UiT The Arctic University of Norway

Faculty of Engineering Science

Tethered Quadcopter Control, Simulation and Modeling platform for a small USV

Sindre Wæringsaasen

Master’s thesis in Aerospace Control Engineering STE-3900 May 2021

Abstract

Maritime Robotics main intent of the thesis is to develop understanding of a small deployable camera platform capable of providing elevated situational awareness for a small USV using a tether. The thesis goes into the state-of-the-art methods of usage and control of a tethered drone. The thesis goes into the building and operating of the QAV250 Pixhawk 4 mini drone and making a drone simulation with the help of Simulink.

Summary

The thesis is a summary and hopefully a guiding way to how tether drone could operate and work. It is also going into the challenges and opportunities in using a tether drone instead of a normal drone. The thesis goes into the state-of-the-art methods of usage and control of a tethered drone. The thesis goes into the building and operating of the QAV250 Pixhawk 4 mini drone. The complexity that happens when you try to use a commercial drone for an experiment that does not comply to the standard operating procedure of the drone. Even though many of the experiments were hampered, the relatively simple solution of how a tether drone could be controlled was found from just two joysticks in ether end of the tether. With the help of a digital weight sensor, it is also possible to know if the tether is slack and the drone is not fling to its maximum extent. The results have shown that most of the sensors can be placed on the USV. Only the USV would need to know its location for the whole tether system to work. The thesis also a goes into the making a drone simulation with the help of Simulink. The results of this simulation where mixed, making a quadcopter model and plant is a complex task that that makes it easy to make mistakes. The simulation made in this thesis made only a partial success in simulating the system. With the system only able to simulate the drone without the IMU and the motor system.

Preface

This reason for this thesis is to give awareness for tethered drones and its emerging market. For this thesis special thanks goes out to Maritime Robotics and there contact person Eirik Moholt. Furthermore thanks goes to aerospace head Lecturer Andersen, Tom Stian and the thesis advisor: Associate Professor Tu Dac Ho. Last, great appreciation to all the classmates of Aerospace control engineering which have been a great motivating and knowledge base for making this thesis complete.

Contents

Acronyms

6DOF six degre of Freedom. 8

BPL Broadband over power lines. 4

CNN Cable News Network. 3

ECEF earth-center earth fixed. 9

LTE Long-Term Evolution. 2

NED North East Down frame. 9

NFL National Football League. 3

NMC (Lithium nickelmanganese cobalt oxide Battery. 1

PDB power distribution board. 27

PID Proportional-Integral-Derivative. 20

RF Radio frequency. 2

TUAV tethered UAV. 2

UAV unmanned aerial vehicle. 1

USV Unmanned Surface Vehicle. ii, 1, 44

List of Figures

1 Introduction

Maritime Robotics is a Norwegian company providing innovative solutions for unmanned systems operating in the air and on the surface. They wish to extend the capabilities of the one of their small (USV) Unmanned Surface Vehicle with a deployable airborne camera platform to provide elevated situational awareness for the USV. The camera platform should be able to maintain position directly over the USV, be small in size and have the ability to rotate for a complete view. To ease landing and take-off a possible tethered solution should be explored. The use of Tethered drones (TUAV) in this master thesis is a reflection on the emerging marked of usage of these types of drones. There have been a steady and growing usage of TUAV over the years. It is still not widely used because of the relatively young marked of conventional unmanned aerial vehicle (UAV). This however has not made research projects and users stop thinking of the benefits of using a TUAV. Many innovative ideas have been proposed and a couple of them are in fact in use at the moment of writing.

1.1 Literature review

1.1.1 Tethered UAV Usage

1.1.1.1 Tethered UAV compared to batteries There are many benefits of having a tethered UAV but the main is that power can be provided continuously to the craft without it needing to land and change batteries. There can also be a reduction I weight that the space craft will gain by removing the on-board batteries. Lithium nickel manganese cobalt oxide (NMC) batteries have a typical specific energy of 200Wh/Kg [4]. This means that for sustained flights over a longer duration the battery pack would have to grow exponentially, and the UAV would have to increase in size. Dji Mavic 2[5] has a battery net weight of 297 grams with a takeoff weight of 907g. There are not a lot of extra weight for takeoff. With the provided battery the Mavik 2 can have a hovering time of 29 minutes. For applications where there is a need for longer hover times specially over the 1 hour mark a tethered solution would be preferred.

1.1.1.2 Uses for tethered UAV Tethered drones today are having many different uses, but they are mainly used to provide constant surveillance capacity that gives a better and more cost effective situational warned that that other monitoring systems can provide. The biggest sectors for their use are disaster relief, Military surveillance and intelligence gathering, cultural events where both surveillance for crowd control and security. At events broadcasters[6] can also use tethered UAV to record videos above the crowds. Tethered drones can also be used to do inspections of wind turbines and oil installations.

1.1.1.3 Disaster relief In disaster relief you can get a overview of the surrounding disaster area. By using thermal imaging, you can find and plan where to place your limited rescue resources. Tethered UAV can stay up for hours recording changes that would not be detected by a single UAV trip. This is especially important in forest fires and volcano eruption where UAV can track the fires and magma for long periods helping warn civilians and rescuers of changing condition[7]. The maps created helps emergency services see where the most urgent areas are and where they do not need to allocate resources. A project called Telelift[8] is developing a TUAV that works as a temporary replacement of cell phone towers in disaster areas and in locations where capacity on the existing cellular network is limited. These TUAV could hover up to a month at the time and provide both rescuers and civilians' cellular network. AT&T[9] has also developed their own form of cellular TUAV that they since have used after Hurricane Michael[10]. There they provided cellular LTE network for over 6500 customers and helped first responders to resume communication and speed up the recovery work.

1.1.1.4 Military Surveillance and intelligence gathering The historical methods of military Surveillance and intelligence gathering have mostly been by using airplanes and satellite reconnaissance. In the last century drones have become more prevalent and cheaper to operate than airplanes the benefit of usage of drones compared to satellites is that you can get a real time information over a longer time. Satellites can only stay over a target for a certain time. And the cost of its services is much more than just buying a drone. However, with a UAV you have to change batteries or fuel after a certain point, this means that you need extra personnel to do this. The benefit of having a TUAV is that the device can operate for long periods of time without needing to be recharge. Another important benefits of using a TUAV is that the communication can be directly transferred vi fiber optic cables to the ground. This means that the live video can be streamed in 4k resolution and even higher. This also benefit in reducing the RF signature of the device since it does not rely on RF signals to operate and to stream video. This is important to conceal the operating group using this to scout out an area undercover. It is also beneficial with fiberoptic cable since they cannot be jammed by RF jammers. In India, the border security force has deployed TUAV from Israel[11] to protect the border with Bangladesh. On the other side of the globe in France, the French defense agency has invested in Elistair[12] which is one of the major world producers of TUAV. The project called AIRWATCH is going to be a complete system with UAV and a tether box that the French military can use for reconnaissance and surveillance. In 2017 United States Secret Service[13] tested the usage of TUAV at Trump National Golf Club to see if it could be a valid way of protecting the president. With the better video quality, they could survey a larger area than if they used a normal UAV. This means that in the future TUAV can be used to protect dignitaries and VIP personnel.

1.1.1.5 Event and media organizations Event and media organizations have for many years now use normal UAV for providing video broadcast of events. The benefit of using UAV‘s is that you do not need to hire expensive helicopters to do the filming for you. The problem is that these drones tend to be large in size since the camera gear is much larger than normal recreational UAVs cameras. Because of the increase weight the battery must be larger to compensate. Transmission systems are a problem to because they must be used over the wireless internet or using the phone 4G/5G system. This is the reason why many broadcast companies have been investing in TUAV. Here the power system can power the TUAV through the entire event without needing to land. This reduces the overall size of the drone, combined with the reduction of power that is needed for the RF transmitter. During the 2019 Super Bowl[14] in Atlanta, USA CNN used a TUAV from Elistair to provide an overview of the stadium for the news coverage. NFL is looking into using TUAV in their future games. One benefit NFL have seen is that since the drones are tethered the risk of the drone being taken by wind gust are lessen since the drone cannot move outside the area that it is tethered. The need for a safety area is therefor only needed for a diameter of the length of the tether. As mentioned earlier the Fox Sport[6] used tethered drones to film the Daytona 500 race for their sport coverage. This is particularly good use of a TUAV since the race runs for about 3 hours and 40 minutes.

1.1.1.6 Offshore There are many potential uses of TUAV in the offshore sector. In the oil sector flare towers[15] can be inspected for longer periods of time than that of conventional UAVs. The benefit of using drones to inspect flare towers is that the flare does not need to bee shut down like it needs to if humans are inspecting the tower. With the no longer need of humans to inspect the tower this reduces the need to work in heights. With inspections with tethered drone the drone operator can be stationed on land and do the inspection work without the need of extra personnel on the platform. There is also a benefit of doing the inspection while the flare tower is active. This is that parts can be pre ordered after such an inspection and can be therefore installed when the scheduled shut off, of the flare stack is happening. It is not only the oil business that can benefit from TUAV. Offshore wind installations are perfect for the use of TUAV and autonomous marine vehicles to inspect their installations. The cost and time, it uses to ship people out to sea and to inspect the windmills. With drones you can install ultrasound[16] equipment that can inspect the structural integrity of the blades of the windmill. This is normally done by a three-man team. There is a project team in UK called MIMRee[17] that is researching and developing a autonomous package to inspect and repair wind installations of the coast of UK. The package includes three pars. The first part is an autonomous ship that can drive to the instillation. The second part is a drone that can inspect and fly a rope up the windmill. Last part is a robot that will climb up and fix any damages that the inspection revealed. The only problem with this solution is that inspecting multiple wind mils will make it necessary to change batteries or charge the drone on the mothership. This can be remedied by using a TUAV. There is also in benefits with using better optics because the transmission speed is increased. There is also possible for drones to have water container or plugged to water hose for TUAV. This can be used both to clean the installations and can bee filled with anti-freeze to prevent ice creation on the rotors of the instillation.

1.1.2 Producers of Tethered UAV

1.1.2.1 ELISTAIR Elistair[18] is the leading tethered UAV company in Europe. Located in Dardilly France they are the main TUAV supplier to the French armed forces. There main product is the Orion 2 tethered drone. This drone has a height limit of 100meters and can carry 2kg of payload. Its theater is made of a Kevlar reinforced micro theater that gives it continuously power. It benefits is that it has a detection range of 10 km making it perfect for military, law enforcement and private security. They are currently offering two types of base stations. The SAFE-T 2 has duel coms with both fiberoptics and BPL[19] data links. The other base station that they offer is LIGH-T 4 base station that together with an air model that you install in the drone you can convert a huge variety of commercially available UAVs to become a tethered drone. The included wire for this unit is 70m long.

1.1.2.2 Skysapience Skysapience[20] is an Israeli TUAV company that is one of the world leading companies in this technology. It was founded in 2010 and have customers from US border control to the Dominican Republic and Israeli defense forces. Its enterprise product has two categories the HoverMast-xx-C variant that is for normal enterprise applications. The Enterprise marine intelligent HoverMast-xx-CN is the other category that is specialized for marine applications. The marine version is especially built for saltwater environments with strengthened salt water resistant components. This product is made for uses on small patrol vessels and unmanned surface vehicles. The enterprise series comes in three different lengths of tether 50m, 100m and 150meters. The cable serves as both a power supply and wideband data link. There are also a defense line that uses the same products that the enterprise version uses there does not seam to be any different in the product between the defense and enterprise version. there might be different software integration on the defense version.

1.1.2.3 Other companies Flare Dynamics[21] is an Aerospace Precision Engineering firm specializing in the arena of Unmanned Systems and Composite Materials. The company in based in Singapore and produces the Lifeline tethered solution. This is a box works as an external battery extending the flight time of the TUAV it can with help of modules be hooked up to a variety of different commercial drones. The box has a 60m line that provides power. One negative aspect of this solution is that the lifeline only comes with power and not data transfer. This makes the communicate system rely on the drones Wi-Fi signals. Unmanned systems and solutions, LLC (USAS)[22] is an American company that produces the Long Endurance Aerial Platform (LEAP). LEAP is an all-in-one system with a heavy lift UAV, a tether with both power and gigabit connection and a base station. With a payload capacity of 9.5kg a cellular booster can be connected to be worked as a base station. A benefit that USAS has is that it will customize their products and design solutions to there needs.

1.1.3 Control theory

Control theory for tether drone are not very prevalent on the internet. “Taut Cable Control of a Tethered UAV” [23] a 2014 IFAC World Congress publication goes into one way of keeping and controlling the cable taut by giving a design of stabilizing control law for a UAV. There solution is to use a cascade control schemes to make the drone asymptotically stable. Some papers go into connecting multiple UAV with each other with cables. The papers Aerial Co-Manipulation With Cables[24] and Systems of Tethered Multi copters [25]. In the paper Tethering System for Unmanned Aerial Vehicles [26] they make a control system that regulates the motor for the spool part of the system so to regulate the tension on the tether, this prevents entanglement and excess energy use from the drone. Taeyoung Lee paper[27] goes more into the control system with focus on a dynamic model of a tethered quadrotor this includes the coupling between distortions of the tether and the actions of the UAV. The paper also explains ways of avoiding singularities and complexities associated with the local coordinates. Most of the control theory try to solve the tension problem of the tether mathematically. From the research only Tethering System for Unmanned Aerial Vehicles uses a external sensor. None of the paper so far has been using a load cell sensor or tension sensor as called in this thesis for the use of controlling the quadcopter. While researching the use of joystick sensors calculate the angles of the tether and drones where not mentioned either in the reports. To help on controlling theory for a tether the lecture from Tether Fundamentals from J. Peláez and tethered theory [28] and the paper from Ricardo Miguel [29] explaining the Catenary Curve Effect on tethered drones.

1.2 Contributions and scope of the thesis

As mention in the Literature review tether drones are a emerging marked with very little research done to it. Control theory without the use of GPS or cameras for guidance on tether drones are rare and few have written about it. The possibility with a tether to gain the information needed makes it superb for usages in environments where external telemetry and visual signal are not available. There are 4 main goals of this thesis that defines the scope of what is going to be done. Survey the state of art of methods used for controlling tethered UAVs in the industry as already done in the Literature review. Develop and simulate a controller that can take-off from a USV, follow and land on the USV. Develop a prototype and perform experimental tests to demonstrate the feasibility of the design. Finally Work closely with Maritime Robotics to test and further develop a solution.

1.3 Report outline

The report first goes into Preliminaries here motions of the vehicle will be explained thereafter the moments of inertia of the system and the coefficients of the system is described. Next rotor inertia angular acceleration and Atan2 is calculated. Part 2 is the Simulink model, the making of the Simulink model is explained here with all the steps needed. Part 3 is the construction of the QAV250 Drone with all the steps that was used. Part 4 explains the drone operating program and the difficulties of controlling the drone. Part 5 is talking about what experiments was done and not done. Part 6 talk about tradeoffs in the work and what could have been done to make it work better. Last is Part 7 where the conclusion of the thesis and the future works are considered.

2 Preliminaries

2.1 Notation

Symbol	Description	Unit
π	mathematical constant	3.14159
ω_1	Angular velocity motor 1	rad/s
ω_2	Angular velocity motor 2	rad/s
ω_3	Angular velocity motor 3	rad/s
ω_4	Angular velocity motor 4	rad/s
ϕ	Roll Angle	rad
ψ	Yaw Angle	rad
θ	Pitch Angle	rad
$\dot{\phi}$	Roll Angular Velocity	rad/s
$\dot{\psi}$	Yaw Angular Velocity	rad/s
$\dot{\theta}$	Pitch Angular Velocity	rad/s
$\ddot{\phi}$	Roll Angular Acceleration	rad/s^2
$\ddot{\psi}$	Yaw Angular Acceleration	rad/s^2
$\ddot{\theta}$	Pitch Angular Acceleration	rad/s^2

U1	Vertical thrust	m·kg/s^2
U2	roll thrust	m·kg/s^2
U3	Yaw thrust	m·kg/s^2
U4	Pitch thrust	m·kg/s^2
$\mathrm{I}_{xx}, I_{yy}, I_{zz}$	Moments of inertia (roll, yaw, pitch,)	Kgm^2
X, Y, Z	Position	m
$\dot{X}, \dot{Y}, \dot{Z}$	Velocity	m/s
$\ddot{X}, \ddot{Y}, \ddot{Z}$	Acceleration	m/s^2
C_T	Thrust coefficient	N/s^2
Cl	Lift coefficient	Unit
Cd	Drag coefficient	Nm/s^2
Ke	Rotor inertia con.	V·s/rad
Kv	Motor velocity con.	rad/V·s
Kt	Motor torque constant	N·m/A
Jr	Rotor inertia	Kgm^2

Table 1: mathematical notation used in the thesis

2.2 Theoretical/Technological preliminaries

2.2.1 Vehicle motion

To understand how the drone is going to function we have to know the motion of the drone and the motions of the controller joystick that is connected to the tether. quadcopters have 6 degrees of freedom (6DOF) this gives the position and orientation. The equation of motions gives change in position and orientation. for this we need a frame of reference from which we can give the initial position. since the drone is only being simulated and tested in small environment, we presume that the reference area is flat meaning that the earth is presumed flat to simplify the simulation.

2.2.1.1 Six degrees of freedom

Six degrees of freedom tells what motion and position a vehicle is moving. There are two important movements that make up the six degrees of freedom these are the translation movement in three perpendicular axis and the rotational movement about the three-perpendicular axis. The translational movements are made up of forward/backup movement which gives the X axis of moment. The Have or up and down moment is represented as the Z axes. The last movement is the sway which is the left to right movement is represented as the Y axis. Then there is the rotational movement which give the angle of off the craft. Here you have roll which means tilting side to side, then you have pitch which is tilting forward and backwards. Yaw is the last rotational moment which is turning left and right. You can see an example of all the movement combined in figure 1.

2.2.1.2 Frame of reference and Orientation

For the frames of orientation, we need to know the rotational moment of the craft as mentioned previously this is pitch (θ), Yaw (Ψ) and Roll (Φ) This are three Euler angles in radians between -π and + π. As you can see on figure 1 where it shows the Euler angles explained. We are using NED[30] frame which stand for North East Down frame and is used to give the acceleration, velocity and the position of the aircraft relative to a fixed point. For this thesis, the fixed point is at the end of the wire that connects to the drone at the other end there is the joystick which has the [0,0,0] reference frame fixed point. The joystick also has its own orientation which is roll and yaw. In the local NED frame, the Y axis is pointed east while the X axis is pointed towards the north. The Up or Z direction is the height of the object this is limited by the tether range of the system. It is also limited that it cannot be a negative Z value. The with a positive (Φ) the drone travels east and with negative it travels west. The same apply to (θ) where a positive here makes it go north and south. Another system of reference that is used is earth-center earth fixed ECEF this is a geographic and cartesian coordinate system here the X, Y, Z, represent coordinates. In this thesis the center is not at the earth, it has the same reference point as the NED frame with the joystick being the Zero-point ECEF makes it easier to see where the drone is however it does not show the orientation of the drone.

2.2.1.3 Euler Angels

There are different methods to get the orientation of the vehicle, there are two schools of thought here Euler angels and Quaternion. While Quaternion are in theory better to use, they tend to be more difficult to simulate an so makes the simulation easier Euler angels are used. Euler angels are also much easier for humans to understand. With Euler angel you can run into the problem of gimbal lock. With Euler angels we have the possibility to get the orientation compared to the body frame. We find this by using the inertial frame to body frame transformation.

$$R_{v2}^{b}(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\phi & sin\phi \\ 0 & -sin\phi & cos\phi \end{pmatrix}$$

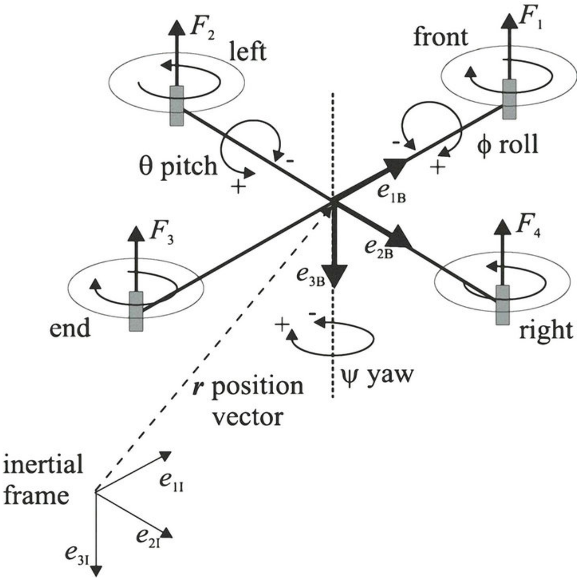


Figure 1: Euler angles:Singh, Inderpreet. “Quadcopter Project- Part 3 – The Math!” The Embedded Code, Inderpreet Singh, 1 June 2015, embeddedcode.wordpress.com/2015/06/01/quadcopter-project-part-3-the-math/.

$$R_{v1}^{v2}(\theta) = \begin{pmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{pmatrix}$$

$$R_{v1}^{v2}(\psi) = \begin{pmatrix} cos\psi & sin\psi & 0 \\ -sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.1}$$

From these three matrix’s (2.1) we can do the transformation from inertial frame to body frame matrix.

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\phi & sin\phi \\ 0 & -sin\phi & cos\phi \end{pmatrix} \times \begin{pmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{pmatrix} \times \begin{pmatrix} cos\psi & sin\psi & 0 \\ -sin\psi & cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.2}$$

$$R_v^b(\theta, \psi, \phi) = R_{v1}^{v2}(\theta) R_{v1}^{v2}(\psi) R_{v2}^{b}(\phi)$$

Now if we multiply these (2.2) together we get the inertial to body frame transformation matrix.

$$R_v^b(\theta, \psi, \phi) = \begin{pmatrix} cos(\theta)cos(\psi) & cos(\theta)sin(\psi) & -sin(\theta) \\ -cos(\phi)sin(\psi) + cos(\psi)sin(\theta)sin(\phi) & cos(\psi)cos(\phi) + sin(\theta)sin(\phi)sin(\psi) & cos(\psi)sin(\phi) \\ sin(\psi)sin(\phi) + cos(\psi)cos(\phi)sin(\theta) & -sin(\phi)cos(\psi) + cos(\phi)sin(\theta)sin(\psi) & cos(\theta)cos(\phi) \end{pmatrix} \tag{2.3}$$

With the formulas from (2.2) we can figure out the orientation of the object with respect to body frame . It is also possible to find the body to inertial frame by using the inverse of the transposed matrix.

2.2.2 Thrust equations

For the Simulink model we need to describe the four Thrust equations of the model. 4 different thrust equations are used. It is important to know the coordinate system for the drone if it is in + or x configuration.

+ coordinate configuration.

Vertical thrust.

$$U1 = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \tag{2.4}$$

Roll thrust.

$$U2 = b(-\omega_2^2 + \omega_4^2) \tag{2.5}$$

Pitch thrust.

$$U3 = b(\omega_1^2 - \omega_3^2) \tag{2.6}$$

Yaw thrust.

$$U4 = d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{2.7}$$

X coordinate configuration

$$U1 = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \tag{2.8}$$

Roll thrust.

$$U2 = bsin(\frac{pi}{4})(-\omega_1^2 - \omega_2^2 - \omega_3^2 + \omega_4^2) \tag{2.9}$$

Pitch thrust.

$$U3 = bsin(\frac{pi}{4})(\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \tag{2.10}$$

Yaw thrust.

$$U4 = d(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{2.11}$$

U1 gives thrust to all four motors to increase the height of the drone. Roll thrust or U2 gives thrust to between motors 4 and 2 and rolls the drone. U3 changes the pitch of the drone while U4 changes the yaw. For U3 and U4, motors 3 and 1 changes, and 2 and 3 changes.

We also need to find the overall angular velocity of all the propellers combined. Since not all the propellers are spinning in the same direction, we cannot just add them together. This can be seen in (2.12) where we both add and subtract the values.

$$\omega^t = \omega^1 - \omega^2 + \omega^3 - \omega^4 \tag{2.12}$$

2.2.3 Moments of inertia

Moments of inertia is the quantity that determines the amount of torque needed for a wanted amount of angular acceleration around a rotational axis. In layman's term it is the amount of moment needed to stop and move a rotating object. Since the drone is an 3D object, we need to know the moment around three axis. This is shown in matrix (2.13) where it is important to note that the drone has a symmetrical body.

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{2.13}$$

2.2.3.1 Calculating Moments of inertia

There are many ways to calculate the Moments of inertia on the drone. With the advanced method you need to calculate all the components of the quadcopter this means the motors the arms and all the wire connectors on the arms to get the most accurate representation for finding the inertia. In this project however we are going to use a simplified method [31] that uses the mass of the drone and that we assume that it is uniform and rigid. Because of this we can say that the mass is distributed the same for all for arms.

M = Mass of the drone = 0.585Kg
L = Length of drone arm =0.0671m
Ls = Length of drone arm from center = 0.124m
We need to multiply the value with 4 since we know that there is 4 arms. we also need to divide the mass by 4 to each of the arms.

$$I_{zz} = (\frac{1}{3}\frac{M}{4}l^2) \times 4 \tag{2.14}$$

$$I_{zz} = (\frac{1}{3} \times \frac{0.585}{4} \times 0.0671) \times 4 \tag{2.15}$$

$$I_{zz} = 0.0031kgm^2 \tag{2.16}$$

To canlculate the Ixx and Iyy you use the same formula as above (2.16) however you change the length by using the distance from the motor to the center of the drone.Ixx and Iyy are equal to each other.

$$I_{xx} = I_{yy} = 0.0021kgm^2 \tag{2.17}$$

2.2.4 Thrust coefficient

The trust coefficient is used to calculate the thrust of the vehicle it can be obtained by finding the trust value at different RPMs however if you do not have the RPM there is a collection of formulas that you can use [32] to get the thrust coefficient

d = Propeller diameter.
P = Propeller pitch.
k = is a found by Cord of the propeller divided by the diameter of the propeller.
e_d = pitch to diameter ratio called Diameter effectiveness A.1.
θ = Blade twist angle = $tan^{-1}(\frac{p}{\pi \times d})$

$$C_T = \frac{4}{3}k\theta[1-(1-e_d)^3] - k(\sqrt{k(1+k)}-\sqrt{k})[1-(1-e_d)^2] \quad (2.18)$$

This formula gives an error rate of ±10% because of the of inflow velocity.

2.2.5 Lift coefficient

Lift coefficient is a number that substitutes advances parameters that normally is used to calculate lift. These advanced parameters can be body shape, air viscosity, compressibility and inclination. It is normally standard to use a wind tunnel to calculate lift coefficient. However, for my simulation it would be to advance to do this even though we have a wind turbine at the University. There is a substitute math formula that is correct however it is not as accurate as the values found in a wind turbine. [33]

Cl= $2 \times \pi \times$ rad angle of attack (A)

2.2.6 Drag coefficient

Drag coefficient is the amount of aerodynamic drag that the body generates. It depends on the inclination size and shape of the propeller and the airflow passing it. There are other factors like humidity but for our simulation we will simplify it. The induced drag coefficient Cd is equal to the square of lift coefficient divided by pi times the aspect ratio of the lifting surface AR times the efficiency factor(2.20).
AR(2.19) is found by taking the span of the wing to the power of 3 divided by area of the wing surface. The power of the span depends on the amount of propeller blades that is used. Since there are 3 propeller blades on each

propeller on the drone. Efficiency factor or Oswald's efficiency factor which it is also called is a value between 0.7 and 1. Where 1 is for an elliptical distribution while 0.7 is normal for a rectangular wing. Since elliptical wing has the least drag compared to other wing shapes.

$$AR = \frac{s^3}{A} \tag{2.19}$$

$$C_d = \frac{Cl^2}{\pi \times AR \times e} \tag{2.20}$$

2.2.7 Rotor inertia

Rotor inertia is the amount of resistance that the rotor must overcome to accelerate or decelerate.

The equation[34] used is using the propeller mas and the radius of the propeller to calculate the moment of inertia.

$$J_r = \frac{1}{2} \times mR^2 \tag{2.21}$$

2.2.8 Angular acceleration

There are not any different between the + and x orientation on the system for angular acceleration calculation.

$$\ddot{\Phi} = \frac{\dot{\Theta}\dot{\Psi}(I_{yy} - I_{zz}) + J_r\dot{\Theta}\Omega_t + l(U2)}{I_{xx}} \tag{2.22}$$

$$\ddot{\Theta} = \frac{\dot{\Phi}\dot{\Psi}(I_{zz} - I_{xx}) - J_r\dot{\Phi}\Omega_t + l(U3)}{I_{yy}} \tag{2.23}$$

$$\ddot{\Psi} = \frac{\dot{\Theta}\dot{\Phi}(I_{xx} - I_{zz}) + l(U4)}{I_{zz}}$$

(2.24)

2.2.9 Atan2

Artan is used to calculate a angle from a X,Y coordinate.

$$Atan2(x,Y) = \begin{bmatrix} arctan(\frac{y}{x}) & & if \quad x > 0, & & \\ arctan(\frac{y}{x}) + \pi & if & x < 0 & and & y \geq 0, \\ arctan(\frac{y}{x}) - \pi & if & x < 0 & and & y < 0, \\ +\frac{\pi}{2} & if & x = 0 & and & y > 0, \\ -\frac{\pi}{2} & if & x = 0 & and & y < 0, \\ undefined & if & x = 0 & and & y = 0 \end{bmatrix} \tag{2.25}$$

2.2.10 Control angle and Drone Higth

$$a = \sqrt{x^2 + y^2} \tag{2.26}$$

$$p = \sqrt{X_{max}^2 + Y_{Max}^2} \tag{2.27}$$

$$z = \sqrt{p^2 - a^2} \tag{2.28}$$

$$Angle(Ca) = Tan^{-1}\frac{a}{z} \tag{2.29}$$

h = Hight of drone Ca= control angle T = Tether length

$$h = Cos(Ca) \times T \tag{2.30}$$

3 Main result

3.1 Simulink Model

The Simulink scrip follows the basic properties of how a Quadcopter system works. In this thesis we are using a closed loop system [1]. This is a universal control system that is used in many of today's machinery and a design of this can be seen in figure 2. Two paper from Usman, Muhammad[31] and Nicholas Ferry[35] where the only other Simulink simulations of a quadcopter on the internet and there work helped a lot in this simulation.

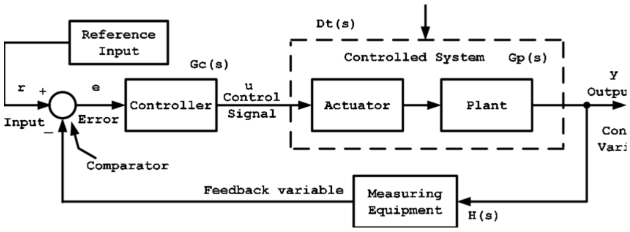
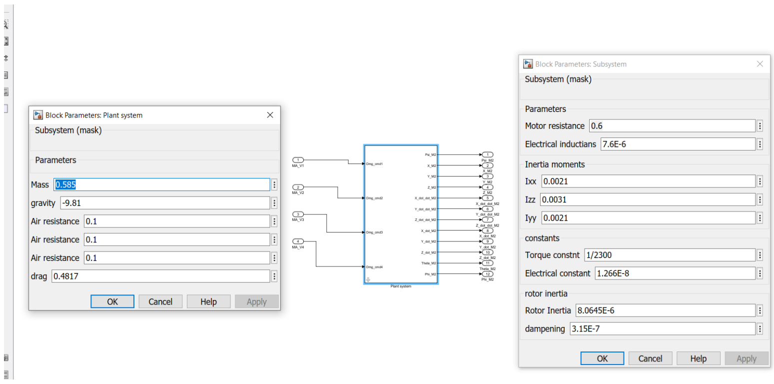


Figure 2: Block diagram of a closed-loop control system[1]

3.2 Plant model

The plant model takes in a angular velocity inputs and in turn turns out a sett of variables that are the position and the orientation angle of the quadcopter. The plant model is represented as a system of different equations that have some constant parameters that does not change throughout the model. We need to take into consideration where our frame of reference is. In this model the frame of reference is where the wire for the drone is connected to the drone. In Simulink we know make a subsystem called plant where we have four inputs $[\omega^1, \omega^2, \omega^3, \omega^4]$ ass an output we need to have six different variables [x, y, z, roll, pitch, and yaw]. We also get out [$\ddot{X}$, $\ddot{Z}$, $\ddot{Z}$ and [$\dot{X}$, $\dot{Z}$, $\dot{Z}$. Here you can se the model of the Quadcopter plant Figure 5 and the input Figure 3.



Block Parameters: Plant system

Subsystem (mask)

Parameters

Mass 0.585

gravity -9.81

Air resistance 0.1

Air resistance 0.1

Air resistance 0.1

drag 0.4817

Block Parameters: Subsystem

Subsystem (mask)

Parameters

Motor resistance 0.6

Electrical inductians 7.6E-6

Inertia moments

Ixx 0.0021

Izz 0.0031

Iyy 0.0021

constants

Torque constnt 1/2300

Electrical constant 1.266E-8

rotor inertia

Rotor Inertia 8.0645E-6

dampening 3.15E-7

Figure 3: Values for the mask of Plant system

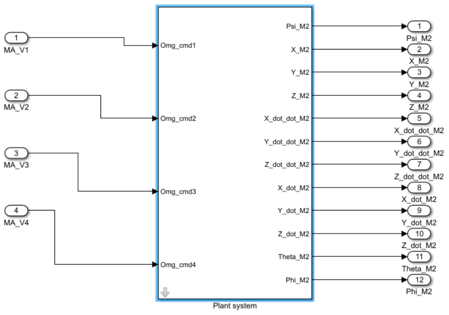


Figure 4: Values for the mask of Plant system

3.2.1 Motor Block

We start with the motor where we get in the omega command. We use the motor block to generate omega total. Omega total is found earlier in the formula (2.12). We also use the motor block to create the thrust inputs. We then need to calculate the thrust outputs we are using the X coordinate configuration with the formulas: (2.8), (2.9), (2.10), (2.11). You can see this how this is made by the Figure 6 in the plant system (Figure 5).

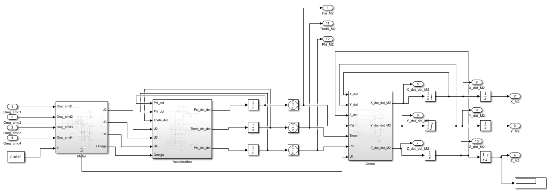


Figure 5: Plant system

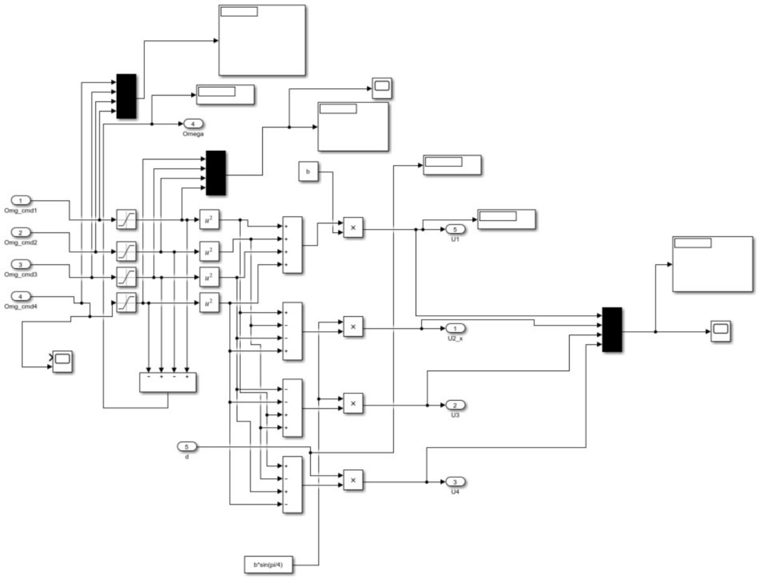


Figure 6: Motor block

3.2.2 Acceleration Block

We can now make the acceleration block that will connect to the motor block. The acceleration block is a subsystem that converts the motors thrust to angular rates. $[\omega^2, \omega^3, \omega^4]$ and $[\dot{\theta}, \dot{\psi}, \dot{\phi}]$ is taken from the motor block and out is $[\ddot{\theta}, \ddot{\psi}, \ddot{\phi}]$. ω^1 is not used however it is transferred to the next block. Inside the block we are going to use the formula (2.22), (2.23), (2.2.8). Ass mentioned there are tree values that we get out this can be seen in figure.

3.2.3 Linear Acceleration Block

With the angular rates that we got from the acceleration block we can now calculate into linear acceleration. In the plant model we integrate the value from the acceleration block down so that we go from double derivative to

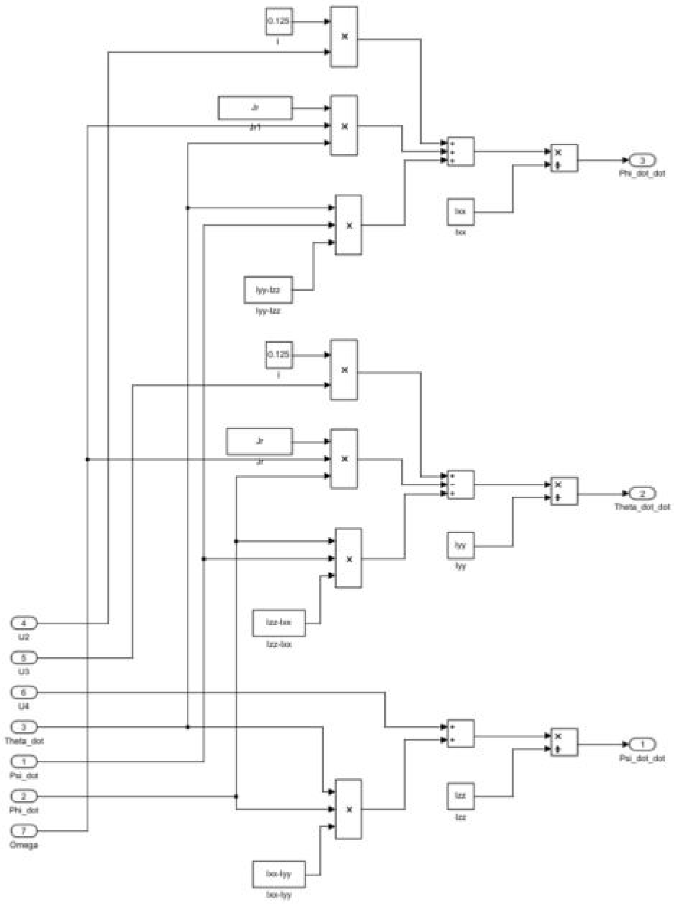


Figure 7: Acceleration Block

single derivate. we then integrate it again to remove all the derivation on the angels. This is then inserted as an input to the Linear acceleration block. For this block we will now need the U1 input. We will also need is $[\dot{X}, \dot{Y}, \dot{Z}]$ and pitch (θ), Yaw (Ψ) and Roll (Φ) as input. Output will be $[\ddot{X}, \ddot{Y}, \ddot{Z}]$.

$$\ddot{X} = \frac{(sin(\Psi)sin(\Phi)-cos(\Psi)sin(\Theta)cos(\Phi))U1-A_x\dot{X}}{m} \tag{3.1}$$

$$\ddot{Y} = \frac{(sin(\Psi)sin(\Phi)-cos(\Psi)sin(\Theta)cos(\Phi))U1-A_y\dot{Y}}{m} \tag{3.2}$$

$$\ddot{Z} = \frac{(mg-(cos(\Theta)cos(\Phi))U1-A_z\dot{Z}}{m} \tag{3.3}$$

The new block uses the formulas shown above [(3.1) (3.3) (3.3)] are used to create the block shown in Figure8. On the plant model we can see how the

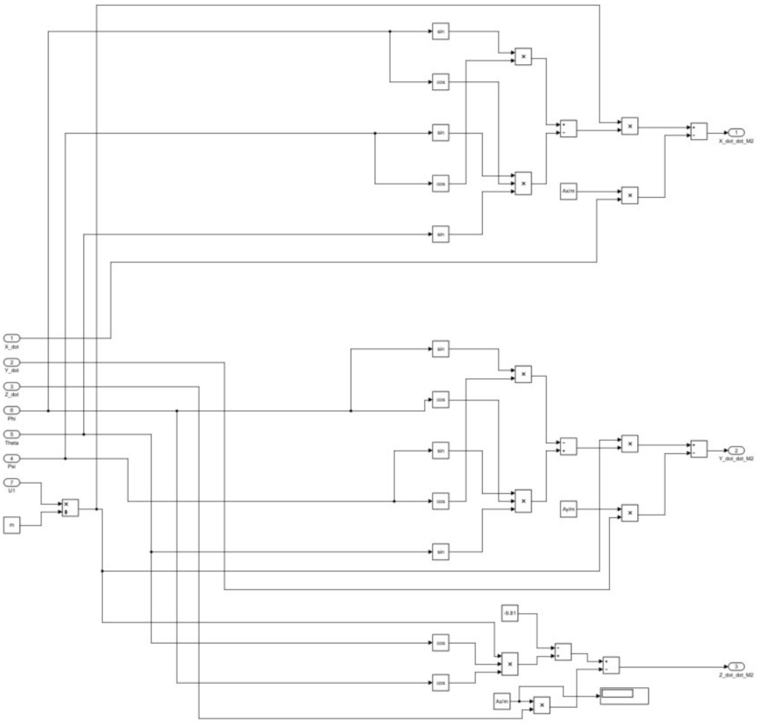


Figure 8: Inside Linear Acceleration Block

xyz velocities are feed into the linear acceleration model. This can be seen in figure9

3.2.4 PID Controller Block

PID or Proportional-Integral-Derivative that it called. It is one of the most common control algorithms that is used. PID are especially good because of there wide range of operating conditions and simplicity in use. PID consist of the parts: proportional, integral and derivative. These are changed in their variable ways to get the best response. The PID sums together the three components it also must-read sensor data to find out the error rate so it can be able to change it. The controller is maybe the most important part of the whole Model. It makes the control of the drone possible and would make the drone unstable. Proportional or P-controller gives a value that is proportional with the current error value. It compares a set value with the actual value. We then multiply this value with the proportional constant and we get the output. I-Controller integrates the error over a certain period of time, so it reaches zero. D-controller derives the constant

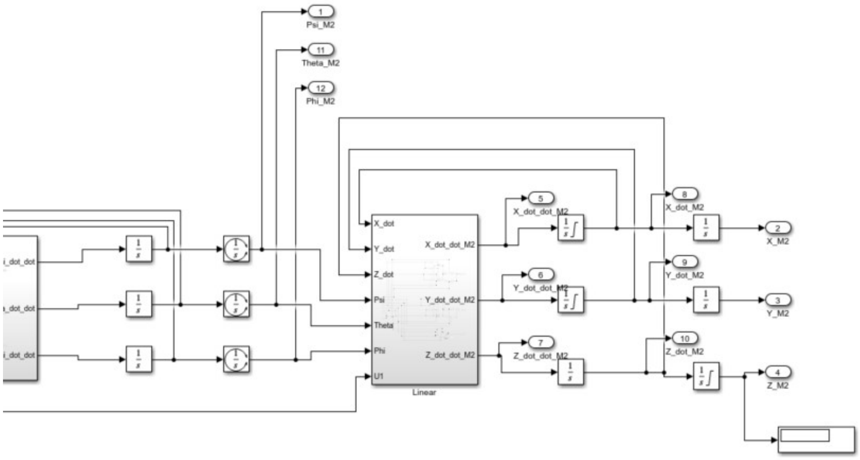


Figure 9: Linear Acceleration Block

of the error and its output depends on the rate of change with time.We can see an example of an PID block in figure 10.

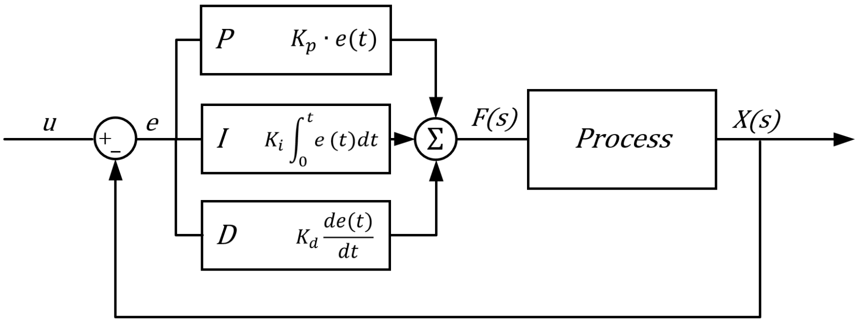


Figure 10: PID Controller

For our PID controller we need 4 PID controllers [Roll pitch yaw] and one for altitude. In Simulink we there exist already a PID block that can be used in our block. This means that we do not need to crate the PID block from scratch. The inputs values have two for each input, one desired input which will be the path that we want the drone to follow. The other is simulated which is the values that we get from the Plant Block. Since we want the PID block to output angular velocity. This is done by using the formulas (2.8), (2.9), (2.10), (2.11) since we are using the X frame. We can see the PID Controller Block below in figure 11.

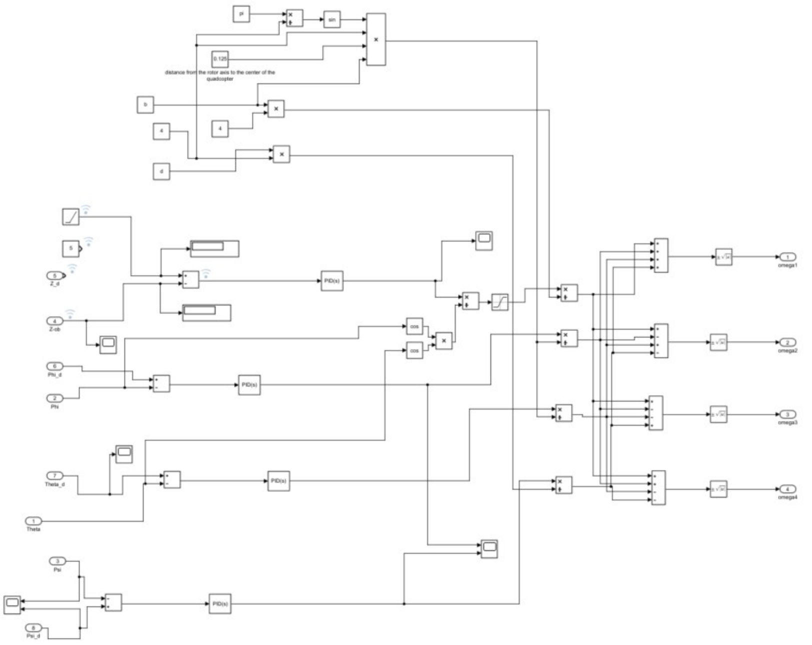


Figure 11: PID Controller Block

3.2.4.1 PID tuning We now have to tune the PID blocks. An tuner built inn version in matlab called Auto tuner was used. With this program it is as simple as just pressing the tune button. We then get up a graph that shows the tuned signal. From here we can change the different parameters of the signal. We can decide if we want a faster or slower response time or if we want to change the transient behavior of the system to aggressive or more robust. Out from this we can get the P, D, I values we also get a N value which is the Filter coefficient value. The filter coefficient determines the location of where the pole is on the filter on the derivative part of the controller. The location depends on the time domain parameters, and of the time domain is Discrete it depends on the filter method parameters.

3.3 Motor system.

For the Simulink system it must have a way to simulate the motors of the drone. The motors used on the QAV250 is 52205 KV2300 brushless DC motors. These motors are a standard type of motor that is produced in china and are not made from the producers that the name of the company that is printed on them are. For a motor model a voltage input is applied to the motors armature. The output of the system is the rotational speed of the shaft. The diagram for a dc electric motor is shown below in figure 12. There are some physical parameters that was needed to be found for the

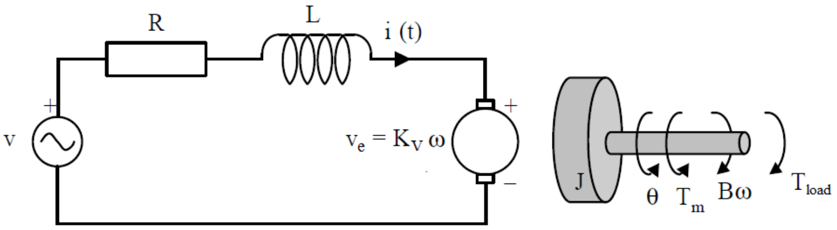


Figure 12: Motor model: http://www.sosw.poznan.pl/tfitzer/dcpid/motor.png

simulation to work. The electrical resistance of the motor was measured with help of a multimeter. The same was done to find the electrical inductance, but instead of a multimeter a purpose-built lab inductance meter was used. The electro magnetic force constant can be found easily by just reading what the KV value of the motor are. The Kv constant is the reciprocal of the back-emf constant [36]. This means that a 2300 KV motor that is used in this project will generate a 1 Volt back-emf if the motor rotates at 2300 RPM. Kemf = Ke and Ke = Kt and therefore Ke = (1/Kv). Motor viscous friction constant or dampening constant is found by $\frac{Kt \times Ke}{R}$. The moment of inertia of the rotor is found by $J = \frac{1}{2}mR^2$. With this we have all the values we need to simulate the motors[37]. The finished system is shown below in figure 13.

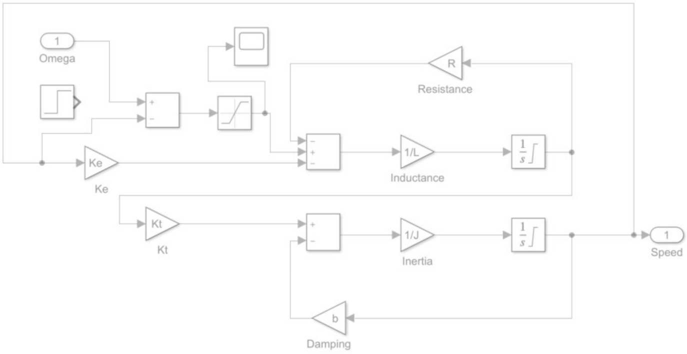


Figure 13: Simulink motor-block

3.3.1 IMU

The IMU "Inertial measuring Unit" is a collection of measurement units that work together to produce data on the location of the vehicle and to measure the impact of all external on the drone. A simple IMU normally has 3 measuring units. Those units are accelerometer that measures velocity and acceleration, Gyroscope which measures rotation and rotational rate and the magnetometer which measures magnetism and establish the cardinal direction of the drone. On the Simulink simulation it was possible to use a inbuilt IMU program which was part of the Sensor Fusion and Tracking Toolbox. There are three input values that are needed for the IMU to work. Linear acceleration, angular velocity and orientation are turned into the system. It was decided to not use the magnetometer because the output value was not needed in the further simulation. The output values where from the accelerometer and the Gyroscope. When simulation was done to see if the system worked error messages of incorrect matrix sizes for the input and the output were given by Simulink. There where no obvious solution of the problem. After not being able to solve the problem a search for a different IMU toolset had to be done. Lucky another alternative where available. This toolset from Aerospace Blockset called Three-axis Inertial Measurement Unit had no of the previous problems and worked great for the simulation. In figure 15 we can see both the different IMU alternatives. In figure 16 we can see how the IMU block are connected. The complete simulink system is shown in figure14 .

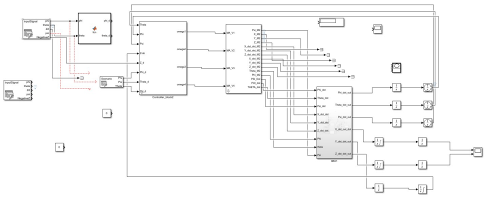


Figure 14: complete simulink system

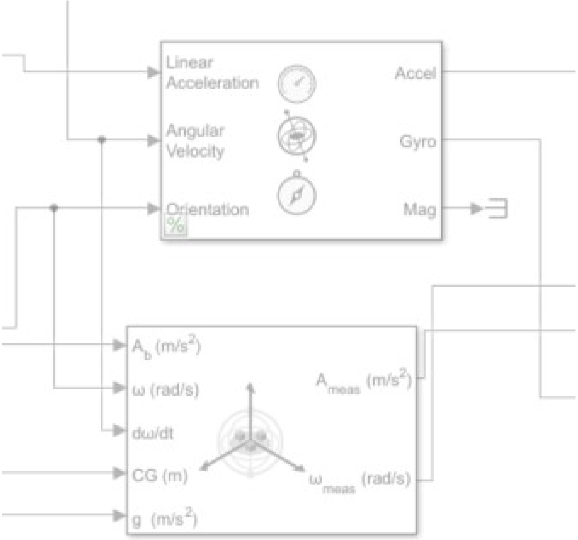


Figure 15: both IMU alternatives

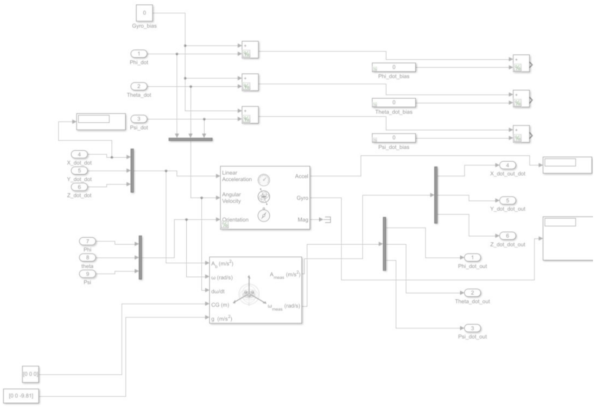


Figure 16: Simulink IMU

3.4 QAV250 Drone

In this thesis Maritime Robotics supplied a Pixhawk 4 mini QAV250 kit. This drone kit is produced by Holybro since this drone came as a kit it had to be assembled. What is nice with this drone is that most of the frame parts are made from carbon fiber. This makes it much lighter than the standard plastic framed drones. As with many product's produced in china there where some parts on this drone that that where sub par quality. There would be many difficult problems in the assembly process. The first problems encountered, started in the assembly manual available online[38]. It was soon clear that the manual and the some of the parts that was received where different, this would cause major problems down the road of the assembly.

3.4.1 QAV250 Drone assembly process

First part of the assembly process are the carbon base plate assembly. The base plate is bolted on to the propeller arms of the drone. Then another plate a bit bigger than the base plate is bolted on top of this. Next step is connecting the carbon fiber legs to the drone, these legs are just clip on legs that fit snugly on the arms of the drone. The completed Frame assembly can be seen in Figure 17.

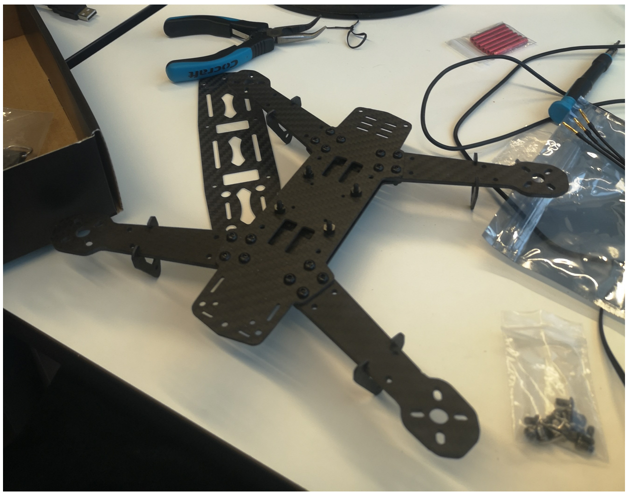


Figure 17: complete Drone Frame assembly

After this step it is time to install the motors. Here it was important to see which way the motors direction of spin is, this step was simple to do since the motors where color coded. The main controller for the drone is the Pixhawk 4 mini that runs Ardupilot. From the motors we must connect the ESC signal lines to Main Out on the Pixhawk. It was important to connect the wires the correct way to the PH4, in the manual the colors where different from the wires that was received. we manage to place the wires the correct way after a bit of troubleshooting. We then must place the power distribution board (PDB) to the frame. The power cables is then connected to the motors. The next step was where one of the largest happened. From the pictures on the manual and online It was possible to see that a component had been added to the PDB without the producers updating their design of the drone. This problem was that they had added a capacitor to the power board going from the positive input terminal to the negative terminal on the battery input point. This capacitor is most likely used to filter noise from the battery. It is fitted to suppress unwanted ranges of signals. These signals might be high frequency signals from motor vibrations and from voltage instability. The problem with this capacitor was that it was place on top of the wires going to the PDB with this, it meant that the total height of the PDB board was now too high to fit the plate that was going on top of the PDB. There where two options to fix this problem. The first obvious one would be to find longer standoff bolts that could be used to lift the plate that would be used on top of the PDB. This however was found out to be difficult to do since it would require to by new standoff screws, and it would take more than two weeks to arrive. It was also found that this method would not function since the top plate of the drone would not be high enough. there was a possibility to have raised it with higher standoff, but this would also have taken time to order the extra standoffs. The other option is to de-solder the battery wires and then move the capacitor down and in between the wires. This sounded easy ting to do. However, the wires that had to be soldered where of a larger diameter than the rest of the wires used in for the drone. since these wires are used for the batteries, they must be larger. The problem that arose with the larger wires where that when de-soldering they removed to much heat. This meant that there was not enough heat to melt the solder. The soldering iron that was used at the beginning had not enough power on it to melt the solder. The chief professor for the lab gave me the most powerful soldering iron they had and still it did not melt. They manage to procure another soldering iron, and this finally was able to melt the solder on the wires. The solution to the problem was to resolder the wires around the capacitor in such a way that the capacitor could be lowered. This meant that plate that went over the PDB now just fit, and next building step could be done. after this the use of double-sided foam tape to fasten the Pixhawk to the plate was applied. Now that this is done it is time to connect the power from the PDB to the

Pixhawk Power port. Figure 18 shows all the connections on the Pixhawk 4 mini.

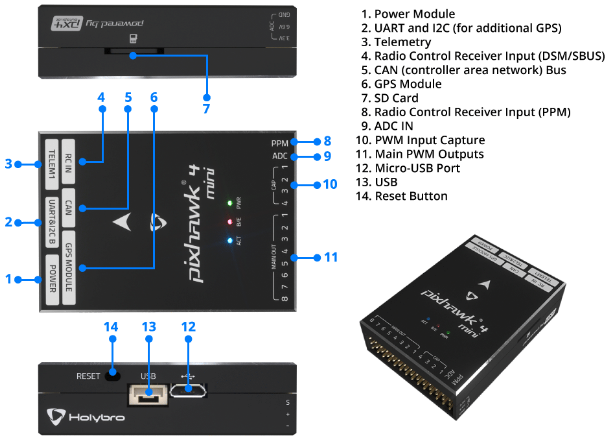


Figure 18: Interfaces on the Pixhawk 4 Mini [2]

We then have to connect the FPV transmitter to the base plate, this is fasten with a zip tie. It was also found out that it was not good enough with just fastening with a zip tie, therefor double sided tape and a Velcro strap was used. Now the wire coming out from the FPV transmitter has to be connected to the FPW power bord. The FPW power board is according to the manual be place on top of the mini telemetry radio. This however did not work out since the telemetry radio that was received was different from what was shown in the manual. Instead, we just placed the FPW power board straight on the base plate with help of double sided tape. With the FPW power bord in place we connect the power cable to the UARTI2C B port of the Pixhawk 4 mini. The top plate of the drone is fixed in place to the base plate with six standoffs butt before this we must attach the GPS antenna to the top plate with double sided plate. the wire from the GPS is threaded thru a hole in the top plate and fastened to the Pixhawk 4 mini-GPS module input port. With the GPS attached we only need to attach the two battery holding straps to the top plate before we can bolt them to the base plate. now that the top plate where installed we could install the camera mount and mount the camera to the drone. The camera will not be used in this thesis so it was not really necessary to fit it however it was assumed that not fitting the camera would make it harder on the software side later down the road, so it was installed it anyway. The camera cable is connected to the FPW power board. As mentioned earlier we had received a different telemetry radio than the one that they have shown in the instruction video. This made it so that it would not fit the inside of the drone body when assembling sins it was too wide to fit between the standoffs that fixes the top and bottom plate of the drone. Since the placement of components on the drone is important to have as centered as possible so not to change the center of mass to much. There where no way to place the telemetry module inside of the space of the drone. It was therefor decided to move the telemetry model outside the on the bottom of the low plate on the drone under the FPV transmitter. This was fastened with the same zip tie and Velcro strap used to fasten the FPV transmitter. To make sure it would not fall of the telemetry receiver was also fasten with double sided tape. The telemetry receiver is plugged into the TELEM1 port of the Pixhawk 4 mini. After doing some cable management and connecting the cables going from the power module to the motors. The last installation step is to connect the propellers to the correct direction. The complete drone can be seen in Figure 19

Figure 19: complete Drone assembly

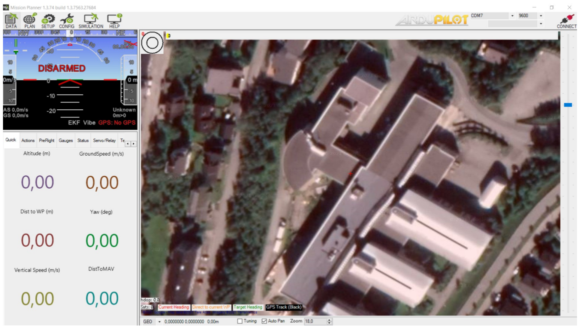
3.4.2 Drone operating program

A drone operating program is the program and user interface that makes it possible to talk from the computer to the drone. It is basically a ground control station for any drone that you want to operate. It makes it more easier to understand every element of the control system of the drone without the need to use code language. With the control program you can make and design the path that the drone are going to fly. Here the high and speed of the drone can be configured to get the perfect flight. The drone program is also responsible to Tune the drone so that all the sensors and GPS is working correctly. This is very important because a few degrees of point the drone calculations will be wrong. This could make a difference of several meters. The program that was recommended by Holybro was QGroundControl this program would be intuitive to use and could do all of the things mentioned above. Installing the program and connecting the drone to it was no problem. With the calibration of the drone, you are suppose to calibrate the RC controller to pitch yaw and roll. This would not be possible since on the drone there were no RC input receiver included. My plan had been to control the drone via the telemetry radio. Therefore, there was two Holybro 433MHz telemetry radios. As mentioned earlier the telemetry radio that was attached to the drone was not supposed to be the Holybro 433MHz telemetry radios as show in figure 20 but a mini version of it.

Figure 20: Holybro 433MHz telemetry unit

The plan was to control the drone via a joystick controller. This controller would connect to the computer via USB and interact with the control program. The QGroundControl unfortunately had no support for a joystick and transmitting the data via telemetry instead of RC transmitter. Because of this it was forced to use a different control program called Mission Planer. Mission planer is a quite like the QGroundControl hoverer it supports the use of joystick. With a joystick a person can control the drone from my PC using the telemetry radio. To sett up the drone to use mission planer you just must connect the drone to any computer. Once connected you will have to install firmware to your drone. This firmware is called Ardupilot and is an open-source piloting program for any drone. At the installation screen you will need to click on the quadcopter X configuration drone icon since the QAV250 is this type of drone. When clicked you will be presented with options on what type of drone and what type of hardware it uses. The Pixhawk 4 mini was selected and it can start the installation. the program will figure out by itself the installation procedure and configure the drone. It is important that during this step that the drone is not connected on the connect button on the Mission Planer menu shown in figure 21.

When instructed to you can connect the drone. Now when this is done



COM7 9600

AS 0,0m/s
GS 0,0m/s

Altitude (m) 0,00
GroundSpeed (m/s) 0,00
Dist to WP (m) 0,00
Yaw (deg) 0,00
Vertical Speed (m/s) 0,00
DistToMAV 0,00

hdop: 0,0
Sats: 0

0,0000000 0,0000000 0,00m
☐ Tuning ☑ Auto Pan Zoom 18,0

Figure 21: Mission Planer user interface

we can connect the joystick to the computer. With the joystick connected, we can go to the setup page of Mission planer and under optional hardware go to the joystick menu. On the joystick menu we must assign all the keys and calibrate the joystick. Roll pitch yaw and trust, are some of the keys that must be assigned. Also, keys to arm the drone and to emergency land the drone must be activated. When this is done it should be able to control the drone via a joystick when it is connected to the computer. Since we do not want to be connected continuously with the drone with a USB cable, the telemetry radio had to be connection had to be made. This quite simple step would be a major problem for me and restriction of testing of my drone. When connecting the telemetry transmitter to the computer according to the manual of Mission planner the user should automatically be given a COM port number that the user then must put into the Connect menu of the mission planer. With the telemetry unit it could not find any com port for my telemetry unit. One of the methods that the internet suggested to solve the problems was to go to device manager on windows and re install the driver of the telemetry units. This did not work ether, and so it was tried to contact the producers of the telemetry unit. Holybro which was the producers did not answer my email and its web page did not help in solving my problems. My next step in problem solving was to test the telemetry unit on the laptops of fellow students in the Aerospace class. There it was found that they have the same problem with not getting any COM port. With this test it meant that there is noting wrong with my computer but something to do with the operating system (Windows). To test this out it was decided to try to plug in the telemetry device in a computer that we had in a satellite tracking control room at the school since we have turned off auto update at the computers there. They use an older version of windows 10 since we do not want to have the risk of getting a unstable version. When the transmitter was tried on this computer, it was able to find the COM port for the transmitter. This is good news since it meant that the transistor was not damaged and could be used to transmit telemetry information to the drone.

The bad thing was that the transmitter had to use this stationary computer to fly my drone. The original plan was to use a personal laptop which could run on batteries to test the drone on a parking lot. With the stationary computer the drone was limited in the range of the transmitter and the short half a meter USB cable that connected the PC to the Transmitter. Because of this there could only be tests of the drone out from the window of the control room. With the transmitter working the work could start on test flights. At the beginning it showed another error where the space craft would not arm the craft since the program could not detect the battery on the drone. This was solved by going into the program and overriding the error message and setting the error value to 0 volts so that the drone would think it was fine with 0 volts. With this done the drone was ready to fly.

3.4.3 Tether Control system

The wish from maritime robotics where conceptualize an idea of how a drone could be controlled and respond from a tethered drone. To visualize it you could imagine a drone fastened with a tether. This tether is then connected to some sort of vehicle that can move separately from that path of the drone. In this thesis this vehicle will only move on a 2D frame of reference. This makes it easier to conceptualize, even though there are ways a 3D frame reference might work that will be explained later. There are two main scenarios that the drone control system can be affected. This is decided if the tether that the drone is using are able to send information or not. Technically it can be also explained of the ground vehicle are able to pass information back to the drone or not. Let's start with the assumption first that the drone is not capable to transmit any information to the drone. With this all the information needed to control the drone has to be already fasten to the drone. Sensors that can pick up the date in the change of position has to also be fastened to the drone.

3.4.3.1 Difficulties in controlling the drone. The basic problem of a drone connected to a moving object are that the drone has to follow the vehicle when the it moves underneath. This movement has to be such lenient that it allows the drone to move around without the drone thinking that the vehicle is moving. A factor that makes this even more difficult is the length of the tether. The longer the tether the smaller the angle differences between the drone and the vehicle. If there is a possibility to measure the angle of the tether compared to the vertical line going along the gravity vector. With this angle it is possible to use inverse sinus angle times the hypotenuse. Whit this the opposite is found this is the distance from the gravity vector to the vehicle. A simple if statement code can be used so that when the distance decrease to around zero the drones know that it is above the drone. Since the drone is in a 3D space we have to find this angle that is going to be used on the sinus angle. The use of a joystick can be use to find this angle, since joysticks used a 2D coordinate plane XY it is possible to calculate the angle. the joysticks used in this these are Arduino compatible joysticks from a unknowing Chinese producer shown in figure 22.

Figure 22: Joystick and Arduino board

First angle that has to be found is the angle direction of vehicle. This can be found by using atan2(y,x). Atan2 is a commonly used in many programing languages and is a function described in (2.25). Now with this angle we know the direction of the vehicle. The angle down to the vehicle now has to be calculated. This is done first by finding the length from origin to the (X,Y) value that the joystick is outputting, this a value is found using Pythagoras theorem and is given variable a. The z value cannot exceed the max value of x or y value since it has to be within an imaginary sphere drawn from the max x and max y values with its center at origo where $X = 0$ and $y = 0$. Using spherical coordinates p can be calculating using Pythagoras of Ymax and Xmax. Then z can be found by using Pythagoras of p subtracted with a. with the z value found it is possible to calculate the angle by using the inverse tangent of the a and z value to get the correct angle(2.29). Then using cosines with the tether length and the angle just found the height of the drone is calculated(2.30). With the height information it is now possible to see where the drone is according to the vehicle. If the height is the same length as the tether it means that the drone has to be vertically above the vehicle. However if the height is less than the tether length the drone has to be in a position away from the vehicle. With the information on the orientation angle that was calculated earlier it is possible to know exactly where the drone is positioned compared to where it is suppose to be at the zero point. This is a good valuable way to control the way the drone, it has however a weakness that it does require the drone to be leveled at all times. If the drone is not horizontally level the math will become off and the height will become wrong. Even if the drone is not horizontally level the orientation angle will be correct since it does not care for how much the joystick is angled. There are two ways of solving these problems of the drone angle of tilt one would be to have a Gyro sensor that could measure the angular rate of the vehicle. With the angular rate it is possible to subtract this rate from the calculated angular rate from the joystick and find the thru height of the drone. The gyro sensor would have needed to be outside the drone since the internal gyro of the Pixhawk has no easy way to send this information out from its unit. There is also the possibility to use electronic weight sensors or load cell sensors. in the thesis a cheep digital baggage scale was used to obtain one of these sensors shown in figure 23.

These sensors measure the change in tension in a piece of metal. A weight sensor can be used to regulate the power output of the motors on the drone system such that it does not waste energy fighting the tether. When the tension is almost equal to zero the drone is at an equilibrium and is not pulling against the tether. Into these calculations the tether weight must be kept into consideration and subtracted from the value given from the tension sensor. With a tension sensor it is also possible to see if the drone has an angle or not. this is done by first observing the directional angle of the joystick. One measurement is needed when the drone is just over the origo point at max tether length. If the drone with the tether shifts now out, let us say 30 degrees but is still vertical to the tether. The pull on the tension sensor will increase because of the drone now has to increase power to both compensate for the gravity and the changing center of lift. With three tensions sensor on the drone, it is possible to triangulate the angle from the drone down to the tether. Another major benefit of the tension sensor is that it can figure out if the tether is loos or if it is tensioned. This means that the control system can know if the drone is flying at its maximum height or if its flying lower with a slack tether. The control system can then command the drone to gain height. The reverse is possible if the tension sensor exceeds a certain threshold value meaning the drone is wasting energy trying to hover. The implication of this makes it possible to design a landing and takeoff system that is using a spool fastened

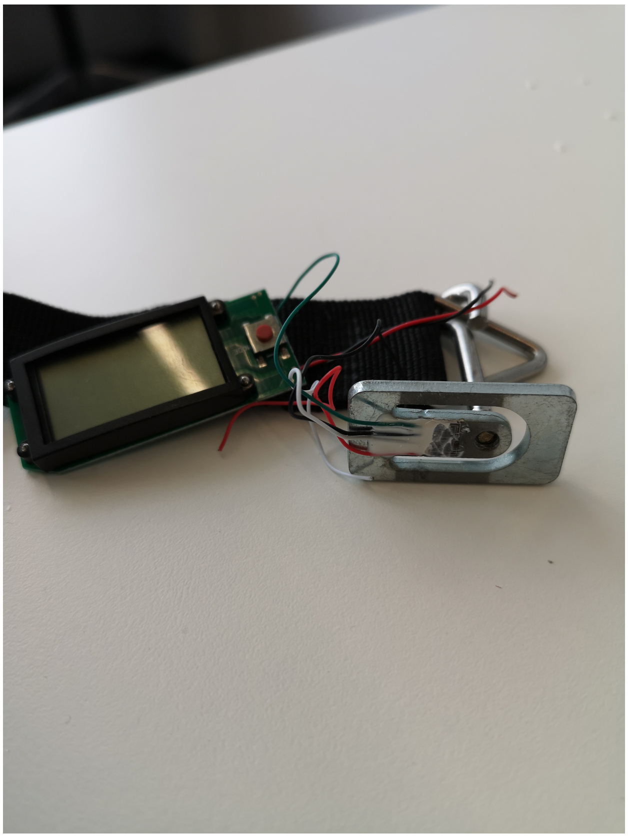


Figure 23: Digital load cell sensor

with a tension sensor. When spooling out the tether the tension is lessened therefore command is given to increase thrust to the drone. When spooling in the tether the tension on the sensor will increase therefor the command to lower thrust will be given. As mentioned earlier in the report there is another way to control the drone. This is when the drone is in constant communication with the vehicle platform bellow. This can be ether through radio signals or through a wire running along the tether. Here the joystick is fastened to both the vehicle and to the drone. With two joysticks it is possible to calculate the position and orientation of the drone. The job of the joystick on the vehicle is that to find the angle rate from the vertical origo point. This angle is compared to the angle of the joystick of the drone. When the drone joystick angle is subtracted from the vehicle angle. When both angles are zero it means that the drone is at the $X = 0$ and $Y = 0$. If the subtracted value is zero that means that the drone's angle is horizontally or at 0 degrees. With a positive angle the drone is moving towards the origo point and when its negative it means the vehicle has a angle going away from origo. To control all of this data and to do all the calculations in this thesis a Arduino nano every board was used. This was the smallest board that is available from the producers of Arduino and uses 5 volts input as operating voltage. For the joystick a Thumb Joystick V1.1 was used this is the good joystick that gives out a 1023-bit value for the Y and X. zero X and Y value is at 512 since the joystick cannot transfer negative values, this is shown in Figure 24. The major problem that made it impossible to test the whole system as one integrated experiment with the drone was that the computer language that Ardupilot would understand the RC inputs where impossible to find online. Since we needed to give the roll and pitch information to the drone from the Arduino for it to follow the vehicle. We know the bus protocol that the Pixhawk has to have to receive the RC signal, Pixhawk uses the SBUS protocol which supports 16 channels using only one wire.

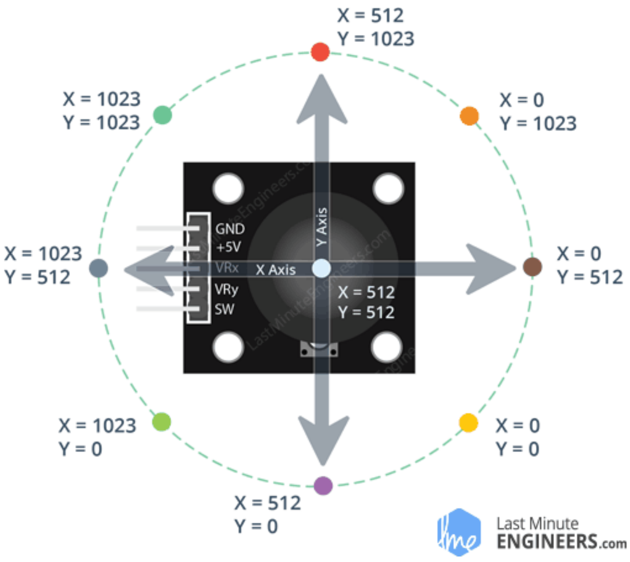


Figure 24: Joystick Module Movement Analog Values [3]

4 Simulation/Experimental results

There were different simulations that was planned for the project. A fair few of these simulations where not possible to do because of time constraints related to COVID and problems that arose with the simulation and the communication code between the systems. With the PID calibration it had to test different values for the calibration of the directions. For each of the output signals that was calculated they were tested and compared to the desired signal. In figure 25 you can see this signal for Psi with the desired and the wanted signal. When varying control signal is introduced it is possible to see that the PSI signal still works. Now the difficult part is tuning the z axis because for some reason it believes that the value goes to extreme heights. This problem is based on the difficulty in tunning a PID controller to track very quick in its tuning in the experiment with the signal input shown in figure 26 you can see that figure 28 showing high response time is not accurate enough. This is especially clear when we see that the z value is going slowly down to the desired Z value (Z=10). Compared to the x and Y graph of figure 27 showing a more realistic view where there are no extreme length definition. These graphs are not connected to the IMU and motor block for simplicity.

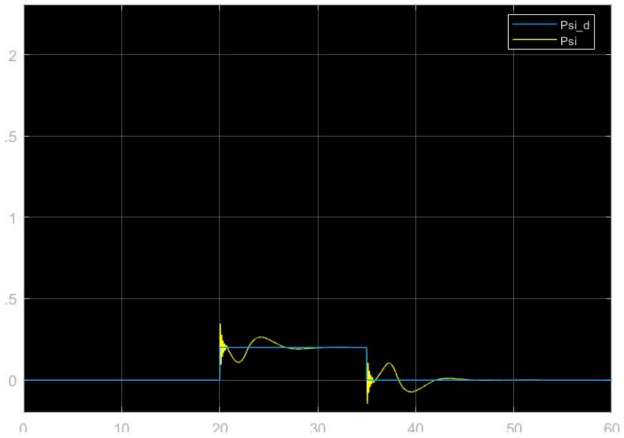


Figure 25: PSI signal with control signal

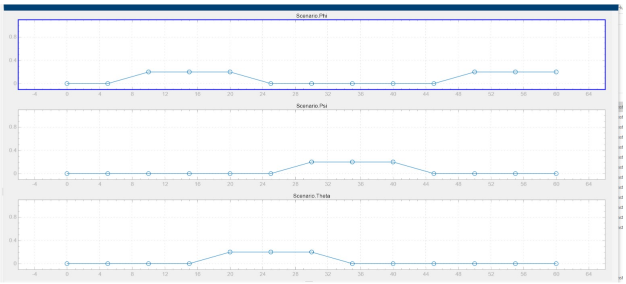


Figure 26: PSI input signal for Psi, Phi, Theta

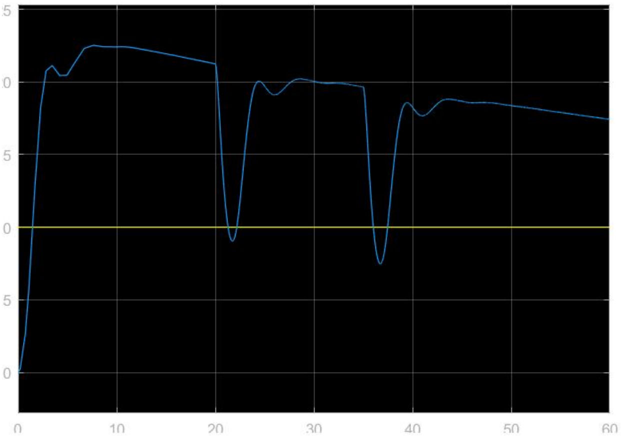


Figure 27: Z axis with control signal

Figure 29 gives a overview from above of the path of the drone compared to the XY direction. it shows the drone reacting quickly to the input signals. With trying to include the IMU. Error message with “Derivative of state ’1’ in block ’modelT/Subsystem2/Plant system/Integrator1’ at time 0.0875 is not finite.” Kept on happening trying to fixed it by changing the step size did not work. Therefor the testing with the IMU could not acquire. The idea of controlling the drone with Joysticks and Arduino board functioned relative good it was difficult to test it without a way of communicating from the Arduino to the board. The control theory works on the mathematical calculation. However, it was not enough time to test this on the Arduino because of the difficulties of the Simulink model and the error that happened with the Com port on the telemetry units took to much time. Two tests where done to see if the Drone would fly. One was a hover test at 5 meters just to see if the drone worked. This test worked fine without any incident. The next test that where done was a path flight where it was to follow an assigned path drawn from the Mission planner. Because of the of the range of the transmitter it did not go correctly, and the drone returned to the home position. It was believe that it was a range issue, and it was later confirmed

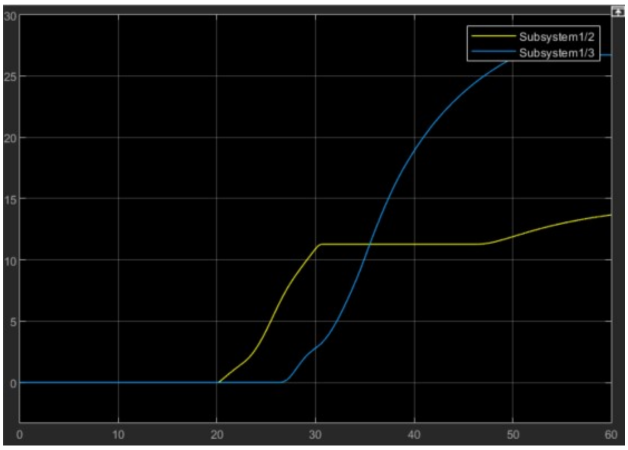


Figure 28: X and Y values compared over time

on the third test when I decreased the path distance from the transmitter. On the third flight the drone was able to follow the path (Figure 30) without interruption, proving that the drone was flight capable.

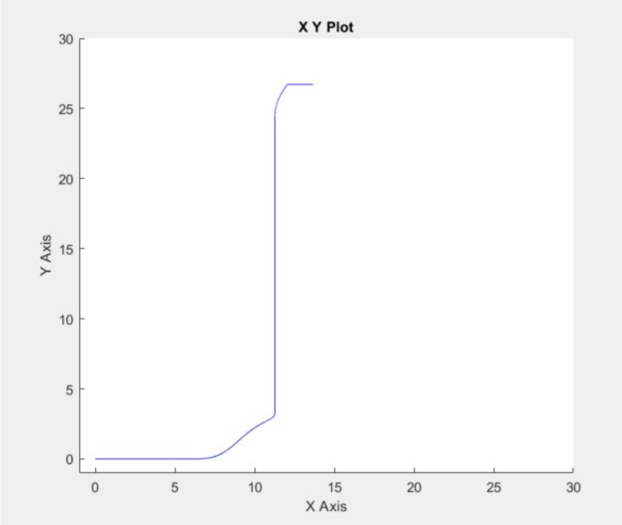
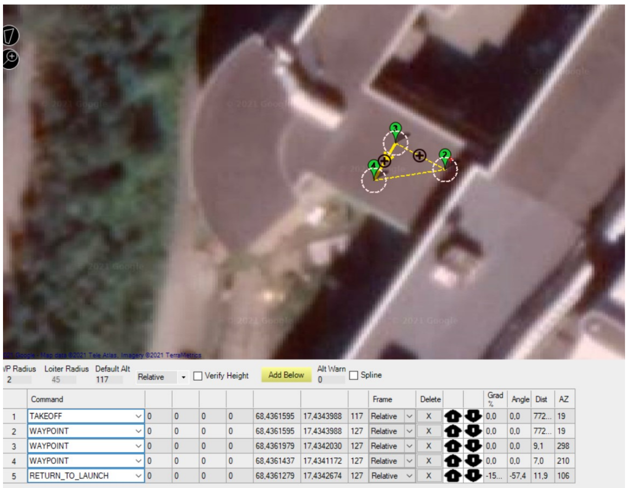


Figure 29: X and Y values plottet against each other.



WP Radius 2
Loiter Radius 45
Default Alt 117
Relative
☐ Verify Height
Alt Warn 0
☐ Spline

	Command								Frame	Delete			Grad %	Angle	Dist	AZ
1	TAKEOFF	0	0	0	0	68,4361595	17,4343988	117	Relative	X	⬆	⬇	0,0	0,0	772...	19
2	WAYPOINT	0	0	0	0	68,4361595	17,4343988	127	Relative	X	⬆	⬇	0,0	0,0	772...	19
3	WAYPOINT	0	0	0	0	68,4361979	17,4342030	127	Relative	X	⬆	⬇	0,0	0,0	9,1	298
4	WAYPOINT	0	0	0	0	68,4361437	17,4341172	127	Relative	X	⬆	⬇	0,0	0,0	7,0	210
5	RETURN_TO_LAUNCH	0	0	0	0	68,4361279	17,4342674	127	Relative	X	⬆	⬇	-15...	-57,4	11,9	106

Figure 30: Drone path plotted in Mission Planer, 3 Flight.

5 Discussion

There were many problems that was found while working on the thesis. Making a Simulink model and a control system for a tether, while building and operating a drone at the same time is a difficult and time-consuming task to do all at the same time. While it has been hard, parts of the assignment have been completed with newfound respect while others have not been possible to complete. The first major problem that arose when working on the project where time delays because of Covid, this was an external cause because delivery of equipment from abroad has been severely hampered with the lower demand of plane travel therefor delaying part delivery for the QAV250 drone and the control system. The simulation of the drone in Simulink seamed liked a easy to do procedure. The main difficulty where that every subblock which was added into the simulation, the chance for something to go wrong increased. If there where any mistakes at the beginning the problem would propagate around the whole system. With the way that the model was first made without any testing, meant that there where so many mistakes made that it would be almost impossible to find all the solutions. This resulted in that the model had to be built up again. This time, step by step building with testing every time had to be done. This took time and with little help from the internet it took a very long time to finish each section. Since there were no code available online which we could compare the result with there were no possibility to quality control the code and making sure that what was done was correct. Finding a way to calculate the values for the drone control system took also to much time to research. With most equations such as induced drag coefficient, many unknown variables that also had to be figured out. In many of the research papers that where found, they simply used an arbitrary number that only worked for there drone without any reference to how they calculated that number. Like the efficiency factor which all the papers just placed as 0.7. This efficiency factor turned out to be the Oswald Factor that from after a long research you needed to know the Lifting-line theory. Which was newer taught in any of the master or bachelor classes and involved complex math that beyond the math knowledge i knew. All these values that needed to be calculated meant that there had to be a concession of ether doing them and waste valuable time or just use standard values that few other thesis papers had been using. This would create values which became uncorrected in my simulation. Another discovery where that there where remarkable few simulations of drones online. Most of the open source simulations are ether extremely complicated or using advanced code. Or they were closed source and not available to be analysed for these thesis. There where some that have built drone controllers with the help of Simulink. They have all followed the same basic control system with a General Control Strategy, Equations of Motion, PID Control, Control Loops, Position and attitude/altitude Control, motor block and IMU. All of them had missing parts that would have made them not work for me. These papers also did not include there code so comparing mine to them to find mistakes did not work. Even though there where a lot of theory online on drone control. None of them explained the math of how they found every single value. Using Simulink also proved difficult to do also with trying to make everything organized. It would have been simpler and more organized to code it in MATLAB or phyton. Specially a lot of the calculation could have been done in matrix form and therefore shortened the number of calculations needed. The loops made it difficult to see where the values came from and many mistakes were made by connecting the wrong values. With the drone it would have worked much better with a simpler software or a more open-source hardware than the Pixhawk 4 mini. Without any way of connecting to a unassigned pin, it forced the project to try to use the RC In Port. There seamed no easy way to fix this problem without buying a RC transmitter and receiver and finding the ardupilot RC protocols. This would have been impractical since both would have needed to be placed on the drone connected to the external Arduino. Instead of the Arduino just being connected to the Pixhawk with a wire. if Arduino Drone had been used instead of a Pixhawk it might have been easier to change the values of the drone and making it turn of its GPS system so that it could have been tested indoors. The Pixhawk 4 will not let you arm the drone if you do not have a GPS fix therefor it cannot fly indoors. With Indoor testing there could have been the possibility to use the drone lab to compare the position that the drone thought it was in and the position it actually had. The PID was also a major milestone to come over. There seems that the PID controllers in MATLAB are not good enough to regulate the system. The problem with a PID controller is that K_p K_i and K_d has to be correct for the regulator to work. And with a strange Quirke with Simulink, a fourth variable N had to also be guessed. The reason why Simulink were used instead of other programmable languages is the visual way you can see the code and manipulate it visually. It is also used because of the much less experience with other coding languages that have been tough in the bachelor and master course on UIT. Even though many of the experiments were hampered, the relatively simple solution of how a tether drone could be controlled from just two joysticks in ether end of the tether with a load cell sensor surprised the most. The benefit of this system may have implications on future work where the drone does not need a GPS signal. With using its own local coordinate system with the USV as its origin point it can be used in a GPS denied area. The drone in theory could forgo almost all its sensors that normal drones use today. The USV would need to be able to know where it is to operate, however the drone could still operate even if the USV does not know here it is located.

6 Conclusion

The thesis has summarized a guiding way to how tether drone could operate and work. It as also gone into the challenges and opportunities in using a tether drone instead of a normal drone. The thesis has shown some the state-of-the-art methods of usage and control of a tethered drone. With the build and operating of the QAV250 Pixhawk 4 mini drone it has shown that the theory of how a tether drone could work. However, in practicality the complexity which happens when you try to use a commercial drone for an experiment that does not comply to the standard operating procedure such type of drone makes it difficult to complete all the desired goals. Even though many of the experiments were hampered, the relatively simple solution of how a tether drone could be controlled was a revelation that was not expected when started working on this thesis. The solution that with just two joysticks in ether end of the tether and with the help of a digital weight sensor. It became possible to know and control the position of the drone. The results have shown that most of the sensors can be placed on the USV. The thesis includes a went into the dept of making a drone simulation with the help of Simulink. The results of this simulation where mixed, making a quadcopter model and plant is a complex task that that makes it easy to make mistakes. Even though simulation made in this thesis made only a partial success in simulating the system. A strong core work has been done and with more time could have been completed. Even though the thesis does not completely solve all the tasks that was it was sett out to do. It gives enough solutions and ways forward that Maritime Robotics can use this data and inspiration to work forward on a working prototype.

6.1 Future work

In the future, it would be interesting to see the simulation that was made in Simulink done in python or C++. The reasoning behind this, is that it would clean out the code and make it easier to simulate a drone. This would hopefully make the possibilities of making mistakes less likely. Changing from a PID controller to something more advanced and simpler to tune would also benefit simplify the simulation. Lastly battery and electrical system management of the simulation is something that should be looked into in future work. Instead of using a Pixhawk for the drone, a homemade solution of using an open-source hardware and program like Arduino could work better than a commercial system. Because the system only simulated the USV in a 2D space instead of a 3D space it would be beneficial to find a control system with using a 3D USV system. A 3D USV is a very realistic system since Maritime Robotics want to use a boat as the USV. This means waves that exist in the ocean influences the 3D space. Another future work that would be interesting in looking closer at is powering the drone via the tethered system. Getting rid of the battery system has many beneficial properties that were mentioned in the literature study. As mentioned in the discussion part off the thesis, comparing the position of the drone control system in a lab motion capture system would significantly show the accuracy of the control theory of the tether drone system. With a motion capture other control algorithm can be compared to the one made in this thesis.

References

[1] A. Alassar, I. Abuhadrous, and H. Elaydi, *Modeling and control of 5 DOF robot arm using supervisory control.* PhD thesis, 03 2010.

[2] Dronecode, a Linux Foundation", "Interfaces," 02 2021.

[3] last minute engineers", "Analog values from joystick."

[4] I. Buchmann, "BU-205: Types of Lithium-ion," 2019.

[5] U. Manual, "Mavic 2 Pro / Zoom," no. 2.2, pp. 1–68, 2020.

[6] P. Kurz, "Tethered Drone Elevates Fox Sports Daytona 500 Coverage," 2018.

[7] D. E. Price, "Unmanned aircraft systems for emergency management: A guide for policy makers and practitioners," *Homeland Security Affairs*, no. March, p. 119, 2016.

[8] E. Reynolds, "Flying cellphone towers: Could drones bring internet coverage to remote areas?," feb 2020.

[9] A. Pregler, "When COWs Fly: AT&T Sending LTE Signals from Drones," 2017.

[10] I. LEE, "AT&T Uses "Flying Cow" Drone to Restore Service to Areas Impacted by Hurricane Michael," 2018.

[11] L. SHIVAMBER, "The Power of a Tethered Drone: What You Need to Know," 2019.

[12] M. McNabb, "French Defense Agency Invests in Elistair Tethered Drones," dec 2017.

[13] D. H. S. U. Pia, S. A. R. Ricciardi, and P. S. Kaplan, "United States Secret Service Counter Surveillance Division Unmanned Aircraft Systems Program Test," *Department of Homeland Security*, no. 202, 2017.

[14] M.-C. Fayot, "Elistair tethered drones selected for broadcasting & security at the Super Bowl," 2019.

[15] W. a. I. Pao, "Applications of tethered drones in oil & gas, law enforcement," 2017.

[16] M. BERNTH, "Autonom drone ser inn i vindmøllevinger med ultralyd," 2019.

[17] MIMREE, "MIMREE PROJECT OBJECTIVES," 2020.

[18] Elistair, “Elistair,” 2020.

[19] IEEE SA, *1901-2010 - IEEE standard for broadband over power line networks: medium access control and physical layer specifications.* 2010.

[20] Skysapience, “Skysapience,” 2020.

[21] F. Dynamics, “LIFELINE,” p. 2, 2020.

[22] USAS, “Unmanned systems and solutions,,” 2020.

[23] M. M. Nicotra, R. Naldi, and E. Garone, “Taut cable control of a tethered UAV,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2014.

[24] M. Tognon, C. Gabellieri, L. Pallottino, and A. Franchi, “Aerial Co-Manipulation with Cables: The Role of Internal Force for Equilibria, Stability, and Passivity,” *IEEE Robotics and Automation Letters*, 2018.

[25] L. Fagiano, “Systems of Tethered Multicopters: Modeling and Control Design,” *IFAC-PapersOnLine*, 2017.

[26] D. o. E. S. E. Quentin Morales-Perryman (Hampton University, Electrical Engineering), Dr. Daniel D. Lee, “Tethering System for Unmanned Aerial Vehicles,” tech. rep., (Hampton University, 2018.

[27] T. Lee, “Geometric controls for a tethered quadrotor UAV,” in *Proceedings of the IEEE Conference on Decision and Control*, 2015.

[28] J. Peláez(Universidad Politécnica de Madrid), “Tether Fundamentals,” tech. rep., Universidad Politécnica de Madrid, Madrid, 2004.

[29] R. M. P. P. d. M. I. S. T. Martins, “Flight control and position estimation for a tethered micro aerial robot,” tech. rep., Instituto Superior T´ecnico, Lisboa, 2019.

[30] G. Cai, B. M. Chen, and T. H. Lee, *Coordinate Systems and Transformations*, pp. 23–34. London: Springer London, 2011.

[31] M. L. U. o. A. S. Usman, *Quadcopter Modelling and Control With MATLAB/Simulink Implementation.* Bachelor’s thesis, LAB University of Applied Sciences Technology Lappeenranta, 2020.

[32] G. Gupta and S. Abdallah, “Propeller force-constant modeling for multirotor UAVs from experimental estimation of inflow velocity,” *International Journal of Aerospace Engineering*, 2018.

[33] NASA, “Modern Lift Equation,” 2021.

[34] A. Portescap, “Rotor inertia,” 2002.

[35] N. F. R. I. O. TECHNOLOGY), *Quadcopter Plant Model and Control System Development With MATLAB.* PhD thesis, KATE GLEASON COLLEGE OF ENGINEERING, 2017.

[36] Eric, “Brushless Motor Kv Constant Explained • LearningRC,” 2015.

[37] T. A. Ayodeji OKUBANJO*,a, Oluwadamilola OYETOLA a, Olawale OLALUWOYEa , Olufemi ALAO a, Adeyemi OLATEJUb, “Modeling and Simulation of DC Motor Using Simelectronics and SimulinkNo Title,” *Gazi Mühendislik Bilimleri Dergisi*, p. 10, 2019.

[38] P. T. Team, “HolyBro QAV250 + Pixhawk4-Mini Build,” 2020.

Appendices

A Math

A.1 table

p/d	e_d *range*	Typical e_d
$0.4 \leq p/d < 0.8$	$0.90 - 0.92$	0.91
$0.8 \leq p/d < 0.9$	0.87-0.89	0.88
$p/d \geq 0.9$	0.80-0.81	0.80

B Digital Attachment

Included in this thesis is the Simulink program of the drone system. Also included is the literature review which was submitted in the half way report. Also attached is the math.bib file with all the References in this thesis.