

**Schlussbericht zum Vorhaben**

**Einsatz autonomer Drohnen zur nachhaltigen  
Pflanzenproduktion in Gewächshäusern - Teilprojekt 2**

**Akronym:** FlyingData  
**FKZ:** 2818507B18  
**Laufzeit:** 01.01.2020 – 31.12.2023

**Zuwendungsempfänger:** RAM GmbH Mess- und Regeltechnik  
Gewerbestr.3  
82211 Herrsching  
**Autor:** Dr. Gökhan Akyazi (Forschungskordinator)



Gefördert durch:



aufgrund eines Beschlusses  
des Deutschen Bundestages

## 1. Inhaltsverzeichnis

1.	Kurzdarstellung .....	3
1.1	Aufgabenstellung .....	3
1.2	Voraussetzungen, unter denen das Vorhaben durchgeführt wurde .....	3
1.3	Planung und Ablauf des Vorhabens .....	4
1.4	Wissenschaftlicher und technischer Stand, an den angeknüpft wurde .....	6
1.5	Zusammenarbeit mit anderen Stellen .....	6
2.	Eingehende Darstellung .....	7
2.1.	Verwendung der Zuwendung und des erzielten Ergebnisses im Einzelnen, mit Gegenüberstellung der vorgegebenen Ziele .....	7
2.1.1.	Entwicklung eines Systems zur Visualisierung der Drohnendaten.....	8
2.1.2.	Mobiles hochaufgelöstes Messsystem für Temperatur und rel. Luftfeuchte .....	20
2.1.3.	Anforderungen zur Nutzung hochaufgelöster Daten für die Klimaregelung.....	23
2.2.	Wichtigste Positionen des zahlenmäßigen Nachweises .....	26
2.3.	Notwendigkeit und Angemessenheit der geleisteten Arbeit .....	26
2.4.	Voraussichtlicher Nutzen, insbesondere der Verwertbarkeit des Ergebnisses im Sinne des fortgeschriebenen Verwertungsplans .....	26
2.5.	Während der Durchführung des Vorhabens dem ZE bekannt gewordenen Fortschritts auf dem Gebiet des Vorhabens bei anderen Stellen.....	27
2.6.	Erfolgte oder geplante Veröffentlichungen der Ergebnisse .....	27
2.7.	Referenzen .....	28
3.	Erfolgskontrollbericht.....	29
3.1.	Beitrag der Ergebnisse zu den förderpolitischen Zielen .....	29
3.2.	Wissenschaftlich-technische Ergebnisse.....	29
3.3.	Fortschreibung des Verwertungsplans .....	30
3.4.	Arbeiten, die zu keiner Lösung geführt haben .....	32
3.5.	Präsentationsmöglichkeiten für mögliche Nutzende – z. B. Anwenderkonferenzen	32
3.6.	Einhaltung der Kosten- und Zeitplanung.....	32
4.	Kurzfassung .....	34

## **1. Kurzdarstellung**

### **1.1 Aufgabenstellung**

Gemäß der Richtlinie über die Förderung von Innovationen für einen Gartenbau 4.0 vom 02. August 2018 und basierend auf den Ergebnissen aus dem Forschungsvorhaben „HortInnova“, wurden Forschungsschwerpunkte identifiziert und in einem Förderschwerpunkt Gartenbau 4.0, bestehend aus den Bereichen Automatisierung, Sensorik und Big Data zusammengefasst. Die detaillierten Inhalte der Ausschreibung und die mit ihr verfolgten Ziele sind dem Ausschreibungstext zu entnehmen (BMEL, 2018).

Im Fokus der Untersuchungen im Forschungsprojekt FlyingData stand die Entwicklung einer autonom fliegenden Drohne für den Einsatz in Gewächshäusern. Während des Drohnenflugs sollen relevante Klimaparameter aufgezeichnet und diese in einer Datenbank gespeichert werden. Im Weiteren werden diese Daten genutzt, um die Verteilung der Klimaparameter im Gewächshaus grafisch darzustellen (Visualisierung).

### **1.2 Voraussetzungen, unter denen das Vorhaben durchgeführt wurde**

Die RAM GmbH Mess- und Regeltechnik ist ein deutschlandweit führendes Unternehmen im Bereich der Gewächshaus- und Gebäudeautomation und verfügt über tiefgehende Erfahrung in der regelungstechnischen Ausstattung von Praxis- und Versuchsgewächshäusern.

Basierend auf dem bewilligten Projektantrag wurde das Verbundprojekt von den beteiligten, unten aufgeführten Partner\*innen bearbeitet.

Hochschule Osnabrück

Sedanstr. 26

49076 Osnabrück

COALA Kompetenzzentrum

- Labor für Biosystemtechnik
- Labor für Mikro- und Optoelektronik
- Labor für Softwaretechnik

iotec GmbH

Albert-Einstein-Straße 1

49076 Osnabrück

RAM GmbH Mess- und Regeltechnik

Gewerbestraße 3

82211 Herrsching

Das Projekt FlyingData wurde zunächst mit einer Laufzeit von drei Jahren (01.01.2020 – 31.12.2022) bewilligt. Signifikante Einschränkungen im Zusammenhang mit SARS- CoV- 2- Pandemie hatten in direkter und indirekter Folge einen zum Teil tiefgehenden Einfluss auf den Versuchsablauf. Um diese Einschränkungen zu kompensieren und weitere Aspekte des Einsatzes autonomer Drohnen im Unterglasanbau zu erfassen, wurde eine Laufzeitverlängerung (Aufstockungsantrag) bewilligt und die Projektlaufzeit bis zum 31.12.2023 verlängert.

### 1.3 Planung und Ablauf des Vorhabens

Das Projekt FlyingData wurde bei der Antragstellung in insgesamt 24 Arbeitspakete unterteilt (siehe Tab.1.1).

Tabelle 1.1 Arbeitsplan – Arbeitspakete, Meilensteine und Zeitplan

Lfd. Nr.	Durchführung	Projektquartal													
		1.1	1.2	1.3	1.4	2.1	2.2	2.3	2.4	3.1	3.2	3.3	3.4		
AP1	RAM	■			■										
AP2	iotec	■													
AP3	iotec		■	■	■	M1									
AP4	Ra		■	■	■										
AP5	Ru	■	■												
AP6	Ru		■	■											
AP7	Ra		■	■	■										
AP8	Ra				■	M2									
AP9	We			■	■	■									
AP10	Ra				■	■	■	■							
AP11	iotec			■	■	■	■	■		■			■		
AP12	Ra					■	■	■	■	M4					
AP13	RAM		■	■	■				■	■			■		
AP14	Ru				■	■	■	■							
AP15	Be				■	■	■	■	M3						
AP16	Be							■	■	■	■				
AP17	RAM					■	■	■	■	■					
AP18	Be							■	■	■					
AP19	RAM									■	■		M5		
AP20	Ra									■	■	■	■	■	
AP21	COALA								■	■	■	■	■	■	
AP22	RAM, iotec								■	■	■	■	■	■	
AP23	COALA, RAM, iotec												■	■	
AP24	COALA, RAM, iotec	■ alle													

1. RAM GmbH (RAM)
2. iotec GmbH (iotec)
3. Labor für Biosystemtechnik, Hochschule Osnabrück, COALA, Prof. Rath (Ra), Prof. Bettin (Be)
4. Labor für Mikro- und Optoelektronik, Hochschule Osnabr., COALA, Prof. Ruckelshausen (Ru)
5. Labor für Softwaretechnik, Hochschule Osnabrück, COALA, Prof. Westerkamp (We)

## **Arbeitspakete FlyingData**

### AP 1

- Definition und Spezifikation von Datenschnittstellen

### AP 13

- Entwicklung notwendiger Userschnittstellen (siehe 2.1.1.1)

### AP 17

- Visualisierung der 3D-Klimadaten und Copterpositionen (siehe 2.1.1.2)

### AP 19

- Datenfusion und Erweiterung des Klimaregelrechnersystems um das wissensbasierte Regelsystem (siehe 2.1.1.1 und 2.1.3)

### AP 22

- Vorstellung erreichter Zwischenergebnisse auf Ausstellungen. Messen und Verbandstreffen, Einholen von Erfahrungen und Kommentaren aus der Praxis (siehe 2.6)

## **Aufstockungsantrag**

### AP 15

- Konzeption einer möglichen Gesamt- RAM- Visualisierung von Bildern und Daten im 3D-Kontext (siehe 2.1.1.2)

### AP 16

- Integration der gesendeten Daten in RAM die neue Softwareumgebung unter der Verwendung des bisherigen Konzeptes (2.1.1.1 und 2.1.3)

### Ap 17

- Softwaretests mit Realdaten und Daten RAM aus realen Betrieben und Optimierung der Darstellung (siehe 2.1.1)

### AP 18

- Visualisierung und Konzept zur Integration der Daten in ein Klima- Regel- System liegt vor (siehe 2.1.1.2, 2.1.2 und 2.1.3)

#### **1.4 Wissenschaftlicher und technischer Stand, an den angeknüpft wurde**

Um eine Pflanzenproduktion möglichst energieeffizient und gleichzeitig unter bestmöglichen Bedingungen für die Kulturen zu gestalten, leistet die Erfassung von Klimadaten und die dafür verwendete Sensortechnologie einen entscheidenden Beitrag. Mithilfe einer verlässlichen und präzisen Datenerfassung können die abgeleiteten und nachfolgenden Regelprozesse bestmöglich unterstützt werden und darüber hinaus auch für eine genaue Erfassung des Ist- Zustandes (Monitoring) eingesetzt werden. Am Beispiel der Temperaturerfassung kann dies verdeutlicht werden:

In Gewächshäusern ist mitunter mit einer großen Inhomogenität, bedingt durch die Auslegung des Heizungssystems, Konvektion, Undichtigkeiten oder (lokal) auftretenden Störgrößen zu rechnen, sodass Gradienten bis 1,5 K und mehr entstehen können. Daraus abgeleitet wird empfohlen die Datenerfassung auf fünf Messorte zu verteilen und daraus einen repräsentativen Mittelwert zu bilden (KTBL, 2008). In der Praxis kommt eine solche Verteilung der Sensoren in der Regel nicht zum Einsatz (Erfahrungen der RAM GmbH Mess- und Regeltechnik). Der Einsatz autonomer Drohnen kann einen Beitrag leisten, die Verteilung von Klimaparametern im Gewächshaus detailliert zu erfassen.

#### **1.5 Zusammenarbeit mit anderen Stellen**

Eine Zusammenarbeit fand seitens der RAM GmbH Mess- und Regeltechnik nur mit den Projektteilnehmenden innerhalb des Konsortiums statt.

## 2. Eingehende Darstellung

### 2.1. Verwendung der Zuwendung und des erzielten Ergebnisses im Einzelnen, mit Gegenüberstellung der vorgegebenen Ziele

Innerhalb des Projektkonsortiums wurden die Arbeitspakete und die darin enthaltenen Aufgabenbereiche gemäß Abschnitt 1.1 unterteilt. Die maßgebliche Aufgabenstellung der RAM GmbH Mess- und Regeltechnik war es, ein System mitsamt der notwendigen Komponenten und Peripherie zu entwickeln und im Prototyp- Maßstab zu etablieren, um die während des Drohnenfluges erfassten und in einer Datenbank abgelegten Messwerte grafisch aufzubereiten. Als gemeinsame Schnittstelle fungierte hier ein quelloffenes Datenbanksystem (InfluxDB).

In Abbildung 2.1 sind die maßgeblichen Aufgabenbereiche der Projektteilnehmenden dargestellt.

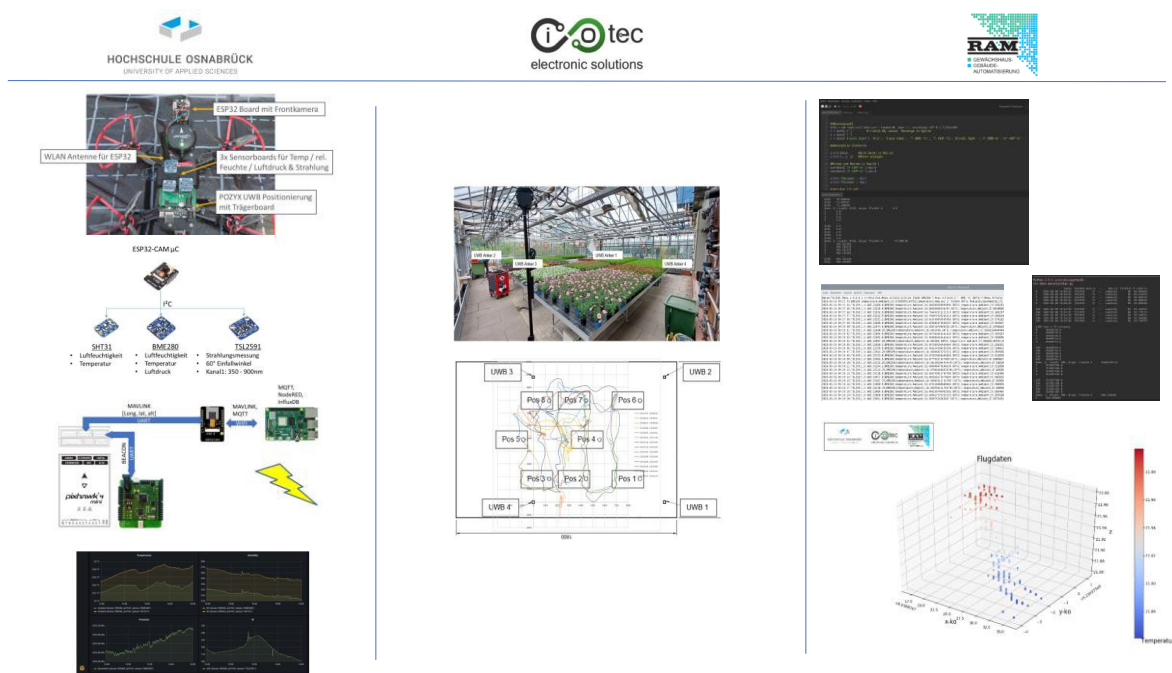


Abbildung 2.1 Aufgabenbereiche der Projektteilnehmenden im Projekt FlyingData

Für die Untersuchungen innerhalb des Projekts FlyingData wurden seitens der HS Osnabrück Drohnen beschafft, optimiert und mit zusätzlicher Sensorik zur Erfassung der relevanten Klimaparameter ausgestattet. Zusammen mit den Werksseitig verbauten, internen Sensoren ergaben sich die in Tabelle 2.1 aufgeführten Messgrößen, die zur weiteren Verarbeitung und anschließenden Visualisierung genutzt wurden.

Tabelle 2.1 Herstellerseitig und zusätzlich verbaute Sensorik auf der Drohne

<b>Sensorbezeichnung</b>	<b>Messgröße</b>
BME 280	Temperatur / rel. Luftfeuchte / Luftdruck
SHT 31	Temperatur / rel. Luftfeuchte
TSL2591	Ir- Strahlung
Interne Sensorik	geografische Länge, Breite und Flughöhe

Über die Systementwicklung gemäß des Arbeitsplans hinaus, wurden die Projektpartner\*innen der HS Osnabrück bei der Versuchsanstellung während der Projektlaufzeit technisch unterstützt. So wurde unter anderem an mehreren Terminen und dann jeweils für einen Zeitraum von drei bis vier Monaten, ein im Hause RAM entwickeltes, mobiles Messsystem zur Erfassung von Temperatur und Luftfeuchte eingesetzt (Installation, Einrichtung, Funktions- und Nutzungseinweisung, regelmäßige Kalibrierung, etc.) um die Auswirkungen des Drohnenflugs auf die Klimaparameter Temperatur und relative Luftfeuchte erfassen zu können. Ferner wurden auch die Untersuchungen in den Praxisbetrieben unterstützend begleitet. Die hierbei gewonnenen Ergebnisse sind im Detail dem Endbericht der HS Osnabrück zu entnehmen.

Der in den folgenden Abschnitten (2.1.1 bis 2.1.3) aufgeführte Quellcode wurde auf die essentiellen Komponenten (Module, Bibliotheken, skizzierte Syntax, etc.) reduziert und ist zum Teil unvollständig und in der dargestellten Form nicht lauffähig. Es handelt sich um das geistige Eigentum der RAM GmbH Mess- und Regeltechnik und wird in dem hier vorliegenden Bericht nicht veröffentlicht.

#### 2.1.1. Entwicklung eines Systems zur Visualisierung der Drohnendaten

Auf Basis eines im Projektkonsortium erstellten Anforderungskataloges (AP 1) wurden die technischen Anforderungen an das zu entwickelnde System festgelegt und mögliche Soft-, sowie Hardwarekomponenten seitens der RAM GmbH Mess- und Regeltechnik dahingehend überprüft und ausgewählt. Ein großes Augenmerk wurde bei der Auswahl auf Stabilität und Verlässlichkeit gelegt. Gleichzeitig sollten die zum Einsatz kommenden Systemkomponenten kostengünstig und / oder unter Einhaltung der Lizenzbestimmungen kostenfrei nutzbar sein (open source).

Die verwendeten Soft- und Hardware-Komponenten sind der Tabelle 2.2 zu entnehmen.

Tabelle 2.2 Verwendete Soft- und Hardwarekomponenten zur Realisierung der Datenvisualisierung

Bezeichnung	Hersteller/Lizenz
<b>Hardware:</b>	
RaspberryPi 4 8 GB	RaspberryPi Foundation, Cambridge - GB
Cooler Master Gehäuse (Pi Case 40)	Cooler Master
Micro SD-Speicherkarte (64 GB)	SanDisk, Milpitas - USA
Netzteil 5,1 V/3,0 Ah	Raspberry Pi Foundation, Cambridge - GB
10,1" Display (LG LP101WH1)	LG Group, Seoul - ROK
<b>Software:</b>	
Betriebssystem RaspberryPi OS	RaspberryPi Foundation, Cambridge - GB
Spyder 3.3.3 und 4.0	Open source - MIT Licence
Datenbank - InfluxDB	InfluxData - Open source - MIT Licence
Webbrowser Mozilla Firefox	Mozilla Foundation Kalifornien - USA
Ubuntu 22.04	The Linux Foundation, GPLv2 Lizenz
Python Vers. 2.7.16 und 3.8.3	Python Software Foundation Licence
Python Pakete u. Bibliotheken: Pandas / Numpy / Matplotlib / Plotly / Time / CSV	

Gemäß der Abbildung 2.2 stellt die von der HS Osnabrück aufgesetzte Datenbank die gemeinsame Datenschnittstelle zwischen den Projektteilnehmenden von der HS Osnabrück und der RAM GmbH Mess- und Regeltechnik da.

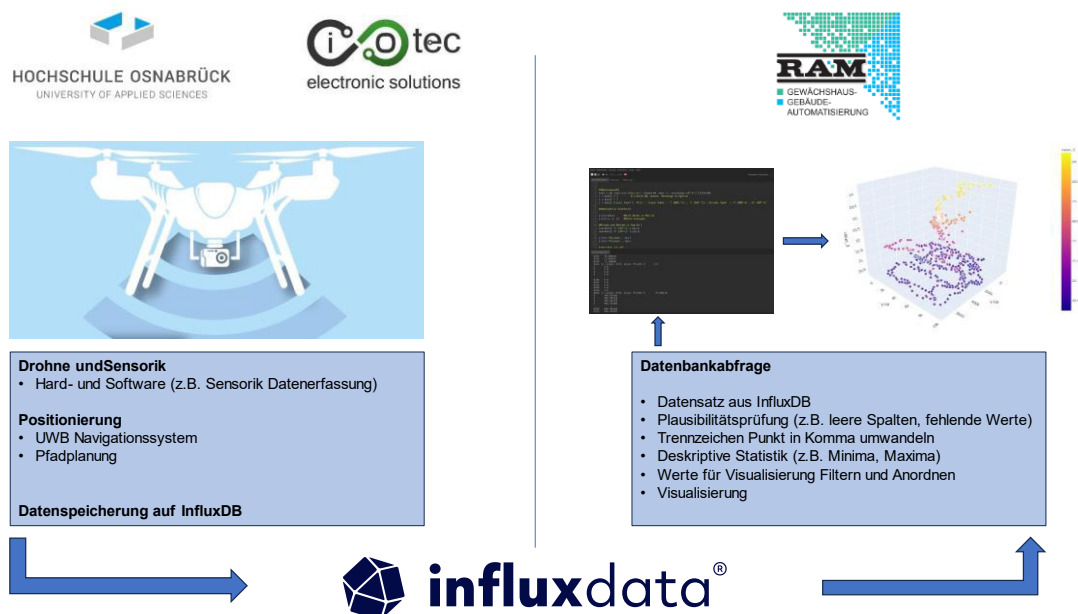


Abbildung 2.2 Aufgabenbereiche der Projektteilnehmenden (stark vereinfacht). Die InfluxDB dient als gemeinsame Schnittstelle.

In einem frühen Stadium der Projektbearbeitung wurden, nach vorangegangener, intensiver Recherche, die in Tabelle 2.2 genannten Komponenten beschafft, installiert und das System eingerichtet, um eine funktionsfähige Entwicklungsumgebung zu realisieren. Diese bestand hardwareseitig aus einem RaspberryPi in der Version 4 mit 8 GB RAM, mitsamt der weiteren, erforderlichen Peripherie (Tastatur, Netzteil, Monitor, etc.).

Softwareseitig diente RaspberryPi OS (Version buster) als Betriebssystem für den Einplatinencomputer. Die erforderlichen Python Pakete, die Entwicklungsumgebung (IDE) SPYDER (Vers. 3.3.3 und 4), sowie weitere erforderliche Softwareanwendungen wurden ebenfalls auf dem System installiert. Weiterhin wurde auch eine InfluxDB lokal aufgesetzt, allerdings auf einem anderem System (weiterer RaspberryPi) um den Datenzugriff auf eine ausgelagerte und entfernte Datenbank bestmöglich simulieren zu können. Diese Datenbank wurde im Folgenden zunächst mit manuell generierten Werten (Dummies) gefüllt. Ein Fernzugriff auf die Datenbank, die an der HS Osnabrück lokalisiert war, konnte bis zum Projektende aufgrund sicherheitstechnischer Bedenken seitens der Hochschule nicht realisiert werden.

Auf der so geschaffenen Infrastruktur basierend, wurden die erforderlichen Programme und Skripte für die Kommunikation, den Datenabruf und die folgende Visualisierung entwickelt. Vorkenntnisse im Umgang mit Python und einigen anderen eingesetzten Technologien, wie der InfluxDB, existierten nicht.

#### 2.1.1.1. Datenabfrage und Datenaufbereitung

Um die in der Datenbank abgelegten Daten im Folgenden zu nutzen und einer weiteren Verarbeitung zuzuführen, wurde die Entwicklungsumgebung (IDE) Spyder in der Version 3.3.3 (im weiteren Projektverlauf Version 4) genutzt. Als Programmiersprache wurde Python zunächst in der Version 2.7.16, im späteren Verlauf dann in der Version 3.8.3 genutzt. Der Umstieg von Python in der Version 2.7.x auf die Version 3.x führte zu einer notwendigen Anpassung sämtlicher bis dato entwickelter Skripte, da die Syntax der Programmiersprache in weiten Teilen geändert wurde. Dieser Schritt wurde notwendig, da die Weiterentwicklung an Python 2.x eingestellt wurde (Python - <https://www.python.org/doc/>).

Allgemein betrachtet bietet Python neben einer sehr umfangreichen Dokumentation auch geeignete Module und Bibliotheken für die Auswertung und Verarbeitung sehr großer Datensätze, sowie zur Erstellung anspruchsvoller Grafiken. Darüber hinaus existiert auch eine frei nutzbare Programmierschnittstelle (API) in Form der 'InfluxDB Python client library' (Python - <https://www.python.org/doc/>).

In regelmäßigen Abständen wurden von der HS Osnabrück angefertigte Datenbankdumps auf das System aufgespielt, sodass die am Versuchsort (HS Osnabrück) erfassten Daten (auch Testdatensätze) auch lokal zur Verfügung standen. Hierfür wurde das InfluxDB command line interface (InfluxDB CLI) genutzt:

```
> influx restore --host http://<serveraddress>:<influx_port> --org <organisation> -t <API_TOKEN> <path_to_bucket>/bucket/<
```

Unter Angabe der IP- Serveradresse und des zu verwendenden Ports (8086) konnte eine Verbindung etabliert werden. Zur Organisation der Struktur auf und Kommunikation mit der Datenbank müssen Organisationen (Arbeitsbereiche) definiert werden. Die Berechtigung zum Zugriff auf die eigentlichen Daten erfolgt über einen spezifischen `token`, die Speicherung der Daten erfolgt in einem `bucket`. Innerhalb eines `bucket` sind die Zeitreihendaten in logischen `measurements` (z.B. Temperatur, Luftfeuchte, etc.) zusammengefasst und gruppiert und können `tags` und `fields` enthalten, die zur Identifizierung und Charakterisierung der Werte gesetzt werden. Jeder in der Datenbank abgelegte Wert erhält einen mit ihm verbundenen `timestamp`. Tiefere Information zur InfluxDB sowie die vollumfängliche Dokumentation finden sich auf der Internetseite des Entwicklers (InfluxData - <https://docs.influxdata.com/influxdb/v2/>).

Das entwickelte Skript zur Datenabfrage und -Aufbereitung besteht im Wesentlichen aus vier aufeinander folgenden Programmzeilen. Im ersten Schritt werden die erforderlichen Module und Bibliotheken geladen:

```
>import datetime as datetime<  
>from influxdb_client import InfluxDBClient, Point<  
>import csv<
```

Im nächsten Schritt werden die Verbindungsparameter für die Kommunikation mit der Datenbank aufgebaut:

```
>url = "xxx.xxx.xxx.xxx:8086"<  
>token = "xxx"<  
>org = "xxx"<  
>bucket = "xxx"<
```

Im Hauptteil des Quellcodes erfolgt der eigentliche Zugriff auf die Datenbank. Dabei wird zunächst der client instanziiert:

```
>client = InfluxDBClient(url=url, token=token, org=org)<<  
>query_api = client.query_api()<
```

Die Datenabfrage erfolgt unter Angabe des gewünschten Zeitraums (`range`) und der dazugehörigen Messwerte (`measurements`). Über die `aggregateWindow`- Funktion werden die Daten in Zeitfenstern gruppiert (z.B. Ausgabe von Mittelwerten je Zeitintervall). Mit dem Befehl `drop` werden einzelne Spalten des zugrundeliegenden Datensatzes entfernt. Der Befehl `group` sortiert die Daten z.B. nach Zeit.

Die Parameter `latitude` und `longitude`, also die Daten die als Koordinaten zur Position der Messwerte herangezogen wurden, werden bei der Speicherung bereits umgerechnet.

```
>with client:<
>q=""<
>from(bucket:"xxx")<
>|> range(start: xxxx-xx-xxTxx:xx:xxZ, stop: xxxx-xx-xxTxx:xx:xxZ)<
>|> filter(fn: (r) => r["_measurement"] == "xxx" or r["_measurement"] == " xxx " or r["_measurement"] == " xxx " )<
>|> toFloat()<
>|> map(fn:(r) => ({ r with _value: r._value / 10000000.0 }))<
>|> aggregateWindow(every: 1000ms, fn: mean, createEmpty: false)<
>|> drop(columns: ["_xxx", "_xxx", "_xxx"])<
>|> group(columns: ["_xxx"])<
>|> yield(name: "last") ""<
```

Als finales Element des Skripts zur Datenabfrage und Aufbereitung werden die abgefragten Werte lokal als Textdatei im CSV- Dateiformat gespeichert. Dabei wird der Header der geladenen Tabelle ersetzt und die Daten sortiert und gruppiert, sodass zusammengehörige Daten (z.B. Messwerte eines Sensors) in Spalten angeordnet werden (Anlage 1). Dieser Schritt ist entscheidend, da die Daten in der Datenbank zeitlich aufeinanderfolgend abgelegt werden und keine Sortierung der Datenbankeinträge (z.B. Messgröße, Sensortyp, etc.) erfolgt. Dieser Schritt ermöglicht erst die automatisierte, weitere Nutzung des Datensatzes für Grafiken oder andere Auswertungsaufgaben.

```

>with open('xxx.csv', mode='w') as data:<
>data_writer = csv.writer(data, delimiter=';', quotechar='"', quoting=csv.QUOTE_MINIMAL)<
>data_writer.writerow(['xxx', 'xxx', 'xxx', 'xxx', 'xxx', 'xxx'])<
>for time, frame in df.groupby("_time"):<
>csvRow = []<
>csvRow.append(frame.iloc[0]["_time"].strftime("%Y-%m-%d %X"))<
>for index, row in frame.iterrows():<
>csvRow.append(row["_xxx"])<
>csvRow.append(row["_xxx "])<
>csvRow.append(row["_xxx "])<
>csvRow.append(row["_xxx "])<
>data_writer.writerow(csvRow)<

```

Die im CSV- Dateiformat gespeicherte Datei wird im folgenden Ablauf in ein weiteres Skript geladen, und einer deskriptiven Statistik unterzogen. Dabei werden neben der Ausgabe der Minima und Maxima der ausgewählten Spalten auch die Gesamtanzahl der Spalten und Zeilen verglichen. Insbesondere die Überprüfung der Anzahl der Spalten kann als einfache Plausibilitätsprüfung gewertet werden, da so z.B. das Fehlen von Werten direkt ersichtlich wird. Der Datensatz wird als Tabelle im pdf- Format gespeichert.

```

>import pandas as pd<
>data = pd.read_csv('xxx.csv', header=0, sep=';', encoding='utf-8').fillna(0)<
>x = data['xxx']<
>y = data['xxx']<
>z = data['(xxx)']<
>min=data['xxx'].max()<
>max=data['xxx'].min()<
>print(data)<
>print(x, y, z)<
>print('Maximum:', min)<
>print('Minimum:', max)<

```

### 2.1.1.2. Visualisierung

Das in 2.1.1.1 beschriebene Vorgehen führt als Resultat zu einem lokal abgespeicherten Datensatz (im CSV- Format), der alle relevanten Daten, mitsamt eines aktualisierten Headers, beinhaltet und im Weiteren verarbeitet und visualisiert werden sollen.

Zur Erstellung der interaktiven Grafiken kommt hier die Python- Grafikbibliothek `Plotly` zum Einsatz (Plotly – <https://plotly.com>). Zunächst werden die erforderlichen Bibliotheken geladen:

```
>import pandas as pd<
>import numpy as np<
>import matplotlib.pyplot as plt<
>import time<
>import csv<
>import plotly.express as px<
>from mpl_toolkits.mplot3d import Axes3D<
>from matplotlib.animation import FuncAnimation<
>from ipywidgets import interactive<
```

Darauffolgend wird im Skript die Datei, die die zur Visualisierung heranzuziehenden Daten enthält, ausgewählt. Es folgt die Auswahl der darzustellenden Wertepaare, wobei der X-Wert durch die geografische Länge ('longitude'), der Y- Wert durch die geografische Breite ('latitude') repräsentiert werden. Beide Werte entstammen der herstellerseitig auf der Drohne verbauten Sensorik. Als Z- Wert wird der Messwert des betreffenden Klimaparameters (externe Sensorik, z.B. Temperatur) genutzt. An dieser Stelle ist entscheidend, dass die CSV- Datei einen angepassten Header erhalten hat. Die dort verwendeten Überschriften werden in dem Skript zur Auswahl der für die Visualisierung gewünschten Parameter angegeben. Es ist dabei Essentiell, dass die Speicherung der Daten aus der Drohne stets im selben Format hinsichtlich der Datenanordnung abgelegt werden.

Für die grafische Darstellung der Messwerte wurde ein 3D- Streudiagramm (Scatterplot) auf Basis eines kartesischen Koordinatensystem gewählt, wobei die Ausprägung des Z- Wertes auch durch die jeweilige farbliche Füllung des Datenpunkts deutlich präsentiert wird.

Es wurden insgesamt zwei verschiedene Darstellungsweisen entwickelt, die auf verschiedenen Python- Modulen und Bibliotheken beruhen.

Das Einlesen der Daten läuft für beide Darstellungsformen identisch ab. Zunächst wird der Datensatz, in dem alle Messwerte gespeichert sind ausgewählt und die grafisch darzustellenden X-, Y-, und Z- Werte unter Angabe der im neuen Header definierten Spaltennamen festgelegt:

```
>data = pd.read_csv('xxx.csv', header=0, sep=';', encoding='utf-8').fillna(0)<
>x = data['xxx']<
>y = data['xxx']<
>z = data['xxx']<
```

Zur Darstellung der Messwerte in einem 3D- Scatterplot und zur Ausgabe als interaktive plot-ly- Grafik werden die Parameter des Scatterplots definiert:

```
>fig = plt.figure(figsize=(16,9) <
>ax = fig.add_subplot(111, projection='3d')<
>ax.scatter(x, y, z, c=z, cmap='coolwarm', marker='.', alpha=1, linewidth=.1)<
>ax.set_title('FlyingData',<
>...
ax.set_xlabel('X',<
>...<
ax.set_ylabel('Y',<
>...<
>ax.set_zlabel('Z',<
>...<
```

Definition der Legende mit Zuordnung der Messwerte:

```
>sc = ax.scatter(x, y, z, c=z, cmap="coolwarm")<
>...<
```

Einfügen eines Logos in den oberen linken Bildschirmrand:

```
>arr_img = plt.imread("logo.png")<
>im = OffsetImage(arr_img, zoom=.25)<
>ab = AnnotationBbox(im, (1, 0), xycoords='axes fraction', box_alignment=(2.76,-7.7))<
>ax.add_artist(ab)<
```

Speicherung der Grafik:

```
>plt.savefig('xxx.png', dpi = 300)<
>plt.show()<
```

In Abbildung 2.3. ist die Visualisierung als Resultat des oben beschriebenen Quellcodes, dargestellt. Die Grafik wird direkt in der IDE erzeugt (wahlweise auch als separat öffnendes Fenster) und ist auf dem Ausgabegerät interaktiv gestaltet. Sie kann mit einer angeschlossenen Maus, oder per Touchscreen- Eingabe, 360 ° rotiert werden. Zusätzlich kann in die Grafik hinein- und heraus- gezoomt werden.

Die Darstellung wird automatisch als png- Datei ('portable network graphics') gespeichert. Dieses Dateiformat wurde gewählt, da es verlustfrei ist und als das meistverwendete im Internet zählt und somit als Standard angesehen wird.

Figure 1

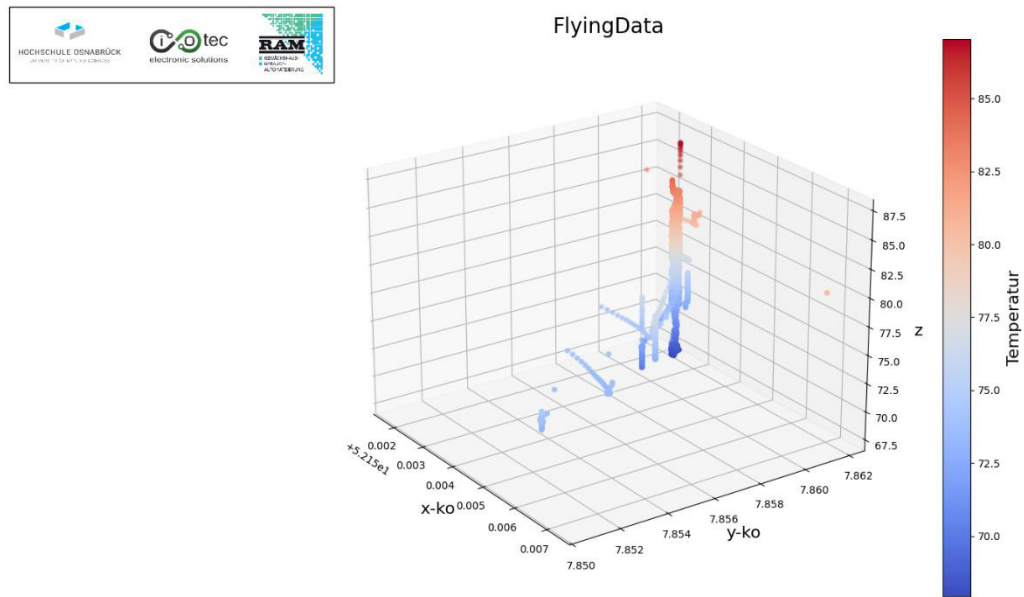


Abbildung 2.3 Screenshot der grafischen Ausgabe am Beispiel der relativen Luftfeuchte, gemessen mit dem Sensirion SHT-31 Sensor. Bei der Ausgaben in der IDE interaktiv (10.03.2023).

Für eine weitere Darstellungsform wurde unter Zuhilfenahme einer anderen, für 3D- Anwendungen optimierten Bibliothek (`plotly.express`) eine anspruchsvollere Darstellungsweise gewählt. Diese ermöglicht ebenfalls eine 3- Dimensionale Darstellung von der Messwerte:

```
>import plotly.express as px>
>fig3d = px.scatter_3d(data, x, y, z, color='xxx',
>...<
```

Die auf diesem Weg erzeugte Grafik wird nach Ausführung des Skripts direkt in dem auf dem jeweiligen System installierten Internetbrowser dargestellt (getestet mit Mozilla Firefox und Microsoft Edge). Die grafische Darstellung basiert auf der Programmiersprache JavaScript und stellt somit höhere Anforderungen an das System. Insbesondere in sicherheitsrelevanten Umgebungen wird JavaScript auf Grund der Manipulierfähigkeit von Internetseiten routinemäßig geblockt, wodurch allerdings auch die erzeugte Grafik blockiert werden würde.

Eine auf diesem Weg erzeugte Beispielgrafik ist der Abbildung 2.4 zu entnehmen. Um die 3D- Funktionalität zu erfahren ist diesem Bericht eine Beispieldatei beigefügt (Anlage 2).

Copterdata

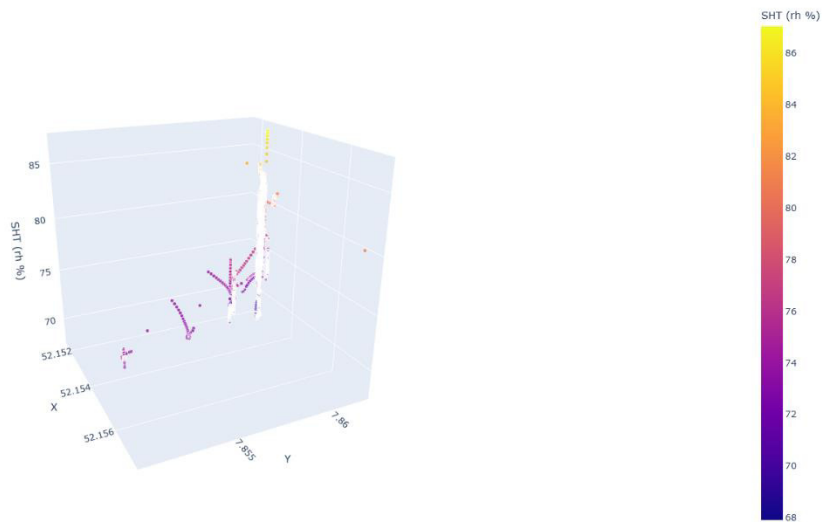


Abbildung 2.4 Screenshot der grafischen Ausgabe am Beispiel der relativen Luftfeuchte, gemessen mit dem Sensirion SHT-31 Sensor, während der Versuche im Praxisbetrieb (10.03.2023).

Zur Darstellung in verschiedenen Höhen gemessener Daten, z.B. in vertikalen Kulturen, ist die gewählte Darstellungsweise in einem scatterplot nicht geeignet. Die Daten werden in einem kartesischen Koordinatensystem wiedergegeben, wobei der 'Z- Wert ' der Ausprägung des Messwerts entspricht und zur besseren Darstellung zusätzlich mit einer Farbkarte unterlegt werden ('cmap= 'coolwarm'). Durch die Nutzung des 'Z-Wertes' zur Anzeige der Flughöhe, wurde die Grafik

sehr unübersichtlich, sodass diese Darstellungsweise verworfen wurde. Stattdessen wurde das Skript so angepasst, dass mehrere Grafiken gleichzeitig (eine Grafik je Flughöhe) erzeugt und in der IDE automatisiert ausgegeben werden. Die Grafiken werden automatisiert im png- Format gespeichert (Abb.2.5).

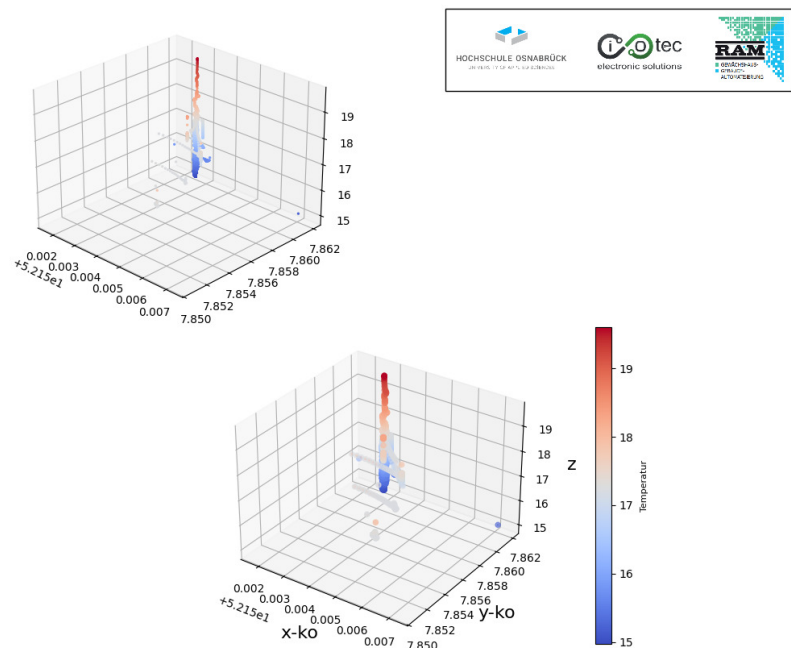


Abbildung 2.5 Darstellung mehrerer Grafiken in der IDE (interaktiv, plotly) und als png-Datei, gemessen in verschiedenen Höhen.

Zur Erstellung der Grafiken wurden die Zeiträume der jeweiligen Flughöhen verwendet. Dabei ist es essentiell, dass die Drohne in den betreffenden Zeiträumen auch ausschließlich die in der Pfadplanung hinterlegte Höhe abfliegt. In Abbildung 2.6 sind die von der werkseitig verbauten, internen Sensorik stammenden Flughöhen (in m ü NN) über die Zeit dargestellt.

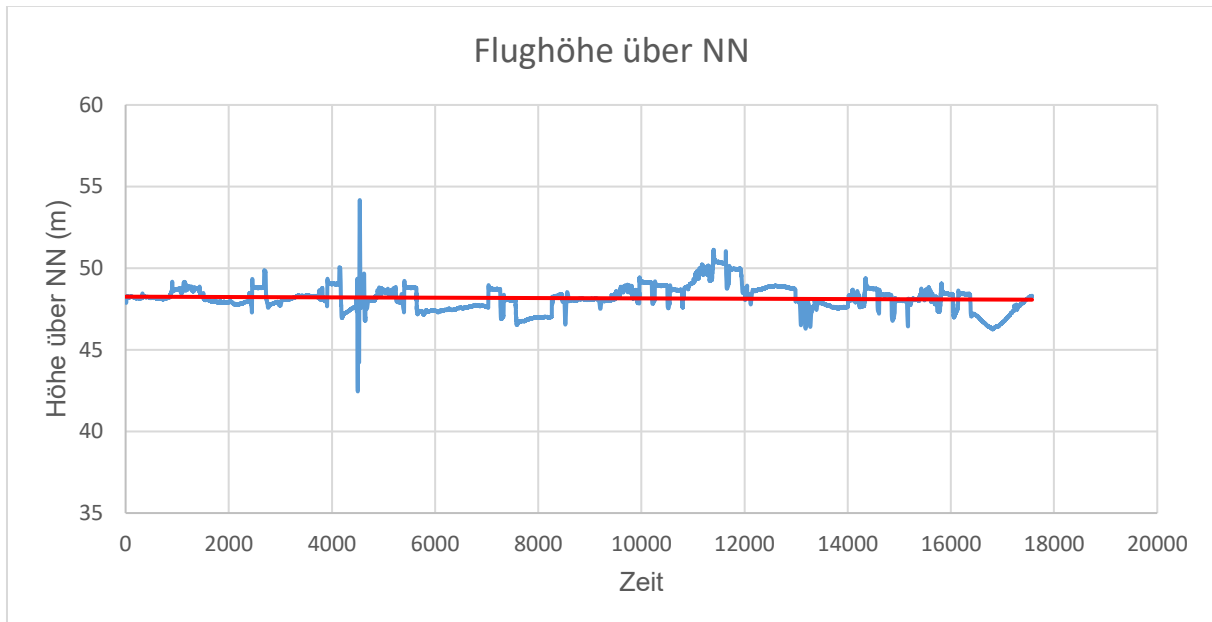


Abbildung 2.6 Daten der internen Sensorik zur Flughöhe

Anhand des Datensatzes, der unter Praxisbedingung am 10.03.2023 gewonnen wurde, konnte eine vereinfachte Kalkulation der Genauigkeit der Einhaltung der Flughöhe hergeleitet werden. Demnach betrug die Flughöhe im Mittel über alle Messwerte 48,16 m über NN (dargestellt als rote Linie in Abbildung 2.6). Die ermittelte Standardabweichung, als Maß für die Abweichung von der Flughöhe, betrug 0,78 m.

### 2.1.1.3. UI zur Visualisierung - Hardware

Zur Visualisierung der Daten im Feldeinsatz wurde eine Bedienkonsole entwickelt, die hardwareseitig zunächst auf Basis eines RaspberryPi realisiert wurde, und die sämtliche relevanten Programme (Programme, Module Bibliotheken, etc.) zur Kommunikation mit der Datenbank, Verarbeitung und Visualisierung der Daten, etc. beinhaltet (Siehe 2.1.1.2).

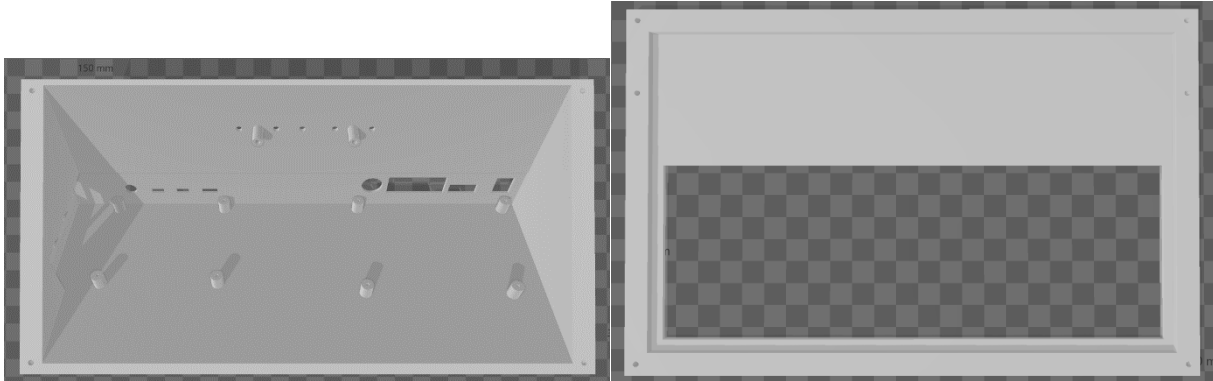


Abbildung 2.8 (links): Bild der STL- Druckdatei des Gehäuseinneren mit Aussparung und Halter für die Platinen

Abbildung 2.7 (rechts): Bild der STL- Druckdatei des Gehäuseoberteils zur Displayfixierung

Die Bedienung erfolgte über eine bluetoothfähige Tastatur mit Touchpad. Als Anzeige diente ein 10“ LCD-Display mit dazugehöriger Interfaceplatine. Auf ein touchfähiges Display wurde verzichtet, da erfahrungsgemäß die Touchbedienung auf einem 10“ Displays, ohne angepasste Programme, einen nur geringen Bedienkomfort bedingt. Die Druckdateien für das Gehäuse (Abb.2.7 und 2.8) wurden mithilfe des Programms FreeCad entwickelt, wobei die Maße der in das Gehäuse zu integrierenden Hardware und weiteren Komponenten in das CAD- Programm übertragen wurden. Mit einem zur Verfügung stehenden 3D- Drucker wurde das so konzipierte und weiterhin optimierte Gehäuse gedruckt. Die fertige und lauffähige Bedieneinheit ist in den Abbildungen 2.9 bis 2.11 aus verschiedenen Perspektiven abgebildet.

Die Bedieneinheit wurde getestet und in einem frühen Stadium der Projektbearbeitung genutzt. Für die weiteren Untersuchungen und im Hinblick auf einen praxistauglichen Einsatz zeigte sich, dass die zum Teil umständliche Bedienung und insbesondere das Fehlen eines Ak-

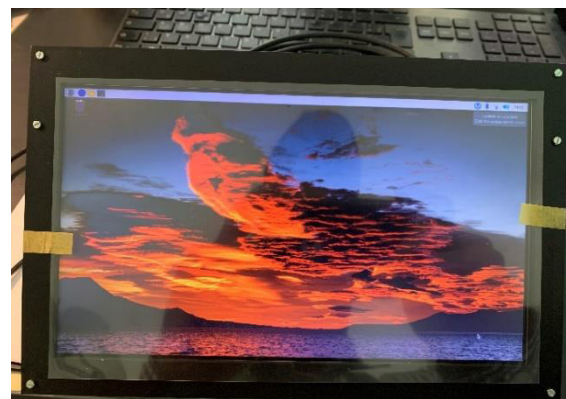


Abbildung 2.9 Frontalansicht Display mit Rahmen



Abbildung 2.10 Ansicht hinten mit Schnittstellen



Abbildung 2.11 Ansicht links mit Schnittstellen

kus für eine mobile Anwendung als nicht positiv zu bewerten war, sodass ein handelsübliches Tablet beschafft wurde. Um eine bestmögliche Kompatibilität bzw. Portierung der zur Visualisierung aufgebauten Softwarestrukturen und des entwickelten Quellcodes von dem RaspberryPi auf das neue System zu gewährleisten, wurde ein Gerät mit einem Intel- Prozessor beschafft. Nach vorangegangenen Recherchen weisen diese Prozessortypen eine sehr gute Kompatibilität mit Linux Distributionen auf. Das Windows Betriebssystem und die dazugehörige Systempartition wurde auf dem beschafften Gerät vollständig gelöscht und ein Linux basiertes Betriebssystem (Ubuntu) installiert.

#### 2.1.2. Mobiles hochaufgelöstes Messsystem für Temperatur und rel. Luftfeuchte

Basierend auf den Erkenntnissen aus dem Projekt Flying Data wurde parallel zur Datenabfrage und -Visualisierung ein mobiles Messsystem zur Erfassung der Klimaparameter Temperatur und relative Luftfeuchte in einem Pflanzenbestand (o.ä.) entwickelt. Diese Entwicklung wurde vorangetrieben, weil aus Sicht der RAM GmbH Mess- und Regeltechnik im späteren Projektverlauf ersichtlich wurde, dass mit Abschluss des Projekts FlyingData keine autonome Drohne zur Erfassung von Klimaparametern im Gewächshaus zur eigenen Verwendung (RAM) zur Verfügung stehen würde. Die Kosten und erforderlichen weiteren Entwicklungsarbeiten wären dabei zu hoch und stünden nicht in einem angemessenen Kosten- Nutzen- Verhältnis. Gleichwohl ist der Bedarf an hochaufgelösten Klimadaten im Bestand, mit dem Ziel des Monitorings, gegeben. Durch die Eigenentwicklung entstehen signifikante Vorteile zu marktverfügbaren Messsystemen:

- Soft- und Hardwareentwicklung im eigenen Haus
- Kommunikation mit RAM- Betriebssystem
- deutlich geringere Kosten
- flexible Anpassung z.B. in der Anzahl der Sensoren an die Betriebsstruktur
- Aufzeichnung der Messwerte in einer Datenbank mit theor. unendlicher Laufzeit

Entscheidend für die Realisierung eines eigenen Messsystems war die Vielzahl bereits zuvor für den Datenzugriff und die Datenvisualisierung entwickelter Skripte. Durch entsprechende Anpassungen konnten diese auch für diesen Einsatzzweck genutzt werden. In Tabelle 2.3 sind die verwendeten Soft- und Hardwarekomponenten aufgeführt.

Der eingesetzte ESP32- Microcontroller lieferte bereits im Einsatz auf der Drohne gute Ergebnisse und ist durch die Vielzahl frei konfigurierbarer Digital-, Analog, Ein- und Ausgänge ein für diesen Einsatz geeignetes Gerät. Kenntnisse im Umgang mit dem Microcontroller existierten nicht und mussten zusätzlich erworben werden.

Tabelle 2.3 Verwendete Soft- und Hardwarekomponenten zur Realisierung des hochaufgelösten mobilen Messsystems

Bezeichnung	Hersteller/Lizenz
<b>Hardware:</b>	
ESP 32 (XIAO ESP32C3)	Seeed Technology Co., Ltd Shenzhen-CB
Arduino Temp. und rF (SHT 31)	Freie Hardware / LGPL/GPL Lizenz
Li-Ion Akku (3,7 V 3000 mAh)	Murata Seisakusho, Nagaokakyo, JP
Gehäuse designed in Freecad	FreeCAD, LGLPLv2 Lizenz
Physischer An/Aus Schalter	
Raspberry Pi 4 8 GB (Datenbank)	Raspberry Pi Foundation, Cambridge - GB
<b>Software:</b>	
InfluxDB	Open source - MIT Licence
Arduino IDE Pakete / Bibliotheken:	

Die installierten Komponenten und das fertige Messsystem sind in den Abbildungen 2.12 bis 2.14 dargestellt.

Das Case für den entwickelten Prototyp wurde in der Farbe blau gedruckt (Abb. 2.12 und 2.13). Um die Reflexionseigenschaften zu verbessern und so eine (zusätzliche) Erwärmung durch die Gehäuse zu minimieren, wurden die finalen Gehäuse mit weißem Filamet gedruckt.

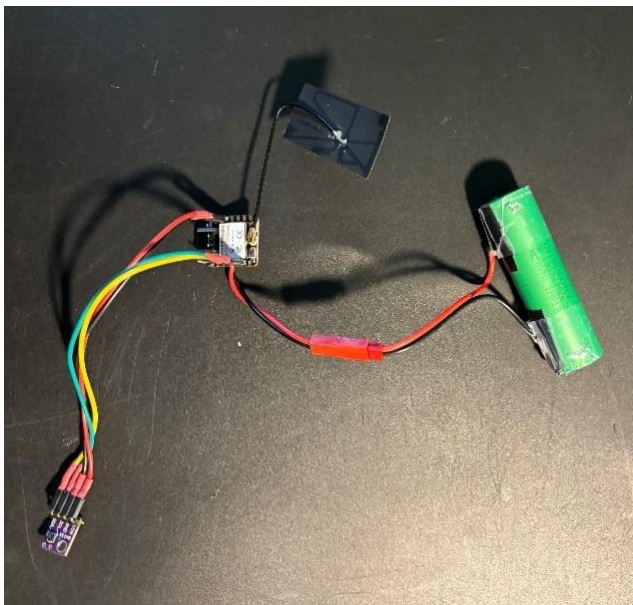


Abbildung 2.12 Elektronikkomponenten

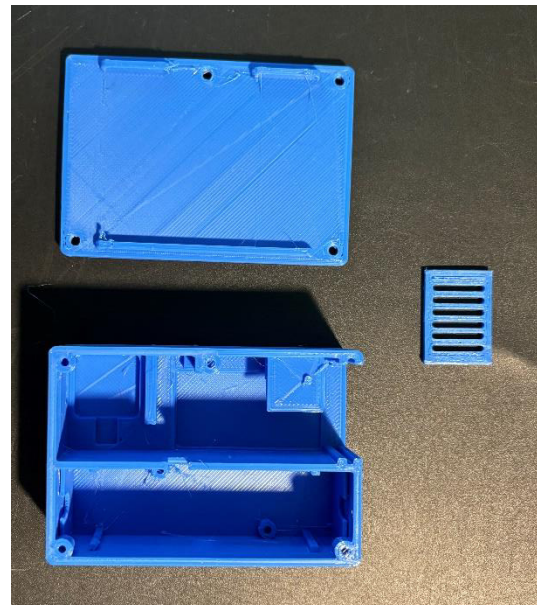


Abbildung 2.13 Gehäuse 3D- gedruckt

Für den Testbetrieb wurden insgesamt drei Geräte fertiggestellt und in einer Versuchsanordnung genauer untersucht. Dabei erfassten die Sensoren in einem nicht klimatisierten Raum, über einen Zeitraum von 5 Wochen, die Faktoren Temperatur und relative Luftfeuchte. Das Messintervall betrug 10 Sekunden. Die Geräte wurden zur Sicherstellung der optimalen Spannungsversorgung netzbetrieben getestet.

Aus dem erhaltenen Datensatz wurde ein Mittelwert über die Daten einer Zeile (Daten zum selben Messzeitpunkt) und daraus wiederum die Standardabweichung, als Maß

für die Streuung um den Mittelwert, am jeweiligen Messpunkt, gebildet. Schließlich wurde ein Mittelwert der Standardabweichung über die gesamte Versuchsdauer (120000 Einzeldaten) gebildet und als Maß für die Abweichung zu den Herstellerangaben (Adafruit.com / Anlage 3) herangezogen (Tab. 2.4).

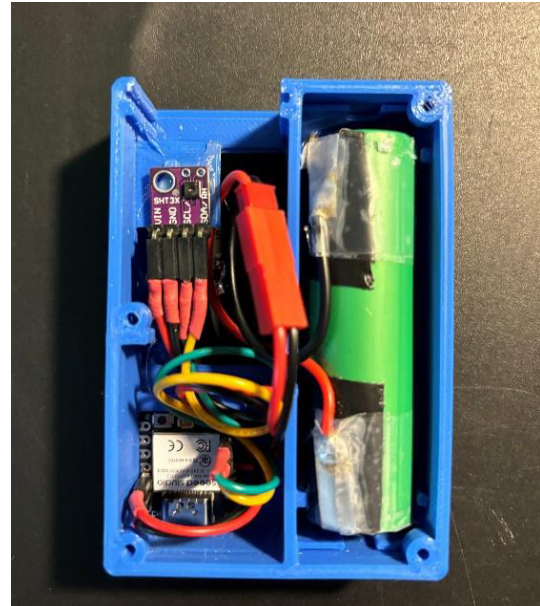


Abbildung 2.14 Gehäuse mit Elektronikkomponenten

Tabelle 2.4 Vergleich der Herstellerangaben mit gemessenen Werten

Parameter	Herstellerangabe	MW der Standardabweichung
Temperatur	$\pm 0,3 \text{ }^\circ\text{C}$	$\pm 0,09 \text{ }^\circ\text{C}$
Rel. Luftfeuchte	$\pm 2 \text{ } \%$	$\pm 2.2 \text{ } \%$

Es konnte gezeigt werden, dass die Streuung unter den eingesetzten Sensoren hinsichtlich der Temperatur geringer war, als vom Hersteller angegeben. Im Hinblick auf den Parameter rel. Luftfeuchte konnte eine über die Herstellerangaben hinausgehende Ungenauigkeit von 10 % ermittelt werden. Für Sensoren in dieser Preisklasse (Kosten pro Sensor ca. 13 €, UVP Hersteller) stellt das Ergebnis der relativen Genauigkeit der Sensoren untereinander einen guten Wert dar. Die absolute Genauigkeit wurde über die Langlebigkeit und Stabilität über einen längeren Zeitraum und unter Praxisbedingungen (z.B. im Gewächshaus) kann an dieser Stelle keine Aussage getroffen werden.

Das System soll nach Projektende weiterentwickelt und auch erweitert werden. Zudem sollen einzelne ausgewählte Betriebe mit der Sensorik ausgestattet werden um Erkenntnisse zur Langzeitstabilität und Verlässlichkeit zu erlangen.

Weitere durchzuführende Untersuchungen und notwendige Updates:

- Langzeitstabilität
- Einsetzbarkeit unter Praxisbedingung (z.B. Gewächshaus)
- Implementierung weiterer Sensoren
- Gehäuse aus dem 3D- Drucker aus ABS statt PLA
- Implementierung eines `Deep- Sleep- Modus` für eine längere Akkulaufzeit

Neben den verwendeten Sensoren zur Erfassung der Temperatur und relativen Luftfeuchte wurde auch der Einsatz weiterer Sensoren (Lichtsensoren GY-49 und TEMT6000) untersucht, wobei sich hier einige Schwierigkeiten und Schwachstellen zeigten. Mit Blick auf Messbereich, Reproduzierbarkeit und Genauigkeit der angezeigten Messwerte erscheinen die verwendeten Sensoren ungeeignet. Auch der Ausgabewert beider Sensoren in der Einheit LUX, stellt im Hinblick auf Messungen in einem Pflanzenbestand eine unpassende Einheit dar.

### 2.1.3. Anforderungen zur Nutzung hochaufgelöster Daten für die Klimaregelung

Wie in Abschnitt 2.1.1 und 2.1.2 gezeigt, können die Daten sehr gut genutzt werden, um eine präzise Abbildung des Ist- Zustandes der Verteilung der Klimaparameter in einem Gewächshaus zu erstellen und diese grafisch darzustellen.

Über die Visualisierung hinaus, sollte seitens der RAM GmbH Mess- und Regeltechnik auch die Frage geklärt werden, ob die Datenerfassung für Regelungs- und Automatisierungsprozesse in Gewächshäusern zukünftig auch durch den Einsatz autonomer Drohnen unterstützt werden könnte. Der Bedarf an hochaufgelösten Daten, für ein Monitoring, zur Unterstützung der Regelung oder zur Entwicklung angepasster Klima- Regel- Strategien ist gegeben, da standardmäßig nur ein Sensor in Verwendung kommt und es dadurch keine Möglichkeit gibt die tatsächliche Situation in Gänze zu erfassen.

Die einfachste Form der Implementierung wäre ein Hinweis, z.B. beim Auftreten von lokalen Temperatur- oder Luftfeuchtegradienten, auf der grafischen Oberfläche des RAM- Betriebssystems (VisuRAM) und/oder eine Alarmmeldung per E-Mail mit der betreffenden Information. Diese könnte dann, entsprechend der Priorität, quittiert werden müssen. Nutzende könnten dann reagieren und manuell Gegenmaßnahmen einleiten.

Ein weitere denkbare Option ist die Entwicklung einer speziellen Software (stand alone), z.B. zur Erfassung des Botrytis- Risikos, deren grundlegende Funktionsweise zur Möglichkeit des gezielten Einsatzes zur Botrytis- Erkennung und -Bekämpfung zusammen mit den Projektpartner\*innen von der HS Osnabrück diskutiert wurde. Dabei könnten die Daten aus der

Drohne (insbesondere die rel. Luftfeuchte) genutzt werden, um anhand mathematischer Modelle die Botrytis- Gefahr abzuschätzen. Als mögliche schnelle Gegenreaktion könnte die Drohne lokal, an der prognostizierten Stelle, durch eine Verwirbelung mit den Rotoren die Luftmassen verteilen und so die Luftfeuchte absenken.

Zur weiteren Verwendung der Daten im RAM Klima- Regel- System muss zunächst eine soft- und hardwareseitige Schnittstelle geschaffen werden. Mögliche Wege, sowie der zu erwartende Entwicklungsaufwand wurden bei der RAM GmbH Mess- und Regeltechnik intern erörtert und am Beispiel eines möglichem Botrytis- Regelmoduls beschrieben:

Das modellierte Botrytis- Risiko (zur Modellerstellung ist zwingend eine Zusammenarbeit mit einer Hochschule erforderlich) könnte, basierend auf den gelieferten Messwerten und dem zuvor erstellten, mathematischen Modell skaliert und als Wert in der InfluxDB abgelegt werden. Mithilfe einer InfluxDB- Softwareerweiterung (Telegraf) können die Daten aus der InfluxDB zu einem Modbus Master gesendet werden. Modbus gilt als Standard bei Industrieanwendungen und die RAM GmbH Mess- und Regeltechnik verfügt grundsätzlich über Möglichkeiten und Techniken die Modbus- Datenpunkte (Messwerte) auf den Klimacomputer RAM CC660 zu Übertragen und dort auszuwerten. Die Umsetzung des Datentransfers über das Kommunikationsprotokoll Modbus ist essentiell, da der RAM CC660 über ein eigenes, von der RAM GmbH Mess- und Regeltechnik entwickeltes, Betriebssystem verfügt und werksseitig aufgrund von Sicherheitsaspekten keinen Datenimport zulässt. Die erforderlichen Softwaremodule, sowie die Entwicklung einer zwei- Wege- API zur bidirektionalen Kommunikation und Wertweitergabe müssten ebenfalls entwickelt werden.

Hardwareseitig verfügt die RAM GmbH Mess- und Regeltechnik über Hardwarekomponenten (Modbus DDC) die mit dem RAM eigenen Betriebssystem kommunizieren und zur Datenweitergabe eingesetzt werden können. Der grundsätzliche Aufbau könnte wie in Abbildung 2.15 gezeigt, realisiert werden.

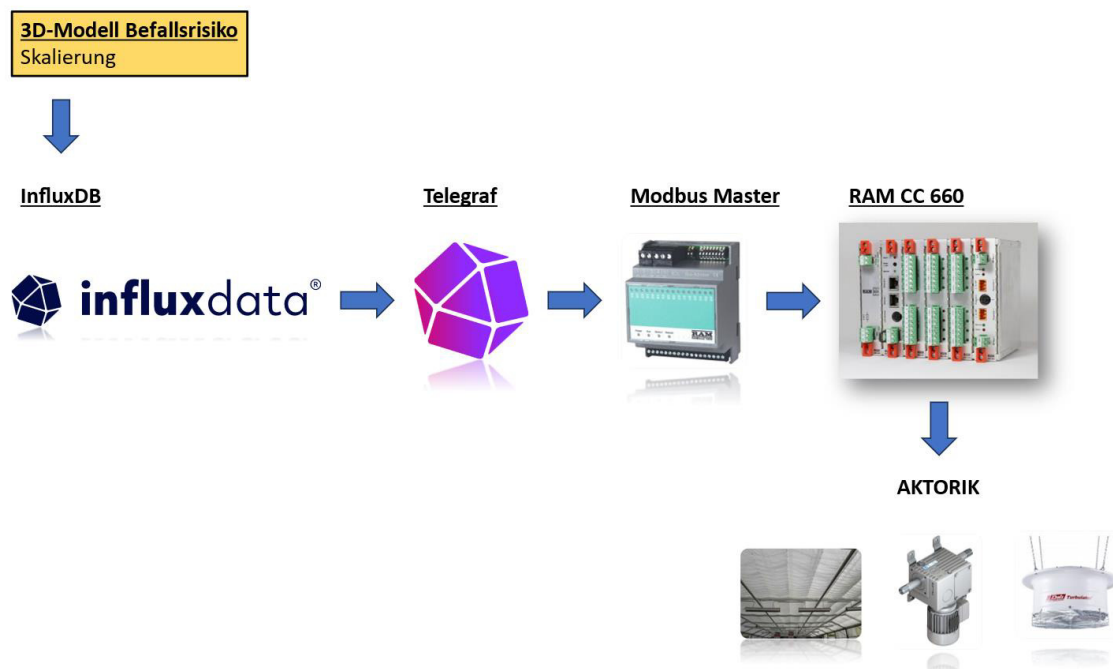


Abbildung 2.15 Aufbau der Kommunikation zur Weiterleitung der Daten aus der InfluxDB an den RAM CC660

Um insgesamt einen signifikanten Mehrwert mithilfe einer autonom fliegenden Drohne gewährleisten zu können, sind weitere, über dieses Projekt hinausgehende, Anstrengungen erforderlich. Maßgeblich für einen solchen Einsatz sind die Nutzendenfreundlichkeit und die Verlässlichkeit des eingesetzten Systems. Dazu zählen in besonderem Maße die Langzeitstabilität des Systems, die Datenerfassung mitsamt der Qualität der gespeicherten Daten und die Sensorqualität. Ein weiterer Punkt für einen potentiellen Einsatz ist die maximal mögliche Flugzeit der Drohne. Eine deutliche Verbesserung, die gleichzeitig für einen späteren Praxiseinsatz unabdingbar ist, kann durch die Nutzung einer Ladestation (siehe Schlussbericht HS Osnabrück) für automatisierte Ladeprozesse erzielt werden. Nur periodisch erfasste Werte oder Zusatzinformation leisten einen nur sehr begrenzten Mehrwert für die Regelung

## **2.2. Wichtigste Positionen des zahlenmäßigen Nachweises**

Die Verwendung der bewilligten Mittel erfolgte antragsgemäß hauptsächlich zur Deckung der Personalkosten (Position 0837). Nur ein sehr geringer Anteil der Bewilligungssumme (0,45 %) entfiel auf die Position Material (0813). Durch den Wegfall eines Industrie PC, wie bei der Antragstellung angenommen, konnten ca. 43 % der Bewilligungssumme aus Position 813 umgewidmet werden.

## **2.3. Notwendigkeit und Angemessenheit der geleisteten Arbeit**

Zur Erfüllung der Ziele (Arbeitspakete mitsamt Meilensteinen), basierend auf den eingereichten Anträgen (Erst- und Aufstockungsantrag), waren die geleisteten Arbeiten aus Sicht der RAM GmbH Mess- und Regeltechnik notwendig und angemessen.

Zum Start des Projekts FlyingData existierten keine nennenswerten Fähigkeiten im Umgang mit den Programmiersprachen Python und C, mit einer InfluxDB und mit einem ESP-Microcontroller und der dazugehörigen Sensorik, sodass diese erst mit der Projektbearbeitung erworben werden mussten. Insbesondere bei auftretenden Problemen in der Syntax musste eine zum Teil langwierige Fehlersuche betrieben werden.

## **2.4. Voraussichtlicher Nutzen, insbesondere der Verwertbarkeit des Ergebnisses im Sinne des fortgeschriebenen Verwertungsplans**

Im Rahmen des Projekts FlyingData wurde der Einsatz autonomer Drohnen, zur Erfassung relevanter Klimaparameter, erfolgreich untersucht. Komplexe Soft- und Hardwareelemente wurden entwickelt und (u.a.) zur Visualisierung der Messwerte aus der Drohne herangezogen. Ob zukünftige Einsätze einer autonomen Drohne, basierend auf dem aktuellen Stand und ohne weitere Untersuchungen, realisiert werden, kann aktuell nicht garantiert werden.

Die eingesetzte Technik und dabei insbesondere die Python basierten Programme, lassen sich durch entsprechende Anpassungen auf andere Problemstellungen (z.B. durch Implementierung weiterer Sensoren) übertragen und liefern damit einen Grundstein für ein hoch anpassbares und modulares Messsystem zur Erfassung und zur grafischen Darstellung weiterer relevanter Klimaparameter. Dabei wurden auch die technischen Anforderungen zur Weitergabe der Zusatzinformation an das RAM Klima- Regel- System diskutiert und beispielhaft projiziert.

Die Weiterentwicklung des mobilen, hochauflösenden Messsystems soll vorangetrieben werden und kurz- bis mittelfristig die Informationsvielfalt zunächst als stand- alone Lösung erhöhen.

## **2.5. Während der Durchführung des Vorhabens dem ZE bekannt gewordenen Fortschritts auf dem Gebiet des Vorhabens bei anderen Stellen**

Es sind keine weiteren Vorhaben bekannt, die vergleichbare Konzepte oder Systeme zur Erfassung hochaufgelöster Klimadaten im Unterglasanbau durch den Einsatz autonomer Drohnen entwickeln.

## **2.6. Erfolgte oder geplante Veröffentlichungen der Ergebnisse**

Im Rahmen der IPM Essen (2020 und 2024) wurden die Ergebnisse der Mitarbeit am Projekt FlyingData dargestellt. Zur Präsentation wurden Rollups und Handouts (Anlage 4) angefertigt und diese auf dem Messestand ausgestellt. In diesem Rahmen fanden auch Vortragsreihen statt.

Auf der Jahrestagung der Deutschen Gartenbauwissenschaftlichen Gesellschaft (DGG) im Jahr 2023 trat die RAM GmbH Mess- und Regeltechnik als Hauptsponsor auf. Bei der Veranstaltung wurden Poster und Handouts präsentiert.

Der jeweils aktuelle Stand der Untersuchungen wurde bei Vorträgen der RAM GmbH Mess- und Regeltechnik unter dem Punkt „Aktuelle Forschungsprojekte“ kommuniziert (Anlage 5).

Vorträge mit einem ausgeprägten inhaltlichen Bezug zum Projekt FlyingData:

- 18.01.2022:

Agrobusiness Niederrhein / Online

- 18.05.2022:

Bundetagung Zierpflanzenberatung (ZVG) / Online

- 18.10.2022:

BLE Innovationstage / Präsenz

- 01.03.2023 – 03.03.2023

DGG Osnabrück / Präsenz

- 06.02.2023 und 19.02.2024:

Staatliche Lehr- und Versuchsanstalt für Gartenbau (LVG) / Online

- 2024

YouTube - Autonomous Quadcopter for Greenhouses - University of Applied Sciences

<https://www.youtube.com/watch?v=-g6vvM-6mLg>

## 2.7. Referenzen

KTBL 2008: Bericht zur Bestimmung und Bewertung des Energiebedarfs von Gewächshäusern. 1. Aufl. – Kuratorium für Technik in Bauwesen und Landwirtschaft e.V.

### Internetquellen:

Adafruit: Herstellerseite Temperatur/Luftfeuchtesensor [online/Zugriff 21.05.2024]

[https://cdn-shop.adafruit.com/product-files/2857/Sensirion\\_Humidity\\_SHT3x\\_Datasheet\\_digital-767294.pdf](https://cdn-shop.adafruit.com/product-files/2857/Sensirion_Humidity_SHT3x_Datasheet_digital-767294.pdf)

BMEL 2018: Richtlinie über die Förderung von Innovationen für einen Gartenbau 4.0 im Rahmen des Programms zur Innovationsförderung [online/Zugriff 21.03.2024]

[https://www.ble.de/SharedDocs/Downloads/DE/Projektfoerderung/Innovationen/BMEL/180821\\_Gartenbau.pdf?\\_\\_blob=publicationFile&v=2](https://www.ble.de/SharedDocs/Downloads/DE/Projektfoerderung/Innovationen/BMEL/180821_Gartenbau.pdf?__blob=publicationFile&v=2)

InfluxData: Get started with InfluxDB V2 [online/Zugriff 29.05.2024]

<https://docs.influxdata.com/influxdb/v2/>

Plotly: Plotly. [online/Zugriff 18.03.2024]

<https://plotly.com/>

Python: Python Docs. [online/Zugriff 18.03.2024]

<https://www.python.org/doc/>

#### **4. Kurzfassung**

Ein vom Bundesministerium für Ernährung und Landwirtschaft (BMEL) gefördertes Forschungsprojekt, mit dem Akronym FlyingData (FKZ: 2818507B18) wurde innerhalb eines Projektkonsortiums bearbeitet.

Ziel war es, die Einsatzbarkeit autonom fliegender Drohen im Einsatz im Unterglasanbau zu untersuchen. Dabei sollten die Drohen entlang eines festgelegten Pfades fliegen und hochaufgelöst relevante Klimadaten aufzeichnen, sodass die gewonnenen Daten für eine detaillierte Abbildung des Ist- Zustandes und zur Unterstützung bei der Klimaführung eingesetzt werden können.

Es konnte gezeigt werden, dass die mit Zusatzsensorik und Microcontroller ausgestatteten Drohnen Messdaten akkurat erfassen und diese in einer Datenbank ablegen (HS Osnabrück). Auch die Pfadplanung und Orientierung der Drohnen unter Laborbedingungen sowie im Praxisbetrieb wurden final entwickelt (iotec GmbH). Auf Basis der gespeicherten Daten wurde ein stand- alone System (Soft- und Hardware) entwickelt und zur Nutzbarmachung und Visualisierung der Messwerte genutzt. Darüber hinaus wurde ein mobiles 3D- Messsystem entwickelt (RAM GmbH Mess- und Regeltechnik). Eine direkte Nutzung im RAM Klima- Regel- System fand nicht statt.

## **Abstract**

A research project funded by the Federal Ministry of Food and Agriculture (Bundesministerium für Ernährung und Landwirtschaft - BMEL) with the acronym FlyingData (FKZ: 2818507B18) was carried out within a project consortium.

The aim was to investigate the usability of autonomous flying drones for use in greenhouse cultivation. The drones were to fly along a defined path and record relevant climate data in high resolution so that the data obtained can be used for a detailed mapping of the actual condition and to support climate control.

It was shown that the drones equipped with additional sensors and microcontrollers accurately record measurement data and store it in a database (Osnabrück University of Applied Sciences). The path planning and orientation of the drones under laboratory conditions and in practical operation were also finalised (iotec GmbH). A stand-alone system (software and hardware) was developed on the basis of the stored data and used for the utilisation and visualisation of the measured values. In addition, a mobile 3D measuring system was developed (RAM GmbH Mess- und Regeltechnik). It was not used directly in the RAM climate control system.