The components used in a research project at Frankfurt University of Applied Sciences are listed in Table 4.3.

Gesamtgewicht und Anschaffungskosten

Zum Aufbau der Drohne empfiehlt sich ein Rahmen mit 3,5 bis 5 Zoll. Soll die Drohne auch innerhalb geschlossener Räume geflogen werden können, empfiehlt sich ein 3,5-Zoll-CineWhoop-Rahmen. Die im Rahmen eines Forschungsprojekts an der Frankfurt University of Applied Sciences verwendeten Komponenten sind in Tabelle 4.3 aufgeführt.

Table 4.3: Components of a Drone for Object Detection with a Raspberry Pi Zero 2 W and a Google Coral TPU Accelerator

Product	Category	Weight [g]	Price [€]
SpeedyBee Bee35 PRO 3.5	Frame	130	55
SpeedyBee F405 AIO 40A 3-6S	Flight Controller	14	70
Radiomaster RP1 2,4GHz	Receiver	3	25
Flywoo NIN 1404 V2 3750KV	Motors	40	65
Gemfan D90-5 3.5 Inch Ducted 5-Blade	Propellers	4	3
SpeedyBee TX800 VTX + SMA Adapter	Video Transmitter	10	40
Foxeer Lollipop 4 RHCP SMA	FPV Antenna	5	20
HGLRC M100-5883	GPS and Compass	8	20
Caddx Ratel Pro	FPV Camera	10	55
Li-Ion 2500mAh 4S1P 14,8V 12C	Battery	200	25
Raspberry Pi Zero 2 WH	Single-Board-Computer	10	20
Raspberry Pi Zero case	Case	15	5
32GB microSD card	Storage	1	7
Raspberry Pi Camera Module 8MP v2	Camera	4	17
Adafruit Camera Module case	Case	8	5
Google Coral TPU	AI Hardware	20	85
USB-C cable for Coral TPU	Cable	10	5
Additional cables, straps and panels	Cables + fixing material	10	5
Sum		502	527

The components listed in Table 4.3 refer exclusively to the essential parts of the drone itself. Additional required equipment such as a remote controller (80-160 €), FPV goggles (120-250 €), and a charger (50-100 €) must already be available or purchased separately. The prices of the individual components depend on personal preferences and financial means.

Die in Tabelle 4.3 aufgeführten Komponenten betreffen ausschließlich die notwendigen Bestandteile der Drohne selbst. Weiteres erforderliches Zubehör wie Fernbedienung (80-160 €), FPV-Brille (120-250 €) und Ladegerät (50-100 €) muss bereits vorhanden sein oder zusätzlich beschafft werden. Die Preise der einzelnen Komponenten hängen auch von persönlichen Präferenzen und finanziellen Möglichkeiten ab.

4.2

Object Detection by using the Live Image on the Ground

An easy way to access a live video feed is to use the AV interface provided by some FPV goggles (see Table 1.8). The analog signal can then be digitized with a video grabber device and transmitted to a computer or another device (e.g., a smartphone) for further processing. Figure 4.5 shows the connection of a video grabber to a pair of FPV goggles.

Objekterkennung durch Auswertung des Livebilds am Boden

Eine einfache Möglichkeit, auf das Livebild zugreifen, bietet die gegebenenfalls vorhandene AV-Schnittstelle der Videobrille (siehe Tabelle 1.8). Das Videosignal kann mithilfe eines Videograbbers digitalisiert und zur Verarbeitung an einen Computer oder an ein anderes Gerät (z.B. ein Smartphone) weitergeleitet werden. Abbildung 4.5 zeigt den Anschluss eines Videograbbers an eine Videobrille.



Figure 4.5: Video Grabber Device connected to the A/V Port of an FPV Goggle

There are various tools and commands for accessing video and audio streams. This section presents a selection of useful Linux tools and commands.

Information from the operating system kernel about the video grabber can be obtained, among other things, using the command `dmesg`.

```
$ sudo dmesg -w
[17896.642801] usb 2-2: USB disconnect, device number 91
[17897.851011] usb 2-2: new high-speed USB device number 92 using
    ↳ xhci_hcd
[17897.991360] usb 2-2: New USB device found, idVendor=534d,
    ↳ idProduct=0021, bcdDevice=21.01
[17897.991376] usb 2-2: New USB device strings: Mfr=1, Product=2,
    ↳ SerialNumber=3
[17897.991383] usb 2-2: Product: USB Video
[17897.991389] usb 2-2: Manufacturer: MACROSILICON
[17897.991395] usb 2-2: SerialNumber: 20200909
[17898.055173] usb 2-2: Found UVC 1.00 device USB Video (534d:0021)
[17898.063678] hid-generic 0003:534D:0021.004E: hiddev4,hidraw6: USB
    ↳ HID v1.10 Device [MACROSILICON USB Video] on
    ↳ usb-0000:06:00.3-2/input4
```

The command `lsusb` provides information about the connected USB devices.

Zum Zugriff auf den Video- und Audiostrom existieren verschiedene Programme und hilfreiche Kommandos. Dieser Abschnitt stellt einige nützliche Programme und Kommandos unter Linux vor.

Informationen des Betriebssystemkerns zum Videograbber liefert unter anderem das Kommando `dmesg`.

Das Kommando `lsusb` liefert Informationen zu den angeschlossenen USB-Geräten.

```
$ lsusb
Bus 002 Device 090: ID 534d:0021 MacroSilicon MS210x Video Grabber
    ↳ [EasierCAP]
```

The command `v4l2-ctl` helps in identifying the desired device `/dev/videoX`.

Das Kommando `v4l2-ctl` unterstützt beim Auffinden des gewünschten Geräts `/dev/videoX`.

```
$ v4l2-ctl --list-devices
USB Video: USB Video (usb-0000:06:00.3-2):
    /dev/video6
    /dev/video7
    /dev/media3
```

The video stream can be viewed in a window using tools such as `cheese`, `gst-launch-1.0` (GStreamer), `guvcview`, `ffplay`, `mpv`, and `vlc` (VideoLAN Client).

Den Videostrom in einem Fenster anzeigen kann man unter anderem mit den Werkzeugen `cheese`, `gst-launch-1.0` (GStreamer), `guvcview`, `ffplay`, `mpv` und `vlc` (VideoLAN Client).

```
$ cheese
$ gst-launch-1.0 v4l2src device=/dev/video6 ! videoconvert !
    ↳ autovideosink
$ guvcview -d /dev/video6
$ ffplay -f v4l2 -i /dev/video6
$ mpv /dev/video6
$ vlc v4l2:///dev/video6
```

Most of the mentioned tools can also redirect (save) video and audio streams to local files and convert them into various file formats.

There are also apps for the Android operating system that can display a video stream provided through the USB interface. One example is USB Camera (Pro).

Die meisten der genannten Werkzeuge können auch Video- und Audioströme in lokale Dateien umleiten (speichern) und in verschiedene Dateiformate konvertieren.

Auch für das Betriebssystem Android gibt es Apps, die einen über die USB-Schnittstelle bereitgestellten Videostrom anzeigen können. Ein Beispiel ist USB Camera (Pro).



Figure 4.6: Object Detection using a USB Video Grabber connected to a FPV Goggles

Once the live analog video signal from the FPV goggles is digitized using a USB video grabber and transmitted to a ground-based computing system, a wide range of real-time video processing techniques can be applied. A practical and widely adopted solution for this purpose is the OpenCV (Open Source Computer Vision) library, which provides efficient tools for real-time image acquisition, processing, and analysis using Python.

The digitized video feed is treated as a standard camera input by the operating system, allowing OpenCV to capture frames in real time. This approach makes the system platform-independent and suitable for Linux, macOS, and Windows environments.

After acquiring the live video frames, various computer vision and object detection algorithms can be applied, including but not limited to:

- Region or area-based detection
- Detection of other drones or aerial objects in the sky
- Obstacle detection and tracking
- Motion analysis and situational awareness

Sobald das analoge Live-Videosignal der FPV-Brille mithilfe eines USB-Videograbbers digitalisiert und an ein bodengebundenes Rechensystem übertragen wurde, können vielfältige Verfahren zur Echtzeit-Bildverarbeitung angewendet werden. Eine praxisnahe und weit verbreitete Lösung stellt die Bibliothek OpenCV (Open Source Computer Vision) dar, die effiziente Werkzeuge zur Echtzeit-Bildaufnahme-, -verarbeitung und -analyse unter Python bereitstellt.

Der digitalisierte Videostream wird vom Betriebssystem als gewöhnliche Kameraquelle behandelt, wodurch OpenCV die Einzelbilder in Echtzeit erfassen kann. Dieser Ansatz ist plattformunabhängig und eignet sich für Linux-, macOS- und Windows-Systeme.

Nachdem die Live-Videobilder erfasst wurden, können verschiedene Verfahren der Bildverarbeitung und Objekterkennung angewendet werden. Dazu zählen unter anderem:

- regions- oder flächenbasierte Erkennung
- Erkennung anderer Drohnen oder fliegender Objekte im Luftraum
- Hinderniserkennung und -verfolgung
- Bewegungsanalyse und Situationsbewertung

```

camera = cv2.VideoCapture(CAMERA_INDEX) # Linux and macOS
camera = cv2.VideoCapture(CAMERA_INDEX, cv2.CAP_DSHOW) # Windows

success, frame = camera.read()

```

Figure 4.7: Reading Video Stream from a Video Grabber using OpenCV in Python

In addition to visual information, FPV video streams can contain embedded telemetry data such as altitude, speed, GPS position, battery voltage, and the current flight mode. This telemetry data can be extracted and synchronized with visual detections. This enables advanced applications such as geo-referenced object detection, situation assessment, and flight-state-aware decision making.

Figure 4.7 shows an example of how a video stream from a USB video grabber can be accessed using OpenCV in Python. The device index specifies the camera input as recognized by the operating system. In many cases, index 0 is used when no additional camera devices are connected. On systems with multiple video sources, a different index may be required. On Windows systems, it is recommended to explicitly use the DirectShow backend by specifying `cv2.CAP_DSHOW` to ensure stable access to USB video grabbers.

Figure 4.8 shows a minimal example that demonstrates real-time drone detection using a USB video grabber and OpenCV. In this example, a pretrained YOLO model is used that is capable of detecting drones in aerial scenes. The model was trained on a dedicated drone detection dataset and is publicly available at <https://www.kaggle.com/code/zlatan599/drone-detection/output>. After the live video frames are captured with OpenCV, they are passed to the YOLO model for inference. Detected objects are then visualized using bounding boxes and associated confidence scores.

4.2.1

Total Weight and Purchase Cost

Table 4.4 lists the components of the drone from Table 4.3, excluding the hardware required for

Zusätzlich zu den visuellen Informationen können FPV-Videostreams eingebettete Telemetrydaten enthalten, beispielsweise Höhe, Geschwindigkeit, GPS-Position, Akkusspannung und den aktuellen Flugmodus. Diese Telemetrydaten lassen sich extrahieren und mit visuellen Erkennungen synchronisieren. Dadurch werden weiterführende Anwendungen wie georeferenzierte Objekterkennung, Lagebewertung und flugzustandsabhängige Entscheidungsfindung ermöglicht.

Abbildung 4.7 zeigt ein Beispiel für den Zugriff auf einen Videostream eines USB-Videograbbers mit OpenCV in Python. Der Geräteindex gibt dabei die vom Betriebssystem erkannte Kameraquelle an. In vielen Fällen wird der Index 0 verwendet, sofern keine weiteren Kameras angeschlossen sind. Bei Systemen mit mehreren Videoquellen kann ein anderer Index erforderlich sein. Unter Windows wird empfohlen, explizit das DirectShow-Backend mit `cv2.CAP_DSHOW` zu verwenden, um einen stabilen Zugriff auf USB-Videograbber sicherzustellen.

Abbildung 4.8 zeigt ein minimales Beispiel zur Echtzeit-Erkennung von Drohnen mithilfe eines USB-Videograbbers und OpenCV. In diesem Beispiel wird ein vortrainiertes YOLO-Modell eingesetzt, das für die Erkennung von Drohnen in Luftaufnahmen geeignet ist. Das Modell wurde mit einem speziellen Datensatz zur Dronenerkennung trainiert und ist öffentlich unter <https://www.kaggle.com/code/zlatan599/drone-detection/output> verfügbar. Nachdem die Live-Videoframes mit OpenCV erfasst wurden, werden sie zur Inferenz an das YOLO-Modell übergeben. Erkannte Objekte werden anschließend durch Begrenzungsrahmen und zugehörige Konfidenzwerte visualisiert.

Gesamtgewicht und Anschaffungskosten

Tabelle 4.4 enthält die Komponenten der Drohne aus Tabelle 4.3, reduziert um die Hardwa-

```

import cv2
from ultralytics import YOLO

CAMERA_INDEX = 0           # SET YOUR CAMERA INDEX HERE
MODEL_PATH = "models/final_drone_model.pt"
IMG_SIZE = (640, 480)

# Open camera (DirectShow is REQUIRED on Windows)
# camera = cv2.VideoCapture(CAMERA_INDEX, cv2.CAP_DSHOW)
camera = cv2.VideoCapture(CAMERA_INDEX)
if not camera.isOpened():
    raise RuntimeError("Could not open camera")

camera.set(cv2.CAP_PROP_FRAME_WIDTH, IMG_SIZE[0])
camera.set(cv2.CAP_PROP_FRAME_HEIGHT, IMG_SIZE[1])
model = YOLO(MODEL_PATH)
class_names = model.names

print("YOLO loaded. Press CRL + c to exit.")
while True:
    ret, frame = camera.read()
    if not ret:
        print("Failed to grab frame")
        break

    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = model.predict(frame_rgb, verbose=False)[0]

    if results.boxes is not None:
        boxes = results.boxes.xyxy.cpu().numpy()
        scores = results.boxes.conf.cpu().numpy()
        classes = results.boxes.cls.cpu().numpy().astype(int)

        for box, score, cls in zip(boxes, scores, classes):
            x1, y1, x2, y2 = map(int, box)

            cv2.rectangle(frame, (x1, y1), (x2, y2),
                          (0, 255, 0), 2)

            label = f"{class_names[cls]} {score:.2f}"
            cv2.putText(frame, label,
                        (x1, max(y1 - 10, 0)),
                        cv2.FONT_HERSHEY_SIMPLEX,
                        0.6, (0, 255, 0), 2)

    cv2.imshow("Drone Detection", frame)
    if cv2.waitKey(1) & 0xFF == 27:
        break

camera.release()
cv2.destroyAllWindows()

```

Figure 4.8: Real-time drone detection using a USB video grabber on the ground system

object detection. Specifically, this includes the Raspberry Pi Zero, the camera module, the Google Coral TPU, and the corresponding cables and enclosures. Additional accessories such as the remote control, FPV goggles, and charger must also be available or purchased separately. In addition, a video grabber (15–25 €) and a computer are required.

re, die für die Objekterkennung erforderlich ist. Konkret sind das der Raspberry Pi Zero, das Kameramodul, die Google Coral TPU sowie die entsprechenden Kabel und Gehäuse. Weiteres notwendiges Zubehör wie Fernbedienung, FPV-Brille und Ladegerät muss selbstverständlich ebenfalls bereits vorhanden sein oder zusätzlich beschafft werden. Zusätzlich werden ein Video-grabber (15–25 €) und ein Computer benötigt.

Table 4.4: Components of a Drone for Object Detection by using the Live Image on the Ground

Product	Category	Weight [g]	Price [€]
SpeedyBee Bee35 PRO 3.5	Frame	130	55
SpeedyBee F405 AIO 40A 3-6S	Flight Controller	14	70
Radiomaster RP1 2,4GHz	Receiver	3	25
Flywoo NIN 1404 V2 3750KV	Motors	40	65
Gemfan D90-5 3.5 Inch Ducted 5-Blade	Propellers	4	3
SpeedyBee TX800 VTX + SMA Adapter	Video Transmitter	10	40
Foxeer Lollipop 4 RHCP SMA	FPV Antenna	5	20
HGLRC M100-5883	GPS and Compass	8	20
Caddx Ratel Pro	FPV Camera	10	55
Li-Ion 2500mAh 4S1P 14,8V 12C	Battery	200	25
Additional cables, straps and panels	Cables + fixing material	10	5
Sum		424	378

5

Autopilot

An autopilot system enables an FPV drone to perform autonomous or semi-autonomous tasks such as position hold, altitude hold, waypoint missions, or return-to-home procedures. These functions rely on a combination of hardware components, sensors, and flight-controller firmware that interprets sensor data and executes control commands.

Various firmware platforms provide differing levels of capability and complexity, including Betaflight, iNAV, ArduPilot, and PX4. The following sections outline the requirements, configuration steps, and supported functions for each of these systems.

5.1

Autopilot with Betaflight

Betaflight (see Section 2.1.1) is primarily designed for manual flight (freestyle and racing) and does not provide a full-featured autopilot. However, limited autonomous functions can be implemented, such as angle-stabilized flight, horizon-assisted stabilization, and basic failsafe procedures.

To achieve these functions, the flight controller requires an accelerometer, a gyroscope, and optionally a barometer. GPS support is available only for basic features such as rescue mode. Betaflight does not support waypoint missions or advanced navigation.

Configuration is performed through the Betaflight Configurator or via the Betaflight web application, where stabilization modes, failsafe behavior, and optional rescue functionality can be assigned to AUX channels.

Autopilot

Ein Autopilot-System ermöglicht es einer FPV-Drohne, autonome oder teilautonome Aufgaben wie das Halten der Position, das Halten der Flughöhe, Wegpunktmissionen oder die Rückkehr zum Startpunkt auszuführen. Diese Funktionen basieren auf einer Kombination aus Hardwarekomponenten, Sensoren und Flugcontroller-Firmware, welche die Sensordaten auswertet und entsprechende Steuerbefehle umsetzt.

Verschiedene Firmware-Plattformen bieten hierfür unterschiedliche Fähigkeiten und Komplexitätsstufen, darunter Betaflight, iNAV, ArduPilot und PX4. Die folgenden Abschnitte beschreiben die Anforderungen, Konfigurations schritte und unterstützten Funktionen der jeweiligen Systeme.

Autopilot mit Betaflight

Betaflight (siehe Abschnitt 2.1.1) ist primär für manuelles Fliegen (Freestyle und Rennen) ausgelegt und stellt keinen vollwertigen Autopiloten bereit. Es lassen sich jedoch begrenzte autonome Funktionen realisieren, etwa stabilisierte Flugmodi (Angle, Horizon) oder grundlegende Failsafe-Verfahren.

Für diese Funktionen werden ein Beschleunigungssensor, ein Gyroskop und optional ein Barometer benötigt. GPS-Unterstützung existiert nur in eingeschränkter Form, beispielsweise für den Rescue Mode. Wegpunktmissionen oder fortgeschrittene Navigationsaufgaben werden nicht unterstützt.

Die Konfiguration erfolgt über den Betaflight Configurator oder über die Betaflight Webanwendung, wo Stabilisierung, Failsafe-Verhalten und optional Rettungsfunktionen auf AUX-Kanäle gelegt werden können.

5.2

Autopilot with iNAV

The INAV firmware (see section 2.1.2) implements complete GPS-assisted flight control and is suitable for navigation tasks such as position or altitude hold, return-to-home, and basic autonomous waypoint missions.

To complete these tasks it is necessary that the acceleration sensor and GPS antenna be configured. In addition, a barometer must be available for altitude measurement. A compass (magnetometer) is not mandatory as of INAV version 7.1, but is strongly recommended for better performance.

If a compass is available, it should be correctly aligned and calibrated and, ideally, located at least 10 cm away from the motors and other electrical components. Airspeed sensors are also supported by INAV.

The *INAV Configurator* is sufficient for configuring the autopilot functions. All flight modes (section 5.2.1) and tuning settings, as well as the sensor calibration can be made here. Not all of the modes described below are full flight modes. Some merely change flight behavior characteristics and must be used in combination with other modes.

5.2.1

Navigation Modes in INAV

NAV ALTHOLD is one such mode. It only sets the flight altitude and must be used in combination with a manual flight mode.

ANGLE is recommended here, as **ALTHOLD** is not suitable for overhead maneuvers. In the *Advanced Tuning Tab*, *Max. Alt-hold climb rate* can be used to configure the range within which the pilot can manually vary the altitude up and down.

NAV GCS is another supporting mode. This allows a connected *Ground Control Station (GCS)* to control the position of the

Autopilot mit iNAV

Die Firmware iNAV (siehe Abschnitt 2.1.2) implementiert eine vollständige GPS-unterstützte Flugsteuerung bereit und eignet sich für Navigation, Position Hold, Altitude Hold, Return-to-Home und grundlegende autonome Wegpunktmissionen.

Hierfür ist es erforderlich, dass der Beschleunigungssensor und die GPS-Antenne konfiguriert sind. Außerdem muss ein Barometer für die Höhenmessung vorhanden sein. Ein Kompass (Magnetometer) ist nicht zwingend erforderlich ab INAV-Version 7.1, jedoch dringend empfohlen für bessere Performance.

Insofern ein Kompass vorhanden ist, sollte dieser korrekt ausgerichtet und kalibriert werden sowie idealerweise mindestens 10 cm von den Motoren und anderen elektrischen Komponenten entfernt sein. Auch Fluggeschwindigkeitssensoren werden von INAV unterstützt.

Zur Konfiguration der Autopilotfunktionalitäten genügt der *INAV Configurator*. In diesem können sämtliche Flugmodi (Abschnitt 5.2.1), Tuning-, und Kalibrierungseinstellungen getätigt werden. Nicht alle der folgend erläuterten Modi sind vollwertige Flugmodi. Einige verändern lediglich Eigenschaften des Flugverhaltens und müssen in Kombination mit anderen Modi verwendet werden.

Navigationsflugmodi in INAV

NAV ALTHOLD ist ein solcher Modus. Dieser setzt lediglich die Flughöhe fest und muss in Kombination mit einem manuellen Flugmodus verwendet werden. Hierbei ist **ANGLE** empfohlen, da **ALTHOLD** nicht für Überkopf-Maneuver geeignet ist. Im *Advanced-Tuning-Tab* kann mit *Max. Alt-hold climb rate* der Spielraum konfiguriert werden, mit welchem der Pilot die Höhe manuell nach oben und unten variieren kann.

NAV GCS ist ein weiterer unterstützender Modus. Dieser erlaubt es einer verbundenen *Ground-Control-Station (GCS)* die Position der Drohne zu kontrollieren. Dies

drone. The GCS can be very useful for the *Follow Me* function.

NAV POSHOLD is a standalone mode that allows the drone to hold its current position. The position can be changed by adjusting the throttle, pitch, and roll sticks. As soon as the respective sticks are returned to their starting position, the drone will hold the new position.

MC BRAKING supports the **POSHOLD** mode and is used in conjunction with it. The difference lies in the braking behavior during position adjustments. With simple **POSHOLD**, the drone returns to the location where the sticks were released after completing the braking process. With **MC BRAKING**, the drone brakes more strongly and holds its position at the location where it comes to a stop.

NAV COURSE HOLD describes the drone attempting to maintain its current course. To adjust the speed while in this mode, the pitch stick can be used. The flight controller adjusts the speed proportionally to the position of said pitch stick.

NAV CRUISE represents a combination of **COURSE HOLD** and **ALTHOLD**.

NAV RTH describes a *Return-To-Home (RTH)* action, which can also be configured as a failsafe. However, failsafe should **not** be activated for a controlled RTH. Under certain circumstances, unexpected behavior may occur. When activating RTH, if the drone is more than 10 meters away from the arming location or the configured safe home, it first climbs to a configured altitude and then flies towards home and lands there as soon as it is within a radius of one meter. In the *Advanced Tuning Tab*, the RTH behavior can be extensively modified with many settings.

kann für die *Follow-Me-Funktion* verwendet werden.

NAV POSHOLD ist ein eigenständiger Modus, welcher die Drohne die Position halten lässt. Durch anpassen der Throttle-, Pitch- und Roll-Sticks kann die Position verändert werden. Sobald die jeweiligen Sticks in die Ausgangsposition versetzt werden, hält die Drohne die neue Position.

MC BRAKING unterstützt den **POSHOLD**-Modus und wird zusammen mit diesem verwendet. Der Unterschied liegt im Bremsverhalten bei Positionsanpassungen. Beim einfachen **POSHOLD** kehrt die Drohne nach Abschluss des Bremsvorgangs an den Ort zurück, an dem die Sticks losgelassen wurden. Mit **MC BRAKING** bremst die Drohne stärker und hält die Position an dem Ort an welchem sie zum Stillstand kommt.

NAV COURSE HOLD versucht die Drohne den derzeitigen Kurs zu halten. Die Geschwindigkeit kann in diesem Modus mithilfe des Pitch-Sticks angepasst werden. Der Flugcontroller passt die Geschwindigkeit proportional zur Stellung des Pitch-Sticks an.

NAV CRUISE entspricht einer Kombination aus **COURSE HOLD** und **ALTHOLD**.

NAV RTH ist der *Return-To-Home (RTH)*-Modus, welcher auch als Failsafe konfiguriert werden kann. Failsafe sollte jedoch **nicht** für ein kontrolliertes RTH aktiviert werden. Unter Umständen kann es zu unerwartetem Verhalten kommen. Wird RTH gestartet und die Drohne ist mehr als 10 Meter von der Arming-Location oder dem konfigurierten Safehome entfernt, klettert sie erst auf eine konfigurierte Höhe und fliegt dann in Richtung Home und landet dort, sobald sie dort im Radius von einem Meter ist. Im *Advanced Tuning Tab* kann das RTH-Verhalten mit vielen Einstellungen weitreichend modifiziert werden.

HOME RESET is a complementary mode to RTH. This should be assigned to a remote control switch that returns to the original position when pressed. This makes it possible to overwrite the current safe home location during flight. By default, the last arming location is the safe home, but up to eight different safe homes can be configured manually, and the closest one will be flown to if it is within the maximum distance that such a safe home may be away.

HOME RESET ist ein komplementärer Modus zu RTH . Dieser sollte auf einen Fernbedienungsswitch gelegt werden, welcher nach dem Drücken auf die Ursprungposition zurückkehrt. Dadurch wird es ermöglicht während des Fluges die aktuelle Safehome-Location zu überschreiben. Standardmäßig ist die letzte Arming-Location das Safehome, jedoch können auch bis zu acht unterschiedliche Safehomes manuell konfiguriert werden, wovon dann jeweils das nächste angeflogen wird, wenn sich dieses innerhalb der maximalen Distanz befindet, die ein solches Safehome entfernt sein darf.

NAV WP allows the drone to fly autonomously along a predefined sequence of waypoints. These waypoints can either be defined manually before takeoff, for example via the Mission Control tab in the *INAV Configurator*, or using the **WP PLANNER** mode during manual flight.

NAV WP erlaubt es der Drohne, autonom an einer vordefinierten Sequenz von Wegpunkten entlangzufliegen. Diese Wegpunkte können entweder manuell vor Ablug definiert werden, beispielsweise über den Mission Control Tab im *INAV Configurator* oder mithilfe des Modus **WP PLANNER** während dem manuellen Flug.

WP PLANNER is not a standalone flight mode, but is used to define waypoints during manual flight. It is important that the actual WP mode is not activated, otherwise no waypoints can be saved. In addition, the **MISSION INFO** block must be activated in the OSD. This block displays the information required for the planner. To save a waypoint, the drone must be moved to the desired location and the switch configured for **PLANNER** must be activated. Once the saving process is complete, a corresponding message will be displayed in the OSD. When saving, please note that the altitude is also saved. This is important because a waypoint that has been saved at ground level will be flown to accordingly. To fly the mission configured in this way, simply deactivate **PLANNER** and activate **WP**. Alternatively, the mission can also be saved with *stick commands* (see section 5.2.3).

WP PLANNER ist kein eigenständiger Flugmodus, sondern dient dazu, Wegpunkte während des manuellen Flugs zu definieren. Wichtig ist, dass WP nicht aktiviert sein darf, sonst kann kein Wegpunkt gespeichert werden. Darüber hinaus muss der Block **MISSION INFO** im OSD aktiviert werden. Dieser Block zeigt die für den Planner notwendigen Informationen an. Um einen Wegpunkt zu speichern, muss die Drohne an den gewünschten Ort bewegt und der Switch, auf dem **PLANNER** konfiguriert ist, betätigt werden. Sobald der Speichervorgang abgeschlossen ist, wird eine entsprechende Mitteilung im OSD zu sehen sein. Beim Speichern ist zu beachten, dass die Höhe entsprechend mitgespeichert wird. Dies ist wichtig, da ein Wegpunkt, der ebenerdig gespeichert wurde, entsprechend angeflogen wird. Um die so konfigurierte Mission zu fliegen, genügt es **PLANNER** zu deaktivieren und **WP** zu aktivieren. Alternativ kann die Mission auch mit *Stick-Commands* gespeichert werden (siehe Abschnitt 5.2.3).

5.2.2

Configure Waypoint Missions with INAV

This subsection serves to describe the preparations for a waypoint mission in more detail. Once the mode has been configured using the appropriate switch, there are still a few options to configure in the *Advanced Tuning Tab* under the tabs *Multirotor Navigation Settings*, *General Navigation Settings*, and *Waypoint Navigation Settings*. The effects of these options are explained in the tab itself and will not be repeated here.

After configuring the Advanced Tuning tab, you can now set the waypoints. This is done in the *Mission Control Tab* of the *INAV Configurator*. The waypoints can be set and strung together by simply clicking on the map. This sequence cannot be adjusted individually at a later stage. To change the flight sequence, the properties of the waypoints must be adjusted, or the waypoints themselves must be completely reset. The former is described in the following paragraphs.

Wegpunktmissionen mit INAV konfigurieren

Dieses Unterkapitel dient dazu, die Vorbereitungen einer Wegpunktmission umfassender zu beschreiben. Nachdem der Modus auf einen entsprechenden Switch konfiguriert ist, sind noch einige Optionen im *Advanced Tuning Tab* unter den Reitern *Multirotor Navigaton Settings*, *General Navigation Settings* und *Waypoint Navigation Settings* zu konfigurieren. Was diese Optionen bewirken, wird im Reiter selbst erläutert und hier nicht noch einmal ausgeführt.

Nach der Konfiguration des Advanced Tuning Tabs kommt es nun zum eigentlichen Setzen der Wegpunkte. Dies geschieht im *Mission Control Tab* des *INAV Configurators*. Die Wegpunkte können durch einfaches Klicken auf der Karte gesetzt und aneinander gereiht werden. Diese Reihenfolge kann nicht nachträglich, einzeln angepasst werden. Um die Flugreihenfolge zu verändern müssen die Eigenschaften der Wegpunkte angepasst werden, oder sie müssen komplett neu gesetzt werden. Ersteres wird im Folgenden beschrieben.

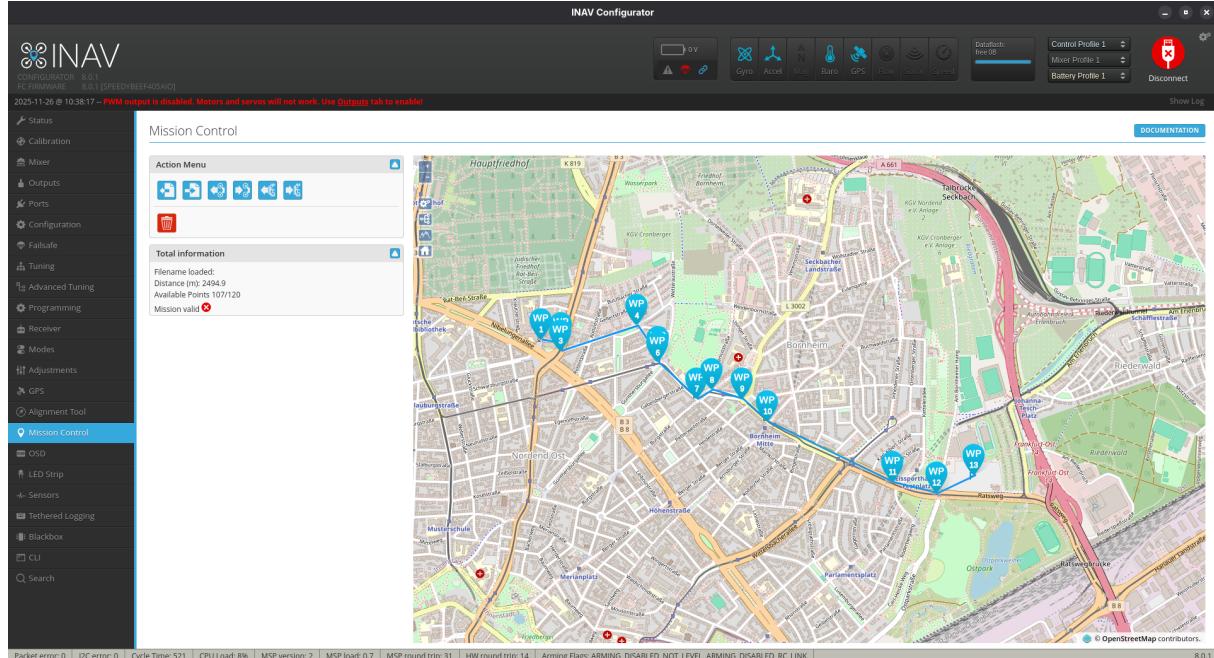


Figure 5.1: Example of a waypoint mission

The editing view of a waypoint is opened by clicking on its symbol on the map. In this view longitude and latitude can be refined using the number field, and it is also possible to configure the *waypoint types* and define *actions*.

Durch erneutes Klicken auf einen jeweiligen Wegpunkt kann die Editier-Ansicht geöffnet werden. In dieser können Längen- und Breitengrade können über das Nummernfeld verfeinert werden und es ist darüber hinaus möglich, den

For waypoint types, you can choose between *Waypoint*, *Position-Hold (PH)*, *Point-of-Interest (POI)* and *Land*. *Waypoint* describes a simple waypoint through which the drone flies.

PH describes a waypoint at which the drone stops for a certain amount of time, which is configured in the *Wait time* field.

POI is a point in the mission at which the orientation of the drone or camera is oriented during flight. If a waypoint is defined as *POI*, the camera will orient itself to this point instead of orienting itself to the path while flying to the next waypoint. *Land* causes the drone to land as soon as the point is reached instead of continuing the mission.

The available actions are **JUMP**, **SET HEAD**, and **RTH**. **JUMP** allows you to jump to a waypoint other than the one with the following number. Enter the ID of the waypoint you want to fly to in **P1** and the number of jump repetitions in **P2**.

SET HEAD can be used to control the alignment of the drone (camera). To do this, enter the corresponding degree value *between 0 and 359* in the **P1** field. As soon as the drone has reached the waypoint, the camera is aligned accordingly for the rest of the flight, unless it is overridden at a later waypoint.

RTH causes the drone to fly back to the configured safe home as soon as it has reached the respective waypoint. This is particularly suitable for the end of missions.

Another important point is the *altitude* at which a waypoint is approached. Here, it must be ensured that the European Unions legal maximum altitude of **120 meters is not exceeded**. In addition, it is also important to ensure that the drone does not fall below the minimum altitude required by the elevation profile of the environment. The *Sea Level Ref* option can be used for this purpose when configuring the altitude.

The *MP Elevation* button can be used to compare the elevation profile for the entire mission with the surrounding area. This prevents the drone from colliding with the ground during the flight between waypoints. To enable this

Wegpunkttypen zu konfigurieren und *Actions* zu definieren.

Beim Wegpunkttypen besteht die Auswahl zwischen *Waypoint*, *Position-Hold (PH)*, *Point-of-Interest (POI)* und *Land*. *Waypoint* beschreibt einen einfachen Wegpunkt, durch den die Drohne hindurch fliegt.

PH beschreibt einen Wegpunkt an dem die Drohne für eine bestimmte Zeit anhält, welche im Feld *Wait time* konfiguriert wird.

POI ist ein Punkt in der Mission an dem sich die Ausrichtung der Drohne oder der Kamera im Flug orientiert. Wird ein Wegpunkt als *POI* definiert, richtet sich die Kamera nach diesem, anstatt sich am Weg auszurichten, während sie zum nächsten Wegpunkt fliegt. *Land* lässt die Drohne landen, sobald der Punkt erreicht wurde, anstatt die Mission fortzuführen.

Bei den Actions hat man **JUMP**, **SET HEAD** und **RTH** zur Verfügung. Mit **Jump** kann zu einem anderen Wegpunkt gesprungen werden, als dem mit der nächstgrößeren Nummer. In **P1** wird die ID des anzufliegenden Wegpunkts eingetragen und in **P2** die Anzahl der Jump-Wiederholungen.

Mit **SET HEAD** kann die Ausrichtung der Drohne gesteuert werden. Dazur wird die entsprechende Gradzahl *zwischen 0 und 359* in das Feld **P1** eingetragen. Sobald die Drohne den Wegpunkt erreicht hat wird die Kamera entsprechend ausgerichtet für den weiteren Flug oder bis es an einem später angeflogenen Wegpunkt überschrieben wird.

RTH lässt die Drohne zum konfigurierten Safehome zurückfliegen, sobald sie am jeweiligen Wegpunkt angekommen ist. Dies eignet sich besonders für das Ende von Missionen.

Ein weiterer wichtiger Punkt ist die jeweilige Höhe, an der ein Wegpunkt angeflogen wird. Hier muss sichergestellt sein, dass die in der EU vorgeschriebene Maximalhöhe von **120 Metern nicht überschritten** wird. Außerdem ist darauf zu achten, dass die Drohne an jedem Wegpunkt die entsprechende Höhe hat, gemessen am Umgebungsprofil. Dazu kann die Option *Sea Level Ref* herangezogen werden, wenn die Höhe konfiguriert wird.

Über den Button *MP Elevation* am kann das Höhenprofil für die Mission, mit dem der Umgebung verglichen werden. Damit wird vermieden, dass die Drohne während des Fluges zwischen Wegpunkten mit dem Boden kollidiert. Um die-

comparison, a form of *take-off location* must be available.

sen Vergleich zu ermöglichen, muss eine Form von *Take-off-Location* vorhanden werden.

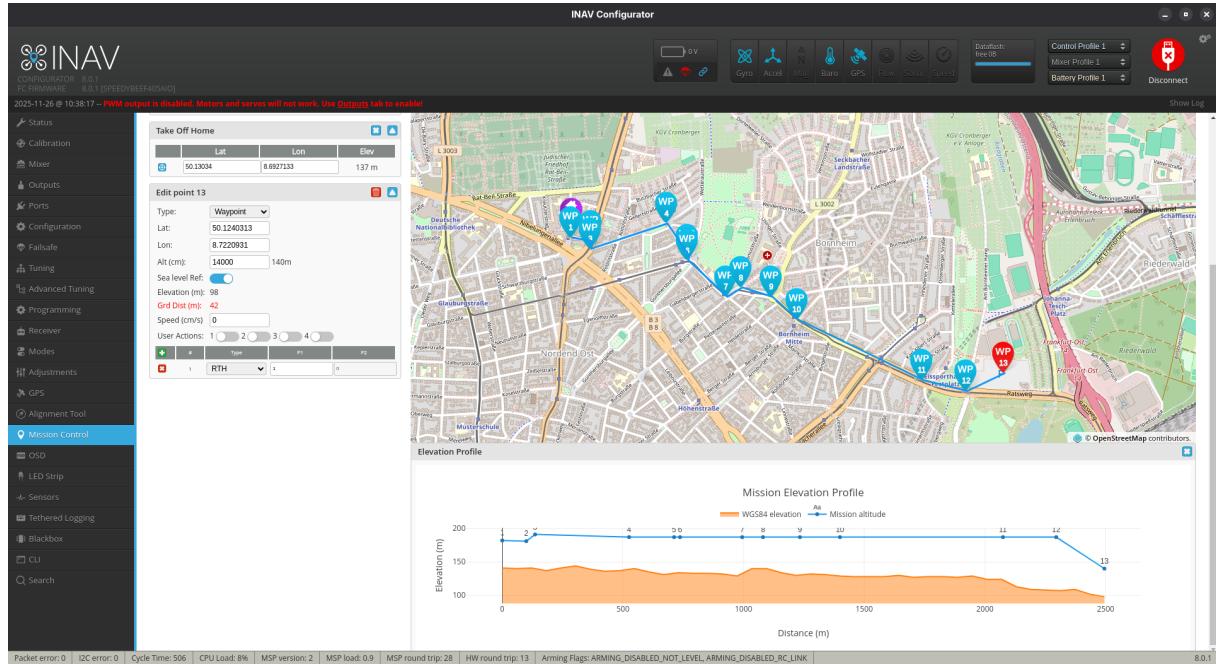


Figure 5.2: Mission Control Example Elevation Profile

Once the waypoint mission has been configured, there are three options for saving it. The first is to save it to a file on the computer, and the second is to save it to the volatile memory of the flight controller (FC). The latter is only suitable if the mission is to be followed immediately, as it will be lost if the system is restarted. If the mission is to be saved permanently, the third option, saving to the EEPROM of the FC, is suitable.

Ist die Wegpunkt-Mission fertig konfiguriert bestehen insgesamt drei Optionen zum Speichern. Die erste ist das Speichern in eine Datei auf dem Computer und die zweite ist das Speichern in den volatilen Speicher des Flugcontrollers (FC). Letzteres ist nur geeignet, wenn die Mission sofort gefolgen werden soll, da sie bei einem Neustart verloren geht. Soll die Mission dauerhaft gespeichert werden, so eignet sich die dritte Option, das Speichern in den EEPROM des FC.

5.2.3

Loading Missions and INAV Stick Commands

If a mission has been written to the EEPROM, it must be loaded into the FC's volatile memory before it can be flown. This can also be done in the *Mission Control Tab* using the Load from EEPROM button. If no computer is available, automatic loading of a mission at boot time can be configured in the *Advanced Tuning Tab* using the *Load Waypoints on Boot* option.

Alternatively, the mission can also be loaded to the volatile memory with a corresponding stick command. Stick commands are various combinations of the two sticks that are nor-

Laden von Missionen und INAV Stickbefehle

Insofern eine Mission in den EEPROM geschrieben wurde, muss sie vor dem Flug in den volatilen Speicher des FC geladen werden. Dies kann ebenfalls im *Mission Control Tab* erfolgen, über den Button Load from EEPROM. Sollte kein Computer vorhanden sein, kann das automatische Laden einer Mission beim Boot im *Advanced Tuning Tab* mit der Option Load Waypoints on Boot konfiguriert werden.

Alternativ kann die Mission auch mit einem entsprechenden Stick-Command geladen werden. Stick-Commands sind verschiedene Kombinationen der beiden Sticks, welche normalerwei-

mally used for flying. They can activate a range of different functions, such as the OSD menu. Figure 5.3 shows the *stick commands* available in INAV.

se zum Fliegen benutzt werden. Sie können eine Reihe verschiedener Funktionen aktivieren, wie beispielsweise das OSD-Menü. Abbildung 5.3 zeigt die *Stick-Commands*, welche es in INAV gibt.

Mode 2 Stick Functions

Profile 1		Trim Acc Left	
Profile 2		Trim Acc Right	
Profile 3		Trim Acc Forwards	
Battery profile 1		Trim Acc Backwards	
Battery profile 2		Save Waypoint Mission	
Battery profile 3		Load Waypoint Mission	
Calibrate Gyro		Unload Waypoint Mission	
Calibrate Acc		Inc WP Mission Index	
Calibrate Compass		Dec WP Mission Index	
Enter OSD Menu (CMS)		Bypass NAV Arm Disable	
In-flight Calibration Controls		Save Setting	
Enter Camera OSD (RuncamDevice)		Exit Camera OSD (RuncamDevice)	
Camera OSD - Confirm		Camera OSD - Navigation	

Figure 5.3: Stick Commands in INAV [Quelle: INAV]

5.3

Autopilot with ArduPilot

ArduPilot (see Section 2.1.3) provides a comprehensive autopilot system for drones. It enables advanced autonomous functions such as waypoint navigation, mission planning, geofencing, object avoidance with additional sensors, automatic takeoff and landing, and precision positioning.

Depending on the firmware variant, ArduPilot requires powerful hardware with sufficient flash memory (see Section 1.2) and sensors including barometer, magnetic compass, accelerometer, gyroscope, and GPS. Additional sensors such as optical flow sensors (Light Detection and Ranging – lidar), or airspeed sensors can be integrated.

Autopilot mit ArduPilot

ArduPilot (siehe Abschnitt 2.1.3) bietet ein umfassendes Autopilotensystem für Drohnen, Flächenflugzeuge, Hubschrauber und andere Plattformen. Es unterstützt fortgeschrittene autonome Funktionen wie Wegpunktnavigation, Missionsplanung, Geofencing, Hindernisvermeidung mit zusätzlichen Sensoren, automatischen Start und automatische Landung sowie präzise Positionsbestimmung.

nach Firmware-Variante benötigt ArduPilot leistungsfähige Hardware mit ausreichend Flash-Speicher (siehe Abschnitt 1.2) sowie Sensoren wie Barometer, magnetischem Kompass, Beschleunigungssensor, Gyroskop und GPS. Zusätzliche Sensoren wie optische Flusssensoren (Light Detection and Ranging – Lidar) oder Fluggeschwindigkeitssensoren können integriert werden.

The Mission Planner software is used for configuration, calibration, and mission definition.

Konfiguration, Kalibrierung und Missionsplanung erfolgen über die Software Mission Planner.

5.4

Autopilot with PX4

PX4 is an open-source autopilot platform similar to ArduPilot, designed for research, industry applications, and autonomous robotics. It supports advanced mission planning, companion computers (e.g., Raspberry Pi Compute Module 4), and complex multi-sensor navigation.

PX4 requires hardware such as Pixhawk controllers or compatible boards with powerful processors and extensive sensor capability. Configuration and mission planning are performed using the QGroundControl software.

Due to its modular design and focus on extensibility, PX4 is frequently used in industrial environments, although some FPV-specific features are less developed compared to Betaflight or iNAV.

Autopilot mit PX4

PX4 ist eine Open-Source-Autopilotplattform, die ähnlich wie ArduPilot auf Forschungs-, Industrie- und Robotikanwendungen ausgerichtet ist. Die Firmware unterstützt fortgeschrittenen Missionsplanung, Companion-Computer (z.B. Raspberry Pi Compute Module 4), und komplexe multisensorbasierte Navigation.

PX4 erfordert Hardware wie Pixhawk-Controller oder kompatible Boards mit leistungsfähigen Prozessoren und umfangreicher Sensorunterstützung. Die Konfiguration und Missionsplanung erfolgt über die Software QGroundControl.

Aufgrund seiner modularen Architektur und Erweiterbarkeit wird PX4 häufig in industriellen Projekten eingesetzt, auch wenn FPV-spezifische Funktionen teilweise weniger ausgeprägt sind als bei Betaflight oder iNAV.

6

Follow-Me Function

The Follow-Me function allows an FPV drone to track and follow a moving target, typically a person carrying a GPS device, a smartphone, or a companion transmitter. This capability is useful for recording moving subjects, supporting inspection tasks, or enabling semi-autonomous flight operations.

Depending on the firmware, the Follow-Me function may rely on external GPS data, telemetry links, onboard sensors, or a combination of these. The following sections describe how Follow-Me is implemented in Betaflight, iNAV, ArduPilot, and provide an outlook on its implementation in PX4.

6.1

Follow-Me with Betaflight

Betaflight does not provide a Follow-Me mode, as the firmware is primarily optimized for manual flight (freestyle and racing). Only basic GPS-assisted functions such as rescue mode are available. Follow-Me operation is not supported because Betaflight lacks autonomous navigation, positional tracking, and target-following logic. Implementing Follow-Me functionality would require a transition to iNAV or ArduPilot.

6.2

Follow-Me with iNAV

iNAV provides a dedicated Follow-Me mode that uses the GPS position of an external device, typically the radio transmitter equipped with an

Follow-Me-Funktion

Die Follow-Me-Funktion ermöglicht es einer FPV-Drohne, einem sich bewegenden Ziel zu folgen, typischerweise einer Person, die ein GPS-Gerät, ein Smartphone oder einen zusätzlichen Sender mit sich führt. Diese Fähigkeit wird unter anderem zur Aufnahme bewegter Szenen, zur Unterstützung von Inspektionsaufgaben oder für teilautonome Fluganwendungen genutzt.

Je nach Firmware basiert die Follow-Me-Funktion auf externen GPS-Daten, Telemetrie-verbindungen, an Bord befindlichen Sensoren oder einer Kombination dieser Quellen. Die folgenden Abschnitte beschreiben die Umsetzung der Follow-Me-Funktion in Betaflight, iNAV, ArduPilot sowie einen Ausblick auf die Implementierung in PX4.

Follow-Me mit Betaflight

Betaflight bietet keinen Follow-Me-Modus, da die Firmware primär für manuelles Fliegen (Freestyle und Rennen) ausgelegt ist. Lediglich grundlegende GPS-gestützte Funktionen wie der Rescue Mode sind verfügbar. Ein Follow-Me-Betrieb wird nicht unterstützt, da Betaflight keine autonome Navigation, keine Positionsverfolgung und keine Zielverfolgslogik bereitstellt. Für den Einsatz der Follow-Me-Funktion wird daher ein Wechsel zu iNAV oder ArduPilot erforderlich.

Follow-Me mit iNAV

iNAV stellt einen eigenen Follow-Me-Modus bereit, der die GPS-Position eines externen Geräts verwendet, üblicherweise die Fernbedienung mit

external GPS module or a compatible telemetry system.

Follow-Me in iNAV requires a flight controller with GPS, barometer, magnetic compass, and stable GPS reception. The external device transmits its position to the flight controller via a telemetry protocol such as MAVLink, MSP, or a proprietary transmitter protocol.

Configuration is performed in the iNAV Configurator by enabling the Follow-Me mode, assigning it to an AUX channel, and configuring the GPS update rate and safe distances. iNAV supports various follow patterns, including direct pursuit and offset tracking.

einem angeschlossenen GPS-Modul oder einem kompatiblen Telemetriesystem. Für Follow-Me in iNAV werden ein Flugcontroller mit GPS, Barometer, magnetischen Kompass sowie eine stabile GPS-Empfangsqualität benötigt.

Das externe Gerät übermittelt seine Position über ein Telemetrieprotokoll wie MAVLink, MSP oder ein proprietäres Fernsteuerungsprotokoll an den Flugcontroller.

Die Konfiguration erfolgt im iNAV Configurator durch Aktivieren des Follow-Me-Modus, das Zuweisen eines AUX-Kanals sowie die Einstellung der GPS-Aktualisierungsrate und der Sicherheitsabstände. iNAV unterstützt unterschiedliche Verfolgungsprofile, darunter direkte Verfolgung und seitlich versetztes Tracking.

6.3

Follow-Me with ArduPilot

ArduPilot provides a comprehensive Follow-Me implementation that supports external GPS devices, companion computers (e.g., Raspberry Pi Zero, Raspberry Pi Compute Module 4/5, NVidia TX1/TX2), and smartphone applications via MAVLink. Follow-Me requires a GPS-equipped external device. The drone must include a barometer, magnetic compass, accelerometer, gyroscope, and GPS module.

Follow-Me is enabled in Mission Planner by selecting the Follow-Me feature and configuring parameters such as follow distance, altitude offset, and update frequency.

ArduPilot supports advanced behaviors including smooth pursuit, orbiting around the target, and terrain-following when additional sensors are available. Its implementation is suitable for professional and research applications.

Follow-Me mit ArduPilot

ArduPilot bietet eine umfassende Follow-Me-Implementierung, die externe GPS-Geräte, Companion-Computer (z.B. Raspberry Pi Zero, Raspberry Pi Compute Module 4/5, NVidia TX1/TX2) und Smartphone-Anwendungen über MAVLink unterstützt. Für Follow-Me wird ein externes GPS-Gerät benötigt. Die Drohne muss über Barometer, magnetischen Kompass, Beschleunigungssensor, Gyroskop und ein GPS-Modul verfügen.

Die Aktivierung erfolgt im Mission Planner über die Follow-Me-Funktion, wobei Parameter wie Verfolgungsabstand, Höhenversatz und Aktualisierungsrate konfiguriert werden.

ArduPilot unterstützt fortgeschrittene Verfolgungsverfahren, darunter sanfte Annäherung, Umkreisung des Ziels und Geländefolgeflug bei Verwendung zusätzlicher Sensoren. Die Umsetzung eignet sich für professionelle und forschungsorientierte Anwendungen.

6.4

Follow-Me with PX4

PX4 provides a Follow-Me feature primarily through the software QGroundControl and companion-computer applications. The function relies on the transmission of position

Follow-Me mit PX4

PX4 stellt eine Follow-Me-Funktion bereit, die hauptsächlich über die Software QGroundControl und Companion-Computer-Anwendungen implementiert wird. Die Funktion basiert auf

data from a smartphone or companion system (e.g., Raspberry Pi Compute Module 4) using MAVLink. PX4 requires a GPS-equipped flight controller with barometer, and magnetic compass.

der Übertragung von Positionsdaten eines Smartphones oder eines Companion-Systems (z.B. Raspberry Pi Compute Module 4) über MAVLink. Für den Follow-Me-Betrieb benötigt PX4 einen GPS-fähigen Flugcontroller mit Barometer und magnetischen Kompass.

7

Drop Mechanisms for FPV Drones

Drop mechanisms enable an FPV drone to release a small payload on command. They are used in research projects, training environments, competition scenarios, and lightweight delivery tasks. The primary purpose of such mechanisms is to hold a payload securely and release it in a controlled manner based on a command from the flight controller.

The most common systems include servo-based and electromagnetic mechanisms. Spring-loaded mechanisms and clamp-based grippers are used less frequently in practice and are therefore not discussed further in this document. Regardless of the underlying mechanism, activation is typically carried out via an AUX channel on the radio transmitter and mapped to an appropriate output on the flight controller.

7.1

Servo-based release mechanisms

Servo-based release mechanisms are one of the most commonly used solutions on FPV drones. A micro servo (see Figure 7.1) actuates a lever, pin, or hook that secures the payload. When triggered by a signal from the flight controller, the servo moves and releases the payload. This method is simple to implement, cost-effective, and offers precise control.

Drop-Mechanismen für FPV-Drohnen

Drop-Mechanismen ermöglichen es einer FPV-Drohne, eine kleine Nutzlast auf Kommando kontrolliert abzuwerfen. Sie kommen in Forschungsprojekten, Trainingsumgebungen, Wettbewerbsdisziplinen sowie bei leichten Transportaufgaben zum Einsatz. Die zentrale Funktion solcher Mechanismen besteht darin, eine Nutzlast zuverlässig zu halten und sie anhand eines Steuersignals des Flugcontrollers kontrolliert freizugeben.

Zu den am häufigsten eingesetzten Lösungen zählen servo-basierte und elektromagnetische Systeme. Federbasierte Mechanismen sowie klammerbasierte Greifer werden in der Praxis deutlich seltener verwendet und daher in diesem Dokument nicht weiter betrachtet. Unabhängig vom zugrunde liegenden Mechanismus erfolgt die Aktivierung in der Regel über einen AUX-Kanal der Fernbedienung, der einem geeigneten Ausgang des Flugcontrollers zugeordnet wird.

Servo-basierte Abwurfmechanismen

Servo-basierte Abwurfmechanismen sind eine häufig eingesetzte Lösung bei FPV-Drohnen. Ein Micro-Servo (siehe Abbildung 7.1) bewegt hierbei einen Hebel, Stab oder Haken, der die Nutzlast hält. Durch ein Signal des Flugcontrollers wird der Servo bewegt und gibt die Nutzlast frei. Diese Methode ist einfach umzusetzen, kostengünstig und präzise steuerbar.

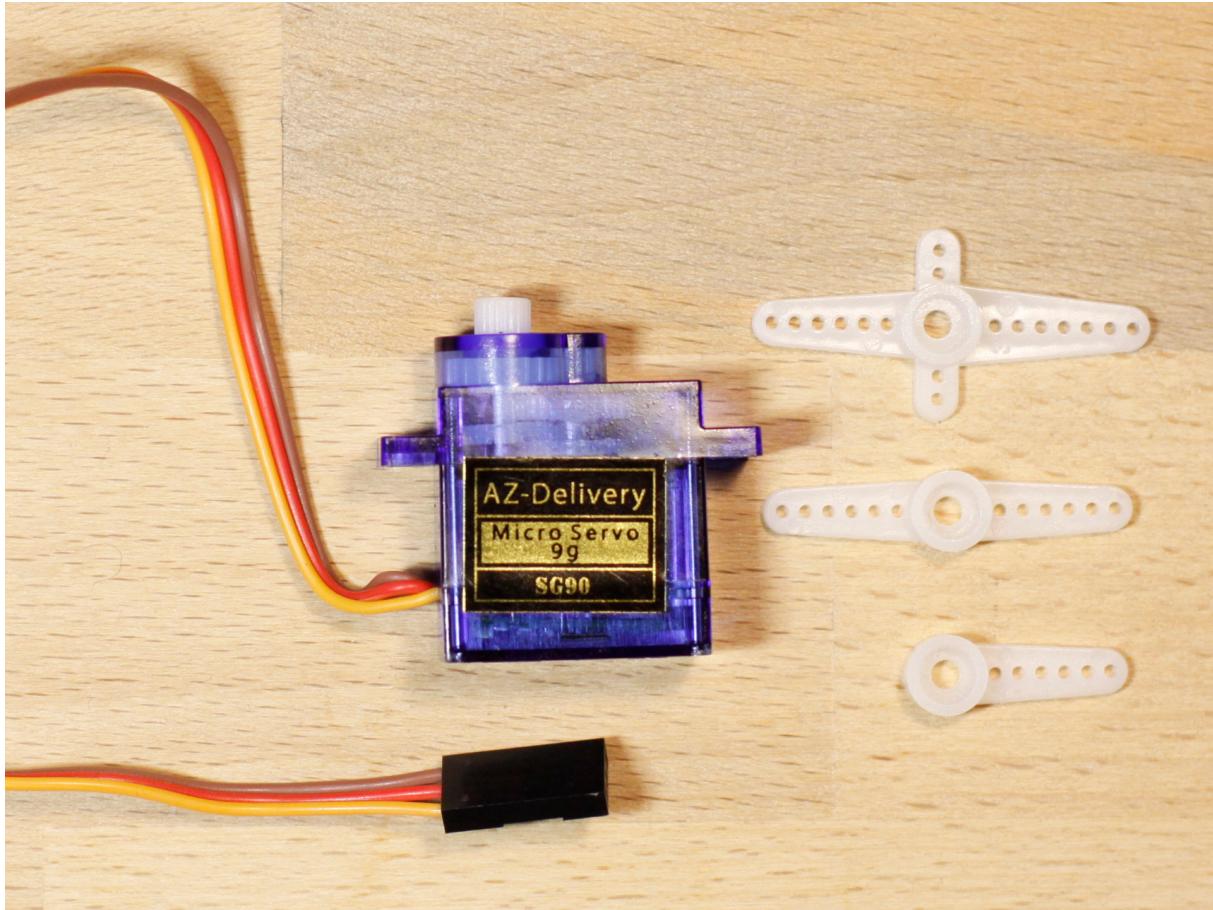


Figure 7.1: Micro-Servo)

7.2

Electromagnetic mechanisms

In electromagnetic mechanisms, a small electromagnet holds a metallic loop or plate. When the current is interrupted, the payload is released. Advantages include the minimal number of moving parts and high operational reliability. A disadvantage is the comparatively higher power consumption.

Elektromagnetische Mechanismen

Bei elektromagnetischen Mechanismen hält ein kleiner Elektromagnet eine metallische Öse oder Platte. Wird der Stromfluss abgeschaltet, fällt die Nutzlast ab. Vorteile sind die geringe Zahl beweglicher Teile und die hohe Zuverlässigkeit. Ein Nachteil ist der im Vergleich zu anderen Lösungen höhere Strombedarf.

7.3

Control and Integration

The activation of a drop mechanism is typically performed through an AUX channel on the radio transmitter. In Betaflight, the AUX channel is assigned to a servo or user channel (USER1/2) and controlled through a PWM-capable output. In ArduPilot, dedicated functions such as GRIPPER or SERVOx_FUNCTION are available, allowing both manual operation and

Ansteuerung und Integration

Die Steuerung eines Drop-Mechanismus erfolgt in der Regel über einen AUX-Kanal der Fernbedienung. In Betaflight wird der AUX-Kanal einem Servo- oder Nutzerkanal (USER1/2) zugeordnet und über einen PWM-fähigen Ausgang angesteuert. In ArduPilot stehen dedizierte Funktionen wie GRIPPER oder SERVOx_FUNCTION zur Verfügung, die auch au-

fully autonomous release actions during mission execution.

tonome Abwürfe im Rahmen von Missionen ermöglichen.

8

Stereo Camera Setup with a Raspberry Pi 5

Stereo cameras are a well established tool for depth perception. The main advantage compared to a simple camera setup is that no model needs to be trained to enable depth perception.

All that is required are two identical cameras and a computer that performs the necessary computations using OpenCV. Alternatively a stereo camera module can be used that is specifically designed for such use cases. These modules are usually more compact and easier to mount and connect. However they incur additional costs.

The following guide is based on a Raspberry Pi 5 with 8 GB of RAM and two camera modules of the model RB-CAMERA-JT-V2-120. This choice was made because only the Raspberry Pi 5 supports the parallel operation of two camera modules. Previous models would require an adapter that does not allow parallel operation of both cameras.

The cameras are mounted next to each other and connected using two ribbon cables. The cameras should be mounted as parallel as possible. Since perfect alignment is not feasible it is corrected later by software.

Such a setup is only suitable for larger drones that can carry batteries with high capacity. On smaller drones the Raspberry Pi 5 would significantly reduce the flight time. For smaller drones a simple camera setup combined with a model should be used.

Stereokameras mit einem Raspberry Pi 5

Stereokameras sind ein etabliertes Werkzeug zur Durchführung von Tiefenerkennung. Der Vorteil gegenüber einem einfachen Kamerasetup besteht darin dass kein Modell trainiert werden muss um die Tiefenerkennung zu realisieren.

Alles was dafür benötigt wird sind zwei bau gleiche Kameras und ein Computer der die notwendigen Berechnungen mithilfe von OpenCV durchführt. Alternativ kann auch ein Stereokameramodul verwendet werden das speziell für solche Anwendungsfälle konzipiert ist. Diese Module sind in der Regel platzsparender und einfacher zu montieren und anzuschließen. Allerdings verursachen sie zusätzliche Kosten.

Die folgende Anleitung basiert auf einem Raspberry Pi 5 mit 8 GB Arbeitsspeicher und zwei Kameramodulen des Modells RB-CAMERA-JT-V2-120. Diese Wahl wurde getroffen weil nur der Raspberry Pi 5 die parallele Nutzung von zwei Kameramodulen unterstützt. Vorangegangene Modelle würden einen Adapter benötigen der keine parallele Nutzung beider Kameras zulässt.

Die Kameras werden nebeneinander montiert und mit zwei Minikabeln angeschlossen. Die Kameras sollten möglichst parallel ausgerichtet werden. Da eine perfekte Ausrichtung jedoch nicht möglich ist wird diese später softwareseitig korrigiert.

Ein solches Setup ist jedoch nur für größere Drohnen geeignet die Batterien mit hoher Kapazität tragen können. Bei kleineren Drohnen würde das Raspberry Pi 5 die Flugzeit sehr stark reduzieren. Für kleinere Drohnen sollte daher mit einem einfachen Kamerasetup in Kombination mit einem Modell gearbeitet werden.

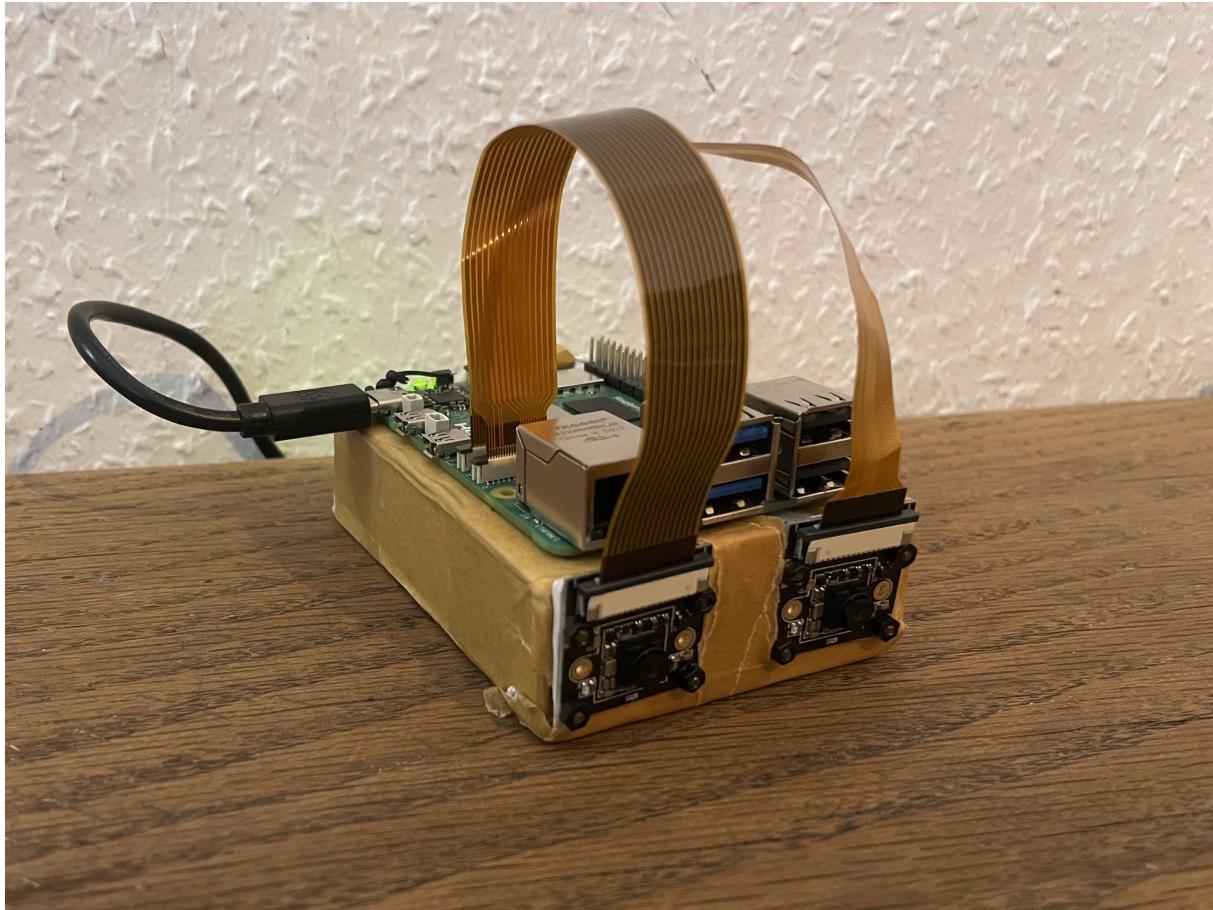


Figure 8.1: Stereo Vision Contraption

8.1

Kamerakalibrierung

The need for camera calibration arises from the construction shown in Figure 8.1 because the two cameras are not mounted perfectly parallel. In theory it would be possible to iteratively adjust the construction until this condition is met. However it is highly unlikely that the lenses of both cameras are exactly identical.

This is particularly relevant for Raspberry Pi cameras because they are so called pinhole cameras. Such cameras often exhibit pronounced radial distortion which causes straight lines to appear curved. Another relevant type of distortion is tangential distortion. It occurs when the camera lens is not perfectly parallel to the observed surface which is especially difficult to achieve when using two cameras.

Camera Calibration

Die Notwendigkeit der Kamerakalibrierung ergibt sich aus der Konstruktion in Abbildung 8.1 da die beiden Kameras nicht exakt parallel montiert sind. Theoretisch wäre es möglich diese Konstruktion iterativ anzupassen bis dies der Fall ist. Allerdings ist es äußerst unwahrscheinlich dass die Linsen der beiden Kameras exakt identisch sind.

Dies ist insbesondere für Raspberry Pi Kameras relevant da es sich hierbei um sogenannte Pinhole Kameras handelt. Solche Kameras weisen häufig eine ausgeprägte radiale Verzerrung auf wodurch gerade Linien gekrümmt erscheinen. Eine weitere relevante Verzerrungsform ist die Tangentialverzerrung. Sie entsteht wenn die Kameralinse nicht vollständig parallel zur aufgenommenen Oberfläche ausgerichtet ist was insbesondere bei der Verwendung von zwei Kameras sehr schwierig zu realisieren ist.

This means that calibration is required regardless of the camera modules used including stereo camera modules. First both cameras are calibrated individually and are then jointly subjected to stereo calibration.

During calibration the so called distortion coefficients are determined. These are denoted by k_1 , k_2 and k_3 for radial distortion and by p_1 and p_2 for tangential distortion.

Das bedeutet dass unabhängig von den verwendeten Kameramodulen einschließlich Stereomodulen eine Kalibrierung erforderlich ist. Dabei werden zunächst beide Kameras einzeln kalibriert und anschließend gemeinsam einer Stereokalibrierung unterzogen.

Bei der Kalibrierung werden die sogenannten Verzerrungskoeffizienten bestimmt. Diese werden mit k_1 , k_2 und k_3 für die radiale Verzerrung sowie mit p_1 und p_2 für die tangentiale Verzerrung bezeichnet.

$$dist = (k_1, k_2, k_3, p_1, p_2)$$

In addition intrinsic and extrinsic camera parameters are computed. The intrinsic parameters describe the internal camera settings represented by the matrix K . These parameters are independent of the observed scene and can be reused after the initial computation as long as the same settings are used. The parameters include the focal lengths (f_x, f_y) and the optical center (c_x, c_y) each expressed in pixel units.

Darüber hinaus werden intrinsische und extrinsische Kameraparameter berechnet. Die intrinsischen Parameter beschreiben die internen Kameraeinstellungen und werden durch die Matrix K repräsentiert. Sie sind unabhängig von der betrachteten Szenerie und können nach der erstmaligen Berechnung wiederverwendet werden solange mit denselben Einstellungen gearbeitet wird. Zu diesen Parametern gehören die Brennweiten (f_x, f_y) sowie das optische Zentrum (c_x, c_y) jeweils als Pixelwerte.

$$K = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

The extrinsic parameters consist of the 3x3 rotation matrix and the three dimensional translation vector denoted by R and t which describe the orientation and position of the camera in the environment. These parameters can be combined into a homogeneous transformation that maps a point in the real world coordinate system (P_w) to a point in the camera coordinate system (P_c).

Bei den extrinsischen Parametern handelt es sich um die 3x3 Rotationsmatrix und den dreidimensionalen Translationsvektor bezeichnet durch R und t die die Orientierung und Position der Kamera in der Umwelt beschreiben. Diese Parameter können zu einer homogenen Transformation kombiniert werden die die Abbildung eines Punktes aus dem Koordinatensystem der realen Welt (P_w) in das Koordinatensystem der Kamera (P_c) beschreibt.

$$P_c = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} P_w$$

To describe where the point is located in the image (P_i) and not only in the camera coordinate system (P_c) it must also be multiplied by the intrinsic camera matrix K .

Um zu beschreiben wo der Punkt im Bild (P_i) liegt und nicht nur im Koordinatensystem der Kamera (P_c) muss er zusätzlich mit der intrinsischen Kameramatrix K multipliziert werden.

$$P_i = K \times P_c$$

Using these parameters it is possible to undistort an image that was captured with a calibrated camera. It is important that the image is captured with the same resolution that was

Mithilfe dieser Parameter ist es möglich ein Bild zu entzerren das mit einer kalibrierten Kamera aufgenommen wurde. Wichtig ist dabei dass das Bild mit derselben Auflösung aufge-

used during camera calibration. If the image is scaled the intrinsic camera parameters must at least be scaled by the same factor.

Calibration is performed using a calibration pattern. The available options include a chessboard pattern a circle grid and a so called ChArUco board as shown in Figure 8.2. The latter is a chessboard pattern that is augmented with ArUco markers.

nommen wird mit der die Kamera kalibriert wurde. Wird das Bild skaliert müssen zumindest die intrinsischen Kameraparameter um denselben Faktor skaliert werden.

Die Kalibrierung erfolgt mithilfe eines Kalibrierungsmusters. Zur Auswahl stehen ein Schachbrettmuster ein Raster aus Kreisen sowie ein sogenanntes *ChArUco Board* wie in Abbildung 8.2 dargestellt. Letzteres ist ein Schachbrettmuster das mit ArUco Markierungen ergänzt wurde.

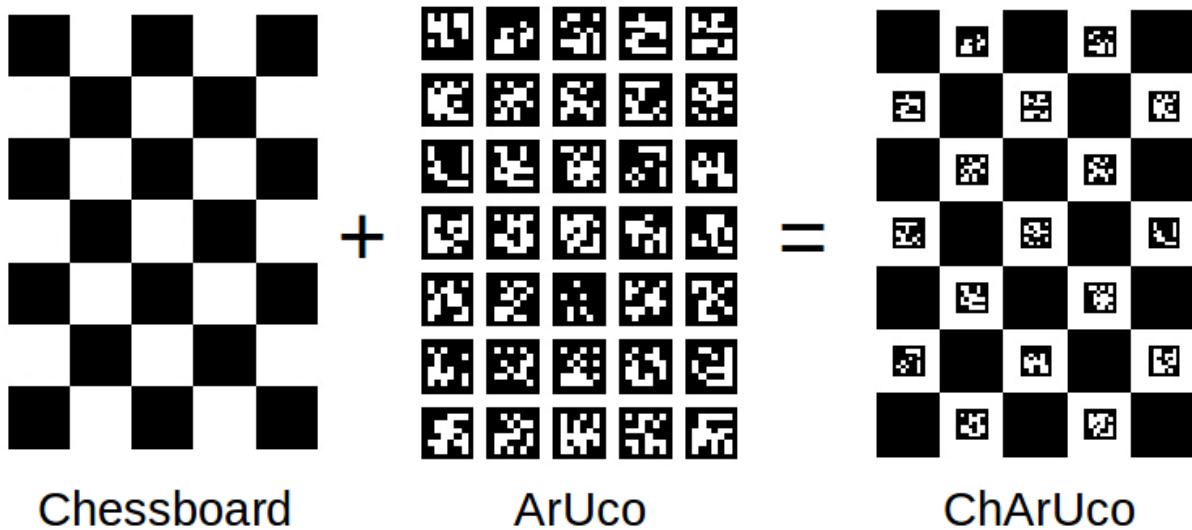


Figure 8.2: ChArUco Board Example [Quelle: OpenCV]

The advantage of the ChArUco pattern is that the ArUco markers are unique and can be individually identified. This makes it possible to capture calibration images that are partially occluded or in which not all chessboard corners are visible. This usually results in higher accuracy and should therefore be preferred over a plain chessboard pattern whenever possible.

For the other two patterns all corners or circles must be visible in the image. This makes it particularly difficult to cover the border areas of the pattern which are often critical for achieving good accuracy. In addition a larger number of images is required. The exact number of required images should be evaluated for each individual case. Ten images are generally a good starting point because at least this number is required for calibration using a chessboard pattern. The other patterns may require fewer images.

Der Vorteil dieses ChArUco Patterns besteht darin dass die ArUco Markierungen eindeutig sind und individuell zugeordnet werden können. Dadurch ist es möglich Kalibrierungsbilder aufzunehmen die teilweise verdeckt sind oder auf denen nicht alle Ecken des Schachbretts sichtbar sind. Dies führt in der Regel zu einer höheren Genauigkeit und sollte daher wenn möglich einem reinen Schachbrettmuster vorgezogen werden.

Bei den beiden anderen Mustern müssen alle Ecken beziehungsweise alle Kreise im Bild sichtbar sein. Dies macht es besonders schwierig Randbereiche der Aufnahmen abzudecken die häufig kritisch für eine gute Genauigkeit sind. Darüber hinaus werden mehr Bilder benötigt. Wie viele Bilder konkret erforderlich sind sollte im Einzelfall getestet werden. Zehn Bilder sind grundsätzlich ein guter Startpunkt da diese mindestens für die Kalibrierung mit einem Schachbrettmuster benötigt werden. Die anderen Muster können unter Umständen auch mit weniger Bildern auskommen.



Figure 8.3: ChArUco Board Detections

Regardless of the chosen calibration pattern the calibration should be performed approximately at the distance at which the camera will later be used. In addition the pattern should cover at least half of the image. It should also be displayed on a flat surface and have a high resolution. Using a screen is recommended for this purpose.

The images should also be captured from different angles and with varying inclinations. Good illumination should be ensured because uneven lighting can cause not all points to be detected in every image.

For capturing images in a dual camera setup it is recommended to synchronize the shutter release. In the case of Python this can be achieved using `threading.Barrier()` as shown in Figure 8.4.

As the final step before calibration the images from both cameras must be evaluated. This is done using

Unabhängig vom gewählten Kalibrierungsmuster sollte die Kalibrierung ungefähr in der Entfernung durchgeführt werden in der die Kamera später eingesetzt wird. Darüber hinaus sollte das Muster mindestens die Hälfte des Bildes bedecken. Außerdem sollte es auf einer flachen Oberfläche abgebildet werden und hochauflösend sein. Es empfiehlt sich hierfür einen Bildschirm zu verwenden.

Die Bilder sollten zudem aus unterschiedlichen Winkeln und mit unterschiedlichen Neigungen aufgenommen werden. Auf eine gute Belichtung sollte ebenfalls geachtet werden da eine ungleichmäßige Ausleuchtung dazu führen kann dass nicht alle Punkte auf jedem Bild erkannt werden.

Für das Aufnehmen der Bilder in einem Dual Camera Setup empfiehlt es sich den Auslöser zu synchronisieren. Im Falle von Python kann hierfür `threading.Barrier()` verwendet werden wie in Abbildung 8.4 gezeigt.

Als letzter Schritt vor der Kalibrierung müssen die Bilder der beiden Kameras ausgewertet werden. Dies geschieht mithilfe von

```
from picamera2 import Picamera2
from time import sleep
import threading

cam_zero = Picamera2(0)
cam_one = Picamera2(1)
barrier = threading.Barrier(2)

def capture_zero(index):
    barrier.wait()
    cam_zero.capture_file(f"/path/to/images_0/{index}.jpg")

def capture_one(index):
    barrier.wait()
    cam_one.capture_file(f"/path/to/images_1/{index}.jpg")

try:
    while True:
        print(counter)
        pic_zero = threading.Thread(target=capture_zero,
                                     args=(counter,))
        pic_one = threading.Thread(target=capture_one,
                                   args=(counter,))
        pic_zero.start()
        pic_one.start()
        pic_zero.join()
        pic_one.join()
        counter += 1
        sleep(3)

except KeyboardInterrupt:
    cam_zero.close()
    cam_one.close()
```

Figure 8.4: Test Listing

`cv2.aruco.CharucoDetector(board)` as shown in Figure 8.3. The detector is used to identify the chessboard corners and the corresponding ArUco markers and to represent them as numerical values. These values are then used as input for `cv2.calibrateCamera()` as shown in Figure 8.5.

The output of this function includes the intrinsic and extrinsic camera parameters the distortion coefficients the rotation matrix and the translation vector as well as the mean squared reprojection error for evaluating the calibration. Ideally this error should be smaller than one. However this alone does not guarantee a good result. If the error is smaller than one but the images cannot be undistorted it is recommended to inspect the reprojection error of the individual images.

The detector does not always identify all relevant points even under good lighting conditions. By removing unsuitable images the result can be further improved. For this purpose the markers detected with `detector.detectBoard(image)` can be visualized using `cv2.aruco.drawDetectedCornersCharuco()` and then written to disk using `cv2.imwrite()`.

Visual inspection makes it easier to identify problems with the calibration images. This is particularly relevant for chessboard patterns because it is not guaranteed that the corners are always detected in the same order.

After calibration the intrinsic camera matrix can be refined using `cv2.getOptimalCameraMatrix()` as shown in Figure 8.6. The scaling factor $\alpha = [0, 1]$ determines how many unwanted pixels remain in the image. If $\alpha = 1$ all pixels of the original image are retained which results in areas with black pixels. If $\alpha = 0$ these areas are minimized which may lead to a loss of parts of the original image information.

For testing purposes the function `cv2.undistort()` can be used. The resulting image can then be written using `cv2.imwrite()`. If too many black pixels

`cv2.aruco.CharucoDetector(board)` wie in Abbildung 8.3 dargestellt. Dieser wird verwendet um die Ecken des Schachbretts und die entsprechenden ArUco Markierungen zu erkennen und als Zahlenwerte zu beschreiben. Diese Zahlenwerte werden anschließend als Input für `cv2.calibrateCamera()` verwendet wie in Abbildung 8.5 gezeigt.

Der Output dieser Funktion umfasst die intrinsischen und extrinsischen Kameraparameter die Verzerrungskoeffizienten die Rotationsmatrix und den Translationsvektor sowie die mittlere quadratische Abweichung zur Bewertung der Kalibrierung. Diese sollte idealerweise kleiner als eins sein. Dies ist jedoch noch keine Garantie für ein gutes Ergebnis. Ist die Abweichung kleiner als eins und lassen sich die Bilder dennoch nicht entzerren empfiehlt es sich die Abweichung der einzelnen Bilder separat zu prüfen.

Nicht immer erkennt der Detector alle relevanten Punkte selbst bei guten Lichtverhältnissen. Durch das Löschen ungeeigneter Bilder kann das Ergebnis weiter verbessert werden. Zu diesem Zweck können die mit `detector.detectBoard(image)` erkannten Marker mithilfe von `cv2.aruco.drawDetectedCornersCharuco()` visualisiert und anschließend mit `cv2.imwrite()` ausgegeben werden.

Durch visuelles Inspizieren lassen sich Probleme mit den Kalibrierungsbildern einfacher erkennen. Dies ist insbesondere bei Schachbrettmustern relevant da nicht garantiert ist dass die Ecken immer in derselben Reihenfolge erkannt werden.

Nach der Kalibrierung kann die intrinsische Kameramatrix mit `cv2.getOptimalCameraMatrix()` verfeinert werden wie in Abbildung 8.6 gezeigt. Der dabei verwendete Skalierungsfaktor $\alpha = [0, 1]$ bestimmt wie viele unerwünschte Pixel im Bild verbleiben. Wird $\alpha = 1$ gewählt werden alle Pixel des ursprünglichen Bildes beibehalten wodurch Bereiche mit schwarzen Pixeln entstehen. Wird $\alpha = 0$ gewählt werden diese Bereiche minimiert wodurch unter Umständen nicht die vollständige Information des ursprünglichen Bildes erhalten bleibt.

Zum Testen kann die Funktion `cv2.undistort()` verwendet werden. Mit `cv2.imwrite()` kann das Bild anschließend ausgegeben werden. Sollten bei der Wahl von

```

import cv2
import numpy as np
import os
from glob import glob

def charuco_calibration(images_glob : str, cam : int):
    dictionary = cv2.aruco.getPredefinedDictionary(ARUCO_DICT)
    board = cv2.aruco.CharucoBoard((SQUARES_VERTICALLY,
                                    ↪ SQUARES_HORIZONTALLY), SQUARE_LENGTH, MARKER_LENGTH,
                                    ↪ dictionary)
    params = cv2.aruco.DetectorParameters()
    detector = cv2.aruco.CharucoDetector(board)
    image_paths = glob(images_glob)
    all_charuco_corners = []
    all_charuco_ids = []
    all_image_pts = []
    all_object_pts = []
    counter = 1

    for image_path in image_paths:
        image = cv2.imread(image_path)
        grayscaled = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        charuco_corners, charuco_ids, marker_corners,
        ↪ marker_ids = detector.detectBoard(image)

        if charuco_corners is not None:
            image =
                ↪ cv2.aruco.drawDetectedCornersCharuco(image,
                ↪ charuco_corners, charuco_ids)
            object_points, image_points =
                ↪ board.matchImagePoints(charuco_corners,
                ↪ charuco_ids)
            all_object_pts.append(object_points)
            all_image_pts.append(image_points)
            all_charuco_corners.append(charuco_corners)
            all_charuco_ids.append(charuco_ids)
            counter += 1

        image_height, image_width = image.shape[:2]
        retval, camera_matrix, dist_coeffs, rvecs, tvecs =
            ↪ cv2.calibrateCamera(all_object_pts, all_image_pts,
            ↪ (image_height, image_width), None, None)
        return all_object_pts, all_image_pts, retval, camera_matrix,
            ↪ dist_coeffs, rvecs, tvecs, image_height, image_width

```

Figure 8.5: ChArUco Board Corner Detection

```

criteria_stereo = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER,
                   100, 0.0001)
retS, new_mtx_zero, dist_zero, new_mtx_one, dist_one, Rot, Trns, Emat,
      Fmat = cv2.stereoCalibrate(obj_pts_one, img_pts_zero,
                                 img_pts_one, mtx_zero, dist_zero, mtx_one, dist_one,
                                 (w_zero, h_zero), criteria_stereo, cv2.CALIB_FIX_INTRINSIC_GUESS)

```

Figure 8.7: Single Camera calibration

```

obj_pts_zero, img_pts_zero, ret_zero, mtx_zero, dist_zero, rvecs_zero,
      tvecs_zero, h_zero, w_zero =
      charuco_calibration(images_glob=images_zero, cam=0)
new_mtx_zero, roi_zero = cv2.getOptimalNewCameraMatrix(mtx_zero,
      dist_zero, (w_zero, h_zero), 0, (w_zero, h_zero))

obj_pts_one, img_pts_one, ret_one, mtx_one, dist_one, rvecs_one,
      tvecs_one, h_one, w_one = charuco_calibration(images_one, 1)
new_mtx_one, roi_one = cv2.getOptimalNewCameraMatrix(mtx_one, dist_one,
      (w_one, h_one), 0, (w_one, h_one))

```

Figure 8.6: Single Camera Calibration & Optimal Camera Matrix

remain and $\alpha = 0$ is used the image can additionally be cropped to the region of interest ROI.

$\alpha = 0$ zu viele schwarze Pixel vorhanden sein kann das Bild zusätzlich auf die *Region of Interest ROI* zugeschnitten werden.

8.2

Stereo Calibration

After successful calibration of the individual cameras it is possible to project a 3D point onto a 2D image and to undistort it. In order to reconstruct the three dimensional information it is necessary to determine the relative pose of the two cameras.

For this purpose the function `cv2.stereoCalibrate` is used as shown in Figure 8.7. The variable `criteria_stereo` allows the computation to be refined. The flag `cv2.CALIB_FIX_INTRINSIC_GUESS` ensures that the intrinsic matrices obtained from single camera calibration are not modified during this process.

It is important that the calibration images are synchronized. This is ensured by the code shown in Figure 8.4. If required a second set of calibration images can be created. In this case the points must be detected again.

Stereo-Kalibrierung

Nach einer erfolgreichen Kalibrierung der einzelnen Kameras ist es jeweils möglich einen 3D Punkt auf ein 2D Bild zu projizieren und dieses zu entzerren. Um die dreidimensionale Information wiederherzustellen muss berechnet werden wie die beiden Kameras zueinander ausgerichtet sind.

Hierfür wird die Funktion `cv2.stereoCalibrate` verwendet wie in Abbildung 8.7 gezeigt. Die Variable `criteria_stereo` ermöglicht es die Berechnung zu verfeinern. Die Flagge `cv2.CALIB_FIX_INTRINSIC_GUESS` stellt sicher dass die intrinsischen Matrizen aus der Einzelkalibrierung während dieses Prozesses nicht verändert werden.

Wichtig ist dabei dass die Kalibrierungsbilder synchronisiert sind. Dies wird durch den Code in Abbildung 8.4 sichergestellt. Bei Bedarf kann jedoch auch ein zweiter Satz Kalibrierungsbilder erstellt werden. In diesem Fall müssen die Punkte erneut erkannt werden.

The result of stereo calibration consists of the rotation and translation vectors R and t between the two cameras as well as the essential matrix E and the fundamental matrix F . The rotation matrix describes the three dimensional rotation from camera one to camera two. The translation vector describes the displacement of the coordinate system of the first camera relative to the second camera. In addition the mean squared reprojection error is returned which should ideally be smaller than three.

The essential matrix is derived from the two previously mentioned vectors according to $E = [t] \times R$. Using this matrix it is possible to determine where a point in the coordinate system of camera one is located in the coordinate system of camera two. The line connecting these corresponding points is referred to as the epipolar line.

The fundamental matrix serves the same purpose but operates on pixel coordinates instead of normalized coordinates. It is computed using the essential matrix and the intrinsic matrices of both cameras.

$$F = K'^{-\top} E K^{-1}$$

Using these matrices the function `cv2.stereoRectify()` can be applied to make the epipolar lines horizontal as shown in Figure 8.8. As a result both images share the same Y coordinates even if they are not mounted perfectly horizontally. This significantly improves the robustness of depth estimation because corresponding points in image one can also be found in image two. The results of the rectification include the rectification and projection matrices as well as the matrix Q and the respective regions of interest of both images.

In this context the ROI describes an area within the image in which all pixels are valid. The rectification matrices describe how the respective images must be adjusted to obtain horizontal epipolar lines. The projection matrices (P_0, P_1) describe the required transformation to project a three dimensional point into the respective two dimensional coordinate system independently of the scene. They can be interpreted as an extension of the intrinsic camera matrix K .

Das Ergebnis der Stereokalibrierung sind die Rotations und Translationsvektoren R und t zwischen den beiden Kameras sowie die essentielle Matrix E und die fundamentale Matrix F . Die Rotationsmatrix beschreibt die dreidimensionale Rotation von Kamera eins zu Kamera zwei. Der Translationsvektor beschreibt die Verschiebung des Koordinatensystems der ersten Kamera relativ zur zweiten Kamera. Darüber hinaus wird die mittlere quadratische Abweichung zurückgegeben welche idealerweise ebenfalls kleiner als drei sein sollte.

Die essentielle Matrix ergibt sich aus den beiden zuvor genannten Vektoren nach $E = [t] \times R$. Mithilfe dieser Matrix lässt sich bestimmen wo ein Punkt im Koordinatensystem von Kamera eins im Koordinatensystem von Kamera zwei liegt. Die Verbindungsline zwischen diesen beiden Punkten wird als epipolare Linie bezeichnet.

Die fundamentale Matrix erfüllt den gleichen Zweck arbeitet jedoch mit Pixelkoordinaten anstelle von normalisierten Koordinaten. Sie wird mithilfe der essentiellen Matrix und der intrinsischen Matrizen beider Kameras berechnet.

Mithilfe dieser Matrizen kann die Funktion `cv2.stereoRectify()` angewendet werden um die epipolaren Linien horizontal auszurichten wie in Abbildung 8.8 dargestellt. Dadurch besitzen beide Bilder die gleichen Y Koordinaten auch wenn sie nicht perfekt horizontal montiert sind. Dies macht die Tiefenerkennung deutlich robuster da alle Punkte in Bild eins auch in Bild zwei gefunden werden können. Die Resultate der Korrektur umfassen die Korrektur und Projektionsmatrizen sowie die Matrix Q und die jeweilige ROI der beiden Bilder.

In diesem Zusammenhang beschreibt die ROI einen Bereich im Bild in dem alle Pixel valide sind. Die Korrekturmatrizen beschreiben wie die jeweiligen Bilder angepasst werden müssen um horizontale Epipolarlinien zu erhalten. Die Projektionsmatrizen (P_0, P_1) beschreiben die notwendige Transformation um einen dreidimensionalen Punkt in das jeweilige zweidimensionale Koordinatensystem zu projizieren unabhängig von der Szenerie. Sie können als Erweiterung der intrinsischen Kameramatrix K interpretiert werden.

```

Stereo_Map_Zero = cv2.initUndistortRectifyMap(new_mtx_zero, dist_zero,
    ↪ rect_zero, proj_mat_zero, (w_zero,h_zero), cv2.CV_16SC2)
Stereo_Map_One = cv2.initUndistortRectifyMap(new_mtx_one, dist_one,
    ↪ rect_one, proj_mat_one, (w_zero,h_zero), cv2.CV_16SC2)

cv_file = cv2.FileStorage("/path/to/stereo_maps.xml",
    ↪ cv2.FILE_STORAGE_WRITE)
cv_file.write("Stereo_Map_Zero_x", Stereo_Map_Zero[0])
cv_file.write("Stereo_Map_Zero_y", Stereo_Map_Zero[1])
cv_file.write("Stereo_Map_One_x", Stereo_Map_One[0])
cv_file.write("Stereo_Map_One_y", Stereo_Map_One[1])
cv_file.release()

```

Figure 8.9: Initializing the stereo mapping function

```

rectify_scale = 0
rect_zero, rect_one, proj_mat_zero, proj_mat_one, Q, roi_zero, roi_one
    ↪ = cv2.stereoRectify(new_mtx_zero, dist_zero, new_mtx_one,
    ↪ dist_one, (w_zero,h_zero), Rot, Trns, rectify_scale,(0,0))

```

Figure 8.8: Stereo Rectification

$$P_0, P_1 = \begin{bmatrix} f_x & 0 & c_x & t_x \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

The added value t_x describes the horizontal offset between the two images expressed in pixel units. The parameters f and c continue to represent the focal length and the optical center. After applying `cv2.stereoRectify()` it should normally hold that $f_x = f_y$.

The matrix Q represents a reprojection matrix. Its usage is explained in Section 8.3.

Since the two cameras are mounted statically it is sufficient to compute rectification once and then store the result. This saves a significant amount of computation time and improves overall performance. For this purpose the function `cv2.initUndistortRectifyMap()` can be used as shown in Figure 8.9. It creates a mapping function between the cameras that can be reused repeatedly.

Der hinzugefügte Wert t_x beschreibt den horizontalen Versatz zwischen den beiden Bildern als Pixelwert. f und c beschreiben weiterhin die Brennweite und das optische Zentrum. Nach der Verwendung von `cv2.stereoRectify()` sollte in der Regel $f_x = f_y$ gelten.

Die Matrix Q beschreibt eine *Reprojektionsmatrix*. Wie diese verwendet werden kann wird in Abschnitt 8.3 erläutert.

Da die beiden Kameras statisch montiert sind genügt es die Rectification einmal zu berechnen und anschließend zu speichern. Dadurch kann viel Rechenzeit eingespart werden was die Performance verbessert. Hierfür kann die Funktion `cv2.initUndistortRectifyMap()` verwendet werden wie in Abbildung 8.9 gezeigt. Diese erstellt eine Mapping Funktion zwischen den Kameras die wiederholt verwendet werden kann.

8.3

Tiefenerkennung

After the images have been aligned along the Y axis the differences between the captured images must be computed. For this purpose the

Depth Estimation

Nun da die Bilder entlang einer gemeinsamen Y Achse ausgerichtet sind gilt es die Unterschiede zwischen den aufgenommenen Bildern zu

block matching algorithm is used. It divides the image into smaller blocks such as 5×5 pixels and searches for corresponding blocks along the epipolar line in the reference image.

Using configurable metrics such as the sum of absolute differences the sum of squared differences or normalized cross correlation the blocks are compared sequentially and a so called disparity map is generated.

This disparity map represents the displacement between the two images within the respective blocks for example encoded as grayscale values. OpenCV provides the class `cv2.StereoBM` which can be used to compute such a disparity map.

Alternatively semi global block matching can be used to compute the disparity map. OpenCV also provides a corresponding class named `cv2.StereoSGBM`. SGBM enables sub pixel computations which can result in sharper textures.

Both BM and SGBM can be refined using a wide range of parameters. This includes for example the block size that is evaluated or whether a pre filter should be applied to enhance image textures. The method `compute(img_l, img_r)` which performs the actual disparity computation returns values that are scaled by a factor of sixteen. To obtain the actual disparity values the result must be divided by sixteen.

From these disparity values D the depth values Z can be computed. This requires the focal length f and the baseline B . These values are not taken from the original calibration but from the results of `cv2.stereoCalibrate()` and `cv2.stereoRectify()`. The baseline corresponds to the first component of the translation vector obtained during calibration and the focal length can be extracted from the projection matrix of camera one because after rectification it typically holds that $f_x = f_y$. To compute the depth value for a given point the following calculation must be performed.

berechnen. Hierzu wird der Block Matching Algorithmus verwendet. Dieser teilt das Bild in kleinere Blöcke wie beispielsweise 5×5 Pixel auf und sucht entlang der epipolaren Linie im Referenzbild nach korrespondierenden Blöcken.

Anhand unterschiedlicher konfigurierbarer Metriken wie der Summe der absoluten Differenzen der Summe der quadrierten Differenzen oder der normalisierten Kreuzkorrelation werden die Blöcke nacheinander verglichen und eine sogenannte Disparity Map erstellt.

Diese Disparity Map repräsentiert die Verschiebung zwischen den beiden Bildern innerhalb der jeweiligen Blöcke beispielsweise als Grauwert. OpenCV stellt hierfür die Klasse `cv2.StereoBM` bereit mit der eine solche Disparity Map berechnet werden kann.

Alternativ kann auch Semi Global Block Matching verwendet werden um die Disparity Map zu berechnen. Hierfür existiert ebenfalls eine OpenCV Klasse mit dem Namen `cv2.StereoSGBM`. SGBM ermöglicht Subpixel Berechnungen wodurch Texturen schärfer dargestellt werden können.

Sowohl BM als auch SGBM können mithilfe einer Vielzahl von Parametern verfeinert werden. Dazu zählen beispielsweise die zu prüfende Blockgröße oder die Verwendung eines Pre Filters um Texturen im Bild besser hervorzuheben. Die Methode `compute(img_l, img_r)` welche die eigentliche Disparity Berechnung durchführt liefert Werte zurück die mit dem Faktor sechzehn skaliert sind. Um die tatsächlichen Disparity Werte zu erhalten muss das Ergebnis durch sechzehn geteilt werden.

Aus diesen Disparity Werten D können die Tiefenwerte Z berechnet werden. Hierfür werden zusätzlich die Brennweite f und die Grundlinie B benötigt. Diese Werte werden nicht aus der ursprünglichen Kalibrierung entnommen sondern aus den Ergebnissen von `cv2.stereoCalibrate()` und `cv2.stereoRectify()`. Die Grundlinie entspricht der ersten Komponente des Übersetzungsvektors aus der Kalibrierung und die Brennweite kann aus der Projektionsmatrix von Kamera eins entnommen werden da nach der Korrektur in der Regel $f_x = f_y$ gilt. Um einen Tiefenwert für einen gegebenen Punkt zu berechnen muss folgende Berechnung durchgeführt werden.

$$Z = \frac{Bf}{D}$$

This computation does not necessarily have to be performed explicitly. The matrix Q returned by `cv2.stereoRectify()` can be used in combination with the method `reprojectImageTo3D` to obtain the depth values. These are stored in `im3d(x, y, ch)` and contain the three dimensional coordinates for the respective point (x, y) .

If the computation is nevertheless performed explicitly it is recommended to normalize the disparity values. Furthermore it must be ensured that $D > 0$ for each point in the image because otherwise incorrect depth values are computed. This can cause the resulting depth values to appear distorted.

A potential benefit of this explicit computation is that it is not necessary to calculate depth values for all pixels but only for regions of interest. This can be combined with object detection and is particularly relevant for higher resolutions. In general the limitations of the Raspberry Pi must be considered. At a resolution of 640x480 the Raspberry Pi is capable of computing the disparity at a usable frame rate. Higher resolutions must be evaluated on a case by case basis.

Diese Berechnung muss jedoch nicht explizit durchgeführt werden. Die Matrix Q die von `cv2.stereoRectify()` zurückgegeben wird kann in Verbindung mit der Methode `reprojectImageTo3D` verwendet werden um die Tiefenwerte zu bestimmen. Diese werden in `im3d(x, y, ch)` ausgegeben und enthalten die dreidimensionalen Koordinaten für den jeweiligen Punkt (x, y) .

Sollte die Berechnung dennoch explizit durchgeführt werden empfiehlt es sich die Disparity zu normalisieren. Darüber hinaus muss darauf geachtet werden dass $D > 0$ für jeden Punkt im Bild gilt da andernfalls falsche Tiefenwerte berechnet werden. Dies kann dazu führen dass die resultierenden Tiefenwerte verzerrt wirken.

Ein möglicher Vorteil dieser expliziten Berechnung besteht darin dass nicht alle Tiefenwerte berechnet werden müssen sondern nur für relevante Regionen. Dies lässt sich in Verbindung mit einer Objekterkennung realisieren und ist insbesondere für höhere Auflösungen interessant. Grundsätzlich sind dabei die Limitierungen des Raspberry Pi zu berücksichtigen. Bei einer Auflösung von 640x480 ist das Raspberry Pi in der Lage die Disparity mit einer nutzbaren Frame-rate zu berechnen. Höhere Auflösungen müssen im Einzelfall geprüft werden.