Quadrotor Drone

Core Product Requirements Document

CONFIDENTIALITY WARNING

This document contains information deemed confidential by Periscope Robotics. If you have not signed a confidentiality agreement, do not read further.

# Table of Contents

[1 Table of Contents 2](#_Toc107605506)

[2 Version History 4](#_Toc107605507)

[3 Referenced Documents 5](#_Toc107605508)

[4 Acronyms 6](#_Toc107605509)

[5 Scope & Purpose 7](#_Toc107605510)

[6 Market & Trade Studies 8](#_Toc107605511)

[7 Product Overview 9](#_Toc107605512)

[7.1 Mechanical Overview 9](#_Toc107605513)

[7.2 Hardware Overview 9](#_Toc107605514)

[7.3 Software Overview 10](#_Toc107605515)

[8 Requirements 11](#_Toc107605516)

[8.1 Product Lifecycle 11](#_Toc107605517)

[8.1.1 Expected Maintenance 11](#_Toc107605518)

[8.1.2 Mechanical Maintenance Access 11](#_Toc107605519)

[8.1.3 Software Maintenance Access 11](#_Toc107605520)

[8.1.4 Expected Calibration 11](#_Toc107605521)

[8.2 Product Performance 12](#_Toc107605522)

[8.2.1 Endurance Time 12](#_Toc107605523)

[8.2.2 Maximum Altitude 12](#_Toc107605524)

[8.2.3 Maximum Climb 12](#_Toc107605525)

[8.2.4 Payload Capacity 12](#_Toc107605526)

[8.2.5 Battery Charging Speed 12](#_Toc107605527)

[8.2.6 Communication Integrity Verification 13](#_Toc107605528)

[8.3 Core Features 13](#_Toc107605529)

[8.3.1 Wireless Communication 13](#_Toc107605530)

[8.3.2 Tethered Communication 13](#_Toc107605531)

[8.3.3 Loss of Communication 13](#_Toc107605532)

[8.3.4 Loss of Communication Behavior 13](#_Toc107605533)

[8.3.5 Loss of Power Behavior 14](#_Toc107605534)

[8.3.6 Rotor Movement Inhibit 14](#_Toc107605535)

[8.3.7 Flight Parameters 14](#_Toc107605536)

[8.3.8 Configurable Parameters 14](#_Toc107605537)

[8.3.9 Battery Unsafe Detection 14](#_Toc107605538)

[8.3.10 Weight on Wheels Detection 15](#_Toc107605539)

[8.3.11 Battery Unsafe Behavior 15](#_Toc107605540)

[8.3.12 External Power 15](#_Toc107605541)

[8.3.13 Ground Power Charging 15](#_Toc107605542)

[8.4 User Interface 16](#_Toc107605543)

[8.4.1 Battery Status Indication 16](#_Toc107605544)

[8.4.2 Accessible Flight Parameters 16](#_Toc107605545)

[8.4.3 Accessible Configurable Parameters 16](#_Toc107605546)

[8.4.4 Loss of Power Indication 16](#_Toc107605547)

# Version History

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision Name** | **Date Issued** | **Author** | **Change Summary** |
| Draft | -- | Jaden Bottemiller | -- |

# Referenced Documents

# Acronyms

|  |  |
| --- | --- |
| AMSL | Above Mean Sea Level |
| GPS | Global Positioning System |
|  |  |
|  |  |
|  |  |

# Scope & Purpose

This document is intended to give an overview of the Quadrotor drone project. It will outline the design intentions and methodology, as well as compare it to existing market solutions. Additionally, it will define requirements, from a broad perspective. These requirements primarily pertain to critical marketable features, safety, and ease of development.

This document is not intended to be a critical design document or contain requirements with specific implementation details.

# Market & Trade Studies

# Product Overview

## Mechanical Overview

## Hardware Overview

The unit electronic hardware consists of four distinct domains: battery management, motor driving, digital processing, and communication.

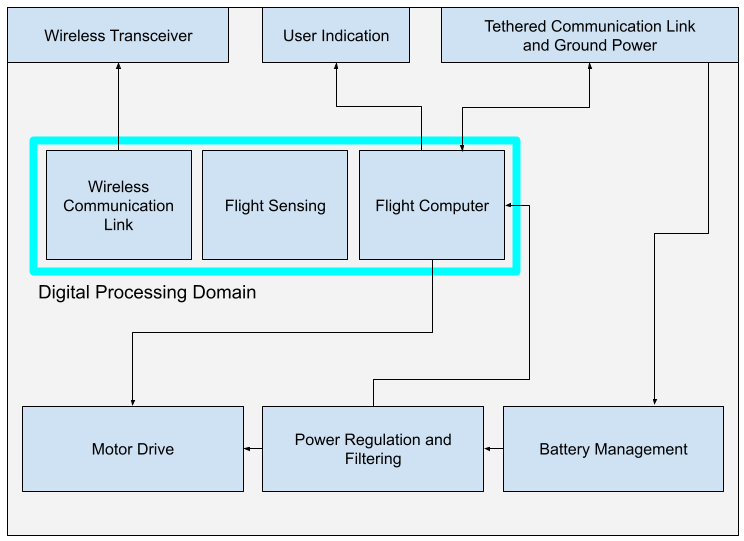


Figure - High Level Hardware Architecture

Battery management monitors and controls the charge and discharge of the battery, and allows power to flow to subsequent power stages, such as to digital electronics.

The motor driving circuit uses the input power from the battery management circuit to drive motors at the command of the digital processing domain. The flight computer will tell the motor drive circuit the power to drive each motor with.

Digital processing handles all flight computing and motion processing actions, motor driving, and communication. It also senses and controls some stages within the motor drive and battery management circuits.

The communication stage is tightly coupled with the digital processing domain; however, it specifically drives physical layer outputs.

## Software Overview

The Quadrotor Drone software interface consists of two primary features: the host application, where user input is handled and negotiated via the communication link to the actual “unit,” and the unit itself, which contains the flight computer and other oversight features.



Figure - High Level Software Architecture

The unit and the host application primarily communicate over a wireless link; however, they may communicate over a tethered communication link while on the ground. The tethered link may be required in some scenarios to update firmware, perform tests, change configuration parameters relating to the wireless link, and more.

The host application can direct the unit to different coordinates in three-dimensional space but does not perform any of the critical flight data sensing and computation.

The host application is intended to run on personal computers or other portable devices. The unit software will be embedded on an onboard microcontroller.

# Requirements

## Product Lifecycle

This section contains requirements related to the lifecycle of the product. For example: expected maintenance, frequency of maintenance, mean time between critical failures, etc.

### Expected Maintenance

Allocation: Unit

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** have an average time between maintenance activities of one thousand (1,000) flight hours.

Context: Expected maintenance does not include maintenance due to equipment mishandling by the end user.

### Mechanical Maintenance Access

Allocation: Mechanical

Safety Critical: No

Minimum Viable Product: No

In the event of any primary mechanical failure, the unit **shall** be separable and repairable by replacement of the primary mechanical feature.

Context: An example of a primary mechanical feature is a strut, or any other feature that when failed, could cause a catastrophic failure of the airframe. This requirement is based on best engineering judgement and marketability.

### Software Maintenance Access

Allocation: Software

Safety Critical: No

Minimum Viable Product: No

In the event of an update to flight software, the unit **shall** be field loadable.

Context: Flight software does not include the bootloader required to load the downloaded software into flash memory.

### Expected Calibration

Allocation: Hardware and Software

Safety Critical: No

Minimum Viable Product: No

The unit **shall** require a user-involved calibration routine a maximum of one (1) time in the product lifespan.

Context: This requirement does not include calibration required due to new software or hardware updates.

## Product Performance

This section contains requirements related to the performance of the product. For example, maximum altitude, endurance time, and payload capacity.

### Endurance Time

Allocation: Unit

Safety Critical: No

Minimum Viable Product: Yes

The unit **shall** be capable of flight for at least twenty (20) minutes immediately after the battery has been charged to its maximum safe capacity.

Context: This requirement is based on best engineering judgment, available battery technology, and marketability.

### Maximum Altitude

Allocation: Unit

Safety Critical: No

Minimum Viable Product: No

The unit **shall** be capable of flight for one (1) minute at a maximum altitude of five thousand (5,000) feet AMSL.

### Maximum Climb

Allocation: Unit

Safety Critical: No

Minimum Viable Product: Yes

The unit **shall** be capable of climbing to and maintaining an altitude three hundred (300) feet higher than the altitude of the lift-off point.

### Payload Capacity

Allocation: Unit

Safety Critical: No

Minimum Viable Product: No

The unit **shall** be capable of lifting an additional one hundred fifty (150) grams of payload without performance degradation.

### Battery Charging Speed

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: Yes

The unit **shall** be capable of charging its battery to its maximum safe capacity in one (1) hour or less.

Context: This requirement is based on best engineering judgement and available technology.

Allocation Justification: This requirement was allocated to hardware because hardware interfaces directly with the battery and the charge control circuitry.

### Communication Integrity Verification

Allocation: Software

Safety Critical: Yes

Minimum Viable Product: Yes

The unit and host application **shall** ignore any incoming messages that do not pass an integrity check.

Context: This is required to prevent potentially unsafe conditions from being accidentally commanded through the communication link.

## Core Features

This section contains features deemed to be vital to the core functions of the product. For example, means of communication, acceptable failure modes, etc.

### Wireless Communication

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** be capable of communicating wirelessly to a host application up to at least five hundred (500) feet.

Context: This requirement is based on ideal conditions for wireless communication and best engineering judgement.

### Tethered Communication

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** have a means of tethered communication on the ground.

### Loss of Communication

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** be able to detect a loss of communication between the device and the host application.

### Loss of Communication Behavior

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: No

After a detected loss of communication while the aircraft is determined to be airborne as described in [Weight on Wheels Detection](#_Weight_on_Wheels), the unit **shall** perform a home base approach and landing until communication is regained and determined to be stable.

### Loss of Power Behavior

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

In the event of a critical loss of power, the unit **shall** execute a landing at its current position.

### Rotor Movement Inhibit

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** inhibit high energy movement of the rotors while the device is tethered to the host application.

### Flight Parameters

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: No

The unit **shall** be able to record one (1) flight’s worth of flight parameters in non-volatile memory.

### Configurable Parameters

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: No

The unit **shall** be able to record, at a minimum, the following configurable parameters in non-volatile memory:

* home-base GPS coordinate
* communication parameters
* battery characteristic parameters

Context: Configurable parameters are required for homing, communication, and more. An example of a communication parameter would be a wireless network name and password. This list is based on best engineering judgement.

### Battery Unsafe Detection

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** have a mechanism to detect the following unsafe battery conditions:

* Over voltage
* Under voltage
* Over discharge
* Out of range temperature

### Weight on Wheels Detection

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The unit **shall** have a mechanism to detect whether the aircraft is in the air or on the ground.

Context: This property is sometimes referred to as “weight on wheels,” or W.O.W.

### Battery Unsafe Behavior

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

If an unsafe battery condition is detected as defined in [Battery Unsafe Detection](#_Battery_Unsafe_Detection), the unit **shall** limit discharge to levels considered safe. Safe levels of discharge depend on the weight on wheels state defined in [Weight on Wheels Detection](#_Weight_on_Wheels).

Context: If the aircraft is in air, then the battery must discharge enough to make a safe landing. If the aircraft is on ground, high current discharge to the motors may be completely limited.

### External Power

Allocation: Hardware

Safety Critical: No

Minimum Viable Product: Yes

The unit **shall** accept an external power input for charging through the same physical interface as the tethered communication link.

Context: This is required to charge the battery.

### Ground Power Charging

Allocation: Hardware and Software

Safety Critical: No

Minimum Viable Product: Yes

The unit **shall** automatically begin the battery charge routine if every following condition is met:

* the battery is determined to be in a safe condition as defined in [Battery Unsafe Detection](#_Battery_Unsafe_Detection),
* the vehicle is determined to be on ground as defined in [Weight on Wheels Detection](#_Weight_on_Wheels),
* external power has been applied

## User Interface

### Battery Status Indication

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: Yes

The unit **shall** have a mechanism to, without any external device, plainly display to the user, at a minimum:

* the state of the charge circuit
* the remaining capacity of the battery
* whether the battery is being limited due to unsafe conditions detected in [Battery Unsafe Detection](#_Battery_Unsafe_Detection)

Context: An example of a feature that would satisfy the requirement is a series of ten (10) LEDs where they indicate fully charged when they are all on, ninety (90) percent charged when nine (9) of them are on, and so forth. Requiring the unit to be plugged into a personal computer would not satisfy the requirement.

### Accessible Flight Parameters

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: No

The unit **shall** have a mechanism to retrieve (read) recorded flight parameters without major disassembly.

### Accessible Configurable Parameters

Allocation: Software and Hardware

Safety Critical: No

Minimum Viable Product: No

The unit **shall** have a mechanism to retrieve and change configurable parameters without major disassembly.

### Loss of Power Indication

Allocation: Software and Hardware

Safety Critical: Yes

Minimum Viable Product: Yes

The host application **shall** indicate a warning to the user if the unit is near the point of no return.

Context: The point of no return is defined as the point at which the remaining capacity of the battery is no longer enough to return to the home-base point or point of lift off.