

system-requirements

SYS-001: Telemetry Data Rate

Rationale

This requirement is derived from BVLOS operational safety standards (EASA SC-RPAS).

Real-time telemetry is essential for situational awareness during Beyond Visual Line of Sight (BVLOS) operations. A 10 Hz minimum rate ensures the ground operator can detect anomalies within 100ms.

Consider increasing to 20 Hz for high-dynamics phases (takeoff/landing).

SYS-001

The UAV system shall transmit telemetry data to the ground station at a minimum rate of 10 Hz under normal flight conditions.

Field	Value
category	performance
priority	critical
status	active
title	Telemetry Data Rate
verified	True

SYS-002: Failsafe Behavior

SYS-002

The UAV system shall activate failsafe mode and execute a return-to-home sequence within 2 seconds of losing the ground station link (updated).

Field	Value
category	safety
priority	critical
status	active
title	Failsafe Behavior on Link Loss
verified	False

SYS-003: Ground Station Interface

SYS-003

The UAV system shall support simultaneous command and control communication with the ground station over encrypted radio links.

Field	Value
category	interface
priority	high
status	draft
title	Ground Station Command Link

system-diagrams

SYS-004

A diagram showing the high-level architecture

Field	Value
priority	medium
status	draft
title	System architecture diagram

software-requirements

REQ-001: Telemetry Pipeline

Background

The telemetry subsystem is critical for real-time monitoring of the UAV during flight. Data must be serialized and transmitted without introducing jitter into the control loops. See [SYS-001] for the system-level data rate requirement.

Requirement

REQ-001

The flight computer software shall implement a non-blocking telemetry pipeline that serializes and transmits sensor data at configurable rates without affecting the flight control loops.

Field	Value
derived	False

Field	Value
parent	SYS-001
priority	critical
related	
status	active
tag	performance, telemetry
title	Async Telemetry Serialization Pipeline
verify	Integration test: test_telemetry_rate_10hz
version	2

Design Notes

The implementation uses a lock-free ring buffer to decouple sensor sampling from the serialization thread. This ensures that even under high CPU load, telemetry frames are queued without blocking the flight controller's main loop.

REQ-002: Failsafe Handler

REQ-002

The flight computer software shall monitor the command link watchdog timer and trigger the failsafe state machine upon timeout, initiating return-to-home guidance mode.

Field	Value
derived	False
parent	SYS-002
priority	critical
related	REQ-001
status	active
tag	safety, failsafe
title	Link Loss Failsafe Handler
verify	Hardware-in-the-loop test: test_failsafe_rth_trigger
version	1

REQ-003: GCS Link Manager

Context

MAVLink v2 is used as the communication protocol between the flight computer and the ground control station. The link must support encryption (AES-256) and handle intermittent connectivity gracefully.

REQ-003

The flight computer software shall manage the encrypted MAVLink session with the ground control station, handling connection establishment, keepalive, and graceful reconnection.

Field	Value
derived	False
parent	SYS-003
priority	high
related	REQ-001, REQ-002
status	draft
tag	interface, communications
title	GCS MAVLink Session Manager
version	1

Open Questions

- Should reconnection attempts be bounded or infinite?
- What is the maximum acceptable reconnection latency?

REQ-004: GCS Contingency Control

REQ-004

The flight computer software shall implement a contingency control mode that allows the ground control station to take over manual control of the aircraft in case of a failure in the autonomous flight system (updated).

Field	Value
derived	False
parent	SYS-003
priority	high
related	
status	draft
tag	interface, communications
title	GCS Contingency Control
version	1

REQ-005: Derived Requirement without a Parent

REQ-005

Derived requirement Without a parent

Field	Value
derived	True
justification	We really need it!
priority	high
related	
status	draft
tag	interface, communications
title	Derived 005
version	1

implementation

SRC-004

```
def activate_contingency_mode(): # Transfer control to GCS pass
```

Field	Value
parent	REQ-004

SRC-002

```
def handle_failsafe(event): if event == 'LINK_LOSS': trigger_rth()
```

Field	Value
parent	REQ-002

SRC-003

```
class MAVLinkManager: def connect(self, uri): # Establish encrypted session
pass
```

Field	Value
parent	REQ-003

SRC-001

```
class TelemetryPipeline:
    def init(self, rate_hz=10):
        self.rate_hz = rate_hz
    def push(self, data):
        # Non-blocking push to queue
        pass
```

Field	Value
parent	REQ-001

verification

TEST-004

```
def test_gcs_takeover():
    # Verify manual control takeover pass
```

Field	Value
parent	REQ-004, SRC-004

TEST-002

```
def test_failsafe_rth_trigger():
    # Simulate link loss and check RTH trigger
    pass
```

Field	Value
parent	REQ-002, SRC-002

TEST-003

```
def test__mavlink_reconnection(): # Test graceful reconnection after link drop
pass
```

Field	Value
parent	REQ-003, SRC-003

TEST-001

```
def test__telemetry_rate_10hz(): pipe = TelemetryPipeline(rate_hz=10) assert
pipe.rate_hz == 10
```

Field	Value
parent	REQ-001, SRC-001