Final Report

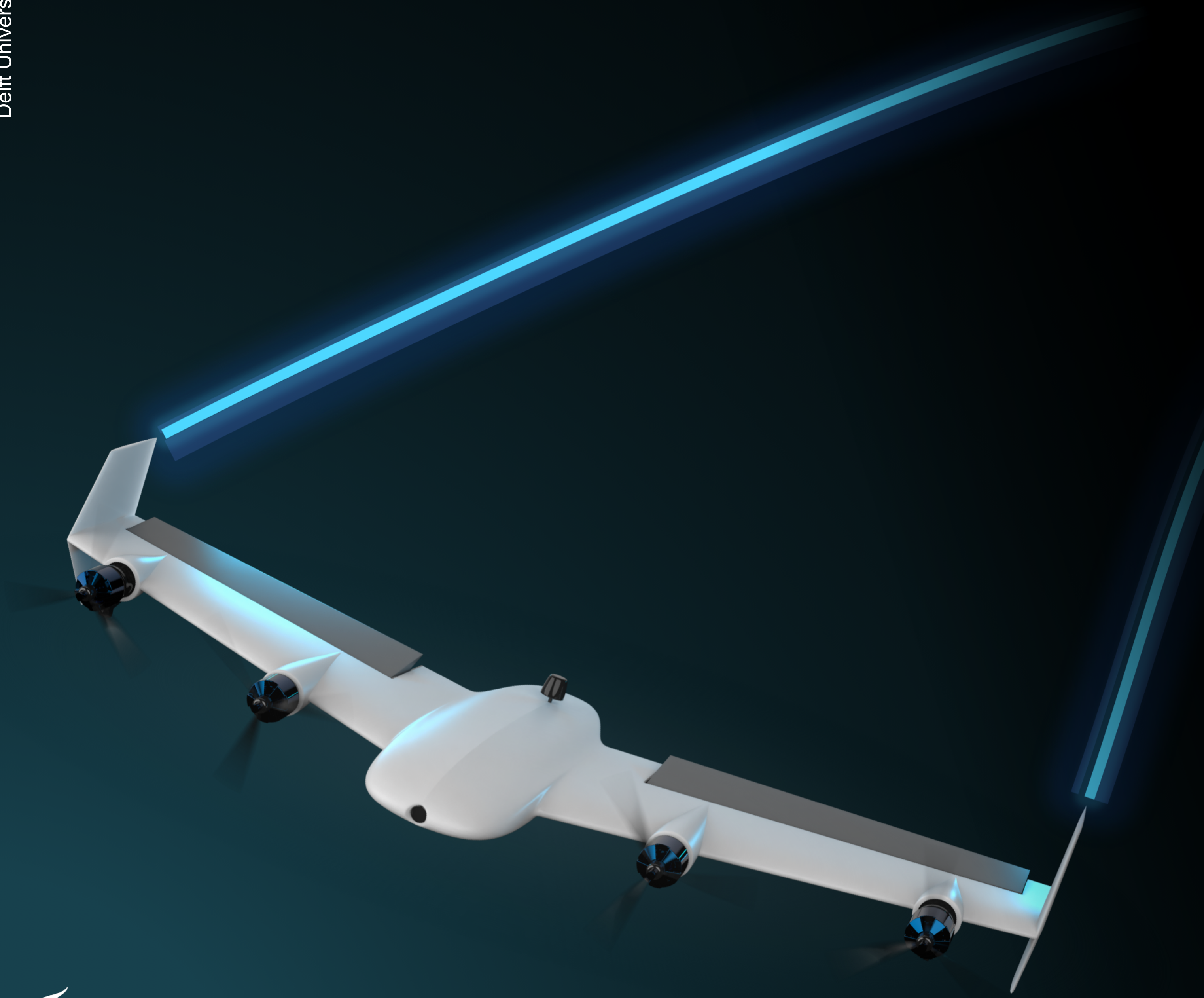
Peregrine Racing Drone

DSE Group 24
High Speed Racing Drone

June 29, 2021

Delft University Of Technology

Delft University Of Technology



TUDelft

Final Report

Peregrine Racing Drone

by

DSE Group 24
High Speed Racing Drone

Name	Student Number
Ammar Saeed	4535731
Casper van Wezel	4863755
Eric Tüschenbönner	4858565
Knut Roar Lende	4785525
Lucas Cohen	4815742
Martin Barragan	4832191
Nikolaos Kalliatakis	4848098
Noah Wechtler	4779649
Tido Houtepen	4854691
Yash Sinha	4794265

In partial fulfilment of the requirements for the degree of
Bachelor of Science, Aerospace Engineering
at Delft University of Technology.

Project Duration:	April 19, 2021 – July 4, 2021	
Tutors:	Dr. ir. Ewoud Smeur,	TU Delft, supervisor
	Dr. Nan Yue,	TU Delft, coach
	M.Sc. Ruipeng Liu	TU Delft, coach

TU Delft

Preface

This report marks the completion of the Design Synthesis Exercise, a 10 week project conducted at the Faculty of Aerospace Engineering of Delft University of Technology. Over the duration of the Design Synthesis Exercise, a high speed racing drone by the name of Peregrine has been extensively designed by 10 students. After performing an analysis on the aerodynamics, structures and materials, power and propulsion, and control and stability subsystems, the performance of the drone was simulated. Along with technical tasks, organisational and project management and system engineering tasks were conducted as well.

The team would like to acknowledge all the help received from the tutor, coaches and the external experts that were able to offer guidance when needed. In no particular order we thank:

Tutor:	Dr.ir. E.J.J. Smeur
Coach:	Dr. N. Yue
Coach:	M.Sc. R. Liu
Expert:	Dr.ir. T. Sinnige
Expert:	Dr.ir. AH van Zuijlen
Expert:	Dr.ir. G. la Rocca
Expert:	B.Sc. S. Schröter
Expert:	Dr. M. Bronz

DSE Group 24 - Peregrine
A. Saeed, C. van Wezel, E. Tüschenbönner, K.R. Lende, L. Cohen
M. Barragan, N. Kalliatakis, N. Wechtler, T. Houtepen, Y. Sinha

Delft, June 2021

Executive Overview

This report marks the culmination of a 10-week project to design a high-speed racing drone with the aim to deliver world-class performance while maintaining manoeuvrability characterstics similar to existing racing drones. In the initial phases of the development, several wing-based designs were conceptualised to achieve this aim. By means of a trade-off based on the most important aspects agreed on by the team, one design concept was chosen. This concept is explored and analysed throughout this report, leading to a detailed final design.

Defining the System

After reviewing the key traits of this concept, the resources and budgets for the different subsystems were re-defined to be used in the subsequent analyses. In addition, the system functions were revised such that operation and maintenance tasks more specific to the current design could be outlined and used in subsystem analysis.

A closer look at the current market confirmed the need for a high-speed drone that could accommodate to both veterans and inexperienced customers in the rapidly growing market. Competition indicated a lack of flying wings, a market gap for Peregrine to exploit. Future market predictions result in Peregrine obtaining an expected market share of 7.5% after 1-2 years of its introduction.

Analysis of Subsystems

To minimise the drag of the drone, first the airfoil needed to be selected. This airfoil was selected such that the wing drag was minimised. For this the NACA 0012 proved to be optimal. Then the planform was determined by analysing the effect of changing the sweep and taper and choosing the optimal combination. Finally several other subystems had to be integrated into the aerodynamic model. The electronics were housed in pods modeled as longer parts of the wing. The size of the elevons were verified, a first order estimate of the stability derivatives was found, the winglets were sized and a preliminary analysis on flutter was done.

In terms of Structures & Materials, to fulfill the mass budget, the structure of Peregrine had to be renewed to include both carbon fiber reinforced polymers (CFRP) of thicknesses 1 [mm] and 3 [mm] and expanded polypropylene foam (EPP) skin of 3 [mm]. Analysis confirmed that having the CFRP act as the load bearing structure whilst having the EPP act as a cushion for impacts was a viable strategy and validated performance requirements with peak deflections of 15 [mm] during critical flight and no plastic deformation. Further analysis into impact scenarios and fatigue still needs to be conducted for proper validation due to uncertainties regarding EPP's suitability given the lack of available data.

To evaluate the performance of the propellers on-board Peregrine, momentum blade-element theory was used. The airfoil used in the propeller analysis is the SG6043. To estimate the battery mass, the UTT1 track was chosen and a race profile was assumed. The outcome of this analysis is that the battery will weigh 169.6 [g] and will have a capacity of 23.31 [Wh]. The propeller that achieved this battery mass has a diameter of 12.8 [cm] and delivers a thrust of 3 [N] at 86 [m/s]. Additionally, the performance required from the motors and actuators were analysed. Finally, The airfoil analysis results and performance graphs from the momentum blade-element theory were verified and validated through unit and system testing and experimental data.

As part of the Control & Stability subsystem, stability characteristics and controllability of attitude rates due to thrust vectoring, differential thrust and elevons were analysed. The latter were sized to fulfil the high manoeuvrability requirements of the drone at high speeds, while thrust vectoring is reserved to maintain agility at lower speeds and stabilise in hover. Moreover, the control system for cruise was designed to be able to effectively and intuitively fly the drone, and implemented in a Simulink simulation. When it comes to hover and cruise, a control logic was established but not yet further developed. Using the above methods, the majority of requirements regarding this subsystem have been verified.

The Performance & Simulation subsystem was concerned with combining the results of all the other subsystems to get Peregrine's final performance, and then put that in a simulation. The performance aspect involved four distinct analysis, namely straight line acceleration, braking performance, angular rates, and minimum turn radius, where it was found that the performance was as expected and met or exceeded all of the requirements. Then, a simulation was also created in which all of these expected performance characteristics could be actually verified. Again, with the help of the simulation, it was found that Peregrine met all of the requirements which had to be verified through simulation, such as speed and turning performance.

System Analysis

Following the analysis of the five individual subsystems, a more complete system analysis was conducted that encompasses risk and sustainability before fully integrating the system.

In order to ensure that there are no large risks that pose a big threat to the project, it is important to have a proper risk management procedure which was performed in the Risk section. This involved identification of all relevant risks, then assessment of these risks and finally a mitigation strategy was devised such that all risks were within acceptable limits. Risk maps were also constructed to visually aid with this task, and it was found that there are no risks that exceed acceptable limits.

To determine the total environmental impact that production of Peregrine will have, life cycle assessment was performed using openLCA software and relevant databases. Material composition with their respective masses was determined. End-of-Life strategies were formed and the recyclability percentages determined from research and literature to conclude that the system achieves a recyclability of 80.41 %.

To ensure the final drone design is fully functional, system integration is performed through the means of a CAD model. This model demonstrates that all the subsystems and components fit within the aerodynamic skin. The final parameters of the design are given in Table 1, and shown in Figure 1.

Table 1: Final Design Parameters for Peregrine

Parameter	Value	Unit
Mass	0.4996	kg
T_{max} per engine	4.55	N
T_{max} reverse per engine	-1.9	N
$C_{L_{MAX}}$ at 60 m/s	0.9906	-
$C_{L_{MAX}}$ at stall	0.745	-
V_{stall}	13.5	m/s
C_{D_0}	0.0082	-
Maximum Load Factor	26.2	-
x_{cg}	0.103877	m

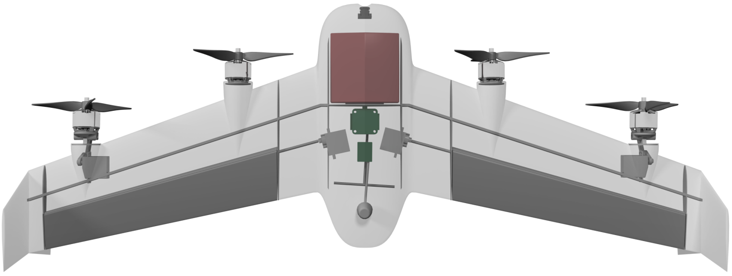


Figure 1: Top View of the Peregrine, including internal structure

Verification & Validation

After integrating the subsystems into the final design, the system level requirements were verified. The labels Pass, To Be Confirmed (TBC) and Fail were assigned. The Passed requirements were verified on analysis methods, and future verification procedures were proposed on the ones marked with TBC. Four requirements received the label Fail, although these requirements do not hinder the final system performance.

Continuation of the Project

Having fully defined the system design outcome until now, plans for potential activities beyond the DSE and their timeline have been crafted. The activities can be grouped into four different phases. The first phase is the corporation development which will build the company through searching partners, coming up with a corporate identity and setting up customer support. Parallel to this corporation building, the product will be developed, produced and sold. The product development consists primarily of increasing the detail of the analysis and design and building and testing prototypes. The production phase will be concerned with looking for the suppliers and executing the production plan. Finally the operations phase will be concerned with updating the drone and making sure there is enough stock.

Contents

List of Symbols

Abbreviations

Abbreviation	Definition
AC	Aerodynamic Center
CG	Center of Gravity
CFRP	Carbon Fibre-Reinforced Polymer
DOD	Depth of Discharge
DSE	Design Synthesis Exercise
EG	Ethylene Glycol
EPP	Expanded Polypropylene
GHG	Greenhouse Gases
IMU	Inertial Measurement Unit
ISA	International Standard Atmosphere
LCA	Life Cycle Assessment
LCIA	Life Cycle Impact Assessment
LE	Leading Edge
LiPo	Lithium-ion Polymer
MAC	Mean Aerodynamic Chord
MMOI	Mass Moment of Inertia

Abbreviation	Definition
NOx	Nitrogen Oxides
NP	Neutral Point
PAN	Polyacrylonitrile
PET	Polyethylene Terephthalate
PTFE	Polytetrafluoroethylene
RANS	Reynolds-Averaged Navier-Stokes
RPM	Rotations Per Minute
RPS	Rotations Per Second
SOx	Sulfur Oxides
TBC	To Be Confirmed
TE	Trailing Edge
TFE	Tetrafluoroethylene
TRL	Technological Readiness Levels
TVL	TV Lines
VLM	Vortex Lattice Method
WMH	Waste Management Hierarchy

Symbols

Symbol	Definition	Unit
a	Axial interference factor	[-]
a'	Radial interference factor	[-]
A	Cross-sectional area	[m^2]
A	Aspect ratio	[-]
b	Wing span	[m]
b_1	Inboard spanwise location of elevon	[m]
b_2	Outboard spanwise location of elevon	[m]
B	(PP) Number of blades	[-]
B	(CS) Control effectiveness matrix	[-]
c	Chord length	[m]
C	(SM) Constant in the Sendeckyj fatigue model	[-]
$\bar{c}$	Mean aerodynamic chord length	[m]
$\bar{c}_e$	Mean aerodynamic chord elevon	[m]
C_D	Drag coefficient	[-]
C_{D_0}	Zero lift drag coefficient	[-]
C_h	(CS) Hinge moment coefficient coefficient	[-]
C_l	(AE) Airfoil lift coefficient	[-]
C_l	(CS) Rolling moment coefficient	[-]
C_L	Wing lift coefficient	[-]
C_{l_p}	Roll damping coefficient	[s/rad]
C_{L_α}	Lift gradient coefficient	[/rad]
C_m	Pitching moment coefficient	[-]
C_{m_alpha}	Pitching moment slope	[-]
C_{m_q}	Pitch damping coefficient	[s/rad]
C_n	Yawing moment coefficient	[-]
C_{n_r}	Yaw damping coefficient	[s/rad]
C_{n_β}	Weathervane stability coefficient for winglet	[-]
C_P	Power coefficient	[-]
C_T	Thrust coefficient	[-]
C_x	Torque force coefficient	[-]
C_y	Thrust force coefficient	[-]
D	Drag	[N]
D	Diameter	[m]
D^*	Contracted slipstream diameter	[m]
e	Oswald efficiency factor	[-]
E	Energy	[J]
F	Prandtl momentum loss factor	[-]
F	Force	[N]
G	Circulation function	[-]
I	Second Moment of Area	[m^4]
I	Current	[A]
l	Rolling moment	[Nm]
l_v	Elevon arm	[m]
L	Lift	[N]
m	Pitching moment	[Nm]
M	Moment	[Nm]
MI	Mass Moment of Inertia	[kg m^2]
n	(CS) Yawing moment	[Nm]
n	(PP) Rotations per second	[s^{-1}]
n	(SM) Load factor	[-]
N_f	Number of fatigue cycles	[-]
n_s	Propeller speed	[s^{-1}]
P	Power	[W]
p	Roll rate	[rad/s]

Symbol	Definition	Unit
p_g	Geometric Pitch	[m]
q	Pitch rate	[rad/s]
Q	Propeller torque	[Nm]
r	Yaw rate	[rad/s]
r	(PP) radial coordinate	[m]
R	Turn radius	[m]
R	(SM) Minimum stress over maximum stress	[-]
Re	Reynolds number	[-]
S	Wing surface area	[m^2]
S	(SM) Constant in the Sendeckyj Fatigue Model	
S_e	Surface are of elevon	[m^2]
S_v	Wing surface winglet	[m^2]
t	Thickness	[m]
T	Thrust	[N]
ΔT	Differential Thrust	[N]
V	Velocity	[m/s]
V	Voltage	[V]
W	Weight	[N]
W	(PP) Local total velocity	[m/s]
x	X-position from the leading edge	[m]
$\bar{x}$	X-position normalised by MAC	[-]
x_{ac}	X-position of aerodynamic center	[m]
x_{cg}	X-position of center of gravity	[m]
x_{LE}	X-position of leading edge	[m]
x_n	X-position of neutral point	[m]
x_o	X-position of outboard propellers	[m]
y	Spanwise position from the center	[m]
y_i	Y-position of inboard propellers	[m]
y_o	Y-position of outboard propellers	[m]
α	Angle of attack	[rad]
α	Angular acceleration	[rads^{-2}]
β	Sideslip angle	[rad]
β_p	Propeller lade angle	[rad]
$\beta_{0.75}$	Blade angle at 0.75R of propeller	[rad]
δ	(CS) Elevon deflection	[rad]
ϵ	Drag to lift ratio	[-]
ζ	Displacement velocity	[-]
η	Efficiency	[-]
θ	(CS) Thrust vectoring angle	[rad]
Θ_x	(CS) Roll angle	[rad]
Θ_y	(CS) Pitch angle	[rad]
λ	Taper ratio	[-]
λ	(PP) Speed Ratio	[-]
Λ	Sweep angle	[rad]
μ	Viscosity	[Pa·s]
ξ	Non-dimensional radius	[-]
ξ	X-position of shear center	[m]
ρ	Density	[kg/m^3]
ρ	(PP) Resistivity	[Ωm]
σ	(SM) Normal stress	[Pa]
σ	(PP) Local solidity	[-]
τ	(SM) Shear stress	[Pa]
τ	(PP) Torque	[Nm]
τ_{ev}	(CS) Elevon effectiveness	[-]
ϕ	Flow angle	[rad]
ω	Angular Velocity	[rad/s]

1

Introduction

Drone racing is a rapidly growing hobby; with highly specialised and customised drones, competitions with prize pools of up to $1 million[1] are attended by pilots around the World. Most of these drones use a conventional quad-rotor frame. While it makes them highly maneuverable, they are limited to achieve very high speeds due to aerodynamics and thrust efficiency. Such a limitation creates a market gap for a radically unique design which excels at high speeds while maintaining maneuverability. Designing such a drone would not only be auspicious but could make large strides in the development and capabilities of drone technology. Given the surging growth of the drone market, it is a matter of time until such a ground breaking development occurs. The Peregrine team seeks to accomplish such a task by making the fastest racing drone on Earth which is still sufficiently maneuverable, through the incorporation of a wing-based drone design.

Thus far, a preliminary analysis was carried out on several unique concepts, leading up to a trade-off with specified selection criteria [4]. The design which emerged from the trade-off will be further explored in this report; therefore the aim of this report is to conduct a detailed design stage and finalise the drone system. This will be done by performing a detailed subsystem analysis, which once integrated will be used to make a final system. To validate the system, a simulation will be created using Simulink. Additional system assessments will also be performed, such as risk, sustainability and operation logistics. Finally, an outlook onto the future development tasks post DSE will be discussed.

The report structure reflects its aim. Prior to subsystem analysis, a review of the concept and revision of the budget and resources, system functions, market analysis is conducted in Chapter 2, Chapter 3 and Chapter 4 respectively. Subsystem analysis in the order of Aerodynamics, Structures & Materials, Power & Propulsion, Control & Stability and Performance & Simulation follows suit in Chapter 5, Chapter 6, Chapter 7, Chapter 8 and Chapter 9 respectively. This analysis includes subsystem specific functions, requirements and their compliance as demonstrated by detailed analysis, risks and verification & validation. It should be noted that due to the current stage of design, certain analyses may be incomplete, leaving opportunity for future work to be conducted. Risk and sustainability assessments are then carried out after subsystem analysis in Chapter 10 and Chapter 11 accordingly. With the primary aspects discussed and developed, the total system can be evaluated in Chapter 12. The system verification and validation is executed in Chapter 13. Finally, system development strategies are defined in Chapter 14 and the Conclusion is presented in Chapter 15.

---

[1] `https://dronelife.com/2016/03/07/dubais-1-million-world-drone-prix/` Accessed 5th June 2021

2

Review of Systems

This chapter serves as a revision on the progress of the Peregrine project, recapping the design and redefining the budgets and resources that will be used in detailed analysis.

2.1. Design Recap

This project started with the goal of designing the World's fastest competitive racing drone. The objective was to specifically design a racing drone that could achieve speeds of at least 90 [m/s] with a unit price of €2500 or less. To do so, the team of 10 aerospace engineering students segregated into organisational and subsystem specific roles (aerodynamics, structures & materials, power & propulsion, control & stability, and performance & simulation). The first stage of design was the creation of a project plan for the 10 week period. Afterwards a baseline analysis was performed which resulted in the creation of a design concept tree and the selection of three potential concepts. These three concepts were then analysed with more detail per subsystem and checked to see which was the most feasible for the design objective. This process is called a trade-off, and concluded with the selection of the concept which will be analysed in even further detail in this paper. The concept in question was called Concept A and is a tailless wing based design. It was chosen based on its simplicity and low mass in comparison to the other concepts, together with its ability to meet all the requirements.

Concept A will henceforth be referred to as the Peregrine drone given it is the chosen concept. To ensure that the team works on a consistent idea of the drone design, the Peregrine drone's specifications and traits will be outlined in this section. Though the details of these traits are to be optimised in this report, the traits themselves will remain unchanged.

Aerodynamically, the drone is fully comprised of airfoil bodies, with all bodies using a symmetric 4 digit NACA airfoil. The wing is swept and tapered, predominantly to increase stability and shift the wing's center of gravity aft. The choice of using a flying wing gives greater aerodynamic efficiency at high speeds in comparison to a more conventional quad rotor design. The structure of the drone will be comprised of a carbon fiber reinforced polymer which offers stiffness and strength whilst maintaining a low weight. The wings will also have a spar positioned at 30% of chord to reduce tip deflection and deal with the aerodynamic load. The structure is defined predominantly on the turning flight condition, as the drone has to withstand a 25 g turn with a speed of 60 [m/s].

To combat the difficulty with hover, there are 2 winglets on each wing tip, one on the upper and lower surface. This allows for the drone to be a tail sitter, meaning it can vertically takeoff. To maneuver from this vertical position after takeoff into horizontal, the drone will pitch downwards through using a pair of vectoring propellers on the outboard side of each half-wing. The vectoring of these propellers is powered through servos and will also help with hover control in case of winds and during low speed turns. To meet the speed requirements, another pair of propellers is used positioned at the inboard side of the half-wings towards the nose. This gives a total of 4 propellers on the drone. At higher speeds, the vectored propellers efficiency drops, requiring another method of control, specifically elevons. The drone thus incorporates elevons as control surfaces on each half wing so that sufficient turning rates may be achieved even at 90 [m/s].

2.2. Budget Revision

In the baseline report [3] budgets and contingencies were determined with respected to mass, cost, recyclability and power. In this chapter, the budgets will be revised as the design has been finalised. Additional components were added and the contingencies have been reduced as the design phase progresses.

There were multiple changes that were incorporated in the revised budgets as compared to the previous design phase. As it is planned that Peregrine will become a part of the drone racing market, the costs of the accessories that will be needed to operate Peregrine have been listed now so the user is aware of the total costs. This would be the cost of a ready to fly kit. These costs have been minimised compared to the baseline report [3] as the market demanded a more inexpensive solution. This minimisation shifted the choice for these components towards more inexpensive (slightly lower performance) alternatives. On top of this some extra components have been added to increase the detail of the budget. The components that have now been taken into account are the wiring and the tilting mechanisms for the propellers. A distinction has been made between actuators needed for the elevons and the actuators needed for the motors which have different masses and costs. The camera and structure have been allocated less mass. For the camera a lighter alternative was found making this feasible, while from the structural analysis in the baseline report [3] it was found that the structural mass could be reduced significantly compared to the previous budget.

The costs for the production of the tilting mechanism, propellers and the structural framework had to be estimated from the literature available on production of CFRP which is the material for the components mentioned. The estimation of the costs involved stems from the annual production rate of 1000 units. However, to keep the estimate conservative the costs included correspond to an annual production rate of 50. These costs include all the processes involved i.e moulding cost, prepreg layup, demoulding, milling, cost of autoclaving etc. Only the cost of assembly jig was excluded. The cost provided for annual production rate of 50 units is 890 [€/kg] which was scaled down to the weight of the components.[31]

The peak power used by the motors was determined from the most recent values as estimated during the previous design phase, this was equal to 1440 [W]. The power consumption of the flight controller includes the camera and antenna as they are powered through the flight controller. The power consumption of the RX and VTX were taken from the spec sheets of the respective parts. The exact recyclability percentages after a sustainability assessment is performed and the reasoning for them per material/component can be found in Chapter 11. The revised budget is shown in Table 2.1 where the parameters were based on representative components and a final contingency of 5% is applied as outlined in the Baseline review [3].

Table 2.1: Revised budgets

Component	Mass [g]	Total mass [%]	Recyclability [%]	Cost [€]	Peak power [W]
Off-the-shelf Components [1]					
Motors	99.8	21.0	65	100	1440
Speed Controller	6.0	1.3	60	40	–
Flight Controller	4.1	0.9	60	45	20
Antenna	3.0	0.6	55	20	–
RX	3.0	0.6	50	25	0.2
VTX	4.8	1.0	50	30	1.95
Camera	3.9	0.8	50	35	–
Battery	161.5	34.0	80	45	–
Actuators motor[2]	38.0	8.0	60	20	15
Actuators elevons [3]	21.4	4.5	60	10	5
24 AWG wire (Silicone)	8.5	1.8	75	2.85	–
14 AWG wire (Silicone)	2.6	0.6	75	4	–
Self Produced					
Tilting mechanism [4]	11.9	2.5	90	20	–
Propellers	15.7	3.3	90	15	–
Structure	90.3	19.0	90	85	–
Accessories					
Transmitter	–	–	–	167	–
FPV goggles	–	–	–	240	–
TX module	–	–	–	40	–
Charger	–	–	–	70	–
Contingency	25	–	–	125	74.1
Total	500.0	100.0	79.56	1138.85	1556.26

---

[1] `https://www.banggood.com/` Accessed on 29th April 2021

[2] `https://www.blue-bird-model.com/products_detail/490.htm` Accessed on 1st June 2021

[3] `https://hobbyking.com` Accessed on 1st June 2021

3
System Functions

With the basis of the final design defined following from the midterm, the system functions can be revised to match the design traits more accurately. The system requirements will also be used in the formulation of the system functions, thus they will be reviewed before the functional flow and breakdown diagrams are created.

3.1. System Requirements

The system requirements themselves are listed in the Chapter 13. The most pertinent, driving requirements will be listed here as they will be used across the subsystems for critical design sizing.

- SYS-PHYS-WGHT-01: The drone shall not exceed a total airborne mass of 500 [g].
- SYS-FLGT-MOVE-01: The drone shall have a maximum horizontal velocity in forward flight of at least 90 [m/s].
- SYS-FLGT-ROTA-01: The drone shall be able to perform a 25g turn in forward flight at a velocity of 60 [m/s].
- SYS-FEAS-COST-01: The drone shall not exceed a production cost of €2500.
- SYS-PLTG-INTR-02: The drone shall provide video feedback to the pilot via first-person-view goggles.
- SYS-PLTG-INTR-01: The drone shall enable the pilot to use a controller to input remote commands.

3.2. Functional Flow

Functional analysis of a system begins with defining top-level functions including their chronological order and then breaking them down into lower level functions maintaining the flow order. For the Peregrine drone, the top level (level 1) functions are defined as 0. Develop & Simulate Racing Drone, 1. Build & Test Prototype, 2. Produce Commercial Drone, 3. Market & Distribute Drone, 4. Operate Drone, 5. Maintain Drone and 6. Retire Drone. The following level 2 and 3 functions along with their chronological flow are presented in Figure 3.1, Figure 3.2 and Figure 3.3 where red boxes link the different level 1 functions between each other to reduce arrow clutter.

3.3. Functional Breakdown

With the first 3 levels of functions defined, it is possible to go in further depth for functions that could benefit from extra detail. The subsequent breakdown of level 3 functions creates level 4 functions. This will be performed on the level 1 functions of 'Operate Drone' and 'Maintain Drone' since they define how the subsystems should approach the problem. For instance, the structures department must be cognizant of the maintenance procedures when designing the drone such that the level 4 functions can be completed adequately. Since this process mainly focuses on the content of the task to be performed as opposed to the tasks that happen before or after it, the breakdown structure does not use arrows to indicate time flow, but rather uses them for connection of order like in an 'and tree' diagram. The finalised breakdown structure is presented in Figure 3.4 and Figure 3.5.

---

[4] `https://rcexplorer.se/product_archive/2020/05/tricopter-tilt-mechanism-product-archive/` Accessed on 1st June 2021

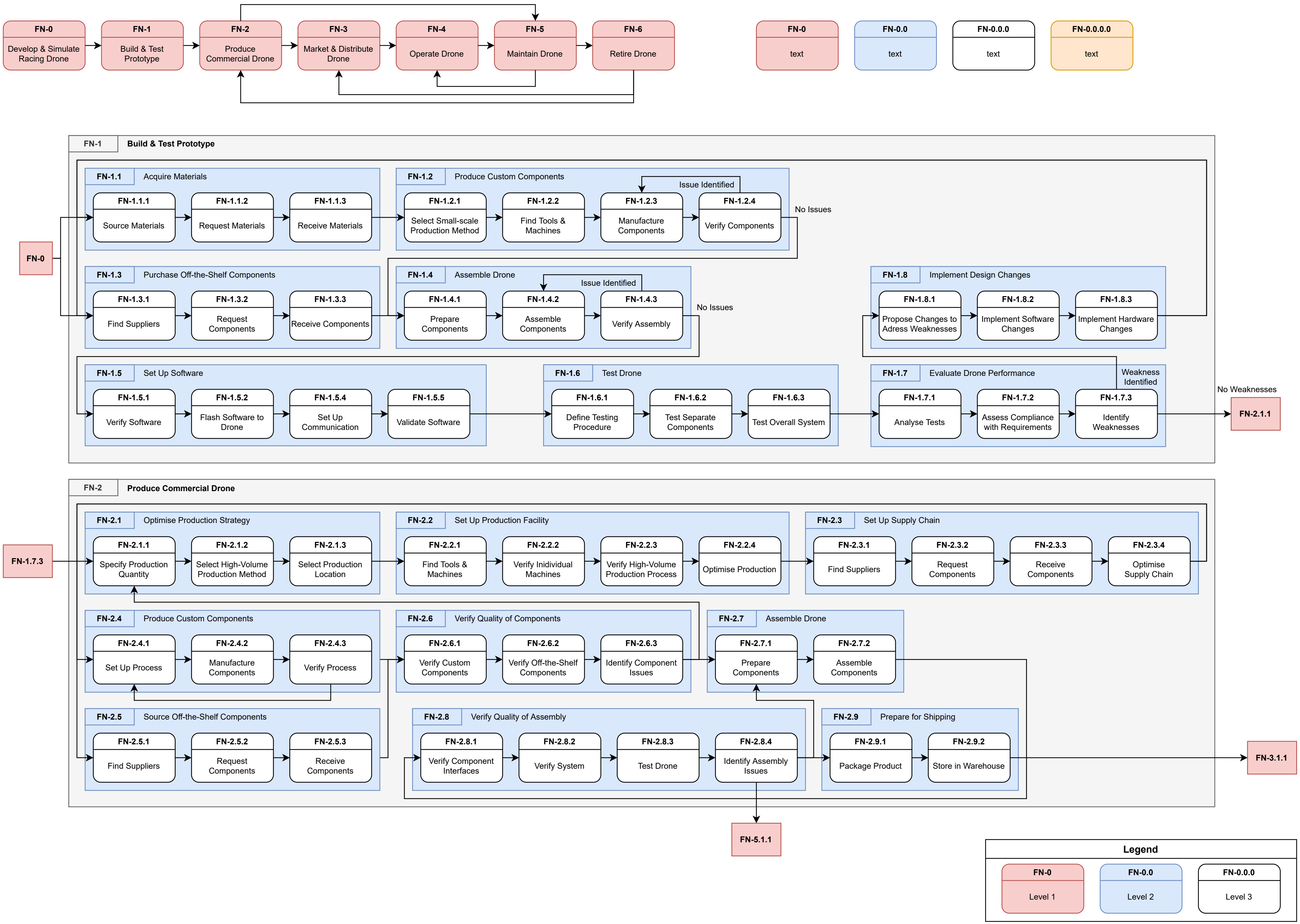


Figure 3.1: Revised functional flow

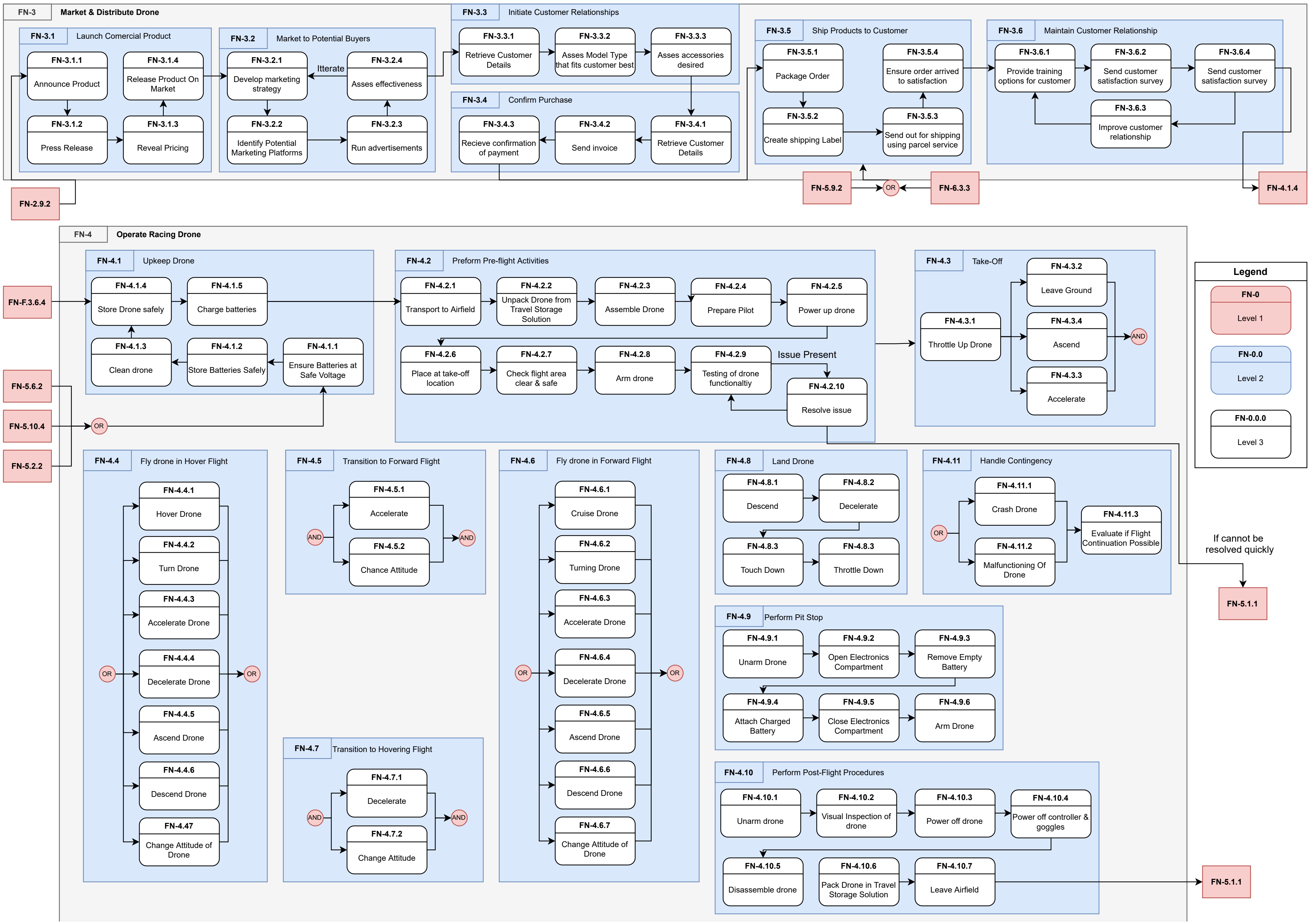


Figure 3.2: Revised functional flow (continued)

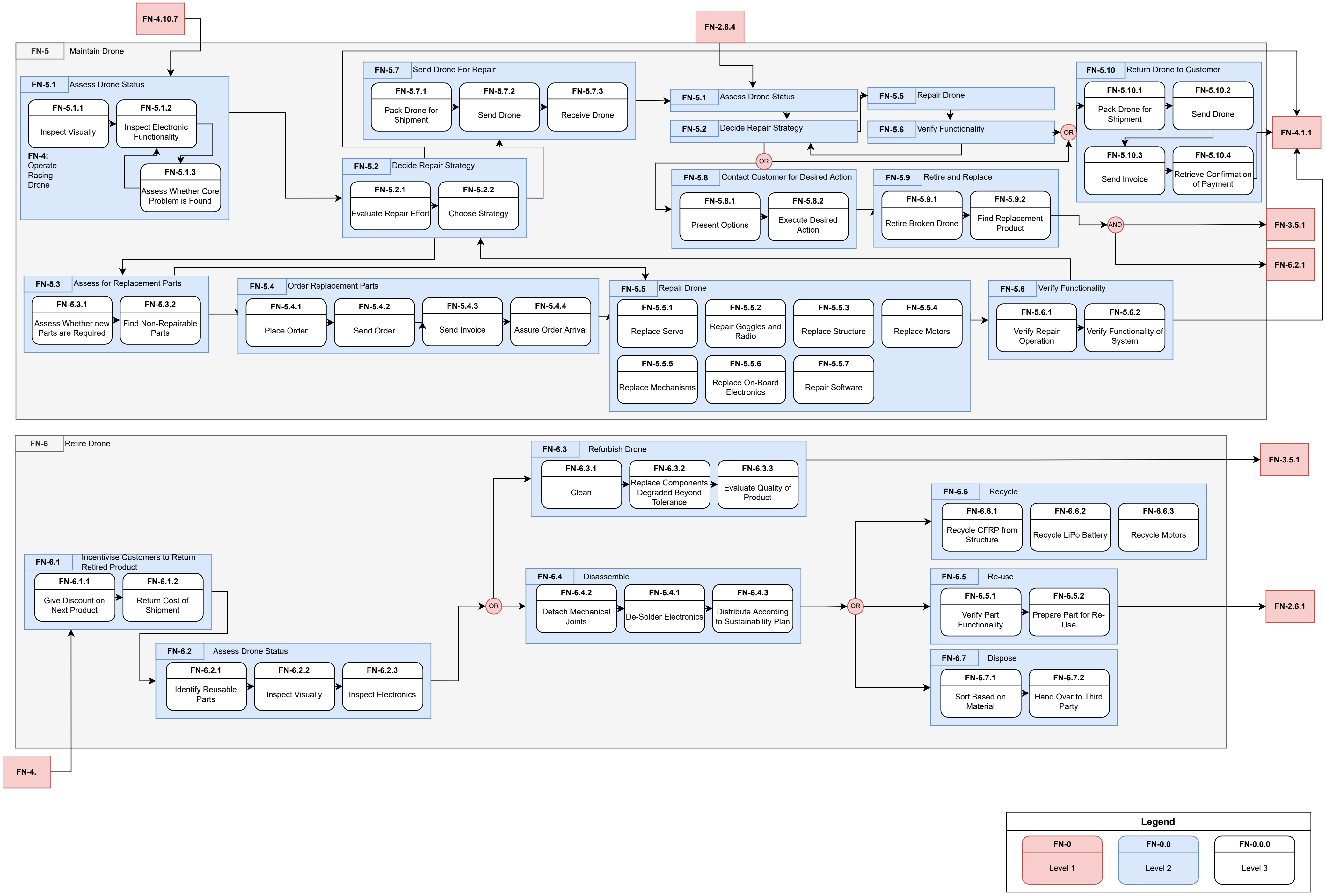


Figure 3.3: Revised functional flow (continued)

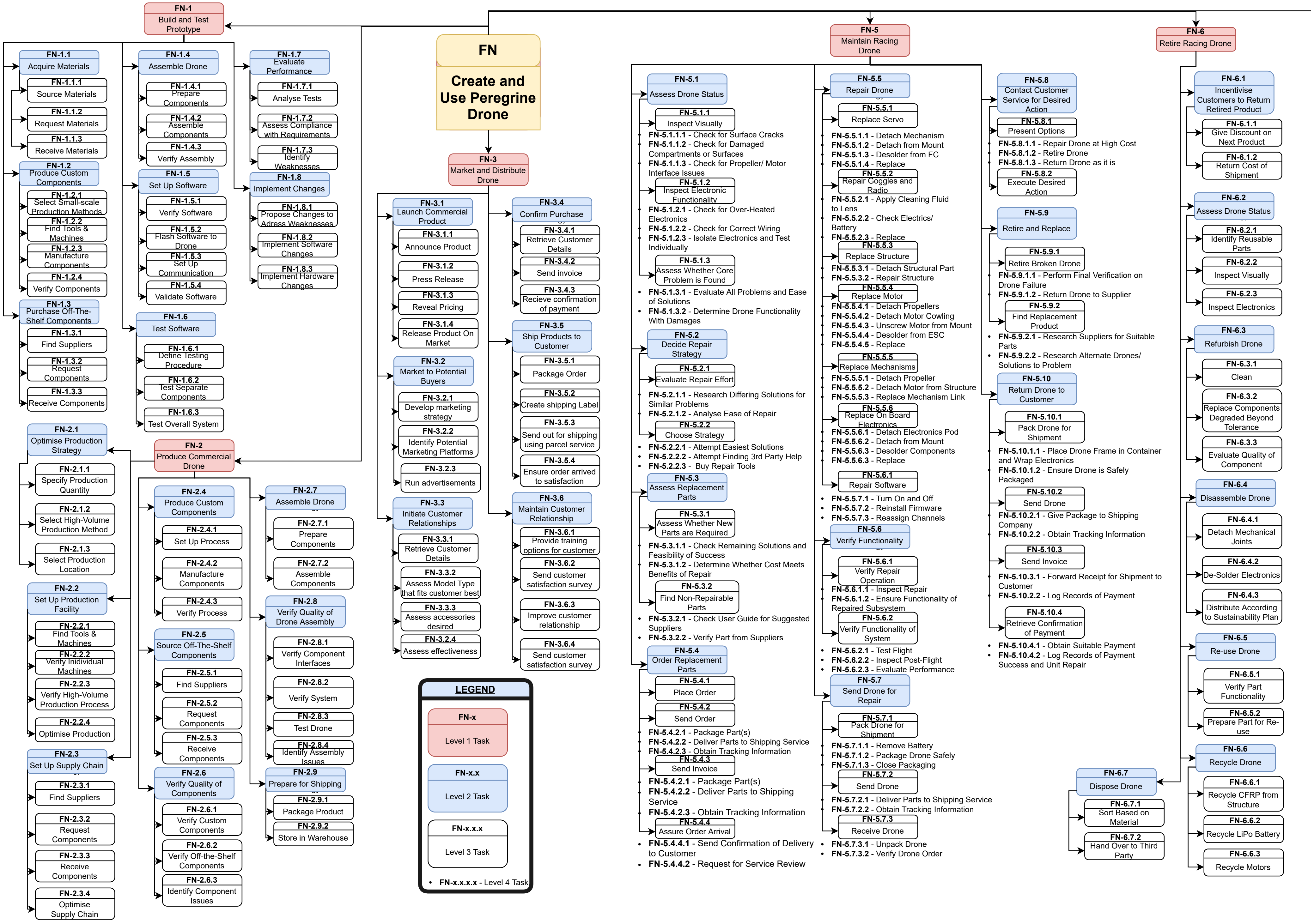


Figure 3.4: Revised functional breakdown structure

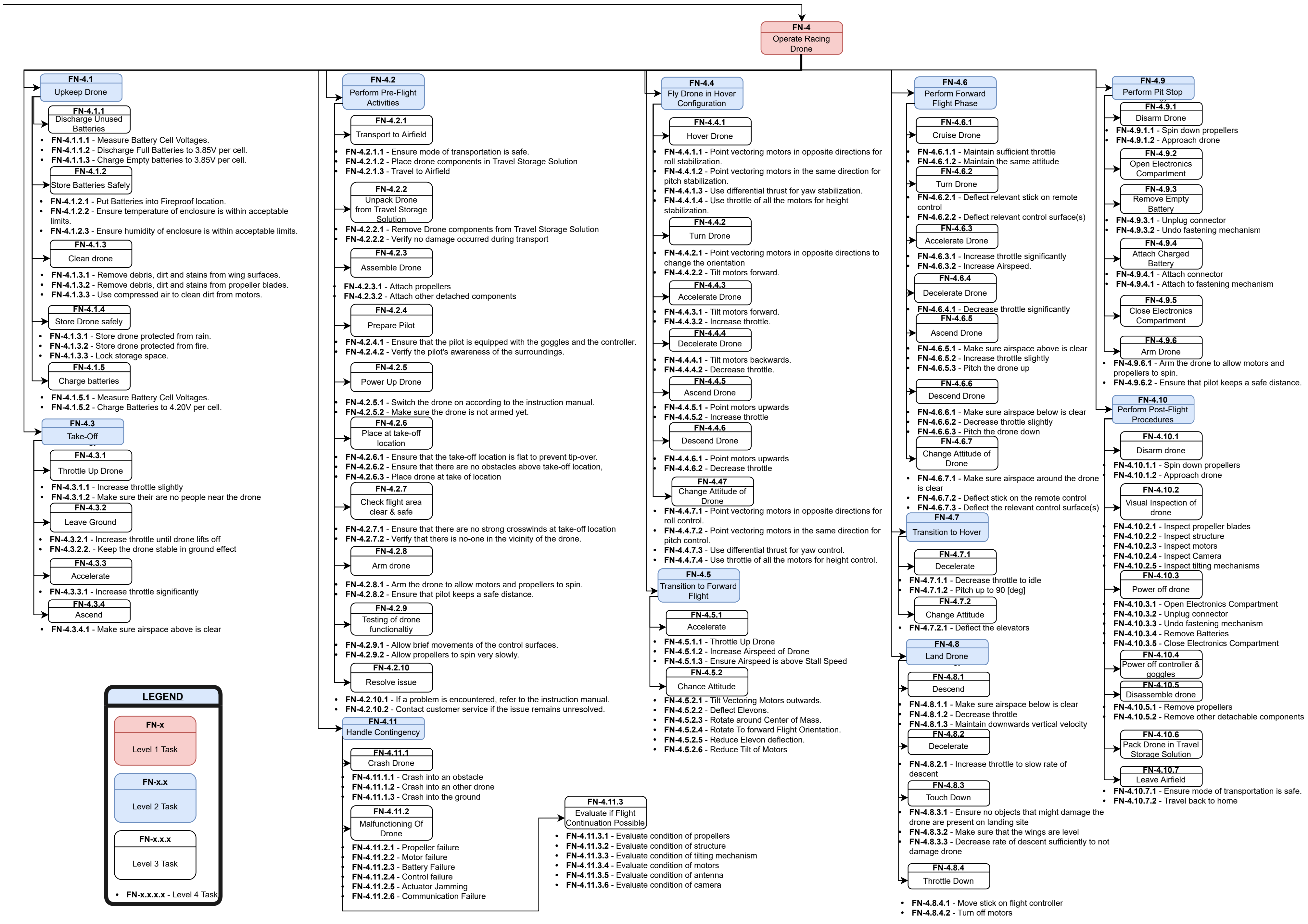


Figure 3.5: Revised functional breakdown structure (continued)

4

Market Analysis

To consider the impact Peregrine will have on the racing drone community, it is necessary to research and evaluate the current market, its growth opportunity, and the competitors that Peregrine will face. The aim of this section is to extrude necessary developments that must be made to the drone design to ensure its success upon market introduction.

4.1. Current Market

The current market will be analysed in terms of its market size and distribution which is then followed by a brief investigation into the types of competitions for drone racing to check the applicability of the Peregrine drone.

4.1.1. Market Size and Distribution

The drone market can be segregated into several sub-categories, one of which is the drone racing market. Whilst it is a relatively new market, drone racing has been under a sustained rise in growth throughout the past decade. As of 2019, the drone racing market alone was sized at 411.8 USD million with a next year growth rate of 22.1% [1]. The growth of the market is forecast to be exponential, such that by 2026, the market will be sized at 2060.7 USD million. Given an expected launch of the Peregrine drone in 2022, the market can be conservatively estimated to be at **711.6 USD million** using a linear growth rate. The distinction of the market into geographical regions reveals that North America is the largest market, carrying 30% of the market. This is then followed by the European and Asian markets. It is expected that this trend will continue in the coming years, indicating that these regions are the main target for Peregrine's market introduction.

Given the sub-category of racing drones in the drone market, there are further categories which are used to delineate the different types of racing drone uses. These categories are race related: drag racing- a race whereby the winner is the competitor who achieved the fastest speed throughout, rotorcross - a group of drones racing in a circuit or track with obstacles for a set number of laps, and the first to finish the required laps wins, and time trials- racing in a straight or circuit for speed records. The market is predominantly dominated by rotorcross. Time trials form the second greatest contribution with drag racing being the least contributing. The Peregrine drone is expected to excel in rotorcross racing specifically as it may require time to accelerate and it follows the conventional racing method. It may also perform well in the time trial race given its high speed requirements. Drag racing is not the most suitable given it is often a tighter and less official method of racing. For the purpose of this analysis, the contributions of rotorcross and time trials will be assumed to be 75% of the total drone racing market, meaning that Peregrine will aim to attack this portion of the market. Applying this ratio leaves Peregrine with an available market size of **533.7 USD million** in its first year. [2]

Another distinction to be made in the market size of drone racing is the type of drones being produced and sold. The two main types of drone products are ready to fly drones (RTF) and almost ready-to-fly drones (ARF). RTF drones are the complete package, an assembled drone including all required components and a controller. Modifications to these drones requires disassembly and replacement. ARF drones are often sold as a set of components and a manual to which the customer assembles the final drone. The main benefit of ARF drones is the ease of replacement for specific components, allowing for customisation and modulation to the customer's needs. Since drone racing is composed mainly of enthusiasts at the moment, ARF drones are the largest contributor to the commercial drone market due to their ease of customisation for skilled pilots. This is also due to the ARF drones surprisingly being more expensive to RTF drones due customisation options at the customers request outweighing the assmebly costs. Although it is forecasted

---

[1] `https://www.researchandmarkets.com/reports/5261971/racing-drone-market-share-size-trends-industry` Accessed on 5th June 2021

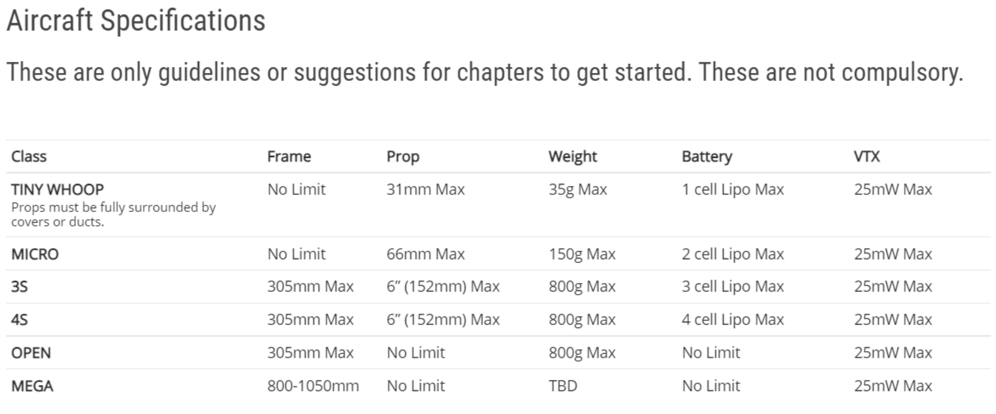
[2] `https://www.electronicspecifier.com/industries/robotics/racing-drone-market-size-growing-to-2-060-7m` Accessed on 5th June 2021

that the ARF drones will remain the key contributor to the market size, Peregrine will be made initially as a RTF drone since, given the large market growth to come, the team expects more beginners to contribute to the market. Additionally, once Peregrine's performance is showcased and publicised, a large influx of customers who will want to purchase the drone is expected, to which having it as an RTF allows Peregrine to maximise branding. Once the RTF drone success peaks, the team expects to expand into the ARF market. [3]

4.1.2. Drone Racing Regulations

Since Peregrine is designed to be a formidable racing drone, one must first analyse the racing formats or regulations that are currently instated. If the Peregrine drone can directly fit into a racing category, or small amendments be made to fit into a category without foregoing requirement compliance, then it will be beneficial to its immediate market growth. That being said, if the Peregrine drone does not meet a specific race or competition's regulations and any changes to be eligible put the requirements at risk, it will not be design critical. As drone racing is itself a novel hobby, it has room for growth. It is thus hoped that exceptions and changes can be made to racing entry requirements and regulations to fit the market's growth and allow for drone designs other than quad-rotors, such as flying wings.

Drone competitions are often distinguished in terms of size, with main classes being micro, open and mega classes. The micro class is for 1-2 cell racing drones and has a maximum weight of 150 [g], which is unacceptable for Peregrine. The open class is available to all multi-rotor designs, allowing for broad competition. As well, the 800 [g] weight cap bodes well for Peregrine's eligibility. However open class is limited to a frame of 305 [mm], which is not possible since Peregrine uses a frame of 700 [mm]. This is an example of a restriction which is well defined for a quad-rotor drone but not for an unconventional frame, such as Peregrine. The remaining weight class is the mega class, which uses frames of 800 to 1050 [mm] and does not have stringent weight limits. This class is somewhat applicable to Peregrine, but due to the larger drones present, Peregrine would not be optimised for the race and would require an unacceptable increase in span size. These class distributions are based off a yearly competition, MultiGP, which garners most enthusiasts around the world. Further specifications are displayed in Figure 4.1[4].



Aircraft Specifications

These are only guidelines or suggestions for chapters to get started. These are not compulsory.

Class	Frame	Prop	Weight	Battery	VTX
TINY WHOOP Props must be fully surrounded by covers or ducts.	No Limit	31mm Max	35g Max	1 cell Lipo Max	25mW Max
MICRO	No Limit	66mm Max	150g Max	2 cell Lipo Max	25mW Max
3S	305mm Max	6" (152mm) Max	800g Max	3 cell Lipo Max	25mW Max
4S	305mm Max	6" (152mm) Max	800g Max	4 cell Lipo Max	25mW Max
OPEN	305mm Max	No Limit	800g Max	No Limit	25mW Max
MEGA	800-1050mm	No Limit	TBD	No Limit	25mW Max

Figure 4.1: MultiGP competition drone classes[5]

As noted, the values shown are simply guidelines, evidencing the ability to potentially accommodate with a changing market or drone. For this reason, the Peregrine racing drone will attempt to fit with the Open class due to the most applicability, though no additional changes to the design will be made with the intention of fitting into the requirements stringently.

4.2. Competitors

As most drones used in racing at the moment are quad-rotor racing drones, there is no definitive competition to having a flying wing racing drone. That being said, analysis on the current quad-rotor racing drones and

---

[3] `https://www.electronicspecifier.com/industries/robotics/racing-drone-market-size-growing-to-2-060-7m` Accessed on 5th June 2021

[4] `https://www.multigp.com/class-specifications/` Accessed 9th June 2021

[5] `https://www.multigp.com/class-specifications/` Accessed 9th June 2021

existing recreational flying wing drones can give insight to the opportunities and flaws of the Peregrine drone.

4.2.1. Racing Drones

A list of current quad rotor racing drones' cost, speed, range, endurance and mass is tabulated in Table 4.1.

Table 4.1: Comparison of racing drones currently on the market.

Drone	Cost	Top speed	Range	Endurance	Mass
DJI FPV Combo	€1349	39 [m/s]	16.8 [km]	20 mins at 18 [m/s]	795 [g]
Walkera F210 Professional Racer	$609	22.3 [m/s]	-	9 mins	379 [g]
Custom made	$300	45 [m/s]	1-5 [km]	3-4 mins	400-500 [g]
Iflight Nazgul	$200	36 [m/s]	1-5 [km]	3-4 mins	470 [g]

For the DJI FPV Combo, the price is significantly larger because the drop is sold as RTF, whereas the other drones list costs that are not the complete drone, for example the Iflight Nazgul cost is the frame without the additional components such as the headset and controllers or battery. Key takeaways are that the speeds of the current drones are inferior that expected of Peregrine, with the greatest speed being half of the top speed of Peregrine's. In terms of mass, Peregrine is expected to fit in the middle of the market, given it is not allowed to be greater than 500 [g]. Endurance wise, the requirement of flight time for 4 minutes puts Peregrine around the same range of a custom made, but less than the DJI or Walkera. This can be seen as a disadvantage, but given the top of the line expected racing performance it is difficult to get larger endurance values with 500 [g]. The key takeaway from racing competitor analysis is the cost of the RTF drone. Peregrine will thus try to reduce its costs to assimilate as close as possible to the DJI, as opposed to meeting the requirement of the €2500 budget.

4.2.2. Flying Wing Drones

Flying wing drones are within the drone market. Drones such as the TBRC Micro Wing, MiniRace Wing and Flybot Flux are examples of such drones. The TBRC Micro Wing is made of a foam based frame, with wingspan of 600 [mm]. The fastest TBRC drone frame is the Villain, which sells as a frame at $120 and can fly at 63 [m/s] in optimal configuration[6]. The MiniRace Wing uses a foam wing, and main pod made of carbon fiber reinforced polymer (CFRP), offering protection to the inside electronics. The span is 95 [cm] with a weight of 450 to 900 [g] depending on assembly and the components used. Interestingly, this drone incorporates a detaching wing structure, such that under impacts, the wings detach from the drone reducing the effects on the main pod. With such detachment, the wing can also be replaced or assembled easier. No speed information was available for this drone unfortunately[7]. The Flybot Flux, like the MiniRace, uses a CFRP fuselage and foam for the wings. It has a wing span of 90 [cm] and take-off weight of 500 to 800 [g][8].

The conclusions to take from this analysis is that the majority of flying drone wings use foam to construct the wings. The main reasoning for this is that the foam allows for a lightweight structure, whilst providing good durability. Additionally, the use of CFRP is common around the electronic structures to ensure protection. One point of highlight is the fact that the drones typically weighed greater than 500 [g], which could indicate the difficulty in designing a better performing, but structurally sound drone that is lighter. Lastly, it may be interesting for future analysis to check the ability of removing the wing structure from the main body, for assembly and impact contingency reasons.

4.2.3. SWOT Analysis

A SWOT analysis represents the expected product performance in the market with respect to its competitors, the market needs, and growth opportunities. This is displayed in Figure 4.2.

---

[6] `https://www.tbrcwings.com/villain` Accessed 9th June 2021

[7] `http://miniracewing.com/` Accessed 9th June 2021

[8] `https://flybot.de/media/pdf/15/08/a7/FLUX_Manual_EN.pdf` Accessed 9th June 2021

Internal

Strengths

- Highest top speed of any drone on the market
- Excellent manoeuvrability
- Identical controller to other drones despite radically different design
- Majority recyclable by weight
- Has a distinctive look
- Has a readily available simulation to verify performance

Weaknesses

- High cost
- Low endurance at top speeds
- Low controller range
- Limited slow speed turning performance
- Poor landing performance
- Hard to control for new pilots
- Needs a transition phase from hover to forward flight

External

Opportunities

- Applications for disaster relief, law enforcement and cinematography
- Further optimisation to lower cost
- Market with a lot of room to grow and gain popularity
- Possibility to improve performance further
- Possibility to add autonomy

Threats

- Potentially trying to capture too niche of a market
- Competitors improving on the design
- Disqualification from drone races (i.e. for not qualifying as a drone)
- Loss of interest in drone racing
- No autonomy

Figure 4.2: Peregrine SWOT analysis.

The primary strength of Peregrine is its high speed and excellent manoeuvrability. Given the competition mentioned in Section 4.2, it is clear that there is a market gap for a high speed, highly maneuverable and low weight drone. Matching with Peregrine's strengths, it is evident that the project is set to capitalise off of this market gap effectively. Therefore it is imperative that the requirements on top speed, turning flight and mass are met within the design. Peregrine's biggest possible weakness in terms of the market is its relatively high cost when compared to its competitors. This could be fairly justified given its maximised performance and unique recyclability, but attempts will be made to reduce the cost of the drone to match its competitors. Given the exceptional performance characteristics of Peregrine, its applicability to other markets is diverse. The specifics of each market and the suitability will be further detailed in Section 4.3. Finally the biggest threat it could face is the absence of a large market audience. Given that Peregrine will launch as a BNF drone, it targets newcomers to the market specifically, meaning it is sanctioning itself from a decently sized veteran market. To ameliorate this, Peregrine aims to also launch as an ARF drone quite soon after its initial market introduction, thereby capturing a more sustainable market share. Another way to deal with threats of upcoming competitors and developing technologies is through the use of frequent updates to the design. This is further employed and specified in Section 14.3, whereby post-launch design iterations are included in the project schedule.

4.3. Peregrine's Place in the Market

Having analysed the current market and Peregrine's competitors, it is now possible to determine Peregrine's place in the market. This will be done by comparing all of Peregrine's top level requirements to existing drones.

- SYS-FLGT-MOVE-0: The drone shall have a maximum horizontal velocity in forward flight of at least 90 [m/s].
  - Achieving a top speed of 90 [m/s] puts Peregrine comfortably ahead of all of its competitors, as can be seen in Table 4.1. Drones that reach such high speeds do exist, but they are optimised for straight flight, which severely compromises their turning performance.
- SYS-FLGT-ROTA-01: The drone shall be able to perform a 25g turn in forward flight at a velocity of 60 [m/s].
  - This also puts Peregrine comfortably ahead of its competitors as high performance race drones typically have a thrust to weight ratio between 7 and 11, with thrust to weight ratios above 15 almost unheard of [9]. Since all of the centripetal acceleration comes from the thrust for conventional race drones, that means that no drone can perform a turn above 15g. At a given speed a higher g turn corresponds to a smaller turn radius, so this makes Peregrine's manoeuvrability at higher speeds

---

[9] `https://www.rcgroups.com/forums/showthread.php?2714261-Acceptable-thrust-weight-ratios` Accessed on 21st June 2021

significantly better than its competitors.

- SYS-FLGT-HOVE-05: The drone shall be stable while hovering.
  - In this respect Peregrine performs slightly inferior to its competition. While Peregrine is fully capable of hovering, it will not be as easy as with conventional drones since the quadrotor configuration is much more efficient for hovering. Peregrine will also be more prone to crosswind problems while hovering than conventional drones. However, it should be noted that hover is likely the shortest phase in Peregrine's flight profile, so this is not a significant disadvantage.
- SYS-FEAS-SUST-01: The drone shall be 80% recyclable by weight.
  - Currently there is not much focus on sustainability in racing drones. For the most part, they can be recycled, though there is no emphasis placed on it nor much follow up to ensure the drone does get recycled at its end of life. Thus, this is a major differentiating factor for Peregrine.
- SYS-PHYS-WGHT-01: The drone shall not exceed a total airborne mass of 500 [g].
  - This puts Peregrine in a similar range to competitors as it falls within the regulations for both recreational use and race use. Thus, it does not give a distinct market advantage nor disadvantage.
- SYS-FLGT-ENDR-02: The drone shall be able to fly in flight profile for at least 4 [min].
  - In this regard Peregrine is again in a similar range to its competitors, due to the fact that while racing there is no real need for more endurance given the length of races.
- SYS-FEAS-COST-01: The drone shall not exceed a production cost of €2500.
  - In terms of cost, Peregrine is significantly more expensive than the majority of its competitors, which may become an issue when targeting beginners. However, this does open the opportunity for Peregrine to be sold as an ARF as well as a RTF drone. In this way, customers that are willing to pay the full price for a RTF version are able to do so, but customers that are not willing to pay as much can also get certain parts for a cheaper price.

Overall, it can be seen that in the majority of areas, Peregrine is a significant improvement to almost all of its competitors. However, it is slightly lacking in hovering performance and more significantly with its cost. Nonetheless, it is expected that Peregrines superior straight line performance, turning performance and recyclability will be sufficient to gain a significant market share soon after launching.

4.4. Market Share

Following from Section 4.1, Peregrine's available market size by 2022 is \$ 533.7 million. In 2019, 45 % of the market revenue was attributed to the top 5 contributors, indicating a relative market dominance. With increasing market size, the balance may change, but it is known that there are many small custom drone build companies [10]. For this reason, in the first year the team assumes that Peregrine does not make take a large share given its radical design and price. Only until Peregrine is actively publicised as superior via winning competitions which showcase the drone's performance through media, does the team expect larger uptake. Thus for the first year, Peregrine is expected to take a market of 2.5%, resulting in **16 USD million** based on the market size. Given a drone price of maximum budget, this would require a minimum of 5340 units to be sold. This price is quite extreme however compared to competitors, and it is assumed that the analysis will result in further cost reductions. Therefore the drone price to be sold will be assumed as half of this value, resulting in a required 10680 drones being sold in the first year. The exact profit margin per drone will thus be dependent on the production costs detailed in Section 14.4. Upon winning more competitions, the interest will grow exponentially, meaning a large portion of this share is expected to occur towards the end of the year after some time and competitions have passed.

By the second year, it is natural to expect to gain a larger percent of market as Peregrine continues to showcase superior performance in the main drone races. The team therefore expects a new market share of 7.5%, resulting in **64 USD million** revenue by 2023 (once again a conservative estimate on overall market growth is applied with a flat growth rate of 22.1%). Using the maximum drone price, this would require a minimum 30464 drones being produced and sold and using the assumed retail price of €1250, a total of 42666 or approximately 45000 drones being produced per year.

4.5. Other Markets and Opportunities

Peregrine has been primarily designed for the racing drone market, all the relevant analysis for the racing drone market has been presented in the sections above. However, apart from being a racing drone, its unique characteristics are also appealing to a wide variety of markets other than the racing drone market. However,

---

[10] `https://www.researchandmarkets.com/reports/5261971/racing-drone-market-share-size-trends-industry` Accessed on 5th June 2021

given its primary purpose, it lacks certain features that may be beneficial for entry into these markets. These potential markets and Peregrine's feasibility of entering these markets will now be discussed in detail.

4.5.1. Aerial Cinematography

Aerial cinematography is currently used in movies to film action sequences or used in sports such as Formula 1 to provide unique camera angles to viewers. Peregrine being a high speed maneuverable drone will be able to capture the required camera angles as it can keep up with the fast paced car chase sequences in movies or Formula one cars. To be able to provide this service Peregrine would require a high definition camera as its payload which would incur significant iteration and structural mass for the additional payload size and mass. Another potential problem is Peregrine's short endurance at high speeds, as Peregrine's maximum endurance is 4 minutes on average speeds, it will not be able to film for more than the specified minutes without recharging its batteries. Finally, the drone will also require a large controller range that will make it controllable even when the pilot is not near the controller as might be the case when filming action sequences. In conclusion, aerial cinematography is a feasible market for Peregrine to enter if the controller range can be increased and its payload capabilities increased.

4.5.2. Search and Rescue

Drones can also be used for search and rescue operations. Through drones it is possible to access remote locations and if the drone is equipped with sensors such as infrared cameras, then it will be able to locate people accurately even through obstructions. Peregrine is well suited for such operation as it can cover a specific area at a high speed and if equipped with an infrared sensor, it will be able to locate people for search and rescue. The infrared camera is expected to however weigh significantly more than the current camera on the drone and hence will cost performance to add onto the system. Some drones are also required to carry supplies as payload to drop medicine, food or oxygen for stranded individuals. For Peregrine to do this there may also be performance cost as carrying supplies will also significantly increase the payload of the drone. Finally it is also required to increase the controller range of the drone such that it is controllable from long distance. Hence, it might be feasible for Peregrine if the controller and payload range can be increased.

4.5.3. Disaster Management

UAVs and drones are used for disaster management for mapping of disaster affected areas or to access locations in these disaster affected areas inaccessible for humans. Peregrine's high speed and excellent maneuverability would allow the drone to cover the disaster area quickly and also access remote locations. To be able to map areas effectively, Peregrine would require a specific sensor that may increase its payload and hence its mass. However, this sensor is expected to weigh much less than for example an infrared camera and could be possibly placed on Peregrine without any performance cost. The drone would also require that its current required controller range is increased, as it would be beneficial to be able to control the drone over large disaster affected areas and ensure that the drone can finish its operation without any communication failure. In conclusion, with minor alterations Peregrine can be used as a drone for disaster management and hence, it is a plausible market for the drone.

4.5.4. Law Enforcement

Drones and UAVs can be typically used to follow speeding cars which do not stop on the indication of the police. Drones such as Peregrine due to their high endurance on speeds such as 35 [m/s], which is typical for a speeding car, will be able to follow these cars and feed its location to the law enforcement authorities at all times. For this specific application, Peregrine will also not require any specific sensors or increased payload as the camera currently on the drone should be good enough to track the position of speeding cars. However, the only problem as in other applications is the need for a long range controller such that the pilot can always control the drone during tracking speeding cars. Hence, if the controller range is increased, law enforcement is a very feasible market for Peregrine to enter.

4.5.5. Conclusion

As explained in the analysis above, Peregrine can feasibly enter all the above described markets as long as the controller range for the drone can be increased. For some of these markets as the payload of the drone would need to be increased, the performance of the drone will decrease. These aspects can be opportunities for future Peregrine designs.

5

Aerodynamics

This chapter covers the aerodynamic analysis performed for Peregrine racing drone. First, a functional recap is presented, followed by the subsystem requirements. Then follows the analysis section, before a technical risk assessment, verification and validation of the tools and models utilised, and at last the compliance matrix.

5.1. Functional Recap

The aerodynamic frame is responsible for generating the loads needed in order to allow desired performance, up to 25g of lift and a limited drag. In order to ensure this capability, detailed analysis and sizing of components affecting the airflow must be analysed and optimised.

The Aerodynamics department further provides drag estimates for propulsion and power, load distributions and structural boundaries for the structures department, and stability derivatives for the Control department.

5.2. Requirements

The subsystem requirements related to the aerodynamic performance are shown in Table 5.1, along with the method of verifying compliance.

Table 5.1: Aerodynamic subsystem requirements

Identifier	Requirement	Method
AE-PHYS-SIZE-01.1	The propellers shall have a maximum diameter of 17.78 [cm].	Review of Design
AE-PHYS-SIZE-01.2	The airframe shall have a span less than 70 [cm].	Review of Design
AE-PHYS-SIZE-01.3	The airframe shall have a thickness of minimum 5[mm].	Review of Design
AE-PHYS-SPMT-03.1	The airframe shall not induce periodic oscillations that cause the load carrying structure to resonate.	Analysis
AE-FLGT-HOVE-04.1	The airframe shall have a drag less than 4.25[N] in hover condition at maximum velocity.	Analysis
AE-FLGT-MOVE-01.1	The drag in horizontal flight condition shall be less than 13 [N] at 90 [m/s].	Analysis
AE-FLGT-MOVE-02.1	The propellers shall not stall within the velocity range.	Analysis
AE-FLGT-MOVE-02.2	The lift in forward flight at minimum speed shall be at least 5 [N].	Analysis
AE-FLGT-MOVE-05.1	The aerodynamic frame shall ensure static stability of the drone in forward flight.	Analysis
AE-FLGT-ROTA-01.1	The aerodynamic frame shall be able to deliver a lift of minimum 123 [N] at 60[m/s].	Analysis
AE-FLGT-ROTA-02.1	The aerodynamic frame shall facilitate a transition from horizontal flight to a 25g turn at 60 [m/s] within 1 [s].	Analysis
AE-ENVI-WTHR-02.1	The aerodynamic frame shall allow operation during maximum crosswinds in any flight condition.	Analysis
AE-ENVI-ALTD-01.1	The aerodynamic frame shall allow operation up to atmospheric condtions at the defined maximum altitude.	Analysis
AE-ENVI-NOIS-01.1	The propellers shall not produce a noise level greater than 90 [dB] at maximum speed.	Similarity

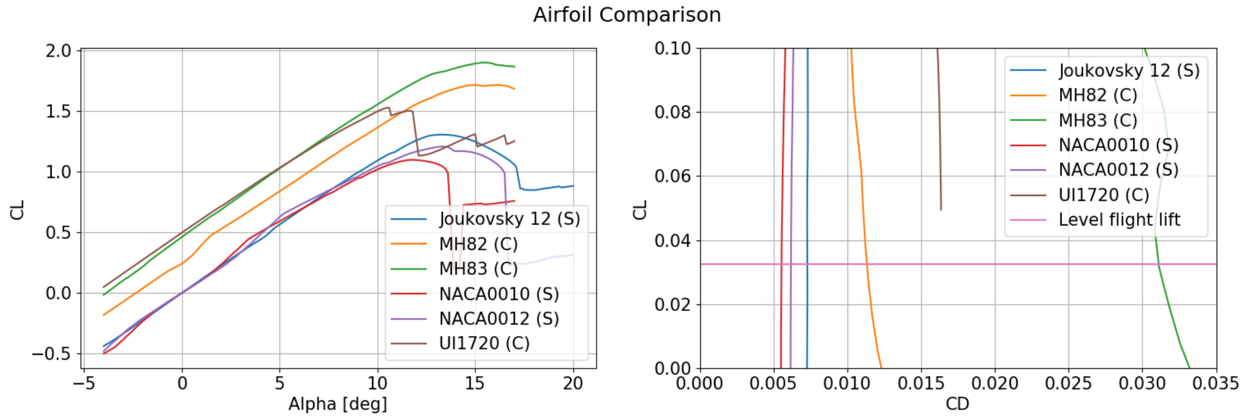


Figure 5.1: The six most promising airfoils compared at Re = 4.5 $\cdot 10^5$, where (S) indicates a symmetrical airfoil and (C) indicates a cambered airfoil.

5.3. Analysis

This section presents the aerodynamic analysis done for Peregrine, starting with the airfoil and planform Optimisation, before the electronic pods are modelled. Then the propeller and elevon effects are presented, followed by the winglet design and preliminary flutter analysis. At last the requirements related to the flight environment are analysed.

5.3.1. Airfoil Optimisation

The aerodynamic performance starts with the most basic shape the wing can have: the airfoil as selecting a suitable airfoil is crucial for obtaining the desired level of performance. Before the airfoil selection can start the design conditions need to be specified. In the case of Peregrine the airfoil has been selected such that the final design would have the lowest zero lift drag as the top speed of 90 m/s is one of the driving requirements. However the airfoil should also produce lots of lift as the turning requirement is quite stringent. During the midterm [4] the NACA 0015 airfoil has been used, however during the sensitivity study it was evident that the zero lift drag decreased with the thickness of the airfoil. Because of this lower thickness airfoils have been investigated.

Cambered airfoil were not used previously, since they would make the hover phase more complicated with the wing producing lift in one direction while hovering. These airfoils might prove to have significant advantages while producing lift however, thus they were taken into account for this analysis.

To perform the analysis the airfoils were imported into XFLR5 using either their available coordinates from the UIUC database[1] or generated by the NACA generator in XFLR5 using 100 points. The analysis has been performed at a Reynolds number of 4.5 $\cdot 10^5$ as the Reynolds number at the mean aerodynamic chord at 60 m/s is roughly 4.5 $\cdot 10^5$. Finally the NCrit was kept at a default value of 9.0. Multiple airfoils have been analysed, both cambered and symmetrical, the results of the six most promising ones are shown in Figure 5.1.

From Figure 5.1 it is visible that the cambered airfoils are able to provide a significantly higher lift, however their drag at the cruising lift (assuming sea level atmosphere and a surface of 0.069 [m^2]) is also significantly higher. This means that there are two strategies in the design. Either choose a highly cambered airfoil that will have a very high lift coefficient and thus need a smaller wing to reach the same amount of lift. Or choose a symmetrical airfoil and design a wing that has a larger surface area. The optimal option is determined by the drag force at the cruise condition. To determine this, first the surface area needed to achieve sufficient lift was calculated using the maximum lift coefficient. Then the drag at 90 [m/s] was calculated using this surface area and the zero lift drag coefficient (as the lift coefficient needed to fly 90 [m/s] is very close to zero). The results of this are displayed in Table 5.2.

From the values in Table 5.2 it can be seen that although the cambered wings would require a significantly smaller planform, their drag at 90 [m/s] is also significantly higher. Because of this symmetrical airfoils are still preferred over the cambered airfoils. When looking at the symmetrical airfoils the NACA 0010 has the lowest drag. Do note however that the difference in drag between the NACA 0010 and the NACA 0012 is relatively small while their surface areas differ significantly. Since the smaller surface area would allow for a

---

[1] `https://m-selig.ae.illinois.edu/ads/coord_database.html` Accessed 10th June 2021

Table 5.2: Drag of the airfoil-wing combination at 90 [m/s]

Airfoil	Drag [N]	Surface area [m^2]
NACA 0012	1.4051	0.06908
Joukovsky	1.5363	0.06389
NACA 0010	1.3823	0.07599
MH82	1.9804	0.048640
UI-1720	3.0077	0.05462
MH83	4.8212	0.04389

Table 5.3: Parameters of the optimised planform.

Parameter	Value	Unit
Span	0.69	[m]
c_r	0.137	[m]
c_t	0.085	[m]
Λ_{LE}	20	[deg]

higher aspect ratio and a lighter structure, the NACA 0012 is chosen.

5.3.2. Planform Optimisation

With an airfoil chosen, the overall wing geometry could be analysed and optimised. The main concern of this optimisation was the ability to generate 184[N] of lift at 60 [m/s], 25g and 1.5 safety factor, which is required during turning. The planforms were further ranked based on their drag profile at a lift equal to the weight across the velocity range, and the drag produced during the 25g turn.

In order to achieve static stability in a flying wing, the mean aerodynamic center (MAC) needs to be behind the center of gravity. In order to allow a larger center of gravity range, the wings are often swept, moving the MAC further aft. For Peregrine, the Mach number is less than 0.3 at 90 [m/s], and thus postponing drag divergence is not a motivation for the presence of wing sweep. Wing sweep further tends to have a negative effect on the lift curve slope, as displayed in Figure 5.10, and also affects the obtainable lift negatively. It was therefore decided to include a limited amount of sweep when optimising the planform.

The lift induced drag is determined by Equation 5.1, with C_L the lift coefficient, A the aspect ratio and e the Oswald efficiency factor. As large amounts of lift are required during turns, emphasis was placed on reducing the induced drag as much as possible by maximising the aspect ratio and Oswald efficiency factor. The aspect ratio is effectively optimised by maximising the span and reducing the mean geometric chord length. The span limit is determined by requirement AE-PHYS-SIZE-01.2.

$$C_{D_{induced}} = \frac{C_L^2}{\pi A e} \qquad (5.1)$$

$$\lambda = 0.2 \cdot (2 - \Lambda_{0.25c_{rad}}) \qquad (5.2)$$

The Oswald factor is determined by the lift distribution, and how well it resembles an elliptical distribution. This depends on the taper ratio and sweep. The optimal taper ratio for a given quarter chord sweep is given by Equation 5.2, yielding a value below 0.4. Iterating the planform design from the midterm review, reducing the taper ratio below 0.35 ended up reducing the obtainable lift below 188[N]. This was ascribed the low local Reynolds numbers at the wing tips, leading to earlier separation and tip stall. In conclusion the shortest tip chord should not be less than 0.085. The geometry of the final planform is given in Table 5.3.

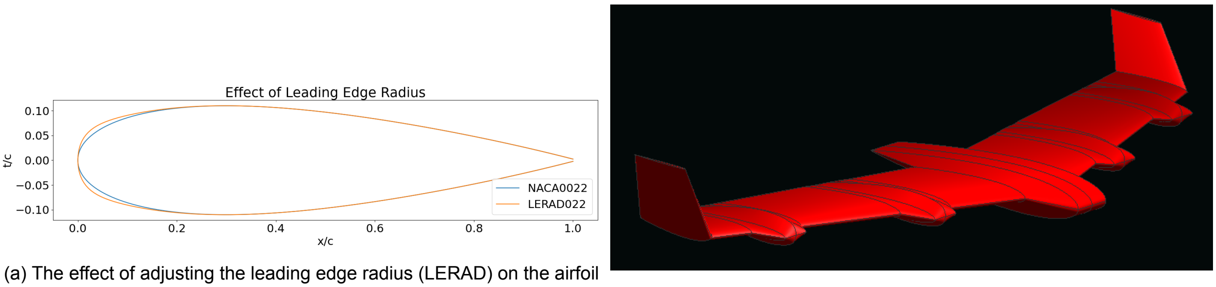
With the mean aerodynamic chord of 13[cm], the thickness of the airframe with a NACA0012 airfoil is 1.5[cm]. Requirement AE-PHYS-SIZE-01.3 is thus verified.

5.3.3. Electronic Pods

The aerodynamic frame should enclose all other components in order to be able to evaluate the aerodynamic performance. The wing itself has a too small thickness to encompass motors and the battery, and thus separate bodies must be designed to enclose them.

During the Midterm phase of the design, it was discovered that XFLR5 does not include the drag of an aircraft body in the drag calculations. It was therefore decided that an attempt on modifying the lifting body should be made. Two different kinds of electronic pods were modelled in this way, motor cowlings and the battery pod.

In XFLR5, a wing is defined by defining the cross section of the wing at a given span-wise location. Between these locations, the wing surface is interpolated. The motor cowlings will stick out in front of the leading edge, and blend into the wing before the trailing edge. The forward offset was determined based on the leading edge sweep and propeller diameter. The airfoil thickness was sized to equal a motor diameter of 4cm given the chord length, the distance between the trailing edge and the forward offset. As the motors have a blunt front, a flat leading edge at the motor location was approximated by increasing the leading edge radius of the regular symmetric NACA00xx airfoil by a factor of two, and blending at 5% of the chord. The geometrical



(a) The effect of adjusting the leading edge radius (LERAD) on the airfoil geometry.

(b) Aerodynamic model with cowlings for battery and motors.

effect can be seen in Figure 5.2a.

The central electronics pod was sized based on the required battery dimensions, battery cg-position and desired airfoil. Equation 5.3 [2] provides the local thickness, y_c, as a fraction of the chord at a given location x_c for a NACA00xx symmetric airfoil. The thickness t_c is the total thickness divided by the chord. The minimum chord length and offset required to enclose the battery was determined and reported.

$$y_c = 5t_c(0.2969\sqrt{x_c} - 0.12160x_c - 0.3516x_c^2 + 0.2843x_c^3 - 0.1015x_c^4) \tag{5.3}$$

Figure 5.2b displays the XFLR5 model of the wing with motor cowlings and electronics pod, using the input in the left part of Table 5.4 and a sweep of 0°. In Figure 5.3, the drag polar for the wing planform without and with motor pods and winglets are plotted. For a given C_L, the model with pods has a higher C_D, as expected. Note that the pods increase the projected surface area used to determine both C_L and C_D. Therefore the actual difference in drag is higher when including pods. The model still predicts an obtainable lift above 187[N], verifying requirement AE-FLGT-ROTA-01.1. Furthermore, with a drag at 90[m/s] of 3.88[N] including a 1.25 safety factor, AE-FLGT-MOVE-01.1 is also verified. At last, the stall speed is 13.5[m/s], which means that AE-FLGT-MOVE-02.1 fails. The accuracy is still not considered high enough to justify a reduction of the 1.5 and 1.25 safety factors.

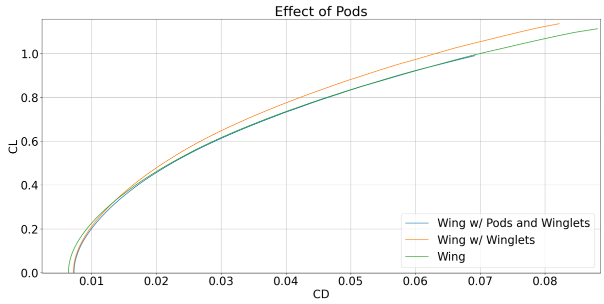


Figure 5.3: Effect of modelling the cowlings and pods as lifting surfaces.

Table 5.4: XFLR5 inputs for modelling the electronics pods. When analysing the effect of deformation for the flutter analysis the **dihedral**°and **twist**°columns on the right are replacing the **dihedral**°in the center. For the regular structure, the twist is 0.0 for all sections.

Sec	**y (m)**	**chord (m)**	**offset (m)**	**dihedral °**	**Airfoil**	**Y-panels**	**dihedral°**	**twist°**
1	0.000	0.247	-0.044	0.0	NACA0015	6	0.0	0.00
2	0.020	0.247	-0.044	0.0	NACA0015	3	0.0	0.00
3	0.030	0.132	0.011	0.0	NACA0012	2	0.9	-0.54
4	0.105	0.121	0.038	0.0	NACA0012	2	0.2	-0.75
5	0.115	0.138	0.024	0.0	NACA0011	2	0.2	-0.80
6	0.130	0.141	0.024	0.0	LERAD021	2	0.2	-0.88
7	0.145	0.144	0.024	0.0	NACA0010	2	0.2	-0.93
8	0.155	0.114	0.057	0.0	NACA0012	6	1.1	-0.95
9	0.275	0.096	0.100	0.0	NACA0012	2	0.1	-1.18
10	0.285	0.112	0.086	0.0	NACA0010	2	0.2	-1.21
11	0.300	0.115	0.086	0.0	LERAD026	2	0.2	-1.19

---

[2] `http://www.aerospaceweb.org/question/airfoils/q0041.shtml` Accessed 4th June 2021

Sec	y (m)	chord (m)	offset (m)	dihedral°	Airfoil	Y-panels	dihedral°	twist°
12	0.315	0.119	0.086	0.0	NACA0009	2	0.1	-1.20
13	0.325	0.088	0.119	0.0	NACA0012	2	0.4	-1.22
14	0.345	0.085	0.126	90.0	NACA0012	3	90.0	-1.18
15	0.403	0.060	0.161		NACA0006			0.00

5.3.4. Propeller Effects

Since Peregrine has propellers directly in front of the wing, there will be some interaction between these propellers and the wing. This interaction will increase the lift obtained by the wing, as the slipstream that washes over the wing has an extra velocity and thus locally increases the dynamic pressure.

Since the VLM models in the software used for the aerodynamic analysis can not account for this effect, an analytical approach was taken from [12]. The analytical model as described in [12] assumes the propeller outputs a cylindrical slipstream. The axial velocity inside the slipstream increases as the distance to the propeller increases, until it reaches a final value. The swirl (rotational) velocity remains constant from the propeller onwards. This swirl velocity will locally increase or decrease the angle of attack of the wing behind it, causing an increase or decrease in the local lift. The slipstream tube of the propeller will contract as the distance from the propeller grows until it reaches it’s final fully contracted diameter.

The relations used in the model of [12] use the following assumptions:

- The swirl of the slipstream can be neglected
- The slipstream tube diameter remains constant
- The axial velocity inside the slipstream tube at the wing is uniform across the wing.

The Lift Coefficient of the flying wing can be decomposed into three components. The Wing and slipstream contribution, the normal force on the propeller and the normal component of the thrust force:

$$C_L = C_{L_{W+S}} + C_{L_P} + C_{L_T} \tag{5.4}$$

$$L_W = \dot{m}_0 V_0 sin\epsilon = \rho\frac{\pi}{4}b_W^2 V_0^2 sin\epsilon \tag{5.5}$$

Since the angles of attack are relatively small, the focus of this analysis will be on the lift generated by the wing and propeller combination. The method outlined in [12] is based on the lifting line theory where the lift generated by a wing can be expressed by Equation 5.5. In this method the lift of the propeller is equal to the momentum generated by deflecting the streamtube down by the wing downwash angle ϵ. The lift generated by this deflection can be computed using Equation 5.6:

$$L_S = n_e \dot{m}_S (V_0 + \Delta V) sin(\epsilon_S) = n_e \rho \frac{\pi}{4} D^{*2} (V_0 + \Delta V)^2 sin\epsilon_S \tag{5.6}$$

Where D^* is the contracted slipstream diameter, which is dependant on the propeller diameter freestream flow velocity and the velocity increase over the propeller. It can be calculated using Equation 5.7. Where the velocity increase denoted as ΔV can be calculated using Equation 5.8.

$$D^* = D\sqrt{\frac{V_0 + \Delta V/2}{V_0 + \Delta V}} \tag{5.7}$$

$$\Delta V = V_0(\sqrt{1 + C_T \frac{S_W}{n_e \frac{\pi}{4} D^2}} - 1) \tag{5.8}$$

To calculate the total lift coefficient, first the total lift is calculated by summing the lift generated by the deflected propeller slipstream with the lift generated by the wing, subtracting the area corresponding to the parts of the wing in the slipstream. Finally this lift is converted to the lift coefficient of the wing by dividing by the dynamic pressure and surface area:

$$C_{L_{W+S}} = \frac{2}{S_W}(\frac{\pi}{4}b_W^2 - n_e\frac{\pi}{4}D^{*2})sin\epsilon + n_e\frac{\pi D^{*2}}{2S_W}\frac{(V_0 + \Delta V)^2}{V_0^2}\sin\epsilon_S \tag{5.9}$$

According to [12] ϵ_S, the downwash due to the slipstream can be calculated using Equation 5.10

$$\sin\epsilon_S = \frac{2C_{L_{\alpha_{s,eff}}}}{\pi A_{s,eff}}\sin\alpha_S \tag{5.10}$$

Where $C_{L_{\alpha_{s,eff}}}$, the effective lift gradient of the wing affected by the slipstream of the propeller can be found in figure 4 from [12]. The effective aspect ratio of the pieces of the wing that are blown over by the propeller ($A_{s,eff}$) can be calculated using Equation 5.11

$$A_{S_{eff}} = A_S + (A_W - A_S)(\frac{V_0}{V_0 + \Delta V})^{(A_W - A_S)} \tag{5.11}$$

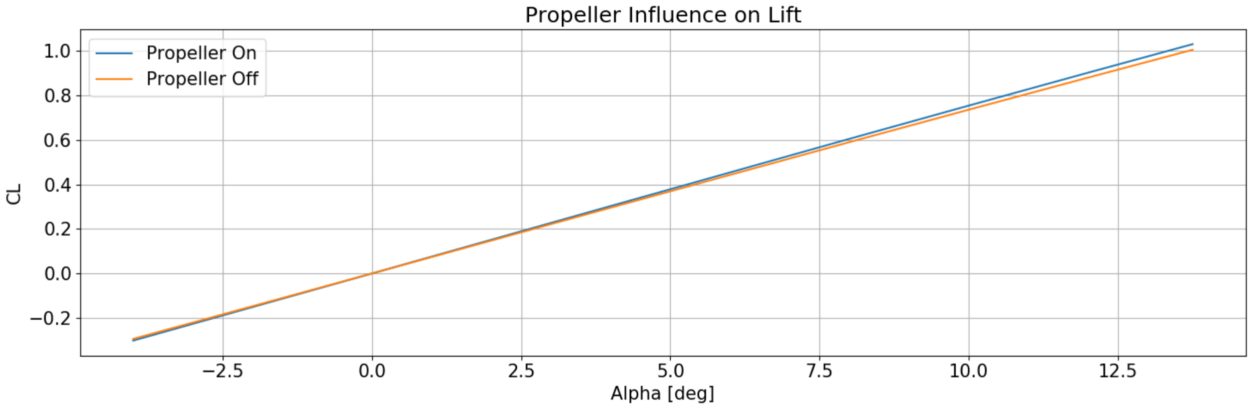


Figure 5.4: Comparison between the propeller on and propeller off condition, at V_0 = 60[m/s]

Finally α_S, the angle between the propeller slipstream centerline and the zero lift angle of attack (α_0, 0 for the selected symmetrical airfoil), can be computed using Equation 5.12. However since the propeller slipstream will accelerate the flow, it will change its direction with α^*. This change in direction is dependant on the thrust generated by the propeller and can be calculated using Equation 5.13. The last remaining component ϵ, the downwash right behind the unaffected wing can be determined using Equation 5.14.

$$\alpha_S = \alpha^* + i_{c_S} - \alpha_0 \quad (5.12)$$

$$\alpha^* = \arctan \frac{V_0 \sin \alpha_R}{V_0 \cos \alpha_R + \frac{\Delta V}{2}} \quad (5.13)$$

$$\sin \epsilon = \frac{2C_{L_W}}{\pi A_W} \quad (5.14)$$

With all of these components expressed in known values the effect of the four propellers on the lift of the wing could be computed. For this the thrust coefficient was taken from Power & Propulsion. The results of these computations are given in Figure 5.4. Here it is visible that the increase in lift because of the propellers at turning velocity (60 [m/s]) is marginal. Because of the limitations in predicting the maximum lift coefficient (elaborated in Section 5.5) this small increase in lift was not taken into account, effectively imposing another, relatively small, safety factor on the design.

5.3.5. Elevon Effects

While the elevon sizing itself was not part of the requirements outlined in Section 5.2, Aerodynamics did provide a means of verification for the Control & Stability department. From Control & Stability several possible combinations between chord ratio and span ratio for the elevons were extracted. All of these combinations would make the attitude rate requirements at a speed of 45 [m/s] with a maximum deflection of 10 degrees. To verify this the different elevon geometries were tested in AVL. Next to the needed deflection, the generated hinge moment was calculated as well. The geometry that was selected runs from 26% to 95% of the span and takes up 40% of the chord (starting from the trailing edge). This configuration was chosen as it proved to have a relatively low hinge moment coefficient and was stiffer in torsion compared to control surfaces that would span a larger part of the wing.

5.3.6. Hover Drag

In order to ensure that Peregrine can be operated in cross-winds and be controllable in hover condition, the hover drag must be evaluated. Tools, such as XFLR5, does not converge at 90°angle of attack. Therefore papers on the drag of bluff bodies were considered. The drag coefficients of cylinders of various cross sections are provided in [24]. Here, a rectangular cylinder with rounded edges has a C_d of 1.6 to 2.2 depending on the height over width. Elliptical cylinders experience a C_d around 1.6. A paper on the NACA0012 airfoil, [45], gave a C_d at 90°angle of attack to be 2.0. Based on these experimental results, it was decided to use a C_D of 2.3, to include a safety factor to account for three dimensional effects.

5.3.7. Winglets

As Peregrine is a tail-sitter, it must be stable with the tail on the ground. Therefore it needs three points or more, that are not located on the same line, to touch the ground at the same time. As this necessitates a vertical offset, additional structure is required. In order to make this structure serve an aerodynamic purpose, it will be included as winglets. The winglets will be symmetric about the tip chord of the horizontal wing, and extend behind the trailing edge, ensuring four contact points on a horizontal surface.

Due to the requirement on fitting through a circle with 70[cm] diameter, the height of the winglets limits the available span of the horizontal wing. At an early stage in the detailed design process, it was decided to estimate the winglet height based on tip over angle. The measurements were taken from the concept design of the midterm phase. The following assumptions and measurements were used:

- 15°tip over angle. The tip over angle is displayed in Figure 5.5.
- Center of gravity is located at 40% c_r from the leading edge.
- The winglet extends 1[cm] behind the trailing edge of the wing.
- $c_r = 0.13$[m], $x_{LE_{tip}} = 0.17$[m], $c_t = 0.089$[m]

Using the Pythagorean theorem and the definition of the tangent, a one-sided winglet height of 5.8[cm] was obtained. Even though the wing-planform reduced in height, the winglet height was considered beneficial for aerodynamic efficiency. Therefore the final iteration has a tip-over angle of about 20°.

After modelling electronic pods and evaluating the hover drag, new estimates could be made with respect to the cross-wind requirement, SYS-ENVI-WTHR-01. With a C_D of 2.3, a winglet height of 8.2[cm] would be required in order to stand still perpendicular to the wind direction. Keeping the 5.8[cm] winglet height, it can stand still on the ground with 4.2[m/s] perpendicular crosswind, or 5[m/s] crosswind at an angle larger than 32.7°with respect to the wing surface normal. The unacceptable directions are showed in red in Figure 5.6. As it seldom will be necessary to land in a specific orientation, the two 110 °sectors were deemed sufficient for take-off and landing. The winglet height was therefore not changed.

Figure 5.5: Illustration of the 15°tip-over angle.

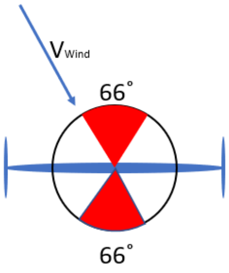


Figure 5.6: Unacceptable Cross wind directions when standing on the ground.

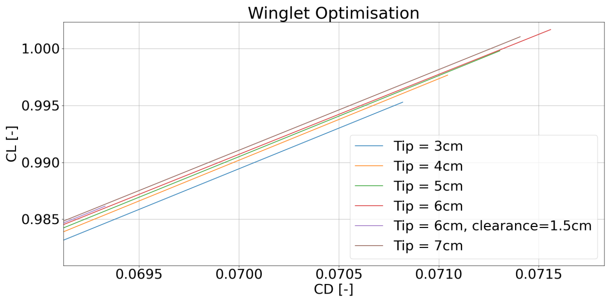


Figure 5.7: Iterations on winglet tip chord length and ground clearance.

Performing iterations on the winglet tip chord length, led to a tip chord length of 6[cm]. The clearance of 1[cm] was also deemed to be beneficial from an aerodynamic perspective. As can be seen from Figure 5.7, the gains in C_L and C_D are relatively small, and further optimisation was not performed. In XFLR5, it is only possible to model winglets on one side of the wings. Therefore the zero lift drag will likely increase. The drag at maximum lift may however decrease. With a span of 69[cm] and a winglet height of 5.8[cm], the Pythagorean theorem can be used to verify requirement AE-PHYS-SIZE-01.1.

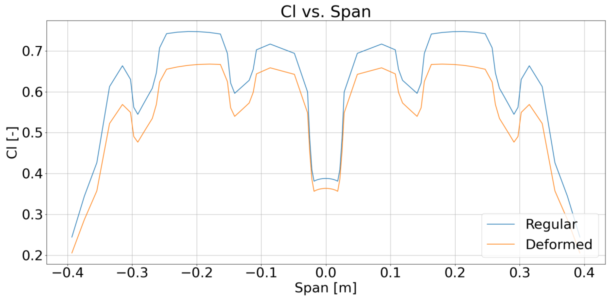
5.3.8. Flutter Analysis

Flutter is an aeroelastic phenomenon, happening due to the interaction between the fluid and a solid elastic structure [16]. For flutter to occur in a wing, two degrees of freedom are required, most often the wing bending and twist angle. In this section, the potential for flutter is evaluated by analysing the redistribution of aerodynamic loads caused by the structural deformation.

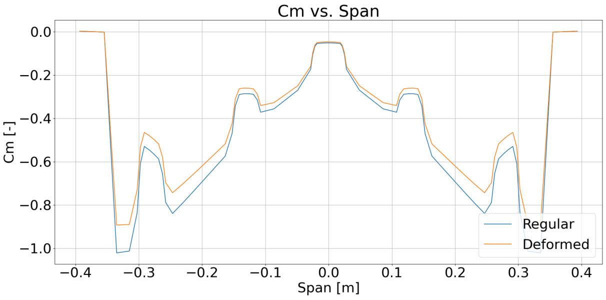
The Structures & Material departments provided results after implementing the finite element method, as described in Section 6.3. The results consisted of two files, one with the initial Cartesian coordinates of every node in the wing, and one with the linear displacement along each axis. The coordinate system is defined from the leading edge of the central pod, with the X-axis pointing forward in the direction of flight, Y-axis towards the right wing and Z-axis down.

In XFLR5, the lifting body must be defined as entire cross sections at given span-wise locations. In order to translate the node displacements into XFLR5, the following assumptions were made:

- The geometry of the local cross section does not change. Therefore the local airfoil shape and chord length remains constant.



(a) Cl vs. alpha for the deformed and regular wing.



(b) Cm vs. span for the deformed and regular wing.

- The bending moment around the X-axis is causing constant Z- and Y-displacements within a cross section.
- The deformations due to shear can be neglected.
- The angles of twist can be considered small, so the small angle approximation can be applied.
- The deformation in X-direction is caused by twist along the Y-axis.
- The deformation in Z-direction is caused by twist along the Y-axis and bending around the X-axis.
- The deformation in Y-direction is caused by bending around the X-axis.

For a given spanwise position, every node located less than 1 [mm] away along the span was included in the cross section calculation. First, the twist angle, $\Delta\beta$ and the linear offset in Z, ΔZ was determined by linear least squares regression, adding Equations 5.15 and 5.16 for every point to the system of equations. Here ΔZ_{Tot} denotes the measured deformation in Z-direction, while Z_0 denotes the initial Z-position, etc. As XFLR5 don't allow vertical offset to be entered directly, the dihedral angle Γ_i was calculated using Equation 5.17. Here ΔZ_{i+1} represents the Z-deflection of the next cross section when going in the direction of the wing tip, and Y_i is the spanwise positions. At last ΔY_{Tot} was determined by taking the mean of the Y-deformations.

$$\Delta Z_{Tot} = \Delta\beta \cdot X_0 + \Delta Z \qquad (5.15) \qquad \Delta X_{Tot} = \Delta\beta \cdot Z_0 \qquad (5.16) \qquad \Gamma_i = \arctan\frac{\Delta Z_{i+1} - \Delta Z_i}{Y_{i+1} - Y_i} \qquad (5.17)$$

The deformation effects on the XFLR5 input can be seen in the two rightmost columns of Table 5.4.

Torsional divergence, where the lifting surface deforms in a way that increases the lift and hampers static stability, will not be an issue. This can be noted from the negative twist angles, and the reduction of spanwise lift distribution seen in Figure 5.8a.

Control surface reversal is another static aeroelastic phenomena that can occur, and happens when the elevon introduces a torque twisting the wing to effectively change the angle of attack in the opposite direction [16]. This was not analysed in depth due to lack of time. It would further require the Structures and Materials department to perform FEM on a wing with a separate elevon, and the cross sectional load distribution would have to be estimated in further detail. If it is an issue, the effect would be countered by making the wing structure stiffer.

The dynamic flutter occurs when the wing fails to damp out the oscillatory aerodynamic loads caused by a disturbance in the aeroelastic system [16]. If the speed is higher than the critical flutter speed, the vibrational amplitudes diverge, causing structural failure. The analysis presented this far can not conclude whether flutter will occur below 90[m/s]. For this, time dependence and structural damping must be modeled with higher fidelity tools. This was considered infeasible in the time frame of the DSE, but have been planned for in the project Gantt-Chart, Figure 14.4. If the flutter speed found to be below 90[m/s], the problem could be solved by changing the local stiffness of the wing or locally redistributing mass.

5.3.9. Environment

In this subsection, the investigation of requirements related to the environment, **AE-ENVI-WTHR-02.1**, **AE-ENVI-ALTD-01.1** and **AE-ENVI-NOIS-01.1**, will be presented.

For **AE-ENVI-WTHR-02.1**, the critical phase was considered to be take-off and landing. With a C_D of 2.3, the hover drag at a crosswind of 5[m/s] would be about 3[N]. Having 6[N] of horizontal thrust to counteract it, this requirement is met while flying. For the case of standing on the ground, see Subsection 5.3.7.

Requirement **AE-ENVI-ALTD-01.1** was verified performing analyses at an ISA-altitude, one thousand meter, by changing the air density to 1.111. The airframe would then be capable of delivering about 90% of the lift at sea level at 60[m/s]. The increased drag and limited thrust available will limit the maximum wing loading while turning, and the maximum velocity, but it will be flyable with a sufficient margin.

AE-ENVI-NOIS-01.1 was to be verified by similarity, comparing a standard five-inch propeller to the propellers of Peregrine. Due to lack of time, experimental data of propeller noise levels have not been found, but are planned among the post-DSE activities in Figure 14.4. Testing would also be an option once the propellers have been manufactured.

5.4. Risk Assessment

Risk from the aerodynamic perspective is primarily related to the aerodynamic forces that must be obtained in order to meet the user requirements. The primary risks and mitigation strategies are as follows:

- AE-R1: The lift prediction is inaccurate, making it impossible to achieve a 25g turn.
  - This is likely, due to the numerical models used being inviscid, with viscous interpolation of the data, while the maximum lift is heavily dependent on viscous phenomena, such as separation. The risk has been mitigated through the use of a large safety factor of 1.5. If the lift is severely underestimated, care should be taken during operations, as the structure will not be designed for higher loads.
- AE-R2: The drag performance is underestimated, making the thrust insufficient to reach 90 [m/s].
  - This is again likely, as it is a viscous phenomena, and has been mitigated by the use of a safety factor of 1.25.
- AE-R3: The surface is assumed too smooth to be manufactured, decreasing performance below accepted levels.
  - The assumed surface roughness is unknown, making the probability and severity significant. The mitigation of this risk is included in the safety factors presented on lift, but will have to be tested with higher fidelity models and prototypes.
- AE-R4: The geometry is not accurately represented, hampering the computed load distribution.
  - The structure is optimised to handle a certain lift and drag distribution. In the worst case, this would lead to structural failure if the local stresses increase beyond the yield strength. In less severe cases, the drone will have a sub-optimal structure that might struggle to meet the requirements.
- AE-R5: The aerodynamic frame is too small to enclose all required components.
  - The risk is accepted, but it is vital to communicate between the subsystems to ensure that every component has a suitable location. Furthermore, the presence of cowlings must be accepted, creating a less optimal aerodynamic design.
- AE-R6: Flutter causes failure of the structure inhibiting regular operations.
  - The risk is significant, as it is a light structure operating under high velocities. It was analysed, but at present stage, it can not be ruled out as a problem. For now it is accepted, but should be assessed in future testing.

5.5. Verification & Validation

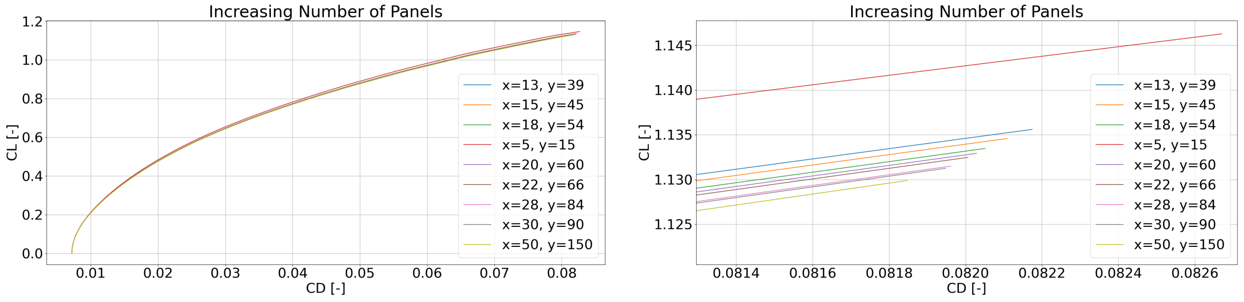
This section will go into the verfication and validation that has been done for the numerical models and tools that have been used during the aerodynamic analysis. Subsection 5.5.1 will discuss XFLR5 while Subsection 5.5.2 will elaborate upon AVL. Finally Subsection 5.5.3 will outline the procedures taken for the produced python code.

5.5.1. XFLR5

Verification

Previously the use of XFLR5 has been verified for the aerodynamic analysis of Peregrine [4]. For the final phase of the conceptual design the detail will need to be increased. This increased detail calls for a more in-depth understanding of the limitations of the tool. Therefore along with the verification outlined in [4] some extra steps have been taken in both the airfoil analysis and the wing analysis.

In the airfoil analysis two variables were changed to see their effect, the Reynolds number and the amount of points the airfoil was described with. From these results it was concluded that as the Reynolds number goes up, the drag goes down and the maximum lift coefficient goes up. Furthermore the results showed that at low Reynolds numbers (around 10^5) the lift curve has two linear parts, indicating a laminar separation bubble. This can be expected as at this flow condition these bubbles can appear. It is also showed that the



(a) Effect of increasing the number of panels.

(b) Closeup effect of increasing the number of panels.

amount of points the airfoil is described with has some influence on the results. In general the airfoils that have more points will obtain a higher lift coefficient. This effect is strongest at the lower Reynolds numbers. One explanation for this could be the increased dependency of the model on the geometry at these flow conditions. The drag is less sensitive to the amount of points used, only giving a difference at the higher angles of attack. This difference mostly originates from the difference in the lift curve.

When proceeding with the wing analysis the amount of panels has been varied to get a grasp on the discretisation error. The results of this variation can be found in Figure 5.9a and the closeup in Figure 5.9b . Here it can be seen that the aerodynamic performance of the wing drops as the amount of panels are increased. However one would expect this decrease in performance to converge to a final value, however this is not the case. Instead, the performance keeps dropping with increasing panel numbers. Because of this the need to validate XFLR5 grew, as this gives an indication on how reliable the default amount of panels is.

Furthermore a sample wing was created on which first the aspect ratio and second the leading edge sweep were varied. The results of this can be found in Figure 5.10 . In general it can be seen that when the aspect ratio increases the lift curve slope increases and the induced drag decreases as one would expect. With increasing sweep the lift curve slope decreases and the induced drag increases, again meeting the expectations. With both the wing analysis and the airfoil analysis verified the validation procedure can start.

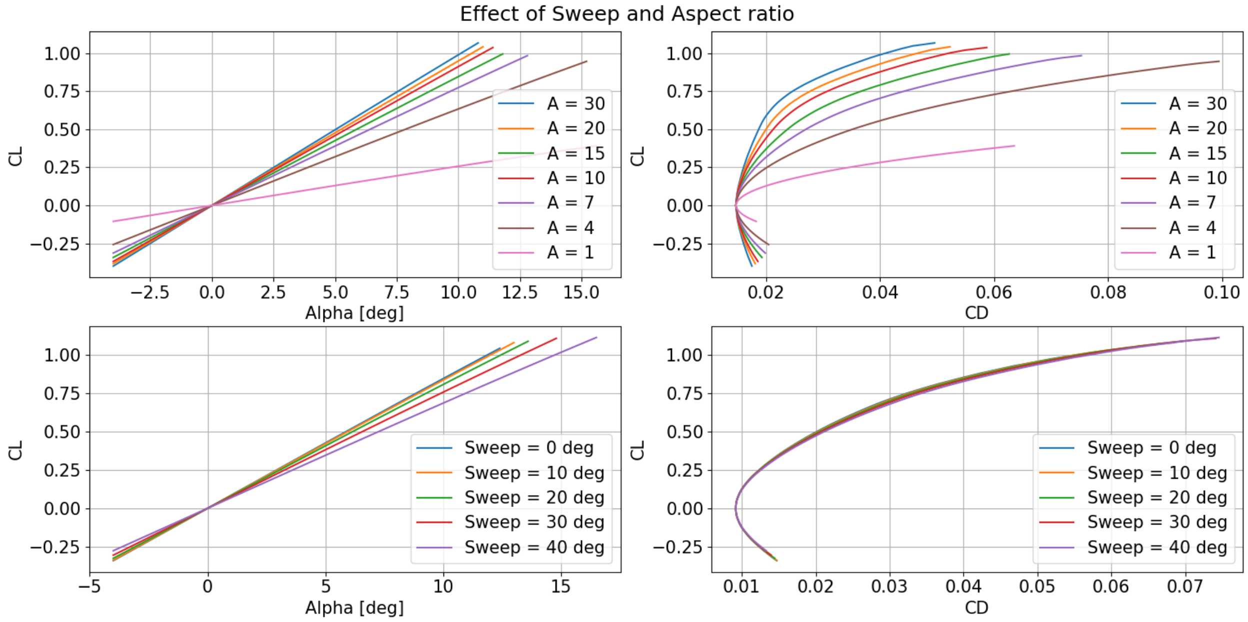


Figure 5.10: Lift curves and drag polars for 3D wings in XFLR5 with varying leading edge sweep and aspect ratio

Validation

In order to validate XFLR5 a similar approach has been taken as in [4]. The validation started at the airfoil analysis and continued with the wing analysis, since this uses data of the airfoil analysis. Both compared wind tunnel tests and numerical results by replicating the windtunnel environment in XFLR5. This was done by using the same Reynolds number or flow conditions (and thus Reynolds number) as was used in the wind tunnel experiment.

The airfoil validation used the NACA 0012 airfoil, as from the midterm analysis it was clear that this airfoil was a great contender (and would go on to become the chosen airfoil). The experimental data was taken from [36] and is displayed together with the XFLR5 results at the same flow conditions in Figure 5.11. The results of XFLR5 at these same flow conditions are depicted in Figure 5.11. In general three trends can be observed. First of all the $C_{L_{max}}$ of the airfoils is overpredicted regardless of the Reynolds number. Second, the drag calculated by XFLR5 is smaller than the experimental drag. Finally it can be seen that in the experimental results the lift curves for the different Reynolds numbers converge as the Reynolds number increases. However this is not the case for the experimental results as the maximum lift keeps increasing with increasing Reynolds number. All of these phenomena might be explained by two factors. First of all XFLR5 uses XFOIL to compute the airfoil characteristics. XFOIL makes use of some analytical and semi-empirical models to predict seperation of the flow. These models might be optimized for different flow conditions and in general are limited in their accuracy. Secondly XFOIL assumes a perfectly smooth airfoil and no disturbance in the oncoming flow. However for the experimental data this is not the case as the airfoil can't be perfectly smooth and there will always be disturbances in the flow such as wall effects. This means that the lift and drag obtained from the airfoil analysis should be used conservatively, as the performance of the airfoil is overestimated.

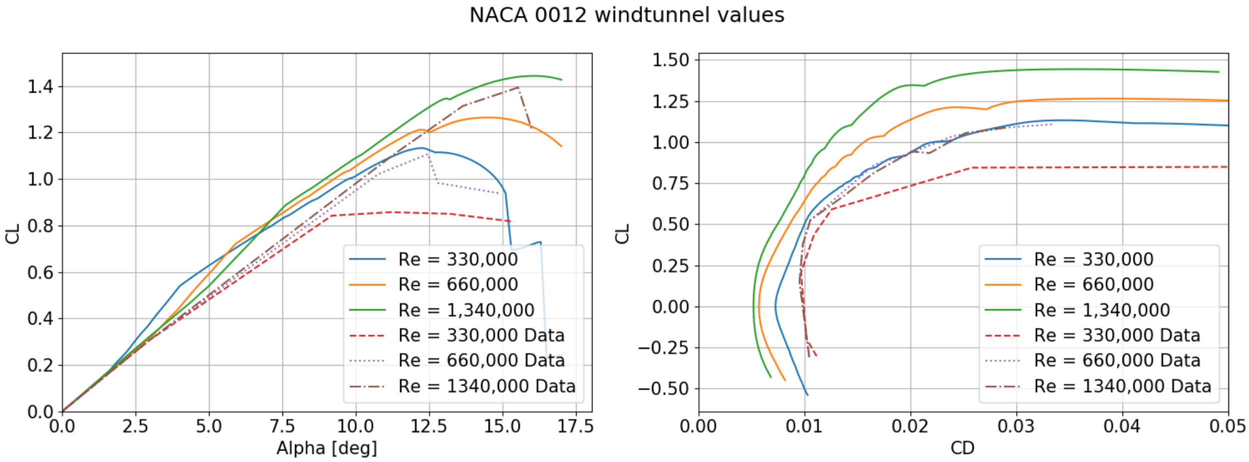


Figure 5.11: Comparison of the lift and drag characteristics of the NACA 0012 airfoil, between XFLR5 and the experimental results from [36]

For validation of the wing analysis two studies have been used to compare XFLR5 with. The first study [48] examined the lift performance of a wing with a NACA 0012 airfoil. In Figure 5.12 the experimental and numerical lift curves are compared. In general it is visible that XFLR5 does not produce results after stall, only modelling the linear part of the lift curve. When comparing the two lift curves it can be seen that XFLR5 slightly overestimates the maximum lift coefficient and has a slightly higher lift curve gradient.

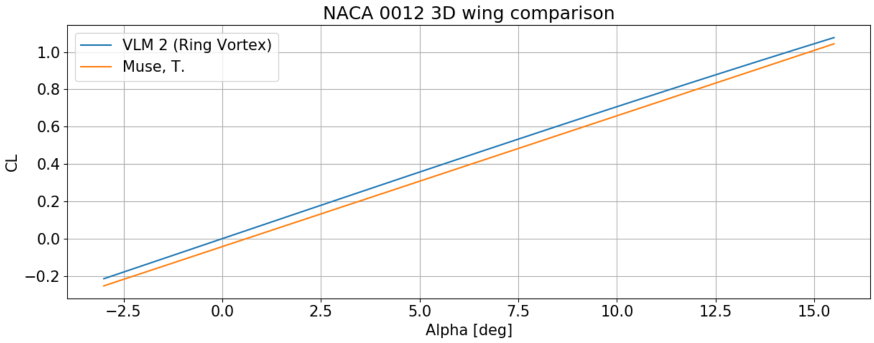


Figure 5.12: Comparison between the lift curves of experimental wind tunnel data and XFLR5 wing analysis at Re = $1.07 \cdot 10^6$

A second study, by Sivells and Spooner [60] has been used to assess the body analysis and the drag computations. The results of this comparison are depicted in Figure 5.13. From this figure it can be seen that in this case XFLR5 actually underpredicts the maximum lift achieved. Also the lift at zero angle of attack is higher, causing the entire lift curve to shift up. When looking at the drag it can be seen that XFLR5 estimates the induced drag relatively well, but the zero lift drag remains smaller than the experimental data, as was the

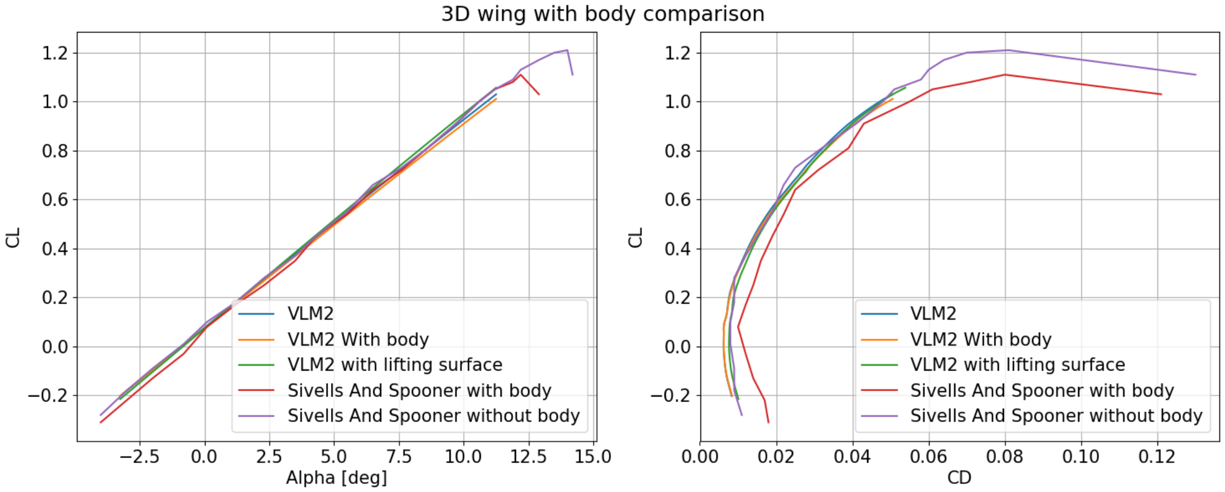


Figure 5.13: Comparison between the experimental results from Sivells and Spooner and the numerical output of XFLR5 with and without body.

case for the airfoil analysis. Finally it can be concluded that the body analysis severely underestimates the drag generated by the body and is thus a not fit for use in the aerodynamic analysis.

From the verification and validation it can be concluded that XFLR5 in general provides calculations that can be used in the design. However for the maximum lift coefficient and the zero-lift drag coefficient, a safety factor will be needed to make sure the final design will meet the requirements. This is because in general XFLR5 underpredicts the drag and overpredicts the obtained lift (except when compared to [60]). Furthermore the amount of panels used in the analysis changes the performance significantly. For these reasons the safety factor used on the maximum lift was 1.5 (So the target lift was 1.5 times higher than the lift needed). The safety factor used on the drag coefficient was 1.25 as in general the drag estimation proved to be less sensitive to the amount of panels and was relatively accurate.

5.5.2. AVL

AVL has primarily been used for the sizing of the elevons and estimating the hinge moments that result from the elevon deflection. In order to validate this calculation the experiments in [35] have been recreated in AVL. The experiment compared the hinge moments of three different aspect ratio wings with a sweep of 35°. The results are given in Table 5.5. From these results it can be concluded that the hinge moment coefficient calculated by AVL gets more accurate with increasing aspect ratio. Furthermore AVL overestimates C_{h_δ} considerably for aspect ratios higher than 3. However for the purpose of this conceptual design it gives a sufficiently accurate result, as it is on the conservative side.

Table 5.5: C_{h_δ} as given by AVL and the experimental results from [35]

Aspect Ratio	AVL Results	Experimental Results
3	-0.0063	-0.0076
4.5	-0.0090	-0.0078
6	-0.0089	-0.0082

5.5.3. Python Scripts

Python scripts were used for creating plots, and geometric computations, such as generating the wing planform input to XFLR5 for a desired sweep, area and taper ratio. Especially as the planform became more complicated, including motor cowlings and pods, automation of the input generation saved time. Python was not used directly to evaluate the aerodynamic performance of the design.

The python codes were verified by the use of unit tests. The output of the program was compared with hand calculation for a problem with corresponding input, and corresponding outputs were ensured to within rounding error.

Code used for geometric computations were validated by entering the input into XFLR5. The display of the wing clearly visualise discrepancies violating design policies, such as a constant trailing edge sweep, and the thickness distribution around the pods.

Code used for plotting was validated by comparing the output graphs with graphs automatically generated by XFLR5.

5.6. Compliance Matrix

After all the analysis and iterations, the final aerodynamic design can be characterised by the results from Table 5.6.

Table 5.6: Final characteristics of the aerodynamic design

Parameter	Value	Requirement
Lift at 60 [m/s]	125 [N] (including safety factor)	124 [N]
Drag at 90 [m/s]	3.88 [N] (including safety factor)	13 [N]
Wing Span	69 [cm]	70 [cm]
Hover Drag at minimum velocity	27.3 [N]	4.25 [N]
Aspect Ratio	6.22	
Root Chord	13.0 [cm]	
Taper Ratio	0.62	
Leading edge sweep	20 [°]	

Table 5.7: Compliance matrix for aerodynamics subsystem.

Identifier	Status	Reference	Identifier	Status	Reference
AE-PHYS-SIZE-01.1	Pass	Subsection 7.3.2	AE-FLGT-MOVE-02.2	Fail	Subsection 5.3.3
AE-PHYS-SIZE-01.2	Pass	Subsection 5.3.2	AE-FLGT-MOVE-05.1	Pass	Subsection 8.4.1
AE-PHYS-SIZE-01.3	Pass	Subsection 5.3.2	AE-FLGT-ROTA-01.1	Pass	Subsection 5.3.3
AE-PHYS-SPMT-03.1	TBC	Subsection 5.3.8	AE-FLGT-ROTA-02.1	Pass	Subsection 9.3.3
AE-FLGT-HOVE-04.1	Fail	Subsection 5.3.6	AE-ENVI-WTHR-02.1	Pass	Subsection 5.3.9
AE-FLGT-MOVE-01.1	Pass	Subsection 5.3.3	AE-ENVI-ALTD-01.1	Pass	Subsection 5.3.9
AE-FLGT-MOVE-02.1	Fail	Subsection 7.3.2	AE-ENVI-NOIS-01.1	TBC	Subsection 5.3.9

As can be seen from Table 5.7 some of the requirements failed and other will need to be confirmed in a later stage. Two of the requirements that were not made, **AE-FLGT-HOVE-04.1** and **AE-FLGT-MOVE-02.1**, were set at the beginning of the DSE when the team's experience with drones was quite limited. The drag at hover condition is too high when staying vertical, however during transition the drone will pitch such that it will become horizontal before it reaches the minimum speed. In this orientation the drone has a significantly lower drag than in the hover orientation, thus the drag at this flight phase will actually be lower than the 4.25 [N] specified. When looking at the propellers, they will stall when the drone flies slower than about 50 [m/s]. However stalling of the propeller does not mean that the drone will not fly. The last failing requirement, **AE-FLGT-MOVE-02.2**, failed as the stall speed is too high. Due to the vectoring motors, the violation of this requirement will not hamper operations as the vectoring thrust can be used to counteract gravity. In Chapter 7 it can be seen that the propeller still provides adequate thrust even when it is stalled. Furthermore, this stall can be explained by the optimisation of the propeller in the 60-90 [m/s] velocity range, where the thrust is much more critical. Two other requirements will need to be researched further: **AE-ENVI-NOIS-01.1** and **AE-PHYS-SPMT-03.1** fall outside of the scope of this DSE. The flutter will be further analysed in the future as outlined in Subsection 5.3.8. The noise made by the propellers can be verified using similar drones and can be tested in the future with a flying prototype.

6

Structures & Materials

This chapter covers the structures and materials detailed analysis of the Peregrine drone. First a functional recap and subsystem requirements are presented, then analysis is conducted to validate as many requirements as feasible whilst designing the structure more definitively. After, subsystem risks are explored and then the verification and validation of the results is performed, finally allowing for the requirements compliance to be evaluated.

6.1. Functional Recap

The function of the structures and materials subsystem is to provide a skeleton for the drone which is capable of withstanding all the loads experienced whilst protecting the electronics and motors. To do so, a detailed stress and stiffness analysis must be performed, ensuring that the structure does not fail and does not allow significant deflections which could alter the performance of the drone in a race. The material (choice, properties and layout), cross sectional properties (thicknesses and sizing) and sub-structure attachments (design) are considered and optimised in structural analysis. Whilst providing support and protection to the electronics, the structural frame should function as an easily repairable and maintainable structure. This means that the drone structure must be designed to ensure that the user can repair and access the electronics without difficulty, all the while having assurance of the internal integrity and protection.

6.2. Requirements

The requirements, and the corresponding methods of validation pertaining to this subsystem are shown in Table 6.1.

Table 6.1: Structural subsystem requirements

Identifier	Requirement	Method
SM-FEAS-COST-01.1	The structure shall not cost more than €300.	Review of Design
SM-FEAS-SUST-01.1	The load carrying structure shall be made from 95% recyclable material by weight.	Review of Design
SM-PHYS-WGHT-01.1	The total mass of the structure shall not exceed 125[g].	Analysis
SM-PHYS-SIZE-01.1	The load carrying structure shall comply with the boundaries defined by the airframe.	Review of Design
SM-PHYS-SPMT-01.1	The load carrying structure shall be manufactured using existing technologies.	Review of Design
SM-PHYS-SPMT-02.1	The load carrying structure shall not deform plastically under maximum loading conditions.	Analysis
SM-PHYS-SPMT-02.2	The load carrying structure shall have a maximum deformation of 50[mm] when in a 25g turn.	Analysis
SM-PHYS-SPMT-03.1	*The propellers shall have a minimum stiffness of at least 0.5[GPa].*	Review of Design
SM-OPRT-TRAN-01.1	The battery shall be able to be replaced in less than 60 [s].	Review of Design
SM-OPRT-TRAN-01.2	The structure shall be assembled from transport configuration in less than 10 [min].	Review of Design
SM-OPRT-TRAN-02.1	The structure shall allow disassembly of the drone sufficient to fit in the defined transport volume.	Review of Design
SM-OPRT-TRAN-03.1	The structural frame shall be able to handle point loads up to at least 25 [N].	Analysis
SM-OPRT-MAIN-02.1	One control surface shall be able to be replaced in less than 5 [min].	Review of Design

Identifier	**Requirement**	**Method**
SM-OPRT-MAIN-02.2	The structure shall allow for direct access of inside electrical components.	Review of Design
SM-OPRT-FAIL-02.1	The structure shall be able to withstand an frontal impact with impact energy of 506.5 [J] without plastic deformation.	Analysis
SM-OPRT-FAIL-02.2	The structure shall prevent damage of the battery up to a maximum impact speed of at least 60 [m/s].	Analysis
SM-ENVI-WTHR-01.1	The structural assembly shall protect the electric components and battery from water splashes.	Review of Design
SM-ENVI-FATG-01.1	*The propellers shall be able to withstand 150 cycles for a stress range equal to the maximum stress experienced during turning flight.*	Analysis
SM-ENVI-FATG-01.2	The wing shall be able to withstand 150 cycles for a stress range equal to the maximum stress experienced during turning flight.	Analysis

6.3. Analysis

To optimise the structure and validate impact requirements, a stress analysis will be performed to ensure that failure does not occur and further understand the load paths internally. First material choice will be outlined, and once the stress is calculated for the critical and failure load cases, the structure will be minimised for its mass. After stress calculation, the point load analysis, impact analysis and fatigue analysis will proceed. After the main structure and material are selected, the supplementary structures like joining mechanisms and necessary cut-outs will be examined for the total structure. With the final structure system obtained, the mass budget and cost analysis can be presented.

For the following analysis, the coordinate frame is defined as the body fixed frame, giving the X axis as positive towards the nose, the Y axis as positive towards the right wing and the Z axis as positive downwards. This is visualized in Figure 6.1.

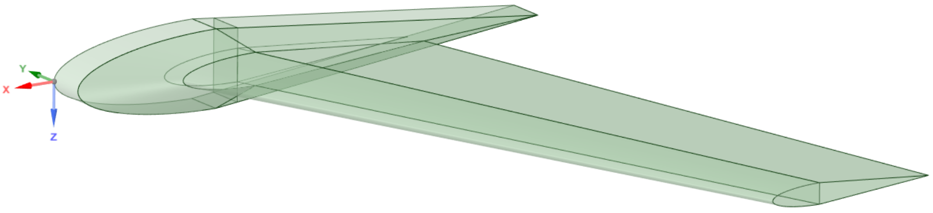


Figure 6.1: Coordinate axis frame

6.3.1. Structural Layout

Before the stresses and failures are analysed in detail, it is important to have a good understanding of the structure. Therefore a few key structural decisions relating to the layout of the structure will be detailed. The layout of structure can be seen in Figure 6.2

Firstly, the aim is to place the majority of the electrical components near the center of the wing. This reduces the moments of inertia of the drone, and reduces the length of electrical wiring required, reducing the weight. This also is desirable from a maintenance perspective. Putting all components close to each other in a easily accessible place makes executing repairs on the drone significantly easier. This central section will be referred to as the pod.

The skin of the drone is an aerodynamic shape designed to produce the aerodynamic forces. This results in a shape that is primarily governed by the aerodynamic design. This structure will be a hollow shell, that is designed to be as light as possible. The pod and wing will smoothly transition into one another, to avoid any undesired aerodynamic effects. Also the motors will be placed on the tip by a cowling the reduce the drag caused by the motors.

The main loads exerted on the structure will be carried trough a spar. This spar spans the full length of the wing and is located at 30% of the local chord: the position where the airfoil is thickest. The skin of the drone will transfer the aerodynamic loads to the spar. This is done to enable the spar to be as tall as possible, and

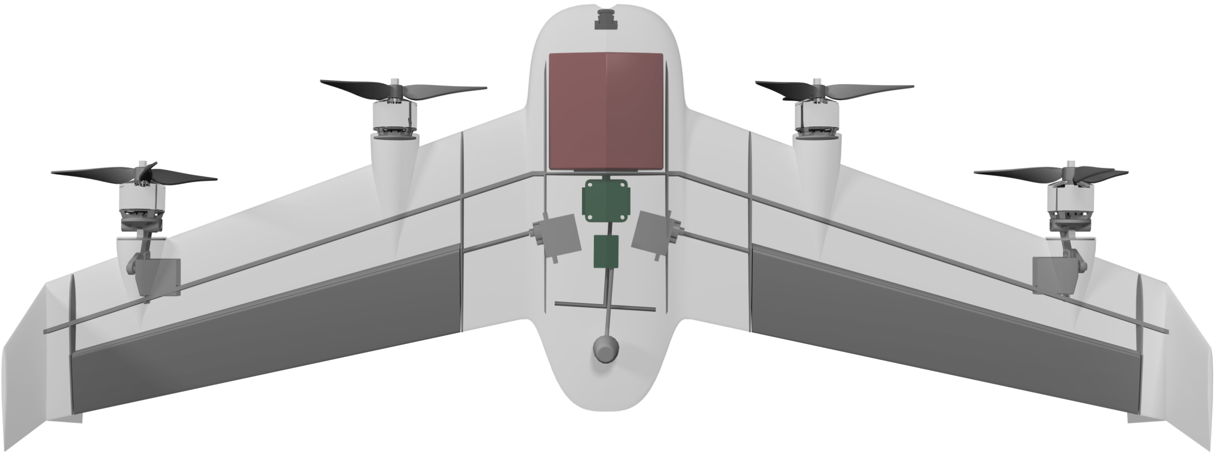


Figure 6.2: Layout of the structure

thus has the best resistance against bending. Throughout the pod, it is chosen to make the spar straight to allow for easier and more compact fitting of components.

Finally the structure will have several ribs. At least two ribs are required in the span of the wing. One at the root of the elevon, and one at the tip of the elevon. These are required since the hinge of the elevon needs some rigid place to be attach and transfer the loading to the rest of the structure. There will be an additional rib added at the end of the pod. This ribs primary function is to strengthen and stiffen the pod structure and the transition to the wing section. It also has the secondary function of protecting the electronics placed within the pods in case of a crash. Together with the spar, the ribs form the structural frame.

6.3.2. Material Selection

Following from the Midterm Review and previous analysis, it is clear that carbon fiber reinforced polymers (CFRP) are a viable and effective structural material for drone design. Due to the high stiffness and strength comparable to metals and lightweight properties, CFRP will be used for the load carrying structures of the drone. To obtain material properties, a fiber percentage and direction must be known. For the sake of analysis and due to resource commonality, epoxy resin will be used in composition with the carbon fibers. The properties of the composite can be changed to meet certain stress values if required later on in analysis, however defining a baseline value will be done. Since composite stress and strain performance is dependent on the direction of the force applied, further analysis will assume that the fibers have a fixed orientation with respect to the body axis of the drone, throughout the entire structure. Future orientation of the material can be detailed in later design stages.

The values that will be used for CFRP material properties for analysis are tabulated in Table 6.2. These values are taken from a study which analysed CFRP high velocity impact energy absorption characteristics [67]. The specific material uses T700 carbon fibers and 3234 epoxy resin, which was cured at 130 °C for 90 minutes and a pressure of 0.5 [MPa]. For a 3 [mm] sheet, it is expected that the material will have 36 plies [67]. The lay-up will be [0/90/45/-45]s °fibers. This makes the material quasi-isotropic. In doing so, the Von Mises stress criterion is valid for yielding and failure analysis. As discussed in [53] which compared different composite lay-ups and the accuracy of failure criteria to predict failure, Von Mises stress is not limited by large errors unlike other criteria, so long as the composite has at least one isotropic plane.

Using the laminate setup, the following properties are calculated[1]:

---

[1] Calculation software= `https://abdcomposites.com/` Accessed on 14th June 2021

Table 6.2: CFRP material properties (Dry) (Room Temperature) in XY frame, calculated using [67] as reference

Parameter	**Value**	**Units**
Fiber Fraction	60	%
Density	1570	[kg/m³]
Young's Modulus (Longitudinal)	52.8	[GPa]
Young's Modulus (Transverse)	52.8	[GPa]
Shear Modulus	20.5	[GPa]
Poisson Ratio	0.28	-
Ultimate Compressive Strength (Longitudinal)	1050	[MPa]
Ultimate Tensile Strength (Longitudinal)	2100	[MPa]

Supplementary to being stiff and strong, the structure must provide adequate protection to the internal electronics and motors. In impact situations where crashes occur, a frequent occurrence in drone flying, the drone loses significant amounts of kinetic energy to elastic energy. A material that is capable of receiving impacts and absorbing energies in all direction is desired for racing drones which have random and unpredictable crashes at high velocities. The capacity to absorb energy is dependent on the area under the force displacement curve [46]. Foam structures are good for this reason since they are capable of creating 'shoulder points' where the load can continue increasing without significant resulting strains, due to increasing stiffness capabilities during compression [46]. Foams are also advantageous since they are quasi-isotropic, cheap and lightweight when compared to solids [2].

Whilst many types of foams are possible, previous market analysis has indicated that the most common used foam in drones, especially flying wing drones, is expanded polypropylene (EPP) [3] [4]. This material can have a very low density (0.2-0.9 [kg/m^3] [5]) whilst having the benefit of excellent durability characteristics due to the high energy absorption possibilities. To add to this, the sustainability of the structure is maintained as the environmental impact of EPP is substantially low, given the proper processing, it is possible to achieve recyclability close to 100% [6]. EPP can also be manufactured and designed to the airfoil specifications, maintaining aerodynamic performance. A commercially available set of EPP data is provided by ARPRO, whom manufacture foams of multiple densities. A higher density foam would have greater energy absorption capabilities and mechanical properties [40], but would also not be beneficial to minimising the structural mass. Similarly a foam with too low a density would not be favoured given the high speed collisions that would occur with the drone. Therefore, a mid-ground is chosen based on the available foams. The foam chosen is that which has a density 60 kg/m^3, since it has relatively good energy absorption capabilities and mass characteristics. ARPRO EPP foam absorbs less than 2% of water [7], meaning that internal electronics will likely not be exposed to water during flight. The properties for this EPP selection are given in Table 6.3.

Table 6.3: EPP material properties [8]

Parameter	Value	Units
Density	60	kg/m^3
Young's Modulus	19	MPa
Compressive Strength at 25% Strain	0.39	[MPa]
Compressive Strength at 50% Strain	0.53	[MPa]
Compressive Strength at 75% Strain	1.07	[MPa]

Due to the clearly low stiffness properties, it is assumed that the load will be carried by the CFRP structure(s) whilst the EPP will act as a cushion and mass reducer to aid in crash performance and budget. This will be verified during the stress analysis; if the foam structure is shown to be carrying little to no stress, it will be assumed to not yield and cause deformation problems.

---

[2] `https://www.arpro.com/en-GB/technical/` Accessed on 9th June 2021

[3] `http://miniracewing.com/` Accessed 10th June 2021

[4] `https://www.tbrcwings.com/villain` Accessed 10th June 2021

[5] `https://www.arpro.com/en-GB/technical/` Accessed on 9th June 2021

[6] `https://www.intcorecycling.com/how-to-recycle-epp.html` Accessed 9th June 2021

[7] `https://www.arpro.com/contentassets/011998d779ef4e2b93d9a41b901be2fc/water-absorption-v02-en.pdf Accessed on 28th June 2021`

6.3.3. Stress Analysis- Von Mises Yield Criterion

The Von Mises Yield Criterion is a metric to check if the total deflections affecting a structure will cause yield or failure [43]. It therefore incorporates the normal stresses and shear stresses in its calculation, giving a holistic load case analysis. A basic Von Mises stress calculation can be made on a simple wing beam model, but for complex structures with multiple materials, it is simpler to use a Finite Element Analysis (FEA), where the geometry of the structure is inputted into the software, a mesh is generated, the loads are applied and a stress is computed. The use of FEA was done for the Peregrine drone design since it allowed for a higher accuracy and understanding of the Von Mises stress in a particular area.

The software used to calculate the Von Mises stress is the Ansys Workbench. It allows for the implementation of a CAD model and user-defined forces and accelerations to simulate the structure's behavior. To do so, the 'Static Structural' analysis tool was used. The material properties for the structure can also be directly attributed to the 'Engineering Data' Library that is associated with the software, where additional modifications to the material properties can be made manually fitting to those outlined in Subsection 6.3.2. Once the CAD model is imported and materials are applied, a mesh can be generated. To optimise the mesh, a convergence was set on the Von Mises stress, such that the mesh would become finer in required areas until the convergence requirement was met. This requirement was user defined to be 0.03 (a loop would be carried until the stress of the current stress was within 3% of the previous). Load application was done as follows for the turning flight procedure:

- The effect of gravity can be quantified as an acceleration acting as a vector dependent on the required bank angle for turning flight. A larger bank angle results in a greater acceleration on the body's Y-axis compared to its Z-axis.
- The lift can be modelled as a distributed acting at the center of pressure along with wingspan. This is valid as the center of pressure is where the various of lift effects are summed into one vector. The lift force uses a safety factor of 1.5 to ensure safety and requirement fulfillment. This modelling approach was chosen, as applying a uniform pressure force to the entire wing surface would significantly overestimate the applied loading at the trailing edge and lead to overestimation of the deflections.
- Drag effects are parallel to the X-axis in the negative direction and can be modelled by a distributed load acting at the trailing edge.
- The magnitude of thrust is such that it equals the total drag experienced on the body and acts as a point load at the propeller in the positive X direction.
- Electronics and motors can be modelled as point weights at their relative positions. Considering the main impact on the wing is the lift force, the error in the relative position is negligible.
- Winglets can be modelled as point masses with a point force equal to their lift effects. This is to simplify the model and design for a critical scenario where winglet stiffness does not aid against wing deflection.

The initial design was made of one material, that being carbon fiber reinforced polymer (CFRP) with a fiber volume of 60%. This meant that both the skin and spar, along with different sections of the drone were all assumed to be one material. This initial design choice was to compare the previous designs, but also functioned since CFRP has high stress carrying capabilities and stiffness. However, upon implementing this base design into the simulation, the mass of the structure was significantly greater than the budget, hovering around 160 g. Although this weight could be optimised by reducing thickness values across the structure to be within budget, it was immediately decided to split the design into multiple materials since further thickness reductions would become unfeasible for manufacturing and durability purpose [9].

For the sake of reducing weight to an acceptable level, the drone design was renewed to have an outer surface made of EPP, where CFRP would form the main load carrying structure. This follows a more conventional flying wing drone design. In the case of the spar and ribs would be made of CFRP and the skin formed of EPP. To test this renewed design, several combinations of skin thicknesses across the wing surface and spar thicknesses were analysed. Changes such as having differing thickness of the skin across the cross section or spar could also be analysed effectively using the Ansys software. Changing the material to a metal alloy was tested by resulted in favor of CFRP due to the lower mass required to be within yield limits.

Ansys results for the Von Mises stress would be outputted through an animation of the stress buildup throughout a 0.5 second loading. This loading follows through the turn time to perform a 180°turn flying at 60 m/s and centripetal acceleration of 25g. Results for a half wing were performed since the drone is symmetric about the xz plane and the loading is symmetric as well. The Von Mises stress results for the renewed design are displayed in Figure 6.3 which use the lift force (with applied safety factor) supplied by aerodynamics.

Initial inspection reveals that the EPP carries little to no load, resulting in almost non-existent stress across

---

[9] The thicknesses for the skin and spar required to be within mass budget were in the order of less than 0.1 [mm], reaching manufacturing limits.

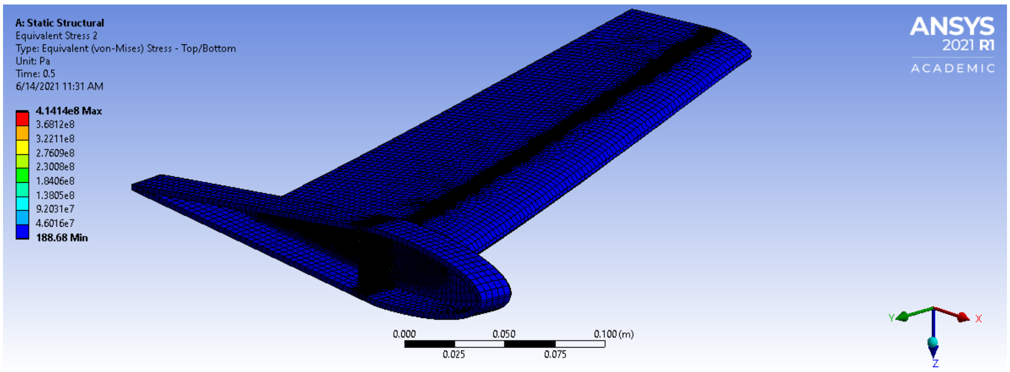


Figure 6.3: Renewed design stress results

the entirety of the outside of the drone. This is intuitive due to the low young's modulus of the EPP in comparison to the CFRP, and confirms the assumption resulting in the CFRP spar carrying almost all the load. Please also note that when the lift force is modeled as a uniform pressure applied across the entire wing surface instead of a distributed force at the center of pressure, the EPP still carried little load compared to the CFRP. It would primarily increase the deflections at the trailing-edge wing tip, due to the unrealistically high pressure near the trailing edge. Regardless the total deformation stays within required limit. Further inspection towards the pod spar indicates stress concentrations, as displayed in Figure 6.4.

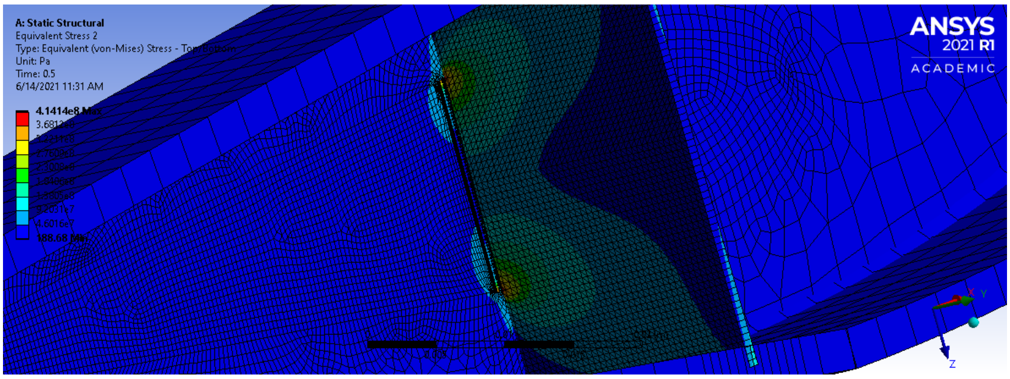


Figure 6.4: Renewed design stress results at pod spar

In addition to the pod spar loading, removing the surrounding EPP structure shows the stress build-up across the CFRP structure towards the root, shown in Figure 6.5.

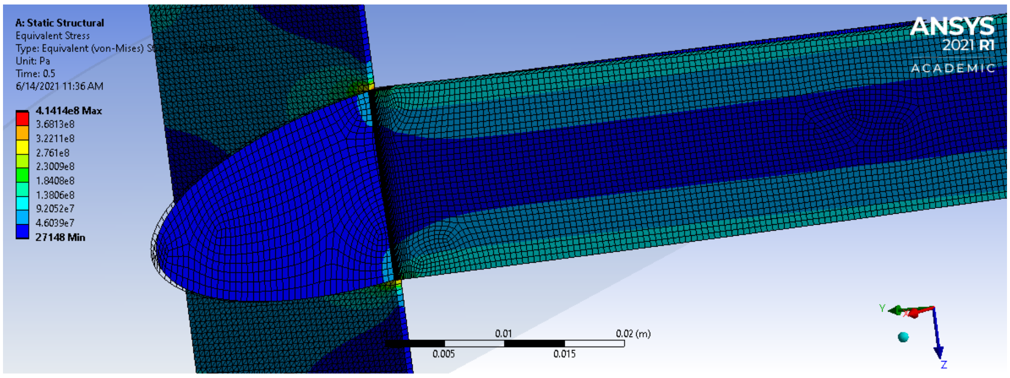


Figure 6.5: Stress progression in CFRP towards root

Removing the connecting face between the pod spar and wing spar reveals the maximum stress, shown in Figure 6.6. The maximum stress actually occurs on the pod spar, towards the top. This is logical given that the wing deforms upwards and twists. Such motion causes a larger compression in the upper part of the spar than the tension in the lower portion. Looking at the normal stresses provided by the program verifies this, as the principal stress in the bottom half is positive and 10 [MPa] smaller in magnitude than the upper half where the stress is negative.

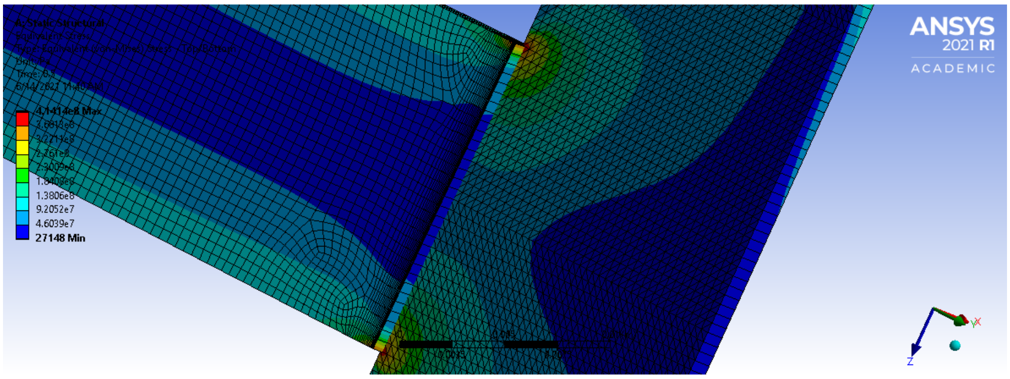


Figure 6.6: Maximum Von Mises stress at pod spar

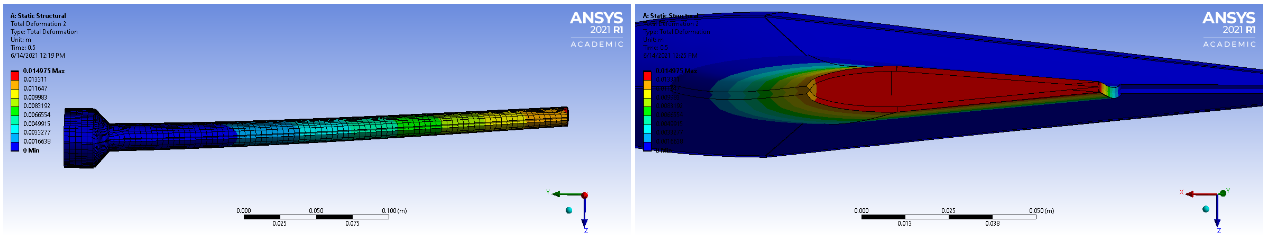
The fact that the pod spar experiences a greater Von Mises stress than the wing spar is because the pod spar is set to 1 [mm] whilst the wing spar is 3 [mm]. This was intentional to ensure the wing does not fail first, preserving aerodynamic performance. Pod spar yielding is also less consequentially significant since the face structure connecting the two spars can be used as a contingency for load carrying in such an event. The difference in stress between the two spars is minute; the wing spar achieves a stress of 384 [MPa] one node before it intersects with the pod face, meaning that if failure does occur, pod failure will be a warning of future wing failure.

To ensure that the maximum Von Mises stress of 414 [MPa] does not cause yielding to the structure, a safety factor must be applied. As discussed in the midterm [4], composite safety factors for failure are larger than metals, typically larger than 2. The decision at the time was to take a safety factor of 3, which was relatively conservative given the accuracy of lower safety factors being sufficient. Since this design phase is more precise and detailed than before, the decision to lower the safety factor to 2.5 was made. This is further justified since the drone is not entirely made of composite material and uses an additional safety factor on the lift force (1.5 x). Applying this factor to the obtained maximum stress gives an ultimate stress of:

$$
\begin{aligned}
\sigma_{ULT} &= \sigma_{max} \cdot 2.5 = 414 \cdot 10^6 \cdot 2.5 \\
\sigma_{ULT} &= 1035[\text{MPa}] < 1050 \\
\sigma_{ULT} &< \sigma_y
\end{aligned}
$$

therefore the ultimate compressive stress does not cause plastic deformation in the structure. This structural model is not the exact same as the real structure, as it does not include the additional ribs and joining mechanisms that are used for the propellers, actuators and elevons. Since these additions actually increase the structural stiffness and strength by offering more load paths and putting more fibers in the direction of deformation, designing the wing to be safe without these additions is conservative and increases safety.

It should be noted that the maximum deflection in such a flight condition is also computed by Ansys. These results are presented in Figure 6.7, where a maximum deflection of approximately 15 [mm] is experienced at the trailing edge of the wing tip due to twist and bending.



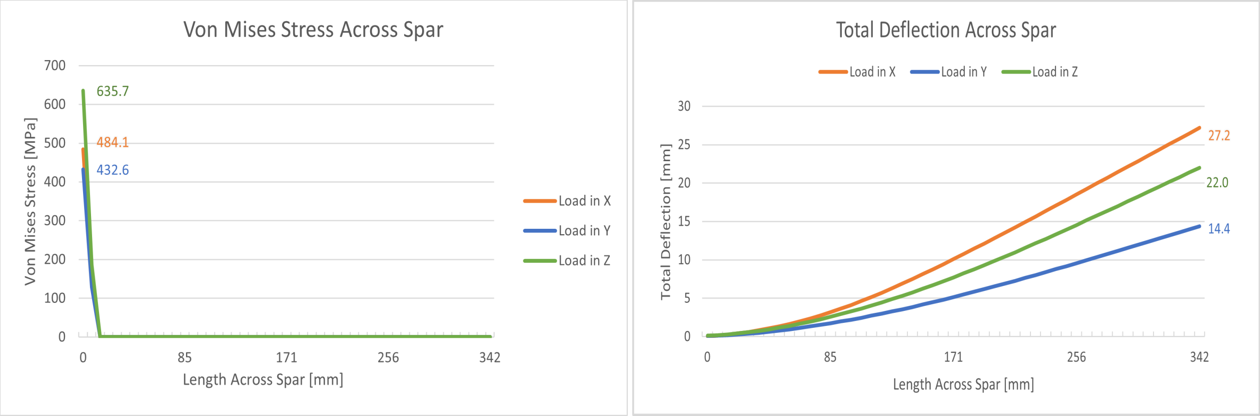
(a) Front view deflection

(b) Side view deflection

Figure 6.7: Deflection during turning flight

These deflection values fulfill the requirement of maximum deflection of 50 [mm] in 25g turning flight.

For requirement SM-OPRT-TRAN-03.1, the point load of 25 [N] will be applied during this critical flight condition at the leading edge of the wing tip to maximise any moment arms and bending stress. This requirement can also be used as a partial sensitivity analysis as applying the point loads verifies the robustness of the stress and deflection results. To holistically analyse the sensitivity of application, the direction of the force will be varied such that it is analysed in the case it is applied in the negative x axis (with drag), axially through the y axis or in the negative z axis (with lift). These applications can simulate sensitivities to increased force magnitudes. The Von Mises stress and deformation across the upper edge of the spar for the 3 different load cases is presented in Figure 6.8.



(a) Von Mises stress across spar due to 25 [N] point load in X,Y,Z axes (b) Deflection across spar due to 25 [N] point load in X,Y,Z axes

Figure 6.8: Point load stress and deflection results

When applying the load in the negative z axis, the greatest stress values at 635.7 [MPa] are present. This is logical as the load application is in support of the lift load, increasing bending stress, which the structure was sized based on. The implications are that if the aerodynamic load increases by 25 [N], and the safety factor of 2.5 is maintained, the structure will fail in compressive loading in this instance. This is not the most realistic situation however, as a safety factor has already been applied to the wing lift and having an additional point force parallel to its vector during a turn is highly improbable. The stress results for a load in the x axis achieve a maximum of 484.1 [MPa], which may also cause structural failure, but marginally so given its proximity to the yield stress. The main concern with a point load in the x axis is the deflection, making the wing tip deflect up to 27.2 [mm], as even though this is still within the requirement margin, flutter effects may take precedence given the distortion across the wing as the planform distorts. Finally for an axial load through the y axis, the stress and deflection values obtained are the smallest compared to the other axes. This is because the axial load does not bear a large moment contribution towards the root.

Therefore for the design critical load case, an additional point load of 25 [N] across the z and x axes are likely to cause structural deformation, but in the y axis, the structure does not deform. This is considered acceptable given the extremity of the situation, and thus for less design critical flight phases, an additional point load of 25 [N] is not expected to cause plastic deformation thereby fulfilling the requirement. To fully validate the requirement, additional tests would need to be performed, preferably on a physical model, which emulates different point load applications which may be more critical than the aforementioned. This analysis has also indicated through which forces the design is sensitive towards: lift force changes result in substantial stress increases with acceptable deflection increases, drag or thrust force changes result in substantial deflection increases with minimal stress increases and in general the design is insensitive to axial loads. The implies that the design is somewhat robust, but could use additional reinforcement against vertical loads. It should be noted that this may be alleviated since, as mentioned before, the actual design contains additional stiffening elements and load paths due to the ribs and ribs for the elevon.

6.3.4. Fatigue Analysis

As per the requirement SM-ENVI-FATG-01.2, the wing structure must be able to withstand 150 cycles of maximum stress during turning flight. Upon revision, this cycle number is heavily understated; for a racing drone which seeks to perform multiple turns and perform in a multitude of competitions, 150 cycles is very limiting. Instead, this requirement will be amended, such that 10000 cycles must be sustained, to allow for a greater representation of the likely repetitive stress cycling during flight. The value of maximum stress referenced follows from the Von Mises analysis, whereby the maximum stress was 414 [MPa]. To evaluate whether or not the structure can withstand 10000 cycles, the total number of cycles until failure must be estimated, which marks the premise of fatigue analysis. Ordinarily fatigue analysis would be performed by using a material's strength cycling curve (S-N curve), however this requires an in-depth tabulation of the material's strength through various tests. Such a practice is not possible for this design phase given the detail and time required. Additionally, since the load carrying structure is composed of a unique composite which is not off-the-shelf, it is not possible to find available and validated S-N curve data. Thus a more statistical approach will be used whereby typical values for a similar composite will be employed in analysis.

The change in material strength due to repeated loading is often characterised by fitting parameters. These fitting parameters are used in various prediction models. Deciding on a prediction model depends on the ratio of minimum to maximum stress (R) and the type of material under analysis. In this case, $R \approx 0$ since the loading changes from a minimum stress of 2140 [Pa] to $414 \cdot 10^6$ [MPa] and does not go negative since the wing loading does not go negative at any point due to the constant lift pressure. For composites materials, a review on the different models given experimental data and R-value indicated that for R=0.1, the Sendeckyj model equation is most suitable [15]. From this model it follows that along with the ultimate tensile strength (σ_{UTS}) and peak stress amplitude per cycle (σ_{max}) can be used to predict number of cycles until failure (N_f) given constant stress amplitude loading and assumably negligible residual stress, as defined by Equation 6.1 [34]

$$N_f = \frac{\sqrt[S]{\sigma_{UTS}}}{C(\sigma_{max} \cdot SF)^{\frac{1}{S}}} + 1 \tag{6.1}$$

where S and C are constants extracted from an external study which determined statistically determined typical model values for a composite in Sendeckyj fatigue based on a glass fiber reinforced plastic [10]. This study also highlighted the high accuracy of the Sendeckyj model for composite fatigue prediction analysis. Since this prediction is using a general statistical model, a safety factor will be applied to the σ_{max} to illustrate the uncertainty in the result. This safety factor will be 3, following from the midterm analysis [4].

The Sendeckyj model relies on a constant stress amplitude due to the use of using estimation on the build-up of residual stress [34] and for the sake of this analysis can be applicable. In reality, the drone will not experience a constant amplitude loading as it will not enter and perform each turn with a 60 m/s velocity and centripetal acceleration of 25 g, but for simplicity and structural safety, a constant load assuming such a strict turn will be used as the cyclic loading. The time per cycle will be equal to the turn time. Given that a 180°turn at minimum turn radius is performed at 60 m/s, the expected cycle time is thus 0.5 seconds approximately. Inserting the values for stress and strength along with the statistical constants found by [10], the following is calculated:

$$N_f = \frac{\sqrt[0.0932]{2100 \cdot 10^6}}{(414 \cdot 10^6 \cdot 3)^{\frac{1}{0.0932}} 0.0034} + 1$$

$$N_f \approx 82400 \text{Cycles to Failure}$$

since this is above the 10000 requirement, the requirement is met.

6.3.5. Impact Energy

In terms of impacts and collisions, the velocity at which the drone is travelling has a large effect on the impact force. The materials absorb the energy of the impact and potentially yield or fracture depending on the magnitude of energy absorbed. In terms of the CFRP impact resistance, this will come into play when the faces of the pod are impacted. It is unlikely that the pod spar and wing spar are part of this stiffness assessment given their internal location and surrounding of foam. Several studies have been conducted which look at the impact energy of composite laminates and the damage incurred. Of these studies, many conclude that the trends are unsubstantial and cannot be generalised, which is evidenced by the significantly differing values for energy absorption and impact velocities.

Composite energy absorption relies on the fiber lay-up and thus it is essential that for adequate results to be obtained, a fiber of similar lay-up is compared. This is due to the complex nature of composite failure, encompassing buckling, delamination, fiber tears, matrix cracking amongst other forms. One study analysed a set of glass fabrics of varying fiber volumes and laminate thicknesses, but with layup of $0\backslash 45\backslash 90\backslash -45$, similar to the chosen layup for CFRP. The composite with the highest fiber volume fraction (0.46) experienced major damage when loaded at 16 [J] and 24 [J] per layer [57]. Taking the smallest of these values for the mean value for energy per layer gives 16 J per layer. Given that the impact is across both of the pod surfaces, a total of 2 [mm] of CFRP is present at impact. This amounts to a total of 24 laminate layers. Applying the expected 16 J per layer gives a total of 384 [J] across the faces. Using the kinetic energy equation:

$$E_k = \frac{1}{2} m \cdot v^2 \tag{6.2}$$

This gives a total maximum impact velocity of 39.2 [m/s]. This value of velocity is not realistic however, since this assumes that the total kinetic energy of the drone is lost during collision, meaning it immediately arrives at a stand-still. Therefore this velocity can be considered as a minimum velocity whereby under max energy absorption, damage occurs. Ordinarily energy is dissipated through sound amongst other energy losses and the drone performs an elastic collision. It is difficult to predict what portion of the total kinetic energy is transformed into elastic energy and thus no further analysis can be carried out. For comprehension of how significant energy losses can be, car collisions can lose up to 75% of the initial system energy during a collision [13]. Additionally, due to having access to all 3 planes, drones have the ability to maintain some kinetic energy after collisions by veering off after impact. It should be noted that using a value of 24 [J] per layer results in a minimum velocity increment of 48 [m/s].

EPP in the instance of a crash is expected to aid in reducing the incident acceleration. It is well established it cannot carry much load, but it's elastic properties allow it to 'sponge' the impact, which increases with high strain rates [40]. Using the ARPRO EPP foam as mentioned in Subsection 6.3.2, the cushioning performance graphs provided by the manufacturer indicate that it is common to see the EPP decrease acceleration by 25 g's with the smallest thickness tested being 1" [10]. Since this thickness is several orders larger than the thickness of the foams in the drone, this effect was neglected during this analysis. However, having deceleration capabilities will aid in reducing the kinetic energy of the structure throughout the impact, which will allow for a greater velocity increment to be achieved in the carbon fiber prior to major damage. These two impact assessments contribute to the requirement of SM-OPRT-FAIL-02.1 and SM-OPRT-FAIL-02.2. Since the energy loss and deceleration effects cannot be properly quantified, it is not possible to properly validate these requirements.

6.3.6. Structural Integration

As the structure is a key hardware component in the drone, the Structures & Materials department is responsible for the hardware integration of all subsystems. The process of integration each of the different subsystems is described in more detail in Subsection 12.2.1. Here, the main integration steps affecting the structure are listed.

- Battery integration involved was vital, as the battery is a large percentage of the total drone mass. For this reason, to ensure a stable drone the placement of this component is vital. To ensure the load exerted by the battery is properly introduced into the structure, a small shelf of CFRP was designed to attach the battery directly to the frame.
- Actuator integration involves placing the actuators inside the skin, such that the do not protrude. These components should also be rigidly mounted to either the spar or rib, to ensure the actuator does not move whilst actuating.
- The remaining electronics are placed in the pod compactly. The FC and ESC are designed to be stacked, and will also be in the Peregrine drone. since the skin is made from foam, the RX-antenna is able to be placed inside the pod. The VTX-antenna does not fit and will be mounted trough the skin on the top surface of the drone.
- The tilting mechanism is designed as a four-bar linkage. This allows for a small mechanical advantages to be gained from the actuator deflection to the motor deflection. This is needed, as the stall-torque of the actuator alone is not sufficient to counteract the moment generated by motor when accelerating into a 25g turn.

6.3.7. Mass Budget

Since the weight of the drone is critical, it is important to confirm that the structural weight is within the specified boundaries. Given all the dimensions of the structure as analysed, we can use the density of the used material to compute the structural mass. The result of this analysis is shown in Table 6.4. Here all the parts contributing to the structure and their respective mass and material are listed, including the mechanism. The final mass is about 1.8 grams lighter than budgeted, and the mechanism mass is about 3.5 grams lighter.

6.3.8. Cost Analysis

Given the structure shown in Figure 6.2, it is possible to use the weight of the constituent materials and the analysis conducted in Chapter 2 to derive a cost estimation.

For the use of CFRP, the manufacturing and production costs for a batch size of 50 units of a wing box was

---

[10] `https://arplankdirect.com/technical-data/material_specifications` Accessed 15th June 2021

Table 6.4: Structural Mass Breakdown

Part	Material	Count	Mass per Part [g]	Mass [g]	Budgeted [g]
Frame	CFRP	1	*45*	45	
Inboard Plates	CFRP	2	*1*	1	
Control Rods	CFRP	2	*3*	6	
Outboard Hinge	CFRP	2	*1*	2	
Outboard Motor Mount	CFRP	2	*3*	6	
Servo Horn	CFRP	2	*1*	1	
Linkage Arm	CFRP	2	*1*	1	
Subtotal				***63***	-
Skin	EPP	2	*13*	26	
Elevon	EPP	2	*4*	8	
Subtotal				***34***	-
Total				**97**	**102.2**
Of which Structure				*89*	*90.3*
Of which Mechanism				*8*	*11.9*

890 [€/kg], excluding assembly costs as per 2016 [31]. Adjusting for inflation to FY2021 gives 935 [€/kg] [11]. Even tough this batch size is significantly different from the projected production size of Peregrine, it will suffice for the preliminary cost estimation required at this stage of the design. This value may also not be accurate for the production of the drone due to the drone using very thin surfaces which may cause deviations in cost. It is well known that for a reduced batch size, the price of production per batch increases [30], where a production cost/kg for a ton of PAN based CFRP is 20 times larger than the production cost for 35 tons. Unfortunately the weight per unit for the article was not provided and thus it is not possible to draw a definitive deviation between the expected costs. Using the mass of CFRP to be 63 [g], the corresponding cost per drone amounts to be €69 for the CFRP structure.

EPP foam costs rely on the raw material being purchased and processed using molds. Recently the use of EPP beads as raw material has allowed for reduced storage and transportation costs, though they are slightly more expensive as a resource. The beads are generated from the expansion of polypropylene pellets, which are then fed through an extruder into the mold and then autoclaved to output a final structure. The cost for raw materials is 4 USD/kg and the production cost 2.5 USD/kg as per a production of 1000 tons per year in 2017. Once again this cost will increase given a smaller batch size, but the quantification of this effect is difficult. To get an idea of the cost, a cost of half of the CFRP cost will be used, since they use similar manufacturing machines (autoclave), but the EPP is cheaper a material. [12]

Given an EPP foam mass of 34 [g] this results in a cost of €16. An additional cost of €50 will be employed for error and assembly costs and adhesive bonding cost. This results in a **estimated final cost of €135**, meaning the requirement of €300 is met.

6.4. Risk Assessment

Structural risks relate to the failure or yielding of the materials. These can be summarised into a list with specific identifiers and an explanation of the risk. Below each risk, a statement on the possibility of future mitigation or reasoning of acceptance is mentioned.

- SM-R1: The wing planform permanently deforms during flight, inhibiting aerodynamic performance substantially.
  - This can occur if the spar loading is underestimated and the CFRP plastically deforms. Such an occurrence can be reduced in likelihood by the magnitude of the safety factor used. Further reduction are possible but require an increased structural weight by adding additional material in the wing, which would negatively impact the budget.
- SM-R2: The foam bonding to the structure shears, stripping the structure.
  - To reduce the likelihood of the risk, the attachment strategy was reformed to increase the at-

---

[11] URL:`https://www.inflationtool.com/euro` Accessed 13th June 2021

[12] `https://www.imgplastec.com/products/epp/` Accessed on 15th June 2021

tachment region area and create one body between the two foam regions, as originally the spar separated the body. A visualisation of this is presented in Figure 6.9. The above analysis used this change.

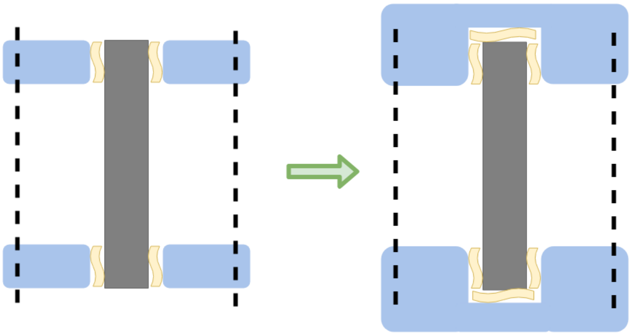


Figure 6.9: Foam-spar adhesive bonding

- SM-R3: The elevons shear out from the attaching rod.
  - Given the low likelihood of this occurrence due to the large surface through which the foam is bonded by an adhesive, this risk is accepted.
- SM-R4: The structure does not protect the inside electronics adequately during flight (through collisions or water damage).
  - To limit the severity of this, the electronics have been placed such that the battery is separated from the rest of the electronics via the pod spar. This will allow for a contingency against battery leakage or thermal effects between the two bodies.
- SM-R5: The winglets deform or break at high speeds.
  - To ameliorate this, the structure would have to be changed to include a stiffer, stronger material, which would likely increase the mass and structural revision. Instead, this risk will be accepted and to improve customer satisfaction in occurrence, the structure can be manufactured such that the winglets are detachable to allow for easy replacement. This is because the winglets service more as a standing platform for the drone, and their absence will not directly inhibit structural performance.
- The wing receives a sharp puncture, tearing part of the EPP.
  - In such an event, the structural integrity of the structure will seemingly be unaffected since the EPP does not carry much of the load. The main impact of this is to the aerodynamic loading which is not design critical for the stress carrying structure since the principle stress is dependent on high bending moments due to the lift. For this reason, the risk is accepted. A future mitigation can be done by implementing a mesh through the EPP, which would limit the amount of torn EPP.
- SM-R6: The impact loads are higher than expected.
  - Higher impact loads would result in structural damage and greater electronic collateral which is undesirable. An attempt was made to minimise the impact of this risk by using an EPP structure to decelerate and absorb the impact. Reducing this effect even further can be done by increasing the EPP layer thickness and CFRP thickness, but this may require a revision of the budget due to increased mass.

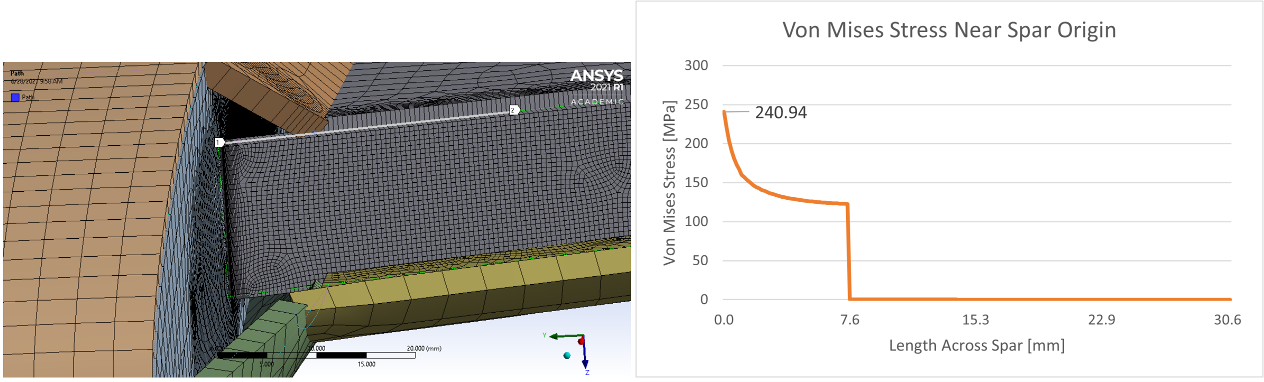
6.5. Verification & Validation

To ensure the results above are reliable, they must be verified to check for consistency and/ or validated to check for accuracy. This is evaluated on the stress analysis provided by Ansys as well as the fatigue and impact calculations.

6.5.1. Stress Calculation

To verify the Ansys software, an analytical solution will be used to determine Von Mises stress and compared to those found in the software. As seen in Figure 6.3, the maximum stress found in the critical load condition was 414 [MPa], which was experienced at the integration between the body spar and wing spar. Additionally, it was noted that assuming all the stress to be carried by the spars alone was a valid assumption. At the wing spar specifically, due to the swept wing, there is a stress concentration directly at the beginning of the joint with the pod surface, resulting in a rapid rise of stresses that are difficult to analyse analytically. Therefore, to verify the results without this concentration's effect, the Von Mises stress across only the spar near the wing root, with an offset of 0.7 [mm] in the y axis from the actual root will be analytically compared. This was also done due to student software license limitations that prevented a large magnitude of points being measured across a user-defined region.

An Ansys plot of the Von Mises stress of a 30 [mm] region from this offset is shown in Figure 6.10b. The exact path of the stress analysis is shown in Figure 6.10a.



(a) Spar stress path

(b) Von Mises stress near spar root

Figure 6.10: Reference results for verification

Following from the graph, the maximum stress experienced, 240 [MPa], is at the first node position, meaning it is as close as possible to the root of the spar. One thing to note is the rapid rise in stress at about 7.6 [mm]. This distance actually corresponds to the separation of the foam from the spar structure as indicated in Figure 6.10a, implying that the spar attachment to the foam skin helps in decreasing the stresses as loads can be transferred from one material to another. When in this 7.6 [mm] space, the spar cannot transfer the load through foam to the adjoining ribs until it directly meets. In the actual design this is not the case as the spar and skin are constantly attached and the spar height increases to match the pod thickness meaning the stress would not rise so rapidly, but the Ansys software could not easily emulate this with the student licensing. Therefore the analytical calculation will aim to obtain a similar Von Mises stress at the root of the spar using a spar only analysis. To do so, the variables within the Von Mises (Equation 6.3) equation must be calculated [43]. This includes the normal stress (σ) in the three axes as well as the shear stresses (τ) in the three planes.

$$\sigma_{VMS} = \sqrt{\left(\frac{(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2(\sigma_z - \sigma_x)^2}{2}\right) + 3\left(\tau_{xz}^2 + \tau_{xy}^2 + \tau_{yz}^2\right)} \tag{6.3}$$

The normal stresses include the bending stress and axial stress due to the moment and forces experienced at the root, giving the following Equation 6.4, for the y axis for example:

$$\sigma_y = \frac{M_x \cdot z}{I_{xx}} - \frac{M_z \cdot x}{I_{zz}} + \frac{F_y}{A} \tag{6.4}$$

where M is the moment contribution from the different axis, z or x is the point about which the stress is analysed, I is the second moment of area about the specified axis, F is the axial force and A is the cross sectional area about the specified axis. The shear stress contribution relies on the torsional shear due to a moment and transverse shear due to a set of forces. Torsional shear is computed with Equation 6.5[43]:

$$\tau_{Torsion} = \frac{3T}{ab^2} \tag{6.5}$$

where T is the torsional moment, a is the longer side length of the section and b is the shorter side length. Note this equation uses a thin walled approximation, which is valid for the yz and yx frame, but not for the xz plane (given the spar thickness is 3 [mm] with a height of 16 [mm] at the root). For non-thin walled structures, the following Equation 6.6[13] is valid for rectangular sections:

$$\tau_{Torsion} = \frac{3T}{8ab^2}\left(1 + 0.6095\frac{b}{a} + 0.8865\left(\frac{b}{a}\right)^2 - 1.8023\left(\frac{b}{a}\right)^3 + 0.91\left(\frac{b}{a}\right)^4\right) \tag{6.6}$$

---

[13] `https://roymech.org/Useful_Tables/Torsion.html` Accessed 15th June 2021

where a and b are *half* the lengths of long and short side respectively. Shear stress due to a force is dependent on the point being analysed and its distance from the forces. However, for a rectangular cross section, the maximum shear stress due to a force can be computed by Equation 6.7

$$\tau_{force} = \frac{3F}{2A} \tag{6.7}$$

where F is the transverse force(s). Adding these two contributions for each plane then gives the total shear stress. Carrying out the calculations using the above equations at the spar root gives the following Table 6.5:

Table 6.5: Analytically calculated forces, moments and stresses at spar root due to turning flight load

Parameter	Value	Unit	Parameter	Value	Unit
F_x	-0.07	[N]	σ_x	-0.39	[MPa]
F_y	-1.26	[N]	σ_y	115.04	[MPa]
F_z	91.95	[N]	σ_z	11.53	[MPa]
M_x	-14.94	[Nm]	τ_{xz}	131.42	[MPa]
M_y	5.44	[Nm]	τ_{yz}	0.57	[MPa]
M_z	0.04	[Nm]	τ_{xy}	0.05	[MPa]

and a resultant Von Mises stress of 252.8 [MPa]. Comparing this stress to the Ansys software gives a percent difference of 4.9%, which is considerably close. The discrepancy can be due to the fact that the analytical calculations simplified the structure into a single spar which had a sweep equal to the leading edge sweep, whereas the wing spar has a slightly lower sweep, which would decrease moments. The loss of the surrounding EPP also causes a larger stress to be experienced, as in reality it does carry some stress, which would lessen the total stress experienced by the spar. Nonetheless the low percent difference is evidence that the Ansys software is verified by the analytical calculations.

To validate the Ansys software, it would require obtaining experimental data for the same or similar models. Unfortunately, due to the novelty that this model attributes, similar concepts are not readily available. Software validation itself can be done by reviewing previous studies which use the Ansys software and compared the results to experimental data. Such studies, for example a study on the use of FEA for composite wing deflection [38], indicate that for wing tip deflection specifically of composite UAV drones, FEA can result in accuracies of 0.3%. For root deflections which are substantially smaller however, the accuracy can be up to a third of the experimental data. Given that deflection in the order of 10-15 millimeters had a mean accuracy of about 8% this is evidence for validation of the Ansys' deflection results. Another study verified and validated the stress analysis that Ansys provides, concluding that Ansys is ”reliable” and ”true” as both analytical and numerical results approach one another [2]. Although these articles along with others can offer indirect validation of the software, the model employed to calculate the stresses cannot be properly validated until the model itself is prototyped and tested under cruise and turning loading scenarios, which would be the next stage of design. This would be done using a wind tunnel to accurately simulate the drag and lift forces. Simple bending could be done with weight loading however. In both instances, strains sensors placed along the root and towards the intersection with the fuselage to measure the deflection could be used along with the pressure distribution found through pressure taps which would allow for a lift and drag estimation. With this data and force sensors, the stress can be found at the root and then compared with the Ansys stress.

6.5.2. Fatigue and Impact Estimation

For fatigue analysis, verification could not be performed well since the failure prediction models differ quite drastically from one another and not all are applicable for composites. Therefore verification was more dependent on seeing the effect of the safety factor on results. Validation can be done by comparing the number of cycles with a somewhat similar composite of which the data was sourced. If the safety factor is put to its minimum value of 2 for example, the cycles to failure instead drastically increases to approximately $6.4 \cdot 10^6$ cycles. This value seems excessively large, however this may be due to the fact that residual stress was ignored for simplicity and is based on statistical constants, meaning changes in safety factor carry these effects and cannot be understated. Comparing to the reference studies [34] which used a composite material of comparative strength 2180 [MPa] and Young's modulus of 135 [GPa], common failure cycle order of magnitudes were 10^5, however this was for a peak stress of 1800 [MPa] per cycle along with residual stress. Using the safety factor of 3, the peak stress in our model reaches 1242 [MPa] which makes this model closer to the one being compared (compared to using a factor of 2). Therefore it is expected that our model, bearing similar stress, strength and stiffness values to that in the study, would have a similar order of magnitude in its results, thereby acting as a partial validation for the fatigue calculations. Further validation would and should be done by testing the material via weight drop tests or repeated bending loads to obtain the correct constants. Nevertheless the exceedingly large order of magnitude can be used to indicate that the requirement is met.

For impact calculations, further validation is required either through a more in-depth collision analysis using computer software to visualise the effect of foam onto the CFRP structure or through the preferred physical testing, whereby the drone would fly into a pole or wall at a designated speed. Through testing, the direct energy losses under differing impact situations could be quantified and used to validate the corresponding requirements in each situation. Due to the arbitrariness of collisions and impacts, it is not possible to suitably validate them at this level of design. Additionally, it is surmised that the 3 [mm] of foam used in the design will not be sufficient and the option of using solid foam must be argued for. This idea was tested in this stage of design, but unfortunately could not be implemented due to it requiring another 12 [g] of structural mass, which would push the total drone mass over budget (including contingencies). Ansys would also not allow for the stress analysis of a solid and surface based structure using the student licensing, meaning comparative results could not be obtained. Solid foam would be the biggest point of interest for a later iteration if changes in mass budget are acceptable for the greatly increased safety and performance.

In conclusion, structural validation would greatly benefit from physical prototyping to ensure that the software and statistical estimations are accurate. Given the current circumstances and time constraints physical prototyping was not a possibility, however if this design process were to be further expanded upon, prototyping and stress tests would be the next stage. Once adequate understanding of the physical results is obtained, the models can be updated and developed such that iterations and optimisation can occur efficiently and sustainably.

6.6. Compliance Matrix

The structural analysis performed allows for the checking of which requirements have been met, conveyed in the compliance matrix of Table 6.6.

Table 6.6: Compliance matrix for structures subsystem.

Identifier	Status	Reference	Identifier	Status	Reference
SM-FEAS-COST-01.1	Pass	Subsection 6.3.8	SM-ENVI-FATG-01.2	Pass	Subsection 6.3.4
SM-FEAS-SUST-01.1	Pass	Chapter 11	SM-ENVI-FATG-01.1	TBC	
SM-PHYS-WGHT-01.1	Pass	Subsection 6.3.7	SM-ENVI-WTHR-01.1	Pass	Subsection 6.3.2
SM-PHYS-SIZE-01.1	Pass	Subsection 6.3.1	SM-OPRT-FAIL-02.2	TBC	
SM-PHYS-SPMT-01.1	Pass	Subsection 6.3.2	SM-OPRT-FAIL-02.1	TBC	
SM-PHYS-SPMT-02.1	Pass	Subsection 6.3.3	SM-OPRT-MAIN-02.2	Pass	Subsection 6.3.6
SM-PHYS-SPMT-02.2	Pass	Subsection 6.3.3	SM-OPRT-MAIN-02.1	TBC	
SM-PHYS-SPMT-03.1	TBC		SM-OPRT-TRAN-03.1	Pass	Subsection 6.3.3
SM-OPRT-TRAN-01.1	TBC		SM-OPRT-TRAN-02.1	Pass	Subsection 6.3.1
SM-OPRT-TRAN-01.2	TBC				

Requirements **SM-PHYS-SPMT-03.1** and **SM-ENVI-FATG-01.1** are both propeller related. Whilst the propeller has been somewhat detailed in Chapter 7, the structural loading on it has not been analysed properly. To define the stiffness and fatigue cycling, it would have to be analysed either using a computer software like Ansys once again, but preferably with physical testing using a prototype to understand potential failure mechanisms and loads. This would be done in the next phase of design.

Requirements **SM-OPRT-TRAN-01.1**, **SM-OPRT-TRAN-01.2** and **SM-OPRT-MAIN-02.1** are not possible to validate at this stage since they deal with assembly of the drone, and the replacement of the components. These requirements would only be validated with a demonstration with a prototyped drone to time the procedures properly.

As mentioned in Subsection 6.3.5, the requirements relating to impact energy and speed, **SM-OPRT-FAIL-02.1** and **SM-OPRT-FAIL-02.2**, cannot be fully validated unless there is a detailed impact analysis using software or once again through physical testing with a prototype or material samples. This would also be the next stage of design.

7

Power & Propulsion

This chapter deals with the analysis of the Power & Propulsion subsystem of Peregrine and verification of its requirements. First the functions of the system are recapped, then the requirements which need to be fulfilled are listed. Subsequently, a set of airfoils is analysed and a suitable airfoil is chosen for the propeller design. Then the propeller design procedure is outlined, followed by the loads on the servos and motors. Following the motors, the battery mass estimation is carried out. Using the outcome of the battery sizing a propeller design is chosen. The feasibility of using regenerative braking is briefly discussed. Then the risks related to the Power & Propulsion subsystem are outlined, followed by the verification and validation of the design. Finally, the compliance matrix is shown.

7.1. Functional Recap

The function of the Power & Propulsion subsystem is to provide power to the electrical components, provide thrust, additional braking and to assist in low speed manoeuvres. To do so propellers will be designed, the loads on the motors and actuators will be evaluated and the battery mass will be determined using the chosen race track.

The Power & Propulsion subsystem receives the system system mass from the structures subsystem and the drag from the aerodynamics subsystem to come up with a set of propellers, motors, actuators and battery. The propeller performance is then given to the control and stability group and the performance and simulation group to integrate into the control loop and simulation. The new component masses are then also given to structures to come up with a new system mass.

7.2. Requirements

During a previous design phase the requirements for Peregrine were set in stone. The requirements which are applicable to the Power & Propulsion subsystem can be found in Table 7.1. These requirements will be verified throughout the chapter and the results will be displayed in the compliance matrix.

Table 7.1: Power & Propulsion requirements

Identifier	Requirement	Method
PP-FEAS-COST-01.1	The battery shall not cost more than €75.	Review of Design
PP-FEAS-COST-01.2	The propulsion system shall not cost more than €218.	Review of Design
PP-FEAS-COST-03.1	The maximum energy use per hour shall not exceed 1 [kWh].	Analysis
PP-FEAS-SUST-01.1	The battery shall be at least 75% recyclable by weight.	Similarity
PP-FEAS-SUST-01.2	The propulsion system shall be at least 70% recyclable by mass.	Similarity
PP-FEAS-SUST-03.1	The propulsion system of the drone shall be electric.	Review of Design
PP-PHYS-WGHT-01.1	The mass of the battery shall be at most 170[g].	Review of Design
PP-PHYS-WGHT-01.2	The total mass of the propulsion system, excluding the battery, shall not exceed 178[g].	Analysis
PP-PHYS-SIZE-01.1	The battery shall not exceed a volume of 216[cm^3].	Review of Design
PP-PHYS-SIZE-01.2	The battery shall fit inside the body of the drone.	Review of Design
PP-FLGT-HOVE-01.1	The drone shall provide a maximum vertical thrust of at least 17 [N] in hover condition.	Analysis
PP-FLGT-HOVE-05.1	The propulsion subsystem shall ensure stability of the drone in hover flight.	Analysis
PP-FLGT-MOVE-03.1	The drone shall have a maximum forward static thrust of at least 17 [N] in forward flight at 60 [m/s].	Analysis
PP-FLGT-MOVE-04.1	The propulsion system shall allow the drone to decelerate from maximum to minimum velocity in 8 [s].	Analysis

Identifier	**Requirement**	**Method**
PP-FLGT-ROTA-03.1	The propulsion subsystem shall make it possible to obtain a pitch rate of at least 1000 [deg/s] from nominal forward flight.	Analysis
PP-FLGT-ROTA-04.1	The propulsion subsystem shall make it possible to achieve max pitch rate from nominal forward flight within 1 [s].	Analysis
PP-FLGT-ROTA-05.1	The propulsion subsystem shall make it possible to obtain a yaw rate of at least 300 [deg/s] from nominal forward flight.	Analysis
PP-FLGT-ROTA-06.1	The propulsion subsystem shall make it possible to achieve max yaw rate from nominal forward flight within 5 [s].	Analysis
PP-FLGT-ROTA-07.1	The propulsion subsystem shall make it possible to obtain a roll rate of at least 1080 [deg/s] from nominal forward flight.	Analysis
PP-FLGT-ROTA-08.1	The propulsion subsystem shall make it possible to achieve max roll rate from nominal forward flight within 1 [s].	Analysis
PP-FLGT-ROTA-09.1	The propulsion subsystem shall make it possible to obtain a pitch rate of at least 180 [deg/s] from nominal hover flight.	Analysis
PP-FLGT-ROTA-10.1	The propulsion subsystem shall make it possible to achieve max pitch rate from nominal hover flight within 1 [s].	Analysis
PP-FLGT-ROTA-11.1	The propulsion subsystem shall make it possible to obtain a yaw rate of at least 180[deg/s] from nominal hover flight.	Analysis
PP-FLGT-ROTA-12.1	The propulsion subsystem shall make it possible to achieve max yaw rate from nominal hover flight within 1[s].	Analysis
PP-FLGT-ROTA-13.1	The propulsion subsystem shall make it possible to obtain a roll rate of at least 90 [deg/s] from nominal hover flight.	Analysis
PP-FLGT-ROTA-14.1	The propulsion subsystem shall make it possible to achieve max roll rate from nominal hover flight within 1 [s].	Analysis
PP-FLGT-ENDR-02.1	The battery shall have a capacity of minimum 0.8 [Ah].	Review of Design
PP-PLTG-SIGN-02.1	The battery shall provide 0.5 [W] of power for the downlink.	Review of Design
PP-OPRT-MAIN-02.1	One propeller shall be able to be replaced in less than 15 [s].	Similarity
PP-OPRT-MAIN-02.2	The electric components on board shall be detachable from all other components excluding cables.	Review of Design
PP-OPRT-PREP-03.1	The power system shall have a dedicated on/off switch.	Review of Design
PP-OPRT-PREP-03.2	The power systems shall supply power to all components within 1[s] when turned on.	Similarity
PP-OPRT-RLBL-01.1	The propulsion system shall have a failure rate of less than 0.62E-3 per flight hour.	Analysis
PP-OPRT-FAIL-01.1	The propulsion system shall be able to stop the motors in case of a collision with the defined impact.	Review of Design
PP-STRD-CSTM-01.1	The battery shall be connected with a standard XT60 connector.	Review of Design
PP-ENVI-WTHR-02.1	The propulsion subsystem shall allow operation during maximum crosswinds in any flight condition.	Analysis
PP-ENVI-WTHR-03.1	The electrical system shall be able to withstand at least the defined temperature range.	Similarity
PP-ENVI-WTHR-03.2	The battery shall be able to withstand at least the defined temperature range.	Review of Design
PP-ENVI-ALTD-01.1	The propulsion system shall enable operation up to atmospheric conditions at the defined maximum altitude.	Analysis

7.3. Analysis

As the propellers require substantial energy from the battery, the geometry and airfoil used have to be optimised. Momentum blade element theory was employed to optimise the propeller geometry. This method is applicable to the design of this drone as this theory was developed primarily for low disc loading (small thrust or power per unit disc area) [1].

To determine the thrust required at different conditions, values for drag were taken from the aerodynamics subsystem as iterations were continuously being performed on the aerodynamic profile of Peregrine.

The aforementioned momentum blade element theory required aerodynamic coefficients of the airfoil that is going to be integrated in the propeller. Hence it was vital to select an optimal airfoil to limit power consumption as much as possible. The results of the airfoil analysis is in Subsection 7.3.1. Implementation of momentum blade element theory is discussed in Subsection 7.3.2 and Subsection 7.3.3. Finally after checking compatibility of the motors and the actuators in Subsection 7.3.4, verification and Validation of the analysis performed is discussed in Section 7.5.

7.3.1. Airfoil Analysis

The momentum blade element theory discussed in Subsection 7.3.2 requires C_l and C_d of the airfoil as an input hence it was important to adopt the best airfoil for an optimum propeller to integrate within the design of Peregrine. Different shapes of airfoils serve different purposes so first an investigation had to be performed to determine the airfoil characteristics needed for an optimum performance at low Reynolds number. The Reynolds number is low despite velocity being high due to the small chord lengths of the propeller, see Equation 7.1. It was determined that the performance of low Reynolds number propellers can be greatly enhanced through reduction in the airfoil profile drag by reducing thickness, camber or both while ensuring that lift generation capabilities is not compromised [63].

$$R = \frac{\rho_\infty V_\infty c}{\mu} \tag{7.1}$$

A total of 32 airfoil coordinates were selected for an initial analysis, most of these had a low thickness and/or camber but it was kept in mind that the airfoil selected is not *very* thin as that would make the propeller structurally fragile. The program chosen for analysis is X-foil, which is an interactive program for analysis of subsonic airfoils in inviscid and viscous flows and is applicable to low Reynolds number analysis also [27]. Viscous mode was run with different speeds and Reynolds numbers and panelling of each airfoils was done before running it. The step size between each angle of attack was set to 0.1 for accuracy.

The different airfoils were run in batches and analysed for lift to drag ratios over a range of angle of attacks from which the six best performing airfoils were chosen and explored further namely: ARA-D 6%, ARA-D 10%, NACA 1408, DU86-08418 8.44%, S2048 and S2062 8%[1]. The coordinates were updated from the TU Delft airfoil repository[2] for as to increase the nodes for better results. The analysis on these airfoils was performed at the design conditions of 60 and 90 [m/s]. As Reynolds number varies along the radius of the propeller due to chords and velocities being different, the maximum was taken to ensure reliability of X-foil results. The chords and velocities were an output of the momentum blade element theory implemented which is discussed in the following section.

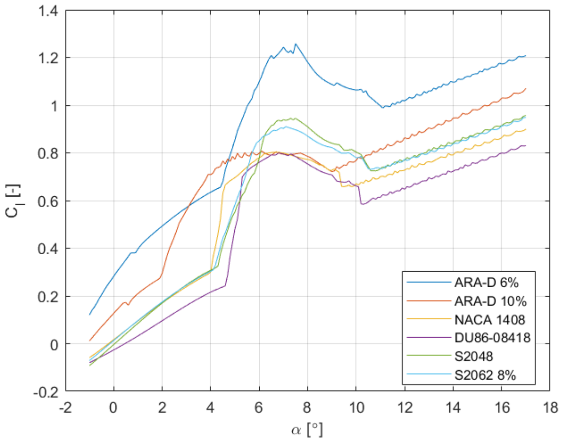


Figure 7.1: C_l vs. α at a Re of 27500.

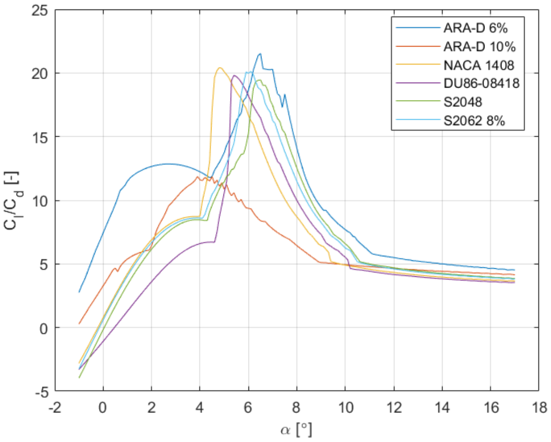


Figure 7.2: $\frac{C_l}{C_d}$ vs. α at a Re of 27500.

Figure 7.1 and Figure 7.2 shows the lift and lift to drag performance at the design condition of 90 [m/s] analysed at a maximum Reynolds number of 27500 corresponding to a radius of 25.75 [mm] from the hub of the propeller resulting from a total velocity of 122.3 [m/s] over the propeller and chord of 3.29[mm][3] The total velocity, W, experienced by the propeller at a certain radius is determined by the axial and rotational interference factors *and* the blade angle discussed in Subsection 7.3.2. It can be observed that over the

---

[1] The percentages correspond to the maximum thicknesses of the airfoils.

[2] `https://aerodynamics.lr.tudelft.nl/cgi-bin/afCDb` Accessed: 28/06/2021

[3] The total radius of the propeller was determined to be 64[mm].

different range of angle of attack that the propeller will experience, ARA-D 6% performs the best. The $\frac{C_l}{C_d}$ ratio starts decreasing between 5° and 7° due to the high lift induced drag.

After iterations and evaluation of drone's performance around the track selected, the design condition was set to 60[m/s] as the drone will spend most of it's time maintaining 60[m/s] or around that speed. Additionally the length of the chord is really low which is detrimental to the structural integrity of the propellers.

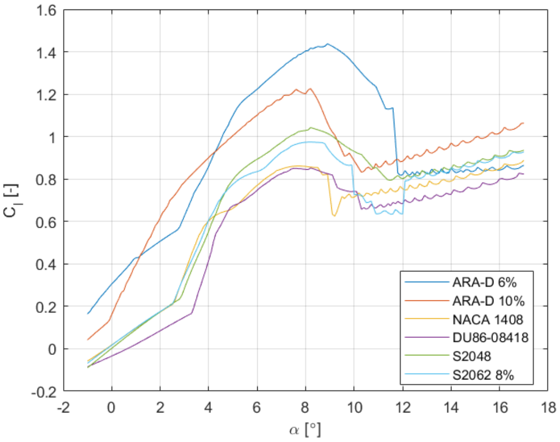


Figure 7.3: C_l vs. α at a Re of 46700.

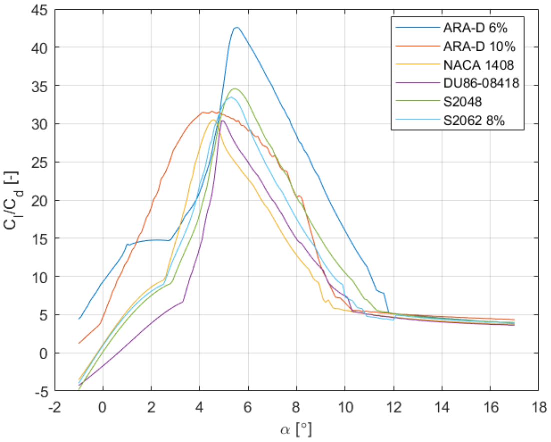


Figure 7.4: $\frac{C_l}{C_d}$ vs. α at a Re of 46700.

Figure 7.3 and Figure 7.4 shows the lift and lift to drag performance at the design condition of 60 [m/s] analysed at a maximum Reynolds number of 46700 at 12.11 [mm] radius from the hub of the propeller resulting from a total velocity of 90.9 [m/s] over the propeller and a chord of 7.5 [mm]. It can be seen that ARA-D 10% starts performing better coming *close* to the performance of ARA-D 6%. The $\frac{C_l}{C_d}$ ratio starts decreasing between 4° and 6° due to the high lift induced drag. As the propeller also experiences higher angles of attack up to 40° during operation, Xfoil was run again with a sequence from 10° to 25°. This was discovered during the design of the propeller (implementation of optimum propeller design procedure by Adkins and Liebeck discussed in Subsection 7.3.2).

From both Figure 7.1 and Figure 7.3, the increase in lift-curve slopes after approximately 3°-5° is due to the change in the flow experience near the leading edge(nose of the airfoil).

It can be observed again from Figure 7.5 that ARA-D 6% is most optimal and from Figure 7.6 two airfoils can be seen performing the best namely ARA-D 6% and S2048 (maximum thickness 8.6%). S2048 start performing slightly better but only at much higher angles of attack for which the results cannot be validated.

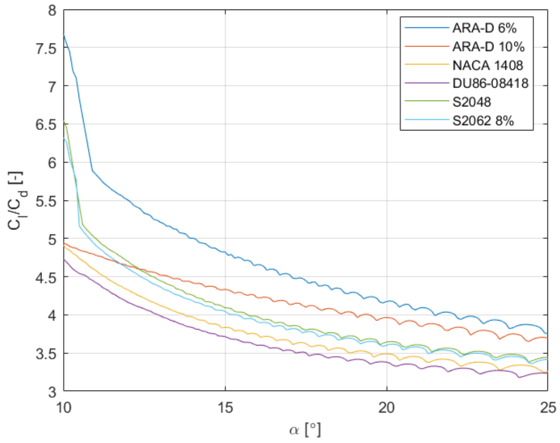


Figure 7.5: $\frac{C_l}{C_d}$ vs. α at a Re of 46700.

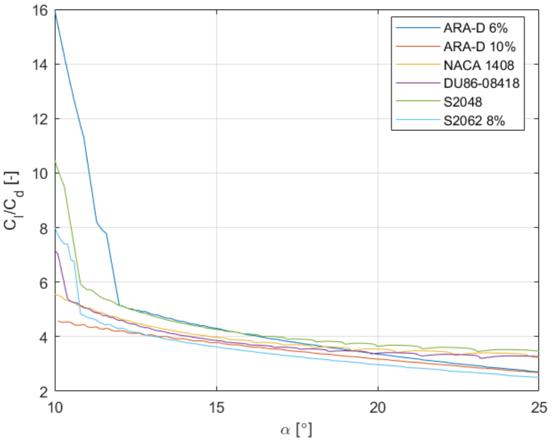


Figure 7.6: $\frac{C_l}{C_d}$ vs. α at a Re of 46700.

At high angles of attack with a low Reynolds number X-foil results do not converge, hence they **cannot** be relied upon. X-foil fails to converge when dealing with separated flow as it was designed to examine in detail the transitions from a laminar to turbulent flow and should not be used for a stalled airfoil[4]. After overall analysis of the performance of airfoils it was deduced that the airfoil ARA-D 6% is the most optimal for the propellers. It can also be realised from Figure 7.1 and Figure 7.3 that with increasing Reynolds number ARA-D 10% starts performing better hence verifying that the low thickness propellers are more desirable at low Reynolds number.

Airfoil Integration
As mentioned beforehand the propellers will experience a wide range of angles of attack during flight and it was concluded that X-foils results cannot be relied upon for all higher angles of attack. It was later discovered towards the final phase of an optimal propeller design it was necessary to change the approach slightly as getting C_T and C_P required airfoil data for higher angles of attack as well. First the airfoil data was being extrapolated to higher α but during verification phase it was realised that the high order interpolation was not enough to get accurate results. Hence experimental data was found where multiple airfoils were tested from 0° to 360°. The data for an airfoil SG6043 up to an α of 45° at a Reynolds number of 64000 was extracted and taken as an input to the propeller design scheme [70]. This airfoil was compared with ARA-D 6% at the same Reynolds number of 64000 from 0° to 15° to confirm if ARA-D 6% performs better or not as this airfoil was not analysed.

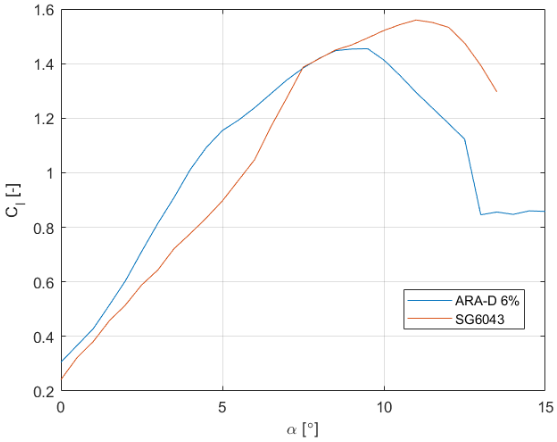


Figure 7.7: C_l vs. α at a Re of 64000.

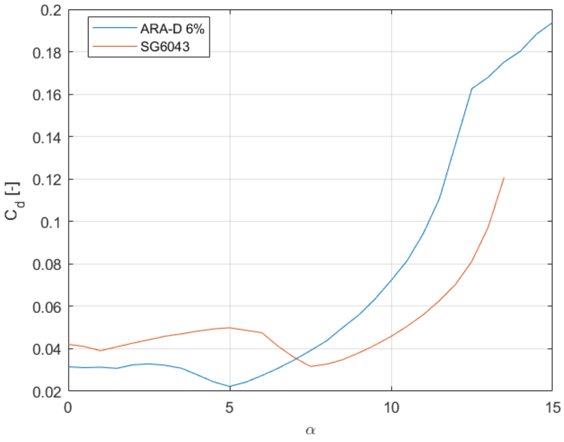


Figure 7.8: C_d vs. α at a Re of 64000.

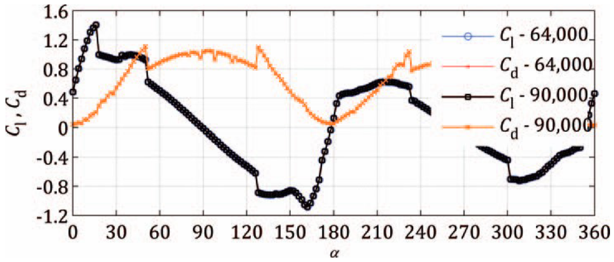


Figure 7.9: Lift and drag coefficients for SG6043 airfoil from wind tunnel experimental set-up [70].

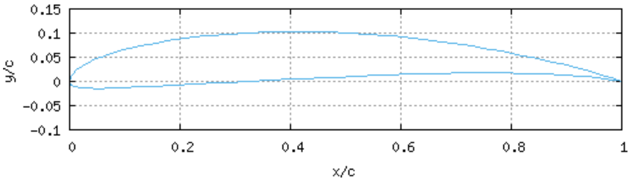


Figure 7.10: Geometrical design of SG6043 airfoil[5].

From Figure 7.7 and Figure 7.8 it can be seen that ARA-D 6% does perform better at low angles. But as experimental data on SG6043 was used for propeller selection, has a better performance at angles of attack from 7° and also has a 4% higher maximum thickness making it more structurally viable. This will also increase the reliability of the results and would not require validation performed as the data is from a wind tunnel test. Therefore airfoil **SG6043** will be integrated in the propeller design. The performance from 0° to 360° can be seen in Figure 7.9 while the geometry of this airfoil is shown in Figure 7.10.

7.3.2. Propeller Geometry

To design the propeller, the design procedure as outlined by Adkins and Liebeck [1] is followed. This design procedure uses a combination of momentum and blade element theory. The first step in the procedure is to

---

[4] `https://web.mit.edu/drela/Public/web/xfoil/` Accessed on 10th June 2021

[5] `https://aerodynamics.lr.tudelft.nl/cgi-bin/afCDb` Accessed on 19th June 2021

determine the operating conditions of the propeller, this includes the RPM, diameter, number of blades, free stream velocity and either the design thrust or power. During the previous design phase the most optimal number of blades was determined to be 3 [4] as it gives a good balance between efficiency and thrust. To the determine the RPM at which the propeller should operate most efficiently, experimental data on motors commonly used on FPV racing drones was consulted to gain an estimate on typical RPM values [6]. It was decided to use an RPM of 31000 for the design of the propeller. The design was then performed for a range of Velocity and propeller diameter. The diameter ranged from 10 to 15 [cm] and the velocity from 60 to 90 [m/s]. The design thrust was also varied such that both the requirements on the straight line acceleration and turning performance can be met. Namely, the drone should be able to accelerate from stall speed to 90 [m/s] within 4.5 seconds and it should be able to perform a 25g turn at 60 [m/s]. As the propeller is designed for high velocities, there is a high likelihood that the blades will stall during hover. However, this is not an issue as long as the propeller is capable of providing sufficient thrust while stalled.

FPV drones most commonly use propellers which have a mounting hole diameter of 5 [mm], to accommodate for this, the hub diameter of the propeller was chosen to be 8 [mm]. The design parameters are then converted into inputs for the procedure. First, angular velocity of the propeller is determined through Equation 7.2 and the design RPM. Similarly, the thrust coefficient, T_c, and the speed ratio, λ, are determined from Equation 7.3 and Equation 7.4 respectively.

$$\Omega = \frac{RPM \cdot 2\pi}{60} \qquad (7.2) \qquad\qquad T_c = \frac{2T}{\rho V^2 \pi R^2} \qquad (7.3) \qquad\qquad \lambda = \frac{V}{\Omega R} \qquad (7.4)$$

The design procedure evaluates the propeller at multiple positions along the span of the blade, the non-dimensional radius, ξ, is given by Equation 7.5. It was chosen to evaluate the propeller at 30 different locations along the span. The positions r are then determined by applying a cosine spacing along the blade of the propeller. The cosine spacing is given by Equation 7.6 where u is given by uniformly spaced points from hub to tip. The benefit of the cosine spacing is that the points at which the propeller is evaluated are closer to each other in areas where the gradient in thrust is largest.

$$\xi = \frac{r}{R} \qquad (7.5) \qquad\qquad r = \left(R_{prop} - R_{hub}\right)\left(\frac{1}{2}\left(1-\cos(u)\right)\right) + R_{hub} \qquad (7.6)$$

Now that the design conditions have been specified, an initial estimate for the displacement velocity, ζ, can be made. As suggested by Adkins and Liebeck, ζ was set equal to zero. Using this estimate of zeta, the flow angle at the tip, ϕ_t, can be determined by applying Equation 7.7. Following, the Prandtl momentum loss factor can be determined using Equation 7.8 and Equation 7.9.

$$\tan(\phi_t) = \lambda\left(1 + \frac{\zeta}{2}\right) \quad (7.7) \qquad F = \left(\frac{2}{\pi}\right)\arccos(e^{-f}) \quad (7.8) \qquad f = \frac{\left(\frac{B}{2}\right)(1-\xi)}{\sin(\phi_t)} \quad (7.9)$$

The estimate for the flow angle at the tip can then be used to determine the flow angle along the rest of the blade using Equation 7.10. The circulation function, G, is then determined through Equation 7.11. The product of total velocity, W, and the chord, c, at each spanwise location is then calculated with Equation 7.12. C_l was determined through the airfoil analysis and is chosen at the point where the lift to drag ratio is maximal. From the product W·c, the Reynolds number at each station of the blade can be calculated, which was then used as a reference for the airfoil analysis.

$$\tan(\phi) = \frac{\tan(\phi_t)}{\xi} \qquad (7.10) \qquad\qquad G = F\cos(\phi)\sin(\phi) \qquad (7.11)$$

$$Wc = \frac{4\pi\lambda G V R \zeta}{C_l B} \qquad (7.12) \qquad\qquad Re = \frac{Wc\rho}{\mu} \qquad (7.13)$$

Next, the drag to lift ratio, ϵ, is calculated from the airfoil analysis data, at the angle of attack where it is minimal. Then the axial and rotational interference factors, a and a', are calculated using Equation 7.15 and Equation 7.16 respectively.

$$\epsilon = \frac{C_d}{C_l} \quad (7.14) \qquad a = \frac{\zeta}{2}\cos^2(\phi)\left(1 - \epsilon\tan(\phi)\right) \quad (7.15) \qquad a' = \frac{\zeta}{2x}\cos(\phi)\sin(\phi)\left(1 + \frac{\epsilon}{\tan(\phi)}\right) \quad (7.16)$$

After determining a and a', the total velocity and the blade angle at each radial position can be calculated using Equation 7.17 and Equation 7.18 respectively.

---

[6] `https://www.miniquadtestbench.com/` Accessed on 11th June 2021

$$W = \frac{V\,(1+a)}{\sin(\phi)} \qquad (7.17)$$

$$\beta = \alpha + \phi \qquad (7.18)$$

Equation 7.3 can be rewritten as a function ζ. This function is given by Equation 7.19, where I_1' and I_2' are given by Equation 7.20 and Equation 7.21 respectively.

$$T_c' = I_1'\zeta - I_2'\zeta \qquad (7.19)$$

$$I_1' = 4\xi G\,(1 - \epsilon\tan(\phi)) \qquad (7.20)$$

$$I_2' = \lambda\frac{I_1'}{2\xi}\left(1 + \frac{\epsilon}{\tan(\phi)}\right)\sin(\phi)\cos(\phi) \qquad (7.21)$$

I_1' and I_2' can then be integrated numerically from ξ_{hub} to ξ_{tip}. Since ζ is constant for an optimum design, for a specified thrust a new estimate of ζ is given by Equation 7.22. This new estimate of zeta then used as the starting point of the next iteration. This iteration was carried out until the values of the new and old ζ were within 0.01%.

$$\zeta = \frac{I_1}{2I_2} - \sqrt{\left(\frac{I_1}{2I_2}\right)^2 - \frac{T_c}{I_2}} \qquad (7.22)$$

7.3.3. Propeller Performance

Once the propeller geometry has been determined for the design condition, it is also possible to evaluate the performance of the propeller at conditions other than the design condition. First the force coefficients C_y and C_x have to be evaluated using Equation 7.24 and Equation 7.23 respectively. An initial estimate for ϕ can obtained through Equation 7.25 and setting ζ equal to zero and x is calculated through Equation 7.26. During normal operation the C_l is calculated using the angle of attack calculated through Equation 7.18, where beta is known from the design phase. When evaluating the performance of the propellers spinning in reverse direction this relationship is replaced with Equation 7.27.

$$C_x = C_l\,(\sin(\phi) + \epsilon\cos(\phi)) \qquad (7.23)$$

$$C_y = C_l\,(\cos(\phi) - \epsilon\sin(\phi)) \qquad (7.24)$$

$$\tan(\phi) = \frac{1 + \frac{\zeta}{2}}{x} \qquad (7.25)$$

$$x = \frac{\Omega r}{V} \qquad (7.26)$$

$$\beta = \phi - \alpha \qquad (7.27)$$

Then the flow angle along the blade can then be determined using Equation 7.28. These flow angles are then used to calculate the factors K and K’ using Equations 7.29 and 7.30.

$$\tan(\phi) = \frac{V\,(1+a)}{\Omega r\,(1-a')} \qquad (7.28)$$

$$K = \frac{C_y}{4\sin^2(\phi)} \qquad (7.29)$$

$$K' = \frac{C_x}{4\cos(\phi)\sin(\phi)} \qquad (7.30)$$

The axial and rotation interference factors have to be calculated using Equation 7.31 and Equation 7.32, where σ is given by Equation 7.33. Under some conditions, namely at the tip convergence of the iterative procedure may not be possible due to the large values of the interference factors. To improve convergence, the magnitude of a and a’ was limited to 0.7 [66]. To calculate the momentum loss factor, F, at off-design conditions, the flow angle at the tip should be evaluated using Equation 7.34.

$$a = \frac{\sigma K}{F - \sigma K} \qquad (7.31)$$

$$a' = \frac{\sigma K'}{F + \sigma K'} \qquad (7.32)$$

$$\sigma = \frac{Bc}{2\pi r} \qquad (7.33)$$

$$\tan(\phi_t) = \xi\tan(\phi) \qquad (7.34)$$

The differential forms of the thrust and power coefficient are given by Equation 7.35 and Equation 7.36. Numerically integrating these coefficients from ξ_{hub} to ξ_{tip} results in the thrust and power coefficients conventionally used to evaluated propeller performance.

$$C_T' = \frac{\frac{\pi^3}{4}\sigma C_y \xi^3 F^2}{\left((F + \sigma K')\cos(\phi)\right)^2} \qquad (7.35)$$

$$C_p' = \frac{C_T'\pi\xi C_x}{C_y} \qquad (7.36)$$

The thrust, power and efficiency of the propeller at varying values of the advance ratio, J, can then be determined using Equations 7.37 - 7.41.

$$J = \frac{V}{nD} \qquad (7.37)$$

$$\eta = J\cdot\frac{C_T}{C_P} \qquad (7.38)$$

$$T = C_T\rho n^2 D^4 \qquad (7.39)$$

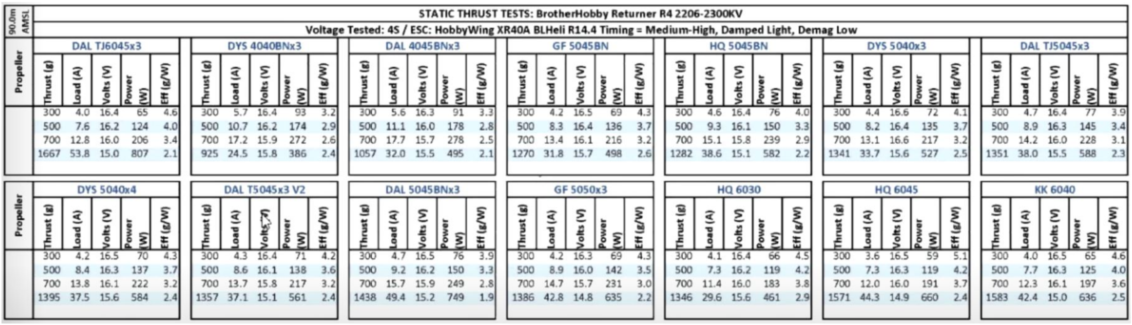
$$P = C_P\rho n^3 D^5 \qquad (7.40)$$

$$Q = \frac{P}{2\pi n} \qquad (7.41)$$

7.3.4. Analysis of Motors and Servos

Analysis on the motor and servo actuators is integral to evaluate compatibility with the subsystem, such that functionality will not be hindered during operation. The motors that are going to be integrated within Peregrine are Brushless DC motors due to their desirable traits. These include high power densities and efficiencies, good speed-torque characteristics and wide speed (RPM) range that they are able to provide. Several motors available on the market for FPV drone racing were looked upon and it was seen that most could not provide enough power or RPM that is required for Peregrine. The motor type finally selected that is going to be integrated within Peregrine is *Brotherhobby Returner R4 2206 2300kv Brushless Motor*.

Analysis was primarily conducted through the use of experimental data on the motors [7]. This testing involved thrust tests of the same motor with 10 different propellers of similar diameters and pitch which aided in getting a reliable torque RPM graph, rather than using data corresponding to a single propeller. Experimental data was requested from the manufacturer as well which verified the reliability of the results of tests performed by the third party. This can be seen in Figure 7.11 where static thrust testing was performed with a nominal voltage of 14.8[V].



90.0m AMSL

STATIC THRUST TESTS: BrotherHobby Returner R4 2206-2300KV

Voltage Tested: 4S / ESC: HobbyWing XR40A BLHeli R14.4 Timing = Medium-High, Damped Light, Demag Low

Propeller	Thrust (g)	Load (A)	Volts (V)	Power (W)	Eff (g/W)
DAL TJ6045x3	300	4.0	16.4	65	4.6
	500	7.6	16.2	124	4.0
	700	12.8	16.0	206	3.4
	1667	53.8	15.0	807	2.1
DYS 4040BNx3	300	5.7	16.4	93	3.2
	500	10.7	16.2	174	2.9
	700	17.2	15.9	272	2.6
	925	24.5	15.8	386	2.4
DAL 4045BNx3	300	5.6	16.3	91	3.3
	500	11.1	16.0	178	2.8
	700	17.7	15.7	278	2.5
	1057	32.0	15.5	495	2.1
GF 5045BN	300	4.2	16.5	69	4.3
	500	8.3	16.4	136	3.7
	700	13.4	16.1	216	3.2
	1270	31.8	15.7	498	2.6
HQ 5045BN	300	4.6	16.4	76	4.0
	500	9.3	16.1	150	3.3
	700	15.1	15.8	239	2.9
	1282	38.6	15.1	582	2.2
DYS 5040x3	300	4.4	16.6	72	4.1
	500	8.2	16.4	135	3.7
	700	13.1	16.6	217	3.2
	1341	33.7	15.6	527	2.5
DAL TJ5045x3	300	4.7	16.4	77	3.9
	500	8.9	16.3	145	3.4
	700	14.2	16.0	228	3.1
	1351	38.0	15.5	588	2.3
DYS 5040x4	300	4.2	16.5	70	4.3
	500	8.4	16.3	137	3.7
	700	13.8	16.1	222	3.2
	1395	37.5	15.6	584	2.4
DAL T5045x3 V2	300	4.3	16.4	71	4.2
	500	8.6	16.1	138	3.6
	700	13.7	15.8	217	3.2
	1357	37.1	15.1	561	2.4
DAL 5045BNx3	300	4.7	16.5	76	3.9
	500	9.2	16.2	150	3.3
	700	15.7	15.9	249	2.8
	1438	49.4	15.2	749	1.9
GF 5050x3	300	4.2	16.3	69	4.3
	500	8.9	16.0	142	3.5
	700	14.7	15.7	231	3.0
	1386	42.8	14.8	635	2.2
HQ 6030	300	4.1	16.4	66	4.5
	500	7.3	16.2	119	4.2
	700	11.4	16.0	183	3.8
	1346	29.6	15.6	461	2.9
HQ 6045	300	3.6	16.5	59	5.1
	500	7.3	16.3	119	4.2
	700	12.0	16.0	191	3.7
	1571	44.3	14.9	660	2.4
KK 6040	300	4.0	16.5	65	4.6
	500	7.7	16.3	125	4.0
	700	12.3	16.1	197	3.6
	1583	42.4	15.0	636	2.5

Figure 7.11: Static thrust testing data by manufacturer of Returner R4 2206 2300kv motor.

The experiments performed on the motor recorded the RPM, current and voltage values at different time steps from which the rotational speed, input power and torque values could be deduced using Equation 7.43, Equation 7.44 and Equation 7.46.

With the power input and output values for different propellers, the efficiencies were calculated using the simple relation Equation 7.45. It was realised that the efficiencies differed slightly per propeller which was due to the RPM values being different. The RPM and η relation can be determined from Equation 7.46. These were taken as data points for Figure 7.12. As it was necessary to verify that the maximum torque needed at a RPM of 31,000 does not exceed the stall torque of the motor, a τ-RPM graph was required to determine the stall torque of the motor, τ_s. The no load speed was easily calculated using the Kv rating of the motor and the nominal voltage of the battery.

$$n_0 = 2300[kv] \cdot 22.2[V] = 51060 \tag{7.42}$$

As torque vs RPM is a linear inverse relation (as long as the curve represents the direct output power), a line through the data points was made which yielded the stall torque of the motor (y-intercept of the graph), 0.2322[Nm]. The maximum torque needed to spin the propeller was determined to be 0.116[Nm], from Equation 7.41.

$$P_{in} = V \cdot I \tag{7.43}$$

$$P_{out} = \tau \cdot \omega = \tau \cdot RPM \cdot \frac{2\pi}{60} \tag{7.44}$$

$$\eta = \frac{P_{out}}{P_{in}} \tag{7.45}$$

$$\tau = \frac{\eta IV}{\omega} = \frac{\eta IV}{RPM} \cdot \frac{60}{2\pi} \tag{7.46}$$

To determine the power the motor can provide as a function of ω, the linear inverse relationship between torque and RPM can be used and Equation 7.47 can be derived.

$$P_{out}(\omega) = \tau_s \cdot \omega - (\frac{\tau_s}{n_0}) \cdot \omega^2 \tag{7.47}$$

With a maximum of 32.5[A] one motor can draw from the battery with a maximum discharge of 120[A] and a nominal voltage of 22.2[V], the maximum theoretical input power is 721.5[W] yielding a maximum output

---

[7] URL: https://miniquadtestbench.com/ Accessed 10th June 2021

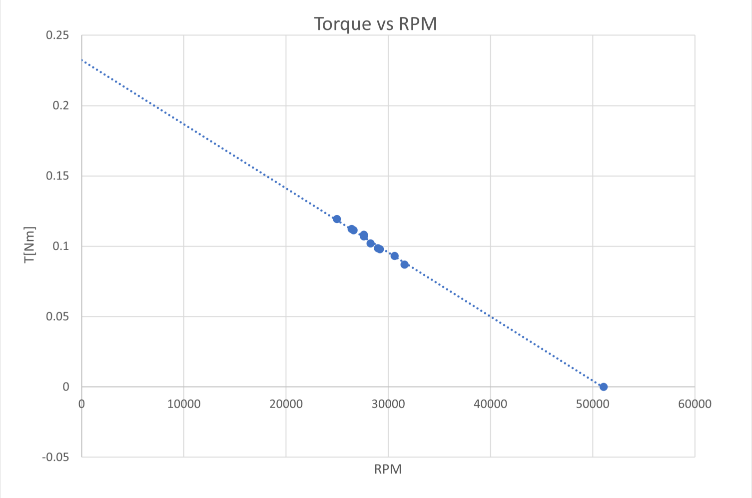


Figure 7.12: RPM plotted against Torque.

power of 577.2[W] with an assumed efficiency of 0.8 (this was the average efficiency for different similar sized propellers). However as the maximum required power by the best performing (most optimal) propeller was 657[W], a new propeller had to be chosen that required less than 577.2[W] so this motor could be successfully integrated. However to increase optimisation of the propulsion subsystem, a new motor(which meets the mass, power and RPM requirement) can be incorporated in the design.

The exact dimensions of the motor were requested from the manufacturer and were then provided to the structural department for integration in CATIA, shown in Figure 7.13.

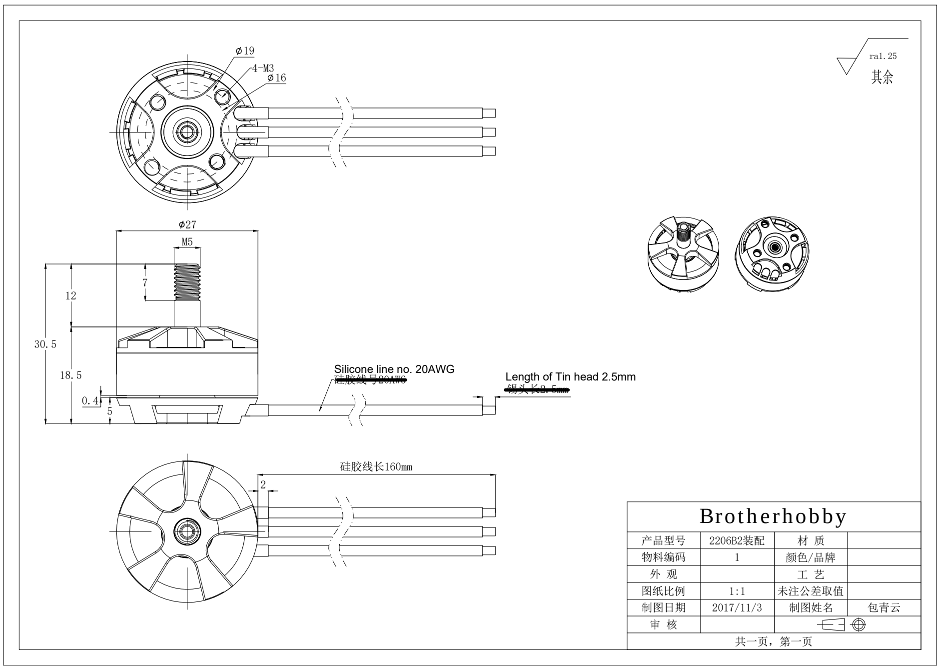


Figure 7.13: Brotherhobby Returner R4 2206 2300kv Brushless Motor dimensions.

Load on Servos

A free body diagram of the hinge, motor and propeller has been drawn to estimate the load on the servo and the speed at which the motors can be vectored. This free body diagrams is shown in Figure 7.14. The assembly has been separated into three parts, namely the hinge, motor and propeller each having their own weight and distance from the axis around which they rotate.

To calculate the angular acceleration of the vectored propeller, Equation 7.48 is used. The sum of torques can be derived from the free body diagram. The angle θ is defined as zero degrees when the motor is pointing forward and the drone is in nominal flight.

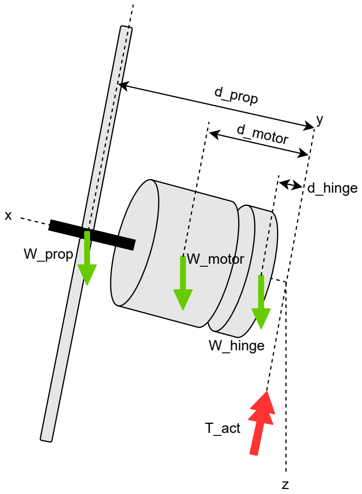


Figure 7.14: Motor actuator free body diagram

$$I_0\alpha_0 = \tau_{act} - W_{prop}d_{prop}\cos(\theta) - W_{Motor}d_{Motor}\cos(\theta) - W_{hinge}d_{hinge}\cos(\theta) \tag{7.48}$$

To calculate the mass moment of inertia of each of the components, the mass was assumed uniform throughout the component. The motor and attachment to the hinge were assumed to be cylinders, whereas the propeller was assumed to be a thin disk. The mass moment of inertia of a cylinder and of a thin disk are given by Equation 7.49 and Equation 7.50 respectively.

$$MI_{cyl} = \frac{1}{12}m\left(3R^2 + h^2\right) \tag{7.49}$$

$$MI_{disk} = \frac{1}{4}mR^2 \tag{7.50}$$

Where applicable, the parallel axis theorem was applied and the total moment of inertia was calculated as the sum of the individual moments of inertia. The maximum loading of the actuator will occur due to the centrifugal force during a 25g turn. The torque that the actuator chosen in the budgets is able to provide is equal to 0.137 [Nm]. However, the moment produced by the centrifugal force acting on the components is equal to 0.181 [Nm] while pointing forward during the turn, this means that the stall torque is too little and the actuator wouldn't be able to keep the components in place. As more powerful actuators would likely be too heavy and that the actuators only have to be able to keep the motors in place during the high speed, high g turns, it was decided to implement a mechanism to give the actuator some mechanical advantage. This mechanism increases the torque which the actuator can provide at the cost of losing some range of motion.

7.3.5. Load Analysis on Battery

To determine the required mass of the battery and the electrical loads the battery has to sustain during flight, the power consumption was calculated at each stage of the track. Note that the actual power consumption heavily depends on the pilots' preferred style of flying. The track can be seen in Figure 7.15. To determine the power consumption at each stage, a flight phase had to be assumed. Also note that the assumed flight profile is only applicable to the battery mass estimation. For this the track was split into multiple parts:

- Straight 1-2: The pilot will accelerate as much as possible for $\frac{2}{3}$ of the straight then maintain speed.
- Turn 2: This turn will utilize the drone's capability of performing a 25g turn at 60 [m/s].
- Straight 2-3: The pilot will maintain 60 [m/s] on this short straight.
- Turn 3: This turn will once again be performed at 60 [m/s]
- Straight 3-4: The pilot will maintain 60 [m/s] on this short straight.
- Turn 4: Another 25g turn at 60 [m/s].
- Straight 4-5: The pilot will accelerate as much as possible for $\frac{2}{3}$ of the straight then maintain speed.
- Turn 5: The final turn of the lap will be performed at 60 [m/s].

In the flight profile defined above the pilot will reach velocities larger than 60 [m/s], however there is no point defined at which the pilot will brake. Furthermore, all accelerations are assumed to be done at full throttle. Finally, The deceleration from before entering corners is assumed to be instant.

---

[8] `https://www.multigp.com/universal-time-trial-utt/` Accessed on 11th June 2021

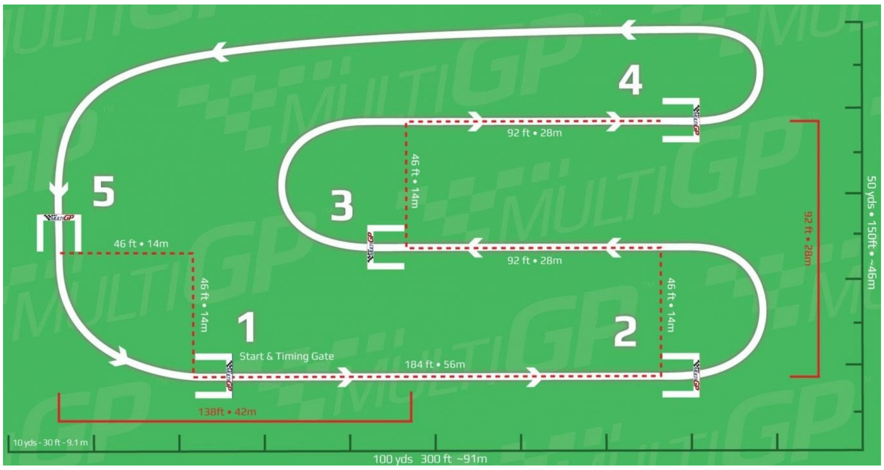


Figure 7.15: MultiGP Universal Time Track 1[8]

The power consumed by the system is then equal to the power absorbed by the propellers divided by the efficiency of the motors. The motor efficiency was assumed to be constant and equal to 0.72. Additionally, the power consumption of the other subsystems are taken into account and modelled as P$_{misc}$ in Equation 7.51. This term also contains the power loss in the cables assuming the cable length used in the budget. The cable power loss was calculated using Equation 7.53.

$$P = \frac{4C_p \rho n^3 D^5}{\eta_{motor}} + P_{misc} \quad (7.51) \qquad I = \frac{P}{V_{bat}} \quad (7.52) \qquad P_{cable} = \frac{I^2 \rho l}{A} \quad (7.53)$$

Integrating the power obtained at each time step leads to the total Energy used by the drone to race for four minutes. Equation 7.55 is then used to obtain the battery mass necessary to maintain 4 minutes of flight. The depth of discharge was taken as 0.85 and the energy density was taken from high end batteries seen on racing drones. This energy density was equal to 138 [$\frac{Wh}{kg}$].

$$E_{tot} = \int_{t=0s}^{t=240s} P dt \quad (7.54) \qquad m_{bat} = \frac{E_{tot}}{DOD \cdot 3600 \cdot E_{dens}} \quad (7.55)$$

From this analysis it follows a lap time of 5.43 seconds and a battery mass of 396 [g] are necessary if the pilot wants to adhere to the previously mentioned flight profile. Although the lap time is exceptional, this shows that it is not feasible to accelerate to 90 [m/s] during the straights of the track as the battery mass would exceed the mass budget by 226 grams. More importantly, from the analysis done on braking distance in Subsection 9.3.2, it shows that a distance of 105 [m] is required to brake from 90 to 60 [m/s], which exceeds the length of the longest straight on the given track. Therefore, a revised racing strategy is given below:

- Straight 1-2: During the first lap the pilot will accelerate from 0 to 60 m/s on this straight on subsequent laps the pilot will maintain 60 [m/s].
- Turn 2: This turn will utilise the drone's capability of performing a 25g turn at 60 [m/s].
- Straight 2-3: The pilot will maintain 60 [m/s] on this short straight.
- Turn 3: This turn will once again be performed at 60 [m/s]
- Straight 3-4: The pilot will maintain 60 [m/s] on this short straight.
- Turn 4: Another 25g turn at 60 [m/s].
- Straight 4-5: The pilot will maintain 60 [m/s].
- Turn 5: The final turn of the lap will be performed at 60 [m/s].

Using this flight profile would increase the time necessary to complete one lap by 0.395 seconds. This would bring the total lap time to 5.82 seconds. However, this time would still below the current record of 8.829 seconds by three seconds[9]. Note that this lap time is only theoretically possible and is very much influenced by the skill of the pilot and how well the flight profile can be adhered to. Flying such a small course at 60 [m/s] might prove difficult, however flying slower would increase the flight time further. The battery will drain faster if a different track is flown where the pilot is capable of accelerating to 90 [m/s]. To estimate this flight time, the average power consumption determined for flight profile 1 is used and applied to the battery capacity. The result of this analysis is that the pilot is capable of flying for 105 seconds. The final battery parameters

---

[9] `https://www.multigp.com/leaderboards/utt1/` Accessed 18th June 2021

are given in Table 7.2.

Table 7.2: Battery specifications

Parameter	Value	Unit	Requirement Passed
V$_{nominal}$	22.2	[V]	N.A.
Capacity	1.050	[A h]	PP-FLGT-ENDR-02.1
Mass	169.6	[g]	PP-PHYS-WGHT-01.1
Volume	105.8	[cm^3]	PP-PHYS-SIZE-01.1
P$_{avg}$	349.65	[W]	N.A.
P$_{peak}$	1835	[W]	N.A.

7.3.6. Chosen Propeller

This section describes the properties of the propeller chosen in Subsection 7.3.5 in more detail. This propeller is designed to provide a thrust of 3.0 [N] at a velocity of 86.0 [m/s] with a diameter of 0.128 [m]. This combination of input parameters led to a blade angle of 32.33 degrees at 75% of the radius. This blade angle can then be converted into the more commonly used propeller pitch, by use of Equation 7.56. The pitch of the propeller is equal to 19.1 [cm]. Furthermore, a set of four propellers is able to provide 18[N] of thrust at 60 [m/s], verifying **PP-FLGT-MOVE-03.1**. The thrust the propulsion system is able to provide during hover is equal to 10.8[N], this is however below the thrust required by **PP-FLGT-HOVE-01.1**, which says that the propulsion system shall be able to provide at least 17 [N] of thrust during hover. This does not have any implications on the drone's capability to hover as the thrust to weight ratio is still larger than one, even considering only two propellers. This also holds at an altitude of 1 km, where the air density is 9.3% lower.

$$p_g = 2\pi\frac{3}{4}R\tan\beta_{0.75} \tag{7.56}$$

The thrust coefficient as a function of the advance ratio is given in Figure 7.16. Notably, the thrust coefficient rapidly declines when the advance ratio is lower than 0.75, this is because the angle of attack at which the propeller operates increases as the advance ratio decreases. When the advance ratio is equal to 0.75 the propeller experiences stall. When the advance ratio decreases even further, the implementation of momentum-blade element theory did not converge anymore. Therefore, it was assumed that the thrust and power coefficients stayed constant once convergence became an issue. This was the case for advance ratios below 0.22. This assumption is supported by measurements done on small UAV propellers operating at low Reynolds numbers [14] [44]. Here it can be observed that the thrust and power coefficients remain almost constant at lower advance ratios even for propellers capable of producing thrust at advance ratios larger than 1.0. However, the chosen propeller has a much larger pitch than the propellers covered in the aforementioned papers. No data could be found on small scale propellers with a large pitch, therefore comparison is made with full scale propellers. The implication here is that the Reynolds number at which full scale propellers operate is much higher. Looking at high pitch propellers, $\beta_{0.75R} > 40°$, the thrust and power coefficient curves increase at low advance ratios [33]. The Reynolds number indicated for the experiments on full scale propellers is 234,000 for a model with a chord of 10 [cm].

The propeller can also be spun in reverse to produce reverse thrust for braking. Due to the large lift coefficient of the SG6043 airfoil at zero angle of attack, the propeller can only provide reverse thrust when operated at advance ratios below 1.31. Furthermore, the reverse thrust the propeller is able to provide is less than half of the forward thrust. A different airfoil with a lower zero lift angle of attack would be beneficial in providing extra reverse at the cost of the propeller stalling sooner during regular operation. The thrust coefficients related to reverse thrust are given in Figure 7.17.

The power coefficient as a function of the advance ratio is given in Figure 7.18. The same assumptions were applied here; the power coefficient is kept constant where momentum blade-element theory ceased to converge. The trends observed for the thrust coefficient can also be observed for the power coefficient. Namely, the power coefficient decreases rapidly after stall occurs. Finally, the efficiency of the propeller is given in Figure 7.19.

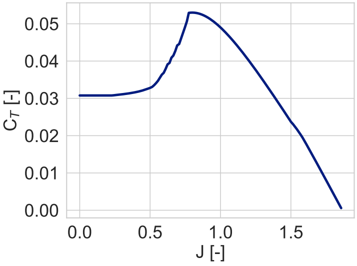


Figure 7.16: C_T as a function of J for the selected propeller

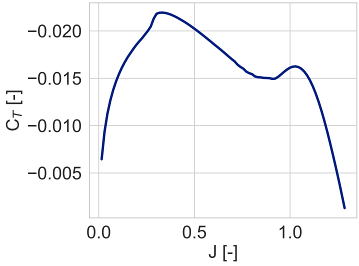


Figure 7.17: Reverse thrust capability of the selected propeller.

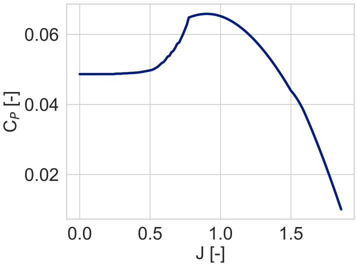


Figure 7.18: C_P as a function of J for the selected propeller.

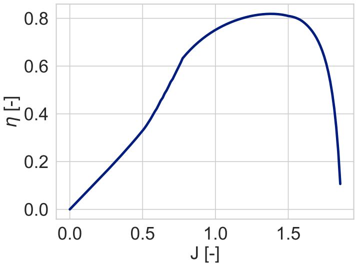


Figure 7.19: Efficiency of the selected propeller.

7.3.7. Feasibility of Regenerative Braking

The mechanism of regenerative braking has to be explored as it optimises performance by utilising the kinetic energy of the propellers attached to the DC brushless motors when the drone is decelerating/braking. As the weight requirement limits the type of battery that can be integrated, regenerative braking can prove useful by increasing the flight time. In regenerative braking the motor acts as a generator when the drone is naturally decelerating i.e. the propellers are in windmilling mode. Windmilling refers to the state when power being supplied to the motors by the battery goes to zero but the incoming flow keeps the propeller and hence the motor rotating in the same direction. This 'free' mechanical energy of the motor can then be transferred to the battery as electrical energy. The motor is controlled by the commutator which ensures commutation, which is a continuous process of switching the magnetic fields in the armature windings of the motor, see Figure 7.20. The position of the rotor which determines the commutation is known by the Hall effect sensors commonly used. [49]

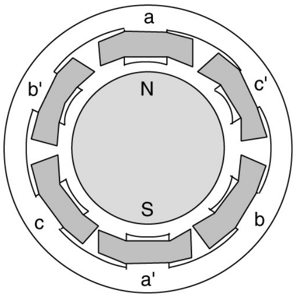


Figure 7.20: Schematic diagram of a three phase BLDC motor.

There are multiple methods to implement regenerative braking but the most applicable, simple and efficient one for Peregrine, after investigation is independent switching in conjunction with pulse-width modulation (PWM) or MOFSET control[49] [19]. Pulse-width modulation reduces the average power delivered by an electrical signal, by effectively chopping it up into discrete parts. This is suited for running motors as they are not easily affected by the discrete switching as their inertia causes them to react slowly although the switching frequency has to be high enough to not affect the load[10]. In independent switching all switches are turned off to allow current to flow back through the flyback diodes which ensures elimination of sudden voltage spikes. Figure 7.22 shows this current flow when the winding pairs of A and B phases are energised during regenerative braking as opposed to the conventional flow shown in Figure 7.21. The type of circuit employed is a half H-bridge with diodes and mofsets on all arms. An H-bridge is capable of switching the

---

[10] `https://www.ligtek.com/en/News_detail/2276/` Accessed on 13th June 2021

polarity in the windings in the stator hence allowing spinning of motors in opposite direction for reverse thrust.

To achieve recovery of energy, the mosfets on the high arms of the H-bridge are turned off and the low arms of H-bridge are controlled with PWM [49]. The switching steps are controlled by the control stage measuring the Hall Effect sensor readings, similar to the motoring process [19].

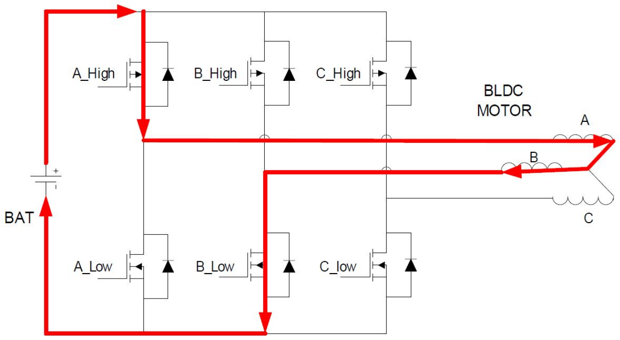


Figure 7.21: Conventional flow of current in a power inverter circuit.[19]

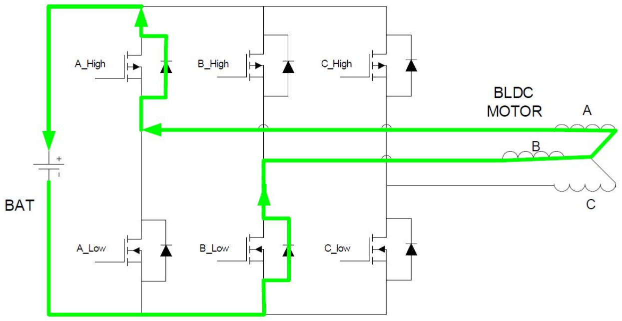


Figure 7.22: Regenerative current flow for a certain step of commutation.[19]

Even though the aforementioned circuit is already set up in the speed controller to allow for regenerative braking, it is not possible to evaluate how much energy can be gained exactly. This is because a mathematical model for the energy regeneration system requires many parameters that are unknown [17]. Due to this the results cannot be relied upon as it would require a batch of assumptions.

Additionally as the total designated flying time of the drone is 4 minutes and most of the time is spent racing i.e either maintaining or accelerating, the power that can be accumulated during racing would be very low as some power would be lost due to friction also. However, to determine the exact efficiency and power regeneration of the regenerative braking on Peregrine experiments will be performed later in the experimental phase of the design.

7.4. Risk Assessment

It is necessary to assess all the major risks associated with the Power & Propulsion subsystem to ensure safety and pilot awareness.

- **PP-R1:** Propeller fracture after high speed crash.
  - This risk can be accepted as the propellers have not been designed for high impact loads. Also it does not impose a severe problem as the propellers are easily replaceable and are not too costly.
- **PP-R2:** Single motor failure.
  - This is where the PID tuning by the control system would play a role in correcting the resulting yawing moment produced by the asymmetry in thrust. The tuning will correct the thrust differential to stabilise the drone. The required thrust differential and the design of the control system is discussed in Section 8.4 and Section 8.5. However, if the drone is competing in a race it is advised that the drone lands and the motor is replaced. Additionally it must be realised this risk has a low probability.
- **PP-R3:** Overheating of the battery.
  - This risk is dealt with the over-discharge protection and over-current protection of the battery. Circuits supporting protection will be integrated hence this risk no longer poses a threat.
- **PP-R4:** Battery runs out of power.
  - This risk can be mitigated by an on screen display warning on the controller of the pilot. This warning will appear at a cut off voltage of 3.5[V] at high throttle levels so the pilot is aware of the battery level being low and can safely land the drone.

7.5. Verification & Validation

This section deals with the verification and validation of the numerical models implemented for the Power & Propulsion subsytem analysis. Unit tests and System tests are discussed in Subsection 7.5.1 and Subsection 7.5.2 respectively, validation of X-foil results is done in Subsection 7.5.3.

7.5.1. Unit Tests

Multiple unit tests were performed on the propeller design. Firstly, the interpolation of the lift coefficient was visually compared to the experimental data points. As shown in Figure 7.23, the interpolation line follows the experimental data very closely and so passes the test. The drag coefficient was verified in a similar manner, but the graph has been omitted for the sake of brevity. Secondly, a test to verify that the angle of attack was constant and equal to the angle of attack that results in the maximum Cl/Cd was conducted. Indeed the outcome of this analysis was that the angle of attack was constant along the blade as shown in Figure 7.24. Lastly, the Prandtl correction factor was plotted along the blade. This correction factor should be equal 1 in the centre and decline to 0 at the tip. Due to the hub of the propeller, the blade starts a bit further out from the centre, hence the correction factor will not reach 1, as shown in Figure 7.25.

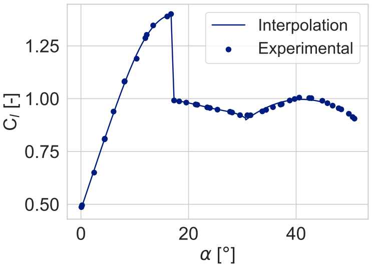


Figure 7.23: Lift coefficient interpolation from experimental data for the SG6043 airoil, Re=64000 [70]

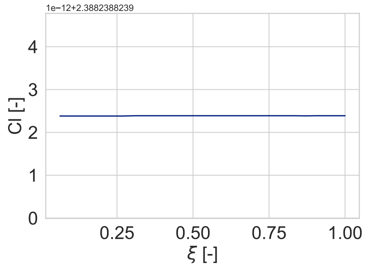


Figure 7.24: Lift coefficient along the propeller at the design condition

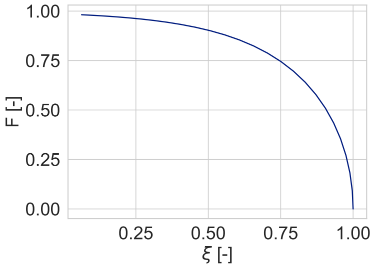


Figure 7.25: Prandtl correction factor along the propeller at the design condition

7.5.2. System Tests

System Test 1

To test whether the first part of the propeller design procedure is implemented correctly 5 propeller designs will be chosen at random and compared to the outcome of JavaProp[11]. JavaProp is a propeller design program that also implements momentum blade element theory, hence the only difference in performance should be attributed to the slight difference in airfoil data used. JavaProp uses data for the ARA-D 6% airfoil evaluated at a Reynolds number of 50000. To pass this verification test the difference between the power consumption at the design condition must be within 5% as the airfoil data affects the results. The design conditions of the propeller are given in the first column of Table 7.3, where D is the diameter, V the velocity and T the thrust. All five randomly selected propellers pass the verification test.

Table 7.3: Verification of propellers at design condition

Propeller	Code [W]	JavaProp [W]	% Difference	Result
D0.1_V86.0_T3.5	362	369	−1.8	Pass
D0.117_V79.0_T4.5	448	433	3.3	Pass
D0.117_V90.0_T3.125	330	334	−1.0	Pass
D0.128_V80.0_T4.0	392	384	2.0	Pass
D0.139_V88.0_T3.75	389	390	−0.45	Pass

System Test 2

A system test was also performed to verify the thrust and power coefficients for different advance ratios. This was done by comparing the output of the momentum blade element theory applied in Subsection 7.3.2 and Subsection 7.3.2 with results from JavaProp. The results can be seen in Figure 7.26 and Figure 7.27.

As seen from the figures above the thrust and power coefficients match quite well advance ratios greater than 1.26. The discrepancies seen for low advance ratios are due to a multiple factors. Even though JavaProp implements the momentum blade element theory, very limited aerodynamic properties can be defined for a propeller which reduces the accuracy of the results and restrains the comparison with different propellers [39]. Slight variations seen can be owed to the different aerodynamic properties. The red line in figures above corresponds to an input defined in JavaProp of ARA-D 6% airfoil operating at a Reynolds number of 50000 (most logical for comparison as the next higher Reynolds number input in JavaProp is 100,000) while

---

[11] `https://www.mh-aerotools.de/airfoils/javaprop.htm` Accessed 14th June 2021

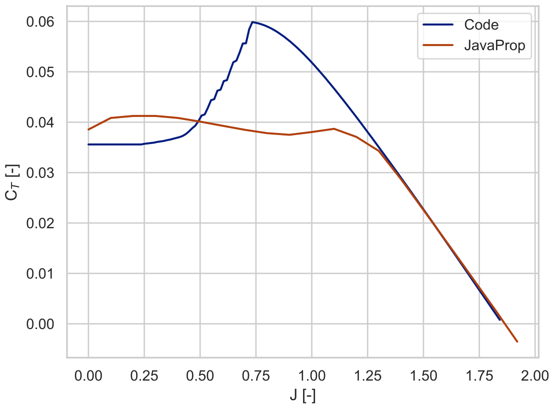


Figure 7.26: Thrust coefficient comparison between JavaProp and the written code.

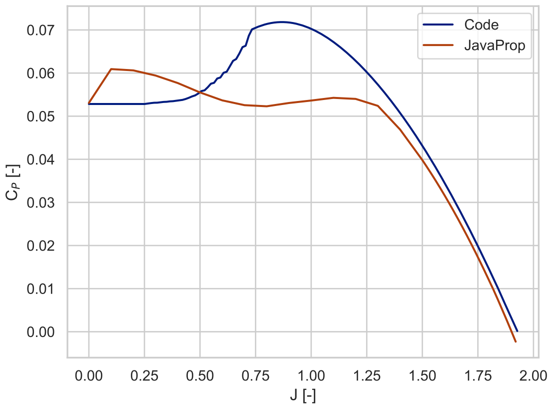


Figure 7.27: Power coefficient comparison between JavaProp and the written code.

the blue line depicts Peregrine's propeller performance which has the airfoil SG6043 operating at a Reynolds number of 64000. So the differences arise from the difference of inputs. From Figure 7.27, it can be observed that mostly C_P is being overestimated hence a conservative result. The peak difference at an advance ratio of 0.75 in C_T graph, Figure 7.26, is due to the aerodynamic profile of the propeller being different combined with the fact that the difference in Reynolds number is 14000 (21.9%). For advance ratios up to 0.50 the C_T is being underestimated (conservative result) and starts to increase for a range of J before matching the performance at J of 1.25. This is mainly due to SG6043 performing better than ARA-D 6% at higher angles of attack as concluded in Subsection 7.3.1. Therefore the results can be accepted.

7.5.3. Validation of Airfoil Performance

Even though as discussed in Subsection 7.3.1 experimental data was taken as an input, results from X-foil still need to be validated as the Reynolds number highly determines the convergence and reliability of the results. This step was performed for validating the comparison of different airfoils. This validation was performed through another set of experimental data. Wind tunnel data on airfoils tested at low Reynolds number was found for different airfoils [55]. To validate the results produced in Subsection 7.3.1, data for the lowest Reynolds number was chosen for comparison with results from Xfoil.

Table 7.4: A18 airfoil experimental data comparison with results in X-foil at a Re of 40400 [55].

α	$C_{l_{Xfoil}}$	$C_{l_{exp}}$	$C_{d_{Xfoil}}$	$C_{d_{exp}}$
-1.20	0.121	0.120	0.0248	0.0257
0.88	0.310	0.329	0.0251	0.0213
2.77	0.460	0.543	0.0336	0.0371
5.00	0.810	0.839	0.0445	0.0438
6.98	1.10	1.00	0.0345	0.0346
9.01	1.20	1.10	0.0670	0.0315

The lowest Reynolds number at which a wind tunnel test was performed was 40400, hence Xfoil was run at the same conditions with the same airfoil that was tested (A18 airfoil)[12]. It can be seen in Table 7.4, both C_l and C_d values are comparable with slight errors that can be accepted as at Reynolds number of 60000 and below, the flow is dominated by laminar separation bubbles that produce large variations in profile drag that make it difficult to confidently assign a measure of accuracy. At an angle of attack of 9.01°, it can be seen that even though Xfoil overestimates the C_l by 10% it also overestimates the C_d considerably hence underestimating the lift to drag ratio. Hence it can concluded that the results produced from Xfoil are acceptable and it does not overestimate overall performance at higher angles of attack. However as mentioned before the airfoil data used was extracted from an experimental setup [70].

---

[12] There was no data present on the airfoil that was chosen in Subsection 7.3.1

7.6. Compliance Matrix

There are still some requirements from power and propulsion which need to be evaluated at a later date. Due to time and resource constraints only a procedure to verify these requirements is outlined below:

- **PP-FEAS-COST-01.1, 01.2**: As the battery had to be customised for weight and volume optimisation, the exact costs still have to be confirmed with the manufacturer. The off-the-shelft battery costs approximately €33 which leaves €42 for the customisation. The portfolio of GREPOW[13] indicates that it is possible to customise the battery within the allocated €42 but this still has to be confirmed.
- **PP-FLGT-HOVE-05.1**: The thrust coefficients at low speeds were presented in Subsection 7.3.6. It can be said with confidence that this requirement can be met as enough thrust would be available to ensure stability. As extensive analysis have not been yet performed for the hover flight phase, it cannot be concluded that the requirement is complied with.
- **PP-FLGT-ROTA-05.1, 06.1, 09.1, 10.1, 11.1, 12.1, 13.1, 14.1**: Verification of all the requirements of roll, pitch and yaw rates from both hover and nominal flight mode have not been done individually. The performance of Peregrine as a whole covering these requirements will be discussed in Subsection 9.4.3.
- **PP-OPRT-RLBL-01.1**: The verification of the failure requirements will be discussed in Section 12.8.
- **PP-OPRT-FAIL-01.1**: The impact has not yet been defined in this stage of the design, a method to ensure the that the motors turn off upon a crash will be outlined in Section 12.9.
- **PP-ENVI-WTHR-02.1**: This requirement can be verified once the hover phase has been added to the simulation. Alternatively, the requirement can be verified in the testing phase by applying artificial crosswinds.
- **PP-ENVI-WTHR-03.1**: This requirement will be verified after acquiring specifications from the manufacturers on the components integrated in the electrical system.
- **PP-ENVI-WTHR-03.2**: This requirement will be verified after acquiring specifications from the manufacturer of the battery. This requirement can also be verified in the testing phase.

Table 7.5 shows the requirements and the current status of the verification.

Table 7.5: Compliance matrix for power & propulsion subsystem.

Identifier	Status	Reference	Identifier	Status	Reference
PP-FEAS-COST-01.1	TBC	Section 7.6	PP-FLGT-ROTA-08.1	Pass	Subsection 9.4.3
PP-FEAS-COST-01.2	TBC	Section 7.6	PP-FLGT-ROTA-09.1	TBC	Subsection 9.4.3
PP-FEAS-COST-03.1	Pass	Subsection 7.3.5	PP-FLGT-ROTA-10.1	TBC	Subsection 9.4.3
PP-FEAS-SUST-01.1	Pass	Section 11.3	PP-FLGT-ROTA-11.1	TBC	Subsection 9.4.3
PP-FEAS-SUST-01.2	Pass	Section 11.3	PP-FLGT-ROTA-12.1	TBC	Subsection 9.4.3
PP-FEAS-SUST-03.1	Pass	Section 11.3	PP-FLGT-ROTA-13.1	TBC	Subsection 9.4.3
PP-PHYS-WGHT-01.1	Pass	Subsection 7.3.3	PP-FLGT-ROTA-14.1	TBC	Subsection 9.4.3
PP-PHYS-WGHT-01.2	Pass	Section 12.6	PP-FLGT-ENDR-02.1	Pass	Subsection 7.3.5
PP-PHYS-SIZE-01.1	Pass	Subsection 7.3.5	PP-OPRT-MAIN-02.1	Pass	Section 12.6
PP-PHYS-SIZE-01.2	Pass	Section 12.6	PP-OPRT-MAIN-02.2	Pass	Section 12.6
PP-FLGT-HOVE-01.1	Fail	Subsection 7.3.6	PP-OPRT-PREP-03.1	Pass	Section 12.4
PP-FLGT-HOVE-05.1	TBC	Section 7.6	PP-OPRT-PREP-03.2	Pass	Section 12.4
PP-FLGT-MOVE-03.1	Pass	Subsection 7.3.6	PP-OPRT-RLBL-01.1	TBC	Section 7.6
PP-FLGT-MOVE-04.1	Pass	Subsection 9.3.2	PP-OPRT-FAIL-01.1	TBC	Section 7.6
PP-FLGT-ROTA-03.1	Pass	Subsection 9.4.3	PP-STRD-CSTM-01.1	Pass	Section 12.4
PP-FLGT-ROTA-04.1	Pass	Subsection 9.4.3	PP-ENVI-WTHR-02.1	TBC	Section 7.6
PP-FLGT-ROTA-05.1	TBC	Subsection 9.4.3	PP-ENVI-WTHR-03.1	TBC	Section 7.6
PP-FLGT-ROTA-06.1	TBC	Subsection 9.4.3	PP-ENVI-WTHR-03.2	TBC	Section 7.6
PP-FLGT-ROTA-07.1	Pass	Subsection 9.4.3	PP-ENVI-ALTD-01.1	Pass	Subsection 7.3.6
PP-PLTG-SIGN-02.1	Pass	Subsection 7.3.5			

---

[13] `https://www.grepow.com/` [Accessed 28/06/2021]

8

Control & Stability

This chapter covers the analysis and design of Control & Stability aspects of Peregrine, as well as the verification of related requirements. Following a functional recap in Section 8.1, the subsystem requirements are outlined in Section 8.2 and assumptions are discussed in Section 8.3. Next, Section 8.4 conducts an analysis regarding stability, selected components and control actuators such as elevons, thrust vectoring and differential thrust. Section 8.5 documents the design of the control system, including its architecture for different flight phases, control allocation as well as implementation. Section 8.6 provides a subsystem risk analysis, followed by verification in Section 8.7 and a requirements compliance matrix in Section 8.8. Finally, recommendations for further design are discussed in Section 8.9.

8.1. Functional Recap

The main function of the Control & Stability subsystem is to ensure that Peregrine is stable and controllable across its entire flight regime. Stability implies the ability to recover from disturbances, such as a sudden change in angle to attack or a wind gust, whereas controllability indicates the degree to which pilot inputs are effectively translated into suitable actuator responses. To realise these matters, the position of the center of gravity is considered, in order to ensure stability. Furthermore, the elevons are sized such that the required attitude rates and accelerations can be achieved. The last important contribution of this subsystem is the design, tuning and implementation of an effective control system.

8.2. Requirements

This section lists all requirements related to this subsystem, see Table 8.1.

Table 8.1: Subsystem requirements for control & stability

Identifier	Requirement	Method
CS-FEAS-COST-01.1	The cost of the control actuators and sensors shall not exceed €157.	Review of Design
CS-FEAS-COST-01.2	The cost of the controller shall not exceed €387.	Review of Design
CS-FEAS-COST-01.3	The cost of the goggles shall not exceed €550.	Review of Design
CS-FEAS-SUST-01.1	The control actuators and electronics shall be recyclable by 0% of its mass.	Review of Design
CS-PHYS-WGHT-01.1	The total mass of the electronics shall not exceed 27[g].	Review of Design
CS-FLGT-HOVE-02.1	The control actuators shall allow the drone to transition from hover to forward flight at minimum speed in less than 5 [s].	Analysis
CS-FLGT-HOVE-03.1	The control actuators shall allow the drone to transition from forward flight at minimum velocity to hover in less than 10 [s].	Analysis
CS-FLGT-MOVE-01.1	The drone shall remain controllable within the full velocity range.	Analysis
CS-FLGT-MOVE-05.1	The control subsystem shall ensure dynamic stability of the drone in forward flight.	Analysis
CS-FLGT-ROTA-02.1	The control surfaces and actuators shall facilitate a transition from horizontal flight to a 25g turn at 60 [m/s] within 1 [s].	Analysis
CS-FLGT-ROTA-03.1	The control actuators shall make it possible to obtain a pitch rate of at least 1000 [°/s] from nominal forward flight.	Analysis
CS-FLGT-ROTA-04.1	The control actuators shall make it possible to achieve max pitch rate from nominal forward flight within 1 [s].	Analysis
CS-FLGT-ROTA-05.1	The control actuators shall make it possible to obtain a yaw rate of at least 300 [°/s] from nominal forward flight.	Analysis
CS-FLGT-ROTA-06.1	The control actuators shall make it possible to achieve max yaw rate from nominal forward flight within 5 [s].	Analysis
CS-FLGT-ROTA-07.1	The control actuators shall make it possible to obtain a roll rate of at least 1080 [°/s] from nominal forward flight.	Analysis

Identifier	Requirement	Method
CS-FLGT-ROTA-08.1	The control actuators shall make it possible to achieve max roll rate from nominal forward flight within 1 [s].	Analysis
CS-FLGT-ROTA-09.1	The control actuators shall make it possible to obtain a pitch rate of at least 180 [°/s] from nominal hover flight.	Analysis
CS-FLGT-ROTA-10.1	The control actuators shall make it possible to achieve max pitch rate from nominal hover flight within 1 [s].	Analysis
CS-FLGT-ROTA-11.1	The control actuators shall make it possible to obtain a yaw rate of at least 180[°/s] from nominal hover flight.	Analysis
CS-FLGT-ROTA-12.1	The control actuators shall make it possible to achieve max yaw rate from nominal hover flight within 1[s].	Analysis
CS-FLGT-ROTA-13.1	The control actuators shall make it possible to obtain a roll rate of at least 90 [°/s] from nominal hover flight.	Analysis
CS-FLGT-ROTA-14.1	The control actuators shall make it possible to achieve max roll rate from nominal hover flight within 1 [s].	Analysis
CS-PLTG-SIGN-01.1	The control subsystem shall be able to transfer control commands to the drone up to a distance of at least 750 [m].	Similarity
CS-PLTG-SIGN-02.1	The control subsystem shall be able to transfer to the pilot up to a distance of at least 750 [m].	Similarity
CS-PLTG-SIGN-03.1	The video signal shall be transmitted in one of 8 different transmission channels.	Similarity
CS-PLTG-SIGN-03.2	The control signal shall be transmitted on a continuously changing transmission channel.	Similarity
CS-PLTG-INTR-01.1	At 100% of minimum maximum control distance, the control input latency shall not exceed 50 [ms].	Similarity
CS-PLTG-INTR-01.2	The stick input directions shall correspond to conventional racing drone controllers.	Review of Design
CS-PLTG-INTR-02.1	The drone pilot shall receive video at a minimum resolution of 0.045 [MP].	Review of Design
CS-PLTG-INTR-02.2	The drone pilot shall receive video at a minimum framerate of 50 [fps].	Review of Design
CS-PLTG-INTR-02.3	At 100% of minimum maximum control distance, the video latency shall not exceed 50 [ms].	Similarity
CS-PLTG-STAT-01.1	The drone shall evaluate its attitude about all three axis with a minimum accuracy of 0.5°.	Review of Design
CS-PLTG-STAT-01.2	The attitude shall have a minimum measurement frequency of 1 [kHz].	Review of Design
CS-PLTG-STAT-01.3	The Control & Stability subsystem shall allow for recalibration of attitude.	Review of Design
CS-PLTG-STAT-02.1	The drone shall evaluate its linear accelerations in three directions with a minimum accuracy of 2.5%.	Review of Design
CS-PLTG-STAT-02.2	The acceleration shall have a minimum measurement frequency of 1 [kHz].	Review of Design
CS-PLTG-STAT-02.3	The Control & Stability subsystem shall allow for recalibration of acceleration.	Review of Design
CS-OPRT-PREP-03.1	The control system shall establish a communication link within 3[s] when turned on.	Similarity
CS-OPRT-PREP-03.2	The control system shall be able to check pre-flight that all components are supplied with power and operational.	Review of Design
CS-OPRT-PREP-03.3	The control system shall be able to check pre-flight that the communication link is established.	Review of Design
CS-OPRT-RLBL-01.1	The control system shall have a failure rate of less than 0.33E-3 per flight hours.	Analysis
CS-OPRT-FAIL-01.1	The control system shall be able to detect whether an external impact has occurred.	Review of Design
CS-STRD-CSTM-01.1	The control system shall use standard drone communication protocols such as CSRF and SBUS.	Review of Design
CS-STRD-CSTM-01.2	The control system shall use a standard transmission channel for transmitting the video.	Review of Design

8.3. Assumptions

In this subsection, the assumptions that have been made for the design of the control system are listed and discussed, with respect to their effects and validity.

1. **Actuator Dynamics** - It is assumed that the control actuators are not limited by response time and rate of movement. This means they are immediately adjusted to the required control position as determined by the control allocation and can thus more effectively stabilise and respond to control inputs. However, this is not realistic as actuators would need some response time and are limited by their rate of movement, so this effect should not be neglected.
2. **Vectored Thrust** - It is assumed that the vertical component of thrust that results from vectoring is only dependent on the total maximum thrust in any given flight condition, not the forward component of the thrust. The individual components are later added together to create the total thrust vector. This means that vertical thrust is also simulated if there is zero forward thrust, which is not realistic as it would imply a vectoring angle of 90°while it is actually limited to 45°. The reason this assumption was made is that it makes the drone much more controllable inside the simulation at low speeds if zero throttle input is given.
3. **Propeller Wash** - It is assumed that the propellers do not affect the airflow behind them and thus the effectiveness of the elevons. In reality, positive thrust would cause the elevons to be more effective, whereas negative thrust would decrease their effect. This is not a negligible effect, however the complex underlying aerodynamics make it hard to quantify in different situations.
4. **Propeller Rolling Moment** - It is assumed that the inner and outer propellers of either side of the drone are synchronised such that the net effect on the rolling moment is zero. In reality, there may be a rolling moment caused due to their rotational speed. However, provided that the propellers are synchronised properly, this is a sensible assumption to make. Moreover, even if this is not exactly the case, the control system should naturally cancel out these effects.
5. **Propeller Velocity** - It is assumed that the velocity experienced by the propeller (that is used to calculate maximum thrust) is equal to the airspeed of the drone. In reality, vectoring of a propeller changes its direction and thus affects its experienced velocity. Provided that the vectoring angle is low, this would be a sensible assumption to make. However, it is less compatible with the above assumption on the vectoring angle. Nevertheless, this assumption has been made to avoid circular dependencies of the thrust on the vectoring angle and related true airspeed.

Beyond the DSE, with more time available to further study the control system of Peregrine, several of these assumptions could be eliminated and taken into account to further optimise and validate the design for real world usage. Nevertheless, despite these assumptions in place that result an effective simplification of the design task, the model already gives a detailed demonstration of how the control system would be implemented in a physical design.

8.4. Analysis

This section covers the analysis of the Control & Stability subsystem, including longitudinal and lateral stability in Subsection 8.4.1, electronic components in Subsection 8.4.2, sizing of the elevons in Subsection 8.4.3, as well as analysis of thrust vectoring and differential thrust in sections 8.4.4 and 8.4.5, respectively. The focus in this and following sections lies on cruise, not hover mode, as this is the most important flight phase in which the drone will spend most of its time. It is also the mode with the highest performance requirements, and thus in most cases the critical design condition.

8.4.1. Stability Analysis

Once Peregrine is in the air, it experiences several disturbances in flight condition. This section is concerned with the recovery of such disturbances, or in other words, the stability of Peregrine. The most relevant examples are a disturbance in angle of attack or a wind gust. These are examples of longitudinal and lateral stability, respectively. Efforts made to ensure static stability for these two cases are described in this subsection. Dynamic stability is not considered, as it is assumed that the control system will take care of eliminating eigenmotions.

Longitudinal Stability

For longitudinal stability in forward flight, it is preferred to have the center of gravity coinciding with the neutral point. Namely, if this is the case, Peregrine has neutral stability and its C_{m_α} is 0 [/rad]. In this case, there will not be any trouble with unstable pitch-up moments due to disturbances in angle of attack, while there is also no limitation on the agility due to constant pitch-down moments. The neutral point was found by the Aerodynamics department, using XFLR5. Although this is a broadly used aerodynamic software, its results can not be fully trusted. To account for possible errors, a safety margin of 5 % of the aerodynamic chord was proposed, as is also custom in aircraft design. Generally speaking, this has a negative effects on controllability, as now a moment opposing the direction of rotation is induced. Luckily, it was already found in previous analysis [4] that Peregrine can pitch very effectively, allowing the pilot to correct for this easily. The corresponding value of the center of gravity was initially used by the Control & Stability and Aerodynamics department. However, when finalising the CAD, Structures & Materials found that the actual CG was more forward than anticipated. As a result, Peregrine is more stable than desired. Luckily, it was found in the elevon sizing (see: Subsection 8.4.3) that rolling was more critical than pitching, therefore the increased longitudinal stability is not a problem for complying with requirements, despite it going at the cost of some agility performance.

Lateral Stability

In [4] it was already explained that for ideal directional stability a marginally positive C_{n_β} is preferred from a stability point of view. A weathervane stability of that order of magnitude will prevent Peregrine from reacting aggressively to wind gusts. In addition, the positive sign will help with making coordinated turns. When flying a coordinated turn, the angle of sideslip is zero and the drag is minimised [47]. Unfortunately, UAV with a flying wing configuration usually suffer from directional instability [41]. One of the ways to overcome this is letting the control system take care of maintaining direction. However, the control system is already heavily loaded, as flying wings are in other aspects considerably less stable than conventional aircraft. Moreover, at full throttle, Peregrine will have reduced yaw control, since thrust differential is used for this and the motors can not spin up anymore. The only option to correct for side slip disturbances (wind gusts) is then to spin down one of the propellers. This interferes with the full throttle command. Even though differential thrust is very effective, as is also analysed in Subsection 8.4.5, meaning high speed yaw can still be very stable, another method to improve on directional stability is adopted. This method is the implementation of winglets, which also benefits the design in other ways. In Equation 8.1 [61] it can be observed that winglets indeed can significantly contribute to C_{n_β}.

$$C_{n_{\beta_v}} = 2C_{L_\alpha}\frac{S_v}{S}l_v \tag{8.1}$$

For this reason, it was proposed to add winglets to the design. As these also have beneficial effects to aerodynamics and after it was verified by the Structures & Materials department that the implementation was feasible, winglets were indeed realised. Through analysis using AVL it was found by the Aerodynamics department that C_{n_β} has a value of 0.019222. This is a small and positive value. Because it satisfies the conditions set by this subsystem it was accepted and no further iterations were performed.

8.4.2. Analysis of Components

In this subsection, the selection of some components for Peregrine is considered. Both the sensors and elements responsible for transmitting signals between radio transmitter and Peregrine are discussed. All of the necessary components already have been assessed in the mass and cost budgets (see: Chapter 2). These components were already carefully selected and in this subsection their feasibility is further investigated by checking whether they comply with the requirements for the control & stability subsystem.

Sensors

The control system for Peregrine requires feedback on its angular rates so an associated sensor must be on board. This sensor is an Inertial Measurement Unit (IMU). Standard IMUs in racing drones make use of accelerometers and gyroscopes. They are included in the flight controller and it is possible to recalibrate these (important for **CS-PLTG-STAT-02.2**). For selection of the flight controller requirements **CS-PLTG-STAT-02.1** and **CS-PLTG-STAT-01.1** are important, which are related to sensor accuracy. These requirements can not be verified unfortunately, as they are badly phrased. For the latter requirement, an accuracy of 2.5 % is not always possible. For example, if the true value is 0 and the measured value something really small (so negligible absolute error), the percentage error will be infinite. Nevertheless, sensor accuracy is not deemed a problem as IMU's mostly suffer from accumulative errors [58] and Peregrine's flight time is only brief. After a race the IMU can be recalibrated, so that the accumulative error is removed. For the other accuracy requirement, an accuracy of 0.5°is also not possible. Namely, the flight controller does not have a tilt sensor, and including this would be a waste of mass as the drone does not need it. To satisfy **CS-PLTG-STAT-01.2** and **CS-PLTG-STAT-02.2** , the gyroscope and accelerometer require a sample frequency of at least 1 [kHz]. The frequently used IMU MPU-6000 has a gyroscope with a data output rate of 8 [kHz] and an accelerometer with a rate of 1 [kHz]. [1]. Another sensor that is necessary to comply with requirements (**CS-OPRT-PREP-03.2** and **CS-OPRT-PREP-03.3**) but not part of the IMU, is a current sensor. This sensor is incorporated in any standard racing drone speed controller (ESC). It can be concluded that selecting common drone racing components of good quality, such as the ones selected in the budgets, are sufficient to comply with the requirements mentioned above.

Communication Devices
Multiple components both on and off board Peregrine are needed to accomplish a remote connection. On board components are the receiver (RX), transmitter (VTX), antenna and camera. Off board components are radio transmitter, goggles and radio receiver. Requirement **CS-PLTG-SIGN-01.1** is complied with as the RX has a range of 3 [km] in an open area, which drone race grounds usually are. The same goes for **CS-PLTG-SIGN-03.1** and **CS-PLTG-SIGN-03.2** as the VTX has exactly 8 transmission channels and transmits signals on a continuously changing frequency. Furthermore, requirement **CS-PLTG-INTR-02.2**, which is related to the video resolution, can be checked off. The camera can record a quality of 0.48 [MP] and the goggles have a quality of 0.9 [MP]. The camera uses pal or ntsc signal system. These facilitate 25 and 30 fps respectively, but are also interlaced, giving them a real refesh rate of 50 and 60 fps [2]. This complies with **CS-PLTG-INTR-02.2**. Standard communication protocols such as CSRF and SBUS can be used for the radio transmitter, verifying **CS-STRD-CSTM-01.1**.

8.4.3. Sizing of the Elevons

The term elevon implies a combination of elevator and aileron into a single aerodynamic control surface that can actuate both roll and pitch as a result of differential and synchronised deflections, respectively. Combining these functionalities is advantageous to reduce the number of parts as well as structural weight.

Naturally, aerodynamic control surfaces are more effective at higher speeds, so the critical condition that influences sizing is the minimum speed at which the elevons alone should be able to control roll and pitch manoeuvres - i.e. without the help of thrust vectoring. Hence, the analysis was conducted at 45 [m/s], the minimum speed at which the elevons should be able to meet the above requirements exclusively. This is not to say that elevons cannot be used at lower speeds, but they will result in lower rates/accelerations and not be able to meet the requirements on their own (effectiveness scales with V^2). In addition, elevon deflections of 15°were used for sizing, independently for roll and pitch motions.

The cruise requirements regarding rates and accelerations for roll and pitch were used, namely **CS-FLGT-ROTA -03.1**, **-04.1**, **-07.1** and **-08.1**, which require roll and pitch rates of 1080 [°/s] = 18.8 [rad/s] and 1000 [°/s] = 17.5 [rad/s], respectively, and acceleration to these maximum rates within one second. For analysis, a safety factor of 1.2 was applied to the rates and 1.5 to the accelerations. The reasoning behind this is that there are arguably more uncertainties in accelerations as they vary over roll/pitch rates - while maximum acceleration is achieved at zero rate, the acceleration becomes zero at the maximum rate by definition. As such, the maximum acceleration was divided by 2 to estimate the average acceleration from zero to maximum rate, inducing additional uncertainty. Given these requirements, an optimisation code was created that would narrow down a list of specified elevon dimensions to those that optimally fulfilled the requirements. This means inputs that did not meet the requirements were discarded as well as those that exceeded critical requirements by more than 1% (after applied safety factors).

Applying a set of specified elevon dimensions to a fixed wing geometry, including chord ratio (c_{ratio}) and spanwise range (inboard b_1, outboard b_2 fixed at 0.95 $b/2$), these inputs would naturally results in values for effectiveness τ_{ev} and x-location x_{ev} (assumed at quarter-chord of the elevon spanwise middle), and thus be used to solve for $C_{l_{\delta_a}}$ and $C_{m_{\delta_e}}$ using equations 8.2 and 8.3, respectively. With additional aerodynamic parameters, the coefficients C_{l_p} and C_{m_q} were also determined using equations 8.4 and 8.5. All four equations have been adapted from [61].

$$\begin{aligned} C_{l_{\delta_a}} &= \frac{2C_{L_\alpha}\tau_{ev}}{Sb}\int_{b_1}^{b_2} y \cdot c(y)\, dy \qquad \Bigg| \quad c(y) = c_r\Big(1 + \frac{2(\lambda-1)}{b}y\Big) \\ &= \frac{2C_{L_\alpha}\tau_{ev}}{Sb}c_r\left[\frac{1}{2}y^2 + \frac{1}{3}y^3\frac{2(\lambda-1)}{b}\right]_{b_1}^{b_2} \end{aligned} \tag{8.2}$$

---

[1] `https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf` Accessed on 28th June 2021

[2] `https://oscarliang.com/fpv-camera-latency/` Accessed on 22nd June 2021

$$C_{m_{\delta_e}} = -C_{L_\alpha}\frac{x_{ev}-x_{cg}}{\bar{c}}\cdot\frac{b_2-b_1}{b/2}\tau_{ev} \tag{8.3}$$

$$C_{l_p} = C_{l_{p_{a=0}}} - \frac{1}{8}\frac{C_L}{\pi\cdot AR\cdot\cos^2\Lambda}\left(1+2\sin^2\Lambda\frac{AR+2\cos\Lambda}{AR+4\cos\Lambda}\right)-\frac{1}{8}C_D \tag{8.4}$$

$$C_{m_q} = -2C_{L_\alpha}(\bar{x}_{ac}-\bar{x}_{cg})\cdot|\bar{x}_{ac}-\bar{x}_{cg}| \tag{8.5}$$

Having determined all previous coefficients, the maximum rates and accelerations for roll (p, $\dot{p}$) and pitch (q, $\dot{q}$), at zero acceleration and rate respectively, follow from equations 8.6-8.9, simply derived from the sum of moments.

$$p_{\dot{p}=0} = -\frac{C_{l_{\delta_a}}\cdot\delta_a}{C_{l_p}}\frac{2V}{b} \tag{8.6}$$

$$q_{\dot{q}=0} = -\left(\frac{C_{m_{\delta_e}}\cdot\delta_e + C_{m_\alpha}\cdot\alpha}{C_{m_q}}\right)\frac{V}{\bar{c}} \tag{8.8}$$

$$\dot{p}_{p=0} = \frac{(C_{l_{\delta_a}}\cdot\delta_a)\,\frac{1}{2}\rho V^2 Sb}{I_{xx}} \tag{8.7}$$

$$\dot{q}_{q=0} = \frac{(C_{m_{\delta_e}}\cdot\delta_e + C_{m_\alpha}\cdot\alpha)\,\frac{1}{2}\rho V^2 S\bar{c}}{I_{yy}} \tag{8.9}$$

This resulted in the feasible optimal configurations shown in Figure 8.1. These results were presented to the Aerodynamics and Structures & Materials departments, who used this set of solutions to pick the optimal one for the design. Out of these, the last configuration was chosen. Its parameters are provided in Table 8.2.

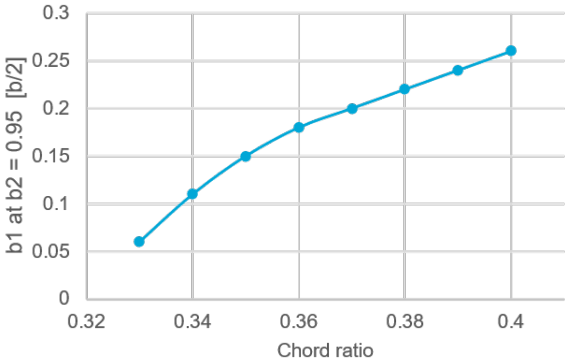


Figure 8.1: Optimal combinations of elevon dimensions for requirements at 45 [m/s] for 10°

Table 8.2: Parameters for chosen configuration: $c_{ratio} = 0.40$ [-], $b_1 = 0.26$ $[b/2]$, $b_2 = 0.95$ $[b/2]$

Parameter:	p [rad/s]	q [rad/s]	$\dot{p}$ [rad/s²]	$\dot{q}$ [rad/s²]	$C_{l_{\delta_a}}$ [/rad]	$C_{m_{\delta_e}}$ [/rad]	C_{l_p} [s/rad]	C_{m_q} [s/rad]
Value:	22.61	517200	246.0	5532	-0.438	-1.351	-0.441	-0.00017
AE:		155.5						-0.5656

This optimisation process resulted in relatively large elevons, due to the very high maneuverability requirements for the drone. As can be seen, the pitch rates and accelerations are quite high, especially for pitch, whereas the roll acceleration precisely meets the requirement times the safety factor.

Implementation by Aerodynamics of the determined elevon size was used to verify these coefficients, which had very similar results across the board apart from the value for C_{m_q} that was found to be much higher through AVL than Equation 8.5. Given that there were already concerns about the very small value of this coefficient and the resulting large pitch rate, the former value was taken over which also resulted in a more realistic maximum pitch rate. The Aerodynamics values are shown in the second row of Table 8.2. Note that this deviation did not affect the sizing of the elevon as pitch is the non-critical condition. In other words, the size of the elevon was solely determined to satisfy roll requirements.

At higher speeds the elevons are more effective - i.e. at 90 [m/s] the deflections will only need to be 1/4 of those at 45 [m/s] as their effectiveness scales with V^2. This means, for instance, in order to achieve maximum roll rate, the elevons have to deflect by only 2.5°. To achieve lower rates and stabilisation, the deflections need to deflect much less, on the order of 0.1°. Even if this is feasible for the actuator, the precision may not be as high as necessary. For this, a gearing mechanism could be applied which improves precision. On the other side, it then takes longer to actuate high deflections at low speeds, however this is not a concern for the chosen actuators as the reaction rates are very high. These points would be covered by further analysis.

Hinge Moments

Lastly, it was analysed whether the hinge moment created by the elevon deflection was acceptable. To do this, the Aerodynamics department found hinge moment coefficients for certain load cases. Using the elevon geometry and this coefficient the hinge moment can be calculated. The hinge moment is then compared to the maximum torque the actuator can deliver. In case the hinge moment exceeds this torque, the elevons must be resized, or a new actuator must be chosen. Equation 8.10 [47] was used to calculate the hinge moment.

$$H = C_h \cdot \frac{1}{2}\rho V^2 S_e \bar{c}_e \tag{8.10}$$

The critical hinge moment coefficient found at 90 [m/s] was when rolling 1080 [°/s]. This coefficient was 0.003187 [-], leading to a hinge moment of 0.6305 [kg cm]. This value is well under the maximum torque of 1.6-2 [kg cm] that the actuators can provide (the selected actuators can be found in Chapter 2). For this reason, the elevons do not have to be resized.

8.4.4. Analysis of Thrust Vectoring

In Subsection 8.4.3 the elevon geometry was determined at a speed of 45 [m/s]. This means that at lower speeds, the elevons are not capable of achieving the rates set in the requirements, according to the calculations performed in the analysis. Especially the requirement on roll rate is rather critical, due to the high roll damping contrasting the low pitch damping. At low velocities, especially close to the stall speeds, elevons are not usable at all, since an elevon deflection causes a reduction in stall angle. A chord ratio of 0.4 and an elevon deflection of 15°will cause the stall angle to be reduced as much as 4.9°[56]. For these reasons, the addition of thrust vectoring is desired to increase the attitude rates. The required thrust force perpendicular to the forward thrust required at stall speeds was calculated to verify if the requirements on the rates can be achieved through thrust vectoring. The addition of thrust vectoring to the moment equations for roll and pitch are formulated in equations 8.11-8.12.

$$l = C_{l_p} \cdot \frac{pb}{2V} \cdot \frac{1}{2}\rho V^2 Sb + 2 \cdot T_z \cdot y_o \tag{8.11}$$

$$m = (C_{m_q} \cdot \frac{q\bar{c}}{2V} + C_{m_\alpha} \cdot \alpha) \cdot \frac{1}{2}\rho V^2 S\bar{c} + 2 \cdot T_z \cdot x_o \tag{8.12}$$

In this equation, y_o and x_o are the offsets of the outboard motors to the CG. The control derivatives used were taken from Aerodynamics, for instance, C_{m_q} was more critical than calculated initially in Subsection 8.4.3. In Equation 8.12, a multiplication of $(q\bar{c}/2V)$ is used instead of the conventional $(q\bar{c}/V)$ as AVL uses the first mentioned expression by default. Setting Equation 8.11 and Equation 8.12 equal to zero, applying once again a safety factor of 1.2 to the required rates and solving for T_z at stall speeds yield the results shown in Table 8.3.

Table 8.3: Required upwards thrust force for satisfying requirements on roll and pitch rate at stall speed

Parameter	Roll	Pitch	Unit
T_z	1.52	3.27	[N]

The value of 3.27 [N] for pitch is unfeasible, as this would result in a total thrust of 6.54 [N], when vectoring the motors by 30°. It was found that vectoring the motors all the way up to 90°is structurally impossible, therefore it must be concluded that the requirements on rates can not be met at the lowest velocities. However, this is not deemed a reason to say the requirements for attitude rates are unfulfilled, since high rates are not considered as important at low velocities. Drone racing happens at high speeds and should be agile in that part of the flight envelope. Furthermore, Peregrine has been designed to make its 25 g turns at 60 [m/s] and at that speed it is capable of rolling and pitching extremely fast.

8.4.5. Analysis of Differential Thrust

So far, only the requirements on pitch and roll have been considered by sizing the elevons. In this section also the yaw rate will be considered. As explained in Subsection 8.4.1, Peregrine has favourable weather vane stability and therefore is not heavily dependent on its yaw capabilities. When necessary, yawing is possible by using differential thrust. This yaw rate is required to be 300 [°/s] = 5.24 [rad/s] by **CS-FLGT-ROTA-05.1**. To calculate the required amount of thrust differential, Equation 8.13 is used.

$$n = C_{n_r} \cdot \frac{rb}{2v} \cdot \frac{1}{2}\rho V^2 Sb + \Delta T \cdot y_o \tag{8.13}$$

The yaw damping C_{n_r} was found by the Aerodynamics department. From this equation ΔT can be found by equating to 0, using the yaw rate from the requirements multiplied by a safety factor of 1.2. This gives a required thrust differential of 0.1355 [N] at the most critical velocity of 90 [m/s]. This number is so low due to an almost negligible yaw damping of -0.008282 [s/rad]. The yaw acceleration can also be calculated by setting Equation 8.13 equal to $\dot{r} \cdot I_{zz}$ and r equal to 0. This gave a ΔT that was even less due to low inertia. The required thrust differential can be easily achieved by the motors, for this reason it can be concluded that the requirements on yaw in cruise flight are fulfilled.

8.5. Design of the Control System

This section discusses the design of the control system, including pilot inputs, control logic during hover mode, the transition and cruise flight. For the latter, a more detailed control architecture is discussed as well as different methods for control allocation.

8.5.1. Pilot Control Inputs

Due to the requirement for hover as well as the necessity to have a wing to achieve the required performance, the drone needs to be able to transition between the flight phases of hover and cruise. In order to keep the drone stable and controllable throughout these phases, distinct control systems that apply different input types are applied, as shown in Figure 8.2. During hover, velocity and attitude position are controlled (similar to a conventional drone), whereas during cruise, throttle and attitude rates are controlled (similar to a racing drone). The transition is intended to occur as an automated sequence that feels similar to taking off for the pilot, in terms of taking over full control.

The controller that will be used for the drone is a conventional racing drone controller (mode 2) where the vertical input on the left stick is not spring-centered. This is in conflict with some of the control inputs that are more suited to a spring-centered input (mode 1) such as vertical speed in hover. However, cruise is prioritised such that mode 2 is still preferred. Challenges with maintaining a centered input can be overcome by adding a filter that sets small stick deflections to zero, as has been implemented in Simulink. This does not fully solve the problem but is considered suitable to give the pilot intuitive control.

Hover Mode

In hover mode, when the drone is in tail-sitter position with the propellers facing upward (usually after take-off or before landing), the idea is to control it in a more conventional way. This means the pilot does not have to stabilise every movement and can focus on the attitude position and linear velocity - useful for getting into position at the start of a race or for finding an appropriate place to land. On the left stick, roll angle ϕ_x and pitch angle ϕ_y are input (thus indirectly sideways speed V_y and forward speed V_x), while on the right stick yaw rate r and vertical velocity V_z are input. Note that that a coordinate system with x facing forward and z facing downward is considered. When both sticks are centered, the drone will hover at zero airspeed and stabilise its attitude.

To achieve these control inputs, throttle is adjusted in sync for vertical velocity, or in differential direction to match the desired roll angle. For pitch angle and yaw rate, the thrust vectoring of the propellers is adjusted in synchronised or differential direction, respectively. Controlling the velocity/rate (as is the case for vertical velocity and yaw on the left stick) implies adjusting the associated actuators to a suitable level and keeping them at these levels until a different input is given. In contrast, controlling the position/angle (as is the case for roll and pitch on the right stick) implies adjusting the associated actuators to a suitable level for only a short moment until the desired angle is reached, then setting the actuators back to neutral position to sustain that angle.

The required feedback variables are estimated from the IMU that includes an accelerometer and gyroscope. For instance, vertical speed is found by integrating data from the accelerometer. While this information should be very precise when the drone takes off in hover mode, after transition from cruise mode back to hover this variable may need recalibration. A sensor that can be added, which is often part of the IMU, is a barometer that measures air pressure, a change of which can be related to an altitude change and thus vertical velocity. In further analysis, these sensors would be evaluated in more detail and sensor fusion researched to optimise data precision.

A PID loop and control allocation will be implemented to determine the required actuator responses of the drone following pilot inputs during flight. Several projects outlined in different articles, such as [50], have demonstrated the feasibility of such hover control system with similar actuators. Nevertheless, appropriate tuning and stabilisation is a quite challenging task. In addition to the previously defined actuators, the elevons could also be used for pitch and yaw control - given the blow-on effect from the propellers is strong enough, which is subject to further study.

Cruise Mode

In cruise mode, when the drone is in aircraft position with the propellers facing forward (most of the time during a race), it is controlled like a racing drone where the pilot has full control. All attitude rates as well as direct throttle are controlled on the right and left sticks as indicated in Figure 8.2. Note that this is again considering a coordinate system with x facing forward and z downward. This control method will be very suitable for racing drone pilots and ensure high agility - the only downside is that a flying wing feels different to control than a quad-rotor in terms of dynamic characteristics, i.e. aerodynamic coefficients.

In this mode, throttle is directly input by the pilot and does not need to run through a PID loop. To achieve the other control inputs, throttle is adjusted between the propellers to achieve the desired yaw rate, and when it comes to roll and pitch the elevons are adjusted in differential and/or synchronised directions, respectively. At low speeds, when the elevons are less effective, thrust vectoring is able to support their role in controlling pitch and roll rates. The control architecture will be explained in more detail in the next subsection.

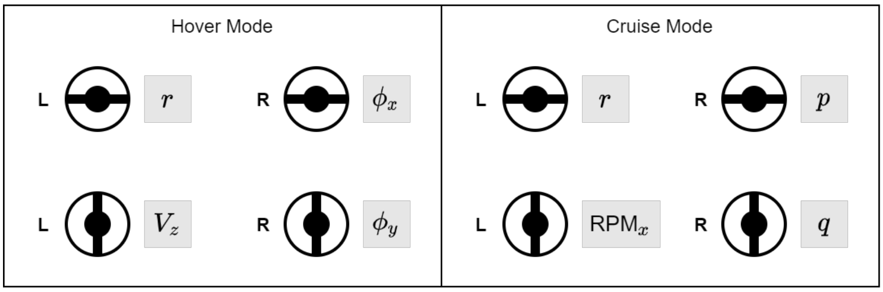


Figure 8.2: Pilot inputs for hover compared to cruise

Transition Logic

In order to change between hover and cruise mode, as discussed previously, a transition logic needs to be established. The idea is to design an automated sequence between hover and stall speed following a specific command by the pilot (press of a button/switch). For the pilot, this should feel similar to taking off or landing - in hover mode they don't have to worry about stabilisation and can simply adjust the position of the drone as needed, whereas in hover they are fully responsible for all rates making it much more challenging to fly but also agile.

In terms of transition algorithm design, the drone has to speed up while changing its pitch heading from upward-facing to forward-facing, until it flies faster than stall speed at an angle of attack such that the wing can generate sufficient lift. At this point, the drone maintains its throttle until the pilot provides the first input - only then the neutral stick position will be set to zero throttle. This is a safety mechanism to avoid immediate stall and gives the pilot some additional time to take over. When going into hover, the drone must pitch up while reducing speed until it faces upward with zero vertical velocity. At that point, it will hover in place until the pilot provides a different input.

Transitioning back into hover mode from any flight condition by the press of a button/switch can be an important safety feature to bring the drone to standstill as quick as possible. For instance, if a pilot is in a dangerous situation that may be destructive otherwise or if someone is new to flying Peregrine and has lost control of the drone, they can use it to get back to safety. This commercial aspect could make the drone more attractive to a wider range of users. Nevertheless, it is hard to implement an algorithm that is able to stabilise from any flight phase before transitioning into hover.

8.5.2. Control Architecture

As the most important flight phase in which the drone will operate most of its airborne time, this is the part of the control system on which the focus lied during the DSE. Hence, it has been worked out in much more detail than both hover and transition, and it has been fully implemented in Simulink (see Subsection 8.5.4). Moreover, the simulation discussed in Chapter 9 is purely based on cruise.

A diagram of the overall control architecture is shown in Figure 8.3, including pilot inputs, PID loops (turquoise), conversion blocks (white) as well as the control allocation (green) and dynamics model (orange). The four pilot inputs are the roll rate p, pitch rate q, yaw rate r and throttle (RPM) in forward direction.

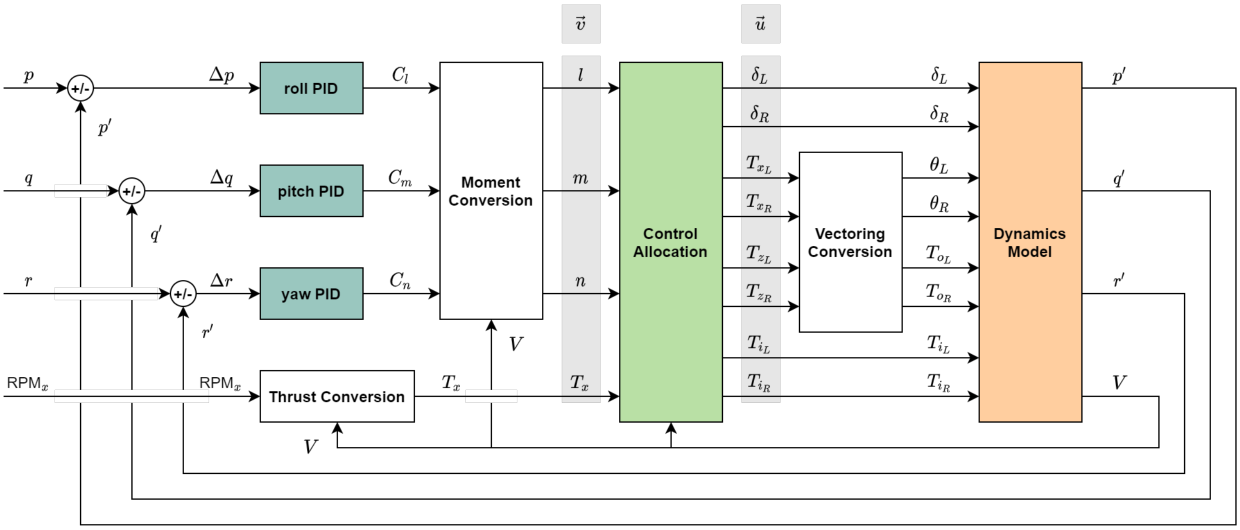


Figure 8.3: Overview of the control architecture for cruise mode

For the three attitude rates, at first the error in the signal goes through a PID loop to determine the optimal output (given appropriate tuning) that is considered a moment coefficient which is then converted to an actual moment using Equation 8.14, ensuring speed independence. At the same time, the fourth input, forward RPM, is converted to forward thrust given the measured velocity V and information about the thrust behaviour of the selected propellers (thrust coefficient C_T that is dependent on V and RPM) using Equation 8.15.

$$\begin{bmatrix} l \\ m \\ n \end{bmatrix} = \begin{bmatrix} C_l \cdot \frac{1}{2}\rho V^2 S b \\ C_m \cdot \frac{1}{2}\rho V^2 S \bar{c} \\ C_n \cdot \frac{1}{2}\rho V^2 S b \end{bmatrix} \tag{8.14}$$

$$T = C_T\left(\frac{V}{(RPM/60)\cdot D_p}\right)\cdot \rho\ (RPM/60)^2\ D_p^4 \tag{8.15}$$

Next, the variables l, m, n, T_x (combined as $\vec{v}$, see Equation 8.16) are fed to the control allocation, which will be discussed in more detail in the next subsection. The outputs of this block are elevon deflections, longitudinal and vertical thrust of the outboard motors as well as inboard thrust - each on the left and right side, resulting in a total of 8 output variables (combined as $\vec{u}$, see Equation 8.17).

$$\vec{v} = \begin{bmatrix} l & m & n & T_x \end{bmatrix}^T \tag{8.16}$$

$$\vec{u} = \begin{bmatrix} \delta_L & \delta_R & T_{z_L} & T_{z_R} & T_{x_L} & T_{x_R} & T_{i_L} & T_{i_R} \end{bmatrix}^T \tag{8.17}$$

Following the control allocation, the longitudinal (x) and vertical (z) thrust are converted to total thrust and vectoring angle of the outboard motors using equations 8.18-8.19. The reason that the latter variables were not chosen directly in the control allocation is to reduce dependencies as the vertical thrust that creates pitch/roll would be dependent on both the vectoring angle and overall thrust, making it much harder to implement.

$$T_{o_{L,R}} = \sqrt{T_{z_{L,R}}^2 + T_{x_{L,R}}^2} \tag{8.18}$$

$$\theta_{L,R} = \arctan\left(\frac{T_{z_{L,R}}}{T_{x_{L,R}}}\right) \tag{8.19}$$

These determined actuator responses flow into the dynamics model to simulate the drone behaviour, as discussed in Chapter 9. The relevant outputs from the dynamics model that flow back into the control system are attitude rates and airspeed.

8.5.3. Control Allocation

This subsection covers the control allocation block shown in Figure 8.3 in more detail. At first, an explanation of the purpose, inputs and outputs of this block is given, followed by a discussion of different methods.

One of the main challenges in designing the control system for a flying wing drone is the correct conversion of pilot inputs to actuator outputs. On the one hand, the pilot controls four parameters, namely the roll rate, pitch rate, yaw rate and RPM. On the other hand, these inputs need to be converted to eight control parameters, which are left and right elevon deflection, left and right thrust vectoring angle and thrust for the four motors. This incompatibility leads to a situation where there are infinitely many ways to satisfy the pilots demands. By effectively implementing the control allocation, the most efficient solution can be found.

The problem can be written mathematically in the following way. Solve for $\vec{u}$ such that Equation 8.20 is satisfied, where $\vec{v}$ is an $m \times 1$ vector containing the virtual control input (pilot input), $\vec{u}$ is an $n \times 1$ vector resembling the true control input (actual controls) and B is the so-called control effectiveness matrix, of shape $m \times n$.

$$B\vec{u} = \vec{v} \tag{8.20}$$

As it is hard to relate, for instance, an elevon deflection to attitude rates, $\vec{v}$ needs to contain something else than rates. This is solved by making use of the fact that aircraft controls can be thought of as moment generators [11]. Using this information and the fact that RPM can be easily converted to thrust, it was chosen to let $\vec{v} = \begin{bmatrix} l & m & n & T_x \end{bmatrix}^T$, containing the aircraft moments and thrust. With $\vec{u}$ already defined and justified previously as $\vec{u} = \begin{bmatrix} \delta_L & \delta_R & T_{z_L} & T_{z_R} & T_{x_L} & T_{x_R} & T_{i_L} & T_{i_R} \end{bmatrix}^T$, the only remaining part is the control effectiveness matrix B. It can be expressed generally as Equation 8.21. Using equations 8.22-8.25, which describe the moments and thrust due to the control actuators, to find all the partial derivatives leads to the matrix in Equation 8.26 of shape 4×8.

$$B = \begin{bmatrix} \frac{\partial v_1(\vec{u})}{\partial u_1} & \frac{\partial v_1(\vec{u})}{\partial u_2} & \cdots & \frac{\partial v_1(\vec{u})}{\partial u_n} \\ \frac{\partial v_2(\vec{u})}{\partial u_1} & \frac{\partial v_2(\vec{u})}{\partial u_2} & \cdots & \frac{\partial v_2(\vec{u})}{\partial u_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial v_m(\vec{u})}{\partial u_1} & \frac{\partial v_m(\vec{u})}{\partial u_2} & \cdots & \frac{\partial v_m(\vec{u})}{\partial u_n} \end{bmatrix}_{\text{ref}} \tag{8.21}$$

$$l = \frac{1}{2} C_{l_{\delta_a}} \left(\delta_L - \delta_R\right) \cdot \frac{1}{2} \rho V^2 S b + \left(T_{Lo_z} - T_{Ro_z}\right) y_o \tag{8.22}$$

$$m = \frac{1}{2} C_{m_{\delta_e}} \left(\delta_L + \delta_R\right) \cdot \frac{1}{2} \rho V^2 S \bar{c} + \left(T_{Lo_z} + T_{Ro_z}\right)(x_o - x_{cg}) \tag{8.23}$$

$$n = \left(T_{Lo_x} - T_{Ro_x}\right) \cdot y_o + \left(T_{L_i} - T_{R_i}\right) \cdot y_i \tag{8.24}$$

$$T_x = T_{x_L} + T_{x_R} + T_{i_L} + T_{i_R} \tag{8.25}$$

$$B = \begin{bmatrix} \frac{1}{2} C_{l_{\delta_a}} \cdot \frac{1}{2} \rho V^2 S b & -\frac{1}{2} C_{l_{\delta_a}} \cdot \frac{1}{2} \rho V^2 S b & y_o & -y_o & 0 & 0 & 0 & 0 \\ \frac{1}{2} C_{m_{\delta_e}} \cdot \frac{1}{2} \rho V^2 S \bar{c} & \frac{1}{2} C_{m_{\delta_e}} \cdot \frac{1}{2} \rho V^2 S \bar{c} & (x_o - x_{cg}) & (x_o - x_{cg}) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_o & -y_o & y_i & -y_i \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{8.26}$$

Moore-Penrose Pseudo-inverse

With all elements in Equation 8.20 defined, the question remains how this equation can be solved for $\vec{u}$. After all, B is not a square matrix and therefore not invertible. Hence, following the Moore-Penrose method, the pseudo-inverse is found in Equation 8.27 to yield a least-squares solution in Equation 8.28. [68]

$$P = B^T (B B^T)^{-1} \tag{8.27}$$

$$\vec{u} = P \vec{v} \tag{8.28}$$

The method described so far is unconstrained, meaning that control parameters can take any value that will solve the least-squares problem in the most efficient way. However, this is not realistic. For example, the elevons can only rotate by a certain amount without too big of a hinge moment being created, and the amount of available thrust is also limited. To take this into account, a constraint matrix N is implemented (diagonal $n \times n$) with entries equal to the reciprocal of the range of motion of the controls. This can be expressed as $N(i,i) = |u_{iMax} - u_{iMin}|$ and is implemented in the pseudo-inverse as shown in Equation 8.29. [11]

$$P = N(BN)^T \left(BN(BN)^T\right)^{-1} \tag{8.29}$$

In addition to constraints, one can implement weights for different elements of the vectors $\vec{u}$ or $\vec{v}$ to prioritise their values when finding an optimal solution. This is done using the weighting matrices W_u or W_v, respectively, which can be implemented as shown in equations 8.30-8.31. [68]

$$P = W_u^{-1} B^T \left(B W_u^{-1} B^T \right)^{-1} \tag{8.30}$$

$$P = B^T W_v^T \left(B B^T W_v^T \right)^{-1} \tag{8.31}$$

Combining the implementation of a constraint matrix and weighing matrix for $\vec{u}$ yields Equation 8.32.

$$P = N W_u^{-1} (BN)^T \left(B N W_u^{-1} (BN)^T \right)^{-1} \tag{8.32}$$

Linear Filter

A more effective way to implement constraints, on both actuator positions and rates, a linear filter can be used. Here, in addition to the current control input $\vec{v}(t)$, the allocation also depends on the the solution from the previous time step $\vec{u}(t-T)$. The problem in Equation 8.33 is solved for $\vec{u}$ that yields a least-squares solution, expressed by Equation 8.34. [54]

$$C(\vec{u}) = \left\| W_p^{-1/2} \vec{u}(t) \right\|^2 + \left\| W_r^{-1/2} [\vec{u}(t) - \vec{u}(t-T)] \right\|^2 \tag{8.33}$$

$$\vec{u}(t) = F\vec{u}(t-T) + G\vec{v}(t) \tag{8.34}$$

The components of Equation 8.34 are expanded in equations 8.35-8.37. [54]

$$F = (I - GB) W^{-2} W_r^{-2} \tag{8.35}$$

$$G = W^{-1} (BW^{-1})^T \left((BW^{-1})(BW^{-1})^T \right)^{-1} \tag{8.36}$$

$$W = (W_p^{-2} + W_r^{-2})^{1/2} \tag{8.37}$$

While B is the same as before, W_p is the weighing matrix of the current control allocation $\vec{u}(t)$, and W_r the weighing matrix of the difference between the current and previous allocation $[\vec{u}(t) - \vec{u}(t-T)]$. Given the structure of the equation, they have been logically chosen as diagonal with the entries as squared actuator position and rate limits, respectively, as shown in equations 8.38-8.40. [54]

$$W_p = \left[(\delta_{L_{max}} - \delta_{L_{min}})^2 \quad \dots \right]^T \tag{8.38}$$

(8.39)

$$W_r = \left[\left(\frac{d\delta_L}{dt} - \frac{d\delta_L}{dt} \right)^2 \quad \dots \right]^T \tag{8.40}$$

Constrained & Prioritised Allocation

Another more direct method has been proposed regarding a constrained least-squares solution. It aims to minimise the cost function given in Equation 8.41, which encompasses optimisation towards the solution as well as the desired value.

$$C(\vec{u}) = \left\| W_u(\vec{u} - \vec{u}_d) \right\|^2 + \gamma \left\| W_v(B\vec{u} - \vec{v}) \right\|^2 = \left\| \begin{bmatrix} \gamma^{1/2} W_v B \\ W_u \end{bmatrix} \vec{u} - \begin{bmatrix} \gamma^{1/2} W_v \vec{v} \\ W_u \vec{u}_d \end{bmatrix} \right\|^2 = \left\| A\vec{u} - b \right\|^2 \tag{8.41}$$

More specifically, the same control effectiveness matrix B ($m \times n$), vectors $\vec{u}$ and $\vec{v}$, weighing matrices W_u and W_v for the previous vectors are used. In addition, $\vec{u}_d$, $\vec{u}_{min}$, $\vec{u}_{max}$ contain the desired, minimum and maximum values of each element of $\vec{u}$, respectively. Parameter γ describes the relative importance of u vs. u_d compared to $B\vec{u}$ vs. $\vec{v}$.

To solve this problem, an existing MATLAB function called ”wls_alloc” inside the QCAT toolbox [3] has been used, that can be called as given in Equation 8.42. With the input parameters as explained above, it iteratively solves for the optimal output $\vec{u}$ (plus the optimal active set, W, that gives the saturation of each output as [-1, 0, 1] and the number of iterations, iter). [32]

[u, W, iter] = wls_alloc(B, v, umin, umax, Wv, Wu, ud, gamma) (8.42)

8.5.4. Implementation in Simulink

This subsection outlines the implementation process of the control architecture, including conversions, PIDs and control allocation, as outlined in the previous subsections. In addition, solutions that have been found to solve practical issues are described and the control allocation methods are evaluated.

---

[3] `http://research.harkegard.se/` Accessed 22nd June 2021

Thrust Conversion & Constraints
As previously explained, the control system takes throttle as its fourth input, which is then converted to thrust. This is done using propeller data from Power & Propulsion, and interpolating the given data points of the thrust coefficient function $C_T(J)$, where J is the advance ratio that contains a ratio of velocity and RPM. This interpolation is implemented in a MATLAB function that solves for the nearest data point and returns its value. The same has been done for reverse thrust when the RPM is negative. If the input value is outside the defined range, which hardly occurs, the thrust is set to zero.

Thrust conversion also plays an important role for determining the maximum available thrust at a given airspeed (depending on the maximum RPM, 30000). This is used to constraint the thrust level that is output from the control allocation.

For low speeds up to 30 [m/s] that require thrust vectoring, the vertical thrust $T_{z_{L,R}}$ is constrained at 60% (compared to 80% forward thrust) of maximum thrust at any given airspeed, ensuring that their combination does not exceed total thrust as $\sqrt{0.8^2 + 0.6^2} = 1$. As such, there is no trade-off between vertical and forward thrust, implying that the pilot can intuitively control the drone even at low speeds. At higher speeds, the vertical thrust contribution linearly decreases until it is set to zero above 45 [m/s]. At this point, the elevons take over completely, with a constraint of $\pm$ 15°.

PID Tuning
Following the implementation of the pseudo-inverse control allocation in Simulink with constraint matrix N, the model was tested with default gains and set up to function with the dynamics simulation, subject to troubleshooting and further changes to the model. At this point, the control system already seemed to work relatively well for most attitude rate inputs, as it was able to stabilise the drone and react quite sensibly. The gains were further adjusted to improve step response characteristics and stability. A proportional and integral gain of both 0.1 was chosen, as this would make the elevon deflections stay within physically feasible boundaries. Derivative gain was neglected due to its impracticality, even though it could have reduced some of the oscillations that were present in the simulation. This impracticality is related to the fact that derivative and high integral gains caused the runtime of the model to slow down by too much. As such, excessive tuning of the PID loop was avoided. Moreover, it was not absolutely necessary as the response characteristics of the drone were already better than expected. Another problem that was not solved by this method despite the inclusion of the matrix N, was the constraining of the actuator outputs to within their limits.

At this point, as mentioned previously, another method (Linear Filter) was discovered that used a slightly different method to constraint both actuator positions and rates, in hopes that it would solve the constraint problem in addition to reducing actuator output oscillations to a more realistic level. However, this also did not function quite as expected and it induced another severe problem: the solution procedure was much too slow. Part of the reason may have been the Simulink memory block that was used to store the previous state, however, research did not yield any suitable alternatives.

Control Allocation
Given the limited success with setting constraints in the first two control allocation methods, the final constrained & prioritised method was implemented. It has the most complete set of features, which all function very well as an isolated system - i.e. the actuator responses are in between their minimum and maximum as expected. However, when integrating the control system with the dynamics simulation, divergence problems in terms of thrust seemed to arise after running the simulation for only a short time. While this must have something to do with the fact that the outputs are constrained and can therefore only respond to stabilise to a limited extend, there must be another more fundamental issue. In theory, the actuators (in this case even without rate limits) should be able to sufficiently stabilise the drone.

Problem-solving by Simplification
In order to solve the problems related to the thrust allocation, a vastly simplified model was attempted. In this model, yaw is completely removed from the model. This was deemed a feasible option as yaw is by far the least import attitude control, the reasons for this are laid out in Subsection 8.4.1. Furthermore, the thrust was removed from the control allocation. Instead, the throttle setting directly controls the forward thrust on the 4 motors. T_z could be removed from $\vec{u}$ since now equation Equation 8.43 could be implemented.

$$T_z = T_x \cdot \theta \tag{8.43}$$

θ, the angle of a vectoring motor, is a linearisation of $tan(\theta)$. T_x is known from the throttle. This leads to $\vec{u} = \begin{bmatrix} \delta_L & \delta_R & \theta_L & \theta_R \end{bmatrix}$ and the control effectiveness matrix b only being a 2 x 4 matrix. This model showed very promising results even when using the unconstrained Moore-Penrose pseudo-inverse method. The controller could follow the control inputs accurately and the thrust of the motors formed a nice curve. However, the absence of yaw control did turn out to be a problem. When rolling, the adverse yaw made the system uncontrollable. To overcome this issue, yaw and thrust differential were added back, but only on the inner motors. Thrust could still be held apart of the control allocation by letting $\vec{u} = \begin{bmatrix}\delta_L & \delta_R & \theta_L & \theta_R & \Delta T\end{bmatrix}$. The thrust differential could be added and subtracted to the two inner motors' thrust. This method indeed solved the problem with yawing, but issues with the control system still arose. This is illustrated in Figure 8.4 and Figure 8.5. The first plot is the pitch response to a step input and the second plot the elevon deflections that were determined after control allocation. You can see that an elevon deflection of over 800 [rad] is demanded by the control system at a time of 1 second. After that, the elevon deflections continue in a path that corresponds to physically acceptable boundaries. When implementing constrained control allocation, the pitch response was not able to follow the step anymore. It seems that only a Dirac Delta like function can force the pitch to go to the requested value. As this is not feasible in real life, also this model can not be pursued without encountering problems.

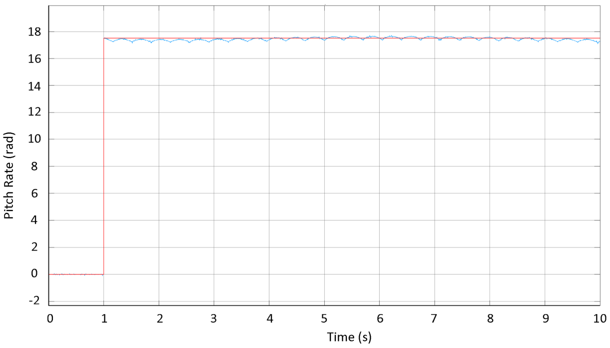


Figure 8.4: Pitch response (blue) plotted with time to step input (red) with $K_p = 0.4$, $K_i = 0.4$ and $K_d = 0.75$.

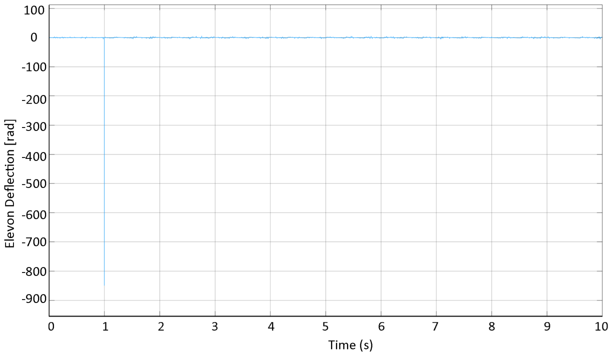


Figure 8.5: Elevon deflection plotted with time to step input on pitch with $K_p = 0.4$, $K_i = 0.4$ and $K_d = 0.75$.

Conclusion

As outlined above, several modifications to the system and smaller unit checks were conducted in an attempt to find the issue in the control system and/or overall model but with limited success to date. In the end, it was decided to instead spend some time on improving the implementation of the first method to make the simulation more intuitive to fly.

Given more time, the team would conduct further troubleshooting and other studies to find the root cause of the error. The reason may also be related to the complexity of the dynamics underlying this flying-wing drone - perhaps a more elaborate control method is required that uses, for instance, incremental control or an inner PID that is compatible with the system and can be optimised to run in real-time. Nevertheless, it is to early to make a definitive judgement regarding this statement.

8.6. Risk Assessment

In this section, the risks associated to the Control & Stability subsystem are discussed, including mitigation measures for these risks.

- **CS-R1**: Elevons incorrectly sized due to inaccurate stability and control derivatives
  - This risk is related to the fact that multiple stability and control derivatives such as C_{m_α} and $C_{m_{\delta_e}}$ can not be guaranteed to be fully correct without the use of wind tunnel and flight test or CFD. C_{m_α} for example, was found by means of the aerodynamic software XFLR, which is not completely accurate. $C_{m_{\delta_e}}$ was calculated by assuming that the lift increment due to an elevon deflection acts at the quarter chord of the control surface. These inaccurate values can lead to an uncorrectly sized elevon. If the surface is too small, then the agility of Peregrine will suffer. To reduce this risk, a safety factor of 1.2 was applied to the required attitude rates and one of 1.5 to the attitude accelerations.
- **CS-R2**: Required elevon deflections at high speeds too small to be accurately realised
  - At 90 [m/s] it was found by the Aerodynamics department through analysis using AVL that an elevon deflection of 1.64°is required to achieve the pitch rate of 1080 [°/s]. For lower pitch rates, this deflection is even smaller. It can not be assumed that these little deflections can be realised accurately. This is especially a problem when you consider the effect of aerodynamic loads that will elastically deform the elevon. To mitigate this risk, a gearing construction can be added to the design, that will help the actuators to deflect the elevons precisely. such a mechanism adds

weight and is also considered to be outside the scope of this design phase and is therefore not discussed in this report.

- **CS-R3**: System too complex to be controlled by PID
  - In [62] it is discussed that PID is not accurate enough to control complex hybrid MAV's. It is mentioned that even when angle of attack and velocity are modeled in the controller, systematic pitch errors can occur in the order of 20°. To improve the controller, PID can be expanded into INDI control. In the subsequent risk it is discussed why this mitigation strategy is not so easily implemented.
- **CS-R4**: No time to test and improve control system due to dynamics model not being finished
  - In the previous risk the proposal to expand PID control to incremental control in case of signs of inaccuracies was made. However, in order to pick up on these inaccuracies the PID first needs to be tested. This is only possible once the dynamics model for Peregrine developed in Chapter 9 is finished. The reason for this is the fact that the feedback required for the control loop is provided by the aforementioned model. Researching and implementing an INDI control is a time-consuming operation, as this is a new and unfamiliar technique. To mitigate the risk of not being able to implement a more advanced control system such as INDI, the functionality of it can be investigated before the dynamics model is finished. However, this is only possible if time allows it.

8.7. Verification & Validation

This section describes the verification and validation activities related to this subsystem. This includes ensuring that all calculations/methods are correct and that the outcomes are realistic. Firstly, these activities are covered with respect to the sizing/analysis of control actuators, followed by the Simulink/MATLAB control system.

8.7.1. Actuators: Elevons & Vectoring

Verification

The code for analysis/sizing of the control actuators was written in an object-oriented way with constant parameters as global variables which could be used by all functions inside a class. As such, errors with input values are effectively avoided and there is a clear overview of all inputs as they only need to be specified once. The functions as part of each class were made as modular as possible, often with the purpose of solving just a single equation, such that they could be tested independently and used as part of different methods.

Unit tests were made to verify these single functions, by giving mock inputs and comparing the output to values provided or calculated manually. In addition, different parameters were increased or decreased to check whether the output would change as expected.

After the successful completion of unit tests, the functions were assembled into more complete methods (often using a packaging function) that would return the desired output. Here it was tested that the functions worked together as expected and without errors, as well as that all steps in the process and the final output made sense (i.e. by printing relevant values). Nevertheless, given that the code for this task was not very complex, system tests were less relevant and added little value to the verification procedure.

Validation

For the elevon sizing the program was verified, but now for validation it is checked whether the calculated rates that were anticipated can actually be achieved. In Subsection 9.3.3, the results for the angular rate are shown. These results come from the simulation that was built. The simulation itself is validated, therefore it can be used to validate the elevons.

The simulation does not take into account the hinge moment created when deflecting the elevons. This hinge moment will be validated in the subsequent. The approach taken here is to use hinge moment coefficient data found from a CFD analysis on a NACA-0012 airfoil. The results from this study can be seen in Figure 8.6 [28]. These curves are for an angle of attack of 0°. The same study shows that the maximum effect of the angle of attack causes approximately a factor 1.1 higher hinge moment coefficients magnitudes. The curves show that the hinge moment coefficient of 0.003187 at circa 5°elevon deflections are very close to the coefficient of the CFD when the hinge is located at 73% of the chord (note: this is different than the chord ratio). With the results from this data the max hinge moment will clearly not be exceeded, and the elevon sizing is said to be validated.

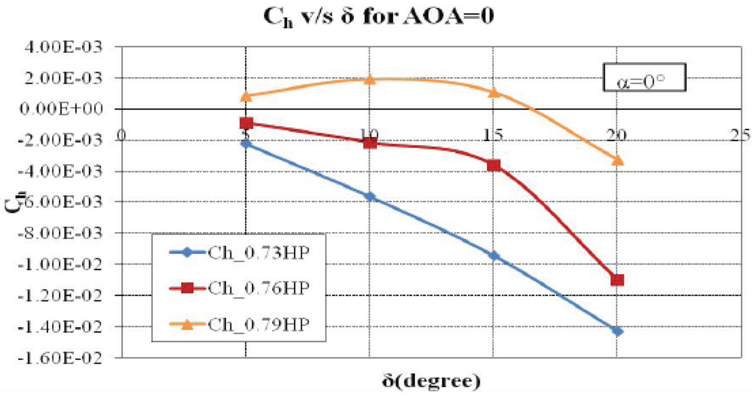


Figure 8.6: Hinge moment data for NACA0012 obtained from CFD analysis [28]

8.7.2. Control System

Verification

In Simulink, MATLAB functions and standard Simulink blocks were used. These require unit tests, so that it can be verified that they function in the way they were meant to. The blocks in Figure 8.3 are all tested, except for the moment conversion and vectoring conversion blocks, since they only contain trivial formulas.

To begin with, the thrust conversion block was tested. In Figure 8.7, the thrust (red line) can be seen for certain throttle inputs together with the velocity (blue). You can see that after about 1 second the throttle was increased. Until the 2.5 second mark it was kept constant. First the thrust rises because the velocity goes up, next it goes down because the tipping point has been reached where the C_T goes down with velocity. This phenomenon is described in more detail in Chapter 7. After 2.5 seconds, the throttle was gradually returned to 0, and the thrust responds accordingly. In shorter terms, the thrust does what is was meant to do and the thrust block is thus verified.

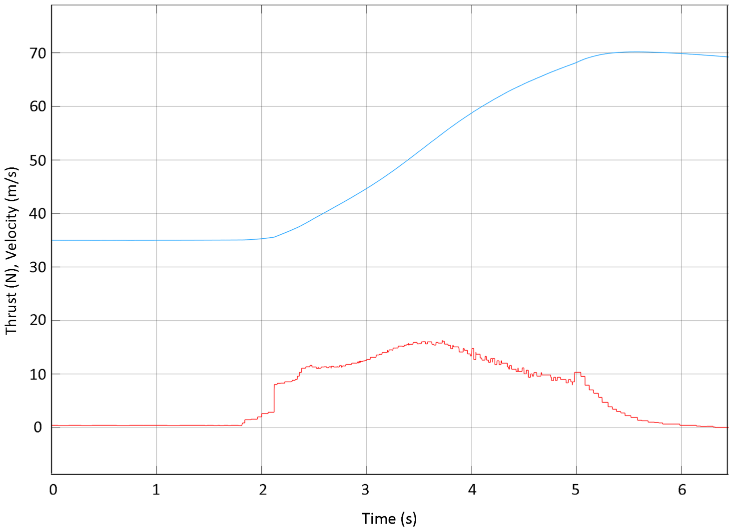


Figure 8.7: Thrust (red) and velocity (blue) resulting from throttle input.

After that, the PID blocks were submitted to unit testing. For this, a step input to roll was given with 4 different combinations of K_p, K_i and K_d. These combinations were compared and it was determined whether the differences made sense. In Table 8.4 the combinations are displayed, including a specification for the location of the response in Figure 8.8.

Table 8.4: Gains used for in Figure 8.8

K_p	K_i	K_d	Figure
0.1	0.1	0	Left
1	0.1	0	Middle Left
0.1	1	0	Middle Right
0.1	0.1	1	Right

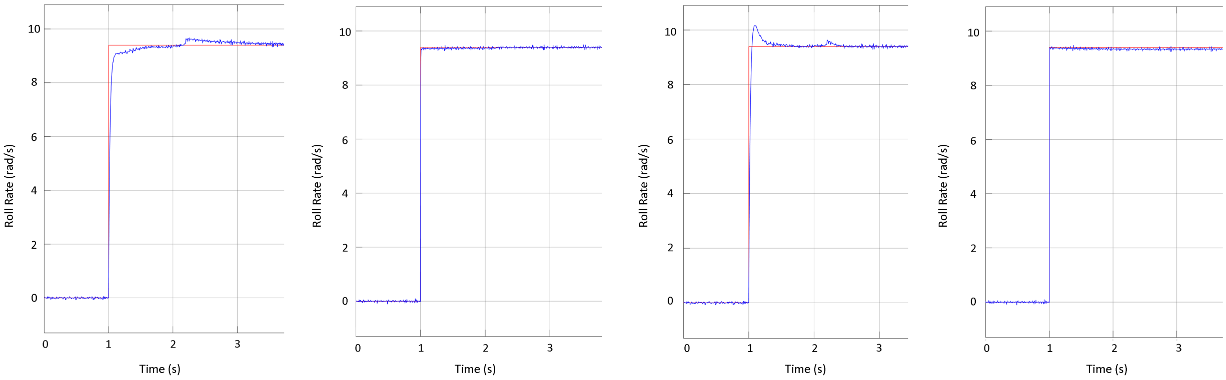


Figure 8.8: Roll rate responses to step. Left: $K_p = 0.1$, $K_i = 0.1$, $K_d = 0$, Middle Left: $K_p = 1$, $K_i = 0.1$, $K_d = 0$, Middle Right: $K_p = 0.1$, $K_i = 1$, $K_d = 0$, Right: $K_p = 0.1$, $K_i = 0.1$, $K_d = 1$.

The combination used in the left graph was used as a reference. When comparing the response to the middle left, which has significantly increased K_p, it can be observed that the rise time drastically decreases. Next, the main difference between the reference and the middle right graph is that the latter has much more overshoot, which is the result of K_i being increased. Then lastly in the right graph, a significant amount of K_d was added, leading to any overshoot being removed and a short settling time (the steady state error is reached very quickly). According to [65], these effects were indeed what was to be expected. The responses are very quick. This is due to the actuator responses being aggressive at the moment the step input starts. An extreme example was shown in Figure 8.5. Because of this, the maximum actuator rates can be exceeded. This can be taken into account by adding actuator dynamics, which were neglected so far, since it is very hard to estimate the parameters that describe these dynamics (damping ratio and natural frequency in the standard Simulink actuator dynamics block). Although the model is not completely realistic because of it, the PID block can be verified. After this, the control allocation block was tested by checking the output logic. When the system was subjected to a step on roll, the elevon deflections (δ) and upwards thrust (T_z) were mostly opposite in sign. When subjected to a step on pitch on the other hand, they were usually all of the same sign. Moreover, if the throttle setting was increased, motor thrust increased. The outputs of the control allocation block all make sense, and thus the control system is verified.

With the model verified, the integration with the system needs to be verified. In Table 9.6.1 tests on the whole system are performed including the pilot joystick and dynamics model. With these tests also the integration of the PID is verified, therefore this section does not report on any testing.

Validation

In order to validate the control system for Peregrine, ideally its responses to different pilot inputs would be compared with that of similarly controlled and configured UAV. Unfortunately, not a lot of useful data was found on fixed wing bi-rotor UAV with elevons. An article about the control system of a VTOL bi-rotor flying wing for hover mode gave some information. This UAV uses the same tail-sitter principle. The control system was also similar with all its close loops controlled by PID. Even though not reported on the control system in cruise mode, this vehicle was flown in test flights. [50]

A second article, on fixed wing UAV control using PID, did report on its attitude response. The UAV type that is considered here is more classical in configuration, using a tail surface with elevators. As mentioned, it was controlled by PID and had a similar noise model to that used in the dynamics model (see: Figure 9.4.1). The responses to a constant roll input of the control system described in this chapter and the reference model are shown in Figure 8.9 and Figure 8.10, respectively. It can be seen that these models have a very similar pattern, with irregularities caused by the model. The noise in Peregrine was significantly increased to match that in the reference article. [69]

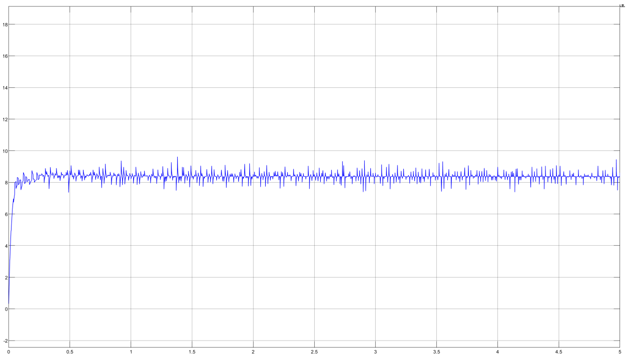


Figure 8.9: Response of Peregrine to step input on roll. Units on vertical axis in [rad/s] and horizontal axis in [s]

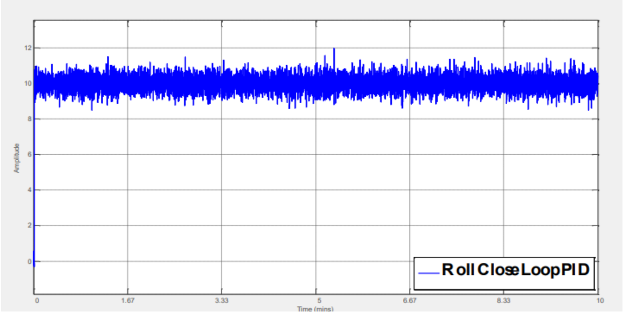


Figure 8.10: Roll response of fixed wing UAV to step input [69]

Another validation effort is based on the note that PID controlled UAV usually have considerable error for pitch [62]. Even though this is obviously not a favourable thing, this information can be used as a base for validation. Namely, if it can be shown that the designed control system has significant pitch error, it shows the same behaviour as other validated systems. In Figure 8.11 it can be observed that indeed the pitch shows a noticeable error.

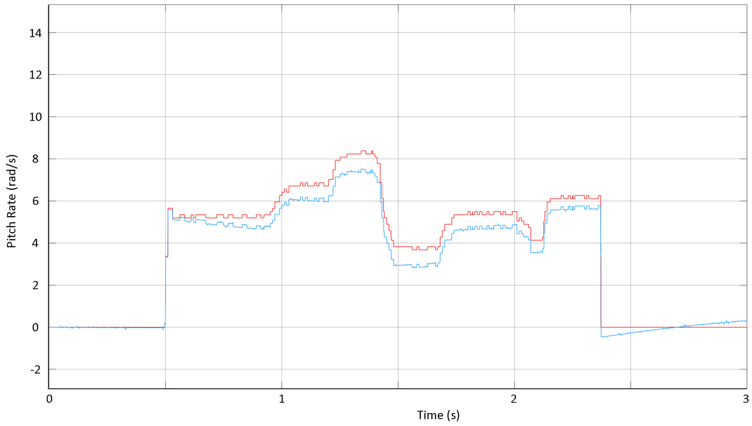


Figure 8.11: Control input on pitch (red) and pitch response (blue).]

8.8. Compliance Matrix

Table 8.5: Compliance matrix for Control & Stability

Identifier	Status	Reference	Identifier	Status	Reference
CS-FEAS-COST-01.1	Pass	Chapter 2	CS-PLTG-SIGN-01.1	Pass	Subsection 8.4.2
CS-FEAS-COST-01.2	Pass	Chapter 2	CS-PLTG-SIGN-02.1	Pass	Subsection 8.4.2
CS-FEAS-COST-01.3	Pass	Chapter 2	CS-PLTG-SIGN-03.1	Pass	Subsection 8.4.2
CS-FEAS-SUST-01.1	Pass	Chapter 11	CS-PLTG-SIGN-03.2	Pass	Subsection 8.4.2
CS-PHYS-WGHT-01.1	Pass	Chapter 2	CS-PLTG-INTR-01.1	TBC	Section 8.8
CS-FLGT-HOVE-02.1	TBC	Section 8.8	CS-PLTG-INTR-01.2	Pass	Subsection 8.5.1
CS-FLGT-HOVE-03.1	TBC	Section 8.8	CS-PLTG-INTR-02.1	Pass	Subsection 8.4.2
CS-FLGT-MOVE-01.1	Pass	Section 8.5	CS-PLTG-INTR-02.2	Pass	Subsection 8.4.2
CS-FLGT-MOVE-05.1	Pass	Section 8.5	CS-PLTG-INTR-02.3	TBC	Section 8.8
CS-FLGT-ROTA-02.1	Pass	Subsection 9.3.3	CS-PLTG-STAT-01.1	Fail	Subsection 8.4.2
CS-FLGT-ROTA-03.1	Pass	Subsection 8.4.3	CS-PLTG-STAT-01.2	Pass	Subsection 8.4.2
CS-FLGT-ROTA-04.1	Pass	Subsection 8.4.3	CS-PLTG-STAT-01.3	Pass	Subsection 8.4.2
CS-FLGT-ROTA-05.1	Pass	Subsection 8.4.5	CS-PLTG-STAT-02.1	Fail	Subsection 8.4.2
CS-FLGT-ROTA-06.1	Pass	Subsection 8.4.5	CS-PLTG-STAT-02.2	Pass	Subsection 8.4.2
CS-FLGT-ROTA-07.1	Pass	Subsection 8.4.3	CS-PLTG-STAT-02.3	Pass	Subsection 8.4.2
CS-FLGT-ROTA-08.1	Pass	Subsection 8.4.3	CS-OPRT-PREP-03.1	TBC	Section 8.8
CS-FLGT-ROTA-09.1	TBC	Section 8.8	CS-OPRT-PREP-03.2	Pass	Subsection 8.4.2

CS-FLGT-ROTA-10.1	TBC	Section 8.8	CS-OPRT-PREP-03.3	Pass	Subsection 8.4.2
CS-FLGT-ROTA-11.1	TBC	Section 8.8	CS-OPRT-RLBL-01.1	TBC	Section 8.8
CS-FLGT-ROTA-12.1	TBC	Section 8.8	CS-OPRT-FAIL-01.1	TBC	Section 8.8
CS-FLGT-ROTA-13.1	TBC	Section 8.8	CS-STRD-CSTM-01.1	Pass	Subsection 8.4.2
CS-FLGT-ROTA-14.1	TBC	Section 8.8	CS-STRD-CSTM-01.2	Pass	Subsection 8.4.2

The reason that **CS-PLTG-STAT-01.1** and **CS-PLTG-STAT-02.1** failed is because of their bad phrasing. Subsection 8.4.2 explains the issues with the requirements and why failure is non-problematic.

The remaining requirements will be verified as explained in Table 8.6.

Table 8.6: Verification strategies for requirements still to be confirmed

Requirements	**Verification Strategy**
CS-FLGT-HOVE-02.1 CS-FLGT-HOVE-03.1	These requirements deal with hover flight. With more available time, the simulation would be extended beyond cruise to include hover control and dynamics, as well as the transition logic that have been outlined in Section 8.5. Alternatively, an estimation can be made by calculating the transition time based on acceleration due to thrust to achieve stall speed and the time required to complete a pitch change to horizontal flight in a stable manner. Given the high performance of the drone, transitioning should not be design critical.
CS-FLGT-ROTA-09.1 CS-FLGT-ROTA-10.1 CS-FLGT-ROTA-11.1 CS-FLGT-ROTA-12.1 CS-FLGT-ROTA-13.1 CS-FLGT-ROTA-14.1	To verify the requirements on attitude rates and accelerations in hover mode (90, 180 and 180 [°/s] for roll, pitch and yaw, respectively, each to be achieved within 1 second), a similar analysis to what has been done in Section 8.4 can be conducted for hover instead of cruise. The dynamics behaviour will be different, so will the coefficients in that flight mode, leading to different results compared to cruise. Nevertheless, a similar methodology can be applied to estimate the rates. These coefficients can also be verified by Aerodynamics and the results can be implemented in the simulation to validate the rates.
CS-PLTG-INTR-01.1 CS-PLTG-INTR-02.3	In order to verify the latency, tests will be performed once the entire system is combined. Simple tests are available for drones that serve this purpose [4]. If the requirement is not met, new components such as a better transmitter or camera can be chosen and verified again.
CS-OPRT-PREP-03.1	To address the time taken to establish a communication link, a physical test of the chosen components should be completed.
CS-OPRT-RLBL-01.1	The reliability of the control system has been roughly estimated in Section 12.8. Result accuracy can be improved through additional tests of running the control system many times and applying statistical methods to determine the failure rate.
CS-OPRT-FAIL-01.1	The ability to detect external impacts can be verified by inputting a sudden deceleration into the control simulation and evaluating its response. In reality, this would be detected by the accelerometer.

8.9. Recommendations for Further Design

With more time at hand to expand on the conducted analysis and design process throughout this chapter, the focus would be put on three main aspects. Firstly and most critically, making sure that the control output constraints are functional inside the simulation so that the full cruise envelope can be realistically tested. Secondly, analysing manoeuvrability in hover and implementing it as well as the transition phase in the simulation. Finally, once sufficient verification of all flight modes is complete, the control system can be fully validated by installing it on a prototype of the drone and conducting physical flight testing.

9

Performance & Simulation

This chapter will aim to present the race performance analysis for Peregrine along with the simulation model created to verify the top level requirements of the drone. Section 9.1 will provide an introduction and background for this subsystem and specify its role in the design of a drone that fulfills all its top level requirements. This is followed by Section 9.2 which will present the requirements that need to be verified by the simulation model. Section 9.3 will then present the various performance analysis done on the drone. This will conclude the performance part of the subsystem which will then be followed by the simulation model. The simulation architecture and the results will be presented in Section 9.4. Section 9.5 will mention the risks associated with this subsystem and their mitigation strategies followed by verification methods for the simulation presented in Section 9.6. Lastly, Section 9.7 will present the compliance matrix for the requirements relevant to this subsystem.

9.1. Functional Recap

The Performance & Simulation subsystem analysis will focus on analysing the race performance of the drone and creation of a simulation model to verify the top level requirements for the drone. The performance part of this subsystem, unlike other subsystems, does not focus on a specific part or function of the drone but rather combines specific facets of analysis performed by each subsystem previously to measure the performance of the drone in a race setting. The aim of this analysis is to determine whether the individual analysis of each subsystem can be combined in a way that the drone is able to perform as required and complete the race lap as fast as possible. The simulation part of the subsystem focuses on the design of a simulation model that is able to verify all the top level requirements of the drone. Due to the limited time scope of the DSE, it is not possible produce the designed drone and hence not possible to verify if the drone will actually be able to perform as designed. Hence, a simulation has been created with the intention of virtually replicating the behavior of the drone in order to verify that, along with fulfilling its top level requirements, the drone is also controllable and gives potential pilots of the drone an opportunity to experience piloting the drone. This simulation is a graphic based simulation and was created using the Simulink software.

9.2. Requirements

This section presents the requirements that need to be verified by the Performance & Simulation subsystem. The relevant requirements have been presented in Table 9.1. As stated earlier, Performance & Simulation does not have its own subsystem requirements but rather serves to verify system level requirements. Most of the requirements will be verified through the simulation except requirement SYS-FLGT-MOVE-04 which will be verified through analysis.

Table 9.1: Performance and simulation requirements

Identifier	Requirement	Method
SYS-FLGT-MOVE-01	The drone shall have a maximum horizontal velocity in forward flight of at least 90 [m/s].	Simulation
SYS-FLGT-MOVE-03	The drone shall be able to accelerate from minimum to maximum speed in forward flight in 5 [s].	Simulation
SYS-FLGT-MOVE-04	The drone shall be able to decelerate from maximum to minimum speed in forward flight in 8 [s].	Analysis
SYS-FLGT-ROTA-01	The drone shall be able to perform a 25g turn in forward flight at a velocity of 60 [m/s].	Simulation
SYS-FLGT-ROTA-03	The drone shall be able to obtain a pitch rate of at least 1000 [°/s] from nominal forward flight.	Simulation

Identifier	Requirement	Method
SYS-FLGT-ROTA-04	The drone shall be able to achieve max pitch rate from nominal forward flight in 1 [s]	Simulation
SYS-FLGT-ROTA-05	The drone shall be able to obtain a yaw rate of at least 300 [°/s] from nominal forward flight.	Simulation
SYS-FLGT-ROTA-06	The drone shall be able to achieve max yaw rate from nominal forward flight in 5 [s]	Simulation
SYS-FLGT-ROTA-07	The drone shall be able to obtain a roll rate of at least 1080 [°/s] from nominal forward flight	Simulation
SYS-FLGT-ROTA-08	The drone shall be able to achieve max roll rate from nominal forward flight in 1 [s]	Simulation

9.3. Performance Analysis

This section will present the performance analysis of Peregrine. To do the performance analysis, it was necessary to determine an approximate flight profile for the drone such that analysis criteria specific to the flight profile could be selected. To determine the flight profile it was important to select a representative race track that a drone like Peregrine would race on. Selection of a race track would indicate exactly what manoeuvres the drone would need to do and also give an indication of the desired performance required to outperform other drones during the same lap.

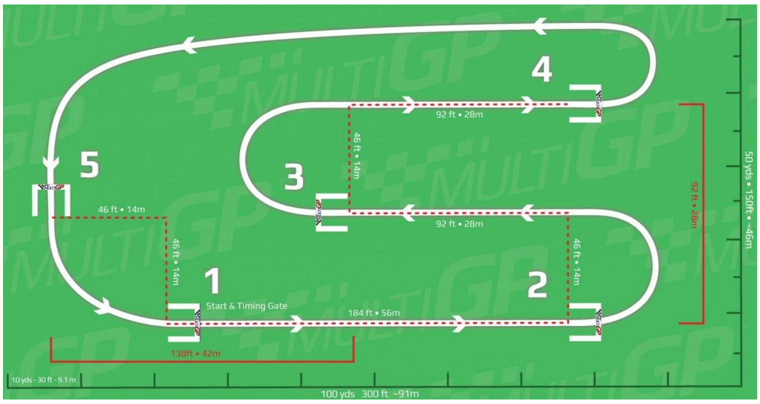


Figure 9.1: MultiGP universal time track 1[1]

The track selected for the drone is the MultiGP Universal Time Trial Track 1 and is presented in Figure 9.1. This track has 2 long straights of 56 [m] each and two shorter straights of 28 [m] each. The drone would need to perform three turns of 180 degrees with radius for each turn equal to 7 [m]. Lastly, it also needs to do two 90 degree turns at the end of the track of radius 14 [m]. The drone will start the race at gate 1 in the figure and has to follow the arrows until it finishes the race by crossing gate 1 again. This track is a relatively shorter track with the longest straight being just 56 [m]. The idea behind selection of this specific track is that Peregrine can display its excellent manoeuvrability, which is found to be lacking in flying wing drones, along with its fast speed. Peregrine being a flying wing drone will find it much easier to finish larger tracks with much longer straights as they will allow the drone to focus more on its speed performance rather than its agility. Thus, if Peregrine can beat conventional (quadrotor) drones on this track, it will also outperform them on larger tracks that are more suited to Peregrine.

To select the performance criteria to be analysed, an approximate flight profile for a section of the track, specifically the straight after gate 1 until the turn after gate 2, was constructed. As the same manoeuvre needs to be performed at other turns, analysing the flight profile for only this section of the track would give an indication of the performance throughout the track. It should be noted that even though the performance criteria to be analysed were selected from a flight profile at this specific race track, the performance analysis for the drone will be relevant for all tracks and flight conditions. The approximate flight profile of the drone constructed from this specific section of the track is as as follows:

- The drone will first accelerate in a straight line as fast as it can.

---

[1] `https://www.multigp.com/universal-time-trial-utt/` Accessed 16th June 2021

- The drone will then brake to prepare for the turn by reducing its speed.
- The drone will then roll and pitch to enter the turn.
- The drone will then finish the turn with a specific radius dependent on the amount of lift it generates.

From the constructed flight profile, four specific types of performance analysis were chosen to be performed. The first analysis determines how fast Peregrine can accelerate from a specific speed on a straight, displaying its straight line performance. The second analysis scopes the braking performance of the drone and analyses how much time it takes for the drone to brake from its maximum speed to stall speed. The third analysis evaluates the agility of the drone by analysing its pitch and roll rates at lower speeds, where it uses its vectored propellers to perform both angular motions. The final analysis evaluates the radius that the drone makes the turn with and whether the radius for the drone is small enough to be competitive on this specific track.

9.3.1. Straight Line Acceleration

The first of the performance analysis to be analysed is the straight line acceleration, as this is the first part of any race. To determine the straight line performance of Peregrine, the maximum thrust data was used from Power & Propulsion along with the drag characteristics found in Aerodynamics. It was also assumed that no thrust vectoring would take place, since this would limit the forward thrust of the drone and thus result in a slower acceleration. In Figure 9.2, the acceleration is plotted as a function of velocity, and in Figure 9.3, the velocity is plotted as a function of distance.

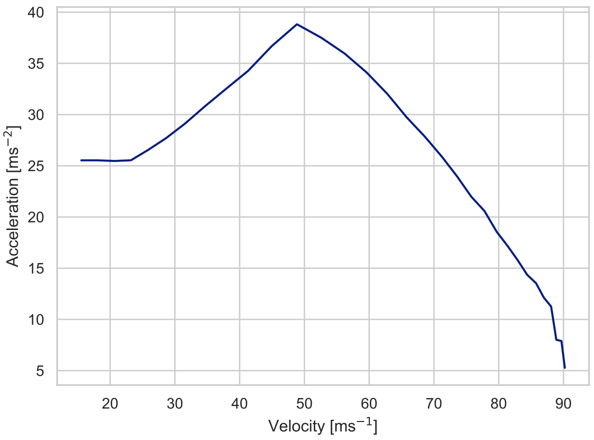


Figure 9.2: Acceleration as a function of velocity.

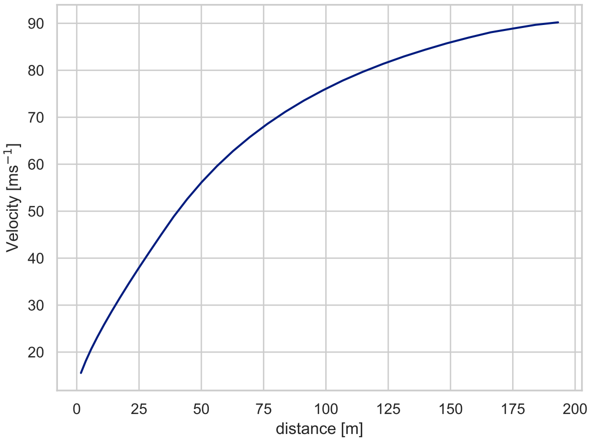


Figure 9.3: Maximum velocity that can be achieved after a certain distance.

The shape of the curve in Figure 9.2 will now be elaborated upon. From stall speed to 22 [m/s] (the flat part in the graph), the thrust data did not converge, so the acceleration was taken to be constant. However, this has a very minor effect on the results, since accelerating from stall speed to above 22 [m/s] takes less than 0.5 [s]. Moving rightwards in the graph, the acceleration begins to steadily increase until around 50 [m/s], which comes as a result of the fact that the propeller stalls at such low velocities so by increasing velocity the propellers slowly stop stalling and at 48 [m/s] the maximum acceleration is achieved. Beyond this point, the high velocity makes thrust generation more difficult as the propeller must accelerate the air more, so there begins to be a drop in acceleration until it reaches roughly 5 $[m/s^2]$ at the maximum velocity of 90 [m/s]. In the left graph, Figure 9.3, this effect of decreasing accelerations towards high velocities can be clearly seen as it takes only 55 [m] to accelerate to 60 [m/s], and then a further 140 [m] to reach 90 [m/s]. Overall, it can be seen that it takes Peregrine 195 [m] to reach its maximum velocity of 90 [m/s] when starting from stall speed, and this is achieved in approximately 3.2 [s]. These results mean that on the chosen track (Figure 9.1) Peregrine will be unable to unleash its full potential given the short straights, but when Peregrine races on larger tracks, this straight line performance is excellent.

9.3.2. Braking Performance

After accelerating to such high velocities, it is necessary to slow down to be able to make a corner. Given the aerodynamic shape of Peregrine, it does not have very high drag levels so decelerating using purely the aerodynamic drag would not yield satisfactory performance. Therefore, it was chosen to also use reverse thrust to help with the braking, since this was easily possible with the chosen motors. To construct the braking performance graphs, the sum of the aerodynamic drag and the reverse thrust were taken to be forces acting in the negative x direction and then Newton's second law was used to compute the deceleration. The results are shown in Figure 9.4 and Figure 9.5. The following graphs show the time and distance required to decelerate to stall speed from top speed for the drone after maximum reverse thrust is achieved. The actual time it takes to actuate reverse thrust is still unknown at this point but is expected to be quite small or negligible as visible in other drones that use reverse thrust.[2]

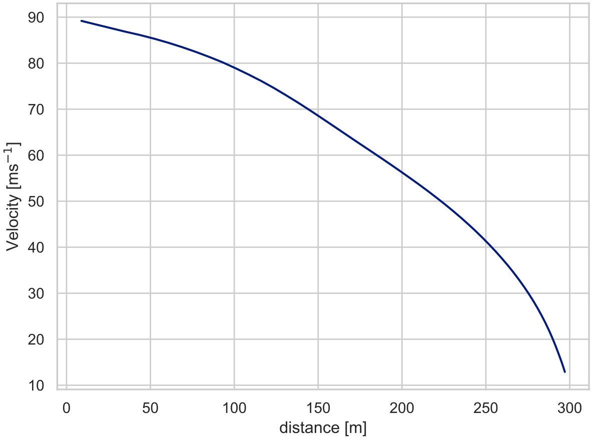


Figure 9.4: Distance needed to brake from a certain velocity.

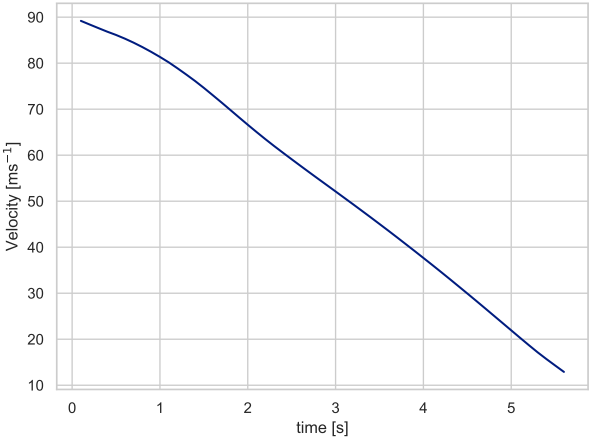


Figure 9.5: Time taken to decelrate from a certain velocity.

In Figure 9.4, it can be seen that it takes approximately 300 [m] to brake from 90 [m/s] to stall speed. Perhaps more importantly, given that the optimal turning performance is at 60 [m/s], it takes roughly 180 [m] to brake from 90 [m/s] to 60 [m/s]. Notice that compared to straight line acceleration, braking takes significantly longer, which is due to the fact that the propellers are optimised for forward flight and thus do not produce as much reverse thrust as they do forward thrust. Looking at Figure 9.5 it can be seen that braking from 90 [m/s] to stall speed takes roughly 5.6 [s], and braking from 90 [m/s] to 60 [m/s] takes roughly 3.1 [s]. Thus, requirement SYS-FLGT-MOVE-04, which states that Peregrine must be able to brake from 90 [m/s] to stall speed within 8 [s], is comfortably met. Like with straight line acceleration however, this braking performance does not allow Peregrine to brake in time on the chosen track (Figure 9.1), though since it cannot reach its top speed in the first place, it is not a problem. When Peregrine races on larger tracks with longer straights, this braking performance is excellent.

9.3.3. Attitude Rates

Next, it is important to look at the pitch and roll rates because they determine how quickly a turn can be entered into. For this, the maximum pitch and roll rates were found as a function of velocity. First it must be noted that in Chapter 8, the elevons were sized to achieve a roll rate of 1080 [°/s] at 45 [m/s] and a pitch rate of at least 1000 [°/s] at 45[m/s], so beyond 45 [m/s] the pitch and roll rates will remain constant at these values. Thus, the rates will only be plotted from stall speed until 45 [m/s]. Below 45 [m/s] thrust vectoring will be used along with the elevons to get better performance. However, since the engine's cannot vector to 90°, there would also be a component in the horizontal direction which would accelerate the drone, and given the high accelerations, 45 [m/s] would quickly be reached. Thus, only the thrust that maintains the horizontal velocity will be used to construct the graphs, because it could be that the pilot would like the help of vectoring engines to roll or pitch, but does not want to speed up. It must also be noted that the plots are produced for rolling or pitching individually by deflecting the elevon to its maximum deflection either as an aileron or elevator. The plot for maximum roll rate is shown in Figure 9.6 and the plot for maximum pitch rate is shown in Figure 9.7.

As can be seen in Figure 9.6 and Figure 9.7, both rates decline as the velocity decreases from 45 [m/s] at a reasonable rate, but then at 27 [m/s] there is a very large jump after which the roll rate and pitch rate both become very small. This large drop in performance at 27 [m/s] comes as a result of the fact that deflecting the elevons to their maximum deflection increases the effective angle of attack by 4.9°and thus would lead to a stall when flying slower than 27 [m/s]. [56] As a result, it can be concluded that, unless minimal performance is required, rolling and pitching should only be done above 27 [m/s] and preferably above 45 [m/s] where

---

[2] https://www.youtube.com/watch?v=TznnQCz6YvI

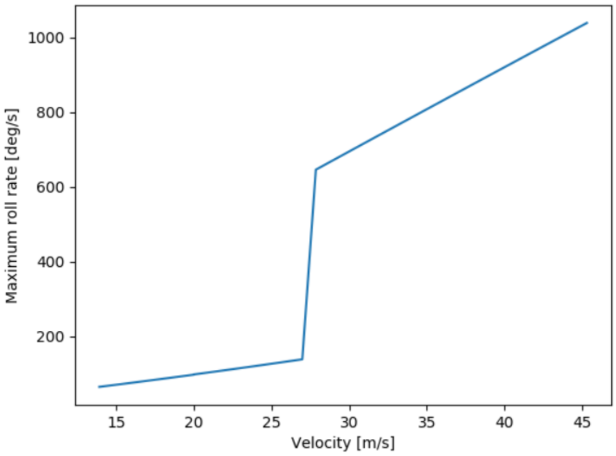


Figure 9.6: Maximum roll rate as a function of velocity.

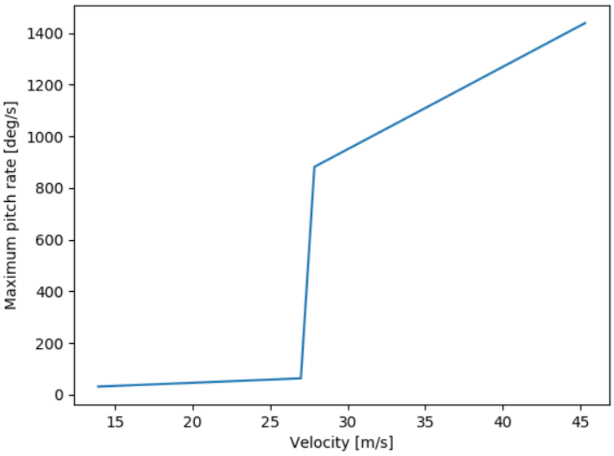


Figure 9.7: Maximum pitch rate as a function of velocity.

the performance is at its maximum. However, it must be remembered that the roll and pitch rates can still be slightly higher by using maximum thrust with the vectored propellers, though this would come at the cost of speeding up, at which point it is likely better to simply speed up in a straight line and then roll or pitch as desired using the elevons.

9.3.4. Minimum Turn Radius

Next the minimum turn radius as a function of velocity will be looked at, which is important since if a tight turn is coming up it is beneficial to know apriori what velocity will be required to make the turn. In order to construct the minimum turn radius, Equation 9.1 was used.

$$R = \frac{V^2}{g\sqrt{n^2 - 1}} \tag{9.1}$$

With

$$n = \begin{cases} \frac{L}{W} = \frac{C_{L_{max}} \frac{1}{2}\rho V^2 S + 2T_{o_{max}} sin(45)}{mg} & 13[m/s] < V < 45[m/s], \\ \frac{L}{W} = \frac{C_{L_{max}} \frac{1}{2}\rho V^2 s}{mg} & 45[m/s] < V < 60[m/s], \\ 25 & V > 60[m/s] \end{cases}$$

Where $T_{o_{max}}$ is the maximum thrust of each outboard engine. The reasoning for a different definition of the load factor n is as follows. At low speeds, from stall speed to 45 [m/s], thrust vectoring can be used to help generate lift, which is why the vertical component of the thrust is considered in addition to the aerodynamic lift from the wings. Beyond 45 [m/s] no thrust vectoring is required so the lift returns to being just the aerodynamic lift. However, beyond 60 [m/s], due to the high velocity, the wings are able to generate sufficient lift to make turns tighter than 25g which would introduce loads beyond the structures capabilities so the load factor is capped at 25. For calculation of the maximum C_L, data was taken from the Aerodynamic analysis (Chapter 5). The minimum turn radius as a function of velocity is shown in Figure 9.8.

There are several interesting features in Figure 9.8 which will be discussed. Firstly, starting from stall speed and moving rightwards in the figure, it can be seen that initially the turn radius gets worse and then at around 25 [m/s] it begins to get better again. This is because at such low velocities, the maximum thrust stays constant since the propellers are designed for 60 [m/s] and as velocity increases the minimum turn radius also increases as seen in Equation 9.1. However between 25 [m/s] and 45 [m/s] the maximum thrust does begin to increase which again brings the minimum turn radius back down. Once the velocity reaches 45 [m/s] and there is no more thrust vectoring, the minimum turn radius jumps by approximately 1 [m] though it begins to come back down as the aerodynamic lift increases, until 60 [m/s]. Beyond 60 [m/s], as explained previously, the limiting factor is the structural loads as opposed to the available lift, so the turn radius rapidly increases with the square of velocity.

As a point of comparison, not using vectoring propellers will result in the minimum turn radius shown in Figure 9.9.

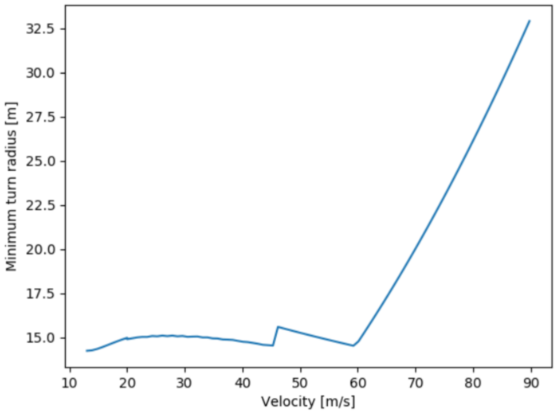


Figure 9.8: Minimum turn radius as a function of velocity from stall speed to 90 [m/s].

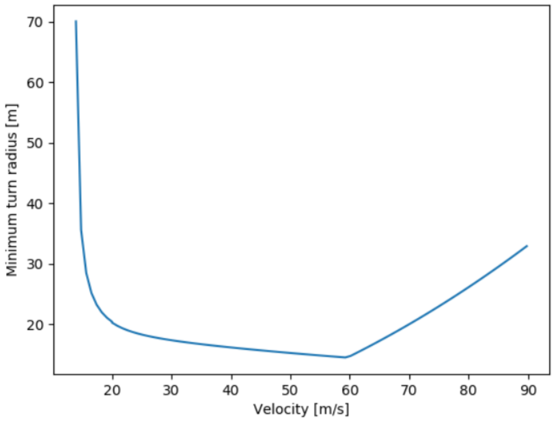


Figure 9.9: Minimum turn radius as a function of velocity when no thrust vectoring is used.

As can be seen in Figure 9.9, the minimum turn radius drastically increases when dropping below 25 [m/s], thus justifying the use of vectoring propellers to aid with lift generation.

Overall, by being able to use the vectoring propellers, the minimum turn radius may be kept roughly constant with velocity until 60 [m/s], only fluctuating between 14.28 [m] and 15.58 [m]. However, beyond 60 [m/s] it does shoot up, indicating that turns should not be made beyond 60 [m/s] unless they have a very large radius. In relation to the chosen track, these results mean that Peregrine will be unable to follow the racing line shown in Figure 9.1 since the 180°turns have a radius of only 7 [m]. Therefore, Peregrine will have to overshoot the gates, making almost a 270°before a 90°to make the gates, though given the high speeds that Peregrine can travel at, this will still result in a quicker time than a conventional drone.

9.4. Simulation

The simulation part of this subsystem corresponds to design of a simulation model for verification of top level requirements. Subsection 9.4.1 will explain the architecture of this simulation in Simulink and the methodology for its design. This will be followed by simulation results which aim to verify and validate the top level requirements of the drone.

9.4.1. Simulation Architecture

The simulation model as mentioned earlier was constructed using Simulink. The simulation was made from separate blocks, where each block is responsible for fulfilling a certain function in the simulation. To create a video based simulation model that could verify and validate the requirements for the drone, it was required to simulate the actual motion of the drone through use of its equations of motions. It was also required to simulate the graphics and the present the chosen race track in virtual world and also control the drone through the joystick such that drone pilots can experience controlling the drone. Each of these function were coded inside separate blocks which were then connected together to form the complete simulation model. This section will present the overview of the whole simulation model before diving into each block and presenting the logic and method for each block.

Overview of the Simulation Model

The top level diagram of the simulation model in Simulink has been displayed in Figure 9.10. The simulation begins by receiving inputs from the pilot which are normally administered through a joystick. The PID controller then takes these pilot commands as inputs, and converts them into inputs for the dynamics model. The working of the PID controller has been explained in the Control & Stability chapter. The outputs of the PID controller are input into the dynamics model directly except the left elevon and right elevon deflections. These deflections are converted into aileron and elevator deflections through a series of operations. Note that there is only one set of elevons on the drone and the aileron and elevator deflections are computed by calculating which deflection would cause roll and which deflection would cause pitch, the deflection causing roll corresponds to aileron deflection and and the deflection causing pitch corresponds to elevator deflections. This conversion is done as the dynamics model will not take left elevon and right elevon deflections as inputs, as no equations can estimate the effect of just deflection of one of the elevons on the whole system. As a result the elevon deflections are converted into aileron and elevator deflection to input into the dynamics model. To convert right and left elevon deflections into aileron and elevator deflections, it is assumed that the ailerons always deflect equally in opposite directions. That means that if the left aileron deflects positively, the right aileron will deflect negatively by the same magnitude. The elevators are always assumed to deflect in the same direction that means that both the right and left elevator always deflect in the same direction by the same magnitude. To compute aileron and elevator deflections from right and left elevon deflections, the elevon deflections are subtracted from each other and the difference is then divided by two. This gives the aileron deflection. To compute the elevator deflection, the calculated difference is added to the elevon with the smaller deflection while it is subtracted from the elevon with the larger deflection. For example, if the PID controller outputs a deflection of 12°for the left elevon and 18°for the right elevon, their difference is 6°which when divided by two is 3°. Hence, 3°is the aileron deflection. When added and subtracted from the left and right elevon respectively, the elevator deflection is found to be 15°. Hence both elevons were deflected 15°to induce pitch after which a 3°aileron deflection was input to induce a roll.

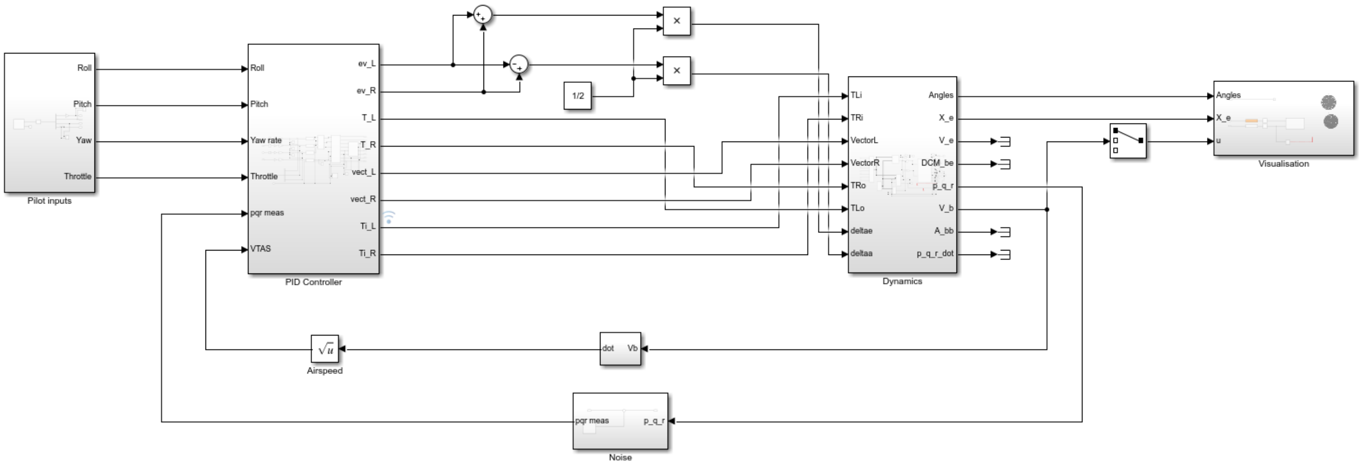


Figure 9.10: Top level diagram of the simulation.

The inputs received by the dynamics model are converted into new states for the drone. The working of the dynamics model will be discussed more in detail in following sections. The outputs of the dynamics model are then fed back into the noise block which adds noise to the measured state to represent use of a real sensor and then these outputs are fed into the PID controller, for the controller to measure between the new state and pilot's desired state. The outputs of the dynamics model are also fed into the visualisation block which initiates animation for the video based simulation and also transforms the co-ordinates of the simulation to that of the virtual world. The working of this block will also be discussed in detail later.

Now that the general overview of the simulation has been discussed, the following sections will dive deeper into some of the blocks of the simulation.

Pilot Inputs Block

The pilot inputs block is set up in a way to receive inputs from a controller and translate them into inputs for the simulation. The overview of the pilot inputs block has been displayed in Figure 9.11

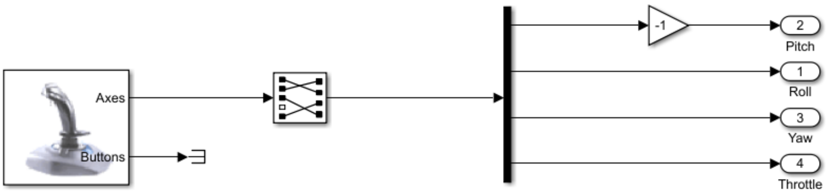


Figure 9.11: The pilot inputs block in Simulink.

A joystick input block is used from the Simulink library and used to sense the joystick inputs. This block is then connected to a selector bus to select the relevant inputs from the joystick. These inputs are then designated to each of the inputs for the PID controller which are pitch, roll, yaw and throttle. The selector also allows designation of each controller stick to its desired output. For example, left stick up and down can be designated to throttle using the selector block as it is for a typical drone. It should be noted that a gain of negative one is multiplied to the pitch input such that a deflection right stick down corresponds to a pitch up motion of the drone.

Dynamics Block

The dynamics block takes inputs from the PID controller and outputs the new states of the drone. Essentially it is responsible for determining how the drone would react to inputs on its control surfaces and plans its motion. The dynamics block for this purpose contains a number of equations, and hence contains various sub-blocks inside it that correspond to different contributions of the equations of motions. The overview of the dynamics block has been displayed in Figure 9.12. The basic layout of this block was inspired from [42].

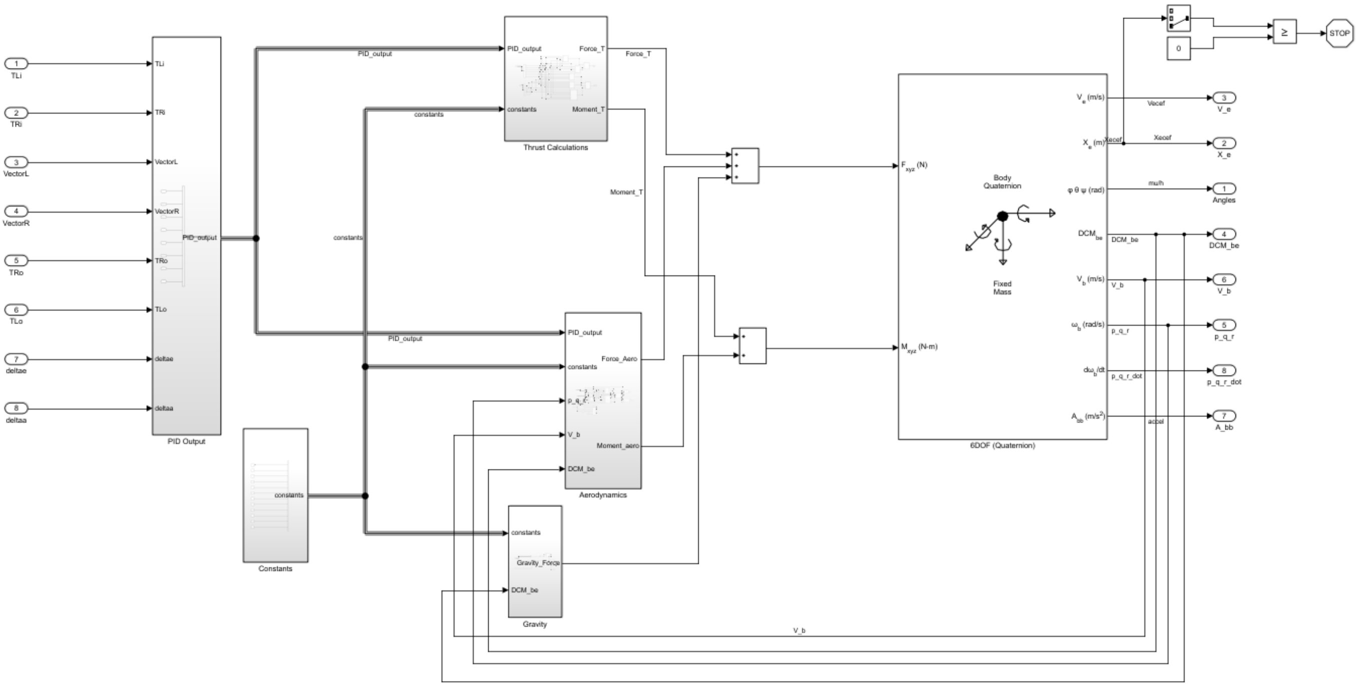


Figure 9.12: The dynamics block in Simulink.

To create the dynamics model, two primary assumptions were made:

- Aerodynamic coefficients do not change with velocity
- The drone flies at an altitude of 0.75 [m], hence the atmospheric properties are taken at sea level ISA.

The dynamics model uses a 6DOF quaternion block from the Simulink library as its engine. This block takes the forces and moments in all three axes and gives a number of outputs such as the drone attitude, attitude rate, body velocity, body acceleration and position co-ordinates of the drone. It was chosen to use a quaternion block as it uses a non linear system of equations which makes the simulation more accurate than use of a linear system which would also have many more assumptions. The quaternion block also reduces the workload of the group because if a linear state space model would have been used, the equations of motion would have needed to undergo a series of transformation to reach a point where they can be converted into a state space model, this would have increased the possibility of making an error in the simulation. The quaternion block already performs these transformations, hence the group is not required to do such transformations. Lastly, the quaternion block from Simulink has already been verified and validated and is used in a number of simulation today, hence it can be entrusted with providing correct results.

To use the quaternion block, the forces and moments around all three axes need to be fed into the block. To calculate these force and moment, outputs from the PID controller along with some constants are used. The constant block shown in the figure defines these constants which includes aspect ratio, chord length, span, surface area, center of gravity of the drone, distance from c.g. to inboard and outboard propellers and atmospheric density.

The outputs of the PID controller and the constants are then fed into the aerodynamics, thruster and gravity block. Each of these blocks corresponds to the contribution of aerodynamics, propellers and gravity to forces and moments. Each of these blocks will now be discussed in further detail.

Aerodynamics Block
The overview of the aerodynamics block has been presented in Figure 9.13. The constants and PID controller outputs supplied to block are input into force and moment coefficient blocks as shown in the figure. These blocks calculate the lift drag and side force coefficients and the pitch, roll and yaw moment coefficients required for determining force and moments due to aerodynamics forces. The equation used for calculating these have been presented in Equation 9.14 - Equation 9.19. Within these blocks there are functions which input parameters such as angle of attack, sideslip, attitude rates and the stability coefficients and output the desired coefficients. It should be noted that the coefficients used in these blocks were calculated and verified by Aerodynamics. These calculated coefficients are then dimensionalised in the next block and converted into a 3x1 vector using the function blocks displayed in the figure. The aerodynamics block outputs moments and forces acting on the drone due to aerodynamics.

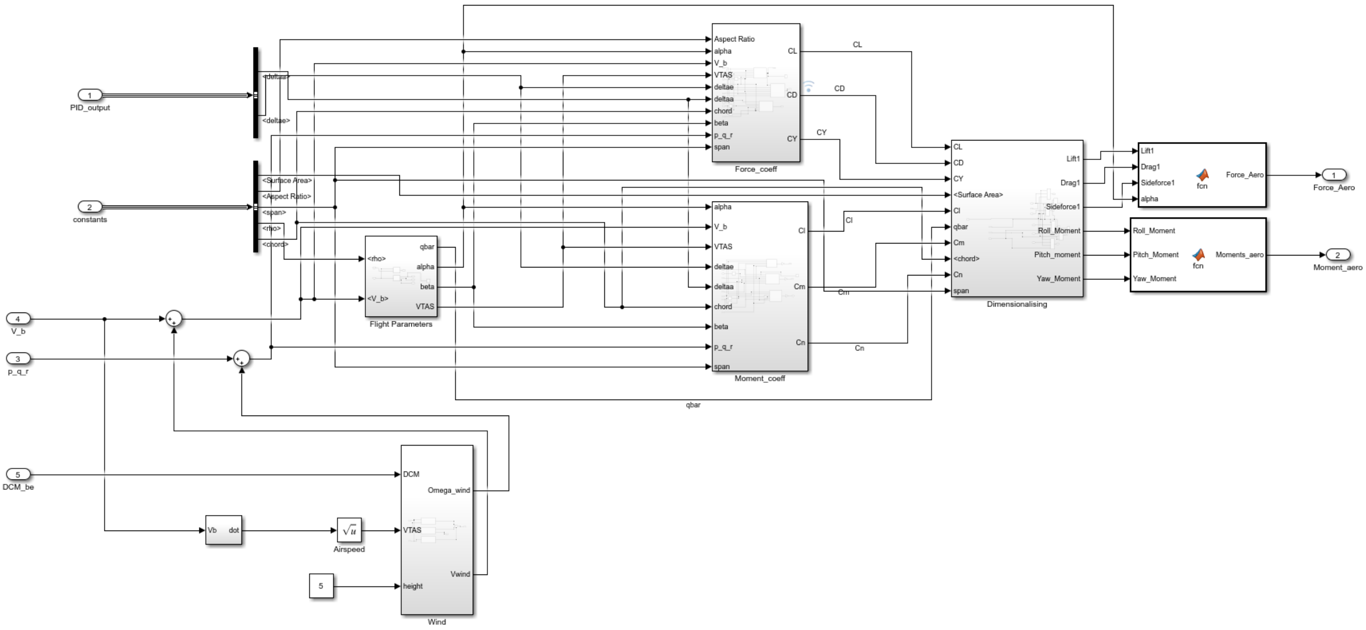


Figure 9.13: The aerodynamics block in Simulink.

To calculate parameters such as angle of attack, sideslip angle, true airspeed which are required for computation of aerodynamic coefficients, the flight parameters block from the Simulink library was used. This block inputs the air density and body velocity in all three axes and performs a number of operation to outputs dynamic pressure, angle of attack, angle of sideslip and true airspeed which are continuously changing. These parameters are then fed into coefficient block.

The aerodynamics block also contains a wind turbulence block. The purpose of this block within the simulation is to replicate a real outdoor drone race where the drone would be subject to wind turbulence. This wind disturbance would affect the performance of the drone, hence to make the simulation as realistic as possible this wind turbulence model was created. The wind turbulence block within the simulation has been displayed in Figure 9.14.

The wind turbulence block uses three blocks from Simulink library which are namely shear wind model block, discrete gust block and dryden wind turbulence model block. These blocks input the velocity of the drone, the direction cosine matrix and altitude at which the drone flies and output velocity of wind while the dryden turbulence model also outputs angular velocity of wind. To incorporate the velocities output the turbulence model in the simulation, the calculated wind velocities are added to nominal velocity and attitude rates of the drone which are then fed to the coefficient blocks.

Thrust Block
The thrust block of the simulation calculates the contribution of the propellers to the forces and moments acting on the drone. It inputs the PID controller outputs and constants such as center of gravity and the distance of inboard and outboard propellers to the center of gravity. Using these parameters and equations

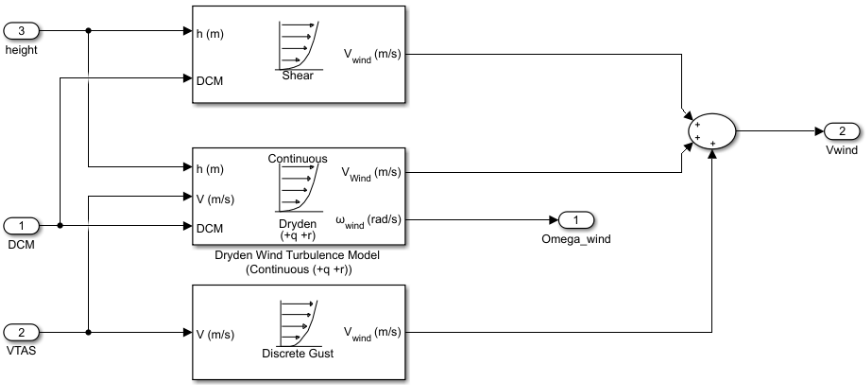


Figure 9.14: The wind turbulence block in Simulink.

of motions presented in 9.2 - 9.7, it is able to calculate the force and moment contribution of the propellers to the drone. It should be noted from the equations of motion, the thrust block only computes the value of propeller contributions, the aerodynamics forces and moments are calculated by the aerodynamics block. The inputs to the block are fed into specific functions which correspond to forces and moments in each of the three axes. The outputs of these functions are then combined into vectors and output from the thrust block.

Gravity block
The gravity block is responsible for calculating the force acting on the drone due to gravity. The block takes the input of mass of the drone and converts it into weight through multiplication with gravity constant. The gravity force is then transformed through multiplication with the direction cosine matrix of the body axis of the drone which is fed into the gravity block from the quaternion block. The gravity block then outputs the gravity force on the drone in the body axis.

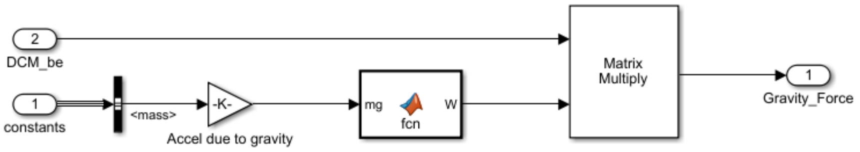


Figure 9.15: The gravity block in Simulink.

Once, contribution to forces and moments from propellers, aerodynamics and gravity has been computed, they are added together using summation blocks and fed into the quaternion block which then output the new state of the drone.

Noise Block
The purpose of the noise block is to replicate use of an actual IMU and hence replicate noise typical in readings of actual sensors. To approximate the noise acting on the IMU, the noise density of the XSens IMU[3] was used. This density was then multiplied by the IMU's bandwidth to give noise generated by the IMU per unit of angular velocity. As this IMU is rather high end and expensive, a safety factor of 5 was assumed to replicate a low end IMU. Through a random number generator block from the Simulink library, the calculated noise was added into the attitude rates output from the dynamics block and the attitude rates combined with noise are then output from the block. This measured angular rate is then fed back into the PID controller. The overview of the noise block is displayed in Figure 9.16.

Visualisation Block
The visualisation block is responsible for the animation of the simulation. The overview of the visualisation block is presented in Figure 9.17. To animate the movement of the drone it was important a virtual world be created where the drone would fly. This virtual world was created using VRealm and the Simulink 3D editor software. This virtual world would include the track that was selected for construction of the flight profile of

---

[3] `https://www.xsens.com/mti-670?utm_term=xsens%20mti%20670&utm_medium=ppc&utm_campaign=NWEC+%7C+Search+%7C+Brand&utm_source=adwords&hsa_cam=11538702436&hsa_src=g&hsa_mt=b&hsa_ver=3&hsa_net=adwords&hsa_tgt=kwd-1079740514638&hsa_acc=1306794700&hsa_grp=115070671173&hsa_kw=xsens%20mti%20670&hsa_ad=494404389338&gclid=CjwKCAjwzruGBhBAEiwAUqMR8JwtpoT3_cGJRW1T7IFwfkEW_dOUYtE_Sk5cvnn69WxSX1CNFq1k-xoCr2QQAvD_BwE` Accessed 21st June 2021

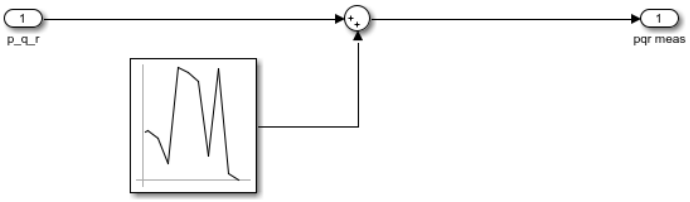


Figure 9.16: Noise block

the drone. The purpose of creating the track inside the virtual world was to define a specific area for the drone to fly while also attempting to break the record for the fastest lap on this specific track within the simulation to validate the requirement of being the fastest racing drone in the world. The two flight instruments shown in the figure, measure the velocity and altitude of the drone in the simulation for convenience of the pilot.

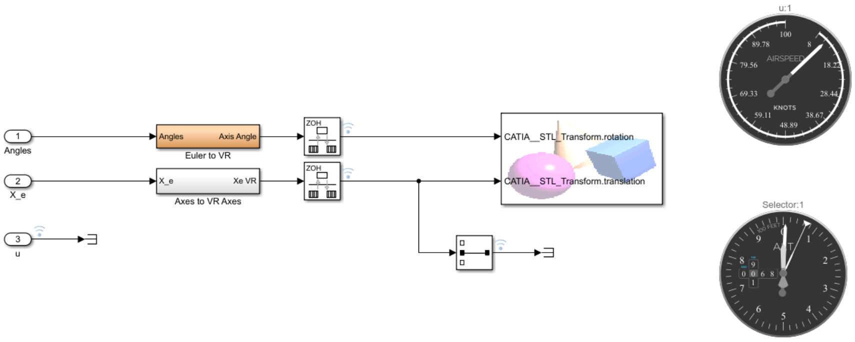


Figure 9.17: The visualisation block in Simulink.

It should be noted that the axes system of the simulation in Simulink and the axes system of the virtual world are different. Hence, it was required to transform the axes system of the simulation to that of the virtual world. This was done using the Euler to VR, and Axes to VR Axes block as shown in the figure. The inputs for this block were generated from the outputs of the dynamics block. These transformed co-ordinates were then fed into the VR sink block which contains the virtual world. The virtual world within this block contains the selected race track as shown in Figure 9.18 and the drone to be controlled. For the convenience of the pilot who is piloting the drone in the simulation, two viewpoints were created. The first one was a third person view of the drone which is shown in Figure 9.19 from its back while the other view is the first person view from the location of the camera installed on the drone which is more typical for racing pilots.

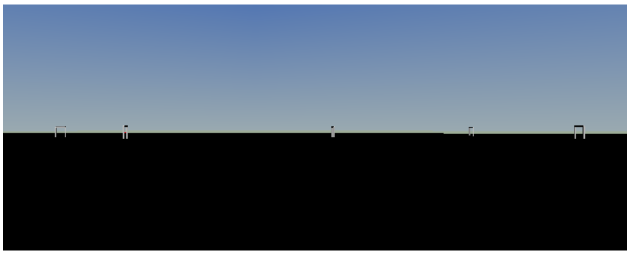


Figure 9.18: View of the track inside the virtual world.

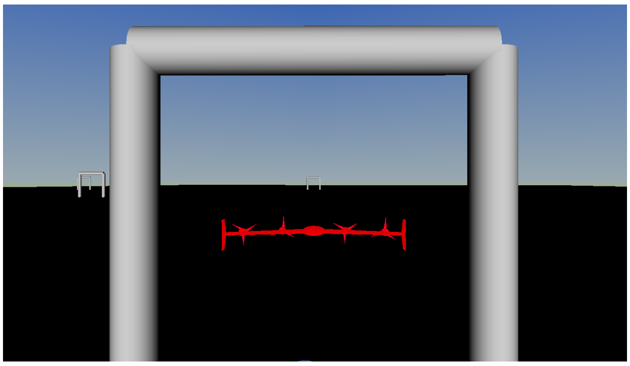


Figure 9.19: Drone inside the simulation with 3rd person drone view point

9.4.2. Derivation of Equations of Motion

As described in Subsection 9.4.1, the 6DOF quaternion block was used for the simulation in Simulink, which requires the sum of forces in x, y and z directions as well as the sum of moments in the same three axis around the center of gravity. In order to construct these equations, literature was consulted where it was found that three forces should be considered (along with the corresponding moments they produce), namely thrust forces, aerodynamic forces, and the gravity force. [42] First, the thrust forces will be considered. To ensure that no errors are made, and to present the (right handed) coordinate system that will be used, the free body diagram in Figure 9.20, shows all of the relevant thrust forces along with the relevant distances required for the moment equations. Note that in the figure, the outboard engines are not physically vectored, but since they are able to vector, the thrust they produce has been shown to act at an angle θ_L and θ_R for the left and right engines respectively. The distance labelled l_s is the servo length about which the outboard engines rotate, which will create a vertical offset that will be accounted for in the moment equations. Also note that the drone is symmetric, so the distance x_{out} shown for the right outboard engine is the same distance that applies for the left outboard engine.

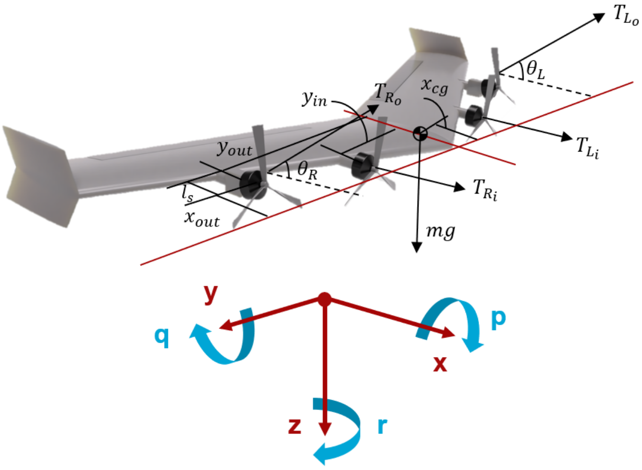


Figure 9.20: Free body diagram (showing all thrust forces and relevant distances) used for the derivation of the equations of motion. The red lines indicate the datum that the distances are calculated from, namely the leading edge of the root chord (for x direction) and the exact middle spanwise location (for y direction).

Next the aerodynamic forces will be considered. These are simply the aerodynamic forces and the corresponding moments around each axis, namely the drag force D, lift force L, side force Y, rolling moment l, pitching moment m and yawing moment n. Finally the gravity force must be considered, which simply acts in the positive z direction and has a magnitude mg. Since it acts at the center of gravity, it produces no moment about the center of gravity. In Equation 9.2-9.7 the full equations of motion are presented.

$$\sum F_x = T_{L_i} + T_{R_i} + T_{L_o} cos\theta_L + T_{R_o} cos\theta_R - D \tag{9.2}$$

$$\sum F_y = Y \tag{9.3}$$

$$\sum F_z = mg - T_{L_o} sin\theta_L - T_{R_o} sin\theta_R - L \tag{9.4}$$

$$\sum M_x = l + T_{L_o} sin\theta_L \cdot y_{out} - T_{R_o} sin\theta_R \cdot y_{out} \tag{9.5}$$

$$\sum M_y = m - T_{L_o} sin\theta_L \cdot (x_{out} - x_{cg}) - T_{R_o} sin\theta_R \cdot (x_{out} - x_{cg}) - T_{R_o} cos\theta_R \cdot l_s sin\theta_R - T_{L_o} cos\theta_L \cdot l_s sin\theta_L \tag{9.6}$$

$$\sum M_z = n + T_{L_i} \cdot y_{in} + T_{L_o} cos\theta_L \cdot y_{out} - T_{R_i} \cdot y_{in} + T_{R_o} cos\theta_R \cdot y_{out} \tag{9.7}$$

Each of the aerodynamic forces and moments can be further resolved into the following.

$$D = C_D \frac{1}{2} \rho V^2 S \tag{9.8}$$

$$L = C_L \frac{1}{2} \rho V^2 S \tag{9.9}$$

$$Y = C_Y \frac{1}{2} \rho V^2 S \tag{9.10}$$

$$l = C_l \frac{1}{2} \rho V^2 S b \tag{9.11}$$

$$m = C_m \frac{1}{2} \rho V^2 S c \tag{9.12}$$

$$n = C_n \frac{1}{2} \rho V^2 S b \tag{9.13}$$

And further, each of the non dimensional aerodynamic force and moment coefficients can be expanded into the following.

$$C_D = C_{D_0} + \frac{C_L^2}{\pi A e} + C_{D_{\delta_e}}\delta_e \tag{9.14}$$

$$C_L = C_{L_0} + C_{L_\alpha}\alpha + C_{L_{\delta_e}}\delta_e + C_{L_q} q \frac{c}{2V_\infty} \tag{9.15}$$

$$C_Y = C_{Y_\beta}\beta + C_{Y_{\delta_a}}\delta_a + (C_{Y_p} p + C_{Y_r} r)\frac{b}{2V_\infty} \tag{9.16}$$

$$C_l = C_{l_\beta}\beta + C_{l_{\delta_a}}\delta_a + (C_{l_p} p + C_{l_r} r)\frac{b}{2V_\infty} \tag{9.17}$$

$$C_m = C_{m_0} + C_{m_\alpha}\alpha + C_{m_{\delta_e}}\delta_e + C_{m_q} q \frac{c}{2V_\infty} \tag{9.18}$$

$$C_n = C_{n_\beta}\beta + C_{n_{\delta_a}}\delta_a + (C_{n_p} p + C_{n_r} r)\frac{b}{2V_\infty} \tag{9.19}$$

All of these aerodynamic coefficients come from the Aerodynamic analysis (Chapter 5), and are presented in Table 9.2. These coefficients were calculated for an angle of attack of 10°and velocity of 60 [m/s].

Table 9.2: All of the aerodynamic coefficients used for the equations of motion.

Parameter	Value	Unit	Parameter	Value	Unit
C_{D_0}	0.00728	[-]	$C_{D_{\delta_e}}$	0.000151	[1/deg]
C_{L_0}	0	[-]	C_{L_α}	4.236787	[1/rad]
$C_{L_{\delta_e}}$	0.032126	[1/deg]	C_{L_q}	4.369381	[s/rad]
C_{Y_β}	-0.172530	[1/rad]	$C_{Y_{\delta_a}}$	-0.001620	[1/deg]
C_{Y_p}	-0.165313	[s/rad]	C_{Y_r}	0.066699	[s/rad]
C_{l_β}	-0.071527	[1/rad]	$C_{l_{\delta_a}}$	-0.007715	[1/deg]
C_{l_p}	-0.477169	[s/rad]	C_{l_r}	0.034607	[s/rad]
C_{m_0}	0	[-]	C_{m_α}	-0.211840	[1/rad]
$C_{m_{\delta_e}}$	-0.009126	[1/deg]	C_{m_q}	-1.131105	[s/rad]
C_{n_β}	0.019222	[1/rad]	$C_{n_{\delta_a}}$	0.000164	[1/deg]
C_{n_p}	0.015213	[s/rad]	C_{n_r}	-0.008282	[s/rad]

Now, the equations of motion are complete and can be inserted into the 6DOF quaternion block. Combined with the rest of the simulation architecture outlined in Subsection 9.4.1, the simulation is now ready to run, and the results will be presented in Subsection 9.4.3.

9.4.3. Simulation Results

Now that the simulation architecture and the equations of motions used in the simulation have been described, the verification of top level requirements for the drone can be performed. The requirements that the simulation needs to verify have been presented in Section 9.2. The following sections will now verify each of the mentioned requirements and also describe briefly the verification process.

Top Speed and Acceleration

Requirement SYS-FLGT-MOVE-01 corresponds to the top speed of the drone being 90 [m/s] while requirement SYS-FLGT-MOVE-03 corresponds to the drone being able to reach this speed within 5 [s] from stall speed. Both these requirements can be verified together by giving a maximum step input on throttle and scoping the velocity against time. If the velocity of the drone reaches 90 [m/s] within 5 [s] then both the requirements are said to be verified.

As visible from the graph in Figure 9.21, when the drone in the simulation is subject to an input of maximum throttle, the velocity of the drone in the X axis reaches 90 [m/s] within 3 [s]. This verifies both the top level requirements for the drone of reaching 90 [m/s] and also reaching top speed within 5 [s]. As visible from the graph, the drone reaches top speed in 3 [s] which is consistent with the calculation of the Power and Propulsion department.

25g turn at 60 [m/s]
Requirement SYS-FLGT-ROTA-01 corresponds to the drone having the capability to perform a 25g turn when it enters the turn at 60 [m/s]. To perform a turn of 25g's it is required that the load factor of the drone is 25. This means that the lift generated by the drone is 25 times that off its weight. Assuming the mass of the drone is 500 [g], the weight of the drone is then 4.905 [N]. Hence, the drone will have a load factor of 25 if the total lift force generated by the drone is 25 times its weight which is 122.625 [N]. To verify this requirement, the drone inside the simulation will be given an initial velocity of 60ms with its throttle adjusted in a way that it flies at constant velocity. The drone will then be made to pitch and roll such that it turn and the lift on the drone will be measured. The result of this manoeuvre in the simulation is presented in Figure 9.22.

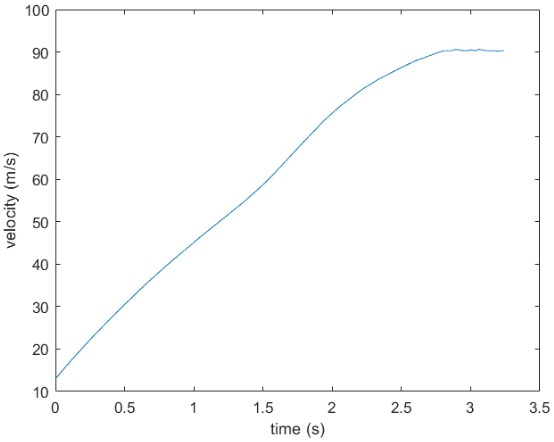


Figure 9.21: Velocity against time to verify the top speed and acceleration requirement.

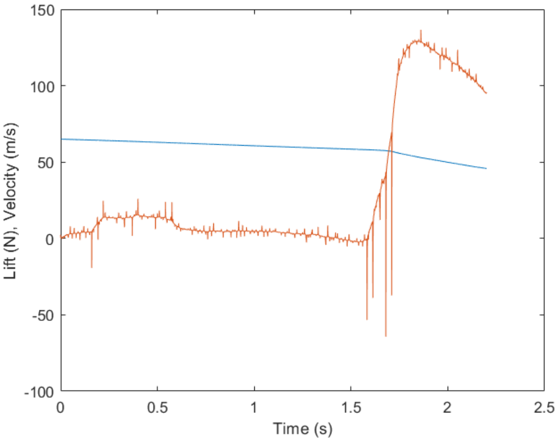


Figure 9.22: Velocity and lift against time to verify the requirement for turning at 25g's. Lift force is shown by the orange line while velocity is shown by the blue line.

In Figure 9.22, the red line is represented by u which denotes the velocity of the drone in the x direction. As can be seen from the graph, the value of this velocity remains almost always constant at 60 [m/s] up until the turn is initiated at around 1.6 [s] after which it decreases slightly. The blue line in the graph represents the lift force generated by the drone. the lift force remains very small until 1.6 [s] as the angle of attack of the drone is almost zero. At 1.6 [s] when the turn is initiated the lift generated by the drone starts increasing. After around 0.75 [s] of initiation of the turn the lift generated by the drone peaks with a value 125 [N]. This value is greater than the required lift 0f 122.625 [N] to generate a load factor 25. Hence, this requirement of the drone performing a 25g turn with velocity of 60 [m/s] can be verified.

Pitch Rate
Requirement SYS-FLGT-ROTA-03 states that the drone should be able to achieve a pitch rate of 1080 [°/s] or 18.85 [rad/s]. Requirement SYS-FLGT-ROTA-04 states that the drone should be able to achieve this within one second of initiating pitch. In the simulation, the drone is given a step pitch input and its pitch rate is scoped to measure its response.

As observed from Figure 9.23, the pitch rate of the drone is zero up until 1 [s] where the pitch is initiated. The deflection of the elevon to induce pitch is gradually increased over the next second and as visible from the graph, the drone achieves a pitch rate of around 20 [rad/s] at 1.7 [s]. This verifies both the requirements as a pitch rate of 20 [rad/s] is more than the required rate of 18 [rad/s] and as this pitch rate is obtained at 1.7 [s], which is also within 1 [s] after pitch is initiated.

Roll Rate
Requirement SYS-FLGT-ROTA-05 states that the drone should be able to achieve a roll rate of 1080 [°/s] or 18.85 [rad/s]. Requirement SYS-FLGT-ROTA-06 states that the drone should be able to achieve this within one second of initiating roll. In the simulation, the drone is given a step roll input and its roll rate is scoped to measure its response.

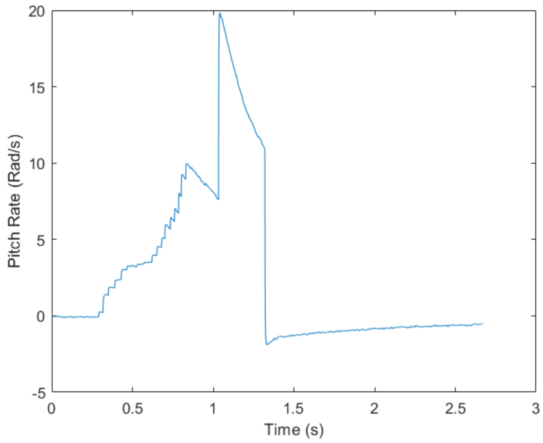


Figure 9.23: Pitch rate against time to verify the requirement for maximum pitch rate.

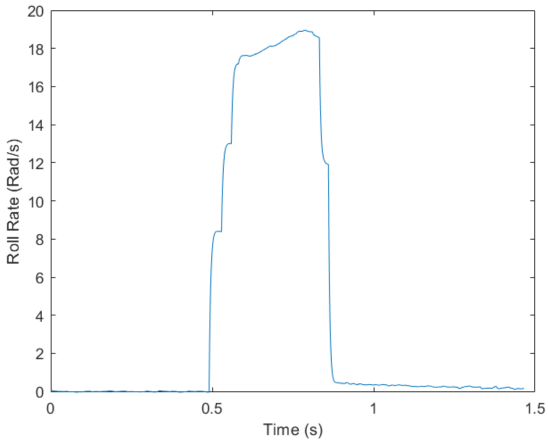


Figure 9.24: Roll rate against time to verify the requirement for maximum roll rate.

As observed from Figure 9.24, the roll rate of the drone is zero up until 1 [s] where the roll is initiated. The roll is initiated by giving a maximum step roll input, as a result the roll rate of the drone immediately jumps up to value of 19 [rad/s]. This again verifies both the requirements as a roll rate of 19 [rad/s] is more than the required rate of 18 [rad/s] and this rate is also obtained instantly which verifies the requirement of obtaining the maximum roll rate within 1 [s] of initiation of roll.

Yaw Rate

Requirement SYS-FLGT-ROTA-07 states that the drone should be able to achieve a yaw rate of 200 [°/s] or 5.25 [rad/s]. Requirement SYS-FLGT-ROTA-08 states that the drone should be able to achieve this within one second of initiating yaw. In the simulation, the drone is given a step yaw input and its yaw rate is scoped to measure its response.

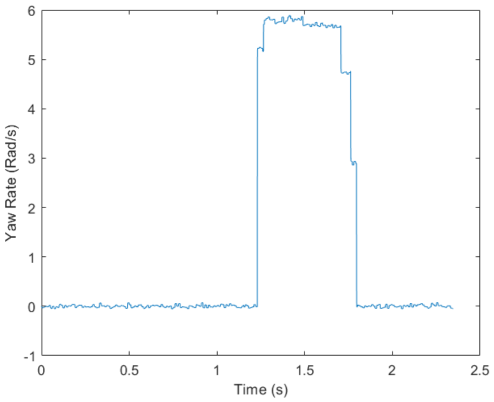


Figure 9.25: Yaw rate against time to verify the requirement for maximum yaw rate

As observed from Figure 9.25, the yaw rate of the drone is zero up until 1 [s] where the yaw is initiated. The yaw is initiated by giving a maximum step yaw input, as a result the yaw rate of the drone immediately jumps up to value of 5.3 [rad/s]. This again verifies both the requirements as a yaw rate of 5.3 [rad/s] is approximately equal to the required rate of 5.25 [rad/s] and this rate is also obtained instantly which verifies the requirement of obtaining the maximum yaw rate within 1 [s] of initiation of yaw.

9.5. Risk Assessment

In Performance & Simulation, the risks are related to the way in which the performance analysis were done, or with the functioning of the simulation. Below is a list of the five most significant risks specific to this subsystem, along with an explanation of why it is a risk and the reasoning behind accepting it or mitigating it.

- **PS-R1**: Braking performance significantly worse due to time taken to reach maximum reverse thrust.

  - As explained in the braking performance analysis, the braking was only analysed after maximum reverse thrust was achieved. Therefore, since this does not occur instantly, the performance will be worse than presented, though given the inability to model it at this stage of the design, the presented braking performance is the best estimate. Given that the relevant requirement was met by over 2 [s], it is very unlikely that this will be exceeded so this risk is no cause for concern.
- **PS:R2**: Optimal turn radius is unrealistic to achieve, making a timely completion of the lap difficult.
  - To achieve the turning performance described in the minimum turn radius analysis, it is necessary for the drone to be flying exactly at its maximum C_L (i.e. exactly the correct angle of attack). Achieving this without stalling may be a near impossible task, so it may be necessary for pilots to leave some margin resulting in a larger turn radius. As a mitigation measure, the simulation has been constructed such that future pilots of Peregrine may gain some experience in the simulator before flying the real drone.
- **PS-R3**: Aerodynamic coefficients not accurately estimated.
  - The aerodynamic coefficients (i.e. stability derivatives) are amongst the most difficult coefficients to estimate in an aircraft or flying wing without measuring them experimentally. Thus, despite the team's best effort to estimate them (along with verification and validation) in the Aerodynamic analysis, it is still possible that they are not very accurate which has a big effect on the drone's handling. However, given the difficulty in estimating such coefficients, not significantly more can be done until a prototype is built and experimental data can be collected - at which point model tuning could be performed.
- **PS-R4**: Hidden error in simulation
  - Given that no team members previously had any experience using Simulink, the team faced a significant number of errors in the construction of the simulation. Despite fixing the large majority of errors, it is still possible that errors exist, which could lead to an unrealistic simulation. To mitigate this risk, an extensive verification procedure was carried out, outlined in Section 9.6.
- **PS:R5**: Drone does not behave like it does in the simulation in real life.
  - If the drone behaves differently in real life than in the simulation, it may lead to pilots who tried the simulation crashing their drone on the first attempt. As a first mitigation measure, the simulation has been validated, as described in Section 9.6. As a second mitigation measure, once a prototype has been built and flown, the simulation will be updated such that it closely resembles the behaviour of the real drone.

9.6. Verification & Validation

In order to have confidence in the results of the simulation, it is important to both verify and validate the simulation, which is the goal of this section. In Subsection 9.6.1, the verification process and results are shown and then in Subsection 9.6.2 the validation approach is presented.

9.6.1. Verification of Simulation

The verification of the simulation will be done through a series of unit tests and system tests which, if successfully passed, will show with a high degree of confidence that the simulation is correctly implemented. The unit tests will test specific blocks of the simulation, namely the thrust block, aerodynamics block, gravity block, pilot inputs block, noise block and visualisation block (all of which can be seen in Subsection 9.4.1). Successfully passing these tests will demonstrate that all of the blocks that make up the simulation are working as they should, but does not yet show that the full simulation is working. For that, the system tests are used. These will involve tests that cover multiple blocks that were verified with unit tests, and upon passing all of these tests, the simulation can then be said to be verified. It should be noted that not all blocks that are visible in the simulation have to be verified, either because they are part of the Simulink library (which is already verified and validated), like the 6DOF quaternion block and flight parameters block (inside aerodynamics) or because there are no functions for which verification is required, such as the constants block. It should also be noted that the PID controller block will not be verified as part of the simulation, since that has been done in Section 8.7. The unit tests will be presented per block, starting with the thrust block.

Thrust block

In the thrust block, all of the unit tests are to check the correct implementation of mathematical operations such as multiplication, division and addition, so it is expected that the results match calculations by hand to computer precision. However, the "show value label" feature in Simulink only shows the output to three significant figures so to speed up the verification process, in order to pass the unit test, the output in Simulink will have to match the manual calculations to three significant figures. Since the calculations are very simple, it can be said with high confidence that if the output matches to three significant figures, it is being done correctly and thus is verified. For all such manual calculations, arbitrary values that are roughly in the range of the specific variable are chosen (though 0 is never taken as that may lead to the variable dropping out of the equation). For example, for the thrust of the outboard right engine, TRo, a value of 2 [N] is used and for the vectoring angle of the right engine, 0.15 [rad] is used. All of the unit tests for the thrust block, along with the outcomes are displayed in Table 9.3.

Table 9.3: Overview and results of unit tests on the thrust block.

ID	Test	Pass/fail?
UT01	Manual calculation for Fxt	Pass
UT02	Manual calculation for Fzt	Pass
UT03	Manual calculation for Mxt	Pass
UT04	Manual calculation for Myt	Pass
UT05	Manual calculation for Mzt	Pass
UT06	Check that output of F_T is the desired vector	Pass
UT07	Check that output of M_T is the desired vector	Pass

As required, all of the outputs of Simulink matched the manual calculations to three significant figures and hence they were all verified. For UT06 and UT07, the outputs of Fxt, Fzt, Mxt, Myt and Mzt were looked at and then it was successfully verified that the F_T and M_T blocks arranged them into the desired vector, which concludes the unit tests for the thrust block.

Aerodynamics block

Verification of the aerodynamics block is very similar to the verification of the thrust block, as most of the blocks which need to be verified involve only very simple mathematical operations and thus can efficiently be compared to a manual calculation. The criterion to pass the test is the same as for the thrust block (Simulink output must match manual calculation to three significant figures). Contrary to the thrust block however, the aerodynamics block has smaller blocks inside it. Unit tests UT08-UT10 are in the ”Force coeff” block, UT11-UT13 are in the ”Moment coeff” block, UT14-UT19 are in the ”Dimensionalising” block and finally UT20 and UT21 are simply in the aerodynamics block. The full list of unit tests as well as their outcomes are shown in Table 9.4.

Table 9.4: Overview and results of unit tests on the aerodynamics block.

ID	Test	Pass/fail?
UT08	Manual calculation for CL	Pass
UT09	Manual calculation for CD	Pass
UT10	Manual calculation for CY	Pass
UT11	Manual calculation for Cl	Pass
UT12	Manual calculation for Cm	Pass
UT13	Manual calculation for Cn	Pass
UT14	Manual calculation for dimensionalising CL	Pass
UT15	Manual calculation for dimensionalising CD	Pass
UT16	Manual calculation for dimensionalising CY	Pass
UT17	Manual calculation for dimensionalising Cl	Pass
UT18	Manual calculation for dimensionalising Cm	Pass
UT19	Manual calculation for dimensionalising Cn	Pass
UT20	Check correct implementation of rotation matrix of forces	Pass
UT21	Check that output of M_A is the desired vector	Pass
UT23	Check values of DCM_{be} and ensure matrix multiplication is correct	Pass

As can be seen in Table 9.4, all of the unit tests related to the aerodynamics block have been passed. For UT08-UT19, the same approach as with the thrust block was taken and no discrepancies were found. For UT20 and UT21, the inputs to the block were looked at and it was checked that the output was as desired, which turned out to be the case so this concludes the verification of the aerodynamics block.

Gravity block

The verification of the gravity block is much simpler than the previous two blocks, as the only thing that happens in this block is that the gravity force is transformed from the Earth reference frame (where gravity acts purely in the z direction) to the body reference frame using the direction cosine matrix DCM_{be}. Thus the sole test that needs to be performed is checking whether the matrix multiplication of the gravity force and direction cosine matrix is being done correctly, which can be done by checking the values of the direction cosine matrix and performing the matrix multiplication by hand. As shown in Table 9.5, this was successfully done, which concludes the verification of the gravity block.

Table 9.5: Overview and results of unit tests on the gravity block.

ID	Test	Pass/fail?
UT23	Check values of DCM_{be} and ensure matrix multiplication is correct	Pass

Pilot inputs block

The pilot inputs block is another relatively small block which can be checked with a series of quick tests. The only thing which must be checked in this block is that the Simulink interprets the inputs as the pilot intends, meaning that throttle controls RPM, yaw controls yaw, etc. To do so, the value of each of the inputs can be looked at individually and then it can be observed by moving the stick on the controller whether or not Simulink is correctly interpreting it. To pass the test, it is necessary that a movement in one stick on the controller only affects the parameter it should in the simulation (i.e. moving the pitch stick results only in a pitch input being given, and no throttle, yaw or roll). An overview of all the tests and the results can be in Table 9.6.

Table 9.6: Overview and results of unit tests on the pilot input block.

ID	Test	Pass/fail?
UT24	Check that roll stick controls roll	Pass
UT25	Check that pitch stick controls pitch	Pass
UT26	Check that yaw stick controls yaw	Pass
UT27	Check that throttle stick controls throttle	Pass

As can be seen in Table 9.6, all of the tests were passed and so this concludes the verification of the pilot inputs block. However, it should be noted that whenever a new controller is used with the simulation, UT24-UT27 must be performed again as different controllers use different numbering for the axes on the controller.

Noise block

Next the noise block needs to be verified, though this only involves one summation block. Thus the only test which must be performed is to ensure that this is done correctly, which can be done by observing the input and the corresponding output. To pass the test, the summation must be correct to three decimal places, as explained earlier. The result of this test is shown in Table 9.7.

Table 9.7: Overview and results of unit tests on the noise block.

ID	Test	Pass/fail?
UT28	Check whether summation is done correctly	Pass

As seen in Table 9.7, the test was successfully passed which concludes the simulation of the noise block.

Visualisation block

Finally, the verification of the visualisation block can be done. This involves checking whether everything is displayed correctly in the simulation. The two tests as well as their results are shown in Table 9.8.

Table 9.8: Overview and results of unit tests on the pilot input block.

ID	Test	Pass/fail?
UT29	Check that gates are placed at desired location	Pass
UT30	Check that all axis are correct in the simulation	Pass

For UT29, the coordinates of each of the gates was carefully checked with Figure 9.1. Then for UT30, pitch, roll and yaw inputs were given to check whether the drone pitched, rolled and yawed respectively. As can be seen in Table 9.8, both tests were passed which concludes the verification of the visualisation block and all of the unit tests.

Verification System Tests
Having completed all of the necessary unit tests, the final step in the verification process is to carry out system tests. These tests are designed to cover as much of the simulation as possible, such that if all of the proposed system tests are passed, the entire simulation is verified. The system tests are shown in Table 9.9, with an explanation to follow in the accompanying text.

Table 9.9: Overview of all proposed system tests.

ID	Test
ST01	Step input on throttle
ST02	Pulse input on roll
ST03	Pulse input on pitch
ST04	Pulse input on yaw
ST05	Check time to 90 [m/s] in straight, horizontal flight

The first thing to note is that all of the system tests were performed with the PID controller connected (which has already been verified in Chapter 8) so that the drone would remain within realistic flight conditions without requiring active inputs to the controller. Also, since no quantitative analysis is possible here, all of the system tests will involve a qualitative analysis to determine whether the test is passed or failed.

With regards to ST01, a step input of 0.5 was given, meaning that half throttle was input for a period of 5 [s] and several results were looked at. These were the thrust forces from each of the four engines, along with the moments generated, the aerodynamic forces and moments and the velocity. The thrust forces in x direction from the four engines varied with velocity in the way expected from the Power & Propulsion analysis and the thrust force in z direction was usually 0 [N], though did deviate occasionally to small thrust values which is consistent with the way in which the PID controller controls roll and pitch at lower speeds. All of the moments due to the thrust were also as expected - low in magnitude and oscillating around 0 to avoid the drone pitching, rolling or yawing. The same can be said of the aerodynamic moments. For the aerodynamic force coefficients, the results are also logical, as C_L continuously reduced (since an increasing velocity means a lower C_L is required), C_D remains low and relatively constant, and C_Y remained roughly 0. The last parameter which was investigated was the velocity, which increased continuously until around 60 [m/s] where it stopped increasing. This can be explained by the fact that only half throttle was applied so it is unable to accelerate past 60 [m/s] since the drag becomes to large. Combining all of these results, it can be said with reasonable confidence that ST01 has been passed.

Moving on to ST02, a pulse input of magnitude 0.5 will be applied for 0.2 [s], with several parameters again investigated. The reason a normal step input is not used is that after around 0.5 [s], the roll angle will become unrealistically large and the drone would crash in real life, so to ensure a realistic flight scenario is replicated, only a pulse input is used with a short duration. The parameters to be investigated are the roll rate, the aerodynamic sideforce, aerodynamic moment around the roll axis, the sideslip angle and velocity. The roll rate was, as it should, approximately 0 [°/s] for the majority of the time, except for when the pulse was applied, where it reached a maximum roll rate of 450 [°/s] which is in line with what Control & Stability would expect (since the maximum roll rate for full deflection is 1080 [°/s] and here half of that would be expected). In the sideforce, there was a spike of sideforce into the turn while the drone was rolling and then returned to roughly 0, as expected. Again, for the rolling moment, a spike could be seen at the moment the roll input was given and then it returned to 0 as expected. The sideslip angle continuously increased over time, starting at when the roll rate was input, to around 10°which is logical given that as the drone goes deeper into the turn, the sideslip angle should increase. Lastly, the velocity slightly increased despite no thrust input, though this can be explained by the fact that this test was performed at a speed of 35 [m/s] so thrust vectoring was used to aid with the roll. Since the vectoring angle cannot reach 90°, there was still a component in the x direction which resulted in a slight acceleration. With all of these results logically explained, it can be said that ST02 has been passed.

Next ST03 was carried out. Applying a pulse input to the pitch rate, the parameters to analysed are: the pitch rate, the aerodynamic moment around the pitch axis, the angle of attack and the velocity. The pitch rate shows a logical trend as it is roughly 0 [°/s] at all moments except when the pitch input is given where it increases to roughly 500 [°/s], as expected given the maximum possible pitch rate is 1000 [°/s]. The aerodynamic moment around the pitch axis is also roughly 0 [Nm] the entire time except for when the input is given, which is again as expected. The angle of attack showed different behaviour, the angle of attack initially jumped up before beginning to come back towards 0°as the drone began to climb. However, there came a point after around 3 [s] where the angle of attack again began to increase. By also looking at the velocity, this can be explained by the fact that at this point the drone stalled, so it is logical that the angle of attack began to increase again. Thus, with all parameters showing the expected behaviour, ST03 has been passed.

For ST04, the same procedure as for ST03 and ST02 is again used, though with a pulse input on yaw. The parameters of interest for this test are the yaw rate, the thrust moment around the yaw axis, and the sideslip angle. The yaw rate response to the pulse input was given, where it reached a yaw rate of approximately 145 [°/s]. Since the maximum expected yaw rate from Chapter 8 is 300 [°/s], this is a very reasonable value. For the thrust moment around the yaw axis, similar behaviour can be observed; it is 0 [Nm] for all times except when the pulse input is given where it jumps to allow the drone to yaw. Lastly, the sideslip angle was observed. At the point where the pulse input was given, the sideslip angle jumped to around 14°and then steadily returned to 0°in around 3 [s], which is the expected outcome since after yawing the drone's inertia should eventually return the sideslip angle to 0°as seen happen in the simulation. Since all of these parameters are as expected, ST04 has been passed.

The final system test is ST05 which compares the thrust and drag characteristics of the simulation with the Power & Propulsion analysis in Chapter 7. There it was found that the acceleration from stall speed to 90 [m/s] should take 3.2 [s], and in Figure 9.21 it can be seen that it took 3.0 [s] in the simulation. Since the simulation also includes wind turbulence and sensor noise, this small difference is understandable and thus, ST05 has been passed.

9.6.2. Validation of Simulation

Since there are no drones with similar performance to Peregrine, there is no experimental data which can be referred to for validation purposes. Therefore, to validate the simulation, it must be ensured that all of the inputs to the simulation are validated. If this is the case, then the simulation itself is also validated.

There are inputs from three other subsystems that influence the simulation, namely the aerodynamic coefficients used in the equations of motion (Equation 9.14-9.19) from Aerodynamics, the thrust coefficients from Power & Propulsion and the PID controller from Control & Stability. As these are all verified and validated in their respective subsystems, this is sufficient to say that the simulation is validated. Combined with the unit tests and system tests performed on the simulation, the simulation can now be said to be verified and validated, which ensures the quality of the results presented in Subsection 9.4.3.

9.7. Compliance Matrix

The requirements and their compliance for Performance & Simulation are indicated in Section 9.7. All requirements were passed.

Table 9.10: Compliance matrix for the Performance & Simulation subsystem.

Identifier	Status	Reference	Identifier	Status	Reference
SYS-FLGT-MOVE-01	Pass	Subsection 9.4.3	SYS-FLGT-MOVE-03	Pass	Subsection 9.4.3
SYS-FLGT-MOVE-04	Pass	Subsection 9.3.2	SYS-FLGT-ROTA-01	Pass	Subsection 9.4.3
SYS-FLGT-ROTA-03	Pass	Subsection 9.4.3	SYS-FLGT-ROTA-04	Pass	Subsection 9.4.3
SYS-FLGT-ROTA-05	Pass	Subsection 9.4.3	SYS-FLGT-ROTA-06	Pass	Subsection 9.4.3
SYS-FLGT-ROTA-07	Pass	Subsection 9.4.3	SYS-FLGT-ROTA-08	Pass	Subsection 9.4.3

10

Risk Assessment

Proper risk management is an important part of project management, especially for those of extensive duration, since risks that are not properly accounted for may lead to the project's failure. Thus it is very important to have a robust risk management system which minimises the probability of any risk leading to failure of the project. Here, all of the most relevant risks for this project, both organisational and technical, will be presented. The process of risk management will be done by first identifying and assessing all the risks in Section 10.1 and then the appropriate mitigation measures will be presented in Section 10.2.

10.1. Risk Identification and Assessment

The first step in the risk management process is the identification and assessment of all the potential risks. Identification of all potential risks was done by considering everything that could go wrong with the system as a whole (for technical risks), as well as the functioning within the group (for organisational risks). Each of the risks was then given its own unique identifier, and it is indicated what type of risk it is. For assessment of these risks, each one was ranked in terms of likelihood of occurrence and severity of impact on the project if it does occur.

NASA identifies four categories of risk that fall under technical risk assessment, namely safety, technical, cost and schedule [1]. For assessing the risks in terms of severity, a scale from 1 to 4 was chosen. The definition of severity varies for the different types of risk. For all of the technical risks (as defined by NASA), the definition of severity for each ranking is shown in Table 10.1.

Table 10.1: Definition of severity by rank for the four types of technical risks.

Rank	Cost	Safety	Schedule	Technical
1	<25€	Very minor injuries	1 working day	One component failure
2	25-100€	Minor injuries	2 working days	Multiple component failure
3	100-250€	Serious injury possible	3-4 w working days	Subsystem failure
4	>250€	Serious injury	5+ working days	System failure

For the definition of severity for organisational risks, the following ranking system is used.

- 1 = Negligible (inconvenience to project)
- 2 = Marginable (minor impact on flow of project)
- 3 = Critical (major impact on flow of project)
- 4 = Catastrophic (leads to project failure)

For the definition of likelihood, a scale from 1 to 5 was chosen, which is explained below. Note that for likelihood, unlike for severity, the definitions are the same for every type of risk.

- 1 = Very low (surprised if it happened during the DSE or entire lifetime of drone)
- 2 = Low (slight possibility of occurrence during DSE or entire lifetime of drone)
- 3 = Moderate (considerable chance of occurrence during the DSE or lifetime of drone)
- 4 = High (realistic chance of occurring or entire lifetime of drone)
- 5 = Very High (expected to happen during the DSE or entire lifetime of drone)

Using this method for assessing the risks, the list of all risks along with their unique identifier, and likelihood and severity ranking are provided in Table 10.2. In the fourth column, severity is abbreviated to ”S”, likelihood to ”L” and the ranking is given in coordinates of the form (severity, likelihood).

---

[1] `https://www.nasa.gov/seh/6-4-technical-risk-management` Accessed 19th May 2021]

Table 10.2: Overview of all risks and their corresponding severity and likelihood ranking.

ID	Type	Risk	Possible Causes	(S,L)
F1	Cost	Cost budget is exceeded	Material and manufacturing costs higher than expected	(3,1)
F2	Cost	There is no market for the drone	Price is too high, not many highly skilled pilots who would purchase this drone.	(4,2)
F3	Organisational	Technical difficulties during a review or the symposium.	Lack of preparation, factors out of our control (i.e. power outage).	(2,3)
F4	Organisational	Remaining necessary tasks to begin producing drone not finished	Lack of motivation, lack of planning	(4,2)
F5	Safety	System failure puts spectators and pilots at risk	Insufficient verification and / or validation of the product	(3,1)
F6	Safety	Drone is used in no fly zone	Ignorance, curiosity	(4,2)
F7	Safety	User too inexperienced to operate such a fast drone	Ignorance, not fully understanding risks	(3,2)
F8	Schedule	Design falls less than two days behind schedule	Underestimation of time required, inefficiency	(2,5)
F9	Schedule	Design falls more than two days behind schedule	Several group members get sick, productivity is too low	(3,2)
F10	Schedule	Unfeasible to manufacture drone as intended	Materials to thin, techniques proposed not appropriate	(4,4)
F11	Schedule	Insufficient storage space for electronics and other objects	Missing components, inaccurate dimensions for off the shelf parts	(4,3)
F12	Technical	Final design does not comply with requirements	Incorrect analysis during design phase	(3,5)
F13	Technical	Subsystems incompatible with each other	Not enough communication between departments	(4,2)
F14	Technical	Incomplete list of requirements	Time constraints, oversight	(3,3)
F15	Technical	Drone too difficult to control during a race	High performance, high manoeuvring speed	(3,3)
F16	Technical	Drone does not comply with race regulations	Requirements do not allow for a legal design	(1,3)
F17	Technical	Inadequate verification and / or validation of final product	Limited time, inexperience	(3,3)
F18	Technical	Structure experiences loads higher than it was designed for	Tight turns at velocities greater than 60 [m/s]	(4,5)
F19	Technical	Inability to hover during strong crosswinds	Large surface area exposed to crosswinds	(2,4)

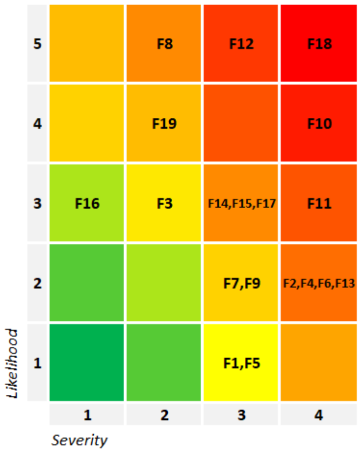
The corresponding reasoning for giving each of the risks in Table 10.2 their corresponding likelihood and severity ranking is given in Table 10.3.

Table 10.3: Reasoning for the likelihood and severity ranking given to each risk in Table 10.2

ID	Explanation for likelihood and severity
F1	For this risk a likelihood of 1 was set and a severity was chosen based on that likelihood. From the market analysis it was determined that the budget was realistic and hence it is not expected to exceed it by more than 10%.
F2	If there is no market for such a drone, it will be very costly - both in terms of time spent on designing the drone and the cost of the prototype and development, thus severity is 4. Given the rising popularity of drone racing however, there is only a slight chance of this risk materialising, thus the likelihood is 2.
F3	Technical difficulties during an important meeting is more significant than just negligible, but also not a major impact, hence the severity is 2. Since technical difficulties are easy to encounter when the presentation file is worked on by 10 different people, the likelihood is 3.

ID	Explanation for likelihood and severity
F4	If the drone does not actually get produced it is a failure of the project so the severity is 4. However this has a low likelihood of occurring given the progress and motivation so far in the project so the likelihood is 2.
F5	It is possible for a drone to seriously injure the handler and / or spectators, due to, for example, the battery exploding, hence a safety risk with severity 3. The chance of such an event occurring is very low because of the many precautions implemented by manufacturers, leading to a likelihood of 1.
F6	Using the drone to fly in a no fly zone, especially given its high speeds can potentially lead to life threatening injuries, hence the severity is 4. The majority of people are aware of the dangers of flying in no fly zones so this is something that has only a slight chance of occurring, hence the likelihood is 2.
F7	Operating this drone will be something that requires practice and experience so if someone buys it as their first drone and just starts flying there is a risk of serious injury, hence the severity is 3. However, that is not something that is very likely to happen given that most people would understand the risks of flying a drone at 300+ [km/h] so the likelihood is 2.
F8	Per definition a schedule risk of 2 days or less has a severity 2, and as has occurred during the DSE thus far, the schedule is very tight so it is expected that at one point the design will fall 2 days or less behind schedule. Thus, the likelihood is 5.
F9	This depends on the magnitude of the delay but a severity of 3 was chosen because it represents a middle ground and given that delays greater than 3 days have not occurred thus far in the DSE, the likelihood is only 2.
F10	A major problem with the design (for example using materials which are too thin) would have a very lengthy delay on the project as it may require a redesign, hence the severity is 4. Given the group's inexperience with producing complex designs to be manufactured in reality, the likelihood is also high (4).
F11	Since this would also likely require a redesign which would cause a delay of more than 5 working days, hence the severity is 4. Given the number of components which must fit in the drone, there is a considerable chance this would happen, hence the likelihood is 3.
F12	The severity will vary considerably depending on which requirement is not met but as a middle ground a severity of 3 was chosen. Furthermore, it is expected that out of all the requirements that have been specified at least one cannot be met hence the likelihood is 5.
F13	If subsystems are not compatible with each other the entirety of the system cannot work hence it will lead to system failure, or in other words severity 4. However, since at this stage of the design (without all details having been worked out yet) all subsystems do appear to be compatible with each other so the likelihood is 2.
F14	The severity of an incomplete requirement list is dependent on what requirement is left out, though likelihood that a subsystem requirement is left out is higher than that of a system requirement, hence the severity is 3. The likelihood that one or more requirements are missing is considerable even through the use of proper PM/SE tools, thus a likelihood of 3 was chosen.
F15	When making a 25g turn, the drone will complete a 180 degree rotation in less than 0.4 seconds which could make it impossible to control. This can be considered a failure of the control subsystem, thus severity is 3. Given the requirement for 25g turns, it is considerably likely, thus the likelihood is 3.
F16	Since complying with racing regulations is not a requirement of the project, the severity is only 1. As a result of it not being a requirement the drone will not be designed with the regulations in mind so there is a considerable chance that it will not comply with all regulations, hence likelihood is 3.
F17	If a subsystem is not verified properly, its performance may be vastly different to what was expected, thus a subsystem failure meaning severity 3. Given the groups relative inexperience with verification and validation, the likelihood is also 3.
F18	If the structure breaks, the drone will crash which is a system failure, hence the severity is 4. Since there is nothing stopping the pilot from (accidentally) making a tight turn when flying faster than 60 [m/s], the likelihood is very high (5).
F19	Not being able to maintain a hover during strong crosswinds is a disruption to the mission, hence the severity is 2. The likelihood is high, because while hovering the entire wing surface area is exposed to the oncoming flow.

To aid with identifying which of the risks pose the biggest threat to the project, a risk map can be made. The risk map is shown in Figure 10.1. Note that each individual cell in Figure 10.1 is colored to represent the corresponding consequence, where redness indicates badness. As can be seen, there are several rights that are very close to the upper right corner, which is the riskiest part of the map to be. In particular, risks F12, F18, F10 are dangerously high. For this reason, it is important to carefully devise a risk mitigation strategy, including specific actions that must be undertaken to keep all risks within acceptable limits. This will be done in the following section.



	1	2	3	4
5		F8	F12	F18
4		F19		F10
3	F16	F3	F14,F15,F17	F11
2			F7,F9	F2,F4,F6,F13
1			F1,F5	

Figure 10.1: Risk map before any mitigation measures have been taken into account.

10.2. Risk Mitigation

In this section, all of the risks will be mitigated to try and move them towards the bottom left of the risk map, which is known as the risk mitigation strategy. There are four ways to mitigate a risk, which are listed and explained below.

- Remove - Eliminate a risk from the project.
- Reduce - Reducing the impact of a risk by either decreasing its likelihood or its severity.
- Transfer - Pass the risk on to a third party.
- Accept - No course of action to be taken at the moment, though the risk should be monitored throughout the project to determine if and when it is no longer possible to accept.

The latter three measures are used in this risk mitigation strategy, while removing a risk is not. This is because removing a risk would require a large design change which is no longer feasible at this design stage. In Table 10.4, all the previously identified risks are collected and the appropriate risk mitigation actions and methods are mentioned.

Table 10.4: Risk mitigation strategy and action to be taken for each of the risks.

ID	**Action**	**Mitigation step**	**Change (S,L)**	**New (S,L)**
F1	Accept	Cost budget revealed this is something very unlikely to happen so at this point accepting and monitoring this risk is sufficient	(0,0)	(3,1)
F2	Reduce	Conducting a proper market analysis	(-1,-1)	(3,1)
F3	Reduce	Use the practice opportunities and test the presentation multiple times beforehand. Try to anticipate possible failures and address them.	(0,-1)	(2,2)
F4	Reduce	Make a clear project Gantt chart to ensure remaining tasks are identified and time allocated to them.	(0,-1)	(4,1)
F5	Transfer	This is the responsibility of the race organizers	(-3,-1)	(0,0)
F6	Reduce	Distribute user manual with the drone detailing dangers of flying in no fly zones to ensure awareness of the customer.	(0,-1)	(4,1)
F7	Reduce	Distribute user manual with the drone detailing assumed drone racing experience and potential risks for inexperienced pilots.	(0,-1)	(3,1)
F8	Accept	A time margin is already in place	(0,0)	(2, 5)
F9	Reduce	Maintain good communication and re-allocate resources early when things start to go wrong.	(-1,-1)	(2, 1)

ID	Action	Mitigation step	Change (S,L)	New (S,L)
F10	Reduce	Make a clear production plan whereby these problems can be identified in advance. Note severity also decreases because with a clear plan in mind, any mistakes will likely not be as large.	(-1,-2)	(3,2)
F11	Reduce	Make a CATIA drawing to scale such that these problems would be exposed. Note that since making such a drawing reduces the magnitude of the problems that could arise, the severity also decreases by 1.	(-1,-1)	(3,2)
F12	Reduce	Verification and validation done in each subsystem to have greater confidence in every result.	(0,-3)	(3,2)
F13	Reduce	Continue good communication and clear overview of interfaces. Note that severity also decreases because when good communication is maintained, there will only be minor incompatibility issues.	(-1,-1)	(3,1)
F14	Reduce	Having followed PM/SE procedures, if a requirement that further constrains the design is deemed necessary, add it to the list of requirements.	(-1,-1)	(2, 2)
F15	Reduce	Make a simulation to test its controllability and make changes if required	(-1,-1)	(2,2)
F16	Accept	Complying with the regulations is not a requirement for this project.	(0,0)	(1,3)
F17	Reduce	Make a proper plan for verification and validation activities including a timeline.	(0,-1)	(3,2)
F18	Reduce	Include airspeed indicator in goggles which blinks red when tight turns are not allowed (i.e. when velocity is >60 [m/s])	(0,-3)	(4,2)
F19	Reduce	Allow for greater aileron deflections while hovering.	(0,-2)	(3,1)

After applying all of the mitigation strategies, a new risk map can be made in which the severity and likelihood are modified to reflect the effect of the mitigation strategy. This is shown in Figure 10.2.

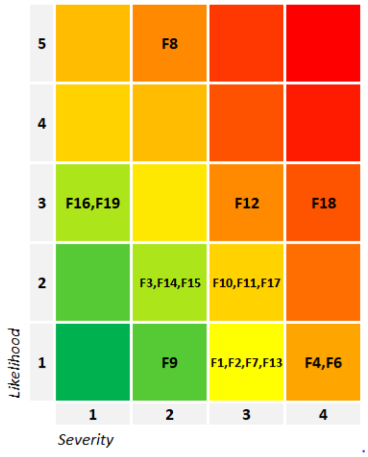


Figure 10.2: Risk map after the risk mitigation measures have been taken into account.

The main risks which still remain towards the upper right of the graph are F8, F12 and F18. F8 has to do with the design falling less than two days behind schedule while F12 has to do with the final design not complying with all the requirements, both of which are understandably large due to the relative inexperience of the team. F18 is the risk that the structure experiences loads that are higher than 25g due to, for example, turning at speeds greater than 60 [m/s]. This is also a large risk since the drone is physically capable of much higher g turns and it is very difficult to make a system that prevents this from happening, so a pilot misjudgement may cause this. A future solution would be to have an indicator in the FPV goggles telling the pilot of over-speed before turning or limiting the motor rotation.

11

Sustainability Assessment

The methodology chosen for a proper sustainability assessment was Life Cycle Impact Assessment (LCIA), which will be performed on all the major components of Peregrine. This chapter will focus on all the environmental footprints left behind due to all the processes involved in production of Peregrine and the end-of-life strategies for the materials used to ensure recyclability requirements are met. LCA is a standardised tool where an analysis is performed on the entire product chain starting from raw material extraction and ending at post consumer processing, often referred as a *cradle-to-grave* methodology [4]. The End-of-Life strategies for each of the material being used will be devised in Section 11.2 in order to ensure the system and subsystem requirements are met. The sustainability assessment performed will prove to be really beneficial if all requirements are met as other drones on market do not have a strong emphasis on the sustainability/recyclability. Hence if the requirements especially the drone being 80% recyclable are obliged with, Peregrine will have a social advantage and hence a huge marketing plus.

11.1. Environmental Footprint

A footprint is a quantitative measurement describing the appropriation of natural resources by humans [20]. A product's footprint indicates the direct and indirect consumption of materials/resources associated with all the relevant activities/upstream processes in the whole production process[1].

In order to deem a product 'sustainable' it is necessary to evaluate the total possible environmental impact of that product so better decisions can be made if the impact is high. It also aids in forming acceptable or favourable End-Of-Life strategies. For evaluation of the environmental impact and footprints of Peregrine, the impact assessment method chosen was life cycle assessment in which the use of openLCA software and relevant databases was employed. For the production of a LiPo battery no process could be found in all the databases made readily available, hence it was necessary to resort to an existing LCIA study which will be discussed separately from the rest of the components.

The impact assessment method integrated in this life cycle impact analysis is through the means of midpoint indicator. Midpoints are considered to be links in the cause-effect chain (environmental mechanism) of an impact category. This assessment method has been kept the same as the previous design phase for consistency of the results [4]. The midpoint method is a characterisation method that provides indicators for comparison of environmental interventions at a level of cause-effect chain that is linked from the emissions and resource consumption from initial stages to the endpoint level defined for a specific impact category. [26]

11.1.1. Production Peregrine

For a complete evaluation of the environmental footprints that will be left behind in the production process of Peregrine using openLCA, having complete material compositions for all products was a prerequisite. These materials were then put as inputs and the output was defined as Peregrine. Use of relevant databases was employed namely ecoinvent, environmental footprints and openLCA LCIA[2].

The structural components, propellers and tilting mechanism materials were determined by the Structures and Material subsystem. However for actuators and motors being integrated it became necessary to assume certain material composition due to lack of sufficient data available, these can be seen in Table 11.1. These assumptions were made logically after performing some research and keeping in mind the sub-components of the motors and the actuators.

In this assessment of the production certain assumptions have to be outlined before addressing the results. The AWG wires being integrated in the design were primarily made of tinned copper and silicone as an insulator. However due to limitations a certain available product flow for AWG wires was found comprising

---

[1] URL: https://www.lifecycleinitiative.org/ [Accessed on 29/04/202129]

[2] URL:`https://nexus.openlca.org/databases` [Accessed on 15/05/2021]

of 55.9% Copper, 43.6% PTFE and 0.5% Tin. This product flow was adopted for much more reliable results rather than evaluating the materials individually.The impact would not be underestimated with this assumption as silicone is more durable and environment-friendly than PTFE (thermoplastic)[3][4].

The production of LiPo battery however will be dealt in Subsection 11.1.2 as no data set could be found that entailed the production of such a battery.

Major components such as on board actuators and motors integrated within Peregrine have been analysed, per material used for it. However some components for which devising a material composition is challenging due to lack of information or the fact that they might not be recyclable namely the camera, antenna, RX, VTX and speed and flight controllers have been omitted from the sustainability assessment. These components weigh a total of 21.0[g] and it will be ensured that sustainability requirements are met incorporating this mass as well.

Table 11.1: The components along with their material composition for impact assessment.

Component(s)	**Material Composition**	**Total Mass [g]**
Structure	62% CFRP and 38% EPP	88.5
Tilting Mechanism	90% CFRP, 10% Stainless Steel	8.5
Propellers	100% CFRP	10.0
14 and 24 AWG wiring	55.9% Copper, 43.6% PTFE and 0.5% Tin	11.1
Motors[5]	50% Al-Mg, 20% Ti and 10% Steel	116.0
Actuators	80% Stainless Steel, 20% Polyethylene(PET)	50.0

It must be noted that CFRP in Table 11.1 corresponds to a 60% carbon fibre fraction while 40% is epoxy resin. It was assumed that the N52H arc magnets in the motors will contribute to 20% of the motors' mass as this neodymium magnet is a permanent magnet which can last decades without losing their strength[6].

LCIA impact analysis results are grouped in multiple *categories* as shown in Table 11.2. This is done in openLCA where all emissions are grouped into a category if it is contributor to that specific category. The total impact of the Peregrine with respect to these categories are also specified in Table 11.2 with their units. The units are in [kg] and [mol] as the category entails all the relevant emissions that are released in the air or water corresponding to that specific category. Contribution per materials towards these categories are displayed graphically in this section.

Acidification refers to the reduction of pH of water bodies as when carbon dioxide dissolves in water to form a weak acid, specifically H_2CO_3 which dissociate to produce carbonate ions, releasing H+ ions and hence decreasing the pH. This can have direct and/or indirect adverse effects on organisms in the ocean [52]. Eutrophication, characterized by excessive plant and algal growth, is caused by the enrichment of minerals in an environment. The consequences include blooms of cyanobacteria (algae), tainted drinking water supplies making it toxic etc. and as human activities have been increasingly contributing to the accelerated rate of eutrophication, this category cannot be neglected [7].

---

[3] URL: https://earthhero.com/silicone-the-plastic-alternative/ [Accessed on 16/06/2021]

[4] URL: https://www.ecoandbeyond.co/articles/silicone-vs-plastic/ [Accessed on 16/06/2021]

[6] URL: https://www.first4magnets.com/tech-centre-i61/frequently-asked-questions-i69 [Accessed on 16/06/2021]

[7] URL: https://www.nature.com/scitable/knowledge/library/eutrophication-causes-consequences-and-controls-in-aquatic-102364466/ [Accessed on 18/06/2021]

Table 11.2: Each category with the **total** impact with units specified.

Category	Impact	Unit
Climate Change	1.982	[kg]
Water usage	0.786	$[m^3]$
Resource Use	32.322	[MJ]
Land Use	3.289	$[m^2]$
Acidification	0.0126	[mol]
Eutrophication[a]	0.0026	[kg]
Ozone Depletion	$1.62\cdot10^{-8}$	[kg]
Resource Use	0.0174	[kg]

---

[a] This include both freshwater and marine eutrophication.

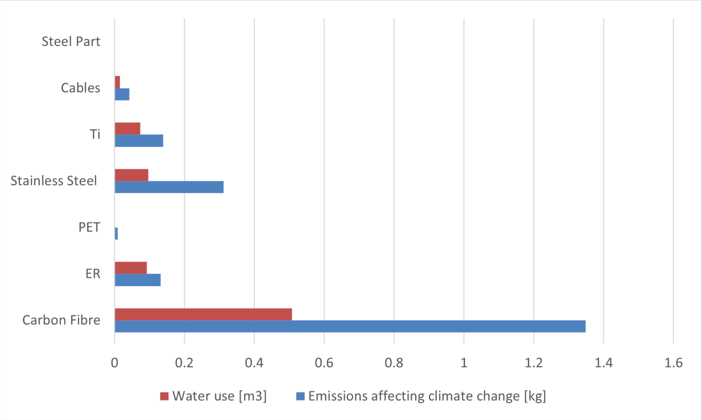
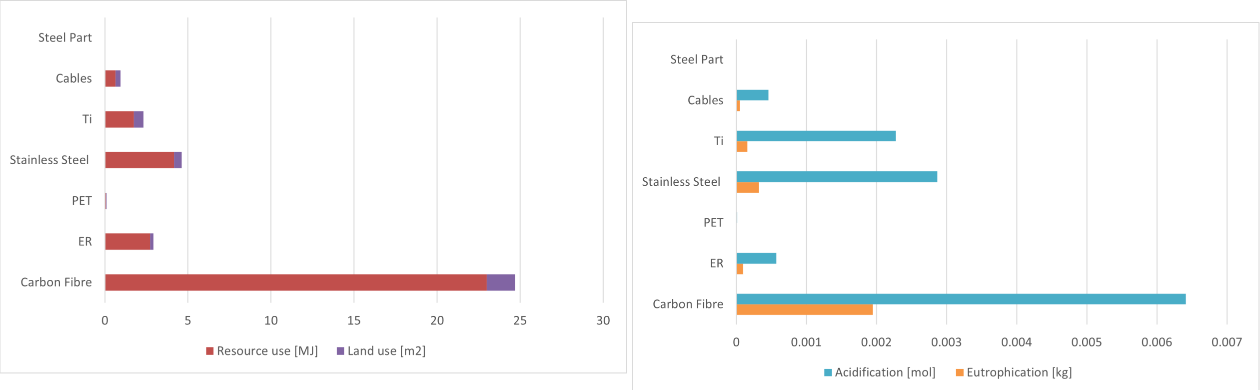


Figure 11.1: Graph of different materials contributing to climate change and displaying the water usage in the production process.

Note that there are two categories by the name of 'Resource Use', the first one indicates the consumption of fossils in [MJ] while the other one shows the usage of the minerals and metals in the upstream processes.

Climate change category entails the emissions of carbon dioxide, CO_2 and methane NH_3 that would be released in all the upstream production/manufacturing processes (Nitrous Oxide, N_2O would also be released but negligible). It can be seen from Figure 11.1 that the leading contributor to CO_2 and NH_3 emissions is the carbon fibre production process followed by stainless steel and then titanium.



(a) Graph of the relative energy consumption and land usage during the production process of the materials.

(b) Graph of materials contributing to acidification and eutrophication.

A total of 35.43 [MJ] would be consumed in the production processes of Peregrine. These production processes involve consumption of uranium, natural gas, crude oil, hard coal and brown coal. Again it can be seen that carbon fibres contribute the most in both land use and resources.

Emissions assisting in acidification were nitrogen mono-oxide NO, nitrogen oxides NO_X, SO_2 and ammonia NH_3 where the upstream processes of stainless steel and cable production and production of CFRP, Ti and epoxy resin all play a role as seen in Figure 11.2b.

The production of cables and stainless steel are the leading contributors to eutrophication, Figure 11.2b, releasing NO in the process (0.141[g]) followed by production of carbon fibre, titanium and lastly epoxy resin. The rest of the emissions that will be released in water are nitrogen oxides NO_x, NH_3, nitrate NO_3^-, ammonium NH_4^+ and phosphate PO_4^{-3}.

The emissions that would add to ozone depletion have not been added in the figures as they can be considered negligible (in the power 10^{-8}). However CFC-10, 1,1,2-trichlorotrifluoroethane and HCFC-22 would be released in the production of Ti, epoxy resin and CFRP.

Required resources (minerals and metals) in the production of Peregrine are silver, gold, copper and zinc in decreasing order, total of 0.01926[kg].

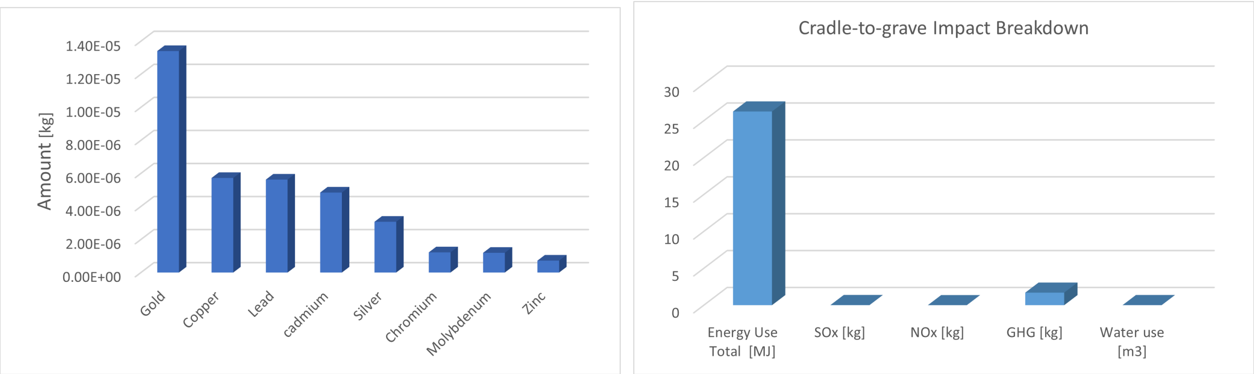
It must be noted that the reason for CFRP being the leading contributor to every category is because it is going to be used for the spar, tilting mechanism and actuator rods in the structure and the propellers leading to a CFRP mass above 68[g], greater than the masses of each other material used. Hence, it cannot be safely concluded that CFRP has the most drastic effect on the environment but it does give an indication. However even though production of CFRP is more energy intensive, the durability of CFRP production still makes it stand out but it must be ensured that it is recyclable to an acceptable percentage in order to make the life cycle of CFRP sustainable. CFRP use has also been reduced in the structure (skin) where EPP was enough to sustain the loads.

Material Choice
A comparison of environmental impact between an Aluminium alloy and CFRP with same masses was also performed using LCA and the results did confirm that production of CFRP has a relatively higher impact. An analysis was then performed by the structures department if an aluminium alloy can replace CFRP or not. The results showed lower deflections but higher stresses and lower tensile strength meaning that the structure would have to be sized again leading to a higher mass as the density of an aluminium alloy is greater. Considering that deflections are not a cause of concern and the mass would exceed the budgeted mass, a decision was made mutually that carbon fibre will remain and not be replaced.

11.1.2. Production of LiPo Battery

The environmental impact of the LiPo battery integrated in Peregrine's design was performed in the previous design phase where a case study was used for its life cycle assessment [4]. This was done as LiPo battery components could not be found in the databases readily available.

The case study, however was on a lithium nickel manganese cobalt oxide (NMC) battery. The difference between the two being that NMC battery has a liquid electrolyte whereas a LiPo battery has a solid or semi-solid(gel) as its electrolyte hence the results of the study can be assumed to be similar. The LCA results are presented for 1 [kWh] of battery materials to facilitate analyses of such batteries. As the functional units of the study is [kWh], useful results can be derived from this assessment [21]. These are displayed and visualised in Table 11.3 and Figure 11.3b.



(a) Resources used from ground

(b) LCA results displaying impact breakdown of the LiPo battery.

Figure 11.3: Resources and energy usage for the battery.

In this research the three biggest contributors to the energy footprint were identified namely, production of NMC powder, aluminium (mostly used for battery casing) and cell production. As NMC powder does not apply to the LiPo battery case, it was neglected in the estimation of total energy usage. "Evidently, 50.8% of the water consumption, 18.1% of the total energy use, 17.0% of the GHG emissions, 15.3% of the PM10[8] emissions, and 9.2% of the NOx emissions can be ascribed to the aluminum content of the battery, whereas the cell production process represents 19.2% of the total energy use, 19.0% of the GHG emissions, 13.2% of the NOx emissions, and 7.4% of the water consumption"[21].

Table 11.3: Environmental footprints grouped in categories

	Amount	**Unit**
Total Energy Usage	26.247	[MJ]
GHG/CO_2	1.70	[kg]
SOx Emissions	18.648	[g]
NOx Emissions	2.259	[g]
Water Usage	0.0175	[m^3]

---

[8] PM10 emissions correspond to the particulate matter/particles having a diameter less than or equal to 10 micrometers.

The sub-process of cell production is one of the driving process in overall environmental production within all impact categories. It is essential that for the battery a sustainable EOL strategy is employed to have minimal environmental impact. This strategy is discussed in Subsection 11.2.5.

11.2. Strategies for EOL

The EOL phase starts at the point of disposal by the consumer. Even though the system functionality cannot achieve its customer needs, sustainability efforts are crucial and must be maintained in this phase of the life cycle[18]. It is necessary to look into strategies for both CFRP structural framework and the LiPo battery as it must be ensured that these components of Peregrine has no or minimal environmental impact after usage. As materials used in these components are not biodegradable this does not apply here and hence will not be discussed. It must be duly noted that the processes of recycling that will be discussed require significant energy consumption and sometimes other resources.

11.2.1. CFRP

As CFRP contributes significantly to the structure of the Peregrine, it was necessary to evaluate it's EOL strategy which was already done so in the previous design phase of the project. [4] CFRP (Carbon Fibre Reinforced Polymer) includes both carbon fibres and the epoxy resin.

"Various technologies have been proposed for recovering carbon fibers from carbon fiber-reinforced epoxy resin composites with excellent characterisation,including mechanical processes, chemical methods, and thermal technology. However, long carbon fibers are difficult to obtain using a mechanical process, and their mechanical properties are seriously damaged with this technique"[25]. Research into such processes/technologies has been performed where waste management hierarchy (WMH) levels and technological readiness levels (TRL)[9] were looked at to acquire insight into the feasibility of the recycling processes.[51]

Before considering recycling, the direct reuse should be considered if possible. As the manufacturing of chopped strands from out-of-life woven fabric has been experimented and the properties retain to an acceptable standard, it is certainly possible to do that[23]. Reuse of CFRP is possible by dissolving the epoxy resin through soaking it in an alcohol solvent that separates the carbon fibers and the resin. However the mechanical properties of the material cannot be relied upon after this process. Additionally, reuse of a material cannot be considered in some cases for example when there is significant damage to the product which is not improbable for Peregrine as it can experience a crash at high speeds, hence recycling from CFRP waste must be taken into account.

Upon research, it was found that recycling of CFRP waste can be optimised through either microwave thermolysis[10] (thermal process) or through the use Peracetic acid (chemical process)[25][22].

The most promising thermal process that was identified is recycling through microwave thermolysis in which a uniform heat treatment process at $420\deg$ for a specific time interval renders CFRP residue at approximately 66%[25]. These thermal processes like pyrolysis and thermolysis are given high ratings in WMH and TRL levels, a rating of 8 at least in both criterion indicating the feasibility and availability of the technology that has to be employed[51].

A certain chemical process cannot be overlooked[11] if it brings about a closed-loop with **approximately** 100% recycling paradigm[71]. This is done through first immersing the composite waste in a solvent, ethylene glycol (EG), and by increasing the temperature, the epoxy matrix dissolves as the EG molecules engage in bond exchange reactions within the covalent adaptable network (CAN)[12], effectively breaking the long polymer chains into small segments. These carbon fibers can be then reclaimed with the same dimensions and mechanical properties as those of fresh ones (the recycled fibers retain 97% of the modulus and 95% of the tensile strength of the original fibers). "Further heating the dissolved solution leads to re-polymerisation of the epoxy matrix, so a new generation of composite can be fabricated by using the recycled fiber and epoxy"[71]. This process can be employed in the EOL phase so the whole structural framework is almost 100% recyclable. As no clear recyclability percentage has been outlined a safety factor can be applied and the percentage can be kept at 95%.

---

[9] TRL is a measure/indicator, developed by NASA, of estimating the maturity of the technologies used in a process.

[10] Thermolysis is the decomposition of the material in consideration through one or different heat treatments in presence of oxygen [25]

[11] Even though for chemical processes the TRL rating is 4 but WMH rating is 8.

[12] Reversible covalent bonds to enable the cross-linked network structure to respond chemically to any applied stimuli.

11.2.2. EPP

EPP EOL strategies can be decomposed into recycling, combustion and landfill. Furthermore EPP can be distinguished into scraps from the production and manufacturing process, called Post-Industrial Recyclate (PIR), and scraps from the customer's use, or the drone decomposition, called Post-Consumer Recyclate (PCR). The following strategies follow from the recycling guidelines reported by the ARPRO EPP manufacturing company [13].

Recycling of EPP relies on the collection, shredding and remolding of the scrap material. Since most collection is expected to occur in-plant or post-processing, this mainly affects PIR EPP. Using the ARPRO EPP, all of its constituent can be recycled, so long as it is collectable. Once collected, the EPP would get shredded such that the expanded PP pellets can be used again in a molding process as done in the original product's manufacturing. The process of remolding shredded EPP is called regrinding. Although all EPP can be regrinded, the quality of the regrinded product is dependent on the quality of processing equipment (degree of shredding, molding and curing temperatures). It has been experimented that for shape molding, the ratio of recycled EPP can reach up to 20%, whereas for block molding, higher ratios of up to 50% can be maintained without 'perceptible optical or physical change' [14]. Alternatively, the shredded EPP can go under a re-extrusion process to be melted into the PP resin, labelled Repro. Repro has the opportunity to be mixed in the initial PP resin, creating more ARPRO EPP. Repro can also be sold as a recycled PP raw material. If the EPP cannot be collected, such as PCR EPP, or cannot be adequately recycled due to insufficient equipment, combustion of EPP is the next EOL strategy.

Combustion of EPP gives an opportunity for energy recycling via the use of incineration. Polyethylene is a polyolefin, meaning it was produced from simple olyefin monomers. This fact means that under combustion, EPP will only generate water and carbon dioxide. Additionally, even though the PP base resin has a lower heat content (44 MJ/kg), the resin alone has a greater heat content than light fuel oil which is 42.5 MJ/kg. This means that EPP can be used as a fuel source for solid fuel co-combustion, indicating high energy recycling capabilities.

The last option for EPP's EOL is that of landfill. As mentioned, EPP is a polyolefin, which results in it taking an extensively long time to degrade due to chemical stability. This fact along with the valued extremely high re-usability that EPP imparts results in landfill being heavily avoided in EPP EOL.

Taking all these strategies into account, EPP can theoretically be 100% recyclable, however given the limitations on the proportion of recycled EPP that is acceptable prior to inducing potential flaws, a safety factor will be applied such that EPP will be considered as 90% recyclable. Given a total EPP mass of 34[g] per drone, this results in a recyclable mass of 30.6[g] and non-recyclable mass of 3.4[g].

11.2.3. Metal and Metal Alloys

In this section the EOL strategy for metals or metal alloys that will be used in Peregrine will be looked at to verify that these materials are indeed recyclable. 75% of all aluminium *ever produced* is still in use due to its durability and favourable recycling properties which saves 95% of the energy relative to the energy required for primary production.[9] In the recycling industry of Aluminium, new scrap is the material that arises during the manufacturing and fabrication of aluminium alloys to the point where they are sold to a final consumer which has a recycling rate of virtually 100%. The 'old scrap' refers to a aluminium product that has been discarded for recycling for which the recycling percentage is in the order of 75% [29]. With 220 aluminium recycling plants in Europe, no issue with aluminium recycling can be expected. [9] Copper is referred to as 'the recycling champion' as it is also possible to recycle all of the copper from both 'old' and 'new' scrap. Even though copper can be recycled fully, the overall recycling efficiency of 60% from International Copper Association can be taken and implemented in Table 11.4[15].

A life cycle study on stainless steel carried out covering 50 countries and territories concludes that on average 85% of stainless steel is recyclable at EOL[16]. This recycled steel is used by almost all sectors/industries. The process of recycling does not differ much for steel and stainless steel. This process is initiated by melting through the use of a furnace at temperatures of 3000° an purified to remove any contaminants to ensure steel with good mechanical properties is retained[17].

---

[13] `https://www.arpro.com/contentassets/6b02f2dc287c4ca9a769213593709ac1/techdoc_recyclingguidelines-epp_2018.pdf` Accessed on 18th June 2021

[14] `https://www.kbm-recycling.com/products/recycling-machines/comparison-of-recycling-units/` Accessed on 18th June 2021

[15] `https://copperalliance.eu/` Accessed on 18th June 2021

[16] `https://www.worldstainless.org/about-stainless/environment/recycling/` Accessed on 18th June 2021

[17] `https://www.rubicon.com/blog/steel-recycling/` Accessed on 18th June 2021

For composition of motors 20% of the masses of motors was assumed to be titanium hence an EOL strategy is also needed for it. Research was conducted on this material and its recyclability rate as well where it was concluded that the recyclability of Ti is above 90% which makes sense as it is an expensive metal and significant effort has been put into recovery of this material. Titanium is recycled by 'dilution with a virgin titanium sponge in the remelting process as long as the impurity concentration remain within an acceptable range. Electron-beam melting, plasma arc melting, and vacuum arc remelting are the general remelting technologies for titanium scrap because they can minimize oxygen contamination'. [64] Electrorefining (electrolysis process) to extract *pure* titanium is also performed in the process later.

As tin in the cables weighs a total of only 0.06[g], it would not contribute too much toward meeting the recyclability requirement, hence it can be neglected.

11.2.4. PTFE and PET

It was assumed that 20% of actuators' mass would be polyethylene terephthalate (PET) and the available product flow for the wires had 43.6% composition of PTFE (polytetrafluoroethylene) leading to a total thermoplastic weight of 16.72[g] hence it was necessary to make sure that these materials can be recycled to an acceptable extent.

Recycling of PTFE can be achieved through the use of pyrolysis, a chemical process. This process renders multiple chemical compounds through slow decomposition of polytetrafluoroethylene compounds occurring at high temperatures. This process gives high yields of TFE (tetrafluoroethylene) and HFP (hexafluoropropylene) [59]. TFE can then be used for synthesis of polytetrafluoroethylene(PTFE) resins, the same product from which it was decomposed. 'It can also be used as a monomer in the synthesis of copolymers and as a propellant for food product aerosols[18]. HFP, the second high yield product of pyrolysis, can be used for other applications specifically in the manufacture of fluoropolymers and has other agro-chemical and pharmaceutical applications [19]. It can safely concluded that PTFE is recyclable/re-usable. Nonetheless, as a recyclability percentage is required and the mass balance to determine exact the masses of recovered TFE and HFP is unfeasible with respect to the time constraint of the project, a reasonably lower recyclability percentage of 60% can be set for PTFE.

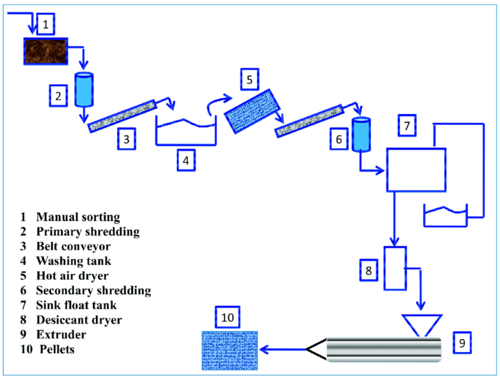


Figure 11.4: Step by step process of PET recycling

Upon research it was found that PET is easily 100% recyclable due to extensive applications of this thermoplastic but recycling rates for PET is mostly low in most countries. The process is outlined in Figure 11.4. However with a safety factor a recyclability percentage of 80% can be applied with confidence.

11.2.5. LiPo Battery

The weight of the battery changed since the previous design phase but as it is a major contributor to Peregrine's mass weighing 158[g], –% of the total weight, means that its end of life strategy will significantly contribute to the fulfilling of the system and power and propulsion subsytem recyclability requirements. The EOL strategy was already devised and discussed in the previous design phase. [4] However it will be reiterated here again briefly.

---

[18] `https://pubchem.ncbi.nlm.nih.gov/compound/Tetrafluoroethylene` Accessed on 18th June 2021

[19] `https://www.chemours.com/en/-/media/files/fluorointermediates/fluorointermediates-hfp-technical-product-info.pdf?rev=46414b01886c4cfdade6100813c0066c&hash=3346B732BCB6B47AC927DF495D2F12C7` Accessed on 18th June 2021

The recycling process can be divided into five individual steps. These steps are: collection, discharge, dismantling, mechanical pre-treatment and chemical treatment[22].

Using this five step process the materials, manganese, lithium, nickel and cobalt can be retrieved in the form of NMC crystals and lithium crystals, which are the chemical used in the anode and cathode of the battery. These chemicals are of the same quality as the raw material, meaning they can be inserted back at the start of the production line. The process used here is called a hydro-metallurgical process [20][22]. A similar hydrometallurgical recycling process involving a chemical precipitation methodology achieving a high recycling rate of 80% is employed by a company named Fortum [21]. The process of recycling the LiPo battery can be outsourced to the mentioned company so it is ensured that the disposal of the LiPo battery is done as sustainable as possible contributing to a more sustainable future.

11.3. Recyclability Requirements

As there is a top-level requirement to make the drone at least 80% recyclable (namely SYS-FEAS-SUST-01), it must be ensured that this is met. All requirements relevant to sustainability are as listed:

- **SYS-FEAS-SUST-01:** The drone must be 80% recyclable.
- **SYS-FEAS-SUST-03:**The drone shall be powered electrically.
- **SM-FEAS-SUST-01.1:**The load carrying structure shall be made from 95% recyclable material by weight.
- **PP-FEAS-SUST-01.1:**The battery shall be at least 75% recyclable by weight.
- **PP-FEAS-SUST-01.2:**The propulsion system shall be at least 70% recyclable by mass.

Table 11.4: Recyclability per material used or assumed.

Material/Component	Total Masses[g]	Recyclability [%]	Recyclable Mass[g]
CFRP	72.52	95%	68.88
EPP	34.0	90%	30.60
AlMg	58.0	75%	43.50
Stainless Steel	40.85	85%	34.72
Steel	11.60	85%	9.86
N52H arc magnets	23.20	*Re-usable*	23.20
Titanium	23.20	90%	20.88
Copper	6.21	60%	3.73
PTFE	4.84	60%	2.90
PET	10.0	80%	8.00
LiPo Battery	169.6	80%	135.68
Components Neglected	21.0	0%	0.0
Peregrine	475.0	**80.41%**	**381.95**

The final total mass of the Peregrine was determined to be 475[g] out of which 381.95[g] is recyclable if the EOL strategies discussed in Section 11.2 are adopted after Peregrine's operational life. This makes the Peregrine 80.41% recyclable, hence the system requirement namely SYS-FEAS-SUST-01 is successfully met. It must be realised that this requirement has been met without incorporating the masses of some components outlined before.

The power to the motors integrated is going to be provided by a LiPo battery meaning the system requirement SYS-FEAS-SUST-03 is already met. As the material being integrated, CFRP, is approximately 100% recyclable (95%) as outlined in Subsection 11.2.1, the requirement SM-FEAS-SUST-01.1 is adhered to.

The LiPo battery that is going to be integrated will be outsourced to a company named Fortum which specialises in recycling solutions and makes use of processes which renders Lithium-ion batteries more than 80% recyclable [22] therefore the requirement PP-FEAS-SUST-01.1 is also met.

---

[20] Chemical process using aqueous solutions

[21] `https://www.fortum.com/products-and-services/fortum-battery-solutions/recycling` Accessed on 26th May 2021
The technology was acquired from another company named CrisolteQ by Fortum.

[22] `https://www.fortum.com/products-and-services/fortum-battery-solutions/recycling` Accessed on 26th May 2021

The whole propulsion subsytem comprises of the battery, four motors, two actuators and four propellers leading to a mass of 327.6[g]. Other than the battery itself materials used for the subsytem are CFRP, AlMg alloy, titanium, N25H arc magnets, stainless steel and PET. Using Table 11.4 it can be calculated that the recyclable mass of the subsystem is 269.5[g] making it 82.26% recyclable. Hence PP-FEAS-SUST-01.2 is also complied with successfully.

Table 11.5: Compliance Matrix for requirements imposed on sustainability of Peregrine.

Identifier	**Compliance**
SYS-FEAS-SUST-01	Pass
SYS-FEAS-SUST-03	Pass
SM-FEAS-SUST-01.1	Pass
PP-FEAS-SUST-01.1	Pass
PP-FEAS-SUST-01.2	Pass

A requirement that was not directly imposed on the sustainability but is related to the noise generated by the drone at maximum speed (AE-ENVI-NOIS-01.1) which can be considered a requirement on social sustainability. As discussed in Subsection 5.3.9 this requirement will be verified in the testing phase of DSE as experimental data of a similar propeller could not be found.

11.4. Economic and Social Sustainability

Along with environmental aspects, sustainability encompasses economic and social perspectives as well. Economic sustainability deals with the maximization of profits by minimizing costs, whether that be manufacturing, operation, marketing or any other form of cost, and/ or increasing revenue. Social sustainability concerns the interaction of the system with surrounding environments and communities. Although these aspects are more dynamic and ambiguous than environmental aspects, they can still be assessed for total sustainability.

One example of cost reduction the system employed is through the inclusion of the material choice of EPP and CFRP as opposed to only CFRP. EPP production is significantly cheaper than CFRP. This is a result of the abundant supply and low cost of raw material in comparison and the ease of automation for forming processes, requiring less manual labor [23]. The costs are also further reduced via the use of simulation software as opposed to physical testing which would increase project time and energy costs, as such is the case for wind tunnel tests and prototyping. Granted these formats would give more accurate results, but in terms of sustainability, having the ability to model the system without the need for additional materials and costs is more sustainable. Profit maximization is done by having the drone release at a more competitively priced entry, rivaling other BNF drones which are cheaper than ARF drones, as mentioned in Chapter 4. Although this reduces profits per sale, it is expected to be overruled by the increased magnitude of sales due to being more budget-friendly, which will be helpful given the expected growth in the market. Future economic sustainability will be achieved by selling the drone components along with an ARF version of the drone, increasing longevity of the product as it adapts to the market's needs.

Socially, the drone adheres to the size and weight regulations for recreational drone use in the Netherlands, although other regulations rely more on user compliance [24]. Due to the radically unique and eco-friendly design, it is also expected that the Peregrine will garner and maintain the concerned society's interest and satisfaction. To ensure that the customers are consistently satisfied, the team hopes to give discounts and coupons to customers who give feedback on their customer experience and help in the retirement and disposal of the drone in the event of damage. This will also increase customer loyalty, which may aid in economic sustainability as a bonus. Even though the racing drone market is dominated by North America, the Peregrine seeks to be socially inclusive by targeting the European and Asian markets as well, increasing social sustainability.

---

[23] `https://www.imgplastec.com/products/epp/` Accessed on 15th June 2021

[24] `https://www.government.nl/topics/drone/rules-pertaining-to-recreational-use-of-drones` Accessed 17th June 2021

12

System Cohesion

This chapter gives a detailed description of the final system, where all subsystems are integrated and the remaining hardware and software on board Peregrine, as well as other aspects, are outlined. The choice of off-the-shelf components is given in Section 12.1, then the subsystems are integrated in Section 12.2. The final system is described in Section 12.6. In Section 12.3 the hardware components and their interactions are defined with the help of a hardware block diagram. Additionally, the software necessary to control Peregrine is given in a software block diagram. Next, the electrical connections are given in Section 12.4, after which the data flow between pilot and drone are described in Section 12.5. Finally, the hover transition of the drone is discussed in Section 12.7, followed by reliability in Section 12.8, safety in Section 12.9 and operations & logistics in Section 12.10.

12.1. Off-the-shelf Parts

The use of off-the-shelf components throughout the design of Peregrine is quite extensive. Because of this, more resources could be allocated to the hardware that makes peregrine different from current racing drones: The structure, control system, propeller and aerodynamic shape. However, in order to make the requirements, the correct parts should be selected. This section will go into each of the parts that have been selected and for each part give a short explanation why this part has been selected. First the video system through which the pilot will receive the live video feed from the drone will be discussed. An analog system was chosen for this, as this is still the benchmark in drone racing, thanks to it's superior and more consistent latency compared to digital systems. On top of that analog video systems can often be more compact and have a lower weight.

Camera
The camera is the first step in getting the video feed to the pilot. This means the camera should be of high quality as a good quality image is paramount for the pilots confidence and ability to fly the drone. However, high quality does not necessarily mean high resolution. It is mostly concerned with the amount of contrast the camera can cope with and the speed with which it adjusts to light changes. The chosen camera, the Runcam Nano 4, has a sufficient image quality. It's main advantages however are the small size (14x14x14 mm) and incredibly small mass (3.3 grams). These were the deciding factors in picking this camera.

Video Transmitter (VTX)
When the camera has generated the video signal, this signal needs to be transmitted to the pilot. This is the job of the video transmitter or VTX. In drone races, the 5.8 GHz frequency is used and from this frequency primarily the raceband or R band. Pilots are often required to choose one of the eight signals on this band. On top of that, there usually is an upper limit on the amount of output power the transmitter is allowed to use of 25 mW. When powering up before actually racing, a feature called 'Pitmode' is desired. This reduces the transmitting power significantly (to about 1 mW), decreasing the chance of interference with other pilots that might be racing at that point in time (as they might be on the same channel). Finally the video transmitter needs to have a feature called Smart Audio, where it communicates with the flight controller using it's audio channel. This will allow for changing settings on the transmitter through the flight controller. Because of this the VTX can be placed in hard to reach locations. The video transmitter that had all of these features, while staying light and relatively inexpensive, was the AKK Race VTX

Antenna (video)
The transmitting power from the VTX will need to go through an antenna so it is actually radiated into the environment. The antenna on the drone should be omni-directional (transmitting in all directions) and circularly polarized. This will ensure the signal can be received from any orientation of the drone (so not only when the drone is flying in a certain direction) and the signal will be more robust against multi-path interference (where the signal gets reflected and causes interference at the receiver end). Finally the antenna should be as durable and light as possible as this is usually a part on the exterior of the drone and will thus be more easily struck in a crash. The selected antenna, the Lumenier AXII 2, has been known as one of the smallest,

most durable antennas on the market.

Goggles
The signal from the antenna is received on the ground by the antenna of the video receiver or VRX. In the case of Peregrine this VRX is built into the included goggles. There are many different kind of goggles with some using one bigger screen and others using one screen for each eye. However the goal of each of these goggles is to show the video feed from the video receiver as immersively as possible. In general a big field of view (or FOV), or in other words how wide the picture is, and a high resolution (within limits since the video feed has a relatively low resolution) are desired. Goggles can be an expansive part, since pilots usually use them a long time. However for Peregrine the ready to fly kit should be competitively priced, thus limiting the choice of goggles to a midrange pair. Finally the Eachine EV200D goggles have been selected, as it includes a competent receiver, has a relatively high resolution and one the biggest FOVs in it's class.

With the video system out of the way, the components related to the control signal on Peregrine will be outlined.

Transmitter (control)
This signal starts with the commands of the pilot. These are transferred from his fingers through a radio controller or transmitter. This transmitter contains all the settings on the control signal and is the only 'physical' interface of the pilot with the drone. In general it is desired to have well built transmitter, that has enough inputs for the pilot to control the craft. Often a radio using OpenTX, an open source software used to configure the transmitter, is preferred. Finally the so called gimbals, parts of the radio that the pilot moves to convey his commands, should be hall effect gimabls, as they are more durable and precise than other comparable gimbals. The Radiomaster TX16S meets all of these desires in a cost effective package and will thus be included in the ready to fly kit.

Transmitter Module
The radio controller is now ready to send the pilot's command to the drone. However there are multiple ways of sending this signal. The protocol that is standard on the Radiomaster TX16S, is not the highest performing protocol. There are external transmitter modules that connect to the radio controller, which have a different protocol. These other protocols are faster and thus provide less latency. On top of that they generally also have more range than the standard protocol used inside the radio controllers. For Peregrine the Happymodel ES24TX was chosen. This is a transmitter module that runs an open source software called Express LRS, which is one of the fastest protocols on the market. On top of that receivers that receive this protocol are incredibly small, making them rather simple to package inside the drone.

Receiver (RX)
After the transmitter module has sent the signal through its included antenna, the signal goes to the receiver (RX) on the drone. This receiver should be compatible with the transmitter module, otherwise it will not be able to decode the signal sent by it. Furthermore, the receiver should be as compact as possible to make it easier to package. The chosen receiver, the Happymodel EP1, can communicate with the transmitter module and weighs only 0.42 grammes (without the antenna).

Flight Controller
The signal from the transmitter gets decoded by the receiver and will be sent to the flight controller. The flight controller will then translate these inputs to desired outputs from the motors and actuators using a PID loop. Most modern flight controllers also include a so-called on screen display or OSD. This part of the flight controller takes the video signal from the camera and overlays key information over it. Examples are battery voltage, amount of amperage drawn, throttle percentage etc. This information will help the pilot in judging the state of the drone. Finally, the flight controller should have sufficient computing power to run the PID loop adequately fast. The chosen flight controller, the HGLRC Zeus F722 mini, has the fastest processor available on flight controllers in this size range, while staying compact with a 20x20 mm mounting pattern.

With the signal inputted into the flight controller the flight controller will use the control surfaces and the motors to control and stabilise the drone. The control surfaces and outboard motors will be moved by the actuators, while the motors will be controlled by the speed controller.

Actuators
There are two different types of actuators, one will control the vectoring motors and the other will control the elevons. Both of the actuators or servos, should contain metal gears, as the more common plastic gears will not be strong enough to handle the loads that will be imposed on them. Since the motors are heavier than the control surfaces the servos for these motors will need to be bigger. For these servos the Bluebird BMS-212DMH is used, since it is known to be a reliable servo that has been used for vectoring the motors before. For the elevons a smaller, lighter servos could be used, so the Turnigy TGY-50090M was selected.

Speed Controller
The battery of the drone provides direct current power, while the motors work on alternating current power. The speed controllers or ESCs convert the power from the battery to the power the motors need, while also controlling the speed at which the motors spin. ESC's come in two forms, individual ESCs that control one motor and 4-in-1 ESCs, where 4 individual ESCs are mounted on a single board. The latter option is the lightest and is most easily integrated in the design. In choosing an ESC it is important that it can brake the motor (by using it as a generator) and reverse the motor direction, such that it can be used to generate reverse thrust. On top of that the ESC should be able to handle the voltage and current drawn by the motors. The selected ESC, the HAKRC 8B35A 35A, is a 20x20 mm mounting pattern 35A ESC, which has all the described features and provides ample current for the motors.

Charger
Since the drone flies on rechargeable LiPo Batteries the ready to fly kit should provide a charger to charge these batteries. Chargers can be powered in two different ways. Either they are powered from the electricity network through a wall outlet, or they are powered from a power supply, feeding it DC power. The latter way of powering is convenient at the flying field as this allows the charger to be hooked up to a car battery. The selected charger, the ISDT 608AC has a small power supply for charging at home, but can also be powered from a bigger battery at the field for field charging. On top of that it can provide enough power to charge the battery at a sufficient charging rate.

- Accelerometer
- Gyroscope (rotation rates)
- Pitot tube (velocity in cruise flight)

12.2. System Integration

Once every subsystem was able to finish their analysis, it was possible to integrate all subsystems into one final system. The integration was split into two parts: software and hardware. In this section integration methodologies for both will be discussed in detail.

12.2.1. Hardware Integration

The subsystems primarily responsible for design of the hardware for the drone were Aerodynamics, Structures & Materials, and Propulsion & Power. The main hardware integration was done by the Structures department through the CAD drawing. As the CAD drawing is a representation of the complete drone system, each hardware designed or selected by every subsystem needed to be integrated in the CAD drawing. Hence, in the process of completing the CAD drawing the hardware of each system was also integrated. As the mechanisms shown in the CAD drawing needed to be realistic, this integration process was slow and was done for each separate hardware component at a time. To start the CAD drawing the wing planform designed by the Aerodynamics department and constructed by the structures department was first created, it was followed by the design of elevons sized by Control & Stability and placement of propellers and battery selected by Power & Propulsion. Finally, the selected off the shelf mentioned in the above section relevant for placement on the drone such as the camera were created. Once every product was individually created using CAD, all hardware was assembled and integrated into one system by the structures department.

Once the CAD drawing was finished, the first iteration for the complete hardware of the drone could be performed. The aim of this iteration was to reduce the final mass of the drone to below 475 grams: 500 grams minus the defined contingency mass. If mass of the assembly of all hardware components created in CAD would be more than 475g then the whole iteration process would be carried out again. The hardware iteration process has been described in Figure 12.1. The team was able to integrate all hardware in the first iteration that means the assembly of all hardware components on the first attempt gave a mass of the drone which was below 475 grams. The result of this iteration will be show in the final system section below.

12.2.2. Software Integration

The subsystems primarily responsible for design of the software for the drone were the Control and Stability and the Performance and Simulation department. The control and stability subsystem were responsible for design of the PID controller that would receive inputs from the drone pilot and then perform control allocation to actuate the required actuators for carrying out the pilot's commands. Performance and Simulation were responsible for design of the simulation for the drone that would verify the top level requirements of the drone. This simulation as a result would have to include the dynamics of the drone. To integrate the software for

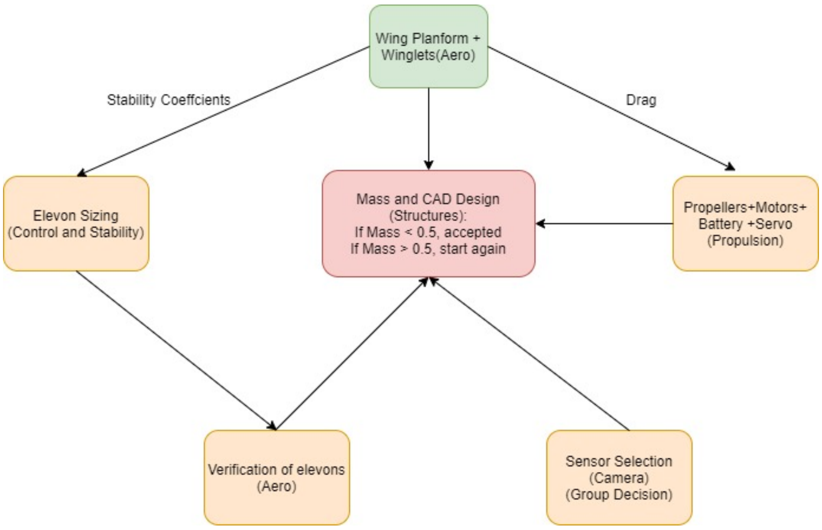


Figure 12.1: Hardware integration process for Peregrine

the drone it was required to integrate the dynamics model and the PID controller inside the simulation. To do so, both the model and controller were initially design and tested by each subsystem responsible for their design. Once the simulation had reached the level where the control surfaces of the drone could be controlled through the pilot's joystick and simulation was logical the PID controller could be integrated. The PID controller was placed in between the joystick and the dynamics block. Now the pilot joystick would not control the control surfaces on the drone but rather just issue commands desiring specific angular rates and throttle, the PID controller would then allocate these commands to specific actuators and calculate their deflection which would then be fed into the dynamics model. The control system for the Peregrine is quite complex and hence integration of the PID controller in the simulation was tedious and involved a lot of debugging. The simulation presented in Chapter 9 already has the PID controller implemented which proves that the software integration for Peregrine was successful.

12.3. Hardware and Software Flow

In addition to the hardware designed during this design phase, the drone will make use of off-the-shelf hardware. To visualise the interaction between the components, a hardware flow block diagram was made which is shown in Figure 12.2. Starting from the battery, which is connected to the rest of the system by means of an XT-60 type connector, power is delivered. Part of the battery's power goes to the flight controller, which based on the received pilot inputs by the RX system, gives power and a PWM signal to the servo actuators responsible for vectoring the propeller or deflecting the flaps. Furthermore, the flight controller gives four DSHOT-600 signals to the speed controller, which in turn spins the motors accordingly. The flight controller is also in charge of distributing power to the camera, transmitter, and receiver. The video feed recorded by the camera is sent to the flight controller, which adds an on-screen display (OSD) to the video feed. Finally, the feed is passed onto the video transmitter to be sent to the pilot.

Since the flight controller and ESC are off-the-shelf they are only compatible with certain firmware. In the case of the flight controller, this would be the open source firmware betaflight. Since betaflight is open source it will be possible to integrate the PID loop designed in Subsection 8.5.2 into the framework. Furthermore, betaflight will be responsible for the image processing necessary to add the OSD to the video feed. The ESC is compatible with BL Heli, which will be responsible for converting the DSHOT-600 signals containing the thrust levels to PWM signals which are used to power the motors. An overview of the various software run by the system are given by Figure 12.3.

12.4. Electrical Block Diagram

To ensure that the drone is powered correctly by the battery, an electrical block diagram was made showing the connections between components. This block diagram can be found in Figure 12.4. The battery has a nominal voltage of 22.2[V], however most of the smaller components run on 5[V]. Therefore, the flight controller steps down the voltage and supplies power to the servos, RX, VTX, and camera through its integrated 5[V] rail. Additionally, the flight controller also sends out PWM and DSHOT-600 signals to the actuators and

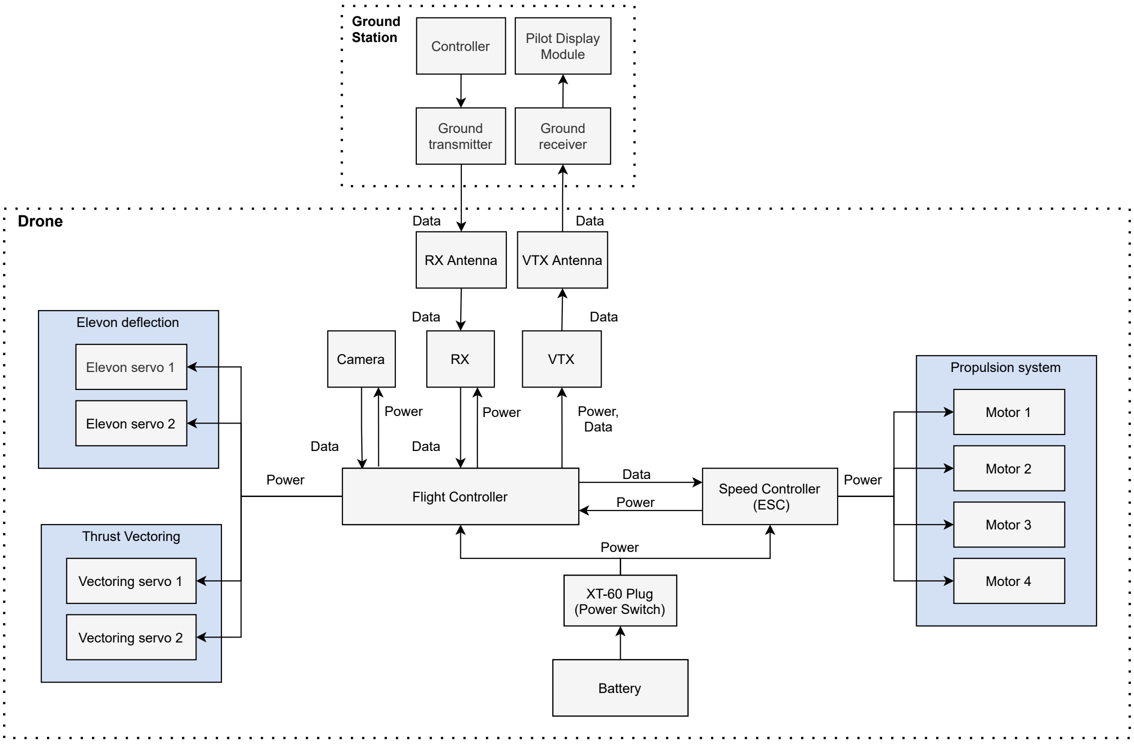


Figure 12.2: Hardware block flow diagram of Peregrine

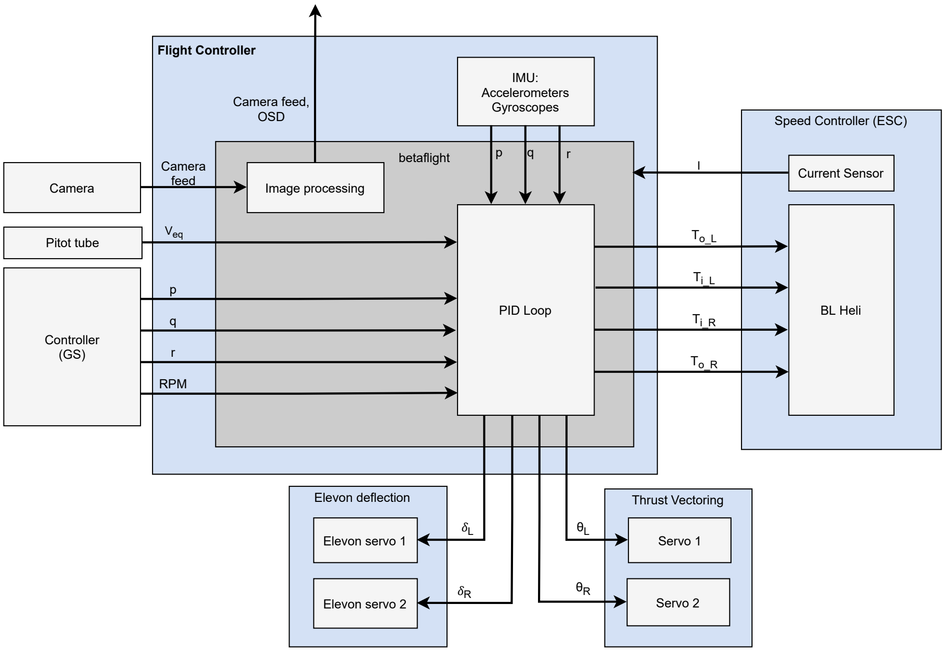


Figure 12.3: Software block diagram of Peregrine

ESC respectively. The ESC uses these signals in addition to the power provided directly by the battery to power the motors. Finally, the data received by the RX is passed onto the flight controller and the camera feed is relayed from the camera to the VTX as explained in Section 12.3.

12.5. Combined Data Handling and Communication Flow

To control the drone the pilot sends inputs via the controller to the drone. The ground transmission system is capable of sending signals at a rate of 25-500Hz and a frequency of 2.4[GHz]. The receiver on the drone passes these signals onto the flight controller as they are received. The flight controller contains a F7 microchip running at a frequency of 248 MHz. The flight controller has 16 MB of on-board memory

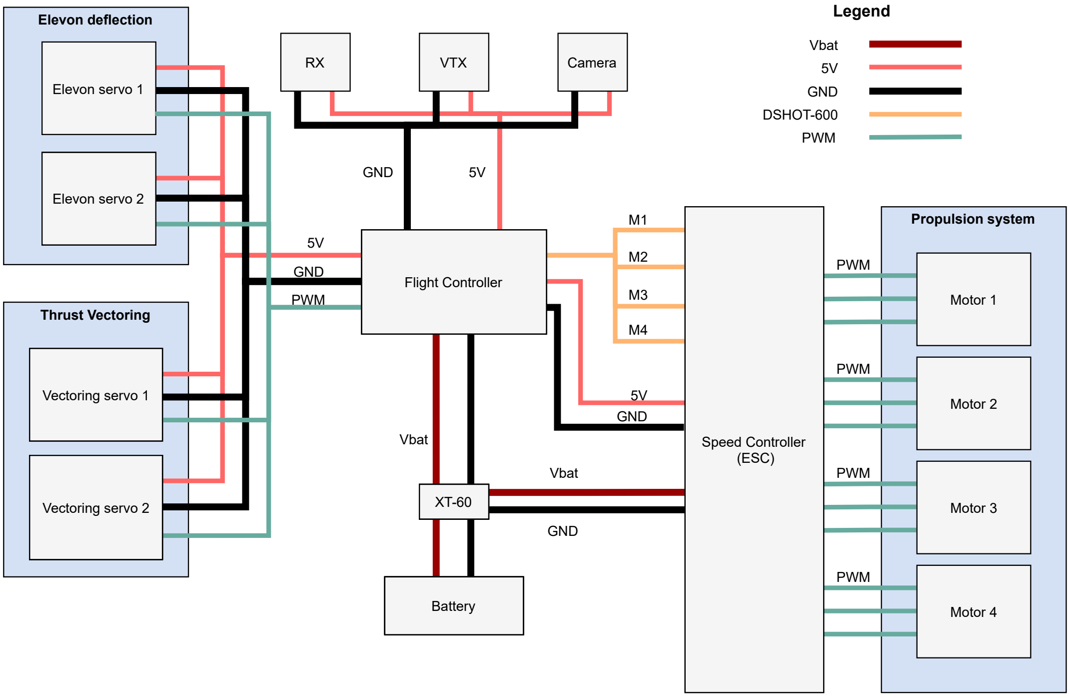


Figure 12.4: Electrical block diagram of Peregrine

and there is the possibility to store data, such as PID profiles by inserting a micro SD. The controller has integrated sensors running at a refresh rate of 4 kHz. Peregrine will also have a pitot tube to measure the equivalent airspeed of the drone which is used in the PID loop on the micro controller. The flight controller also sends PWM signals to the actuators and DSHOT-600 signals to the ESC. The ESC then converts these DSHOT-600 signals into 48kHz PWM signals going to the motors.

Finally, the feed of the analog camera with a horizontal TVL of 800 is passed onto the flight controller, where an on-screen display is added. TVL is defined as the maximum number of light and dark vertical lines an analog camera can display, the chosen camera is capable of displaying 800 TVL. Furthermore, the camera has a horizontal field of view of 155 degrees. Once the OSD has been added, the feed is transmitted to the pilot's goggles via the video transmitter which sends signals on the 5.8 GHz band.

It was decided to combine the Data Handling Block Diagram and the Communication Flow Diagram into one, due to the simplicity of the communication flow. The only communication happens between the ground station and the drone, and the data transferred between these elements would be displayed in the data handling diagram. The combined diagram can be found in Figure 12.5.

12.6. Final System

Once the final system integration was performed and the hardware, software and data flow of the drone were defined the final design for Peregrine can now be presented. Figures 12.6a, 12.6b and 12.7 shows the isometric view, top view and front view of the Peregrine racing drone respectively. The drone is equipped with 4 propellers and 2 elevons. The outboard propellers on the drone are able to vector to an angle of 45 degrees. The top speed of the drone is 90 [m/s] and the drone is able to perform a turn at 25g when it enters the turn at 60 [m/s]. Table 12.2 displays the most important parameters for Peregrine. The mass displayed in the table is the final mass of the system calculated from the CAD assembly file, with a detailed breakdown provided in Table 12.1. The maximum thrust per engine is achieved at a velocity of 53 [m/s]. The maximum C_{D_0} has been presented at a velocity of 45 [m/s].

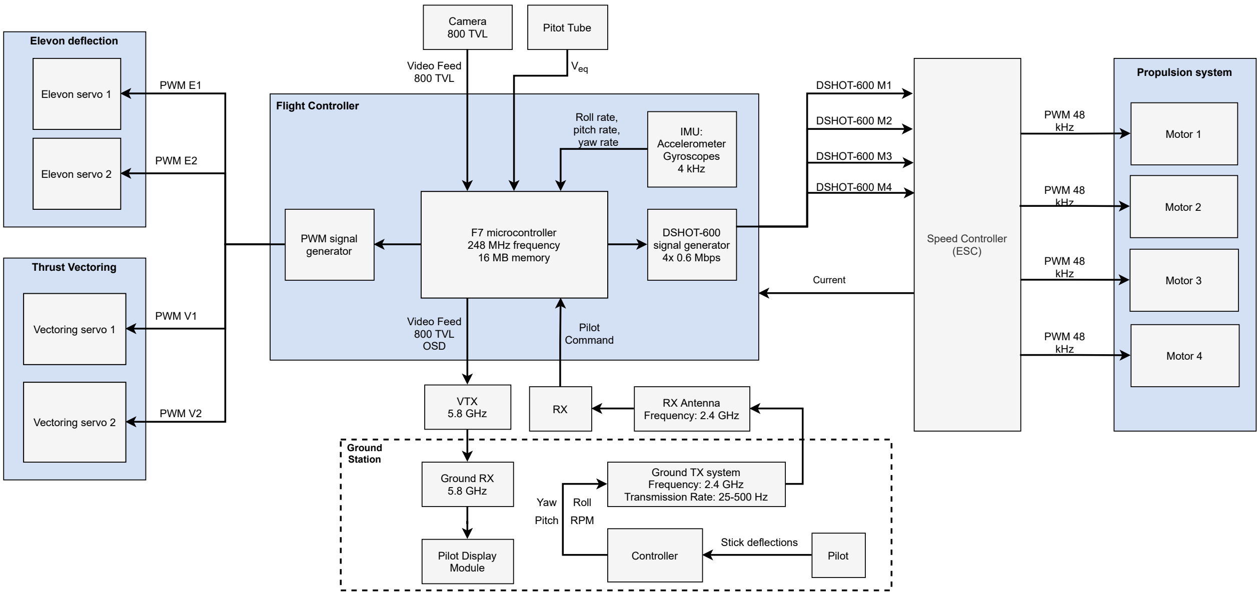


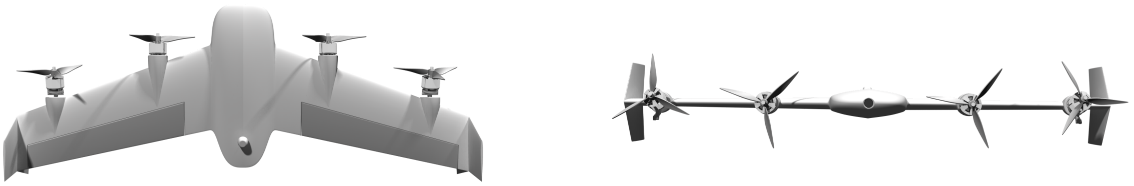
Figure 12.5: Combined Data flow and communication flow diagram of Peregrine

Table 12.1: Detailed mass breakdown of all the components accounted for in the CAD assembly file.

Part	Count	Mass [g]	Total mass [g]
Frame	1	45.1	45.1
Inboard plates	2	0.7	1.4
Control rods	2	3.0	6.0
Outboard hinge	2	1.0	2.0
Outboard motor mount	2	3.0	6.0
Servo horn	2	0.6	1.2
Linkage arm	2	0.6	1.2
Skin	2	13.0	26.0
Elevon	2	4.0	8.0
BLDC motors	4	29.0	116.0
Vector actuator	2	16.0	32.0
Elevon actuator	2	9.0	18.0
Propellers	4	2.5	10.0
Camera	1	3.0	3.0
Battery	1	169.6	169.6
Flight controller	1	4.0	4.0
ESC	1	6.0	6.0
RX	1	0.5	0.5
RX-antenna	1	0.5	0.5
VTX	1	4.0	4.0
VTA-antenna	1	3.0	3.0
24 AWG wires	1	8.5	8.5
14 AWG wires	1	2.6	2.6
Contingency	1	25	25
Total			**499.6**

Table 12.2: Final Design Parameters for Peregrine

Parameter	Value	Unit
Mass	0.4996	kg
T_{max} per engine	4.55	N
T_{max} reverse per engine	-1.9	N
$C_{L_{MAX}}$ at 60 m/s	0.9906	-
$C_{L_{MAX}}$ at stall	0.745	-
V_{stall}	13.5	m/s
C_{D_0}	0.0082	-
Maximum Load Factor	26.2	-
I_{xx}	0.013	kgm^2
I_{yy}	0.001	kgm^2
I_{zz}	0.014	kgm^2
x_{cg}	0.103877	m
y_{cg}	0	m
z_{cg}	0	m



(a) Top view of Peregrine

(b) Front view of Peregrine

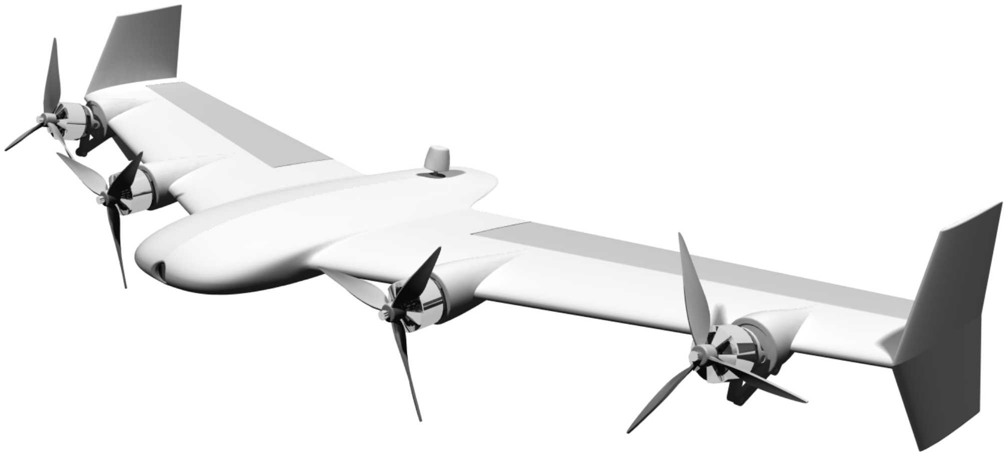


Figure 12.7: Isometric view of Peregrine

12.7. Hover & Transition

This section will describe the transition of the drone from resting in its initial position as a tail-sitter to cruise flight in horizontal position.

The drone has 4 propeller, of which 2 are inboard and 2 outboard. The outboard propellers can be vectored to generate vertical thrust. Figure 12.8 describes the hover transition for the drone. At position 1, the drone is sitting in its initial tail-sitting position. To begin hovering, initially all 4 propellers are actuated with a small thrust level. This slowly actuates the drone in it forward direction as shown in the transition from position 1 to position 2. The drone would hover in the air with all 4 of its propellers providing a small level of thrust and facing forward. During this hover period, the drone will be stabilised using a PID controller. The methodologies for stabilising the drone during hover have been described in the Control & Stability chapter, specifically Subsection 8.5.1.

To begin the transition to forward flight, the outboard propellers of the drone are vectored up to an angle of 45 degrees and inboard motors are switched off as shown in position 2. The vectored outboard propellers then generate thrust in the horizontal and vertical direction. As a result, the drone accelerates in the direction of the resultant thrust as can be seen from position 3. Since the outboard vectored propellers are behind the center of gravity, the thrust generated causes a clockwise moment and makes the drone tilt in the orientation as shown in position 3 for the drone. As the drone keeps tilting in the shown direction, the horizontal component of the drone starts becoming more dominant which straightens the drone, horizontally slowly transitioning to position 4 and then to position 5. At position 4 and 5 when the drone is almost horizontal, and not yet at stall speed, the vertical thrust component of the vectored propeller is responsible for generating the lift that keeps the drone at constant altitude and prevents it from falling. Once the drone is in position 5, the inboard motors are switched on again which accelerates the drone to a velocity beyond the stall speed and hence, the wings of the drone are able to generate the lift required to sustain the drone in straight flight. Until the drone reaches stall speed, the lift required to sustain the drone after position 5 is provided by the vertical component of the vectored thrust.

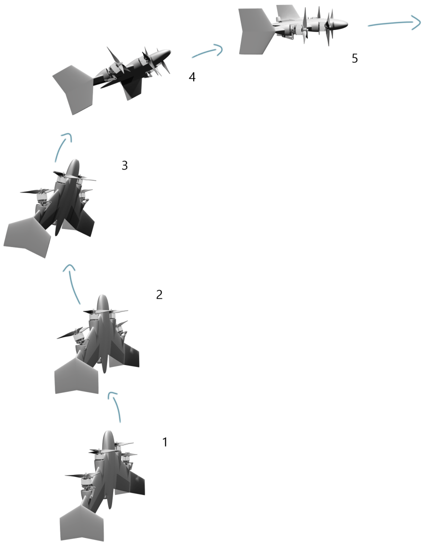


Figure 12.8: Hover transition for Peregrine

12.8. Reliability

Requirement SYS-OPRT-RLBL-01 states that the probability of a safety critical malfunction of the drone should be less than 0.001 per flight hour. It is understood from the requirement that when the drone is newly built with all parts recently installed, the probability of a safety critical failure of the drone for 1 hour of flight time should be less than 0.001. For verification of this requirement, all safety critical parts of the drone are considered. These involve the brushless motors, the servo motors, the drone structure, the electronic speed controller, the flight controller, CMOS image sensor, the battery and the electrical system involving receiver and the transmitter. The reliability data collected for all safety critical parts was found for a certain number of flight hours, to convert this data from several flight hours to one flight hour the following equation was used, where t-flight hours is the specific number of hours, reliability has been specified for.

$$Reliability_{1flighthour} = (Reliability_{t-flighthours})^{\frac{1}{t}} \tag{12.1}$$

The probability of failure of a typical brushless motor is calculated in [37]. It defines the probability of failure for a single brushless motor to be 0.05 after 5000 working hours. The probability of a failure in first hour for 4 motors is then calculated be 0.000041.

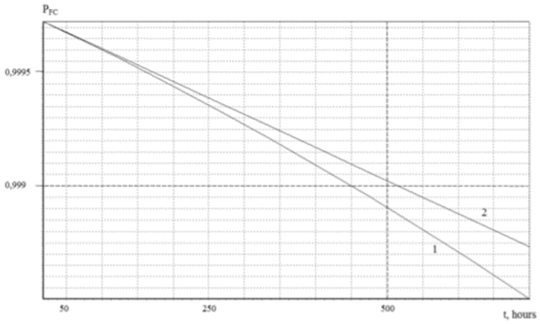
The servo motors are not considered to be safety critical parts of the drone as even if the vectoring fails the drone will still be able to fly due to its inboard motors and wing. If the outboard motors cause instability in the drone due to jamming of the servo motors, then it is expected that the PID controller would solve this stability issue.

The drone structure has a fatigue cycle of 82400 cycles. That means for ideal weather, the drone is expected to last at least 82400 cycles where a cycle is defined to be going in and out of a turn. Assuming that 1 hour of flying time has 800 cycles which means it makes a turn after every 5 seconds, it is expected that the drone structure will not fail due to fatigue in these many flying cycles.

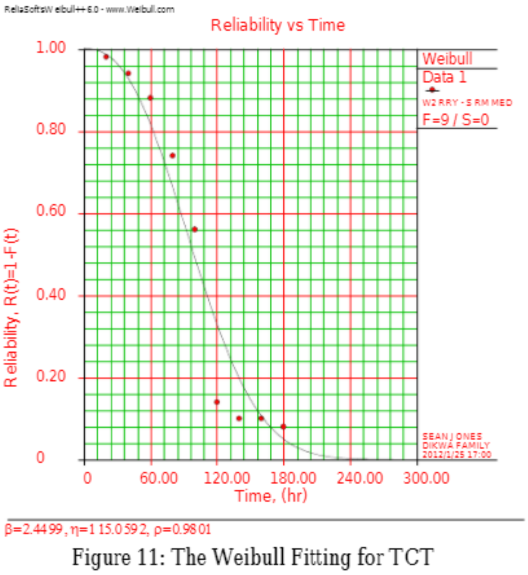
[7] states that the probability of a failure of a typical ESC used on a drone is 0.713 after 250,000 hours of use. For one hour the probability of failure for the same ESC can then be calculated to be 0.0000051.

[8] presents the rate of failure of a flight controller over time. This graph has been presented in Figure 12.9a. As visible from the graph, the reliability of the flight controller for 50 hours is 0.9995. Using this value, the reliability for 1 flight hour can be computed which is 0.99995. Hence, the probability of failure after 1 hour is 0.00005

[5] presents the reliability for a typical CMOS image sensor through Weibull fitting when the image sensor is subject to various temperature changes. As visible from the graph displayed in Figure 12.9b, the reliability of the image sensor for first 12 hours is approximately 1. Hence, it is assumed that the image sensor will not fail in the first hour. The battery for the drone is replaced frequently. For most drones the battery is considered replaceable and is mostly replaced when they are discharged. For this calculation it is assumed that the



(a) Flight controller reliability over time [8]



(b) CMOS sensor reliability over time [5]

battery would only be replaced when it is discharged and as the battery will not get discharged in the first hours, the reliability of the batter is taken to be 1.

[6] specifies the failure rate for an electrical system of a UAV which also involves the receiver and the transmitter to 5 for every million hours. Hence, its probability of failure per 1 hour can be computed to be 0.000005. Adding the probability of failure for all safety critical components gives the probability of failure of the drone for 1 hour which equals to $1.011*10^{-4}$. This verifies the system requirement for reliability. It should be noted that the above calculation is only a first order estimate of the reliability of the system for 1 hour. If more time is provided, it would be optimal to look into the reliability of specific parts being used in the drone even though the first order estimate performed in this section is for components very similar to the ones in the drone and the result is not expected to vary significantly. The reliability for safety critical components has also been measured for the first hour, with more time the reliability of these parts could be investigated after further into their lifetime.

12.9. Safety

Requirement **SYS-OPRT-FAIL-01** states that the drone should turn off the motors once a crash is detected. First the ways to detect a crash are explored and finally a possible way to implement this in the flight control software is outlined.

A crash can be detected in three ways, using different sensors on the drone. One way would be to measure the variation in power required to rotate a motor. If it suddenly changes, this might be a sign that the propeller broke after an impact. This requires using a different ESC, as the power must be monitored for each individual motor. Secondly, sudden large discrepancies between the PID target and angular rates or linear accelerations can indicate impact with an external body. The gyros and accelerometers required for these measurements are already part of flight controller. A third method to detect a crash used in other systems, strain sensors can be used in order to detect structural deformations. This will not be done on Peregrine.

With these three methods of detecting the crash, it still remains a challenge to know for sure that what has been detected is actually caused by a crash and not by the pilots commands. The turning acceleration can be as high as 25 g, which would be higher than a light crash. Because of this it is more useful to look at the difference between the attitude rate requested by the pilot and the achieved attitude rate. When the difference between these two is large the drone has been disturbed by an external influence. When this disturbance is big enough it can only be explained by a crash. Thus when the error in the PID loop is bigger than a predefined threshold the throttle should go to zero and the motors should be disarmed. Testing this threshold should be done in the prototype testing.

12.10. Operations & Logistics

A diagram showing the operations and logistics related to Peregrine, with respect to different stakeholders, is shown in Section 12.10. It outlines the flow from production, through storage & distribution, flight operations and maintenance until the end-of-life phases. In addition, it is colour-coded according the the project stakeholders that include supplier/contractor, logistics company, customer/pilot, safety/racing authorities, repairshop, recycling branch / waste company.

This process has directed attention towards some requirements in the future verification procedure. For example, SYS-PLTG-SIGN-03, allowing multiple drones to operate at the same time, will be of high importance for the Race Authorities. Due to its similarity with the functional flow diagram, Figure 3.1, no new requirements and functions of the drone were derived from this diagram.

The involvement of the separate stakeholders depend on the project group initiative. Therefore, entering into cooperation with the different stakeholders, and maintaining interactions between the stakeholders are tasks that must be planned for in the future. For example, the contracts and agreements should be made with the Supplier/Contractors, Logistics Company and the Waste Company, while the Customer should be reached via efficient marketing techniques. Further the Repair-shop and Recycling Branch should be initiated and maintained, requiring people and necessary funds.

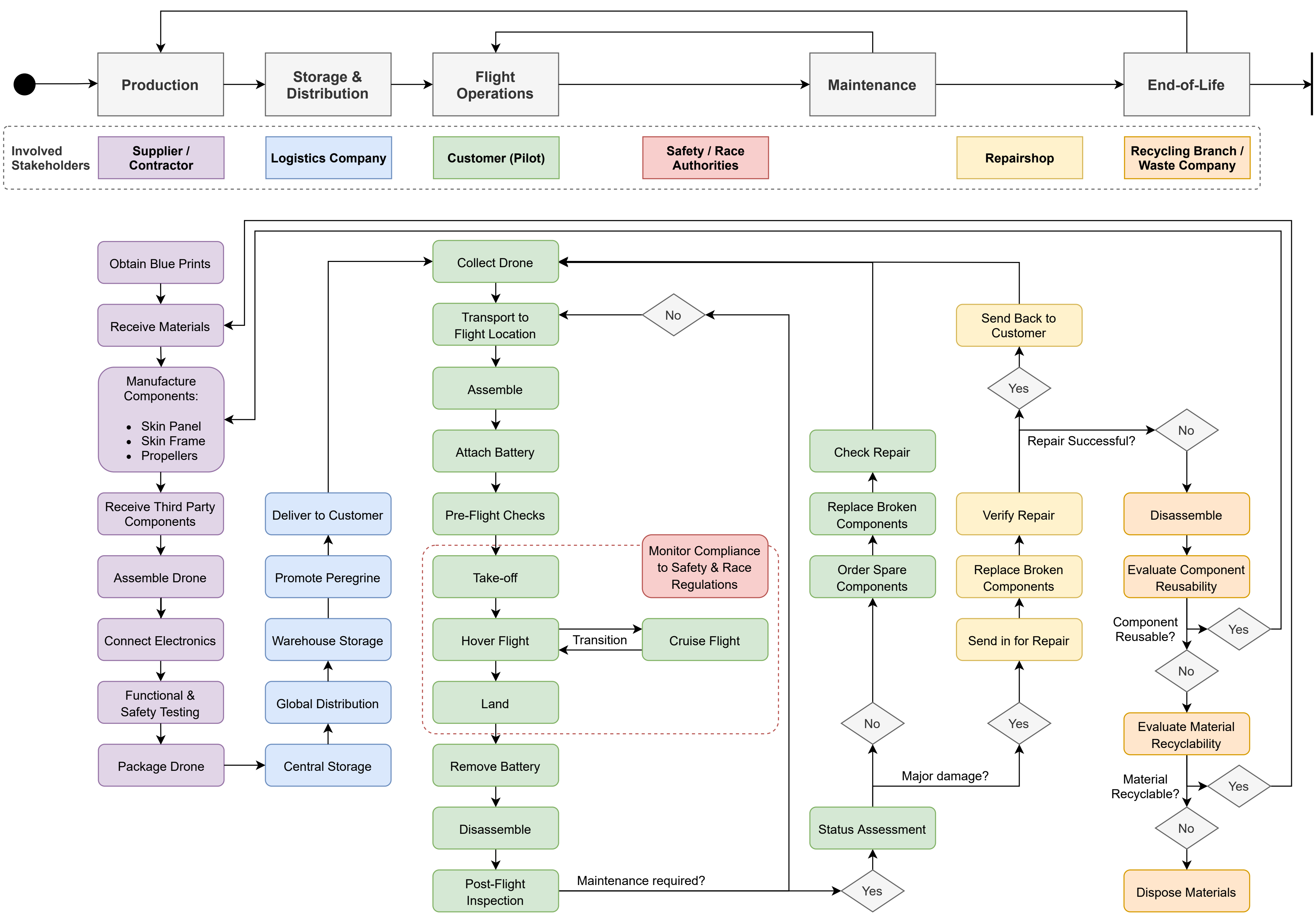


Figure 12.10: Operations & logistics diagram

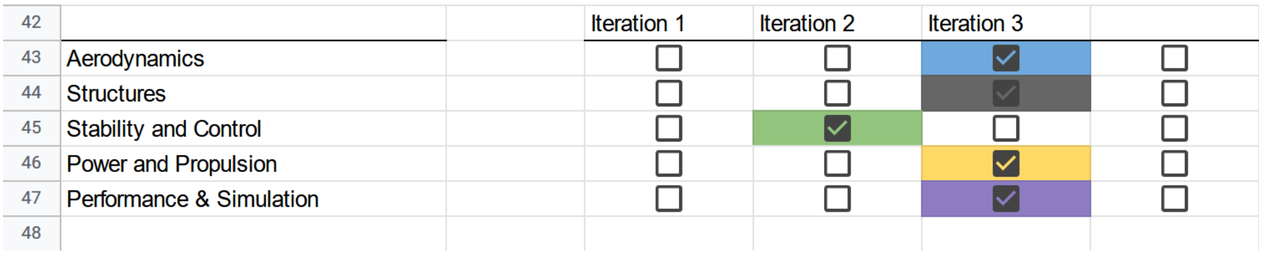
13

Verification & Validation

This chapter presents the verification and validation performed on the system level of Peregrine. Section 13.1 presents the procedures ensuring every subsystem staying up to date, and Section 13.2 presents the verification of the system level requirements.

13.1. System Verification

To ensure that all the subsystems were using the same values, an excel sheet was established where each of the subsystems posted their latest estimated values. The other subsystems then incorporated these values and ticked off the iteration they incorporated. An example of this is shown in Figure 13.1. Confirmation that each department was aware of the updated parameter values was asked at internal group meetings, and further inter-department communication was used to ensure the values were updated. Unfortunately, minutes were not made from these minor meetings.



42			Iteration 1	Iteration 2	Iteration 3	
43	Aerodynamics		☐	☐	☑	☐
44	Structures		☐	☐	☑	☐
45	Stability and Control		☐	☑	☐	☐
46	Power and Propulsion		☐	☐	☑	☐
47	Performance & Simulation		☐	☐	☑	☐
48						

Figure 13.1: Example of how the integration of parameters was tracked.

Code integration across subsystems were only performed between the Control & Stability and the Performance & Simulation departments. The verification of this process can be found in Subsection 9.6.1, under **Verification System Tests**.

13.2. Requirements Verification

This section runs through the system level requirements. Each requirements is either given a pass, to be confirmed (TBC) or a Fail. A method for verification will be given for each requirement which still needs to be confirmed. Additionally, the impact of the requirements which are not met will be discussed. The system level requirements, in addition to their current status can be found in Table 13.2. Requirements, which still need to be confirmed in addition to the procedure which will be used to verify said requirements are given in Table 13.1.

There are a total four requirements which failed verification. Each of the requirements is given below, additionally a short explanation is given as to why the requirement failed verification and what implications this has on the performance and or operation of the system.

- **SYS-FEAS-COST-03**: Comparing with a similar system, the battery would need to be replaced too often to meet this target. The estimated cost per flight hour amounts to €1.5 for the battery alone. Increasing the cost to €2 per flight would probably be sufficient to include other aspects.
- **SYS-FLGT-HOVE-04**: This is a poor requirement, as failing it does not rule out functionality of the system. Before the accelerating to above stall speed, one can rotate the wing to be parallel to the ground, decreasing the drag by a factor of 10. If one were to fly with a horizontal velocity of 15 [m/s] and the wing vertical, 9.26[N] of thrust would have to counteract 27.3[N] of drag.
- **SYS-FLGT-MOVE-02**: Including a safety factor of 1.5 on the required lift, a stall speed of 13.5 [m/s] is found instead. This does not necessarily hinder operations, but along with failing SYS-FLGT-HOVE-04

the transition phase become less flexible, and the vectoring propellers must be used to counteract the weight for an extended period of time.

- **SYS-STRD-RGLT-01**: Once the customer receives the drone, they are responsible for how they use the product. Guidelines can be provided, but it is impossible to enforce said guidelines with the customer.

The requirements still to be confirmed are presented in Table 13.1 along with the proposed verification procedure. The main reason these have not been addressed is the limited time and resources available at this stage of the project, but time has been allotted in Figure 14.4 to analyse and verify them. Some, such as SYS-PHYS-SPMT-03, require higher fidelity tools which must be learned, understood, verified and validated before the results can be used in the analysis. Others, such as SYS-ENVI-NOIS-01 require a physical drone.

Table 13.1: List of requirements remaining to be validated

Label	**Verification Procedure**
SYS-PHYS-SPMT-03	The analysis of this requirement is inconclusive from the Aerodynamics point of view. In order to properly verify it, time dependence must be modeled demanding higher fidelity models.
SYS-FLGT-HOVE-05	This requirement has not been properly analysed, but there have been stable designs similar to this.
SYS-FLGT-ROTA-05	Analyse the moment caused by thrust differential during nominal flight.
SYS-FLGT-ROTA-06	Analyse the moment caused by thrust differential during nominal flight.
SYS-FLGT-ROTA-09	Analyse the moment caused by thrust vectoring during nominal hover flight.
SYS-FLGT-ROTA-10	Analyse the moment caused by thrust vectoring during nominal hover flight.
SYS-FLGT-ROTA-11	Analyse the moment caused by thrust vectoring and elevons during nominal hover flight.
SYS-FLGT-ROTA-12	Analyse the moment caused by thrust vectoring and elevons during nominal hover flight.
SYS-FLGT-ROTA-13	Analyse the moment caused by thrust differential during nominal hover flight.
SYS-FLGT-ROTA-14	Analyse the moment caused by thrust differential during nominal hover flight.
SYS-PLTG-SIGN-01	Compare with a drone using similar RX and transmitter.
SYS-PLTG-SIGN-02	Compare with a drone using similar VTX and antenna.
SYS-PLTG-SIGN-03	Compare with drones using similar up- and downlink systems.
SYS-PLTG-INTR-03	Poor requirement, as it non-specific in terms of piloting skill and experience. The requirement still served a purpose, as it drew attention to the pilotability of the drone.
SYS-OPRT-PREP-02	Write a detailed user manual before product distribution.
SYS-OPRT-PREP-03	Have not yet been implemented for Peregrine.
SYS-OPRT-RLBL-01	Analyse reliability of all separate components and combine to get overall reliability.
SYS-OPRT-FAIL-01	Implement the solution proposed in Section 12.9 and test using crash tests.
SYS-OPRT-FAIL-02	Fly the drone into an obstruction at the specified velocity.
SYS-ENVI-WTHR-03	Subject drone to this temperature range.
SYS-ENVI-NOIS-01	Set up a microphone array and test the noise level in decibels of the drone at full thrust.

Table 13.2: System Level Requirements Compliance Matrix

Label	Origin	Priority	Requirement	Verification	Status
SYS-FEAS-COST-01	Stakeholders	M	The drone shall not exceed a production cost of €2500.	Review	Pass
SYS-FEAS-COST-03	Stakeholders	W	The drone shall not exceed a operation cost of 1€ per flight hour.	Analysis	Fail
SYS-FEAS-SUST-01	Sustainability	M	The drone shall be 80% recyclable by weight.	Review	Pass
SYS-FEAS-SUST-03	Sustainability	M	The drone shall be powered electrically.	Review	Pass
SYS-PHYS-WGHT-01	Stakeholders	M	The drone shall not exceed a total airborne mass of 500 [g].	Analysis	Pass
SYS-PHYS-SIZE-01	Constraints	M	The drone shall have a longitudinal cross-section that fits inside a circle of diameter 0.7 [m].	Review	Pass
SYS-PHYS-SPMT-01	Constraints	M	The drone shall have a design complexity that allows for manufacturing using existing technologies.	Review	Pass
SYS-PHYS-SPMT-02	Functions	M	The drone shall not deform plastically under stresses due to maximum load conditions.	Analysis	Pass
SYS-PHYS-SPMT-03	Functions	S	The drone shall not experience unstable structural oscilations due to aeroelasticity.	Analysis	TBC
SYS-FLGT-HOVE-01	Functions	S	The drone shall be able to accelerate upwards at a rate of at least 25 [m/s²] in hover flight.	Analysis	Pass
SYS-FLGT-HOVE-02	Functions	S	The drone shall transition from hover to forward flight at minimum speed in less than 5 [s].	Analysis	Pass
SYS-FLGT-HOVE-03	Functions	S	The drone shall transition from forward flight at minimum speed to hover in less than 10 [s].	Analysis	Pass
SYS-FLGT-HOVE-04	Functions	S	The drone shall have a maximum horizontal velocity in hover flight of at least 15 [m/s].	Analysis	Fail
SYS-FLGT-HOVE-05	Functions	S	The drone shall be stable while hovering.	Analysis	TBC
SYS-FLGT-MOVE-01	Functions	M	The drone shall have a maximum horizontal velocity in forward flight of at least 90 [m/s].	Analysis	Pass
SYS-FLGT-MOVE-02	Functions	S	The drone shall have a minimum speed in forward flight of at most 12 [m/s].	Analysis	Fail
SYS-FLGT-MOVE-03	Functions	S	The drone shall be able to accelerate from minimum to maximum speed in forward flight in 5 [s].	Analysis	Pass
SYS-FLGT-MOVE-04	Functions	S	The drone shall be able to decelerate from maximum to minimum speed in forward flight in 8 [s].	Analysis	Pass
SYS-FLGT-MOVE-05	Functions	S	The drone shall be stable in forward flight.	Analysis	Pass
SYS-FLGT-ROTA-01	Stakeholders	M	The drone shall be able to perform a 25g turn in forward flight at a velocity of 60 [m/s].	Analysis	Pass
SYS-FLGT-ROTA-02	Functions	S	The drone shall be able to transition from horizontal, symmetric forward flight into a 25g turn at 60 [m/s] within 1 [s].	Analysis	Pass
SYS-FLGT-ROTA-03	Functions	S	The drone shall be able to obtain a pitch rate of at least 1000 [°/s] from nominal forward flight.	Analysis	Pass
SYS-FLGT-ROTA-04	Functions	S	The drone shall be able to achieve max pitch rate from nominal forward flight in 1 [s]	Analysis	Pass
SYS-FLGT-ROTA-05	Functions	S	The drone shall be able to obtain a yaw rate of at least 300 [°/s] from nominal forward flight.	Analysis	TBC
SYS-FLGT-ROTA-06	Functions	S	The drone shall be able to achieve max yaw rate from nominal forward flight in 5 [s]	Analysis	TBC
SYS-FLGT-ROTA-07	Functions	S	The drone shall be able to obtain a roll rate of at least 1080 [°/s] from nominal forward flight	Analysis	Pass
SYS-FLGT-ROTA-08	Functions	S	The drone shall be able to achieve max roll rate from nominal forward flight in 1 [s]	Analysis	Pass
SYS-FLGT-ROTA-09	Functions	S	The drone shall be able to obtain a pitch rate of at least 180 [°/s] from nominal hover flight.	Analysis	TBC
SYS-FLGT-ROTA-10	Functions	S	The drone shall be able to achieve max pitch rate from nominal hover flight in 1 [s].	Analysis	TBC
SYS-FLGT-ROTA-11	Functions	S	The drone shall be able to obtain a yaw rate of at least 180 [°/s] from nominal hover flight.	Analysis	TBC
SYS-FLGT-ROTA-12	Functions	S	The drone shall be able to achieve max yaw rate from nominal hover flight in 1 [s].	Analysis	TBC
SYS-FLGT-ROTA-13	Functions	S	The drone shall be able to obtain a roll rate of at least 90 [°/s] in nominal hover flight.	Analysis	TBC
SYS-FLGT-ROTA-14	Functions	S	The drone shall be able to achieve max roll rate from nominal hover flight in 1 [s].	Analysis	TBC
SYS-FLGT-ENDR-01	Functions	S	The drone shall be able to hover at ISA sea level conditions for at least 3 [min].	Analysis	Pass

Label	Origin	Priority	Requirement	Verification	Status
SYS-FLGT-ENDR-02	Functions	M	The drone shall be able to fly a UTT1 Time Trial flight profile for at least 4 [min].	Analysis	Pass
SYS-PLTG-SIGN-01	Stakeholders	M	The drone shall be able to operate at a maximum distance from the controller of at least 750 [m].	Similarity	TBC
SYS-PLTG-SIGN-02	Stakeholders	M	The drone shall provide a video downlink up to a distance of at least 750 [m].	Similarity	TBC
SYS-PLTG-SIGN-03	Stakeholders	C	The system shall allow up to at least 4 drones shall be able to fly simultaneously without signal interference.	Similarity	TBC
SYS-PLTG-INTR-01	Stakeholders	M	The drone shall enable the pilot to use a controller to input remote commands.	Review	Pass
SYS-PLTG-INTR-02	Stakeholders	M	The drone shall provide video feedback to the pilot via first-person-view goggles.	Review	Pass
SYS-PLTG-INTR-03	Functions	C	The drone shall not require training that exceeds 5 [h] in the simulator for experienced pilots.	Review	TBC
SYS-PLTG-STAT-01	Functions	M	The drone shall be able to measure its attitude rates.	Review	Pass
SYS-PLTG-STAT-02	Functions	M	The drone shall be able to measure its acceleration.	Review	Pass
SYS-PLTG-STAT-03	Functions	S	The drone shall have a maximum of 8 input types from the interface.	Review	Pass
SYS-PLTG-STAT-04	Functions	S	The drone shall limit the amount of output types to interface to 10.	Review	Pass
SYS-OPRT-TRAN-01	Stakeholders	C	The drone shall be able to be assembled after transport in less than 10 [min].	Review	Pass
SYS-OPRT-TRAN-02	Stakeholders	C	The drone shall fit into a space of 0.7x0.2x0.3 [m^3] when disassembled.	Review	Pass
SYS-OPRT-TRAN-03	Stakeholders	C	The drone shall be designed such that it can be transported without damage.	Analysis	Pass
SYS-OPRT-MAIN-01	Functional	C	The drone shall allow all external surfaces to be cleaned superficially.	Review	Pass
SYS-OPRT-MAIN-02	Functional	S	The drone shall have detachable connections for components that need to be accessed during maintenance.	Review	Pass
SYS-OPRT-PREP-01	Functional	W	The drone shall be able to be set up when first received by the user within 20 [min].	Review	Pass
SYS-OPRT-PREP-02	Stakeholders	W	The drone shall be detailed in a user manual.	Review	TBC
SYS-OPRT-PREP-03	Functional	W	The drone shall, when turned on, follow an automatic start up procedure including pre-flight system checks.	Review	TBC
SYS-OPRT-RLBL-01	Stakeholders	M	The drone shall have a safety-critical malfunction probality lower than 1E-3 per hour of flight time.	Analysis	TBC
SYS-OPRT-Fail-01	Risks	M	The drone shall stop the rotation of its motors upon impact.	Review	TBC
SYS-OPRT-Fail-02	Risks	S	The drone shall not be damaged after an impact at 20 [m/s].	Analysis	TBC
SYS-STRD-RGLT-01	Stakeholders	C	The drone shall not be used for surveillance or other unauthorised purposes.	Similarity	Fail
SYS-STRD-RGLT-02	Stakeholders	S	The drone shall adhere to the European drone regulations A1.	Review	Pass
SYS-STRD-RGLT-03	Stakeholders	C	The drone shall adhere to relevant drone competition rules.	Review	Pass
SYS-STRD-CSTM-01	Market	S	The drone shall be compatible with standard drone equipment.	Review	Pass
SYS-STRD-CSTM-02	Market	C	The drone shall be designed as a complete ready-to-fly kit.	Review	Pass
SYS-STRD-CSTM-03	Market	C	The drone shall enable customisation by allowing customers to purchase components separately.	Review	Pass
SYS-ENVI-WTHR-01	Functions	C	The drone shall be able to withstand water splashes from any direction.	Review	Pass
SYS-ENVI-WTHR-02	Functions	S	The drone shall be able to fly under a maximum crosswind of at least 5 [m/s].	Analysis	Pass
SYS-ENVI-WTHR-03	Functions	S	The drone shall be able to operate within a static temperature range of -10 to 50 [°C].	Analysis	TBC
SYS-ENVI-ALTD-01	Functions	S	The drone shall be able to operate at a maximum ISA altitude of at least 1 [km].	Analysis	Pass
SYS-ENVI-FATG-01	Functions	C	Drone shall be able to perform at least 150 flying cycles for [Pa] stress range.	Review	Pass
SYS-ENVI-NOIS-01	Stakeholders	C	The drone shall not exceed a noise level of 90[dB].	Similarity	TBC

14

Future Development Strategy

This chapter contains the planning of activities to be performed after the DSE until product end of life. First the production plan is presented in Section 14.1 before the project design and development logic, project Gantt chart and the cost breakdown structure follows in Sections 14.2 to 14.4.

14.1. Manufacturing Assembly & Integration Plan

To understand how the drone can best be produced, a Manufacturing Assembly & Integration Plan is produced. This plan is best detailed through a flow diagram, shown in Figure 14.2.

The manufacturing stage will involve the manufacturing of thee types of components. Skin Panels made from EPP will be made through injection molding, whereby polypropylene pellets are expanded and then autoclaved in the mold. These panels will include the bottom and top main wing panels, winglets, elevons and electrical hatch covers. The structural frame will be manufactured using injection resin molding. A fiber layup will be made and pre-formed, before being placed into the mold for the resin infusion process. Finally the custom size propellers will be injection molded. Parts bought from third-party suppliers are brought in without further manufacturing steps.

The assembly stage consists of a series of consecutive steps, all of which are to be completed for every drone. The steps can be completed at different stations, where the aim is to have a similar through put time for efficiency. Assembly starts with placing the elevon mechanism in the frame, as this cannot be done at a later stage. Then all the electronics are installed step by step: first the antennae and camera, and last is the motor and tilting mechanism. After the assembly the software integration and quality control follow. Quality control consists of acceptance testing, where the desired performance of every drone is ensured.

14.2. Project Design & Development Logic

The design stage does not end with the DSE. In order to enter the market successfully, there are a number of activities that need to be performed, as outlined in Figure 14.3. Activities are divided into four phases: Corporation Development, Product Development, Product Manufacturing and Operations. Note that the Corporation Development runs in parallel with the other. For the third level tasks the flow has been visualized with arrows. Arrow absence means the task might be performed in sequence or parallel with other tasks.

Corporation Development
Corporation Development is concerned with handling external relations by making and maintaining contact with companies and communities of interest. For example, it contains tasks aimed at gaining attention in the public space, such as visiting drone fairs and high profile races. Further, acquiring funds and partners is necessary for further product development and a successful market delivery. Detailing and executing the marketing strategy are also tasks to be found in this branch, along with designing and maintaining the interfaces through which the customer and partner communication can happen. Although placed on a different branch, it is of prime importance that the time of task execution is well coordinated with the Product Development, as displayed in Figure 14.4.

Technical Development
Technical Development is the post DSE design phase. This includes the modelling, optimising and testing of the system. Modelling examples are a dynamic flutter analysis or performing viscous CFD on the full system. Optimisations are then performed on these or prior results to finalise the system. When the final design is done, all the components needed to assemble a prototype can be acquired. Part of these components will be off the shelf components that need to be purchased from partners or other suppliers (such as the motors), and part of them will be manufactured in-house such as the tilting mechanism. Quality checks will be performed on these components to ensure quality and compliance. Expensive tests, such as windtunnel tests, are omitted because of the inexpensive prototypes. Testing and iterating prototypes will be cheaper

than expensive subsystem tests.

After all the components have been tested in the subsystem tests the final prototype can be assembled. This will be done according to the production plan outlined in Section 14.1. With the prototype built it will be subjected to a significant amount of tests, ranging from testing the endurance (or flight time) of the drone to its crash resistance. Beside this in-house testing, prototypes will also be sent out to a select group of test pilots, who will fly the drone and give feedback to the design team. Finally the outcome of these tests, that will be checked for their validity, will be used to determine if this iteration meets the requirements. If it does not meet the requirements, a new iteration will need to be performed. This iteration will start over at the post DSE design level, starting again with the analysis, detail design and test design. If the iteration did meet the requirements the design can be finalised by completing the documentation, finishing the certification and completing the final drawings. After the finalisation the manufacturing stage can begin.

Product Manufacturing
Product Manufacturing is the phase in which the final design will be delivered to the public. This phase contains activities further detailed in Section 14.1. First the logistics and storage will be planned and orders for third party components will be placed, before manufacturing and quality control can take place. The third party company orders should be made in advance, such that the first delivery arrives at the production site on time for production start. The same holds for the production planning. Two main types of quality control will be executed, one process focused, ensuring sufficient quality of every drone, and random sample testing to detect deteriorating practices in the production procedure.

Operations
The manufactured drones will be stored in a warehouse. Based on the current warehouse stock and market demand, drones may be distributed, orders for more drones can be made, or the product can be phased out. During and after the phase of drone distribution effort should be made to update the product with better or cheaper components and software, ensuring utilisation of state of the art technologies and prolonged market demand. The distributed drones will then be recycled, where part of the recycled materials may be re-used in new drones. At last the product will be phased out market demand decreases and production facilities will be needed for other products. A while after this, the customer support for the product will also be phased out, allowing relocation of resources to more novel products and ideas.

14.3. Project Gantt-Chart

All of the tasks listed in the Project Design and Development Logic have been put on a preliminary timeline. The result of this can be seen in Figure 14.4. Following the DSE, the project is expected to continue in September. From there, it is assumed that work will be performed five days per week, for eight hours per day. No holidays are included, but as tasks are not yet allocated to individual team members, so individual team members might be able to take individual vacations in less busy periods. There will be a concentration of tasks before the launch, since afterwards the sale of Peregrine will start and the launch itself needs to be planned and executed.

Some tasks will have a throughput time different from what is displayed in the chart. For example, acquiring partners was given a period of 80 days, because it was assumed necessary to ensure funding to continue the project before investing too much effort in it. However, all the tasks concerned with acquiring partners will not each demand 80 days of continuous attention. The tasks concerned with purchasing off the shelf components on the other hand, will have a longer throughput time than displayed in the chart. These tasks will combine about a day of effort, but the delivery is assumed to take about three weeks. Therefore the testing of these components can not start right after purchase.

The time allocated for the tasks related to production and customer support is too short, and the tasks are expected to go on for longer than displayed in the chart. This underestimation is done to the scaling of the chart, where details would be lost in the chart for shorter tasks. The longest task, recycling the drones, was limited to 140 days, but is expected to be in the magnitude of years.

14.4. Cost Breakdown Structure

Figure 14.1 displays all the costs that will be incurred after the completion of the design phase in the DSE. These costs were divided in the categories as shown: Further Design, Suppliers, Production, Testing, Distribution, Marketing, Customer Service and Disposal. The further design cost mostly stem from the product development activities, where more detailed analysis and design will be performed. Also stemming from this phase are the testing costs, that prove to be quite significant. The costs under 'Suppliers' and 'Production' are mostly related to the production of the drone, however do note the costs for obtaining the contracts and deals with these suppliers are included as well. Finally the costs under 'Distribution' are most closely related with the operations phase and the costs under 'Marketing', 'Customer Service' and 'Disposal' stem from the Corporation Development.

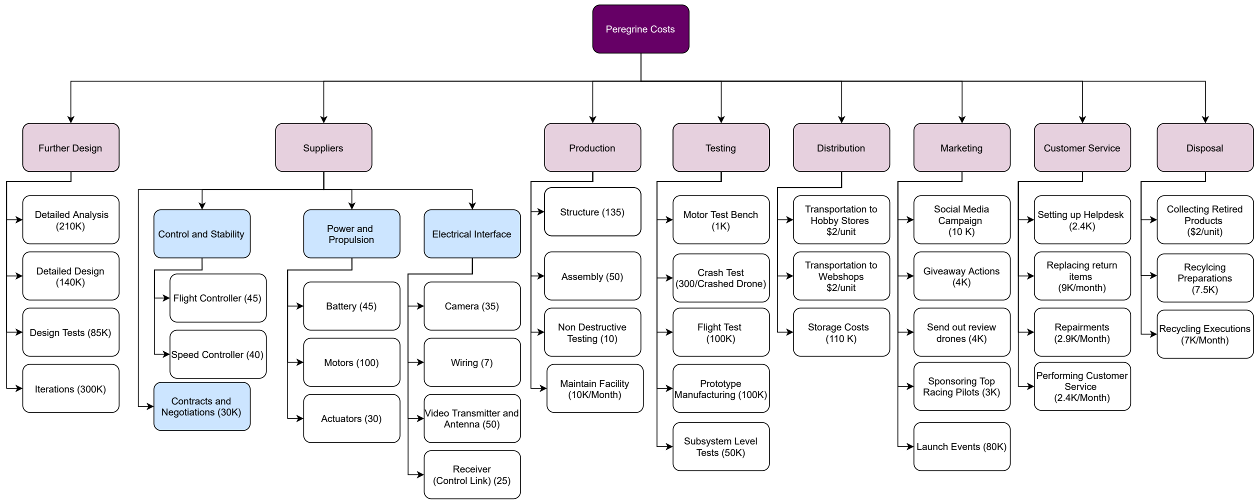


Figure 14.1: Post-DSE cost breakdown structure

Apart from the different cost adding components, also the amount these components will cost is displayed in the figure. Note that there are multiple different types of cost. Some single time, such as development costs, while others are recurring, like the off the shelf parts, or customer support. To come up with these costs three different approaches were taken:

- Prices from the cost budget in Table 2.1.
- Duration of these tasks from the Gantt chart (Figure 14.4), with a suitable amount of people and their hourly rate (€150 for an engineer and €15 for a non-engineer)
- Estimates from external sources (storage costs[1], Transportation costs[2], Recycling Preparation[3] and Social Media costs[4])

Market analysis gave an annual production of 45 thousand drones for the year 2023, yielding a monthly production rate of 3750. Assuming 15 thousand drones to be manufactured before 1st of September 2022, 4.5 months into operations, the first year costs could be estimated, as displayed in Table 14.1. This amounts to €10.15 million. With a cost per drone would be €678 excluding transmitter and goggles, the break-even point would be reached within this year. Assuming a cost of €620 per drone would yield a deficit after one year of €860 thousand, and 53 thousand drones would have to be sold in order to reach the break even point. Thus the price of the drone significantly affects the economic sustainability of the Peregrine project.

The prices for off the shelf components have been found in online hobby stores. The hope is that the price per piece will go down when when they are ordered in bulk directly from the manufacturer. With the current assumed costs, the manufacturing price per drone is about €600. Including the additional components that allow this to be a BNF drone results in a total cost of **€1115** per drone produced. Given the assumed retail price of €1250 as mentioned in Section 4.4, this results in €135 profit per drone sold, or 12.1%.

Table 14.1: First Year Costs

Cost Type	Amount
Single time	€1.15M
Recurring	€9M
Total	€10.15M

---

[1] https://strongbuildingsystems.com/how-much-does-it-cost-to-build-a-warehouse/ Accessed 21st June 2021

[2] https://quicargo.com/nl/transportkosten-berekenen/ Accessed on 21st June 2021

[3] https://www.generalkinematics.com/blog/cost-starting-full-force-recycling-program/ Accessed on 21st June 2021

[4] https://www.upwork.com/resources/social-media-campaign-cost Accessed on 21st June 2021

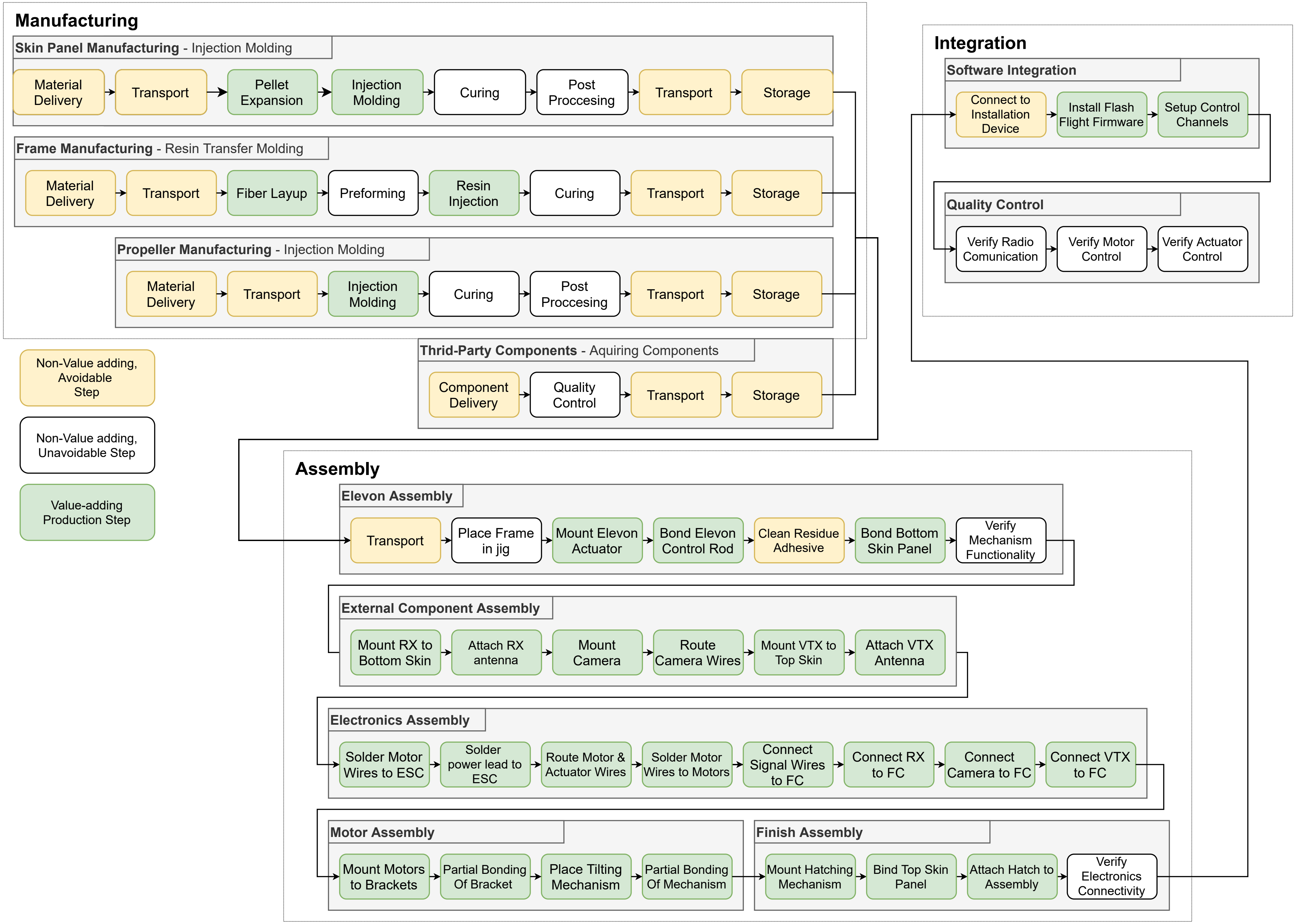


Figure 14.2: Manufacturing, Assembly & Integration for the Peregrine Drone

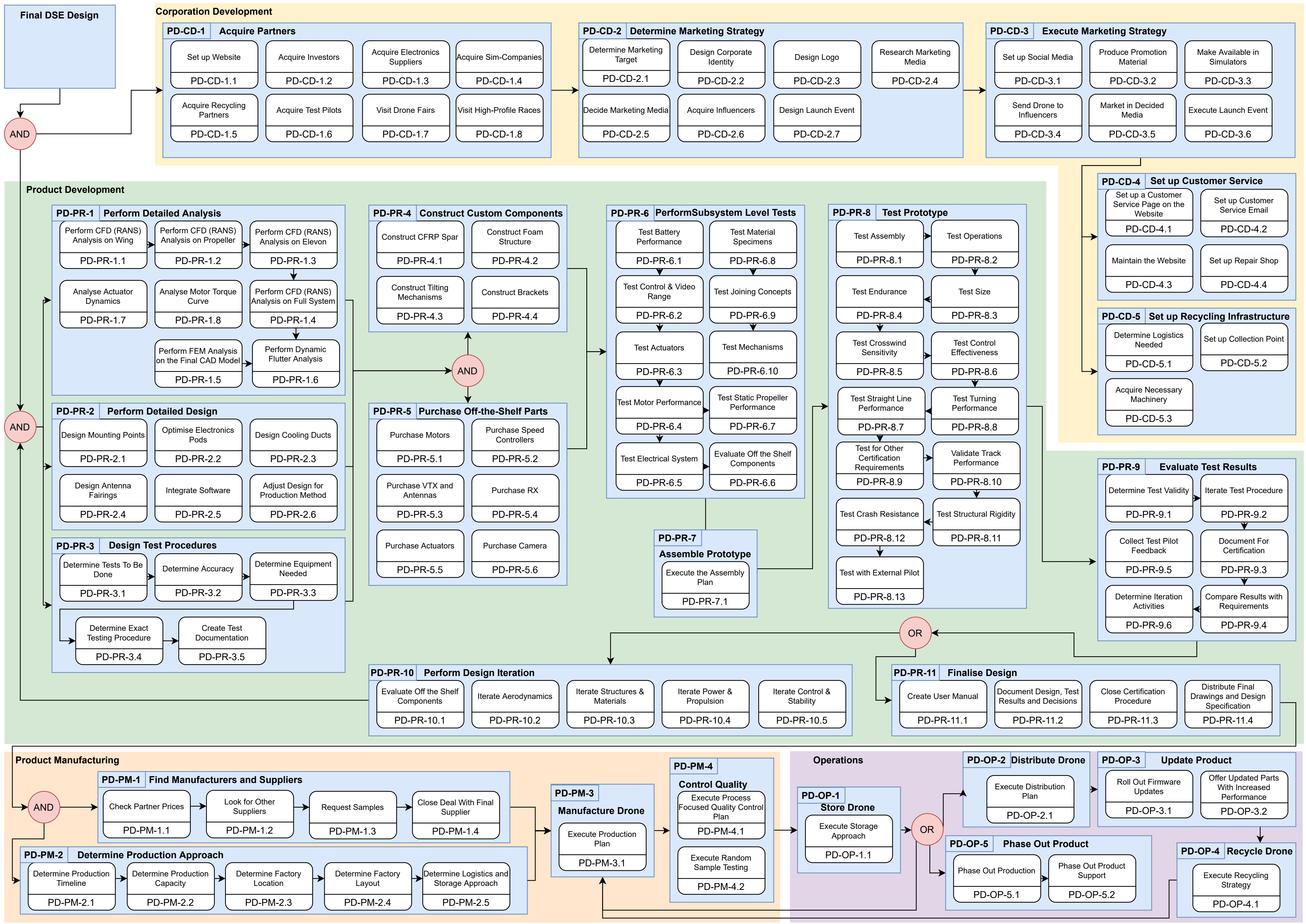
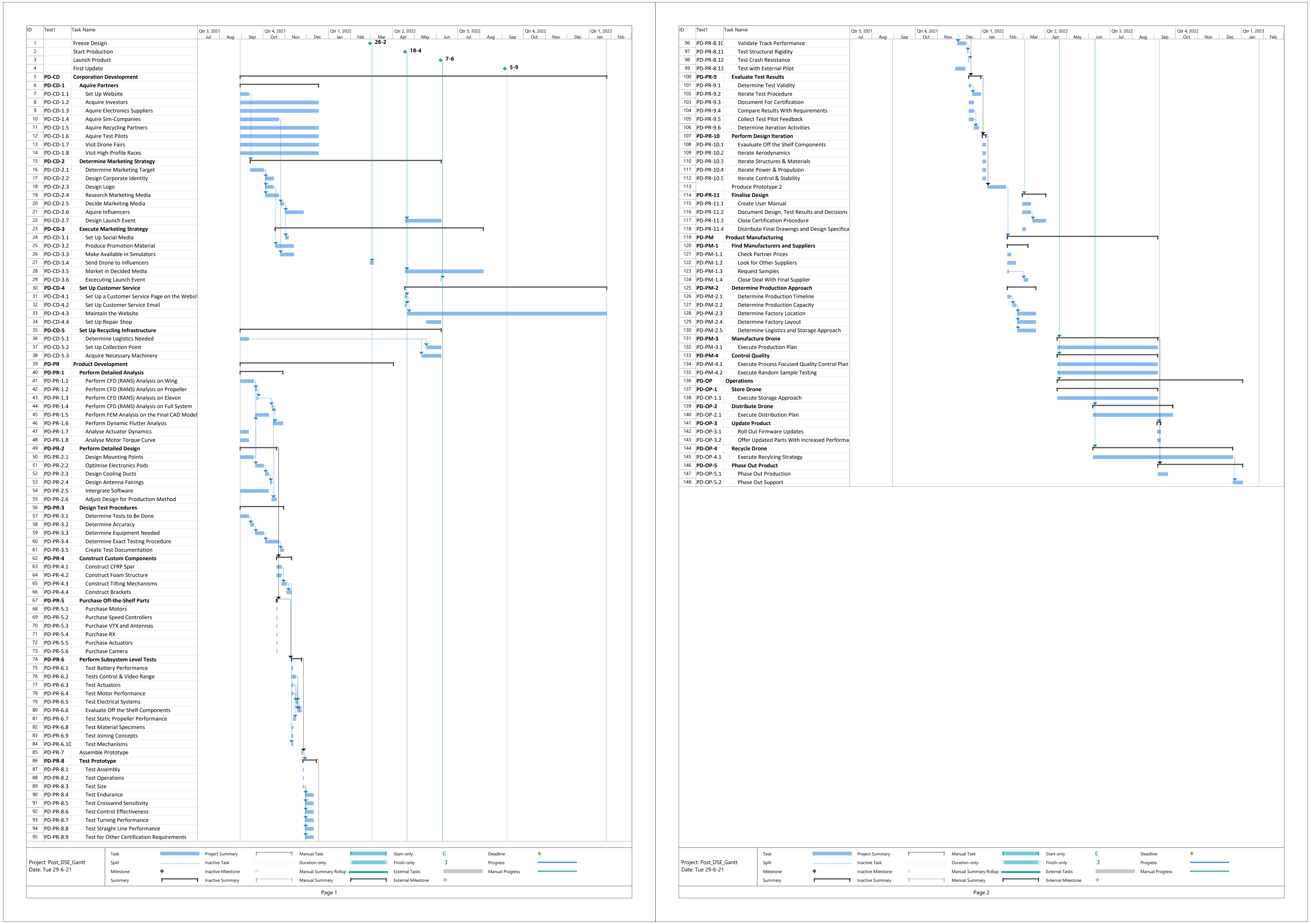


Figure 14.3: Project design and development logic



ID	Text1	Task Name
1		Freeze Design
2		Start Production
3		Launch Product
4		First Update
5	**PD-CD**	**Corporation Development**
6	**PD-CD-1**	**Aquire Partners**
7	PD-CD-1.1	Set Up Website
8	PD-CD-1.2	Acquire Investors
9	PD-CD-1.3	Aquire Electronics Suppliers
10	PD-CD-1.4	Aquire Sim-Companies
11	PD-CD-1.5	Aquire Recycling Partners
12	PD-CD-1.6	Aquire Test Pilots
13	PD-CD-1.7	Visit Drone Fairs
14	PD-CD-1.8	Visit High-Profile Races
15	**PD-CD-2**	**Determine Marketing Strategy**
16	PD-CD-2.1	Determine Marketing Target
17	PD-CD-2.2	Design Corporate Identity
18	PD-CD-2.3	Design Logo
19	PD-CD-2.4	Research Marketing Media
20	PD-CD-2.5	Decide Marketing Media
21	PD-CD-2.6	Aquire Influencers
22	PD-CD-2.7	Design Launch Event
23	**PD-CD-3**	**Execute Marketing Strategy**
24	PD-CD-3.1	Set Up Social Media
25	PD-CD-3.2	Produce Promotion Material
26	PD-CD-3.3	Make Available in Simulators
27	PD-CD-3.4	Send Drone to Influencers
28	PD-CD-3.5	Market in Decided Media
29	PD-CD-3.6	Excecuting Launch Event
30	**PD-CD-4**	**Set Up Customer Service**
31	PD-CD-4.1	Set Up a Customer Service Page on the Websi
32	PD-CD-4.2	Set Up Customer Service Email
33	PD-CD-4.3	Maintain the Website
34	PD-CD-4.4	Set Up Repair Shop
35	**PD-CD-5**	**Set Up Recycling Infrastructure**
36	PD-CD-5.1	Determine Logistics Needed
37	PD-CD-5.2	Set Up Collection Point
38	PD-CD-5.3	Acquire Necessary Machinery
39	**PD-PR**	**Product Development**
40	**PD-PR-1**	**Perform Detailed Analysis**
41	PD-PR-1.1	Perform CFD (RANS) Analysis on Wing
42	PD-PR-1.2	Perform CFD (RANS) Analysis on Propeller
43	PD-PR-1.3	Perform CFD (RANS) Analysis on Elevon
44	PD-PR-1.4	Perform CFD (RANS) Analysis on Full System
45	PD-PR-1.5	Perform FEM Analysis on the Final CAD Model
46	PD-PR-1.6	Perform Dynamic Flutter Analysis
47	PD-PR-1.7	Analyse Actuator Dynamics
48	PD-PR-1.8	Analyse Motor Torque Curve
49	**PD-PR-2**	**Perform Detailed Design**
50	PD-PR-2.1	Design Mounting Points
51	PD-PR-2.2	Optimise Electronics Pods
52	PD-PR-2.3	Design Cooling Ducts
53	PD-PR-2.4	Design Antenna Fairings
54	PD-PR-2.5	Intergrate Software
55	PD-PR-2.6	Adjust Design for Production Method
56	**PD-PR-3**	**Design Test Procedures**
57	PD-PR-3.1	Determine Tests to Be Done
58	PD-PR-3.2	Determine Accuracy
59	PD-PR-3.3	Determine Equipment Needed
60	PD-PR-3.4	Determine Exact Testing Procedure
61	PD-PR-3.5	Create Test Documentation
62	**PD-PR-4**	**Construct Custom Components**
63	PD-PR-4.1	Construct CFRP Spar
64	PD-PR-4.2	Construct Foam Structure
65	PD-PR-4.3	Construct Tilting Mechanisms
66	PD-PR-4.4	Construct Brackets
67	**PD-PR-5**	**Purchase Off-the-Shelf Parts**
68	PD-PR-5.1	Purchase Motors
69	PD-PR-5.2	Purchase Speed Controllers
70	PD-PR-5.3	Purchase VTX and Antennas
71	PD-PR-5.4	Purchase RX
72	PD-PR-5.5	Purchase Actuators
73	PD-PR-5.6	Purchase Camera
74	**PD-PR-6**	**Perform Subsystem Level Tests**
75	PD-PR-6.1	Test Battery Performance
76	PD-PR-6.2	Tests Control & Video Range
77	PD-PR-6.3	Test Actuators
78	PD-PR-6.4	Test Motor Performance
79	PD-PR-6.5	Test Electrical Systems
80	PD-PR-6.6	Evaluate Off the Shelf Components
81	PD-PR-6.7	Test Static Propeller Performance
82	PD-PR-6.8	Test Material Specimens
83	PD-PR-6.9	Test Joining Concepts
84	PD-PR-6.10	Test Mechanisms
85	PD-PR-7	Assemble Prototype
86	**PD-PR-8**	**Test Prototype**
87	PD-PR-8.1	Test Assembly
88	PD-PR-8.2	Test Operations
89	PD-PR-8.3	Test Size
90	PD-PR-8.4	Test Endurance
91	PD-PR-8.5	Test Crosswind Sensitivity
92	PD-PR-8.6	Test Control Effectiveness
93	PD-PR-8.7	Test Turning Performance
94	PD-PR-8.8	Test Straight Line Performance
95	PD-PR-8.9	Test for Other Certification Requirements
96	PD-PR-8.10	Validate Track Performance
97	PD-PR-8.11	Test Structural Rigidity
98	PD-PR-8.12	Test Crash Resistance
99	PD-PR-8.13	Test with External Pilot
100	**PD-PR-9**	**Evaluate Test Results**
101	PD-PR-9.1	Determine Test Validity
102	PD-PR-9.2	Iterate Test Procedure
103	PD-PR-9.3	Document For Certification
104	PD-PR-9.4	Compare Results With Requirements
105	PD-PR-9.5	Collect Test Pilot Feedback
106	PD-PR-9.6	Determine Iteration Activities
107	**PD-PR-10**	**Perform Design Iteration**
108	PD-PR-10.1	Evauluate Off the Shelf Components
109	PD-PR-10.2	Iterate Aerodynamics
110	PD-PR-10.3	Iterate Structures & Materials
111	PD-PR-10.4	Iterate Power & Propulsion
112	PD-PR-10.5	Iterate Control & Stability
113		Produce Prototype 2
114	**PD-PR-11**	**Finalise Design**
115	PD-PR-11.1	Create User Manual
116	PD-PR-11.2	Document Design, Test Results and Decisions
117	PD-PR-11.3	Close Certification Procedure
118	PD-PR-11.4	Distribute Final Drawings and Design Specifica
119	**PD-PM**	**Product Manufacturing**
120	**PD-PM-1**	**Find Manufacturers and Suppliers**
121	PD-PM-1.1	Check Partner Prices
122	PD-PM-1.2	Look for Other Suppliers
123	PD-PM-1.3	Request Samples
124	PD-PM-1.4	Close Deal With Final Supplier
125	**PD-PM-2**	**Determine Production Approach**
126	PD-PM-2.1	Determine Production Timeline
127	PD-PM-2.2	Determine Production Capacity
128	PD-PM-2.3	Determine Factory Location
129	PD-PM-2.4	Determine Factory Layout
130	PD-PM-2.5	Determine Logistics and Storage Approach
131	**PD-PM-3**	**Manufacture Drone**
132	PD-PM-3.1	Execute Production Plan
133	**PD-PM-4**	**Control Quality**
134	PD-PM-4.1	Execute Process Focused Quality Control Plan
135	PD-PM-4.2	Execute Random Sample Testing
136	**PD-OP**	**Operations**
137	**PD-OP-1**	**Store Drone**
138	PD-OP-1.1	Execute Storage Approach
139	**PD-OP-2**	**Distribute Drone**
140	PD-OP-2.1	Execute Distribution Plan
141	**PD-OP-3**	**Update Product**
142	PD-OP-3.1	Roll Out Firmware Updates
143	PD-OP-3.2	Offer Updated Parts With Increased Performa
144	**PD-OP-4**	**Recycle Drone**
145	PD-OP-4.1	Execute Recylcing Strategy
146	**PD-OP-5**	**Phase Out Product**
147	PD-OP-5.1	Phase Out Production
148	PD-OP-5.2	Phase Out Support

Figure 14.4: Project Gantt chart. The tasks below manufacture drone are expected to last for longer than indicated, for example 500 days.

15

Conclusion

The aim of this report was to conduct a detailed design stage and finalise the drone system. To do so, the report started with a brief revision of the design budgets, functions and market analysis. Information from these sections was then used to detail the subsystems and validate subsystem requirements. The subsystems requirements of Aerodynamics, Structures & Materials, Power & Propulsion, Control & Stability and Performance & Simulation were mostly validated, with remaining requirements marked as TBC and paired with proposed validation tests. Subsystem integration was then performed creating the final system, whereby risk and sustainability was assessed, and system requirements were then verified and validated. Finally, post-DSE tasks were detailed and operations and logistics evaluated.

Although the detailing of subsystems was successful, the result's degree of accuracy is questionable given the use of simulation and statistical techniques as opposed to physical testing. Testing of the hardware and design through prototyping and physical testing would greatly benefit the validation, offering reliable techniques to improve analysis quality. Naturally, the next stage of design would be to incorporate these tests, or at least offer more time to get acquainted with external simulation software to perform additional computations and validate remaining requirements. Improvements to the system tests could also be made. The simulation was used to evaluate Peregrine's performance and indicated that it met the project's goal, attaining the title of fastest racing drone. But this fact was never directly proven, rather verified by the speeds it obtained and lap-times. Thus the success of this design phase relies more on the accuracy of the simulation software and the user of the software as opposed to drone performance characteristics.

Given the conclusion of the DSE, the design phase would continue to the goal of obtaining a final system. In the remaining phase(s), it is expected that subtle changes may be made to optimise the system, such as making the foam a solid core, which may require a budget revision, which may require slight alterations to the requirements. Overall, the work done in this report will act as a new framework for all further analysis and thus the significance of the current results cannot be understated in the contribution to making the fastest racing drone on Earth.

Bibliography

[1] C.N. Adkins and R.H. Liebeck. “Design of optimum propellers”. In: *Journal of Propulsion and Power* 10.5 (1994), pp. 676–682. DOI: `10.2514/3.23779`. eprint: `https://doi.org/10.2514/3.23779`. URL: `https://doi.org/10.2514/3.23779`.

[2] E. Akca. “Validation of Stresses with Numerical Method and Analytical Method”. In: *PERIODICALS OF ENGINEERING AND NATURAL SCIENCES* 4 (Jan. 2016). DOI: `10.21533/pen.v4i1.47.g64`.

[3] A. Saeed et al. *Baseline Report - High Speed Racing Drone*. Tech. rep. TU Delft, Apr. 2021.

[4] A. Saeed et al. *Midterm Report: Peregrine - Racing Drone*. Tech. rep. TU Delft, May 2021.

[5] B.J.Lwo et al. “Reliability analyses on a TSV structure for CMOS image sensor”. In: ().

[6] E. Petritoli et al. “Reliability and Maintenance Analysis of Unmanned Aerial Vehicles”. In: ().

[7] X. Shua et al. “Reliability study of motor controller in electric vehicle by the approach of fault tree analysis”. In: ().

[8] Y. Pashchuk et al. “Reliability Synthesis for UAV Flight Control System”. In: ().

[9] European Aluminium Association. *RECYCLING ALUMINIUM A PATHWAY TO A SUSTAINABLE ECONOMY*. Aluminium for Future Generation.

[10] J.F. Barbosa et al. “Probabilistic S-N fields based on statistical distributions applied to metallic and composite materials: State of the art”. In: *Advances in Mechanical Engineering* 11.8 (2019), p. 168781401987039. ISSN: 1687-8140. DOI: `10.1177/1687814019870395`. URL: `https://dx.doi.org/10.1177/1687814019870395`.

[11] K.A. Bordignon. “Constrained Control Allocation for Systems with Redundant Control Effectors”. In: (1996), pp. 22–33.

[12] T. Bouqet and R. Vos. “Modeling the Propeller Slipstream Effect on Lift and Pitching Moment”. In: *American Institute of Aeronautics and Astronautics Inc. (AIAA)* (2017).

[13] R.M. Brach. “Energy Loss in Vehicle Collisions”. In: *SAE Transactions* 96 (1987), pp. 1279–1288. ISSN: 0096736X, 25771531. URL: `http://www.jstor.org/stable/44472891`.

[14] J. Brandt and M. Selig. “Propeller Performance Data at Low Reynolds Numbers”. In: *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. DOI: `10.2514/6.2011-1255`. eprint: `https://arc.aiaa.org/doi/pdf/10.2514/6.2011-1255`. URL: `https://arc.aiaa.org/doi/abs/10.2514/6.2011-1255`.

[15] I. Burhan and H. Kim. “S-N Curve Models for Composite Materials Characterisation: An Evaluative Review”. In: *Journal of Composites Science* 2.3 (2018), p. 38. ISSN: 2504-477X. DOI: `10.3390/jcs2030038`. URL: `https://dx.doi.org/10.3390/jcs2030038`.

[16] J. Cecrdle. *Whirl flutter of turboprop aircraft structures*. Cambridge, UK: Woodhead Publishing, 2015.

[17] J.X. Chen, J.Z. Jiang, and X.Y. Wang. “Research of energy regeneration technology in electric vehicle”. In: *Journal of Shanghai University 7* 7 (2003), pp. 173–177.

[18] N.D. Ciceri, M. Garetti, and S. Spérandio. “From Product End-of-Life Sustainable Considerations to Design Management”. In: vol. 338. Sept. 2009, pp. 152–159. DOI: `10.1007/978-3-642-16358-6_20`.

[19] J. Cody et al. “Regenerative braking in an electric vehicle”. In: (Jan. 2009).

[20] L. Čuček, J.J. Klemeš, and Z. Kravanja. “Chapter 5 - Overview of environmental footprints”. In: *Assessing and Measuring Environmental Impact and Sustainability*. Ed. by Jiří Jaromír Klemeš. Oxford: Butterworth-Heinemann, 2015, pp. 131–193. ISBN: 978-0-12-799968-5. DOI: `https://doi.org/10.1016/B978-0-12-799968-5.00005-1`. URL: `https://www.sciencedirect.com/science/article/pii/B9780127999685000051`.

[21] Q. Dai et al. “Life Cycle Analysis of Lithium-Ion Batteries for Automotive Applications”. In: *Batteries* 5.2 (2019). DOI: `10.3390/batteries5020048`. URL: `https://www.mdpi.com/2313-0105/5/2/48`.

[22] M. Das, R. Chacko, and S. Varughese. “An Efficient Method of Recycling of CFRP Waste Using Peracetic Acid”. In: *ACS Sustainable Chemistry & Engineering* (2018).

[23] M. Dauguet et al. “Recycling of CFRP for High Value Applications: Effect of Sizing Removal and Environmental Analysis of the SuperCritical Fluid Solvolysis”. In: *Procedia CIRP* 29 (2015). The 22nd CIRP Conference on Life Cycle Engineering, pp. 734–739. ISSN: 2212-8271. DOI: `https://doi.org/10.1016/j.procir.2015.02.064`. URL: `https://www.sciencedirect.com/science/article/pii/S2212827115001079`.

[24] N.K. Delany and N.E. Sorensen. “Low-speed drag of cylinders of various shapes”. In: (1953).

[25] J. Deng et al. “Recycling of Carbon Fibers from CFRP Waste by Microwave Thermolysis”. In: *Processes* (2019).

[26] C. Di Noi Dr A. Ciroth and M. Srocka T. Lohse. *openLCA 1.10 Comprehensive User Manual*. 1.10. GreenDelta. Feb. 2020.

[27] M. Drela. “An Analysis and Design System for Low Reynolds Number Airfoils”. In: *MIT Department of Aeronautics and Astronautics* (2001).

[28] “Effect of leading edge shape and hinge line location on hinge moment characteristics of NACA0012 Airfoil”. In: July 2015.

[29] Aluminium Federation. *Aluminium Recycling*. ALFED.

[30] A.S. Gill et al. “Cost Estimation Model for PAN Based Carbon Fiber Manufacturing Process”. In: 2016. DOI: `10.1115/msec2016-8724`.

[31] M.K. Hagnell, B. Langbeck, and M. Åkermo. “Cost efficiency, integration and assembly of a generic composite aeronautical wing box”. In: *Composite Structures* 152 (2016), pp. 1014–1023. ISSN: 0263-8223. DOI: `https://doi.org/10.1016/j.compstruct.2016.06.032`. URL: `https://www.sciencedirect.com/science/article/pii/S0263822316309163`.

[32] O. Harkegard. *Quadratic Programming Control Allocation Toolbox (QCAT)*. `https://www.mathworks.com/matlabcentral/fileexchange/4609-qcat`. 2021.

[33] E. P. Hartman and D. Biermann. *The Aerodynamic Characteristics of Full-Scale Propellers Having 2, 3, and 4 Blades of Clark Y and R.A.F. 6 Airfoil Sections*. Tech. rep. National Advisory Committee for Aeronautics, Jan. 1938.

[34] R.J. Huston. “Fatigue life prediction in composites”. In: *International Journal of Pressure Vessels and Piping* 59.1-3 (1994), pp. 131–140. ISSN: 0308-0161. DOI: `10.1016/0308-0161(94)90148-1`.

[35] Jr J.B. Dods. *ESTIMATION OF LOW-SPEED LIFT AND HINGE-MOMENT PARAMETERS FOR FULL-SPAN TRAILING-EDGE FLAPS ON LIFTING SURFACES WITH AND WITHOUT SWEEPBACK*. Tech. rep. NATIONAL ADVISORY COMMITTEE FOR AERONAUTICS, 1951.

[36] E.N. Jacobs and A. Sherman. *Airfoil Section Characteristics as Affected by Variations of the Reynolds Number*. Tech. rep. 586. National Advisory Committee for Aeronautics, 1939.

[37] K.M. Kambushev. “ANALYSIS OF THE RELIABILITY OF DC BRUSHLESS ELECTRIC MOTORS WITH POWER UP TO 200W USED IN MAVs”. In: ().

[38] G. Kanesan, S. Mansor, and A. Abdul-Latif. “Validation of UAV Wing Structural Model for Finite Element Analysis”. In: *Jurnal Teknologi* 71.2 (2014). ISSN: 2180-3722. DOI: `10.11113/jt.v71.3710`.

[39] P. Klein. “Parametric Modeling and Optimization of Advanced Propellers for Next-Generation Aircraft”. MA thesis. Delft University of Technology, Aug. 2017.

[40] Y.S. Lee, N.H. Park, and H.S. Yoon. “Dynamic Mechanical Characteristics of Expanded Polypropylene Foams”. In: *Journal of Cellular Plastics* 46.1 (2010), pp. 43–55. ISSN: 0021-955X. DOI: `10.1177/0021955x09346363`.

[41] S. Lei et al. “Method for Improving the Natural Lateral-Directional Stability of Flying Wings”. In: *Journal of Aerospace Engineering* (2016).

[42] H.I. Leong. “Development of a 6DOF Nonlinear Simulation Model Enhanced with Fine Tuning Procedures”. MA thesis. Graduate Faculty of the University of Kansas, 2008.

[43] T.H.G. Megson. *Aircraft Structures for Engineering Students*. 6th ed. Butterworth-Heinmann inc., 2017.

[44] M. Merchant and L. S. Miller. “Propeller Performance Measurement for Low Reynolds Number UAV Applications”. In: *44th AIAA Aerospace Sciences Meeting and Exhibit*. DOI: `10.2514/6.2006-1127`. eprint: `https://arc.aiaa.org/doi/pdf/10.2514/6.2006-1127`. URL: `https://arc.aiaa.org/doi/abs/10.2514/6.2006-1127`.

[45] A. Michos, G. Bergeles, and N. Athanassiadis. “Aerodynamic Characteristics of NACA 0012 Airfoil in Relation to Wind Generators”. In: *Wind Engineering* Vol.7.No.4 (1983), pp. 247–262.

[46] MIT. *Lecture 15: Energy Absorption Transcribed*. Lecture Notes available online at `https://ocw.mit.edu/courses/materials-science-and-engineering`. 2015.

[47] J.A. Mulder et al. *Flight Dynamics - Lecture Notes*. Faculty of Aerospace Engineering, Delft University of Technology, Mar. 2013.

[48] T.C. Muse. *SOME EFFECTS OF REYNOLDS AND MACH NUMBERS ON THE LIFT OF AN NACA 0012 RECTANGULAR WING IN THE NACA 19-FOOT PRESSURE TUNNE*. Tech. rep. National Advisory Committee for Aeronautics, 1943.

[49] X. Nian, F. Peng, and H. Zhang. “Regenerative Braking System of Electric Vehicle Driven by Brushless DC Motor”. In: *Industrial Electronics, IEEE Transactions on* 61 (Oct. 2014), pp. 5798–5808. DOI: `10.1109/TIE.2014.2300059`.

[50] S.G. Nieto et al. “Motion Equations and Attitude Control in the Vertical Flight of a VTOL Bi-Rotor UAV”. In: *MDPI Electronics* (2019).

[51] A.A. van Oudheusden. “Recycling of composite materials”. MA thesis. Faculty of Industrial Design Engineering, Delft University of Technology, 2019.

[52] R. Ramesh et al. *A Policy Brief: Eutrophication and Ocean Acidification*. A joint publication of GPNM, NCSCM, and IAN/UMCES. 2014.

[53] V. A. Romashchenko. “Limitations Of Some Strength Criteria For Composite Materials”. In: *Strength of Materials* 51.6 (2019), pp. 828–833. ISSN: 0039-2316. DOI: `10.1007/s11223-020-00131-8`.

[54] E. Sadien et al. “A detailed comparison of control allocation techniques on a realistic on-ground aircraft benchmark”. In: *American Control Conference* (2019).

[55] M.S. Selig et al. *Summary of Low-Speed Airfoil Data*. Vol. 1. SoarTech Publications, 1995.

[56] O. Al-Shamma, R. Ali, and H. S. Hasan. “An instructive algorithm for aircraft elevator sizing to be used in preliminary aircraft design software”. In: *Journal of Applied Engineering Science* 15 (2016), pp. 489–494.

[57] T.W. Shyr and Y.H. Pan. “Impact resistance and damage characteristics of composite laminates”. In: *Composite Structures* 62.2 (2003), pp. 193–203. ISSN: 0263-8223. DOI: `https://doi.org/10.1016/S0263-8223(03)00114-4`. URL: `https://www.sciencedirect.com/science/article/pii/S0263822303001144`.

[58] B. Siciliano and O. Khatib, eds. *Springer Handbook of Robotics*. Springer Science & Business Media, 2008. ISBN: 9783540239574.

[59] C.M. Simon and W. Kaminsky. “Chemical recycling of polytetrafluoroethylene by pyrolysis”. In: *Polymer Degradation and Stability* 62.1 (1998), pp. 1–7. ISSN: 0141-3910. DOI: `https://doi.org/10.1016/S0141-3910(97)00097-9`. URL: `https://www.sciencedirect.com/science/article/pii/S0141391097000979`.

[60] J. C. Sivells and S. H. Spooner. *Investigation in the Langley 19-foot Pressure Tunnel of Two Wings of NACA 65-210 and 64-210 Airfoil Sections with Various Type Flaps*. Vol. 1579. NACA, 1948.

[61] F. O. Smetana, D. C. Summey, and W. D. Johnson. *Riding and handling qualities of light aircraft - a review and analysis*. National Aeronautics and Space Administration (NASA), Mar. 1972, pp. 63–109.

[62] E.J.J. Smeur, M. Bronz, and G.C.H.E. de Croon. “Incremental Control and Guidance of Hybrid Aircraft Applied to a Tailsitter UAV”. In: *Journal of Guidance, Control and Dynamics* 43 (2020).

[63] S. Sunada, A. Sakaguchi, and K. Kawachi. “Airfoil Section Characteristics at a Low Reynolds Number”. In: *Journal of Fluid Engineering* (Mar. 1997).

[64] O. Takeda, T. Ouchi, and T.H. Okabe. “Recent Progress in Titanium Extraction and Recycling.” In: *Metallurgical and Materials Transactions* B (2020).

[65] C. Visser. *Introducing PID Control*. slides lecture 7 AE2235-I.

[66] L. Viterna and D. Janetzke. “Theoretical and experimental power from large horizontal-axis wind turbines”. In: *NASA Technical Memorandum* (Oct. 1982).

[67] B. Wang et al. “Energy absorption efficiency of carbon fiber reinforced polymer laminates under high velocity impact”. In: *Materials & Design* 50 (2013), pp. 140–148. ISSN: 0261-3069. DOI: `10.1016/j.matdes.2013.01.046`.

[68] Z. Wang, J. Zhang, and L. Yang. “Weighted pseudo-inverse based control allocation of heterogeneous redundant operating mechanisms for distributed propulsion configuration”. In: *Energy Procedia* 158 (2019), pp. 1718–1723.

[69] F.A. Warsi et al. “Yaw, Pitch and Roll controller design for fixed-wing UAV under uncertainty and perturbed condition”. In: *2014 IEEE 10th International Colloquium on Signal Processing and its Applications*. 2014, pp. 151–156. DOI: `10.1109/CSPA.2014.6805738`.

[70] S. Worasinchai, G. Ingram, and R. Dominy. “A low-Reynolds-number, high-angle-of-attack investigation of wind turbine aerofoils”. In: *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*. SAGE, July 2011.

[71] K. Yu et al. “Carbon Fiber Reinforced Thermoset Composite with Near 100% Recyclability”. In: *Advanced Functional Materials* 26 (July 2016). DOI: `10.1002/adfm.201602056`.