BSc Thesis

eVTOL design, Control System

Joris Holshuijsen
Joost van der Weerd

Delft University of Technology | June 18st, 2021

BSc Thesis

eVTOL Design, Control System

by

Joris Holshuijsen
Joost van der Weerd

Student Number:	4859669	Joris Holsuijsen
	4704320	Joost van der Weerd
Project Duration:	April 2021 – June 2021	
Thesis Committee:	Dr. ir. G.R. Chandra Mouli,	TU Delft, supervisor
	Dr. J. Dong,	TU Delft
	Dr. D. Cavallo,	TU Delft

EE3L11 Bachelor Graduation Project Electrical Engineering
Faculty of Electrical Engineering, Mathematics & Computer Science

TU Delft

Abstract

In a world with limited resources and a growing awareness of the destructive effects greenhouse gasses, the concept of sustainability is becoming more and more prominent in research. Some of the biggest polluters are the aerial shipping and travel industries, for this reason it is vital that they become more sustainable. Therefore a transition from fossil fuels to electric is needed. To completely transition to electric flight seems impossible but nevertheless big companies such as Airbus and start-ups such as Volocopter have started research and development of eVTOL's which are meant as urban air mobility (UAM) vehicles.

This thesis will explore the design of a control system for such an eVTOL. This thesis will be combined with two other theses to create a full theoretical design of an eVTOL. The other two theses will describe the physical design of the eVTOL, the energy storage system and the power train.

The end result of this thesis will be a description of the complete control system for this eVTOL, from what kind of sensors it needs, the mathematical model describing the eVTOL, the design of the control algorithm used, and simulations of the eVTOL in vertical flight.

Preface

This thesis is written in the context of the Bachelor Graduation Project of the Bachelor Electrical Engineering at the Delft University of Technology. The goal of this project was to make a theoretical design of an electrical vertical take-off and landing vehicle in accordance with challenge created by the AIAA and IEEE. During our graduation project we have been challenged to create a design that requires a lot of knowledge outside of our own field of study which has been both challenging and fun.

We would like to express our gratitude to Gautham Ram Chandra Mouli, Jianning Dong, Wiljan Vermeer, Tomas Sinnige, and Daniele Ragni for their support and expertise during this period.

Joris Holshuijsen & Joost van der Weerd
Delft, June 2021

Contents

List of Units and Abbreviations

Units

$\vec{a}$	Acceleration [m/s^2]
$\vec{E}$	Euler angles [°]
$\vec{F}$	Force [N]
$\overrightarrow{F_d}$	Drag Force [N]
$\overrightarrow{F_g}$	Gravitational Force [N]
$\overrightarrow{F_L}$	Lift Force [N]
$\overrightarrow{F_t}$	Thrust Force [N]
I	Inertia Matrix [A]
$\vec{M}$	Body Frame Moments [-]
$\overrightarrow{M_D}$	Drag Moments [-]
$\overrightarrow{M_L}$	Lift Moments [-]
$\overrightarrow{M_T}$	Thrust Moments [-]
$\overrightarrow{M_\tau}$	Propeller Moments [-]
ω	Propeller speed [rpm]
$\overrightarrow{\Omega_B}$	Angular velocity body frame [m/s]
ρ	(Air) Density [kg/m^3]
R	Rotation matrix
T	Thrust [N]
α	Wing to Body Angle [°]
v_∞	eVTOL Velocity [m/s]
I_{xx}	Moment of inertia around x
I_{yy}	Moment of inertia around y
I_{zz}	Moment of inertia around z
k_d	Drag coefficient of eVTOL in vertical flight
k_T	Thrust coefficient in vertical flight
L_{x1}	Length from y-axis to closest propeller, parallel to x-axis
L_{x2}	Length from y-axis to farthest propeller, parallel to x-axis
L_y	Length from x-axis to propellers, parallel to y-axis
c_τ	Rotor torque coefficient
A_w	Area of a single wing
L_{x3}	Distance to center of wing
k_{T_c}	Thrust coefficient in cruise mode
c_{τ_c}	Torque coefficient in cruise mode
$c_{l_{\delta_a}}$	Aileron roll control power
c_d	Zero lift + induced drag coefficient
c_l	Wing lift coefficient
k_{T_c}	Thrust coefficient in cruise
c_{τ_c}	Torque coefficient in cruise

Abbreviations

UAM	Urban Air Mobility
eVTOL	Electrical Vertical Take-Off and Landing
AIAA	American Institute of Aeronautics and Astronautics
eVTOL	electrical Vertical Take-Off & Landing
IEEE	Institute of Electrical and Electronics Engineers
MTOW	Maximum Take-Off Weight
EIS	Entry Into Service
EASA	European Union Aviation Safety Agency
LOS	Line of Sight
BLOS	Beyond Line of Sight
GS	Ground Station
PID	Proportional, Integral, & Derivative
RPM	Revolutions Per Minute
SOC	State-of-Charge
SOH	State-of-Health
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
PID	Proportional integral derivative controllers
FCS	Flight Control System
IMU	Inertial Measurement UNIT

1 Introduction

The prospects of a new means of transportation called Urban Air Mobility (UAM) which allows for passengers and cargo to be transported in between and inside of cities without using any traditional infrastructure such as roads and railways. The development of electrical Vertical Take-Off and Landing (eVTOL) aircraft plays a large role in the realization of UAM because they are quieter and more environmentally friendly to operate which makes them ideal for urban areas. Therefore it is interesting to investigate how the propulsion, energy storage, and the control of eVTOLs will develop in the years to come.

1.1. Project Objective

This thesis was inspired by a design challenge published by the American Institute of Aeronautics and Astronautics (AIAA) and Institute of Electrical and Electronics Engineers (IEEE) [1]. The challenge required a design that maximized the payload of the eVTOL while adhering to certain specifications outlined by the challenge. The design was divided into three main topics:

- The Aircraft Design Considerations, which is a description of the aerodynamic properties of the eVTOL, e.g. the wings, propellers, the fuselage, etc. Moreover, it requires an explanation how it is integrated with the eVTOL's other systems, i.e. electric propulsion and control.
- The Electric Propulsion System Design, which describes how the energy the eVTOL requires is stored, refueled and how it is distributed to the rotors and other components of the eVTOL.
- Concepts of Operation, which explains how the eVTOL can be controlled, whether or not passengers will be on board, and other aspects that might be important for the design.

However, an emphasis was put on the electrical engineering aspects of the design so three main theses were written by three subgroups on the electrical propulsion system, the energy storage system, and the control system of the eVTOL.

This thesis discusses the eVTOL's control system for which an accurate mathematical model of the eVTOL in horizontal and vertical flight is developed. Moreover, the developed model is implemented in Simulink which allows it to be used as a plant in a feedback control loop. Consequently, PID controllers are derived and synthesized using the developed vehicle description which give a first indication of the controllability of the eVTOL. Furthermore, a short overview of other aspects important to the control of a remotely-operated eVTOL are discussed such as the communication link, altitude and attitude sensors, and design of ailerons to provide for additional actuation in horizontal flight.

1.2. Thesis Outline

This thesis will start with a short overview of additional design considerations relevant for operation and control in vertical and horizontal flight in chapter 3. This includes aileron design for roll control in horizontal flight, a communication link so that the pilot can oversee the eVTOL, and a review of current attitude and altitude sensing methods. chapter 4 develops a mathematical model of the eVTOL, starting with a review of the mathematical concepts used and then continuing by describing all the forces and moments acting on the aircraft in vertical and horizontal modes. The subsequent section, chapter 5, discusses the derivation of the control formulas that will be used to design the classical PID controllers. chapter 6 touches on the implementation in Simulink of the mathematical model and the PID controllers, developed in chapter 4 and chapter 5 respectively. Subsequently the tuning of the controllers and their responses are presented in chapter 7. These results and recommendations for future research are discussed in chapter 8. Appendix A contains the total overview of the Simulink model and its individual components. Appendix B contain the matlab functions used in the Simulink model and the .m file used to initialize it.

2 Program of requirements

As mentioned before this thesis is part of a combination of three theses that design an eVTOL according to the specifications set by the challenge of the AIAA/IEEE. The total design has been split into three different components which have been designed in three separate theses. This thesis will focus on the design of a control system for the eVTOL. In the following sections the requirements will be listed, first the overall design requirements for the challenge will be listed and then the requirements for this specific subgroup will be listed.

2.1. Design requirements set by AIAA/IEEE

The design of the eVTOL for submission into the challenge has to have to following specifications:

- A range of 100km + 25km of backup
- A cruise speed of 250 km/h
- An operating altitude relative to the ground: 150m, and maximum altitude above sea level: 1,070m
- A maximum Take-Off Weight (MTOW) of 2.5 metric tons
- The largest aircraft planform dimension should fit within a 15-meter diameter circle
- Anticipated Entry into Service (EIS) 2030
- The eVTOL must produce zero CO2 at vehicle level

2.2. Control system

The following requirements are set for the control system of the eVTOL.

- A description of flight management
- A description of sensors needed in the control systems
- Mathematical models describing the dynamics of the eVTOL in vertical and horizontal flight
- Controller designs for vertical an horizontal flight
- Simulations showing the stability of the eVTOL

3

eVTOL Design overview

3.1. Description & Architecture of the eVTOL

The general design for the eVTOL can be described as a fixed-wing, fixed-rotor, tandem wing aircraft. Fixed-wing, fixed-rotor means that there are no rotating parts on the plane, other than the rotors, i.e. neither the wings nor the motors tilt. A tandem wing means that there are two wings connected to the fuselage. In particular, for this eVTOL, the wings will not be at similar heights, seen from the front when flying horizontally, so the wings will be at an angle, compared to the body. This will minimize the effects that the front propellers and wing will have on the rear propellers and wing. The exact dimensions and physical aspects for the wings, fuselage, and propellers are worked out in detail in the report of the Energy Storage subgroup, a render of the design can be seen in Figure 3.1. The next sections discus the additional features of the eVTOL relevant for auto-stabilizing the aircraft when in in horizontal and vertical flight. These features include the ailerons used to control the eVTOL's roll in horizontal flight, a communication link for a remote operator to oversee the eVTOL when in autopilot, and an overview of current sensing methods for attitude and altitude.

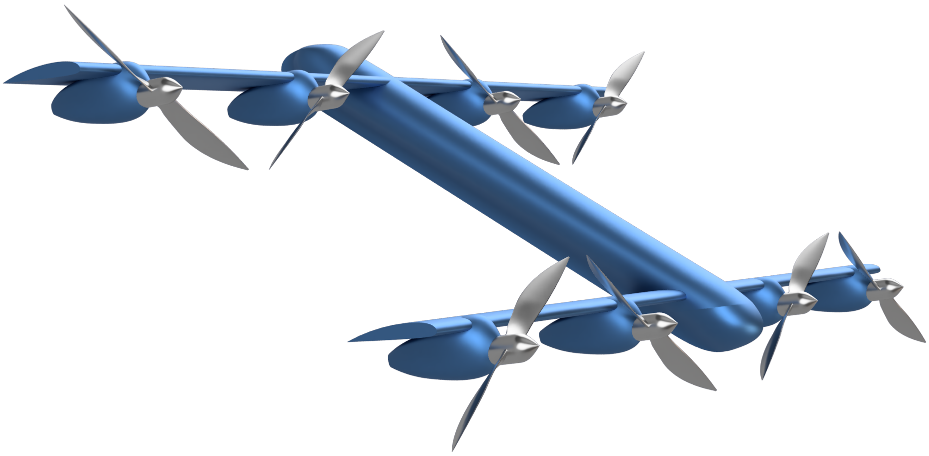


Figure 3.1: 3D Render of the eVTOL Design

3.1.1. Ailerons

In order to control the eVTOL in cruise flight, ailerons are included in the design to produce forces that cause a rolling moment. They are positioned on the front wing where they are the closest to unobstructed airflow, which would not be the case if they were positioned on the back wings. For this design, propeller wing interactions are not taken into account. By controlling the aileron deflection angle, δ_a, it is possible to alter the lift forces on the eVTOL which will be useful when designing the controllers in chapter 5. The dimensions of a single aileron are based on typical values seen in aircraft design and the aileron deflection angle is limited between $\pm 20°$ [2, Chapter 12.4.1].

It is possible to find an estimation of the lift coefficient k_{cl} of a wing as a function of the aileron deflection angle δ_a [2, Chapter 12.4]. The deflection angle determines the lifting force, which produces a roll moment. The lifting force is given by:

$$F_{L1,3} = \frac{1}{2}\rho A_w v_\infty^2 (c_l + \frac{c_{l_{\delta a}}}{2}\delta_{a1,3}) \tag{3.1}$$

Where c_l is the lifting coefficient of the wing without ailerons, v_∞ is the eVTOL's airspeed, A_w the area of single wing, ρ the density of air and $c_{l_{\delta a}}$ is the roll control power [2, Chapter 12.4] given by:

$$c_{l_{\delta a}} = c_r c_{l_{a_w}} \frac{2\tau}{A_w L_w}\left[\frac{L_w^2 - L_{a_{ie}}^2}{2}\right] \tag{3.2}$$

here, c_r is the chord length of the wing, $c_{l_{a_w}}$ the lift coefficient of the aileron surface, L_w the span of a single wing, $L_{a_{ie}}$ the distance to the ailerons inner edge, and τ is the control surface effectiveness which is around 0.35 for a control-surface-to-lifting-surface-chord ratio of 0.15 [2, Figure 12.3]. In practice, the roll-rate caused by the aileron lift forces create counter-acting drag forces, however, this will be ignored for simplicity of the model and under the assumption that the roll rates will be sufficiently low. Moreover, ailerons will also create additional drag force opposite to the direction of flight which are also ignored for the sake of simplicity.

3.1.2. Remote Piloting

The eVTOL will be remote controlled to increase the amount of cargo that can be transported by removing the weight of the pilot and manual control equipment. Furthermore, gradually reducing the number of pilots required to operate the eVTOL to a number below one through automation, will positively affect the operating costs due to a smaller labour force. Moreover, autonomous flight regulation is not yet in place, and according to the European Union Aviation Safety Agency (EASA), will not allow for autonomous commercial flight until 2035. [3] However, regulation for Unmanned Aerial Vehicles (UAV) is already in place. Finally, by designing for remote control, most of the sensors that are required for autonomous flight will already be included in the current design. This means that when changes in regulations and technological advancements allow for autonomous flight, the eVTOL can transition to autonomous operations without altering the design. The scope of this report is to auto-stabilize and control the eVTOL in horizontal and vertical flight, therefore, it will only go into detail into the communication aspect of remote flight. This is due to the fact that, in general, the pilot functions only as a supervisor during these stages of flight so the communication link is the only essential aspect of remote control in this case.

3.1.3. Communication

There are two main types of communication in Unmanned Aerial Systems (UAS), Line of Sight (LOS) and Beyond Line of Sight (BLOS). For our purposes LOS communication will be sufficient as this can provide coverage up to a couple hundred kilometers for sophisticated communications systems [4], which is sufficient for the eVTOL which has a maximum range of 125 kilometers.

Protocol Security is of utmost importance to prevent malicious entities compromising the eVTOL through the communication link. An analysis done by [5] shows that the most common attacks on UAVs are GPS spoofing, GPS jamming, and deauthentication attacks which resulted in the capture or crash of several UAVs. [6] proposes a Security Protocol for Drone-to-Ground Control System (SP-D2GCS). This protocol ensures the transmitted data is confidential and authentic and ensures perfect forward and backward secrecy, i.e. old and future session keys are not compromised when an unauthorized entity gets hold of a master key [6]. Eventually four messages are sent between the eVTOL and the GS with a total byte size of 2411 with a latency of 213.2196 ms when the drone is next to the ground control station [6]. This protocol seems viable but still has to be put to the test in the real world.

Bandwidths The bandwidths required for the data link between the eVTOL and the ground station will differ for the sensor data, camera feeds, the control commands, and perhaps voice channels for ATC relay. After a secure communication link is established using the proposed protocol, the data can be sent across the channel. An indication of the bandwidths required for the different types of data is 1.61 MHz for the command and control data [7], i.e. sensor data and ground station commands. Moreover, the required bandwidth for the air traffic control relay, which transmits data from air traffic control to the ground station via the eVTOL, is 2.72 MHz [7]. The video feeds and weather data will require a much higher bandwidth, around 23.51 MHz [7]. In 2012 the International Telecommunications Union (ITU) approved the 5030-5091 MHz frequency band for UAS communication [8], which should be sufficient for the required bandwidth.

Link-loss Procedure In the event the communication link is compromised or lost, the eVTOL should have a procedure in place that ensures the vehicle and payload safety. The procedure for most UAVs is to autonomously fly to a predetermined point where it will activate a backup communications system and seek contact with the ground station [9]. Depending on whether or not the eVTOL can reestablish the connection once arrived at the predetermined location, it will either hand over control to the pilot for landing or will try to land autonomously. Moreover, an added redundancy measure could be to also provide the eVTOL with Beyond-Line of Sight (BLOS) communication, so that when communication is lost, the link can be reestablished using a satellite as a relay station.

3.1.4. Automatic stability control

There has have been done plenty of research on the accurate sensing technologies for flight control systems (FCSs), ranging from single axis senors to more commonly used packages as inertial measurement unit (IMU) which use 3 accelerometers ans 3 rate gyros to measure angular velocity [10]. Even more complicated systems such as attitude heading and reference system (AHRS) that integrate accelerometers, magnetometers and rate gyros and use an estimation method to counteract sensor bias and noise. There are many different methods for the estimation problem, but one of the most commonly methods used are recursive methods based on the Kalman Filters [11]. For the altitude estimation traditionally altimeters have been used to give a rough estimation of height but research has been done but more advanced systems exist, a system that uses the output of the AHRS, a barometric sensor and a Kallman Filter as designed in [12] which has an accuracy of 0.1m. One additional consideration for safety is that using only one of these systems can lead to catastrophic failure even if just of the sensors fails therefore it is important to integrate redundant systems in the design.

4

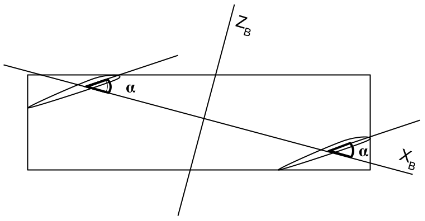
Vehicle dynamics

4.1. Six-degrees of freedom

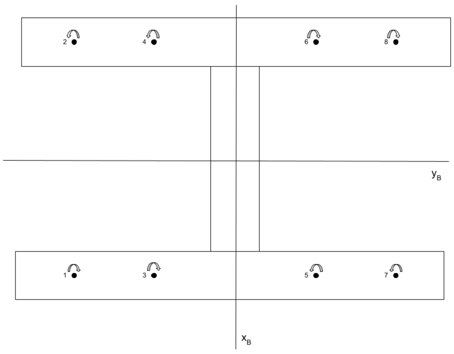
The first three degrees of freedom are of course the $\vec{x}$ position of the center of gravity of our eVTOL in 3-D space or the inertial/world frame. For simplicity the assumption will be made that the world frame consists of a flat earth with a constant gravitational constant. Moreover, the assumption is made that body is rigid, meaning that all components have fixed position with respect to the center of gravity. After the position of the center of gravity is known the location of its components can be derived due to their fixed position with respect to the center of gravity. The last three degrees of freedom are then $\vec{E}$ which represent the attitude of the eVTOL. To represent the attitude of thee eVTOL Euler angles will be used which will be explored more in depth later on [13].

4.2. Reference frames

To model the vehicle dynamics two reference frames have to be defined. These are the inertial frame and the body frame for which the coordinates will be indicated by $\vec{x}$ and $\overrightarrow{x_B}$ respectively. For the inertial frame a flat earth at sea level is assumed. The body frame is defined as in Figure 4.1a and Figure 4.1b. The x,y-plane is through the points where the forces from the motors act on the wings. Because the wings are not vertically centred the x,y-plane of the body frame is slightly tilted with respect to the vertical center of the body, thereby increasing the wing to body angle (α) from 37°to 40°. The origin of the body frame is in the center of mass of the eVTOL and the positive z direction is chosen such that the body frame becomes a right handed system meaning that in Figure 4.1a the positive y-axis is into the paper and in Figure 4.1a the positive z-axis is coming out of the paper.



(a) Body frame side view



(b) Body frame top view

4.3. Euler angles

Euler angles are a way to describe the attitude of the eVTOL via three rotations, these are successive rotations about specific axes. These rotations are not about the final body or world axes since the axes change after the first rotation, the order of the rotations matter but all of the different rotation matrices produce the same result. To translate between the two frames Equation 4.1 and Equation 4.2 can be used. The rotation matrix and its inverse used in this project can be seen in Equation 4.3 and Equation 4.4 respectively. In these equations $s_\beta = \sin\beta$ and $c_\beta = \cos\beta$ for shorter notation.

$$\vec{x} = R \cdot \overrightarrow{x_B} \tag{4.1}$$

$$\overrightarrow{x_B} = R^{-1} \cdot \vec{x} \tag{4.2}$$

$$R = \begin{bmatrix} c_\theta c_\phi & c_\theta s_\phi & -s_\theta \\ -c_\psi s_\phi + s_\psi s_\theta c_\phi & c_\psi c_\phi + s_\psi s_\theta s_\phi & s_\psi c_\theta \\ s_\psi s_\phi + c_\psi s_\theta c_\phi & -s_\psi c_\phi + c_\psi s_\theta s_\phi & c_\psi c_\theta \end{bmatrix} \tag{4.3}$$

Since the rotation matrix is orthonormal the inverse of the matrix is equal to its transpose ($R^{-1} = R^T$). Which gives the rotation matrix as seen in Equation 4.4.

$$R^{-1} = \begin{bmatrix} c_\theta c_\phi & s_\psi s_\theta c_\phi - c_\psi s_\phi & s_\psi s_\phi + c_\psi s_\theta c_\phi \\ c_\theta s_\phi & c_\psi c_\phi + s_\psi s_\theta s_\phi & -s_\psi c_\phi + c_\psi s_\theta s_\phi \\ -s_\theta & s_\psi c_\theta & c_\psi c_\theta \end{bmatrix} \tag{4.4}$$

4.4. Rate of change Euler angles

Since Euler angles are not related to the axes in the world or body frame it follows that the rate of changes of the Euler angles ($\vec{E}$) are not equal to the angular velocity ($\overrightarrow{\Omega_B}$) in the body frame. Since only the angular acceleration/velocity in the body frame can be measured a relation between the angular velocity in the body frame and the rate of change of Euler angles is needed. These can be derived by a similar procedure as the rotation matrix. The relation between angular velocities and the Euler rates can be seen in Equation 4.5. The inverse of that relation can be seen in Equation 4.6 respectively [14]. In which $\overrightarrow{\Omega_B} = \begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix}$ and $\frac{\partial \vec{E}}{\partial t} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$.

$$\frac{\partial \vec{E}}{\partial t} = \begin{bmatrix} 1 & s_\psi t_\theta & c_\psi t_\theta \\ 0 & c_\psi & -s_\psi \\ 0 & \frac{s_\psi}{c_\theta} & \frac{c_\psi}{c_\theta} \end{bmatrix} \overrightarrow{\Omega_B} \tag{4.5}$$

$$\overrightarrow{\Omega_B} = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\psi & s_\psi c_\theta \\ 0 & -s_\psi & c_\psi c_\theta \end{bmatrix} \frac{\partial \vec{E}}{\partial t} \tag{4.6}$$

4.5. Equations of motion

Now the link between the two frames of reference has been made the dynamics of the eVTOL can be modelled. The linear and angular motion will be evaluated separately and for convenience linear motion will be evaluated in the inertial frame and angular motion in the body frame.

4.5.1. Linear motion

As mentioned above linear motion will be evaluated in the inertial frame. The acceleration is of course given as in Equation 4.7. The main challenge will be to create a model of the forces that act on the eVTOL this will be done separately for vertical and horizontal flight in section 4.6 and section 4.7 respectively since the force that act on them differ. One thing to take into account is that the forces all have to be in the inertial frame so the forces in the body frame need to be translated to the inertial reference frame using rotation matrix R.

$$\vec{a} = \frac{\vec{F}}{m} \tag{4.7}$$

4.5.2. Inertial motion

The rotational motion of the eVTOL will be expressed in the body frame. Therefore Euler's equation for rotational motion is used (Equation 4.8), where I is the inertia matrix, $\vec{\Omega}$ are the angular velocities in the body frame and $\vec{M}$ are the moments in the body frame [15].

$$\vec{M} = I\dot{\vec{\Omega}} + \vec{\Omega} \times (I\vec{\Omega}) \tag{4.8}$$

Rewriting the formula to give angular acceleration gives us Equation 4.9.

$$\dot{\vec{\Omega}} = I^{-1}(\vec{M} - \vec{\Omega} \times (I\vec{\Omega})) \tag{4.9}$$

All that remains is to define the inertia matrix of the eVTOL and model the forces and moments acting on the aircraft.

4.5.3. Inertia matrix

The general form of the inertia matrix is given in Equation 4.10, but simplifications can be made. For a general three dimensional body it is always possible to define an orthogonal set of axes such that the products of inertia become zero. This orthogonal set of axes are also called thee principal axes of inertia. For symmetrical bodies these axes can be found very easily, since they correspond with the symmetry axes [16].

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \tag{4.10}$$

For the model of the eVTOL the assumption has been made that it is symmetric around the x- and y-axis of the body frame, which results in the diagonal inertia matrix:

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{4.11}$$

4.6. Vertical flight

Variable	Symbol
Moment of inertia around x	I_{xx}
Moment of inertia around y	I_{yy}
Moment of inertia around z	I_{zz}
Drag coefficient of eVTOL in vertical flight	k_d
Thrust coefficient in vertical flight	k_T
Wing to body angle	α
Length from y-axis to closest propeller, parallel to x-axis	L_{x1}
Length from y-axis to farthest propeller, parallel to x-axis	L_{x2}
Length from x-axis to propellers, parallel to y-axis	L_y
Rotor torque coefficient	c_τ

Table 4.1: eVTOL vertical flight constants

4.6.1. Linear forces

For vertical flight, three linear forces will be taken into account for the model which are the gravitational force acting on the center of gravity F_g, the drag force F_d, and the thrust F_t created by the motors. Since the thrust force of the rotors is in the body frame, it will need to be translated to the inertial frame first before they can be added which is done using the inverse of rotation matrix R^{-1}(Equation 4.1). The total force on the system is then as seen in Equation 4.12

$$\vec{F} = \vec{F_g} + R^{-1}\vec{F_t} + \vec{F_d} \tag{4.12}$$

The gravitational force is shown in Equation 4.13. The drag force is given as seen Equation 4.14 where the drag is modelled to have a linear relation to the velocity because of the low air speeds.

$$\overrightarrow{F_g} = \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \tag{4.13}$$

$$\overrightarrow{F_d} = \begin{bmatrix} -k_d v_x \\ -k_d v_y \\ -k_d v_z \end{bmatrix} \tag{4.14}$$

From Figure 4.2 the thrust force in the body frame is easily derived as Equation 4.15. With ω_i being the rotor speed of each motor.

$$\overrightarrow{F_t} = \begin{bmatrix} \sum_{n=1}^{8} T_{x,i} \\ 0 \\ \sum_{n=1}^{8} T_{z,i} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{8} k_T \cos \alpha \omega_i^2 \\ 0 \\ \sum_{i=1}^{8} k_T \sin \alpha \omega_i^2 \end{bmatrix} \tag{4.15}$$

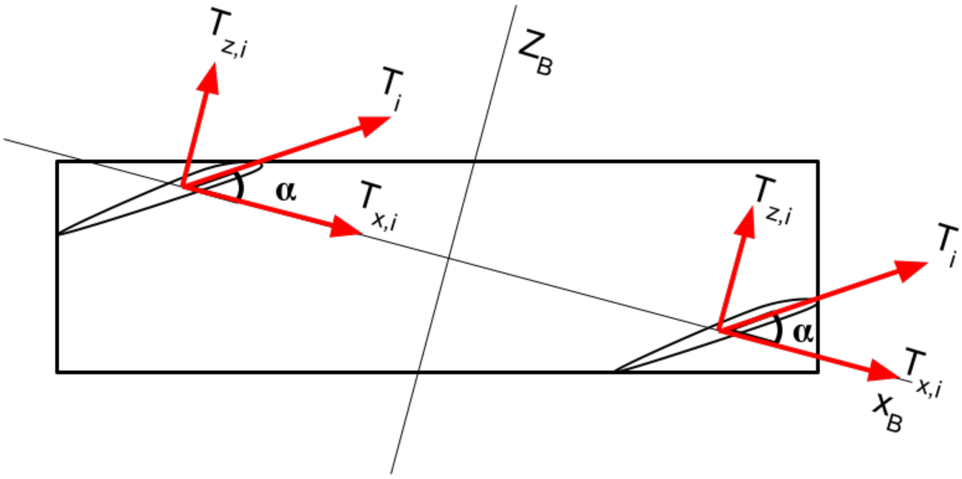
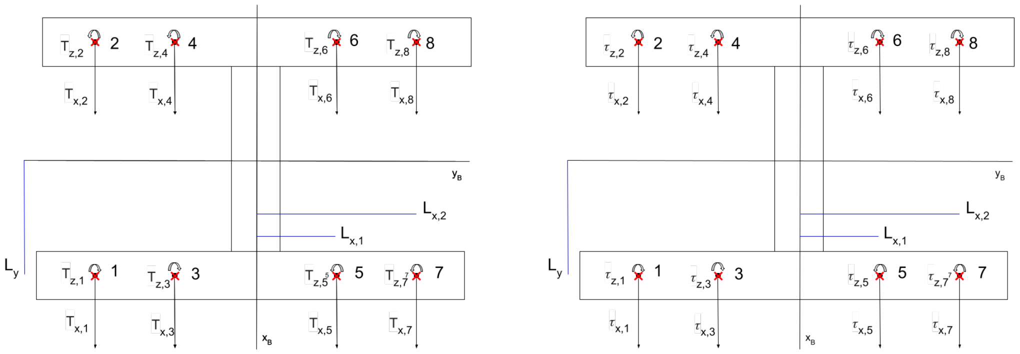


Figure 4.2: Thrust in body frame

4.6.2. Moments

The total moment, M, of the eVTOL is given by Equation 4.16 where M_T is the moment generated by the thrust forces acting on the rigid body of the eVTOL and M_τ is the total moment created by the eight spinning propellers.

$$\overrightarrow{M} = \overrightarrow{M_T} + \overrightarrow{M_\tau} \tag{4.16}$$



(a) Forces in body frame for vertical flight

(b) Torques in body frame for vertical flight

Where M_T is given by:

$$\overrightarrow{M_T} = \begin{bmatrix} L_{x2}(T_{z,1} + T_{z,2} - T_{z,7} - T_{z,8}) + L_{x,1}(T_{z,3} + T_{z,4} - T_{z,5} - T_{z,6}) \\ L_y(T_{z,1} - T_{z,2} + T_{z,3} - T_{z,4} + T_{z,5} - T_{z,6} + T_{z,7} - T_{z,8}) \\ L_{x2}(T_{x,1} - T_{x,2} - T_{x,7} + T_{x,8}) + L_{x1}(T_{x,3} - T_{x,4} - T_{x,5} + T_{x,6}) \end{bmatrix} \tag{4.17}$$

And M_τ can be determined using Equation 4.18, where c_τ is the rotor torque coefficient. Motors 1, 3, 6, and 8 rotate clockwise and motors 2, 4, 5, and 7 rotate counterclockwise as denoted by their sign.

$$\overrightarrow{M_\tau} = \begin{bmatrix} c_\tau \cos\alpha(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2 + \omega_5^2 - \omega_6^2 + \omega_7^2 - \omega_8^2) \\ 0 \\ c_\tau \sin\alpha(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2 + \omega_5^2 - \omega_6^2 + \omega_7^2 - \omega_8^2) \end{bmatrix} \tag{4.18}$$

4.7. Horizontal flight

Variable	Symbol
Density of air	ρ
Area of a single wing	A_w
Distance to center of wing	L_{x3}
Thrust coefficient in cruise mode	k_{T_c}
Torque coefficient in cruise mode	c_{τ_c}
Aileron roll control power	$c_{l_{\delta_a}}$
Zero lift + induced drag coefficient	c_d
Wing lift coefficient	c_l

Table 4.2: eVTOL horizontal flight constants

4.7.1. Linear Forces

In horizontal flight there are two extra forces, the lift force and the drag force generated by the wings and rotors. First lets define all the forces in the body frame. The forces can be seen in Equation 4.20 and Equation 4.22 adding this to Equation 4.15 to get the total forces on the body frame in cruise mode. Also, the thrust and torque constants k_T and c_τ in horizontal flight are adjusted for the dynamic cruise case, in which air speed also plays a role, and denoted by k_{T_c} and c_{τ_c}. The propulsion team calculates these constants in Section 5.2.2 of their report.

$$\vec{F} = F_g + R^{-1}(F_L + F_D + F_t) \tag{4.19}$$

The total lift force is given by:

$$\vec{F_L} = \begin{bmatrix} -\sin\alpha(F_{L1} + F_{L2} + F_{L3} + F_{L4}) \\ 0 \\ \cos\alpha(F_{L1} + F_{L2} + F_{L3} + F_{L4}) \end{bmatrix} \tag{4.20}$$

where $F_{L1,2,3,4}$ are given by Equation 3.1 and Equation 4.21, where v_∞ is the eVTOL's airspeed.

$$F_{L2,4} = \frac{1}{2} c_l \rho A_w v_\infty^2 = k_L v_\infty^2 \tag{4.21}$$

The total drag force is given by:

$$\overrightarrow{F_D} = \begin{bmatrix} \sum_{i=1}^{4} -F_{D,i} \cos\alpha \\ 0 \\ \sum_{i=1}^{4} -F_{D,i} \sin\alpha \end{bmatrix} \tag{4.22}$$

with the drag force on each wing:

$$F_{D,i} = \frac{1}{2} c_d \rho A_w v_\infty^2 = k_D v_\infty^2 \tag{4.23}$$

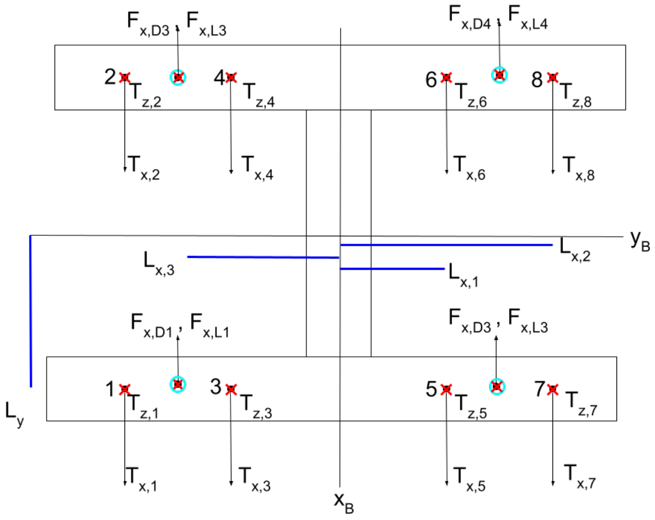


Figure 4.4: Linear Forces on body

4.7.2. Moments

These extra forces will also add to the moments in the body frame

$$\overrightarrow{M} = \overrightarrow{M_T} + \overrightarrow{M_\tau} + \overrightarrow{M_L} + \overrightarrow{M_D} \tag{4.24}$$

Under the assumption that the zero-aileron-deflection lift coefficient c_l is equal for all four wings and that $\delta_{a1} = -\delta_{a3} = \delta_a$ the following relation is obtained for the moments due to lift:

$$\overrightarrow{M_L} = \begin{bmatrix} L_{x3}(F_{z,L1} + F_{z,L2} - F_{z,L3} - F_{z,L4}) \\ L_y(F_{z,L1} - F_{z,L2} + F_{z,L3} - F_{z,L4} \\ L_{x3}(F_{x,L1} + F_{x,L2} - F_{x,L3} - F_{x,L4}) \end{bmatrix} = \begin{bmatrix} L_{x3} k_{La} v_\infty^2 \delta_a \cos\alpha \\ 0 \\ -L_{x3} k_{La} v_\infty^2 \delta_a \sin\alpha \end{bmatrix} \tag{4.25}$$

with $k_{La} = \frac{1}{2} c_{l_{\delta_a}} \rho A_w$.
Under the assumption that all the drag coefficients c_d of the wings are equal the moment due to drag becomes:

$$\overrightarrow{M_D} = \begin{bmatrix} L_{x3}(F_{z,D1} + F_{z,D2} - F_{z,D3} - F_{z,D4}) \\ L_y(F_{z,D1} - F_{z,D2} + F_{z,D3} - F_{z,D4} \\ L_{x3}(F_{x,D1} + F_{x,D2} - F_{x,D3} - F_{x,D4}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{4.26}$$

5 Control

The dynamics of the eVTOL are worked out and expressed in terms of virtual control inputs. In order to control these inputs, PID controllers will be designed as they are easy to implement and to understand [17]. Moreover, PID controllers are widely used which increases the research efforts to optimize design and tuning methods [18]. Once the virtual inputs are calculated by the controllers, they will be converted to actuator inputs, i.e. rotor speeds (ω) and aileron deflection angle (δ_a). This allows for stabilization and control of the eVTOL.

5.1. Vertical Flight

5.1.1. Assumptions

For simplification of the model and easier control, the propellers on each wing spin at the same angular velocity and in the same direction.

$$\omega_1 = \omega_3 \longrightarrow \omega_1 \tag{5.1}$$

$$\omega_5 = \omega_7 \longrightarrow \omega_3 \tag{5.2}$$

$$\omega_2 = \omega_4 \longrightarrow \omega_2 \tag{5.3}$$

$$\omega_6 = \omega_8 \longrightarrow \omega_4 \tag{5.4}$$

5.1.2. Control Formulas

Attitude It is important to have functional attitude control to stabilize the eVTOL. In this section, the control formulas and virtual control inputs for the attitude will be derived. The angular accelerations of the body ($\dot{\Omega_x}, \dot{\Omega_y}, \dot{\Omega_z}$) are obtained using Equation 4.9, and plugging in the values for the inertia matrix, I, it's inverse, I^{-1}, and the total moment of the eVTOL $\vec{M}$ given by Equation 4.16. This results in:

$$\begin{bmatrix} \dot{\Omega_x} \\ \dot{\Omega_y} \\ \dot{\Omega_z} \end{bmatrix} = \begin{bmatrix} \frac{1}{I_{xx}}(k_T \cos\alpha(-\omega_1^2 - \omega_2^2 + \omega_3^2 + \omega_4^2)(L_{x2} + L_{x1}) + c_\tau \cos\alpha(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) + (I_{yy} - I_{zz})\Omega_y\Omega_z) \\ \frac{1}{I_{yy}}(2k_T \cos\alpha(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2)L_y + (I_{zz} - I_{xx})\Omega_x\Omega_z) \\ \frac{1}{I_{zz}}(k_T \sin\alpha(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2)(L_{x2} + L_{x1}) + c_\tau \sin\alpha(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) + (I_{xx} - I_{yy})\Omega_x\Omega_y) \end{bmatrix} \tag{5.5}$$

Now let us express Equation 5.5 in terms of its virtual control inputs ($u_{1,2,3}$) defined as:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} \cos\alpha(k_T(-\omega_1^2 - \omega_2^2 + \omega_3^2 + \omega_4^2)(L_{x2} + L_{x1}) + c_\tau(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2)) \\ 2k_T \cos\alpha(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2)L_y \\ \sin\alpha(k_T(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2)(L_{x2} + L_{x1}) + c_\tau(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2)) \end{bmatrix} \tag{5.6}$$

Now Equation 5.5 turns into:

$$\begin{bmatrix} \dot{\Omega}_x \\ \dot{\Omega}_y \\ \dot{\Omega}_z \end{bmatrix} = \begin{bmatrix} \frac{u_1 + (I_{yy} - I_{zz})\Omega_y \Omega_z}{I_{xx}} \\ \frac{u_2 + (I_{zz} - I_{xx})\Omega_x \Omega_z}{I_{yy}} \\ \frac{u_3 + (I_{xx} - I_{yy})\Omega_x \Omega_y}{I_{zz}} \end{bmatrix} \tag{5.7}$$

Subsequently, Equation 5.7 is linearized for hover conditions [19], where in the case of the eVTOL $\phi \approx 0°, \theta \approx 50°, \psi \approx 0°$. Since the angles are almost constant, this linearization insinuates that the angular velocities are zero, which reduces Equation 5.7 further. Moreover, the angular accelerations expressed in the inertial frame are obtained by taking the derivative of Equation 4.5, which results in:

$$\ddot{\phi} = \dot{\Omega}_x + \dot{\Omega}_z \tan 50° = \frac{u_1}{I_{xx}} + \frac{u_3 \tan 50°}{I_{zz}} \tag{5.8}$$

$$\ddot{\theta} = \dot{\Omega}_y = \frac{u_2}{I_{yy}} \tag{5.9}$$

$$\ddot{\psi} = \frac{\dot{\Omega}_z}{\cos 50°} = \frac{u_3}{I_{zz} \cos 50°} \tag{5.10}$$

Position The equations used to develop position control are obtained by dividing the total force acting on the eVTOL, described in section 4.6, by its mass to obtain the accelerations. This results in Equation 5.11, where X, Y, and Z represent the position of the eVTOL.

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} = \frac{1}{m} R^{-1} \begin{bmatrix} 2k_T \cos\alpha(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \\ 0 \\ 2k_T \sin\alpha(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \frac{1}{m} \begin{bmatrix} -k_d \dot{X} \\ -k_d \dot{Y} \\ -k_d \dot{Z} \end{bmatrix} \tag{5.11}$$

Equation 5.11 turns into:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} = \frac{u_4}{m} R^{-1} \begin{bmatrix} \cos\alpha \\ 0 \\ \sin\alpha \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \frac{1}{m} \begin{bmatrix} -k_d \dot{X} \\ -k_d \dot{Y} \\ -k_d \dot{Z} \end{bmatrix} \tag{5.12}$$

where:

$$u_4 = 2k_T(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \tag{5.13}$$

The formula for the altitude is given by Equation 5.14. Where ψ, θ, ϕ are the Euler angles between the inertial and body frames.

$$\ddot{Z} = \frac{(-s_\theta c_\alpha + s_\alpha c_\theta)u_4 - k_d \dot{Z}}{m} - g \tag{5.14}$$

5.1.3. PID Control

The equations describing the relation between the virtual control variables and the angular and linear accelerations, are rewritten in terms of the virtual inputs and the accelerations are replaced by the formula for a PID controller. In these formulas, K_p, K_d, K_i, are the proportional, differential, and integral control constants and $e_{\phi,\theta,\psi,z}$ is the difference between the desired value in attitude or altitude and the current value, e.g. $e_p = p_{desired} - p_{eVTOL}$. Moreover, the altitude controller compensates for gravity. This results in the following relations:

$$u_1 = I_{xx}(-\frac{u_3 \tan 50°}{I_{zz}} + K_{p,\phi} e_\phi + K_{d,\phi} \dot{\phi} + K_{i,\phi} \int e_\phi) \tag{5.15}$$

$$u_2 = I_{yy}(K_{p,\theta} e_\theta + K_{d,\theta} \dot{\theta} + K_{i,\theta} \int e_\theta) \tag{5.16}$$

$$u_3 = I_{zz} \cos 50°(K_{p,\psi} e_\psi + K_{d,\psi} \dot{\psi} + K_{i,\psi} \int e_\psi) \tag{5.17}$$

$$u_4 = \frac{m(g + K_{p,z} e_z + K_{d,z} \dot{Z} + K_{i,z} \int e_z)}{\sin\alpha \cos 50° - \sin 50° \cos\alpha} \tag{5.18}$$

5.2. Horizontal Flight

5.2.1. Assumptions

Just like the vertical flight case, there need to be some restrictions placed on the actuators to avoid complexity. In this case, all the propellers on the lower wings will spin at the same speed. The propellers at the back wings will spin separately from each other, the same as in vertical flight mode. Moreover, the ailerons on the front wing will have the same deflection angle, δ_a, but in opposite directions to provide for the roll moment in horizontal flight.

$$\omega_1 = \omega_3 = \omega_5 = \omega_7 \longrightarrow \omega_1 \tag{5.19}$$

$$\delta_{a1} = -\delta_{a2} \longrightarrow \delta_a \tag{5.20}$$

$$\omega_2 = \omega_4 \longrightarrow \omega_2 \tag{5.21}$$

$$\omega_6 = \omega_8 \longrightarrow \omega_3 \tag{5.22}$$

There are several ways to control the altitude of the eVTOL such as pitching up- or downwards or adjusting the airspeed. For cruise mode, the changes in attitude will probably not deviate much, therefore the airspeed will be adjusted to remain at the correct altitude as this is less complex than adjusting the attitude.

5.2.2. Control Formulas

Attitude Same as in the previous section, the angular accelerations ($\dot{\Omega}_{x,y,z}$) of the eVTOL are calculated, however, this time the moment due to lift created by the wings and ailerons (Equation 4.25) is also included:

$$\begin{bmatrix} \dot{\Omega}_x \\ \dot{\Omega}_y \\ \dot{\Omega}_z \end{bmatrix} = \begin{bmatrix} \frac{1}{I_{xx}}(k_{T_c}\cos\alpha(-\omega_2^2+\omega_3^2)(L_{x2}+L_{x1}) + L_{x3}k_{La}v_\infty^2\delta_a\cos\alpha + c_{\tau_c}\cos\alpha(\omega_2^2-\omega_3^2) + (I_{yy}-I_{zz})\Omega_y\Omega_z) \\ \frac{1}{I_{yy}}(2k_{T_c}\cos\alpha(-2\omega_1^2+\omega_2^2+\omega_3^2)L_y + (I_{zz}-I_{xx})\Omega_x\Omega_z) \\ \frac{1}{I_{zz}}(k_{T_c}\sin\alpha(\omega_2^2-\omega_3^2)(L_{x2}+L_{x1}) - L_{x3}k_{La}v_\infty^2\delta_a\sin\alpha + c_{\tau_c}\sin\alpha(\omega_2^2-\omega_3^2) + (I_{xx}-I_{yy})\Omega_x\Omega_y) \end{bmatrix} \tag{5.23}$$

The virtual control inputs are now:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} k_{T_c}\cos\alpha(-\omega_2^2+\omega_3^2)(L_{x2}+L_{x1}) + L_{x3}k_{La}v_\infty^2\delta_a\cos\alpha + c_{\tau_c}\cos\alpha(\omega_2^2-\omega_3^2) \\ 2k_{T_c}\cos\alpha(-2\omega_1^2+\omega_2^2+\omega_3^2)L_y \\ k_{T_c}\sin\alpha(\omega_2^2-\omega_3^2)(L_{x2}+L_{x1}) - L_{x3}k_{La}v_\infty^2\delta_a\sin\alpha + c_{\tau_c}\sin\alpha(\omega_2^2-\omega_3^2) \end{bmatrix} \tag{5.24}$$

Which again results in Equation 5.7. Now the horizontal flight model is linearized for cruise mode, where $\phi \approx 0°, \theta \approx -40°, \psi \approx 0°$. Therefore, the accelerations in the world frame can still be obtained using Equation 5.8, Equation 5.9, and Equation 5.10 and replacing $50°$ with $-40°$.

Position For linear acceleration the same steps are taken as for the vertical flight case, however, this time with the added wing/aileron lift and drag forces, which results in:

$$\begin{bmatrix} \ddot{X} \\ \ddot{Y} \\ \ddot{Z} \end{bmatrix} = \frac{1}{m}R^{-1}\begin{bmatrix} 2k_{T_c}\cos\alpha(2\omega_1^2+\omega_2^2+\omega_3^2) - k_{La}\sin\alpha\delta_a v_\infty^2 - 4(k_D\cos\alpha + 4k_L\sin\alpha)v_\infty^2 \\ 0 \\ 2k_{T_c}\sin\alpha(2\omega_1^2+\omega_2^2+\omega_3^2) + k_{La}\cos\alpha\delta_a v_\infty^2 - 4(k_D\sin\alpha - 4k_L\sin\alpha)v_\infty^2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \tag{5.25}$$

Just as before, a virtual input is chosen:

$$u4 = 2k_{T_c}\sin\alpha(2\omega_1^2+\omega_2^2+\omega_3^2) + k_{La}\cos\alpha\delta_a v_\infty^2 \tag{5.26}$$

and the formula for the altitude becomes:

$$\ddot{Z} = \frac{(-s_\theta c_\alpha + s_\alpha c_\theta)(u_4 - 4(k_D\sin\alpha - 4k_L\sin\alpha)v_\infty^2)}{m} - g \tag{5.27}$$

5.2.3. PID Control

The formulas to synthesize the PID controllers for the attitude remain the same as in subsection 5.1.3, however, the altitude controller takes on a different form, namely:

$$u_4 = \frac{m(g + K_{p,z}e_z + K_{d,z}\dot{Z} + K_{i,z} \int e_z)}{\sin\alpha \cos -40° - \sin -40° \cos\alpha} \tag{5.28}$$

6 Model Implementation

In order to make the eVTOL fly, it is important to implement attitude and altitude control. However, since a physical realization of the eVTOL is unavailable, a control system , derived from the mathematical description given in section 4.6 is designed in Simulink. This model will consist of controller and formulas and a plant of the control loop discussed in chapter 5. Moreover, the structure of the model is based on the Simulink model developed in [20]. All the constants in the hover model are initiated by running a separate Matlab file, which can be found in section B.1.

An overview of the entire vertical flight implementation in Simulink is shown in Figure 6.1. On the left the desired values for altitude and attitude can be seen, which together with thee current state form the inputs to the controller. Then the output of the controller, the rotor speeds, get fed into the eVTOL model which has the updated state as output.

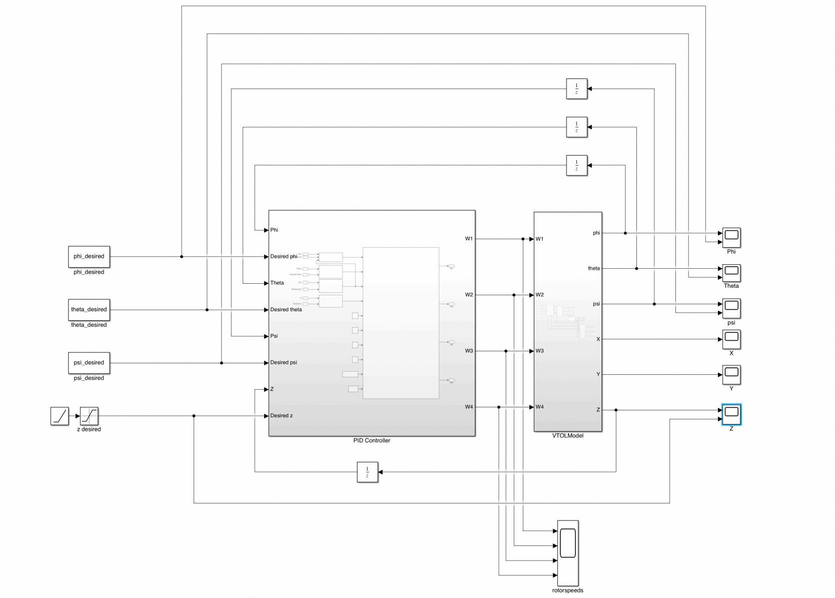


Figure 6.1: Overview of the hover model implementation

6.1. Plant model

An overview of the plant model can be seen in Figure A.1. The vertical flight model consists of three parts: Calculating the forces in the body frame (Figure A.2), updating the Euler angles (Figure A.3 and Figure A.4), and updating the position (Figure A.5).

The first part calculates the thrust and torques in the body frame caused by the propellers using Equation 4.15 for the thrust and Equations 4.16, 4.17, 4.18 for torque. The second part first calculates the roll, pitch, and yaw accelerations in the body frame using Equations (5.8, 5.9, and 5.10), then after integrating the accelerations to find the roll, pitch, and yaw velocities they are translated to Euler rates by using Equation 4.5. The last step is to integrate the Euler rates to find the new attitude of the eVTOL. The third section is responsible for calculating the x, y, and z position of the eVTOL in the inertial frame. This is done by calculating the total force acting on the eVTOL as in Equation 4.12. These forces are then used to calculate the accelerations in the world frame, which is then twice integrated to obtain the new position in the inertial frame.

6.2. Controller

An overview of the controller model can be seen in Figure A.6. The controller consists of two parts, the four individual PID controllers and the function that calculate the rotor speeds to keep the correct altitude and attitude. The subsystems for the four individual PID controllers can be seen in Figures (A.7, A.8, A.9, and A.10), the Matlab function that calculates the rotor speeds can be seen in section B.3.

7 Results

7.1. Tuning

The tuning process used in this project to find the correct values for the constants has been derived from the trial and error method in [21] and has been done by hand. The values found for the PID controllers can be seen in Table 7.1. As can be seen ϕ, θ, and ψ have been reduced to PD controllers while tuning it was found that for these control values there was no steady state error. This is a consequence of the way our model is implemented. Since our model uses dead reckoning but has feedback without noise or drift in the sensors, in real circumstances this would never be the case and an integrating factor should be added.

Table 7.1: Parameters of the PID controllers

Parameter	Value
$K_{z,P}$	7.9
$K_{\phi,P}$	1.4
$K_{\theta,P}$	1.2
$K_{\psi,P}$	3.1

Parameter	Value
$K_{z,I}$	1.4
$K_{\phi,I}$	0
$K_{\theta,I}$	0
$K_{\psi,I}$	0

Parameter	Value
$K_{z,D}$	3.5
$K_{\phi,D}$	5.2
$K_{\theta,D}$	5.8
$K_{\psi,D}$	8.7

7.2. Testing

To test performance of the controller designed in this project 5 simulations where done to test the stability for the different control values. The first 4 simulations where done to test the control variables individually. The test done for ϕ, θ, and ψ was to give the angles initial offset of 20° from its desired angle. The desired angles are (0°, 310°, 0°). To test $\dot{Z}$ a flight path was set to climb from 0m tot 20m in 20 seconds and then hover at that height. The last test was to simulate a take-off in which it is unrealistic to assume the eVTOL to be completely level therefore the same flight path was used as in the test for $\dot{Z}$ but this time the eVTOL had an offset of 20° in both ϕ and θ no offset was used for ψ since that is a rotation about the z-axis in the inertial frame and that does not effect take-off. The results are shown below.

Z response The z response has a maximum deviation of 2m and a settling time of 7.5s.

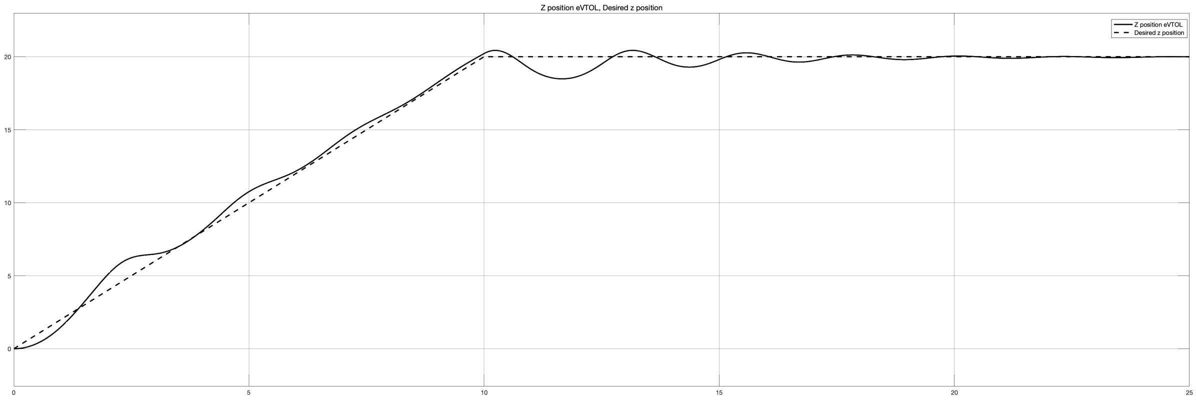


Figure 7.1: Z response

Phi response The phi response has an overshoot of 11%, a settling time of 5.5s, and a rise time of 2s.

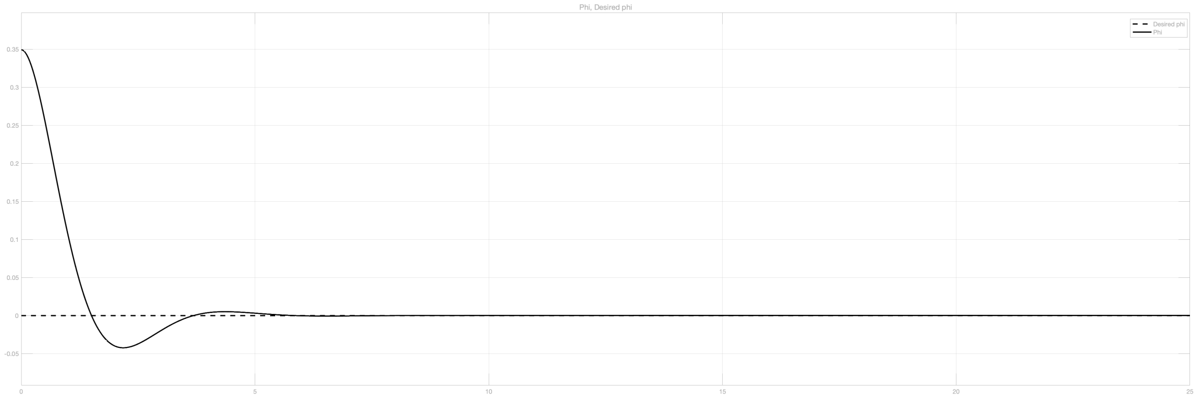


Figure 7.2: Phi response

Theta response The phi response has an overshoot of 11%, a settling time of 6.5s, and a rise time of 2.5s.

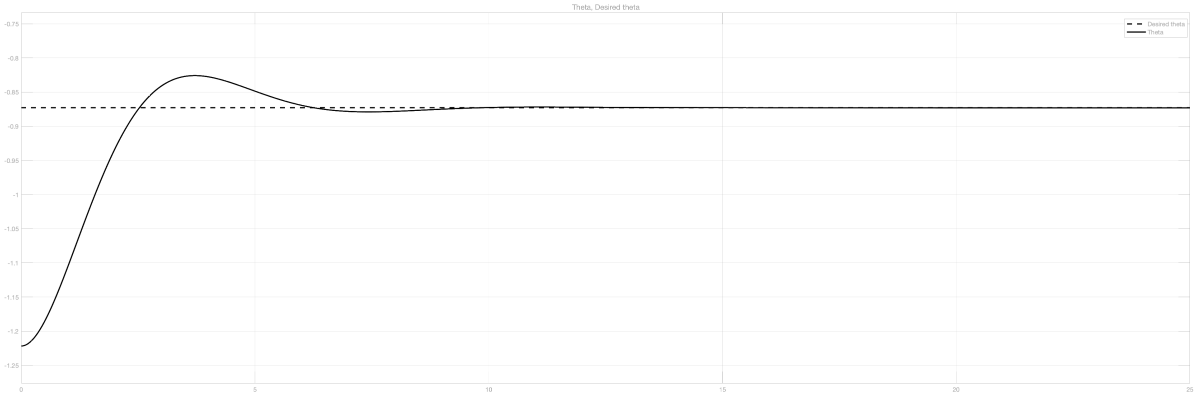


Figure 7.3: Theta response

Psi response The phi response has an overshoot of 8%, a settling time of 4.0s, and a rise time of 2.5s.

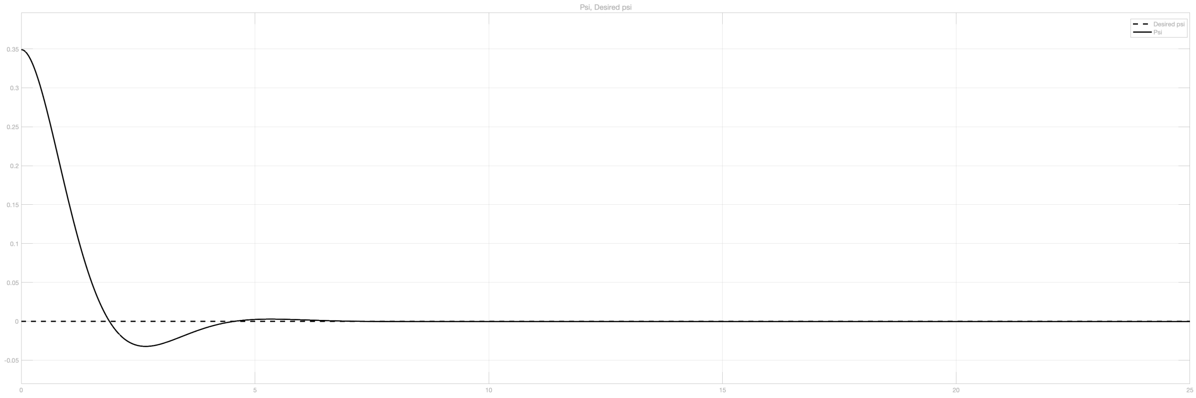


Figure 7.4: Psi response

Complete response
The z response has a maximum deviation of 2m and a settling time of 7.5s.

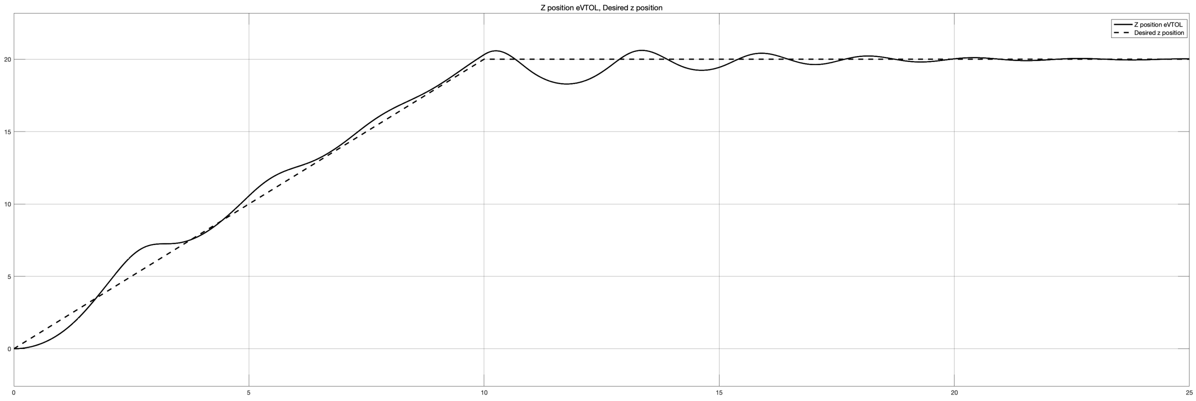


Figure 7.5: Complete response z

The phi response for the complete test has an overshoot of 11%, a settling time of 5.5s, and a rise time of 2s.

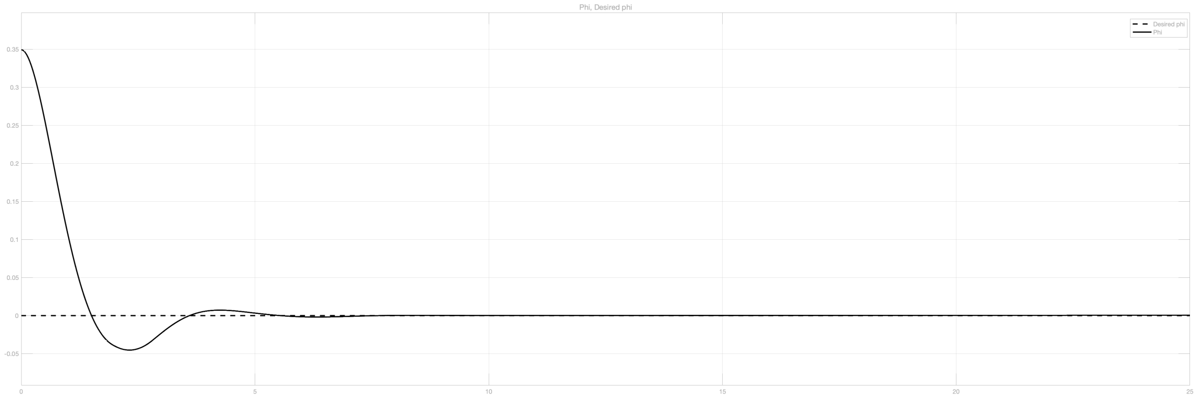


Figure 7.6: Complete response phi

The phi response for the complete test has an overshoot of 33%, a settling time of 8s, and a rise time of 2.5s.

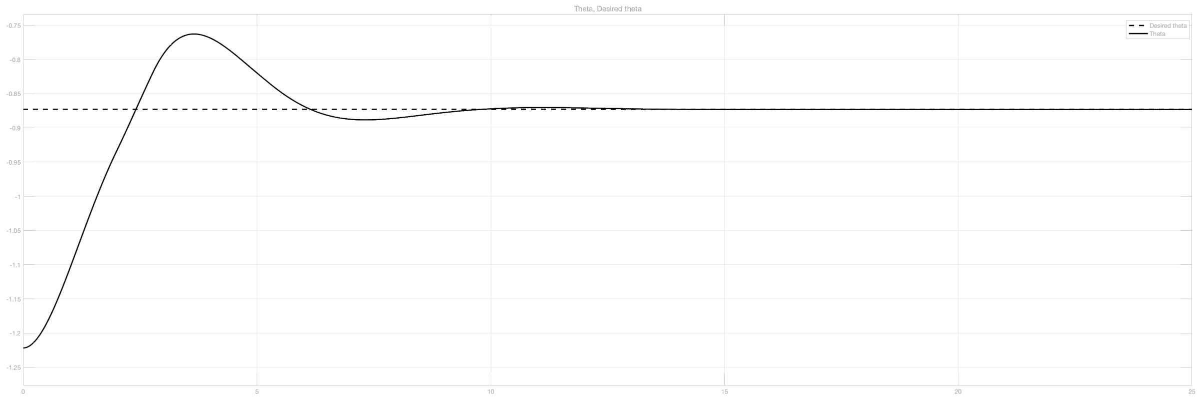


Figure 7.7: Complete response theta

8

Conclusion & Future work

8.1. Conclusion

The way the control systems for the eVTOL has been designed is not completely according to the program of requirements as mentioned in chapter 2. Due to bugs in the simulation of horizontal flight we have not been able to perform the simulations to prove the stability of the control system mentioned in section 5.2. The rest have been fulfilled with the following specifications:

- The eVTOL will be remotely piloted assisted by automatic stability control as described in chapter 3.
- The sensors that need will be used will be two AHRS systems and two altitude sensors based on a barometric senor and the attitude given by the AHRS systems.
- The developed models are given in chapter 4 which behave according to physical principles (i.e. when no thrust is delivered the eVTOL fall out of the sky with an acceleration equal to the gravitational constant).
- The developed models are given in chapter 5 the control model for vertical flight has been verified through the simulations done in chapter 7.
- The simulations for vertical flight shows a system that when offset falls back to its stable state with a maximum settling time of 8s

8.2. Future work

While the results of the controller for vertical flight are promising, the ultimate goal of a control system that can be used to take-off, fly, and then land in its desired location has not been reached. To achieve this there is more work to be done. First of all the controller which has been designed for cruise flight needs to be implemented and tuned. After this is done the two separate models have to be combined in such a way that also the transition between the two states can be handled and controlled properly. When this goal is reached further improvements regarding the model and the controller can be made.

Model improvements As with any model the model used in this project has it shortcomings that can be expanded and improved on. One area wherein a lot of simplifications have been made are the aerodynamic properties of the eVTOL. For this iteration of the model, only rudimentary drag and lift forces have been taken into account even though aerodynamics is one of the more complex and non-linear fields of study. Another part for which the model can be enhanced is the thrust and torque models of the motor and rotor combination. As the eVTOL design progresses, the real thrust and torque curves of a motor and rotor combination should be used. The final improvement for our model would be to implement a model for the sensors that includes noise instead of the dead reckoning used in the simulation.

Controller improvements One improvement for the controller used in this project would be the implementation of an automatic control algorithm, since tuning an PID controller by hand can be a painstaking process which does not always yield the best results. Also using an other controller than PID control can be taken into consideration as the complexity of the eVTOL design increases.

A

Simulink figures

A.1. Plant

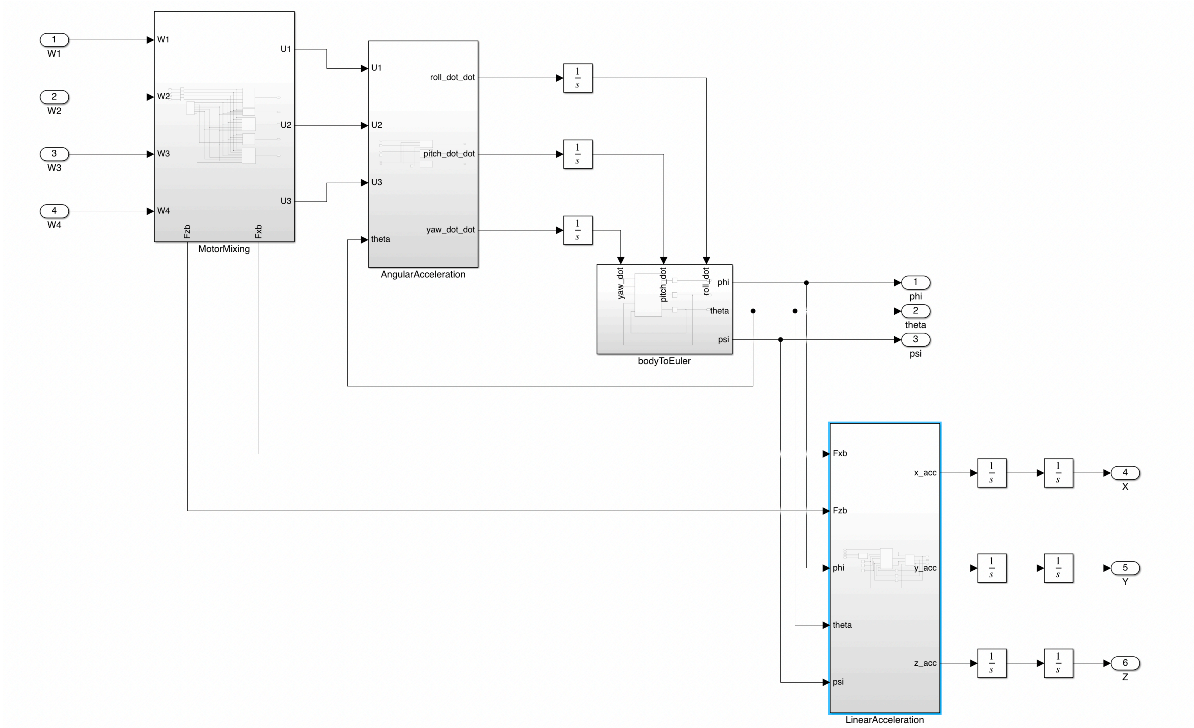


Figure A.1: Simulink block: Plant

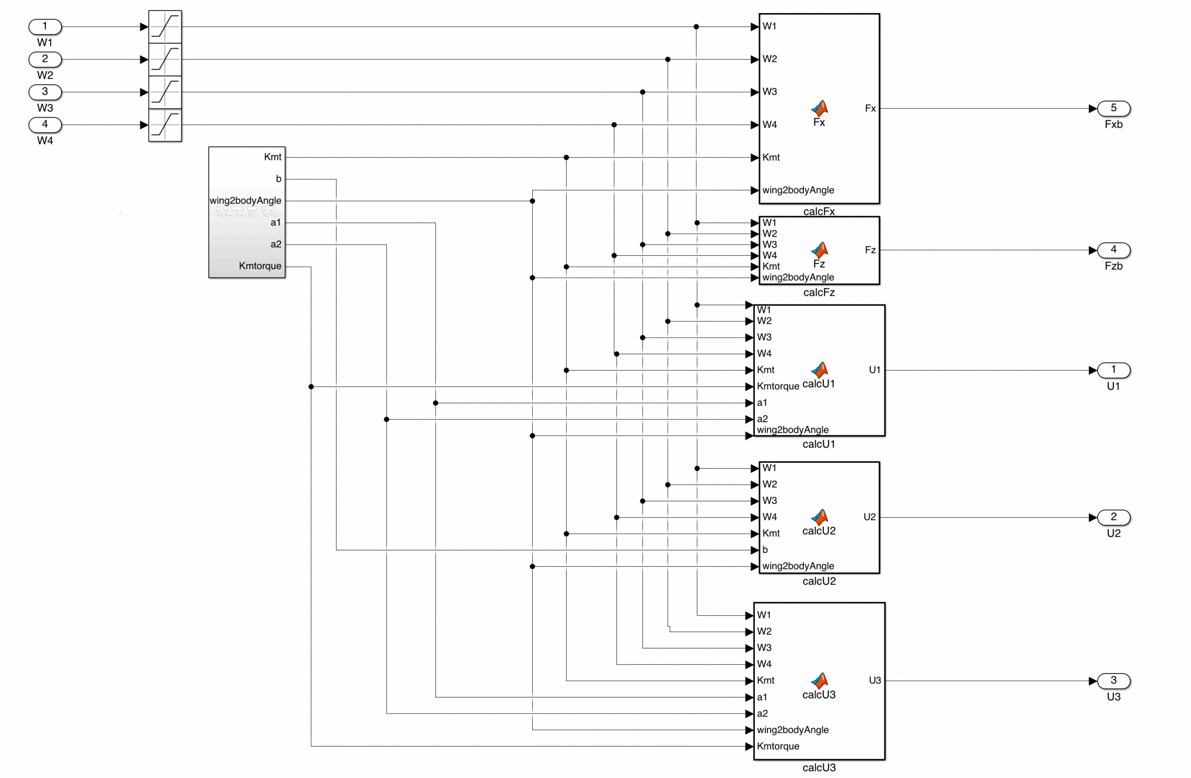


Figure A.2: Simulink block: Calculate forces

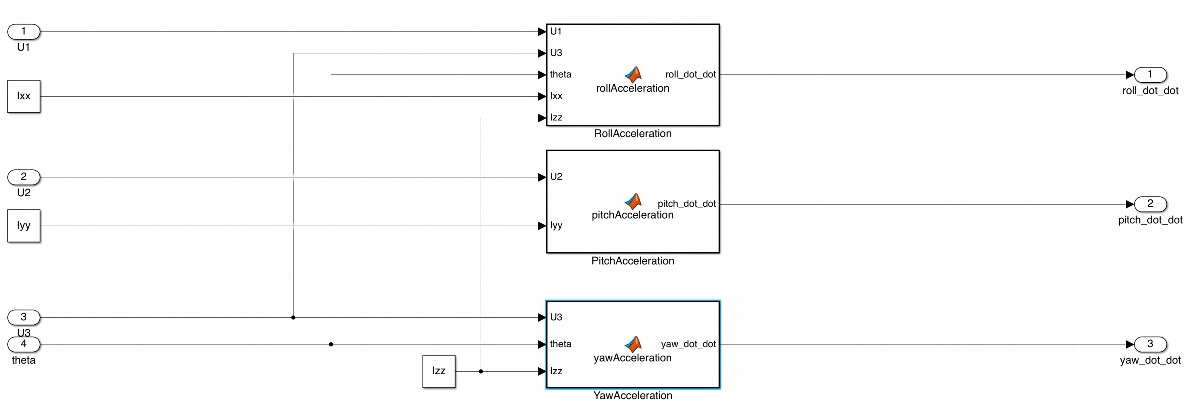


Figure A.3: Simulink block: Calculate angular acceleration

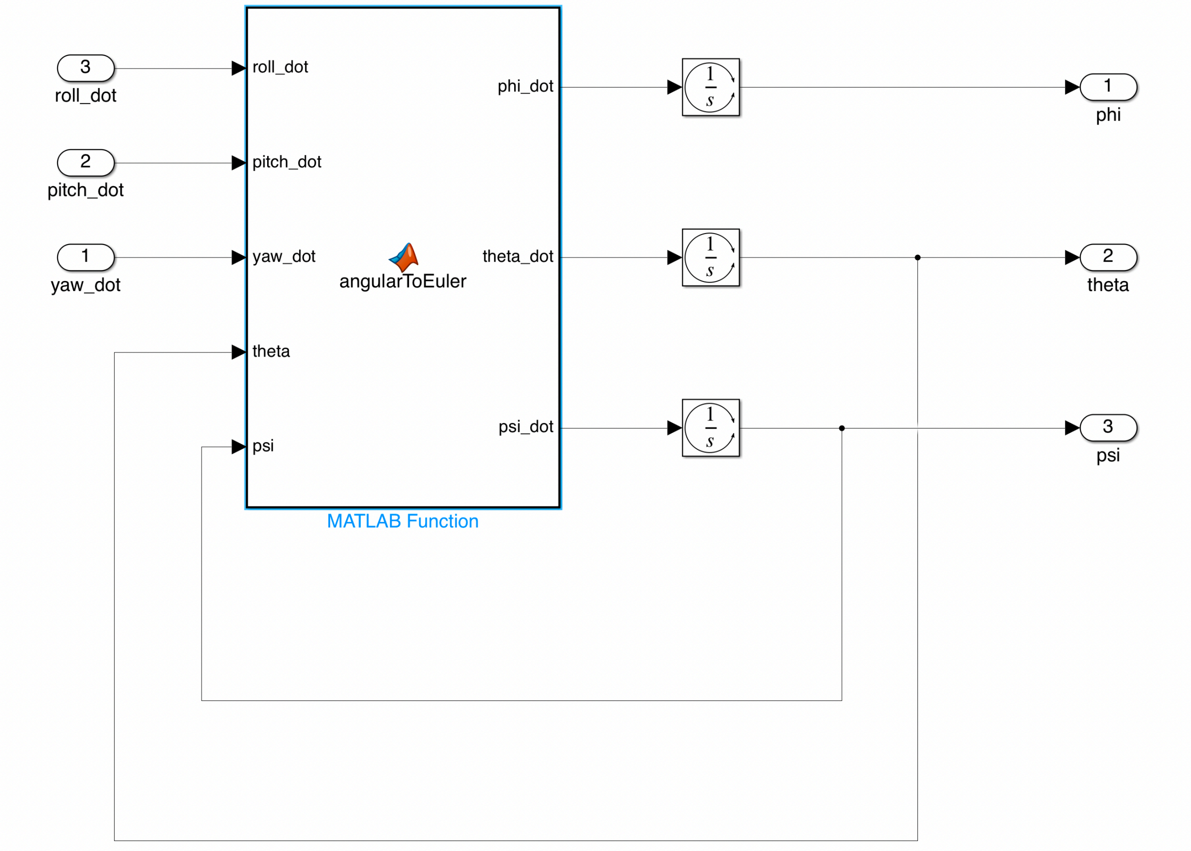


Figure A.4: Simulink block: Calculate Euler angels

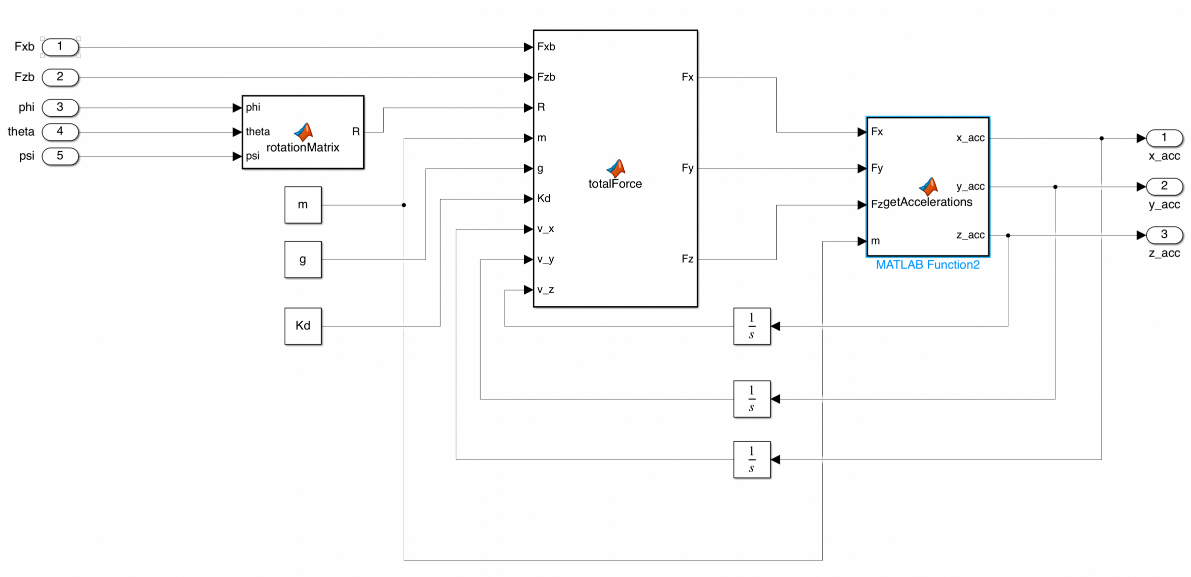


Figure A.5: Simulink block: Calculate linear accelerations

A.2. Controller

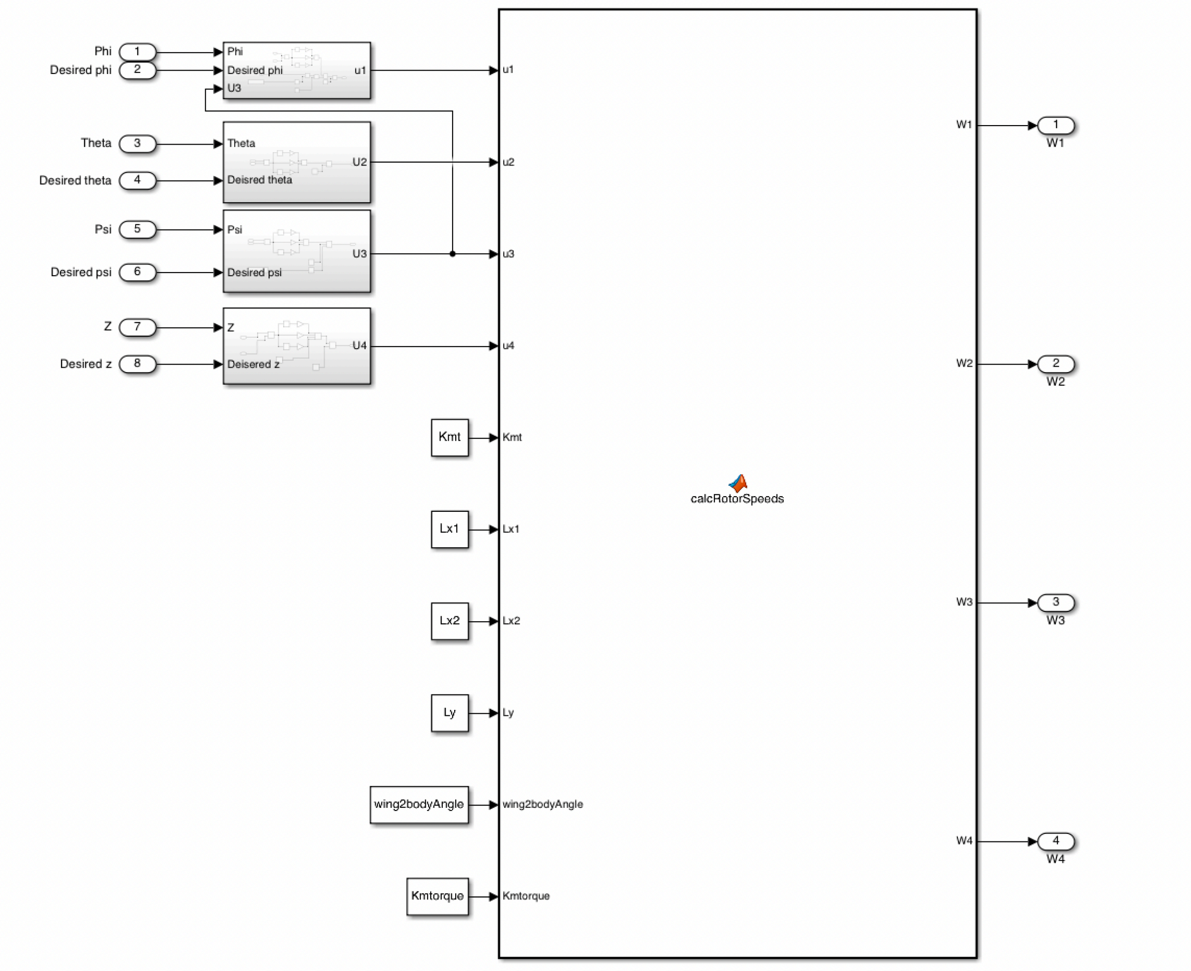


Figure A.6: Simulink block: Control system

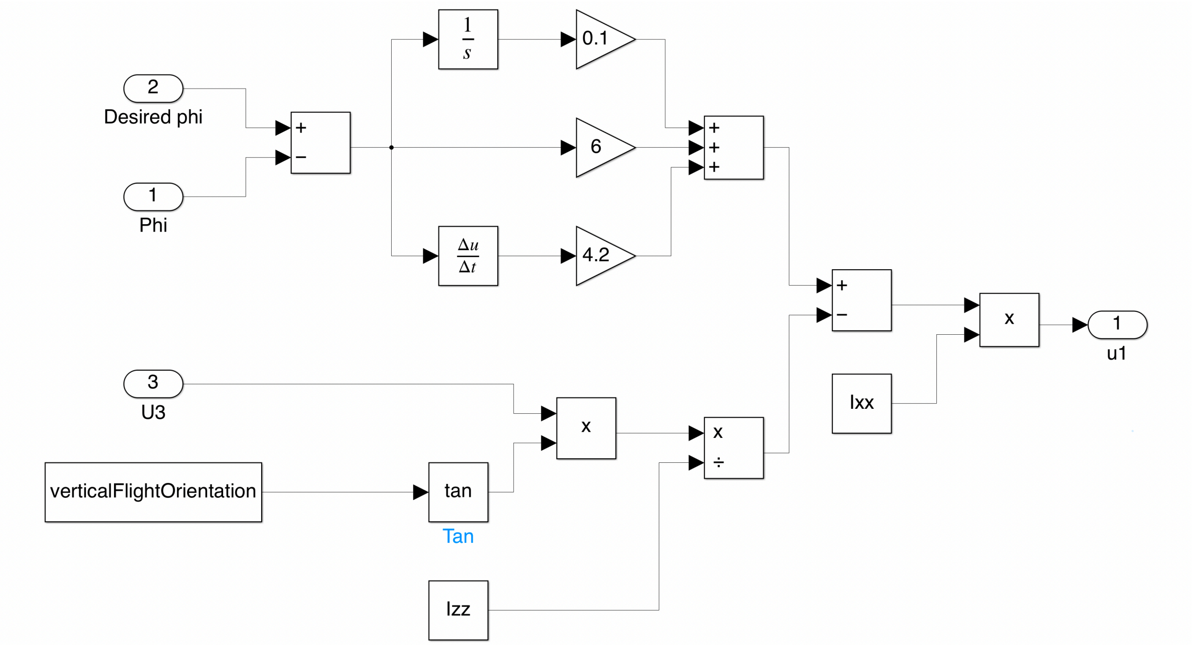


Figure A.7: Simulink block: PID controller u1

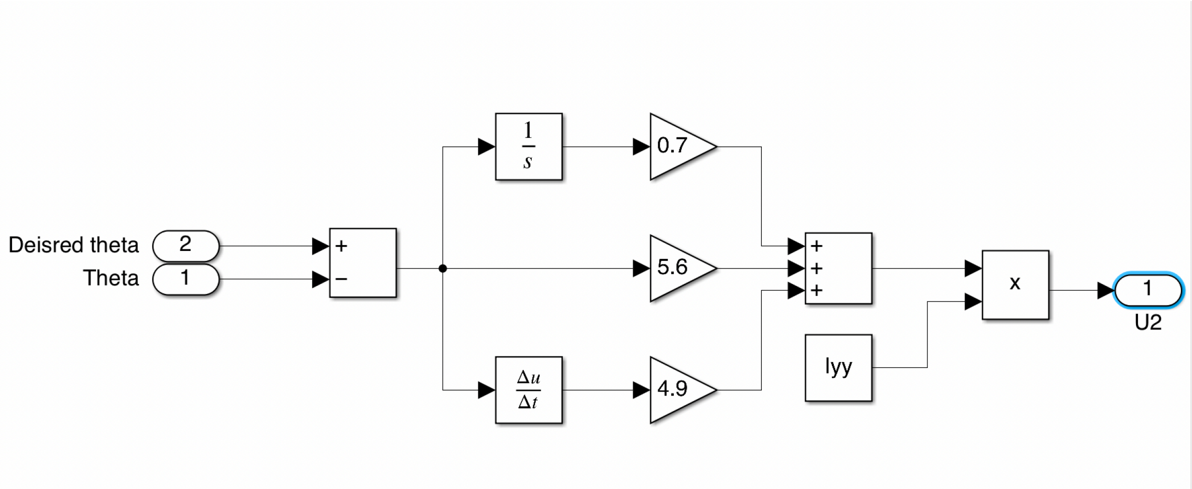


Figure A.8: Simulink block: PID controller u2

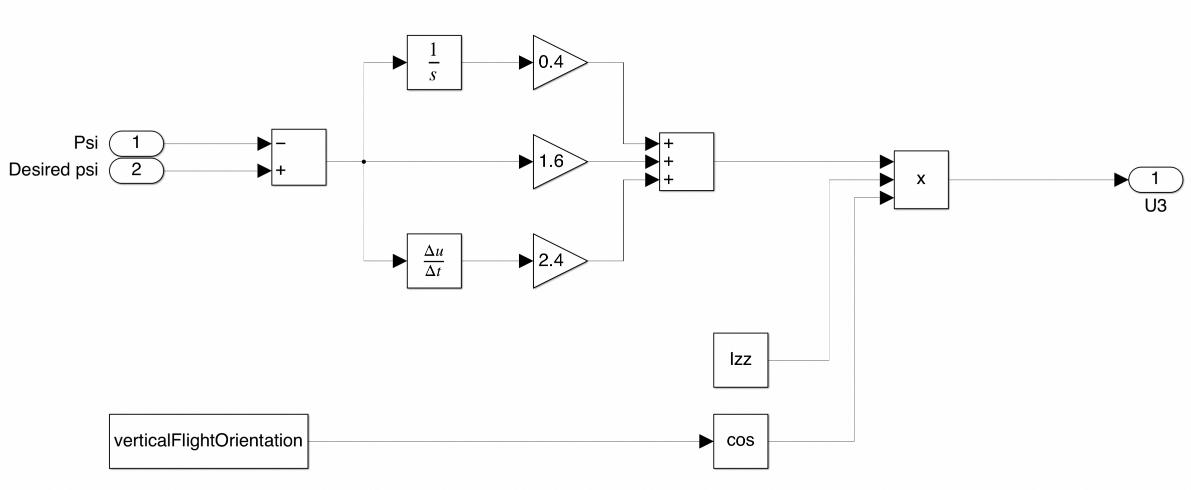


Figure A.9: Simulink block: PID controller u3

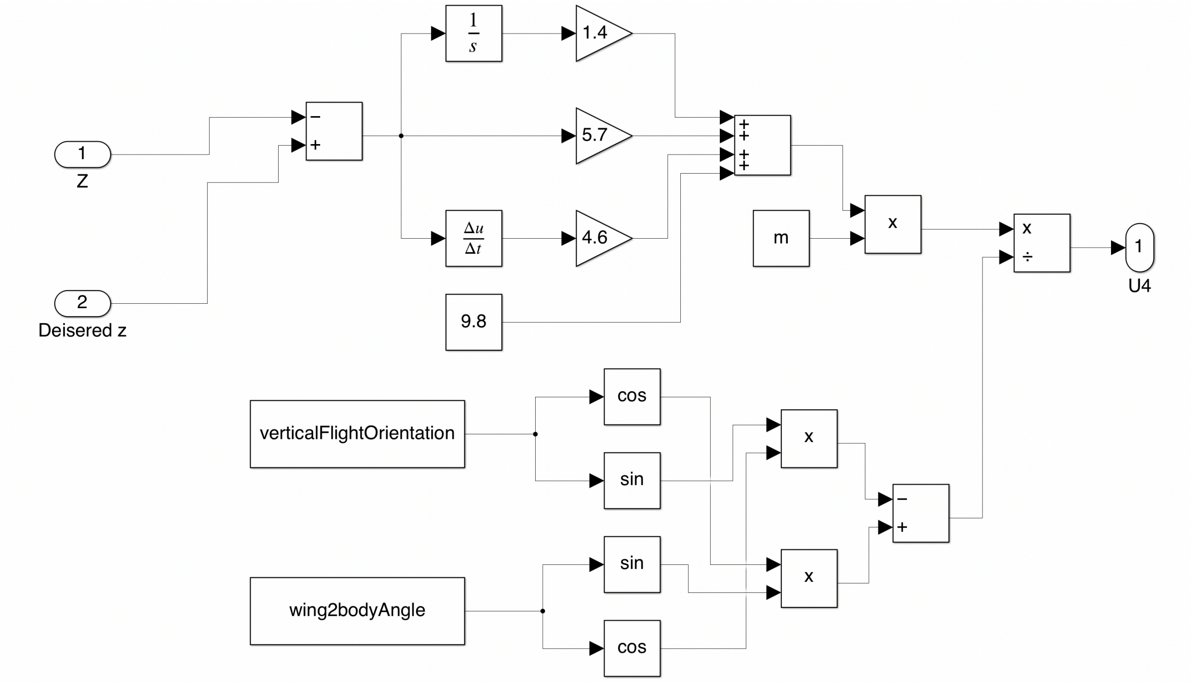


Figure A.10: Simulink block: PID controller u4

B Matlab functions

B.1. Model setup

```
g = 9.8; %m/s^2
air_density = 1.2754; %kg/m3
air_speed = 0; %m/s, 0 for vertical flight

startingRotorSpeed = 1240; % about hover rp
verticalFlightOrientation = wrapToPi(deg2rad(50));
startingPhi = wrapToPi(deg2rad(0));
startingTheta = wrapToPi(deg2rad(310));
startingPsi = wrapToPi(deg2rad(20));
startingZ = 0;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Control reference values
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

phi_desired = wrapToPi(deg2rad(0)); %radians
theta_desired = wrapToPi(deg2rad(310)); %radians
psi_desired = wrapToPi(deg2rad(0)); %radians
z_desired = 10; %meters

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Aircraft attributes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

wing2bodyAngle = wrapToPi(deg2rad(40));
Ly = 2.5; %meters, distance from wing to C.O.G. parallel to x
Lx1 = 1.6; %meters, distance to first propeller parallel to y
Lx2 = 3.2; %meters, distance to second propeller parallel to y

mw1 = 220; %kg, mass of front wing
mw2 = 220; %kg, mass of back wing

m = 2500; %kg, total mass of VTOL
[m1, m2, m3, m4, m5, m6, m7, m8] = deal(34.5); %kg, mass of propeller +
  ↪ motor combination
Ixx = Ly^2*(m2 + m4 + m6 + m8 + mw1 +mw2);
Iyy = Lx1^2*(m3 + m4 + m5 + m6) + (Lx1+Lx2)^2*(m1 + m2 + m7 +m8);
```

```
Izz = ((Lx1+Lx2)^2 + Ly^2)*(m1 + m2 + m7 + m8) + (Lx1^2 + Ly^2)*(m3 + m4 +
 ↪  m5 + m6) + Ly^2*(mw1+mw2);


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Propeller attributes
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Kd = 1;
Kmt   = 0.002;         %relation between rpm^2 and thurst
Kmtorque = 0.0003;   %relation between rpm^2 and torque
```

B.2. Plant

B.2.1. Simulink block: calculate forces

```
function Fx = Fx(W1, W2, W3, W4, Kmt, wing2bodyAngle)
Fx = 2*Kmt*(W1^2 + W2^2 + W3^2 + W4^2)*cos(wing2bodyAngle); %in hover
end
```

```
function Fz = Fz(W1, W2, W3, W4, Kmt, wing2bodyAngle)
Fz = 2*Kmt*(W1^2 + W2^2 + W3^2 + W4^2)*sin(wing2bodyAngle); %in hover
end
```

```
function U1 = calcU1(W1, W2, W3, W4, Kmt, Kmtorque, a1, a2,
 ↪  wing2bodyAngle)
U1 = Kmt*cos(wing2bodyAngle)*(a2*(-W1^2 - W2^2 + W3^2 + W4^2) + a1*(-W1^2
 ↪  - W2^2 + W3^2 + W4^2)) + cos(wing2bodyAngle)*Kmtorque*(-W1^2 + W2^2 +
 ↪  W3^2 - W4^2);
end
```

```
function U2 = calcU2(W1, W2, W3, W4, Kmt, b, wing2bodyAngle)
U2 = Kmt*b*sin(wing2bodyAngle)*(-2*W1^2 + 2*W2^2 - 2*W3^2 + 2*W4^2);
end
```

```
function U3 = calcU3(W1, W2, W3, W4, Kmt, a1, a2, wing2bodyAngle,
 ↪  Kmtorque)
U3 = Kmt*sin(wing2bodyAngle)*(a2*(-W1^2 + W2^2 + W3^2 - W4^2) + a1*(-W1^2
 ↪  + W2^2 + W3^2 - W4^2)) + sin(wing2bodyAngle)*Kmtorque*(-W1^2 + W2^2 +
 ↪  W3^2 - W4^2);
end
```

B.2.2. Simulink block: calculate angular acceleration

```
function pitch_dot_dot = pitchAcceleration(U2, Iyy)
temp = U2/Iyy;
pitch_dot_dot = round(temp*1000)/1000;
end
```

```
function roll_dot_dot = rollAcceleration(U1, U3, theta, Ixx, Izz)
temp = U1/Ixx + U3*tan(theta)/Izz;
roll_dot_dot =  round(temp*1000)/1000;
end
```

```
function yaw_dot_dot = yawAcceleration(U3, theta, Izz)
temp = U3/(Izz*cos(theta));
yaw_dot_dot =  round(temp*1000)/1000;
end
```

B.2.3. Simulink block: Calculate Euler angles

```
function [phi_dot, theta_dot, psi_dot] = angularToEuler(roll_dot,
  ↪ pitch_dot, yaw_dot, theta, psi)
   angEuler = [1 sin(psi)*tan(theta) cos(psi)*tan(theta);
             0 cos(psi) -sin(psi);
             0 sin(psi)/cos(theta)  cos(psi)/cos(theta)];
   euler = angEuler*[roll_dot; pitch_dot; yaw_dot];

   phi_dot = euler(1);
   theta_dot = euler(2);
   psi_dot = euler(3);
end
```

B.2.4. Simulink block: Calculate linear accelerations

```
function R = rotationMatrix(phi, theta, psi)
R = [cos(theta)*cos(psi)
  ↪  -(cos(phi)*sin(psi))+(sin(phi)*sin(theta)*cos(psi))
  ↪  (sin(phi)*sin(psi))+(cos(phi)*sin(theta)*cos(psi));
    cos(theta)*sin(psi) (cos(phi)*cos(psi))+(sin(phi)*sin(theta)*sin(psi))
      ↪  -(sin(phi)*cos(psi))+(cos(phi)*sin(theta)*sin(psi));
    -sin(theta) sin(phi)*cos(theta) cos(phi)*cos(theta)
    ];
end
```

```
function  [Fx, Fy, Fz] = totalForce(Fxb, Fzb, R, m, g, Kd,v_x, v_y, v_z)

    F_thrust_b = [Fxb; 0; Fzb];
    F_g = [0; 0; -m*g];
    F_drag = [-Kd*v_x; -Kd*v_y; -Kd*v_z];

    F_world_thrust = R*F_thrust_b;

    F_total = F_g + F_world_thrust + F_drag;

    Fx = F_total(1);
    Fy = F_total(2);
    Fz = F_total(3);

    Fx = round(Fx*1000)/1000;
    Fy = round(Fy*1000)/1000;
    Fz = round(Fz*1000)/1000;
end
```

```
function [x_acc, y_acc, z_acc] = getAccelerations(Fx, Fy, Fz, m)
    x_acc = Fx/m;
    y_acc = Fy/m;
    z_acc = Fz/m;

end
```

B.3. Controller

```
function [W1, W2, W3, W4] = calcRotorSpeeds(u1, u2, u3, u4, Kmt, Lx1, Lx2,
 ↪  Ly, wing2bodyAngle, Kmtorque)

X1 = (Lx1 + Lx2)*Kmt*cos(wing2bodyAngle);
Y1 = 2*Ly*Kmt*cos(wing2bodyAngle);
Z1 = (Lx1 + Lx2)*Kmt*sin(wing2bodyAngle);

torques_thrust = [
    -X1 -X1 X1 X1;
    -Y1 Y1 -Y1 Y1;
    -Z1 Z1 Z1 -Z1];

X2 = Kmtorque*cos(wing2bodyAngle);
Y2 = 0;
Z2 = Kmtorque*sin(wing2bodyAngle);

torques_torque_rotor = [
    -X2 X2 X2 -X2;
    -Y2 Y2 Y2 -Y2;
    -Z2 Z2 Z2 -Z2];
```

```
Atorque = torques_thrust + torques_torque_rotor;
Athrust = [2*Kmt 2*Kmt 2*Kmt 2*Kmt];

A = [Atorque; Athrust];

u = [u1; u2; u3; u4];

rpms = A\u;

W1 = rpms(1);
W2 = rpms(2);
W3 = rpms(3);
W4 = rpms(4);

if W1 < 0
    W1 = 0;
elseif W1 > 3600^2
    W1 = 3600^2;
end

if W2 < 0
    W2 = 0;
elseif W2 > 3600^2
    W2 =3600^2;
end

if W3 < 0
    W3 = 0;
elseif W3 > 3600^2
    W3 = 3600^2;
end

if W4 < 0
    W4 = 0;
elseif W4 > 3600^2
    W4 = 3600^2;
end

W1 = sqrt(W1);
W2 = sqrt(W2);
W3 = sqrt(W3);
W4 = sqrt(W4);

end
```

Bibliography

[1] AIAA/IEEE, “Aiaa/ieee eats student design competition,” November 2020.

[2] M. H. Sadraey, *Aircraft design: A systems engineering approach*. John Wiley & Sons, 2012.

[3] EASA, “Artificial intelligence roadmap,” tech. rep., 2020.

[4] H. Veerman, “Preliminary multi-mission uas design,” 2012.

[5] C. L. Krishna and R. R. Murphy, “A review on cybersecurity vulnerabilities for unmanned aerial vehicles,” in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, pp. 194–199, IEEE, 2017.

[6] Y. Ko, J. Kim, D. G. Duguma, P. V. Astillo, I. You, and G. Pau, “Drone secure communication protocol for future sensitive applications in military zone,” *Sensors*, vol. 21, no. 6, p. 2057, 2021.

[7] ITU, “Characteristics of unmanned aircraft systems and spectrum requirements to support their safe operation in non-segregated airspace,” tech. rep., 2009.

[8] R. J. Kerczewski and J. H. Griner, “Control and non-payload communications links for integrated unmanned aircraft operations,” *NASA*, 2012.

[9] R. S. Stansbury, M. A. Vyas, and T. A. Wilson, “A survey of uas technologies for command, control, and communication (c3),” in *Unmanned Aircraft Systems*, pp. 61–78, Springer, 2008.

[10] A. El-Fatatry, “Lecture notes mems aerospace applications,” March 2003.

[11] J. Guerrero-Castellanos, H. Madrigal-Sastre, S. Durand, N. Marchand, W. Guerrero-Sánchez, and B. Salmerón, “Design and implementation of an attitude and heading reference system (ahrs),” in *2011 8th International Conference on Electrical Engineering, Computing Science and Automatic Control*, pp. 1–5, 2011.

[12] P. Gąsior, S. Gardecki, J. Gośliński, and W. Giernacki, “Estimation of altitude and vertical velocity for multirotor aerial vehicle using kalman filter,” in *Recent Advances in Automation, Robotics and Measuring Techniques* (R. Szewczyk, C. Zieliński, and M. Kaliczyńska, eds.), (Cham), pp. 377–385, Springer International Publishing, 2014.

[13] J. Diebel, “Representing attitude: Euler angles, unit quaternions, and rotation vectors,” October.

[14] F. S. H. Triantafyllou, Michael S, “Lecture notes maneuvering and control of marine vehicles,” Fall 2004.

[15] C. tytler, “Modeling vehicle dynamics – quadcopter equations of motion,” 2017.

[16] S. W. J. Peraire, “Lecture l26 - 3d rigid body dynamics: The inertia tensor,” Fall 2008.

[17] Q.-G. Wang, T.-H. Lee, H.-W. Fung, Q. Bi, and Y. Zhang, “Pid tuning for improved performance,” *IEEE Transactions on control systems technology*, vol. 7, no. 4, pp. 457–465, 1999.

[18] H. Bolandi, M. Rezaei, R. Mohsenipour, H. Nemati, and S. M. Smailzadeh, “Attitude control of a quadrotor with optimized pid controller,” 2013.

[19] K. T. Öner, E. Çetinsoy, E. SIRIMOĞLU, C. Hançer, M. Ünel, M. F. Akşit, K. Gülez, and I. Kandemir, “Mathematical modeling and vertical flight control of a tilt-wing uav,” *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 20, no. 1, pp. 149–157, 2012.

[20] M. Usman, “Quadcopter modelling and control with matlab/simulink implementation,” 2020.

[21] G. M. van der Zalm, “Tuning of pid-type controllers: literature overview.”