Aerostat Systems

AlfaOPS Command & Control (C2) Software Development

Scope of Work

Total Page Numbers								DOCUMENT NO	DATE	REV
Cover	Document	ANNEXES								
xiv	162	A	B	C				A6RAAOSOW101	1st September 2024	2.0.0

Contents

1. Introduction ... 4
2. Legacy System ... 5
2.1. Aerostat System Components: ... 5
2.2. Components Pictures: ... 6
2.3. Avionics Suite: ... 7
2.3.1. Components of the Avionics Suite: ... 7
2.3.1. Interconnectivity and Communication: ... 9
2.3.2. Operational Capabilities: ... 9
3. Data Lifecycle and Integration ... 10
3.1. Data Generation ... 10
3.2. Data Conversion and Transmission ... 10
3.3. Data Routing and Control ... 11
3.4. Data Reception and Processing at the GCS ... 11
3.5. Data Archival and Security ... 11
4. Ground Control Station (GCS) ... 12
4.1. Utility Vehicle ... 12
4.2. GCS Block Diagram: ... 13
4.3. GCS Displays ... 14
5. GCS Telemetry Software ... 15
5.1. Confluence Point Tension Display ... 16
5.2. Heading and Wind Angle Display ... 16
5.3. Wind Speed Display ... 16
5.4. Temperature Displays ... 16
5.5. Barometric Pressure Displays ... 16
5.6. Hull Helium and Ballonet Air Pressure Displays ... 17
5.7. Aerostat Heading Display ... 17
5.8. Aerostat Attitude Displays ... 18
5.9. Data Logging Display ... 18
5.10. Blower/Lights/Radio Link Panel ... 19
5.11. Recent Screenshot of the Interface ... 19
6. Operator Console Assessment: ... 20
6.1. Hardware Overview ... 20
6.2. System Configuration ... 21
6.3. Software Analysis ... 22
6.3.1. Key Software Directories: ... 22
6.3.2. Software Characteristics: ... 22
6.4. Script Analysis ... 22
6.4.1. Script Overview: ... 22
6.4.2. Analysis: ... 22
6.5. Device Configuration ... 23
6.5.1. Moxa NPort Overview ... 23
6.5.2. Moxa Serial Settings ... 24
6.5.3. Operating Settings ... 25
6.5.4. Line Monitoring and Async Settings ... 26

6.5.5. Configured IPs and Ports 26
6.6. Data Output 27
6.6.1. Heading Sensor Output (Digital Compass HMR3000) 27
6.6.2. Pressure Sensor Output (Helium and Ballonet Pressure) 27
6.6.3. 3. Weather Station Output (Airmar PB200) 27
6.7. Launching of Legacy Telemetry Software 28
6.7.1. Key Shell Scripts and Execution Paths 28
6.7.1.1. Script: aerostatgcs 28
6.7.1.2. Script: TetherGCS 29
6.7.1.3. Script: RadioGCS_new 30
6.7.1.4. Script: TetherGCS_new 31
6.8. Relevant Directories and Executables 32
7. Problem Statement 33
8. Scope of Work 35
8.1. Overview 35
8.2. Agile Project Management Approach 36
8.2.1. Build 1: AlfaOps on Legacy Hardware 36
8.2.2. Build 2: Updated Hardware and Sensors 36
8.2.3. Build 3: Advanced Features and AI Integration 36
8.3. Key Deliverables 37
8.4. Technical Requirements 37
8.5. Functional Requirements 37
8.6. Non-Functional Requirements 38
8.7. Quality Requirements 39
8.8. Key Performance Indicators (KPIs) 40
8.9. Implementation Timeline 40
8.10. Acceptance Criteria 40

1. Introduction

This document outlines the Scope of Work for the development of the AlfaOps Command and Control software for aerostat systems by Alfa 6 LTA Systems. AlfaOps is designed to replace the existing legacy system used in the ASADI A200 Aerostat System, with the aim of enhancing operational capabilities, improving user interface experiences, and integrating modern technological advancements to meet current and future operational needs.

The primary objective of this document is to provide a comprehensive overview of the legacy ASADI A200 Aerostat System, identify the functional and non-functional requirements for the new Alaos software, and define the scope, deliverables, and key milestones for the software development project. This document will serve as a guideline for all stakeholders involved in the project, ensuring a smooth transition from the legacy system to the new, advanced AlfaOps platform.

2. Legacy System

The ASADI A200 Aerostat System is a tethered, non-rigid, aerodynamically shaped balloon system used for surveillance and communication relay missions. It operates at an altitude of 500 feet (150 meters) and consists of several key components, including the aerostat itself, mission equipment, a base unit, a Ground Control Station (GCS), a Remote Video Terminal (RVT), and a support vehicle.

The ASADI A200 is a fully self-contained system that provides all necessary support equipment and launch/recovery facilities on a trailer that can be towed by a utility vehicle. The aerostat is designed with an inverted "Y" empennage and is constructed from specialized fabrics that offer high strength, low weight, durability, and superior gas retention properties. Helium is used as the lifting medium, and the aerostat is secured to a base unit via a tether.

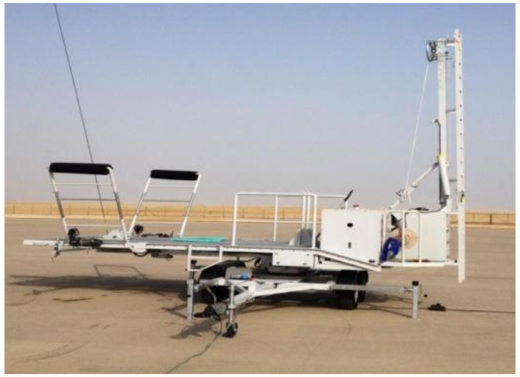
2.1. Aerostat System Components:

Key components of the system include:

1. **Aerostat:** A non-rigid balloon designed for high stability and durability, constructed with specialized fabrics.
2. **Avionics Suite:** Includes all sensors, electronics, telemetry equipment, and harnesses necessary for aerostat operation, as well as their interconnectivity.
3. **Payload:** Mounted at the confluence point using a payload bar, enabling the aerostat to fulfil its surveillance and communication roles.
4. **Tether:** Provides power and data connectivity between the aerostat and the ground-based equipment. It incorporates single-mode fiber optics and ADAM 6542/W15 Ethernet-to-fiber converters for reliable data transmission.
5. **Base Unit (Mooring Station):** Serves as the anchor point for the aerostat, providing stability and housing the winch and control mechanisms required for launching, mooring, and retrieving the aerostat. The base unit includes:
6. **Ground Control Station (GCS):** Housed in a modified utility vehicle, the GCS contains all controls and displays necessary for operating the aerostat and monitoring its payloads.
7. **Remote Video Terminal (RVT):** A portable station that allows for real-time viewing of payload sensor video up to a distance of 5 km.
8. **Support Vehicle:** A trailer unit that carries the aerostat and associated support equipment, ensuring mobility and ease of transport.

This document will primarily focus on the Avionics Suite and the Ground Control Station. This will assurance that the required information for AlfaOPS software developed is through and available.

2.2. Components Pictures:

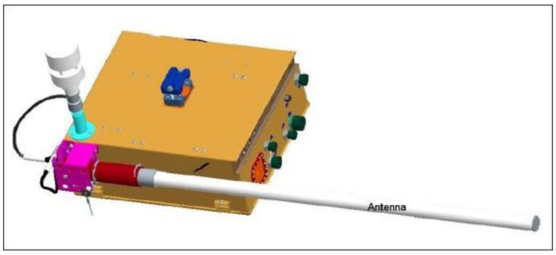


2.3. Avionics Suite:

The Avionics Suite of the ASADI A200 Aerostat System is designed with the philosophy of converting all sensor and telemetry signals into UDP/IP for efficient routing to the Ground Control Station (GCS). This ensures streamlined communication and data integration between the aerostat and the GCS, allowing for real-time monitoring and control of the system's operational parameters. The avionics components are primarily housed within the Payload Avionics Box, which is strategically located on the aerostat to facilitate optimal performance of its sensors and communication devices.

2.3.1. Components of the Avionics Suite:

a) **Payload Avionics Box:** The Payload Avionics Box is a critical component of the avionics suite, containing various sensors, communication devices, and power supply units. It is connected to the transducer box via a cable, ensuring accurate data transmission. The primary components housed in the Payload Avionics Box include:

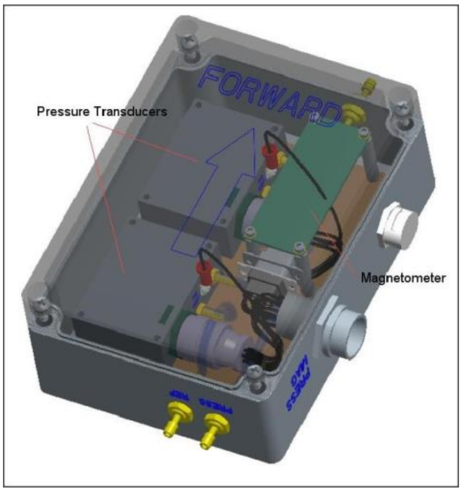


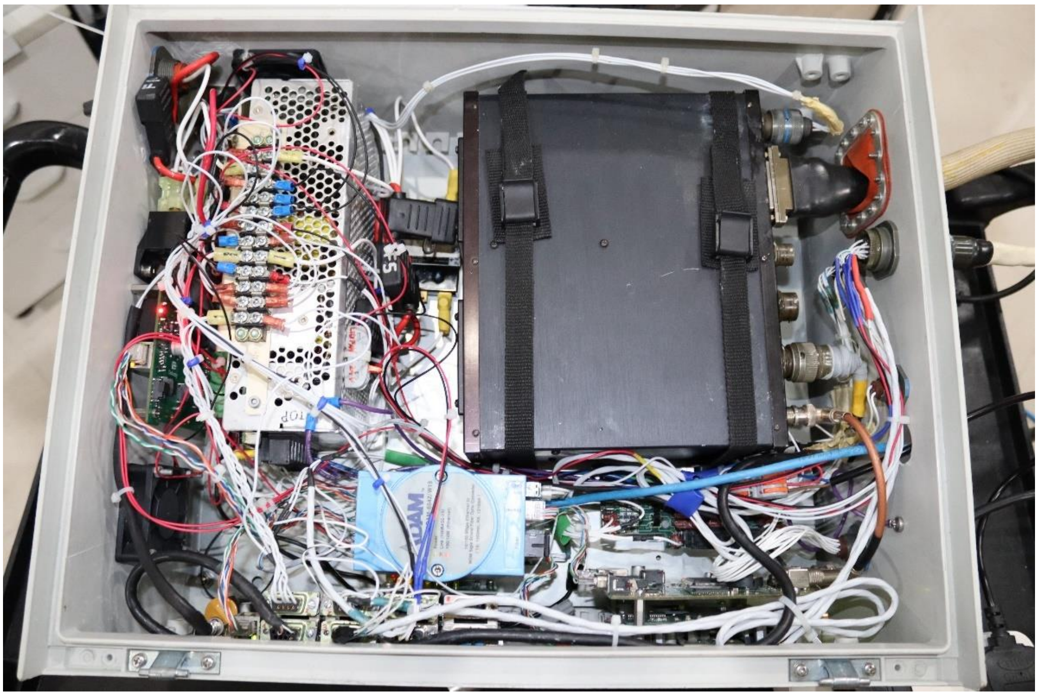
	Component	Description
1	ADAM 6520 Ethernet Switch	Facilitates data communication between various avionics components, providing Ethernet connectivity to the system.
2	ADAM 6542 W15 Fiber Optic to Ethernet Converter	Converts data signals from fiber optics to Ethernet, enabling seamless integration with the ground station.
3	MOXA NPort 5650-DTL-T Serial Server	A serial-to-Ethernet converter that allows multiple serial devices to communicate over a TCP/IP network.
4	AXIS 241S Video Server	A single-port MPEG4 video server that transmits live video feeds from payload sensors back to the GCS.
5	ADAM 4016 Strain Gauge Signal Conditioner Module	Processes data from strain gauges, including the load cell at the confluence point, to provide accurate tension measurements.
6	Prosoft Technology RadioLinx RLXIB-IHW RF Bridge	An RF bridge that enables wireless data transmission when required.
7	Vicor VI-LJW1-EW 24/12 DC-DC Converter and CUI VPM-S800-24R-N Power Supply	Power supply units that ensure reliable operation of the avionics components.
8	Honeywell HMR3000 Magnetometer	Provides precise heading, pitch, and roll data critical for maintaining the aerostat's orientation and stability.
9	Honeywell PPT0001DWW2VB-B Pressure Transducer	Measures the hull helium and ballonet air pressures, essential for monitoring the aerostat's lift and stability.
10	Airmar PB200 Weather Station	Monitors wind speed, direction, air temperature, barometric pressure, and magnetic compass heading, providing comprehensive environmental data to the GCS.
11	Cisco Luxul XW-5XO-FP7 Antenna	A high-gain RHCP 2.4GHz antenna used for reliable communication between the aerostat and the GCS.
12	Load Cell	Details Required

b) **Transducer Box:** The Transducer Box is located on the underside of the envelope, inside the RAM Air Scoop, just forward of the Ballonet Blowers. It houses critical sensors for monitoring the aerostat's performance, including:

i) **Pressure Sensors:** Used to measure hull helium pressure and ballonet air pressure, providing data essential for maintaining the aerostat's buoyancy and stability.

ii) **Magnetometer:** Measures the aerostat's heading and attitude, providing real-time data on its orientation relative to the ground.





---- Add Picture of Transducer Box

2.3.1. Interconnectivity and Communication:

All sensors and avionics components are interconnected via a robust network of Ethernet switches, fiber optic converters, and serial servers. This network architecture supports the seamless integration of sensor data into the overall telemetry system, allowing for comprehensive monitoring and control of the aerostat's performance. The use of a MOXA Serial TCP Server facilitates the transmission of RS232 data over the Ethernet network, further enhancing the system's flexibility and scalability.

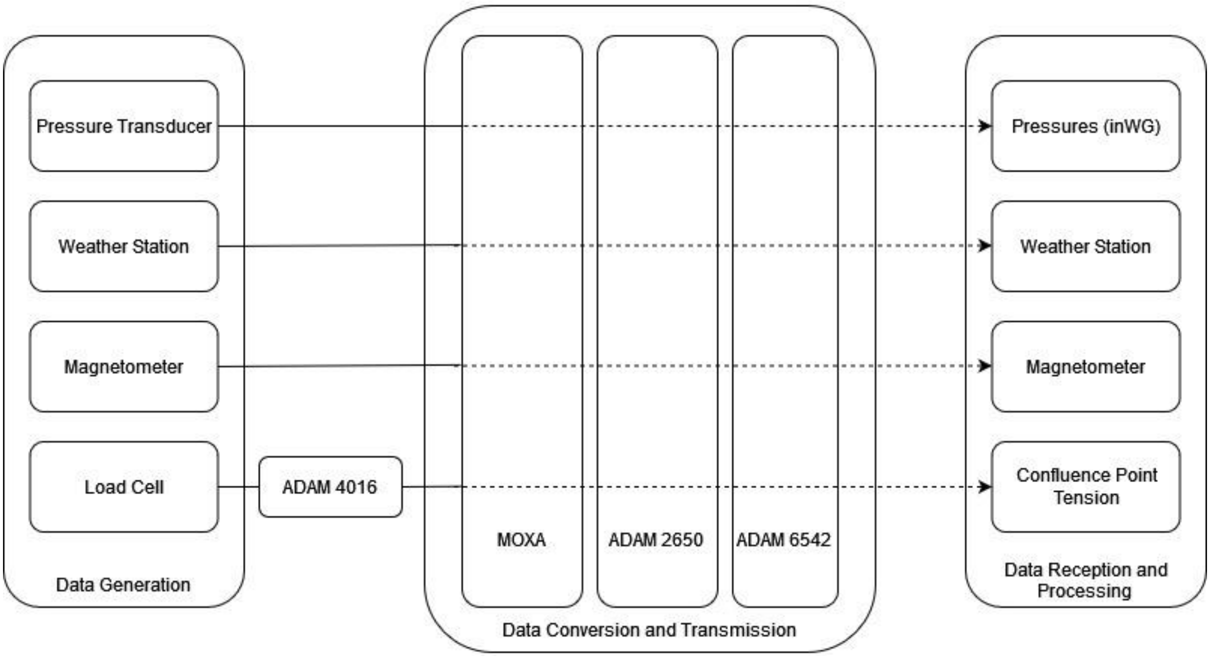
---- Add Interconnectivity Diagram

2.3.2. Operational Capabilities:

The Avionics Suite's integrated design allows for the efficient routing of data from the aerostat's various sensors and subsystems to the GCS. The use of UDP/IP protocols ensures that data is transmitted with minimal latency, enabling operators to make informed decisions based on real-time telemetry. Additionally, the system's reliance on Ethernet and fiber optic connections enhances its reliability and resilience, reducing the risk of data loss or interference.

3. Data Lifecycle and Integration

The data lifecycle in the Aerostat 200 system is a critical process that ensures accurate and real-time monitoring of the aerostat's performance and environmental conditions. This section outlines the journey of data from its generation by various sensors to its final display and archival at the Ground Control Station (GCS). The primary objective is to maintain situational awareness and control over the aerostat during its operations.



3.1. Data Generation

The data generation process begins with the various sensors and avionics components installed on the aerostat and payload bar. These include:

1. **Helium and Ballonet Pressure Transducers (Honeywell PPT0001DWW2VB-B):** Measure the internal pressures of the helium envelope and ballonet, which are crucial for assessing lift and stability.
2. **Weather Station (AIRMAR PB200):** Provides real-time data on wind speed, wind direction, air temperature, and barometric pressure, essential for operational decisions.
3. **Magnetometer (Honeywell HMR3000):** Delivers accurate readings of the aerostat's heading, pitch, and roll, vital for navigation and stability.
4. **Load Cell (Honeywell, connected via ADAM 4016 DAQ Module):** Monitors tether tension at the confluence point, providing critical data on the aerostat's lift and anchor status.

3.2. Data Conversion and Transmission

Data generated by the sensors is initially in analog or proprietary digital formats, which must be converted into standard network protocols for seamless transmission and integration. The process is facilitated by the following components:

1. **MOXA NPort 5650-DTL-T Serial Server:** Converts serial data from RS232 sensors into TCP/IP format, allowing multiple serial devices to communicate over the network.
2. **ADAM 4016 DAQ Module:** Converts strain gauge signals from the load cell into digital data, which is then processed and transmitted over the Ethernet network.
3. **ADAM 6520 Ethernet Switch:** Aggregates all data from various sensors and avionics components, providing a central hub for data routing.
4. **ADAM 6542 W15 Fiber Optic to Ethernet Converter:** Converts Ethernet data to fiber optic signals, ensuring reliable high-speed data transmission over the tether's single-mode fiber line.

3.3. Data Routing and Control

Once converted into UDP/IP packets, the data is transmitted through the tether's fiber optic line or via the RF bridge, depending on the operational requirements. The data flow is managed by:

1. **Ethernet Network via the Tether:** Utilizes a single-mode fiber optic line within the tether, connected through the ADAM 6542 W15 Fiber Optic Converter, ensuring a robust and secure data link to the GCS.
2. **Prosoft Technology RadioLinx RLXIB-IHW RF Bridge:** Provides a backup wireless communication link for telemetry and control data, especially useful when the tether is under stress or not fully deployed.

3.4. Data Reception and Processing at the GCS

At the GCS, data is received through the utility vehicle's smart Ethernet switch and routed to the GCS computer. The telemetry software processes this data, integrating various sensor inputs into a cohesive display on the operator's interface. Key features include:

1. **Real-time Data Visualization:** Displays telemetry data, including aerostat position, environmental conditions, and system health, on dedicated screens within the GCS.
2. **Command and Control:** Allows operators to send commands back to the aerostat via the tether or RF link, such as adjusting the blower controls or toggling the strobe lights.
3. **Archival and Playback:** The GCS logs all telemetry data, providing playback capabilities for mission analysis and review, allowing operators to replay data logs for post-mission assessment.

3.5. Data Archival and Security

Data collected during operations is securely archived within the GCS for future reference, compliance, and training purposes. The archiving process ensures:

1. **Secure Storage:** All data is stored on encrypted drives within the GCS to prevent unauthorized access.
2. **Data Integrity:** Regular checks are performed to ensure the integrity and availability of historical data, supporting operational continuity and reliability.
3. **Replay Functionality:** Archived data can be replayed using the GCS telemetry software, aiding in detailed mission debriefs and continuous improvement of aerostat operations.

4. Ground Control Station (GCS)

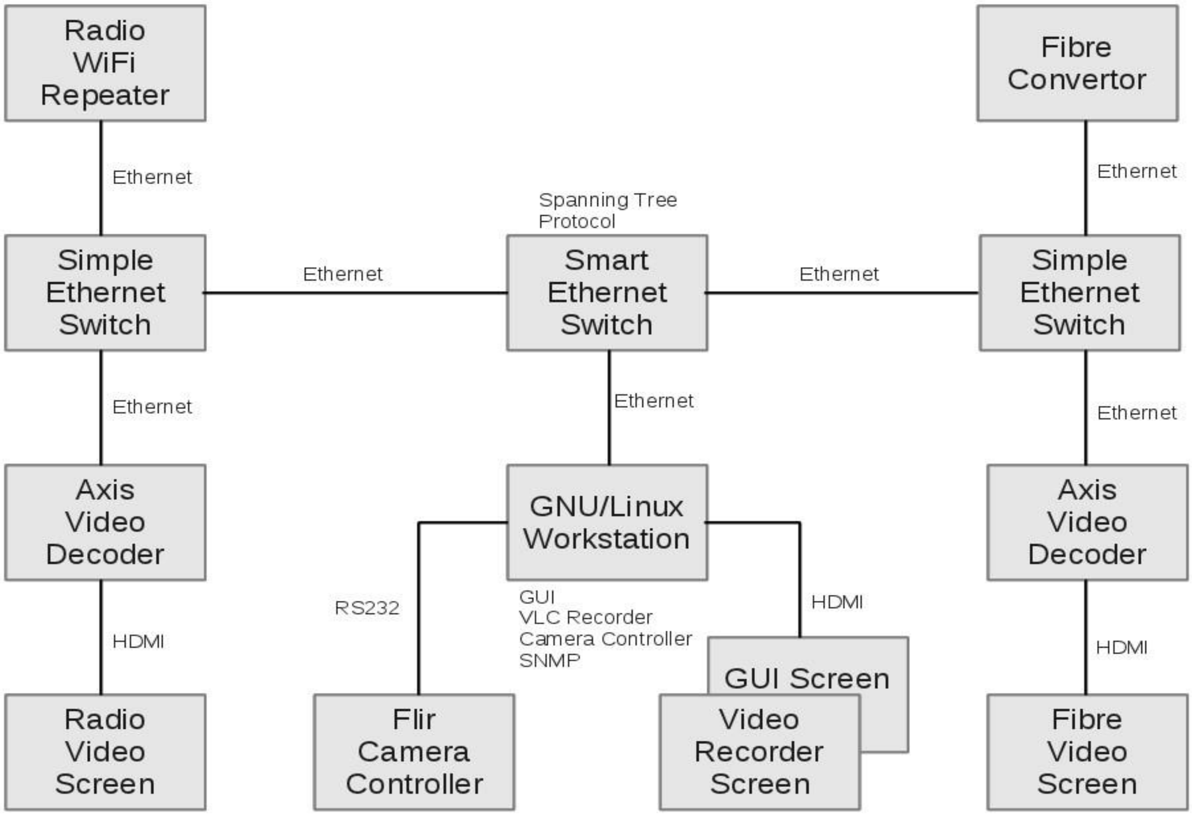
The Ground Control Station (GCS) for the ADASI A200 Aerostat System is a critical component that enables the operation, control, and monitoring of the aerostat during missions. The GCS is housed within a modified utility vehicle, specifically a Ford F350 turbo diesel V8, which provides mobility and flexibility for operations in various environments. The GCS is designed to accommodate all necessary equipment for aerostat management, ensuring efficient and effective control over the aerostat and its payload.

4.1. Utility Vehicle

The utility vehicle serves as the physical platform for the GCS, equipped with essential modifications to support the aerostat's operations. Key components of the utility vehicle include:

- **Ford F350 Turbo Diesel V8:** Provides the necessary power and mobility for the GCS, allowing it to operate in various terrains and environments.
- **Ground Antenna and Antenna Holder:** Facilitates communication between the GCS and the aerostat, ensuring a stable link for command and telemetry data.
- **Generator:** Supplies power to the GCS equipment, enabling continuous operations even in remote locations.
- **Crew Seats (2):** Provides seating for the operators, allowing them to manage the aerostat comfortably and effectively.
- **Air Conditioning Unit:** Ensures a controlled environment within the vehicle, protecting sensitive electronics from extreme temperatures.
- **Equipment Racks (2):** Metal racks that securely house the GCS electronics, including computers, displays, and networking equipment.
- **Displays (4):** Visual interfaces for operators, presenting telemetry data, payload videos, and rolling logs.
- **Ground Harness and Ethernet Switch:** Facilitates connectivity and communication within the GCS network.
- **RF Modem:** Supports radio frequency communication, serving as a backup link for command and payload data.
- **Keyboard/Mouse and GCS Computer:** Provides the primary interface for operators to interact with the GCS software and manage aerostat operations.
- **GCS Computer Operating System:** The system runs on a ruggedized GETAC X500 laptop with Fedora 19, providing a stable and robust platform for the telemetry and control software.
- **Smart Ethernet Switch:** Manages data traffic within the GCS, ensuring efficient communication between connected devices.
- **Telemetry Software:** Custom-developed software that processes telemetry data and presents it to the operators in a user-friendly format.

4.2. GCS Block Diagram:



The Ground Control Station (GCS) block diagram outlines the interconnected components crucial for operating and monitoring the Aerostat 200 system. Central to the GCS is the GNU/Linux Workstation, which manages data processing, telemetry, and control of the aerostat and its payload through a user-friendly interface. The system integrates multiple Simple and Smart Ethernet Switches, facilitating data communication between the various components, including video decoders and fiber converters. The FLIR Camera Controller connects via RS232 for precise control of the surveillance cameras, while Axis Video Decoders handle video feeds, outputting them to dedicated display screens for real-time monitoring. The GCS employs redundancy and advanced networking protocols to ensure reliable data flow and robust performance, essential for the effective operation of the Aerostat 200.

4.3. GCS Displays

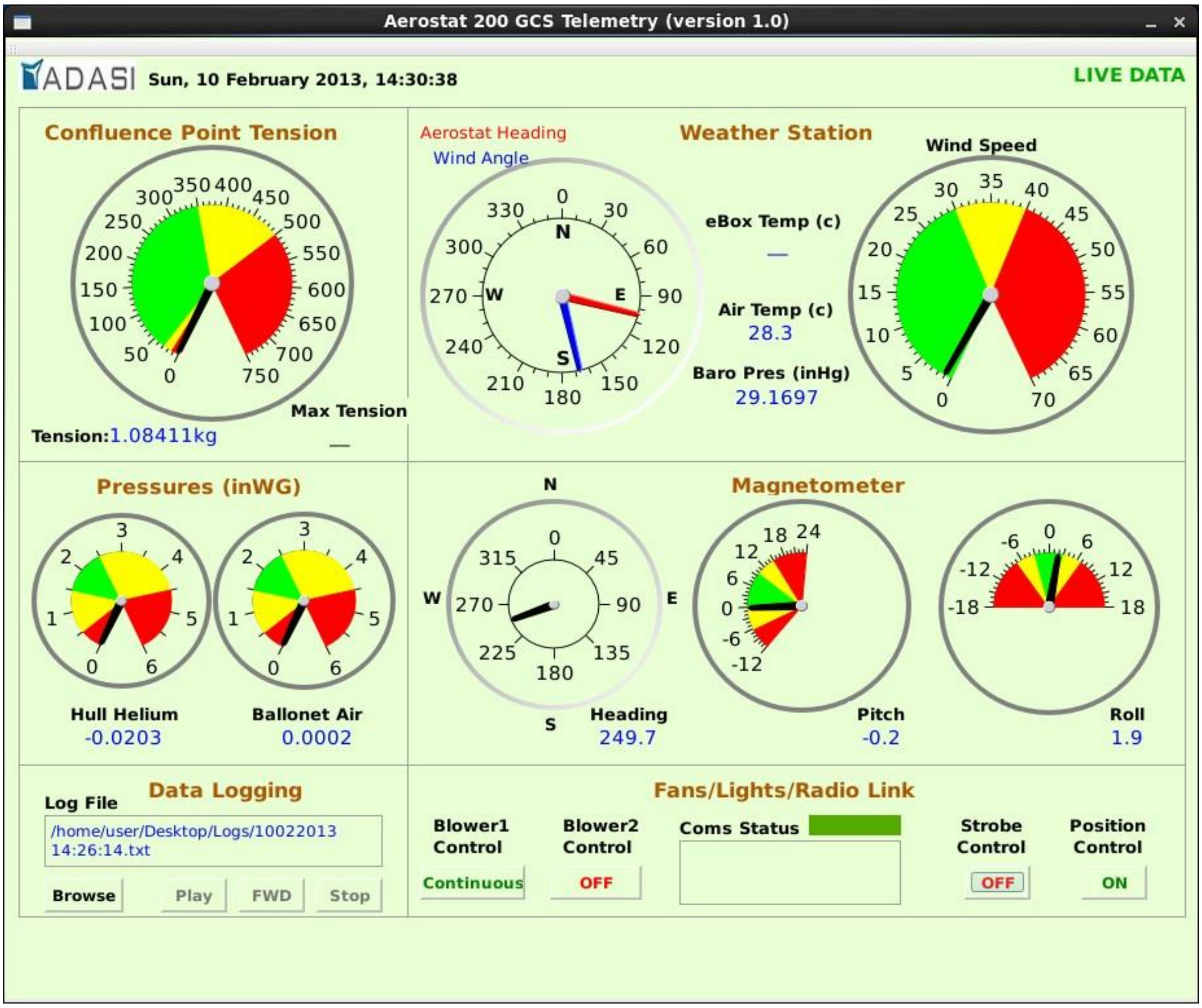
The GCS is equipped with four displays housed within the utility vehicle's cabin. Each display serves a specific function:

1. **Telemetry Display:** Shows real-time aerostat data including sensor readings and system status.
2. **Payload Video Display:** Provides visual feedback from the payload sensors, aiding in mission management.
3. **Rolling Log Display:** Presents a continuous log of operational data, allowing operators to track changes over time.
4. **Auxiliary Display:** Used for additional data or system diagnostics as needed.

5. GCS Telemetry Software

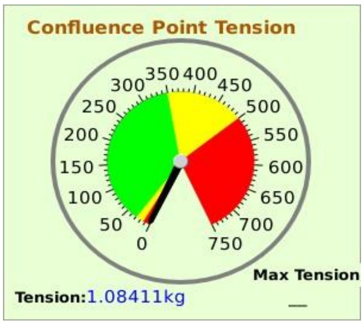
The GCS Telemetry GUI is a sophisticated software interface that displays critical real-time data from the aerostat and its sensors. This interface enables operators to monitor the aerostat's performance and respond to any anomalies promptly. Key features of the GUI include:

- **Telemetry Data Display:** Shows crucial parameters such as confluence point tension, heading, wind speed and angle, temperatures, barometric pressure, and pressures of helium and ballonet.
- **Control Features:** Allows operators to control blowers, strobe lights, and position lights, and monitor the status of communication links.
- **Data Logging:** Enables the recording of telemetry data for post-mission analysis. Operators can start, stop, and replay logs to review mission performance.
- **Graphical Representation:** Uses color coding (green, yellow, red) to indicate the operational status of various parameters, aiding operators in quickly assessing system health.
- **Alerts and Warnings:** Provides visual alerts and warnings when parameters fall outside of acceptable ranges, prompting operators to take corrective action.



5.1. Confluence Point Tension Display

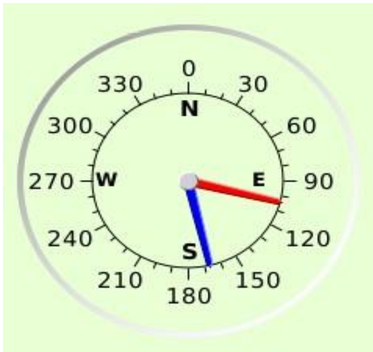
The Confluence Point Tension panel displays both graphically and numerically the tether tension sensed by the load cell installed at the confluence point. The maximum tension experienced by the tether is also displayed. Using this reading, excess lift, and therefore helium fill state, can only be accurately assessed when the aerostat is within 50ft of the Base Unit in zero winds and with no superheat.



PARAMETER	UNIT	FULL RANGE	LOWER RED RANGE	LOWER YELLOW RANGE	GREEN RANGE	UPPER YELLOW RANGE	UPPER RED RANGE
Tension	kg	0 – 750	0 – 15	15 – 30	30 – 350	350 – 500	500 – 750

5.2. Heading and Wind Angle Display

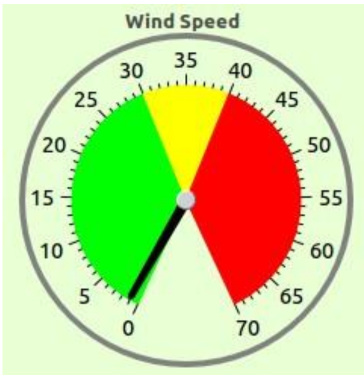
The Weather Station panel displays graphically the magnetic heading of the aerostat and the true wind angle on a single display.



PARAMETER	UNIT	FULL RANGE
Heading	degrees	0 – 360
Wing Angle	degrees	0 – 360

5.3. Wind Speed Display

The Weather Station panel also displays graphically wind speed in knots.



PARAMETER	UNIT	FULL RANGE	GREEN RANGE	YELLOW RANGE	RED RANGE
Tension	Knots	0 – 70	0 – 30	30 – 40	40 – 70

5.4. Temperature Displays

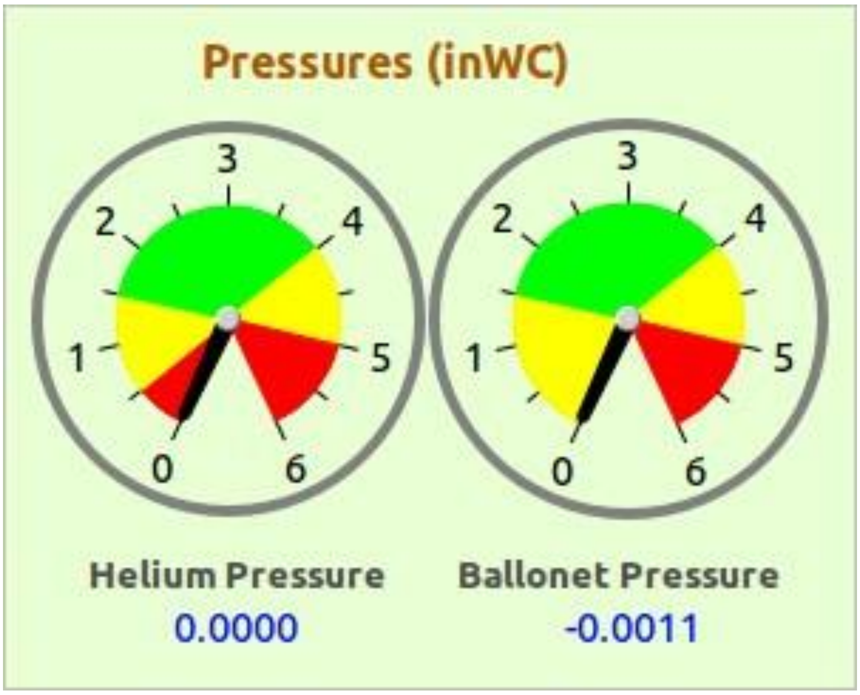
The Weather Station panel also numerically displays the Box Temperature and ambient Air Temperature in degrees Celsius.

5.5. Barometric Pressure Displays

The Weather Station panel also numerically displays the Barometric Pressure in inches of mercury (inHg).

5.6. Hull Helium and Ballonet Air Pressure Displays

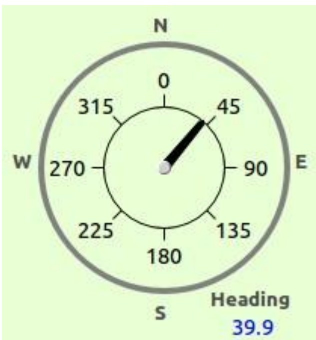
The Pressures panel displays both graphically and numerically the pressures sensed by the pressure sensors in the Transducer Box (Section 4.2.4) for helium and ballonet pressure in inches of water gauge (inWG).



PARAMETER	UNIT	FULL RANGE	LOWER RED RANGE	LOWER YELLOW RANGE	GREEN RANGE	UPPER YELLOW RANGE	UPPER RED RANGE
Helium Pressure	inWG	0 – 6	0 – 0.5	0.5 – 1.5	1.5 – 2.5	2.5 – 4.5	4.5 – 6.0
Ballonet Pressure	inWG	0 – 6	0 – 0.5	0.5 – 1.5	1.5 – 2.5	2.5 – 4.5	4.5 – 6.0

5.7. Aerostat Heading Display

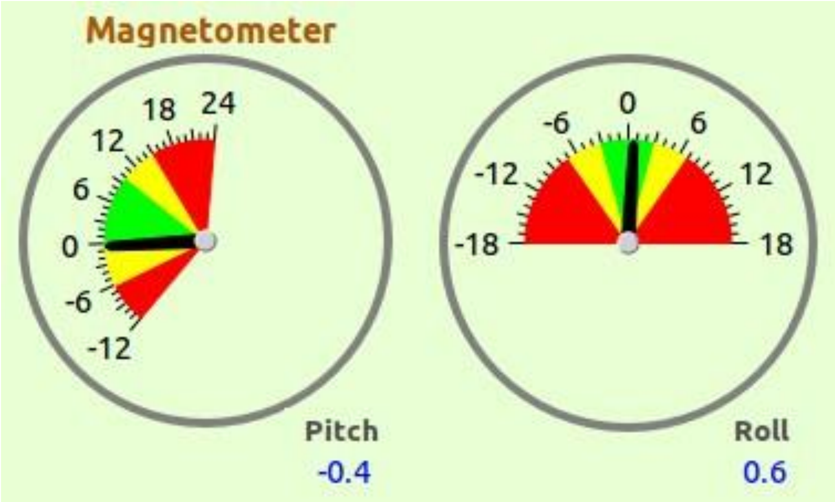
The Magnetometer panel displays the magnetic heading both graphically and numerically sensed by the magnetometer in the Transducer Box.



PARAMETER	UNIT	FULL RANGE
Heading	Degrees	0 – 360

5.8. Aerostat Attitude Displays

The Magnetometer panel also graphically and numerically displays the pitch and roll angles sensed by the magnetometer in the Transducer Box.



PARAMETER	UNIT	FULL RANGE	LOWER RED RANGE	LOWER YELLOW RANGE	GREEN RANGE	UPPER YELLOW RANGE	UPPER RED RANGE
Pitch	degrees	-12 – 24	-12 – -6	-6 – 0	0 – 10	10 – 15	15 – 24
Roll	degrees	-18 – 18	-18 – -7	-7 – -3	-3 – 3	3 – 7	7 – 18

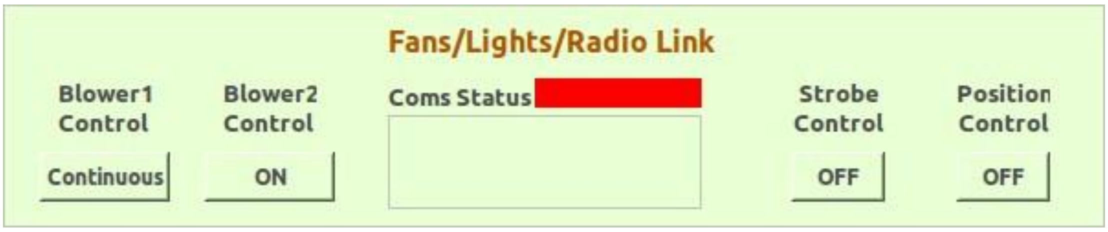
5.9. Data Logging Display

The Data Logging panel allows telemetry data to be recorded. Note play back is currently

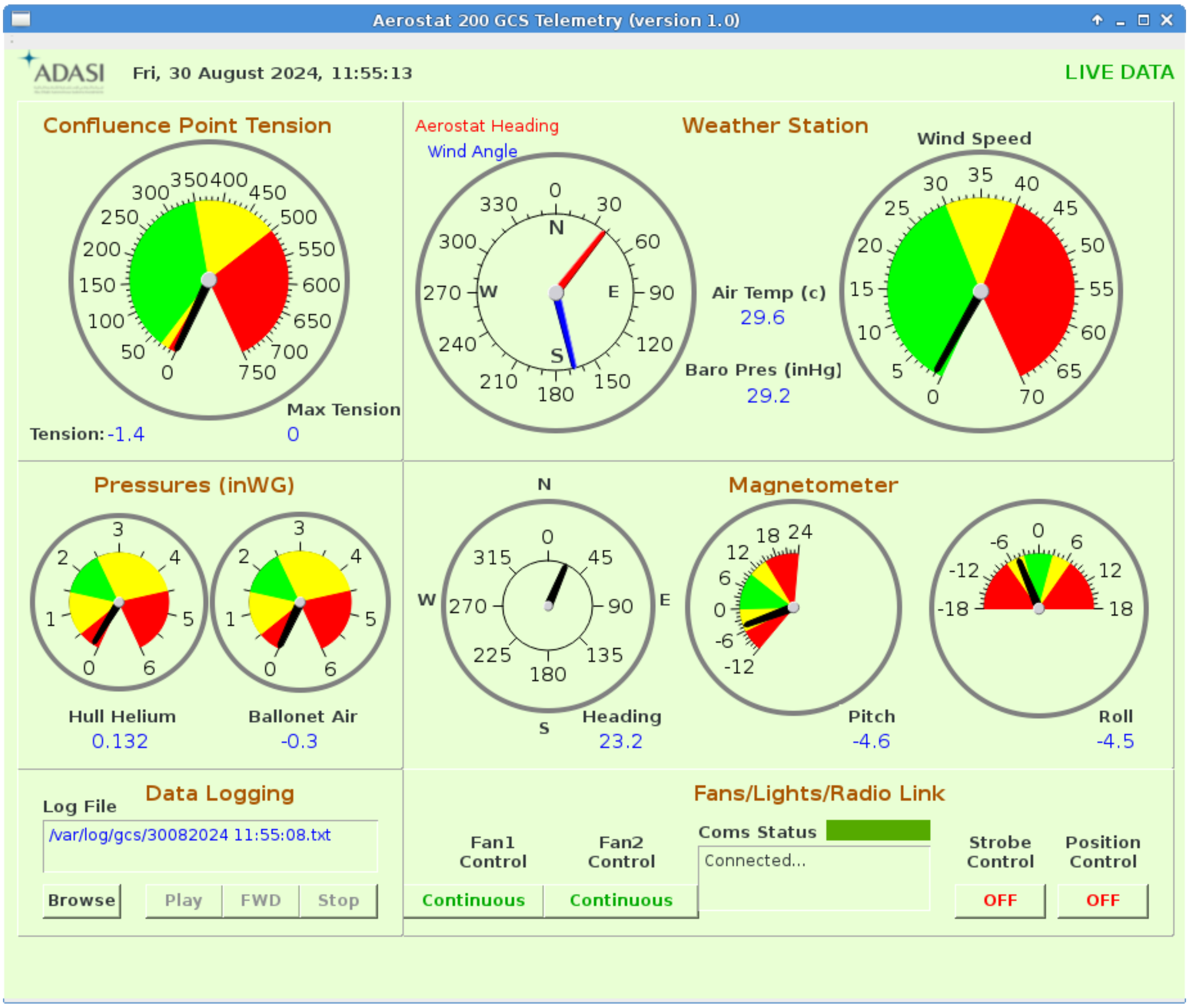
disabled.

5.10. Blower/Lights/Radio Link Panel

This panel allows the status of the two blowers (Section 4.1.3.4) to be controlled. It also allows the strobe and position lights to be controlled. Also displayed is the status of the RF link between the airborne mission equipment (Section 0) and the GCS.



5.11. Recent Screenshot of the Interface



6. Operator Console Assessment:

The Operator Console is a crucial element of the aerostat system, offering operators the necessary interface to monitor and control aerostat missions. It serves as the primary point of interaction between the operator and the aerostat, ensuring efficient and accurate command and control. This section provides a detailed analysis of the current configuration, hardware specifications, system configuration, software environment, and scripts that govern the operation of the console. This analysis is vital for reverse engineering the system and developing the AlfaOps command and control software.

6.1. Hardware Overview

The Operator Console is a ruggedized GETAC X500 laptop, designed for harsh environments and robust aerostat operations. Below are the key specifications:

Component	**Details**
Model	GETAC X500
Processor	Intel Core i7 M260 (Quad-core)
Memory	8 GB RAM
Storage	280 GB Hard Drive
Display	15.6-inch sunlight-readable screen
Connectivity Ports	Ethernet, USB, RS232, HDMI, VGA
Ruggedness Standards	MIL-STD-810G, IP65
Serial Number	RD663X0252
Part Number	242128000001

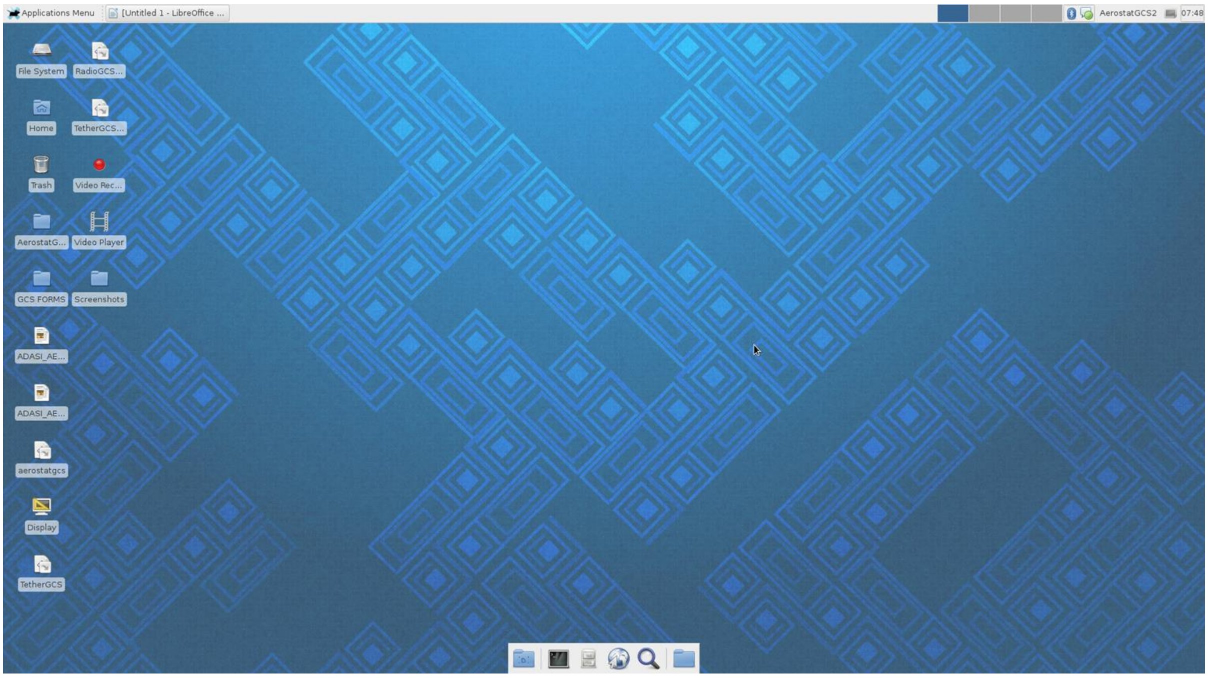
These specifications ensure that the console meets the operational demands of aerostat missions, providing reliable performance under diverse and challenging conditions.

- 

6.2. System Configuration

The Operator Console runs Fedora 19, a Linux-based operating system known for its stability and security. The console primarily utilizes the Aerostat Control Software, which facilitates monitoring and control of aerostat operations. The software is supported by additional utilities that aid in operational and maintenance tasks, as detailed below:

- **User Login:** XFCE session with username **AerostatGCS2**, domain localhost.localdomain, and password **PassWorD**.
- **Development Tools:** Includes kernel-devel, qt3, and qt3-designer for software updates and modifications.
- **Network IP Configuration:** Assigns static IPs for critical components such as weather stations and pressure sensors, each with specific ports for data transmission.
- **Network and Diagnostic Tools:** Tools such as Wireshark and nmap for network monitoring and troubleshooting.
- **Text Editors and Utilities:** gedit, joe for text editing, cutecom for serial communication, and hexedit for binary file editing.
- **System Paths:** Key executables and scripts are stored under /usr/local/bin/ and /usr/local/GCSSW/, providing structured access and execution paths.



6.3. Software Analysis

The Aerostat Control Software is installed in multiple versions, primarily located in the /usr/local/GCSSW/ directory. The software handles various aerostat control functions, such as telemetry, data acquisition, and command execution. Key elements of the software include configuration files, batch scripts, and executable binaries, which are critical for system operation.

6.3.1. Key Software Directories:

1. /usr/local/GCSSW/AerostatGCS
2. /usr/local/GCSSW/AerostatGCS_ver2.1

6.3.2. Software Characteristics:

- Primary Function: Aerostat control and monitoring
- Configuration Management: Managed through batch files and config scripts that set operational parameters for different aerostat components.

6.4. Script Analysis

Scripts play a vital role in automating system tasks and ensuring the smooth operation of the console. The following scripts are critical to the system's functionality:

6.4.1. Script Overview:

1. **network-tether and network-radio**: Manage Ethernet configurations and ensure proper network setup with aerostat components.
2. **flirrestart:** Resets the Flir Controller, a critical component for imaging and surveillance tasks.
3. **aerostatgcs:** Initiates the Aerostat Control Software, ensuring all necessary processes are running for operation.

Script Paths and Functions:

1. /usr/local/bin/network-tether: Configures tether-related network settings.
2. /usr/local/bin/network-radio: Configures radio communication settings, stops NetworkManager, sets IP addresses, and performs connectivity checks.
3. /usr/local/bin/flirrestart: Kills existing processes for the Flir Controller and restarts it, using zenity for user feedback.
4. /usr/local/bin/aerostatgcs: Launches the primary Aerostat Control Software, killing any existing instances before restarting.

6.4.2. Analysis:

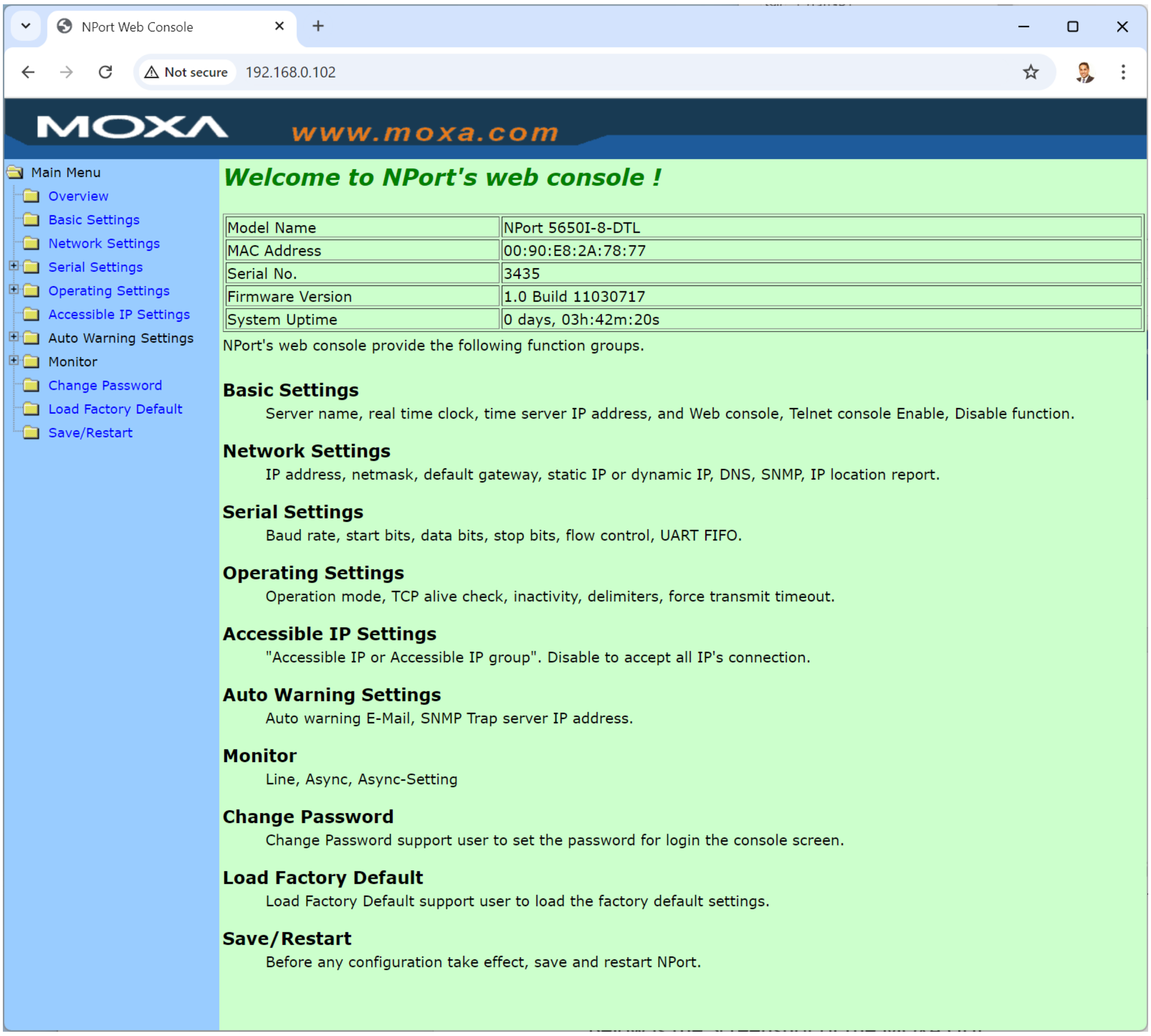
Each script automates specific tasks crucial for the operation of the aerostat system. For example, the network-radio script sets up communication with the aerostat by configuring Ethernet interfaces, stopping conflicting services, and validating connectivity via ping tests. The flirrestart script ensures the imaging system is reset and operational, crucial for real-time surveillance and monitoring.

6.5. Device Configuration

The Moxa NPort 5650I-8-DTL is a critical component in the aerostat system, serving as a serial-to-Ethernet converter that integrates various avionics and sensor data with the Ground Control Station (GCS). This device enables seamless data communication between multiple serial devices and the GCS over a TCP/IP network, facilitating real-time monitoring and control of the aerostat system.

6.5.1. Moxa NPort Overview

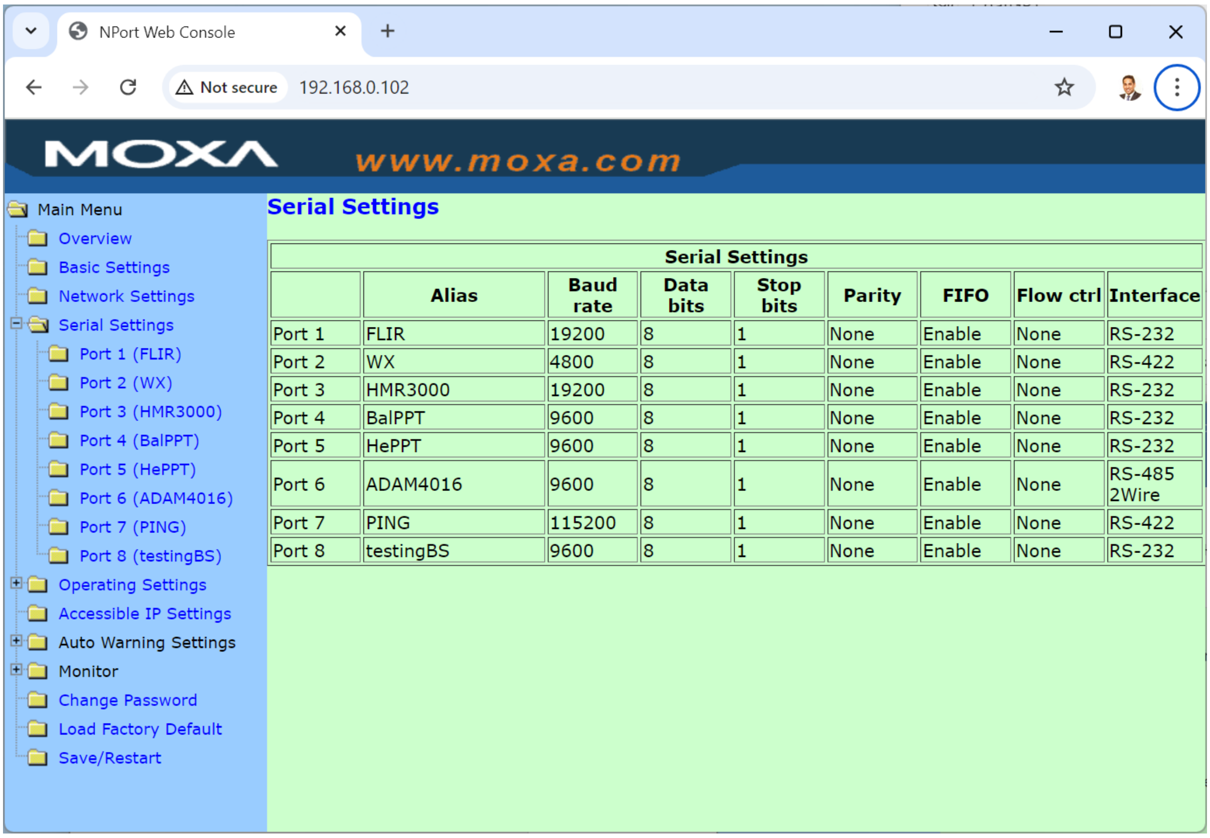
The Moxa NPort 5650I-8-DTL device is configured to operate in TCP Server Mode, allowing the GCS to connect and retrieve data from connected serial devices. The serial settings are tailored to the specific requirements of each connected sensor, ensuring reliable data transmission.



6.5.2. Moxa Serial Settings

The Moxa NPort 5650I-8-DTL provides configurations for up to eight serial ports, each designated for a specific device or sensor. The serial settings are meticulously configured, including baud rate, data bits, stop bits, parity, and flow control settings. Key configurations include:

- **Port 1 (FLIR):** Configured with a baud rate of 19200, RS-232 interface, enabling data communication from the FLIR camera.
- **Port 2 (WX):** Set to a baud rate of 4800, RS-422 interface, for data from the weather station.
- **Port 3 (HMR3000):** Uses RS-232 with a baud rate of 19200, providing heading, pitch, and roll data from the magnetometer.
- **Port 4 (BalPPT):** Configured with RS-232 interface and a baud rate of 9600 for ballonet pressure transducer data.
- **Port 5 (HePPT):** Setup with RS-232 at 9600 baud for helium pressure sensor data.
- **Port 6 (ADAM4016):** Connected to the data acquisition module via RS-485 interface with 9600 baud rate.
- **Port 7 (PING):** Uses RS-422 at 115200 baud rate, dedicated for system diagnostic purposes.
- **Port 8 (testingBS):** Configured with RS-232, primarily reserved for backup or testing functionalities.

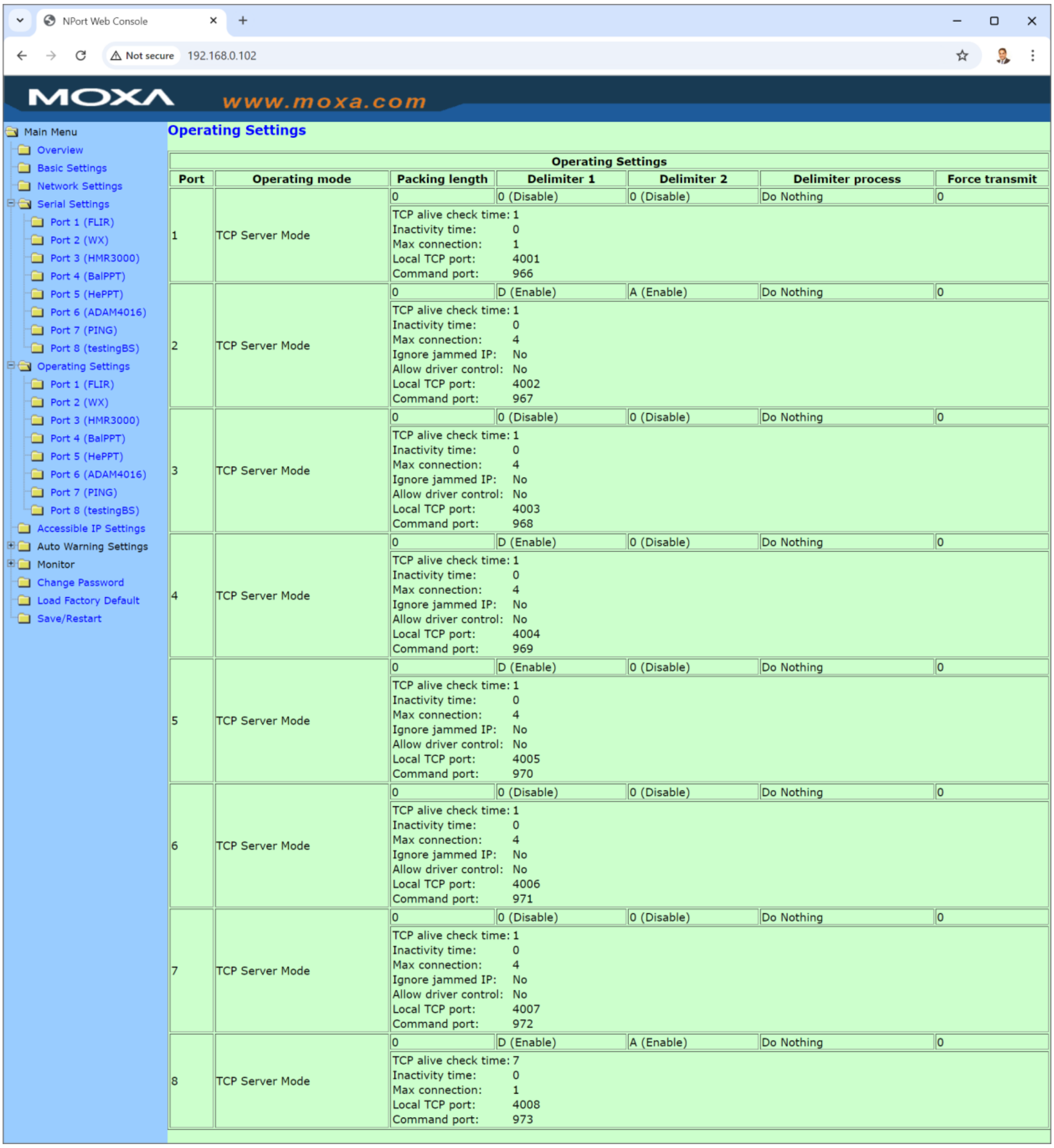


Serial Settings

	Alias	Baud rate	Data bits	Stop bits	Parity	FIFO	Flow ctrl	Interface
Port 1	FLIR	19200	8	1	None	Enable	None	RS-232
Port 2	WX	4800	8	1	None	Enable	None	RS-422
Port 3	HMR3000	19200	8	1	None	Enable	None	RS-232
Port 4	BalPPT	9600	8	1	None	Enable	None	RS-232
Port 5	HePPT	9600	8	1	None	Enable	None	RS-232
Port 6	ADAM4016	9600	8	1	None	Enable	None	RS-485 2Wire
Port 7	PING	115200	8	1	None	Enable	None	RS-422
Port 8	testingBS	9600	8	1	None	Enable	None	RS-232

6.5.3. Operating Settings

The Moxa device's operating settings are finely tuned to ensure optimal performance, including TCP alive checks, inactivity timeouts, and force transmit options, allowing the system to maintain robust communication links even in adverse conditions.



Operating Settings						
Port	**Operating mode**	**Packing length**	**Delimiter 1**	**Delimiter 2**	**Delimiter process**	**Force transmit**
1	TCP Server Mode	0	0 (Disable)	0 (Disable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 1 Local TCP port: 4001 Command port: 966				
2	TCP Server Mode	0	D (Enable)	A (Enable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 4 Ignore jammed IP: No Allow driver control: No Local TCP port: 4002 Command port: 967				
3	TCP Server Mode	0	0 (Disable)	0 (Disable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 4 Ignore jammed IP: No Allow driver control: No Local TCP port: 4003 Command port: 968				
4	TCP Server Mode	0	D (Enable)	0 (Disable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 4 Ignore jammed IP: No Allow driver control: No Local TCP port: 4004 Command port: 969				
5	TCP Server Mode	0	D (Enable)	0 (Disable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 4 Ignore jammed IP: No Allow driver control: No Local TCP port: 4005 Command port: 970				
6	TCP Server Mode	0	0 (Disable)	0 (Disable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 4 Ignore jammed IP: No Allow driver control: No Local TCP port: 4006 Command port: 971				
7	TCP Server Mode	0	0 (Disable)	0 (Disable)	Do Nothing	0
		TCP alive check time: 1 Inactivity time: 0 Max connection: 4 Ignore jammed IP: No Allow driver control: No Local TCP port: 4007 Command port: 972				
8	TCP Server Mode	0	D (Enable)	A (Enable)	Do Nothing	0
		TCP alive check time: 7 Inactivity time: 0 Max connection: 1 Local TCP port: 4008 Command port: 973				

6.5.4. Line Monitoring and Async Settings

Line monitoring and async settings ensure proper data flow management and error-free communication:

- **Line Monitoring:** Displays current line status, IP addresses in use, and monitoring activity across all ports, ensuring that each device remains connected and operational.

Monitor Line

Line					
Port	OP Mode	IP1	IP2	IP3	IP4
1	TCP Server Mode	Listen			
2	TCP Server Mode	Listen	Listen	Listen	Listen
3	TCP Server Mode	Listen	Listen	Listen	Listen
4	TCP Server Mode	192.168.0.55	Listen	Listen	Listen
5	TCP Server Mode	Listen	Listen	Listen	Listen
6	TCP Server Mode	Listen	Listen	Listen	Listen
7	TCP Server Mode	Listen	Listen	Listen	Listen
8	TCP Server Mode	Listen			

- **Async Monitoring:** Displays transmitted and received data counts, monitoring each port's activity, and detecting any communication issues in real-time.

Monitor Async

Async							
Port	TxCnt	RxCnt	TxTotalCnt	RxTotalCnt	DSR	CTS	DCD
1	0	2910524	0	3499799	OFF	OFF	OFF
2	0	1073	0	197337	ON	ON	ON
3	0	462	0	166169	OFF	OFF	OFF
4	0	951496	0	951875	OFF	OFF	OFF
5	0	182	0	65726	OFF	OFF	OFF
6	8	18	1576	3546	ON	ON	ON
7	0	0	0	0	ON	ON	ON
8	0	0	0	0	OFF	OFF	OFF

6.5.5. Configured IPs and Ports

The Moxa NPort 5650I-8-DTL is configured with a primary IP address of 192.168.0.102 for network identification. Specific port assignments for the various sensors include:

- **Weather Station:** Port 4002
- **Digital Compass (HMR3000):** Port 4003
- **Ballonet Pressure Sensor:** Port 4004
- **Helium Pressure Sensor:** Port 4005
- **Data Acquisition Module (ADAM):** Port 4006
- **EBox Controller:** Port 4007

6.6. Data Output

To assess the data flow from each sensor connected to the aerostat system, telnet sessions were conducted on the configured ports for each sensor. Below is a summary of the observed data outputs, illustrating how sensor data is streamed and formatted before being relayed to the GCS:

6.6.1. Heading Sensor Output (Digital Compass HMR3000)

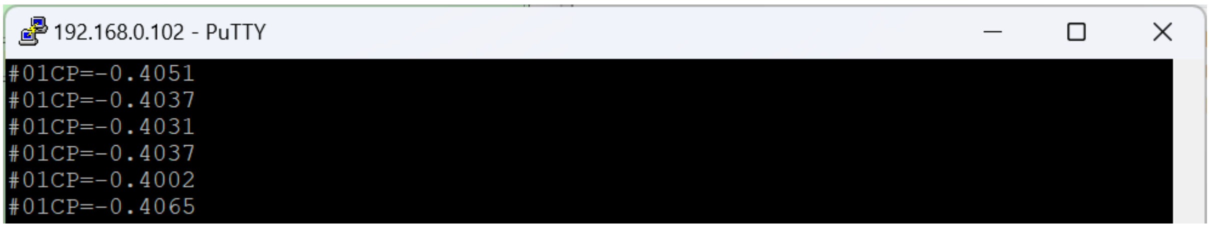
The digital compass outputs heading, pitch, and roll data in NMEA sentence format, which includes parameters such as heading angle, pitch angle, and roll angle. These data streams provide real-time information on the aerostat's orientation and are critical for maintaining operational stability.

Sample Output: *$PTNTHPR,24.8,N,-3.8,N,-4.0,N*05*

6.6.2. Pressure Sensor Output (Helium and Ballonet Pressure)

Pressure sensors provide continuous readings of helium and ballonet air pressures. These readings are essential for monitoring the aerostat's lift and ensuring that the envelope maintains optimal pressure levels.

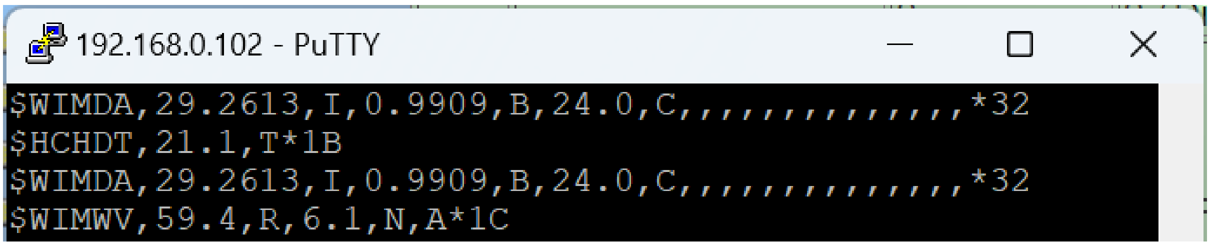
Sample Output: *#01CP=-0.4051*



6.6.3. 3. Weather Station Output (Airmar PB200)

The Airmar PB200 weather station delivers comprehensive environmental data, including wind speed, direction, air temperature, and barometric pressure. This data is crucial for flight operations as it helps the GCS to make informed decisions based on real-time environmental conditions.

Sample Output: $WIMDA,29.2613,I,0.9909,B,24.0,C,,,,,,,,,,*32 - $WIMWV,59.4,R,6.1,N,A*1C



6.7. Launching of Legacy Telemetry Software

The legacy telemetry software for the Aerostat system is launched using various shell scripts located on the desktop of the operator console. Each script is designed to initialize the telemetry software in specific modes depending on the type of connectivity—whether it is through tethered fiber optic connections or wireless radio links. Below is an analysis of the key scripts and their functionalities:

6.7.1. Key Shell Scripts and Execution Paths

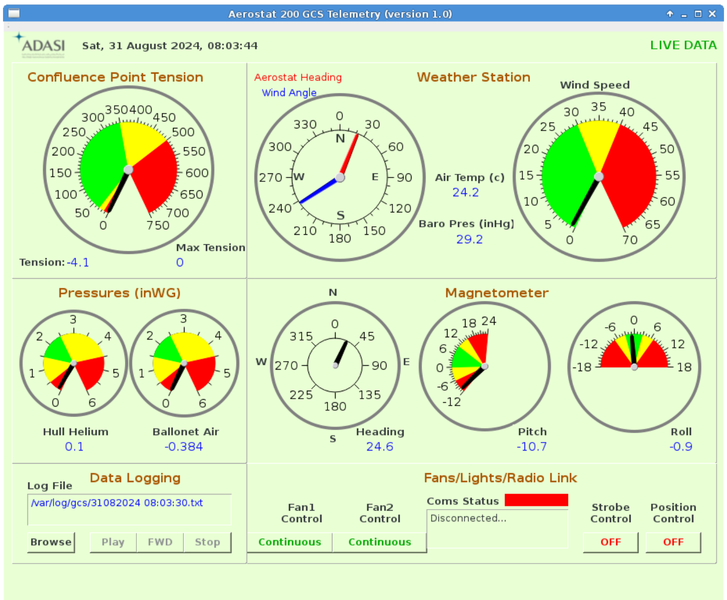
6.7.1.1. Script: aerostatgcs

- **Shortcut Target**: /usr/local/bin/aerostatgcs
- **Script Details**:

bash Copy code

```bash
#! /bin/bash
# Start the ADASI Aerostat Controller
killall AerostatGCS
/usr/local/GCSSW/AerostatGCS/AerostatGCS &
```

- **Function:** This script is used to start the ADASI Aerostat Controller, terminating any existing instances of the software before launching the GUI. The user simply double-clicks the shortcut, leading to the loading of the Aerostat GCS Telemetry version 1.0 window.
- **Upon launching the below Screens are loaded:**

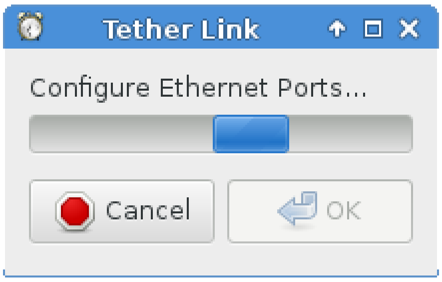


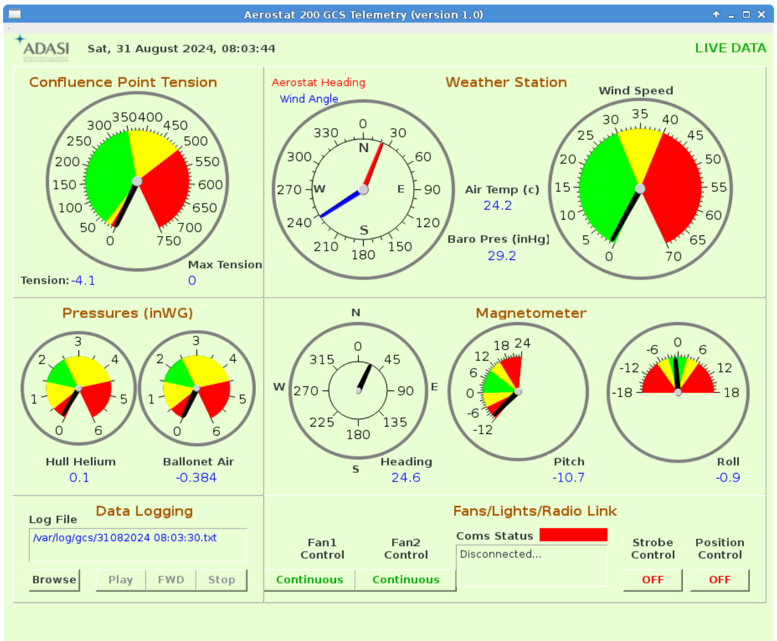
6.7.1.2. Script: TetherGCS

- **Shortcut Target:** /usr/local/bin/tether
- **Script Details:**

```bash
#! /bin/bash
/usr/local/bin/network-tether
/usr/local/bin/flirrestart
cd /usr/local/GCSSW/AerostatGCS
/usr/local/bin/aerostatgcs
```

- **Function:** This script is designed to connect the telemetry system using the fiber optic connection via the tether. Upon activation, it opens an initial connection window, followed by the Aerostat GUI, reflecting connectivity via the tether and showing version 1.0 of the software.
- **Upon launching the below Screens are loaded:**



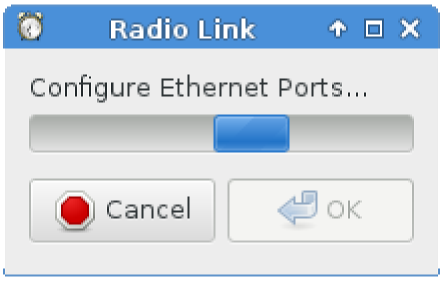


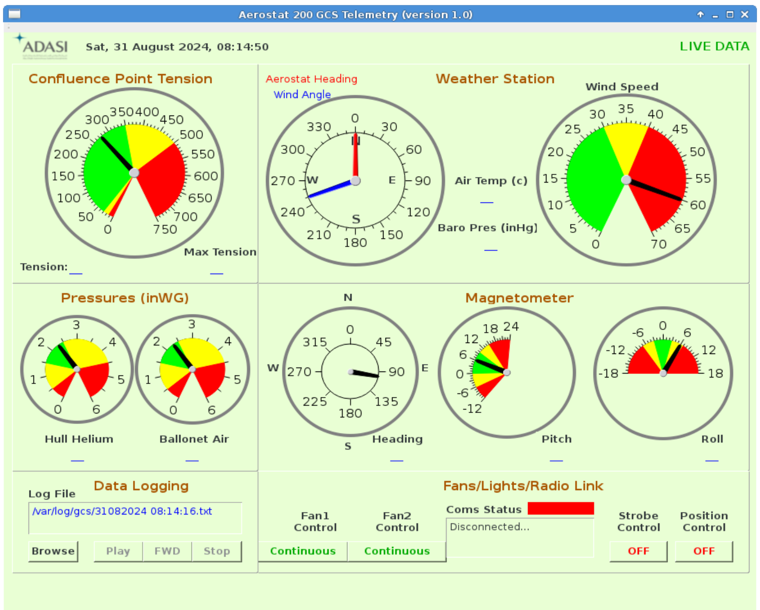
6.7.1.3. Script: RadioGCS_new

- **Shortcut Target:** /usr/local/bin/radio_ver2.1
- **Script Details:**

```bash
#! /bin/bash
/usr/local/bin/network-radio
/usr/local/bin/flirrestart
cd /usr/local/GCSSW/AerostatGCS
/usr/local/bin/aerostatgcs
```

- **Function:** This script facilitates connectivity via a wireless radio link. Upon execution, an initial connection window appears, followed by the loading of the Aerostat GCS Telemetry version 1.0. Notably, in this mode, connectivity is often indicated as not established, with the connectivity status displayed in red.
- **Upon launching the below Screens are loaded:**





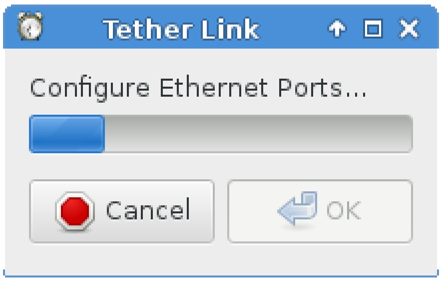
6.7.1.4. Script: TetherGCS_new

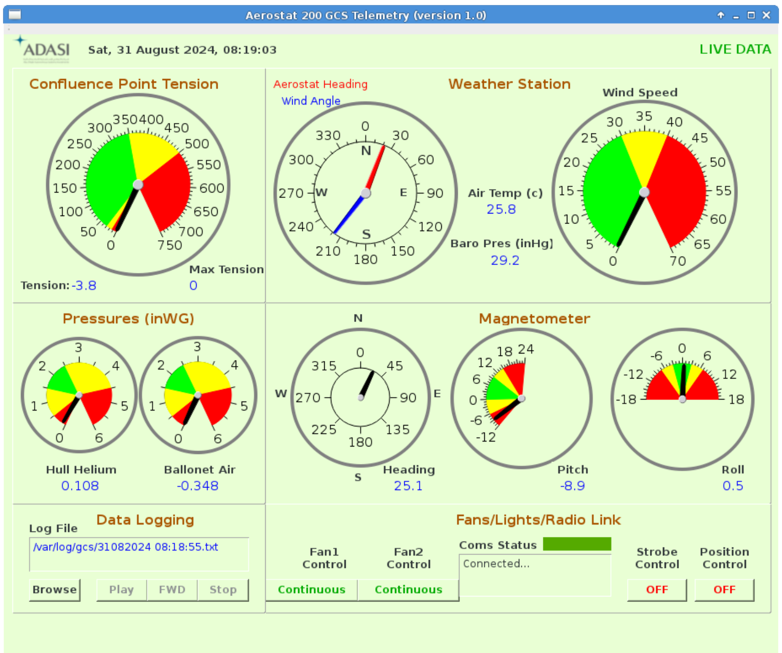
- **Shortcut Target:** /usr/local/bin/tether_ver2.1
- **Script Details:**

bash Copy code

```bash
#! /bin/bash
/usr/local/bin/network-tether
/usr/local/bin/flirrestart
cd /usr/local/GCSSW/AerostatGCS
/usr/local/bin/aerostatgcs
```

- **Function:** Similar to the TetherGCS script, this variant attempts to use the newer version of the tethered telemetry software. Despite the version designation in the script, the GUI loaded remains version 1.0, indicating that there may be inconsistencies or misconfigurations in the system.
- **Upon launching the below Screens are loaded:**





6.8. Relevant Directories and Executables

The following directories and executables were identified as critical for the operation of the Aerostat GCS Telemetry Software:

- **Source Code Directory:** /home/aerostatgcs2/Desktop/AerostatGUI_GCS_ver2.1 Fedora_Fixed CMD - This directory houses the C++ source files for the telemetry software.
- **Executable Scripts Directory:** /usr/local/bin/ - Contains various scripts for launching telemetry and FLIR recording applications.
- **Software Versions Directory:** /usr/local/GCSSW/ - Holds the different versions of the AerostatGCS Applications.

The key executables initiated by the scripts include:

1. /usr/local/bin/aerostatgcs
2. /usr/local/GCSSW/AerostatGCS/AerostatGCS
3. /usr/local/bin/network-tether
4. /usr/local/bin/flirrestart
5. /usr/local/bin/radio_ver2.1
6. /usr/local/bin/tether_ver2.1

These scripts and executables form the core of the legacy telemetry system's operation, each serving specific connectivity and data retrieval functions essential for real-time monitoring and control of the Aerostat system. The noted observations and inconsistencies will need to be addressed in the development of the new AlfaOps software to ensure enhanced functionality and seamless user experience.

7. Problem Statement

The legacy ADASI A200 Aerostat System, while still operational, faces significant limitations and challenges that impede its effectiveness, adaptability, and readiness for future enhancements. These issues necessitate the development of the new AlfaOps Command and Control (C2) software for Alfa 6 Aerostat Systems. The key challenges and limitations identified are as follows:

1. **Outdated Software Architecture**: The legacy system is based on an outdated software architecture using older technologies and operating systems like Fedora 19, which are no longer supported. This creates vulnerabilities in security and stability and limits the ability to update or integrate newer technologies seamlessly. The system is not future-proof, making it challenging to adapt to evolving mission needs or technological advancements.

2. **Limited Integration Capabilities**: The current software is heavily reliant on specific, legacy hardware configurations and serial-to-Ethernet converters, which restricts its integration versatility. It struggles to incorporate newer sensor technologies, advanced data acquisition systems, and Internet of Things (IoT) devices without extensive modifications, limiting the system's scalability and adaptability to diverse operational environments.

3. **Lack of Advanced Data Management and Analytics**: The legacy system's data management is rudimentary, lacking advanced capabilities for data lifecycle management, real-time analytics, or machine learning integration. This limits the ability to leverage artificial intelligence (AI) and computer vision technologies, which are increasingly critical for enhancing operational awareness and decision-making. Additionally, the current data logging and archiving methods are not robust enough to support comprehensive mission analysis or historical data review.

4. **User Interface and Usability Challenges**: The existing graphical user interface (GUI) is outdated and lacks the modern usability and interactivity features expected in contemporary systems. It provides limited data visualization and customization options, making it difficult for operators to intuitively interact with the system or respond quickly to changing mission conditions. This also poses a barrier to training new operators efficiently.

5. **Performance and Reliability Issues**: Performance bottlenecks, reliability concerns, and frequent software crashes are prevalent due to the reliance on legacy hardware and software components. These issues can lead to operational delays, inaccurate sensor readings, and, in some cases, mission failures. The system's inability to consistently maintain high performance under varying conditions diminishes overall operational effectiveness.

6. **Lack of Modular, Scalable, and Future-Proof Design**: The legacy software is monolithic and lacks a modular architecture, which significantly complicates efforts to upgrade, scale, or customize the system. Adding new features, adapting to different payloads, or scaling operations requires substantial redevelopment, increasing costs and turnaround times. A future-proof design is necessary to accommodate rapid technological advancements and evolving mission requirements.

7. **Operational Constraints and Manual Processes**: Many routine operations, such as system calibrations, sensor diagnostics, and data logging management, require manual intervention. This increases the risk of human error and operational inefficiencies, especially during high-pressure mission scenarios. Automation and streamlined workflows are essential to minimize operator workload and improve system resilience.
8. **Incompatibility with Modern Technologies**: The current system architecture is incompatible with modern technological advancements such as AI-driven analytics, computer vision technologies (CVT), and other emerging IoT solutions. This limits the potential for enhancing situational awareness, automating data analysis, or integrating predictive maintenance capabilities, all of which are critical for modern aerostat operations.
9. **Dependence on Specific Hardware and High Maintenance Costs**: The legacy system's tight coupling with specific hardware components, such as the GETAC X500 Laptop and MOXA Serial Servers, limits flexibility in hardware selection and increases maintenance complexity and costs. Replacing or upgrading these components is difficult, often requiring complete system revalidation, further extending downtimes and operational costs.
10. **Inadequate Remote Accessibility and Control**: The legacy system lacks robust remote accessibility features, limiting its control and monitoring capabilities to the immediate vicinity of the Ground Control Station (GCS). This constraint hampers the ability to conduct operations or diagnostics remotely, which is increasingly necessary in modern, distributed operational environments. Without remote access, operators cannot perform system checks, troubleshoot issues, or update configurations from a distance, leading to increased downtime and the need for on-site personnel, thereby driving up operational costs and complexity.

8. Scope of Work

8.1. Overview

The scope of work outlines the requirements and deliverables for developing the AlfaOps Command and Control (C2) Software. The new software will serve as an advanced, versatile, and future-proof solution for managing and operating Alfa 6 Aerostat Systems. AlfaOps will overcome the limitations of the legacy system by incorporating modern technology stacks, seamless integration capabilities, and enhanced user interfaces. The development will follow an Agile approach, allowing for incremental builds that progressively introduce new features and capabilities.

Objectives

The main objectives of the AlfaOps C2 Software development are as follows:

1. **Modernization and Versatility:** Develop a software architecture that is modular, scalable, and adaptable to various aerostat configurations and operational scenarios.
2. **Enhanced User Interface:** Create an intuitive and user-friendly GUI that improves operator efficiency, provides real-time data visualization, and supports advanced analytics.
3. **Integration with New Technologies:** Ensure compatibility with AI algorithms, computer vision technologies, IoT devices, and modern sensor arrays to enhance operational capabilities and system intelligence.
4. **Improved Data Management:** Implement robust data lifecycle management, including data acquisition, processing, storage, and retrieval functionalities, to support comprehensive data analytics and mission replay capabilities.
5. **Seamless Communication:** Facilitate reliable communication between the Ground Control Station (GCS) and aerostat avionics, including the integration of UDP/IP protocols, fiber optics, and RF links to support data integrity and reduce latency.
6. **Future-Proof Design:** Design a software solution that can be easily updated or upgraded to accommodate future technological advancements and changing mission requirements.
7. **Enhanced Security:** Implement security measures to protect data integrity, ensure secure communication channels, and prevent unauthorized access to critical systems.

8.2. Agile Project Management Approach

The development of AlfaOps will be managed using Agile project management principles, allowing for iterative development, stakeholder feedback, and flexibility in accommodating changes. The project will be divided into three main builds, each with specific goals and deliverables.

8.2.1. Build 1: AlfaOps on Legacy Hardware

Objective: Develop and deploy AlfaOps software on the existing legacy hardware and systems.

Features:

- Basic Command and Control functions compatible with the current hardware setup.
- Initial GUI design reflecting legacy system capabilities with improved user interaction.
- Integration of existing sensors and telemetry systems with UDP/IP communication.

Timeline: 1 month.

Deliverables:

- Operational AlfaOps software compatible with legacy hardware.
- Basic user interface with essential telemetry display and control features.
- Preliminary testing and validation on legacy systems.

8.2.2. Build 2: Updated Hardware and Sensors

Objective: Enhance AlfaOps to operate on upgraded hardware and incorporate modern sensors.

Features:

- Integration of new hardware components, including advanced avionics and sensors.
- Enhanced data processing and visualization capabilities.
- Improved system performance and reliability with updated hardware configurations.

Timeline: 2 months.

Deliverables:

- Updated AlfaOps software fully integrated with new hardware and sensors.
- Expanded GUI with advanced data visualization and control options.
- Full system testing, including performance benchmarking on the updated platform.

8.2.3. Build 3: Advanced Features and AI Integration

Objective: Develop advanced features, including AI integration, payload management, and sophisticated data analytics.

Features:

- Integration with AI and machine learning algorithms for predictive maintenance, anomaly detection, and enhanced decision-making.
- Full payload control capabilities, including video feeds, environmental monitoring, and data-driven mission planning.
- New features such as automated flight path adjustments, environmental adaptations, and intelligent alert systems.

Timeline: 3 months.

Deliverables:

- Final AlfaOps software with full AI integration and advanced feature set.
- Comprehensive GUI supporting complex operations and data management.
- Complete system validation, including field trials and operational testing with payloads.

8.3. Key Deliverables

1. **Software Architecture Design:** A detailed design document outlining the architecture of AlfaOps, including module specifications, data flow diagrams, and integration points with existing hardware and systems.
2. **User Interface Development:** A fully functional and aesthetically pleasing user interface that supports multi-display setups and provides operators with customizable dashboards.
3. **Integration with Avionics Suite:** Full integration of AlfaOps with the Alfa 6 Aerostat Systems' avionics suite, ensuring seamless communication with all onboard sensors and devices.
4. **Testing and Validation:** Comprehensive testing protocols to validate system performance, reliability, and compliance with operational requirements. This includes system integration testing, user acceptance testing, and field trials.
5. **Documentation and Training:** Complete documentation of the software, including user manuals, maintenance guides, and training materials. Training sessions will be provided for operators and maintenance personnel to ensure smooth transition and adoption.
6. **Maintenance and Support Plan:** A post-deployment support plan that includes software maintenance, updates, and technical support to address any issues that arise during the operational lifecycle of the AlfaOps software.

8.4. Technical Requirements

1. **Platform Compatibility:** The software must be compatible with existing GETAC X500 laptops and any future planned hardware upgrades.
2. **Operating System:** Develop AlfaOps to operate on a secure and stable Linux-based OS, with Fedora as the preferred distribution due to its current use in the legacy system.
3. **Programming Language:** Use modern, secure, and widely supported programming languages such as Python, C++, or Java for the development of AlfaOps components.
4. **Database Management:** Implement a reliable database management system capable of handling large volumes of telemetry and mission data, with support for data redundancy and backup.
5. **Network Protocols:** Utilize UDP/IP for primary data transmission, with fallback mechanisms for RF communication to ensure continuous operation in various network conditions.
6. **Scalability:** The system must be designed to scale, allowing for the addition of new sensors, data inputs, and functional modules without requiring extensive rework.

8.5. Functional Requirements

The functional requirements define what the AlfaOps software must do to meet the operational needs of the Alfa 6 Aerostat Systems. Key functional requirements include:

1. **Command and Control Operations:**
    - Provide real-time control of aerostat launch, flight, and recovery operations.
    - Support remote control and monitoring of avionics and payload systems.
    - Enable command execution for system status checks, calibration, and emergency procedures.

2. **Data Acquisition and Display:**
   - Collect, process, and display telemetry data from multiple sensors, including weather, pressure, tension, and heading data.
   - Present data in a user-friendly graphical interface, with options for customization based on operator preferences.
   - Provide data logging, replay, and analysis capabilities for mission review and troubleshooting.
3. **System Integration:**
   - Integrate seamlessly with legacy hardware, including existing avionics suites and ground control station equipment.
   - Support integration with new sensors, payloads, and communication technologies in future builds.
   - Facilitate bi-directional communication with onboard systems via Ethernet, fiber optics, and RF links.
4. **Security and Access Control:**
   - Implement user authentication and role-based access control to safeguard sensitive operations and data.
   - Ensure secure communication channels between the GCS and aerostat systems to prevent unauthorized access and data breaches.
5. **Alerts and Notifications:**
   - Generate alerts for system anomalies, environmental thresholds, and operational status changes.
   - Provide audible and visual cues for critical warnings and alerts, with options for operator acknowledgment.

8.6. Non-Functional Requirements

Non-functional requirements define the performance, reliability, and usability standards for the AlfaOps software:

1. **Performance:**
   - The software must process and display telemetry data with minimal latency, ensuring real-time operation.
   - System response time for command execution should be under 1 second for critical operations.
2. **Reliability and Availability:**
   - AlfaOps must be available 99.9% of the time during mission-critical operations, with built-in redundancies to prevent downtime.
   - The system should be robust against common failure modes, including hardware malfunctions and network interruptions.
   -

3. **Scalability:**
   - The software architecture must support scaling to accommodate additional aerostat systems, sensors, and payloads without significant rework.
   - AlfaOps should be able to handle increased data throughput as new devices are integrated.
4. **Usability:**
   - The user interface must be intuitive, with minimal training required for operators to perform standard tasks.
   - Provide contextual help and tooltips within the GUI to assist operators in understanding system functions and alerts.
5. **Compatibility:**
   - AlfaOps must be compatible with existing legacy hardware and software, including operating systems and communication protocols.
   - Ensure support for a wide range of third-party devices, sensors, and future technologies.

8.7. Quality Requirements

Quality requirements focus on ensuring the software meets high standards of development and operational excellence:

1. **Maintainability:**
   - The software codebase should be modular, well-documented, and designed to facilitate easy updates and bug fixes.
   - Implement automated testing and continuous integration practices to ensure code quality and rapid iteration.
2. **Portability:**
   - The software must be portable across different hardware platforms, allowing for easy deployment on upgraded GCS hardware or other operating environments.
   - Ensure compatibility with multiple Linux distributions to accommodate diverse deployment scenarios.
3. **Security:**
   - Implement robust encryption for data at rest and in transit to protect sensitive information.
   - Regularly update and patch software components to protect against known vulnerabilities.
4. **Compliance:**
   - AlfaOps must comply with relevant industry standards and regulations for aerostat command and control systems, including data privacy and security guidelines.

   - Ensure documentation and reporting capabilities align with regulatory requirements for defence and aerospace systems.

5. **Resilience:**
   - Design AlfaOps to handle adverse operational conditions, including extreme weather, signal interference, and hardware failures.
   - Include fallback mechanisms to maintain control and data integrity during unexpected disruptions.

8.8. Key Performance Indicators (KPIs)

To measure the success of AlfaOps, the following KPIs will be monitored and evaluated:

1. **System Uptime:** Targeting 99.9% availability during mission-critical operations.
2. **Data Latency:** Real-time telemetry data updates with a maximum delay of 1 second.
3. **Command Response Time:** Less than 1 second for critical commands.
4. **User Satisfaction:** Positive feedback from operators on usability and functionality during testing and deployment phases.
5. **Incident Rate:** Minimal incidents of system failures or data inaccuracies during operations.

8.9. Implementation Timeline

A detailed project timeline will be provided, outlining key milestones from initial design through to deployment and testing. The timeline will include:

- **Phase 1:** Requirements Gathering and System Design
- **Phase 2:** Development and Initial Testing
- **Phase 3:** Integration and System Testing
- **Phase 4:** User Training and Acceptance Testing
- **Phase 5:** Deployment and Go-Live
- **Phase 6:** Post-Deployment Support and Maintenance

8.10. Acceptance Criteria

The project will be deemed successful and completed upon:

1. Demonstration of full functionality as per the agreed requirements in real-world operational scenarios.
2. Completion of all testing phases with results that meet or exceed the predefined success metrics.
3. Delivery of all documentation and training materials to the satisfaction of Alfa 6 LTA Systems.
4. A signed-off user acceptance test (UAT) by Alfa 6 LTA Systems, indicating that the software meets operational expectations.