






B) Anhang B Software, Parametrierungen, Testprotokolle & Skripte

Dieser Anhang dokumentiert die zur Umsetzung des autonomen Landeverfahrens verwendete Softwareumgebung. Er umfasst die zentralen Python-Skripte, Systemdienste (systemd), Parameterdateien des Flugcontrollers sowie ergänzende Hilfsprogramme (Batch- und Auswertungsskripte).

Zur Nachvollziehbarkeit und Reproduzierbarkeit ist das gesamte Projekt als öffentliches GitHub-Repository verfügbar. Dort sind sämtliche im Anhang beschriebenen Dateien strukturiert abgelegt und versioniert:

GitHub-Repository <https://github.com/plexgen/Autopilot-Microcruiser-RC-Flugzeug>

-  **scripts/** – Hauptskripte zur Bildverarbeitung, MAVLink-Kommunikation und Zeiterfassung
- - autoland_trigger_on_land.py – Hauptskript zur autonomen Triggerung des Landemodus bei erkannter Bahn
- - Sensordaten_plott_und_excel.py – Auswertungsskript (BIN → HTML + Excel-Export)
- - zeiterfassung.py – Skript zur Zeiterfassung aus der Vorprojektphase (Erstellt mit Unterstützung eines KI-Hilfsmittels, GPT-5)
-  **batch/** – Datentransfer
- - Datendownload.bat – Automatischer Video- und Log-Download vom Raspberry Pi
-  **configs/** – Konfigurationen
- - microcruiser.xml – Fahrzeugkonfiguration
- - reset_microcruiser.xml – Reset-Konfiguration
-  **params/** – Parameterdateien
- - pixracer_autoland.param – Pixracer-Parameterdatei (ArduPlane 4.6.2)
-  **docs/** – Begleitdokumentation
- - Anhang_B_Autopilot_Microcruiser_final.pdf – Begleitdokumentation (dieser Anhang)

Systemübersicht

Trägerplattform: Microcruiser RC-Flugzeug (JTPaero – Microcruiser)

- Spannweite \approx 460 mm, klassisches Starrflügelmodell
- Antrieb: Elektromotor (Impeller)
- Steuerung: Quer-, Höhenruder, Gas
- Missionsprofil: Wegpunktflug, autonomes Landeverfahren

Autopilot: Pixracer R15 (ArduPlane 4.6.2)

- Fluglagenregelung, Missionsmanagement und Sensorfusion
- Kommunikation über MAVLink (UART) mit dem CM4
- Unterstützte Kommandos: SET_MODE, DO_LAND_START, SET_POSITION_TARGET_GLOBAL_INT
- Firmwarebasis: ArduPlane v4.6.2

Compute Module 4 (CM4):

- Raspberry Pi Compute Module 4
- CSI-Kamera (IMX708)
- Eigenes UBEC (5 V) zur Versorgung
- Laufzeitumgebung: python3-venv mit pymavlink, picamera2, opencv
- Systemd-Service autoland.service für automatischen Start beim Boot

Funktion:

- Visuelle Bahnerkennung in Echtzeit
- Automatischer Wechsel in den Landemodus
- Log- und Videoerstellung zur Nachweisführung

CM4 Setup (Kurzüberblick)

1. 1. System aktualisieren:

```
sudo apt update && sudo apt upgrade -y
```

2. 2. Kamera aktivieren & Tools installieren:

```
sudo apt install -y libcamera-apps python3-picamera2 python3-opencv python3-numpy python3-simplejpeg
```

3. 3. Virtuelle Umgebung einrichten:

```
python3 -m venv ~/mavenv --system-site-packages
```

```
source ~/mavenv/bin/activate
```

```
pip install pymavlink pyserial
```

4. 4. Autostart (systemd):

Service-Datei /etc/systemd/system/autoland.service mit ExecStart=/home/pi/mavenv/bin/python3 /home/pi/autoland_trigger_on_land.py

5. 5. Service aktivieren:

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable autoland.service
```

```
sudo systemctl start autoland.service
```

Parameterdatei (Pixracer)

Die vollständige ArduPlane-Parameterdatei (pixracer_autoland.param) ist im Ordner params/ des GitHub-Repositories abgelegt. Sie enthält alle für das Projekt relevanten Einstellungen (u. a. Flugmodi, Sensorik, EKF3-Konfiguration, Landemodi).

Dokumentation & Nachweise

Alle Nachweise (Video, Logdaten, Screenshots) sowie das vollständige Anhangsdokument:

docs/Anhang_B_Autopilot_Microcruiser_final.pdf

GitHub: <https://github.com/plexgen/Autopilot-Microcruiser-RC-Flugzeug>