

# Automated Landing of a Quadrotor Unmanned Aerial Vehicle on a Translating Platform

by

Paul Daniel Stefanus Möller

*Thesis presented in partial fulfilment of the requirements for the degree of  
Master of Science in Engineering  
at Stellenbosch University*



Supervisor:

Mr J.A.A. Engelbrecht  
Department Electrical and Electronic Engineering

December 2015

# Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

December 2015

Copyright © 2015 Stellenbosch University  
All rights reserved

# Abstract

In this thesis the design, implementation and practical testing of an autolanding system responsible for landing a quadrotor Unmanned Aerial Vehicle (UAV) on a moving platform are presented.

The goal of the system was to land the quadrotor used in the Surface Launched Aerial Decoy electric (SLADe) project autonomously on the deck of a moving ship. The quadrotor, which is designed completely in-house in the Electronic Systems Laboratory at the University of Stellenbosch, has full autonomous flight capabilities and has a payload carrying capacity of  $6\ kg$ . This thesis specifically concentrates on the translational movement of the ship deck, and not rolling, pitching and heaving motion. The goal of the project was successfully obtained through modification of the quadrotor's flight control system as well as the Differential Global Position System (DGPS) sensing method of the platform and the UAV, and through implementation of a state machine that guided the aircraft through the autolanding procedure.

The flight control system of the UAV was thoroughly analysed, and shortcomings in the control system for this project were identified. The flight control system was not capable of tracking a moving platform with zero steady-state error and lacked disturbance rejection. The shortcomings were amended by modifying the architecture and increasing the bandwidth of the control loops responsible for horizontal velocity and horizontal position control of the UAV. A fundamental change was made to the aircraft's positional sensing method, expanding the DGPS capabilities of the vehicle. The DGPS, which is manufactured by Novatel, was modified to provide centimetre-level accurate relative position measurements between the platform and the UAV. The modifications allowed for accurate control of the position of the UAV relative to the moving platform. A state machine was implemented to safely guide the quadrotor through the landing procedure. The state machine consists of four consecutive states, and incorporates several safety checks to ensure that the quadrotor touches down safely within the required bounds.

The quadrotor successfully performed autonomous landings on a translating platform at speeds of  $10$ ,  $20$  and  $30\ km/h$ , and touched down within  $36\ cm$  of the designated landing mark.

# Uitreksel

In hierdie tesis word die ontwerp, implementering en praktiese toetsing aangebied van 'n outomatiese landingstelsel wat verantwoordelik is vir die landing van 'n vierrotor- onbemande vliegtuig op 'n bewegende platform.

Die doelwit van die stelsel was om die vierrotor wat in die projek genaamd "Surface Launched Aerial Decoy electric"(SLADe) gebruik is, outonom op die dek van 'n bewegende skip te land. Die vierrotor, wat in sy geheel in die Universiteit Stellenbosch se Elektroniese Stelsel-laboratorium ontwerp is, het ten volle outonome vlugvermoë en 'n loonvrag-drakapasiteit van  $6\ kg$ . Hierdie tesis konsentreer spesifiek op die translasionele beweging van die skipdek, en nie die rol-, hei- of dein-bewegings nie. Die doelwit van die projek is suksesvol bereik deur aanpassings te maak aan die vierrotor se vlugbeheerstelsel asook aan die Differensiële Globale Posisioneringstelsel-waarnemingsmetode van die platform en die onbemande vliegtuig, en deur die implementering van 'n toestandsmasjien wat die vliegtuig deur die outomatiese landingsprosedure kon lei.

Die vlugbeheerstelsel van die onbemande vliegtuig is deeglik ontleed, en tekortkominge in die beheerstelsel van die projek is geïdentifiseer. Die vlugbeheerstelsel was nie in staat om 'n bewegende platform te volg met 'n bestendigetoestandsfout van nul nie, en het 'n gebrek aan steuringsverwerping getoon. Die tekortkominge is aangespreek deur die argitektuur aan te pas en die bandwydte te verhoog van die beheerlusse wat verantwoordelik is vir die horisontale snelheid en horisontale posisiebeheer van die onbemande vliegtuig. 'n Fundamentele verandering is aan die vliegtuig se posisiewaarnemingsmetode aangebring om die DGPS-vermoë van die voertuig uit te brei. Die DGPS, wat deur Novatel vervaardig word, is aangepas om relatiewe posisiometings met sentimetervlak-akkuraatheid te kan voorsien tussen die platform en die onbemande vliegtuig. Die aanpassings het die akkurate beheer van die posisie van die onbemande vliegtuig relatief tot die bewegende platform moontlik gemaak. 'n Toestandsmasjien is geïmplementeer om die vierrotor veilig deur die landingsprosedure te lei. Die toestandsmasjien bestaan uit vier opeenvolgende toestande, en inkorporeer verskeie veiligheidstoetse om te verseker dat die vierrotor veilig binne die vereiste grense land.

Die vierrotor het suksesvol outonome landings op 'n transferende platform uitgevoer teen  $10, 20$  en  $30\ km/h$ , en het binne  $36\ cm$  van die aangewese landingsmerk neergestryk.

# Contents

<b>Abstract</b>	<b>iii</b>
<b>Uitreksel</b>	<b>iv</b>
<b>List of Figures</b>	<b>ix</b>
<b>List of Tables</b>	<b>xiv</b>
<b>Nomenclature</b>	<b>xv</b>
<b>Acknowledgements</b>	<b>xix</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Project Background . . . . .	1
1.2 Problem Statement . . . . .	2
1.3 Literature Study . . . . .	3
1.3.1 Automated Landing Systems And Strategies . . . . .	3
1.3.2 Sensor Configurations For Automated Landings . . . . .	6
1.3.3 Mathematical Models . . . . .	8
1.4 Project Overview . . . . .	10
1.5 Thesis Layout . . . . .	11
<b>2 Overview of Quadrotor Unmanned Aerial System</b>	<b>13</b>
2.1 The Unmanned Aerial System . . . . .	13
2.2 The Quadrotor . . . . .	15
2.3 Unmanned Aerial System Hardware And Software . . . . .	16
2.3.1 Quadrotor Hardware And Software . . . . .	16
2.3.1.1 Sensors . . . . .	16
2.3.1.2 Processing . . . . .	17
2.3.1.3 Communication . . . . .	18
2.3.2 Ground Station Hardware and Software . . . . .	18
2.3.2.1 Sensors . . . . .	18
2.3.2.2 Processing . . . . .	19
2.3.2.3 Communication . . . . .	20
2.3.3 Differential GPS . . . . .	20
2.4 Quadrotor Control System . . . . .	20
<b>3 Automatic Landing Strategy</b>	<b>22</b>
3.1 Proposed Concept for Autoland on a Moving Platform . . . . .	22
3.2 Proposed Sensor Configuration . . . . .	24
3.3 Flight Demonstration . . . . .	25
<b>4 Mathematical Aircraft Model</b>	<b>27</b>

4.1	Axis Systems and Notation . . . . .	27
4.1.1	Inertial Axes . . . . .	27
4.1.2	Body Axes . . . . .	28
4.1.3	Wind Axes . . . . .	28
4.1.4	Notation . . . . .	29
4.2	Kinetics and Kinematics . . . . .	30
4.2.1	Kinetics . . . . .	30
4.2.2	Kinematics . . . . .	31
4.2.2.1	Euler Angles . . . . .	31
4.2.2.2	Attitude Dynamics . . . . .	32
4.2.2.3	Position Dynamics . . . . .	32
4.3	Forces and Moments . . . . .	34
4.3.1	Actuators . . . . .	34
4.3.2	Aerodynamics . . . . .	35
4.3.3	Gravity . . . . .	37
4.4	Sensor Models . . . . .	37
4.5	Wind Model . . . . .	38
<b>5</b>	<b>Control System Analysis and Design</b>	<b>39</b>
5.1	Analysis of Pre-Existing Control System For Autolanding On A Moving Platform . . . . .	39
5.1.1	Horizontal Control System . . . . .	40
5.1.1.1	Pitch Rate Dynamics . . . . .	41
5.1.1.2	Pitch/Roll Rate Controllers . . . . .	41
5.1.1.3	Tilt Angle Controller . . . . .	48
5.1.1.4	Horizontal Velocity Controller . . . . .	52
5.1.1.5	Horizontal Position Controller . . . . .	57
5.1.2	Yaw Control System . . . . .	60
5.1.2.1	Yaw Dynamics . . . . .	60
5.1.2.2	Yaw Rate Controller . . . . .	61
5.1.2.3	Yaw Angle Controller . . . . .	66
5.1.3	Vertical Control System . . . . .	69
5.1.3.1	Heave Dynamics . . . . .	70
5.1.3.2	NSA Controller . . . . .	70
5.1.3.3	Climb Rate Controller . . . . .	72
5.1.3.4	Altitude Controller . . . . .	76
5.2	Control System Re-Design For Autolanding On a Moving Platform . . . . .	79
5.2.1	Design Goals . . . . .	80
5.2.2	Horizontal Roll/Pitch Rate Controller Re-Design . . . . .	83
5.2.3	Tilt Angle Controller . . . . .	89
5.2.4	Horizontal Velocity Controller . . . . .	91
5.2.5	Horizontal Position Controller . . . . .	97
5.3	Non-Linear Software Simulation . . . . .	100
5.3.1	Non-Linear Simulation Setup . . . . .	101
5.3.2	Simulation Results . . . . .	102
5.4	Discussion . . . . .	106
<b>6</b>	<b>Vehicle and Platform Sensing</b>	<b>108</b>
6.1	Vehicle and Platform Sensing Requirements . . . . .	108
6.2	GPS and DGPS Background . . . . .	109
6.3	Pre-Existing DGPS System . . . . .	109
6.4	Proposed Sensing Method . . . . .	111
6.5	Re-Designed GPS and DGPS Hardware and Software Architecture . . . . .	113

6.5.1	Hardware Modifications . . . . .	113
6.5.2	Software Modifications . . . . .	116
6.6	Sensing Validation Tests . . . . .	117
6.6.1	Sensing Validation Test Plan . . . . .	117
6.6.2	Static Test . . . . .	118
6.6.3	DGPS Communication Failure Test . . . . .	119
6.6.4	Dynamic Test - Master Sensor Moving . . . . .	120
6.6.5	Dynamic Test - Rover Sensor Moving . . . . .	121
6.6.6	Dynamic Test - Both Sensors Moving . . . . .	122
6.6.7	Dynamic Test - RC Piloted Flight . . . . .	124
6.6.8	Dynamic Test - Autopiloted Flight . . . . .	124
6.6.9	Velocity Measurement Error Test . . . . .	125
6.6.10	Platform Velocity Send Rate Test . . . . .	128
6.6.11	Magnetic Interference Tests . . . . .	128
6.6.12	Test Campaign Discussion . . . . .	130
6.7	Sensing Safety and Propagation . . . . .	130
6.7.1	Sensing Safety . . . . .	130
6.7.2	Sensor Propagation . . . . .	132
6.7.3	Sensing Safety and Propagation Discussion . . . . .	134
<b>7</b>	<b>Automated Landing State Machine</b>	<b>135</b>
7.1	Proposed Autoland State Machine . . . . .	135
7.1.1	Tracking State . . . . .	136
7.1.2	Homing State . . . . .	136
7.1.3	Descending State . . . . .	137
7.1.4	Shutdown State . . . . .	138
7.2	Autoland State Machine For Flight Demonstration . . . . .	138
7.2.1	Emergency Stop Feature . . . . .	139
7.2.2	Standby State . . . . .	140
7.2.3	Tracking State . . . . .	141
7.2.4	Descending State . . . . .	142
7.2.5	Shutdown State . . . . .	142
7.3	State Machine User Interface . . . . .	143
<b>8</b>	<b>HIL Simulation</b>	<b>145</b>
8.1	Pre-Existing HIL Setup . . . . .	145
8.2	HIL Setup Modifications And Additions . . . . .	146
8.3	HIL Simulation Conditions . . . . .	149
8.4	Simulation Results . . . . .	149
8.5	Discussion . . . . .	153
<b>9</b>	<b>Practical Flight Tests</b>	<b>156</b>
9.1	The Translating Platform . . . . .	156
9.2	Overview of Flight Test Campaign . . . . .	158
9.3	Flight Test 1 (Coetzenburg, 6 November 2013) . . . . .	159
9.4	Flight Test 2 (Coetzenburg, 6 November 2013) . . . . .	163
9.5	Flight Test 3 (Helderberg Radio Flyers, 8 November 2013) . . . . .	165
9.6	Flight Test 4 (Coetzenburg, 29 November 2013) . . . . .	172
9.7	Flight Test 5 (Wingfield Aerodrome, 4 December 2013) . . . . .	176
9.7.1	Autoland Results At 10, 20 and 30 km/h . . . . .	178
9.7.2	Autoland Results At 40 km/h . . . . .	181
9.7.3	Final Flight Test Summary . . . . .	182
9.8	Discussion . . . . .	183

<b>10 Summary and Recommendations</b>	<b>185</b>
10.1 Summary and Conclusions . . . . .	185
10.2 Recommendations . . . . .	187
<b>A Additional Technical Information</b>	<b>189</b>
A.1 Power System . . . . .	189
A.2 IMU . . . . .	189
A.3 GPS . . . . .	190
<b>B Additional Modelling Information</b>	<b>194</b>
<b>C Control System Design</b>	<b>196</b>
C.1 Horizontal Dynamics State Space To Transfer Function . . . . .	196
C.2 Yaw Dynamics State Space To Transfer Function . . . . .	196
C.3 Vertical Dynamics State Space To Transfer Function . . . . .	197
<b>D Antenna Tube Design</b>	<b>198</b>
<b>E Additional HIL Simulation Results</b>	<b>201</b>
E.1 5 m/s Simulation . . . . .	201
E.2 7 m/s Simulation . . . . .	204
<b>F Additional Flight Test Results</b>	<b>207</b>
F.1 Flight Test 2 Attempt 2 . . . . .	207
F.2 Flight Test 3 . . . . .	208
F.2.1 Attempt 1 . . . . .	208
F.2.2 Attempt 4 . . . . .	210
F.3 Flight Test 4 . . . . .	211
F.3.1 Attempt 2 . . . . .	211
F.3.2 Attempt 3 . . . . .	212
F.3.3 Attempt 4 . . . . .	213
F.3.4 Attempt 5 . . . . .	213
F.4 Flight Test 5 . . . . .	214
F.4.1 Attempt 1 . . . . .	214
F.4.2 Attempt 2 . . . . .	216
F.4.3 Attempt 4 . . . . .	218
<b>Bibliography</b>	<b>220</b>

# List of Figures

1.1	SLADe Ducted Fan Concept . . . . .	1
1.2	SLADe Quadrotor Concept . . . . .	2
1.3	Thesis Layout . . . . .	11
2.1	The Quadrotor . . . . .	13
2.2	The Ground Station . . . . .	14
2.3	Command Structure . . . . .	14
2.4	Quadrotor Details . . . . .	15
2.5	Avionics Architecture . . . . .	17
2.6	Ground Station Architecture . . . . .	18
2.7	Ground Station Application Screenshot . . . . .	19
2.8	High-Level Control System Overview . . . . .	20
3.1	The States Of The Landing Procedure . . . . .	23
3.2	GPS Modes Of Operation (Not To Scale) . . . . .	24
3.3	Coordination Of All Vehicles During Test Execution . . . . .	26
4.1	Inertial Axis System . . . . .	27
4.2	Body Axis System . . . . .	28
4.3	Wind Axis System . . . . .	28
4.4	Vehicle Notation Used . . . . .	29
4.5	Velocity Vector Notation . . . . .	30
4.6	Euler Angle Representation . . . . .	31
4.7	Axis Yaw Rotation . . . . .	32
4.8	Actuator Forces . . . . .	34
4.9	How The Drag Reference Area And Coefficients Were Determined . . . . .	36
4.10	The Wind Model . . . . .	38
5.1	Top-Level Block Diagram of the Flight Control System . . . . .	39
5.2	Mixing Matrix . . . . .	40
5.3	Pre-Existing Pitch Rate Controller Architecture . . . . .	42
5.4	Pitch Rate Controller Block Diagram Used For Analysis . . . . .	42
5.5	Different Control Architectures for Pitch Rate Controller . . . . .	43
5.6	Step Response of Pre-Existing Pitch Rate System . . . . .	45
5.7	Disturbance Response of Pre-Existing Pitch Rate System . . . . .	45
5.8	Step Response of Pre-Existing Pitch Rate System With Limiters Included . . . . .	46
5.9	Open-Loop Bode Plot of Pre-Existing Pitch Rate System . . . . .	46
5.10	Root Locus of Pre-Existing Pitch Rate System . . . . .	47
5.11	Pre-Existing Tilt Angle Controller Architecture . . . . .	48
5.12	Tilt Angle Calculation by the Tilt Angle Controller . . . . .	49
5.13	Tilt Angle Controller Block Diagram Used For Analysis . . . . .	49
5.14	Open-Loop Bode Plot of Pre-Existing Tilt Angle System . . . . .	50

5.15 Step Response of Pre-Existing Tilt Angle System . . . . .	50
5.16 Root Locus of Pre-Existing Tilt Angle System . . . . .	51
5.17 Pre-Existing Horizontal Velocity Controller Architecture . . . . .	52
5.18 Acceleration To Angle Relationship . . . . .	53
5.19 Horizontal Velocity Controller Block Diagram Used For Analysis . . . . .	53
5.20 Root Locus of the Pre-Existing Horizontal Velocity System . . . . .	54
5.21 Open-Loop Bode Plot of the Pre-Existing Horizontal Velocity System . . . . .	55
5.22 Step Response of Pre-Existing Horizontal Velocity System . . . . .	55
5.23 Disturbance Response of Pre-Existing Horizontal Velocity System . . . . .	56
5.24 Step Response of Pre-Existing Horizontal Velocity System With Limiters . . . . .	56
5.25 Pre-Existing Horizontal Position Controller Architecture . . . . .	57
5.26 Horizontal Position Controller Block Diagram Used For Analysis . . . . .	57
5.27 Step Response of Pre-Existing Horizontal Position System . . . . .	58
5.28 Step Response of Pre-Existing Horizontal Position System With Limiters . . . . .	58
5.29 Open-Loop Bode Plot of Pre-Existing Horizontal Position System . . . . .	59
5.30 Root Locus of Pre-Existing Horizontal Position System . . . . .	59
5.31 Pre-Existing Yaw Rate Controller Architecture . . . . .	61
5.32 Yaw Rate Controller Block Diagram Used for Analysis . . . . .	62
5.33 Open-Loop Bode Plot of Pre-Existing Yaw Rate System . . . . .	63
5.34 Step Response of Pre-Existing Yaw Rate System . . . . .	64
5.35 Disturbance Response of Pre-Existing Yaw Rate System . . . . .	64
5.36 Step Response of Pre-Existing Yaw Rate System With Limiters . . . . .	65
5.37 Root Locus of Pre-Existing Yaw Rate System . . . . .	65
5.38 Pre-Existing Yaw Angle Controller Architecture . . . . .	66
5.39 Yaw Angle Controller Block Diagram Used For Analysis . . . . .	67
5.40 Open-Loop Bode Plot of Pre-Existing Yaw Angle System . . . . .	68
5.41 Step Response of Pre-Existing Yaw Angle System . . . . .	68
5.42 Root Locus of Pre-Existing Yaw Angle System . . . . .	69
5.43 Pre-Existing NSA Controller Architecture . . . . .	70
5.44 Open-Loop Bode Plot of Pre-Existing NSA System . . . . .	71
5.45 Step Response and Disturbance Response of Pre-Existing NSA System . . . . .	71
5.46 Root Locus of Pre-Existing NSA System . . . . .	72
5.47 Pre-Existing Climb Rate Controller Architecture . . . . .	72
5.48 Climb Rate Controller Block Diagram Used For Analysis . . . . .	73
5.49 Open-Loop Bode Plot of Pre-Existing Climb Rate System . . . . .	73
5.50 Step Response of Pre-Existing Climb Rate System . . . . .	74
5.51 Disturbance Response of Pre-Existing Climb Rate System . . . . .	74
5.52 Step Response of Pre-Existing Climb Rate System With Limiters . . . . .	75
5.53 Root Locus of Pre-Existing Climb Rate System . . . . .	75
5.54 Pre-Existing Altitude Controller Architecture . . . . .	76
5.55 Altitude Controller Block Diagram Used For Analysis . . . . .	76
5.56 Open-Loop Bode Plot of Pre-Existing Altitude System . . . . .	77
5.57 Step Response of Pre-Existing Altitude System . . . . .	77
5.58 Root Locus of Pre-Existing Altitude System . . . . .	78
5.59 Root Locus (Near the Origin) of Pre-Existing Altitude System . . . . .	78
5.60 Proposed Position Controller Architecture . . . . .	81
5.61 Proposed Pitch/Roll Rate Control Loop Architecture . . . . .	84
5.62 Open-Loop Bode Plot of Pitch/Roll Rate System With Integral Control Law . .	85
5.63 Open-Loop Bode Plot of Pitch/Roll Rate System With PI Control Laws . . .	86
5.64 Open-Loop Bode Plot of Pitch/Roll Rate System With Lead Compensators . .	87
5.65 Open-Loop Bode Plot of Re-Designed Pitch/Roll Rate System . . . . .	87
5.66 Step Response and Disturbance Response of Re-Designed Pitch/Roll Rate Systems	88

5.67 Open-Loop Bode Plot of Proposed Control Structures for Pitch/Roll Rate Controller Re-Design . . . . .	89
5.68 Bode Plot of Re-Designed Tilt Angle Controller . . . . .	90
5.69 Step Response of Pre-Existing and Re-Designed Tilt Angle Systems . . . . .	91
5.70 Proposed Horizontal Velocity System Architecture . . . . .	92
5.71 Open-Loop Bode Plot of Horizontal Velocity Plant And Integral Term . . . . .	93
5.72 Open-Loop Bode Plot Comparison of Low- and High-Frequency Lead Compensators . . . . .	94
5.73 Open-Loop Bode Plot of Horizontal Velocity System with Low- and High-Frequency Lead Compensators . . . . .	94
5.74 Open-Loop Bode Plot Of Re-Designed Horizontal Velocity System . . . . .	95
5.75 Step Response of Pre-Existing and Re-Designed Horizontal Velocity Systems . . . . .	95
5.76 Step Response of Pre-Existing and Final Re-Designed Horizontal Velocity Systems, With Limiters . . . . .	96
5.77 Proposed Horizontal Position Controller Architecture . . . . .	97
5.78 Open-Loop Bode Plot of Pre-Existing and Re-Designed Horizontal Position Systems . . . . .	98
5.79 Step Response of Pre-Existing and Re-Designed Horizontal Position Systems . . . . .	99
5.80 Step Response of Pre-Existing and Re-Designed Horizontal Position Systems, With Limiters . . . . .	100
5.81 The Simulink Block Diagram Used For SIL and HIL Simulations . . . . .	102
5.82 The Three Dimensional Visualization Environment . . . . .	102
5.83 Non-Linear Simulation North Position Step Response . . . . .	103
5.84 Non-Linear Simulation East Position Step Response . . . . .	103
5.85 Non-Linear Simulation Acceleration Command During North Position Step . . . . .	104
5.86 Non-Linear Simulation Pitch And Roll Angles During Position Steps . . . . .	105
5.87 Non-Linear Simulation Disturbance on Heave Control System . . . . .	105
5.88 Non-Linear Simulation Disturbance on Yaw Angle . . . . .	106
6.1 ENU Vector Definition . . . . .	111
6.2 Platform and Vehicle Position and Velocity Sensing Communication Structure . . . . .	114
6.3 The Wifly Module Alone and Installed on the Breakout Board . . . . .	114
6.4 The Wifly Module Positioned Below the Telecommand and Telemetry Antenna . . . . .	115
6.5 The Tab Used for Configuring the Vehicle and Ground Station for Align <sup>TM</sup> or RTK Operation . . . . .	116
6.6 The Software Structure Used to Extract GPS Measurements on the Ground Station and OBC . . . . .	117
6.7 MasterPos and RoverPos Bias Error . . . . .	118
6.8 ENU Bias Error . . . . .	119
6.9 DGPS Communication Failure Test. . . . .	120
6.10 Master Sensor Movement Test Result . . . . .	121
6.11 Rover Sensor Movement Test Result . . . . .	122
6.12 Movable Align <sup>TM</sup> DGPS Test Rig . . . . .	122
6.13 Logged MasterPos and RoverPos Measurements During Translating and Rotating Test . . . . .	123
6.14 Logged ENU Measurements During Translating and Rotating Test . . . . .	123
6.15 ENU Measurements When the Vehicle Was Flown and Carried . . . . .	124
6.16 Measurement Data From The ENU Vector During The Step Command Test . . . . .	125
6.17 Master And Rover Sensor Speed Measurement Error . . . . .	126
6.18 Velocity Measurement Error PDF . . . . .	126
6.19 GPS Heading Measurement Test Results . . . . .	127
6.20 The Vehicle Stationed on the Platform . . . . .	129
6.21 Magnetometer Influence Test . . . . .	129

6.22 DGPS Communication Failure Detection and Safety Mechanism . . . . .	131
6.23 ENU Vector Propagation Test Setup . . . . .	132
6.24 ENU Vector And Propagated ENU Vector . . . . .	133
 7.1 The Error Bounds During the Tracking State . . . . .	136
7.2 The Homing Action and the Error Bounds During the Homing State . . . . .	137
7.3 The Error Bounds During the Descending State . . . . .	138
7.4 Diagrammatic Representation of the Sequence of Events During the Flight Demonstration Test . . . . .	140
7.5 The State Machine Status Indicators and Adjustable Parameters . . . . .	143
 8.1 Pre-Existing HIL Setup . . . . .	145
8.2 Final HIL Setup . . . . .	147
8.3 Ground Station Application and Virtual Platform Application . . . . .	147
8.4 HIL Simulation and Flight Demonstration Test Execution Sequence . . . . .	148
8.5 The Three-Dimensional Platform Model . . . . .	148
8.6 HIL Simulation Absolute Horizontal Position Error During 3 m/s Simulation . .	150
8.7 HIL Simulation ENU Vector Up Component During 3 m/s Simulation . . . . .	150
8.8 HIL Simulation Vehicle and Platform Velocity During 3 m/s Simulation . . .	151
8.9 HIL Simulation Vehicle and Platform Heading During 3 m/s Simulation . . .	151
8.10 HIL Simulation Vehicle Pitch Angle and Vertical Accelerometer Spike During 3 m/s Simulation . . . . .	152
8.11 The Touchdown Locations Relative to the Center of the Platform During 3 m/s Simulation . . . . .	152
8.12 Touchdown Locations During 5 m/s and 7 m/s HIL Simulations . . . . .	153
 9.1 The Perforated Platform On Which the Vehicle Will Land . . . . .	156
9.2 The Moving Platform Assembly . . . . .	157
9.3 The Vehicle Pitching During a Forward Position Step . . . . .	159
9.4 How The ENU Vector Is Used During Position Steps . . . . .	160
9.5 Forward Step Response Using Pre-Existing and Re-Designed Horizontal Controllers . . . . .	160
9.6 Pitch Angle During Forward Step for Pre-Existing and Re-Designed Horizontal Controllers . . . . .	161
9.7 Vertical Disturbance Rejection Due To Horizontal Position Step, Using Re-Designed Controllers . . . . .	162
9.8 Vertical Disturbance Rejection Due To Horizontal Position Step, Using Pre-Existing Controllers . . . . .	162
9.9 The Vehicle Decelerating Aggressively During an Emergency Stop . . . . .	163
9.10 Forward Velocity References and Measurement During the Second Flight Test . .	164
9.11 Recorded Pitch Angle During The Second Flight Test. . . . .	165
9.12 The Vehicle After Touchdown on the Virtual Platform . . . . .	165
9.13 Tracking of the Virtual Platform During Second Attempt . . . . .	168
9.14 Illustration of the Phenomenon in the Virtual Platform Position Due to Telemetry Link Delays . . . . .	169
9.15 Virtual Landing Absolute Horizontal Position Error During Second Attempt . .	169
9.16 Virtual Landing Vertical Reference and Position During Second Attempt . . .	170
9.17 Tracking of the Virtual Platform During Third Autolanding Attempt . . . . .	171
9.18 Virtual Landing Absolute Horizontal Position Error During Third Attempt . . .	171
9.19 The Vehicle Homing in on the Platform During a Stationary Autonomous Landing	172
9.20 The Touchdown Locations On the Platform . . . . .	173
9.21 Stationary Autolanding Absolute Horizontal Position Error During the First Attempt . . . . .	174
9.22 The ENU Up Component During The First Stationary Landing. . . . .	175

9.23 Recorded Body z-Axis Accelerometer Measurements For the First Stationary Autolanding . . . . .	175
9.24 The Vehicle Tracking the Platform on the Trailer . . . . .	177
9.25 Coordination Of All Vehicles During the Flight Demonstration Test . . . . .	178
9.26 Absolute Horizontal Position Error During 30 km/h Autolanding . . . . .	178
9.27 The Vertical Error While Landing At 30 km/h. . . . .	179
9.28 Platform and Vehicle Velocity and Heading During 30 km/h Autolanding . . . . .	180
9.29 Body z-Axis Specific Force During 30 km/h Autolanding . . . . .	180
9.30 The Vehicle Missing the Platform During the 40 km/h Landing Attempt . . . . .	181
9.31 Absolute Horizontal Position Error During 40 km/h Autolanding Attempt . . . . .	182
9.32 Actual Landing Locations for 10, 20 and 30 km/h Autolandings, As Well As Believed Landing Location for 40 km/h Autolanding Attempt . . . . .	182
9.33 The Vehicle Tracking the Platform . . . . .	183
 B.1 GPS Position Sensing Noise Model . . . . .	195
B.2 Rotor Step Response . . . . .	195
 D.1 DGPS Communication Antenna Holder Drawing . . . . .	198
D.2 Telemetry Communication Antenna Holder Drawing . . . . .	199
D.3 Antenna PVC Tube . . . . .	200
 E.1 HIL 5 m/s Absolute Horizontal Position Error . . . . .	201
E.2 HIL 5 m/s Simulation ENU Vector Up Component . . . . .	202
E.3 HIL 5 m/s Simulation Vehicle and Platform Velocity, Heading . . . . .	203
E.4 HIL 5 m/s Simulation Vehicle Pitch Angle, Vertical Accelerometer Spike . . . . .	203
E.5 HIL 7 m/s Absolute Horizontal Position Error . . . . .	204
E.6 HIL 7 m/s Simulation ENU Vector Up Component . . . . .	205
E.7 HIL 7 m/s Simulation Vehicle and Platform Velocity, Heading . . . . .	205
E.8 HIL 7 m/s Simulation Vehicle Pitch Angle, Vertical Accelerometer Spike . . . . .	206
 F.1 Forward Velocity Reference and Response During the Second Attempt at the Second Test . . . . .	207
F.2 Recorded Pitch Angle During the Second Attempt at the Second Flight Test . . . . .	208
F.3 Tracking Of The Virtual Platform During First Attempt of Third Flight Test . . . . .	209
F.4 Virtual Landing Absolute Horizontal Position Error During First Attempt at Third Flight Test . . . . .	209
F.5 Tracking Of The Virtual Platform During Fourth Attempt at Third Flight Test . . . . .	210
F.6 Virtual Landing Absolute Horizontal Position Error During Fourth Attempt at Third Flight Test . . . . .	211
F.7 Absolute Horizontal Error, Flight Test 4, Attempt 2 . . . . .	212
F.8 Absolute Horizontal Error, Flight Test 4, Attempt 3 . . . . .	212
F.9 Absolute Horizontal Error, Flight Test 4, Attempt 4 . . . . .	213
F.10 Absolute Horizontal Error, Flight Test 4, Attempt 5 . . . . .	213
F.11 Absolute Horizontal Position Error During 10 km/h Automated Landing . . . . .	214
F.12 Vertical Error During 10 km/h Automated Landing . . . . .	215
F.13 Platform and Vehicle Velocity, Heading, During 10 km/h Automated Landing . . . . .	215
F.14 Absolute Horizontal Position Error During 20 km/h Automated Landing . . . . .	216
F.15 Vertical Error During 20 km/h Automated Landing . . . . .	217
F.16 Platform and Vehicle Velocity, Heading, During 20 km/h Automated Landing . . . . .	217
F.17 Vertical Error During 40 km/h Automated Landing Attempt . . . . .	218
F.18 Platform and Vehicle Velocity, Heading, During 40 km/h Automated Landing Attempt . . . . .	219

# List of Tables

4.1	Kinetic States Measured By On-Board Sensors . . . . .	37
5.1	Pre-Existing Pitch Rate Controller Limits and Their Functions . . . . .	42
5.2	Pre-Exisiting Yaw Rate Controller Limits and Their Functions . . . . .	61
5.3	Horizontal Control System Frequency Characteristics . . . . .	82
5.4	Pre-Existing and Re-Designed Horizontal Velocity System Parameters . . . . .	96
5.5	Re-Designed and Pre-Existing Horizontal Position System Parameters . . . . .	98
6.1	Novatel Position Types and Their Associated Accuracies . . . . .	110
6.2	Packets Available to the Vehicle (Master) . . . . .	112
6.3	Packets Available to the Platform (Rover) . . . . .	113
6.4	Sensor Test Campaign Strategy . . . . .	117
6.5	Average Rover And Master PDF Characteristics . . . . .	127
8.1	Relevant Parameters At Touchdown For Each HIL Simulation . . . . .	154
9.1	Flight Test Campaign Summary . . . . .	158
9.2	The Packet Sent to the Vehicle During Virtual Platform Landings . . . . .	166
9.3	Parameters Used During the Virtual Platform Landing Flight Test . . . . .	167
9.4	Stationary Platform Landing Accuracy . . . . .	174
9.5	Maximum Accelerometer Spike Measured In z Body Axis. . . . .	176
9.6	Translating Platform Landing Accuracy . . . . .	183
A.1	Quadrotor Power System Technical Details . . . . .	189
A.2	IMU Sensor Information . . . . .	190
A.3	DGPS Sensor Information . . . . .	190
A.4	BestPos Packet Binary Format . . . . .	190
A.5	BestVel Packet Binary Format . . . . .	191
A.6	ENU Packet Binary Format . . . . .	191
A.7	MasterPos Packet Binary Format . . . . .	192
A.8	RoverPos Packet Binary Format . . . . .	192
B.1	Gyroscope, Accelerometer and GPS Simulation Noise Parameters . . . . .	194

# Nomenclature

## Abbreviations and Acronyms

ATOL	Automated Take-Off And Landing
BLWN	Band-Limited White Noise
CAN	Controller Area Network
CG	Center of Gravity
CPU	Central Processing Unit
DC	Direct Current
DCM	Direct Cosine Matrix
DGPS	Differential Global Positioning System
EKF	Extended Kalman Filter
ENU	East-North-Up
ESC	Electronic Speed Control
ESL	Electronic Systems Laboratory
FLIR	Forward Looking Infrared
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GSA	Ground Station Application
HIL	Hardware-In-the-Loop
HRF	Helderberg Radio Flyers
IGE	In-Ground Effect
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IP	Internet Protocol
LED	Light-Emitting Diode
LiDAR	Light Detection And Ranging
LQ	Linear Quadratic

MSL	Mean Sea Level
NA	Not Applicable
OBC	On-Board Computer
PC	Personal Computer
PI	Proportional Integral
PID	Proportional Integral Derivative
PSD	Power Spectral Density
PWM	Pulse Width Modulation
RC	Radio Control
RMS	Root Mean Square
RPT	Return Path Tracking
RTK	Real-Time-Kinematic
SEP	Spherical Error Probable
SIL	Software-In-the-Loop
SLADE	Surface Launched Aerial Decoy electric
SRGPS	Ship Relative Global Positioning System
TCP	Transmission Control Protocol
UAV	Unmanned Aerial Vehicle
VPA	Virtual Platform Application

**Greek Letters**

$\alpha$	Angle of attack
$\beta$	Angle of side slip
$\delta_{(.)R}$	Reference for virtual actuator (.)
$\delta_A$	Virtual aileron deflection
$\delta_E$	Virtual elevator deflection
$\delta_R$	Virtual rudder deflection
$\delta_T$	Throttle value
$\theta$	Euler roll angle
$\mu$	Mean value (Statistics)
$\rho$	Air density
$\sigma$	Standard deviation (Statistics)
$\tau$	Time constant
$\phi$	Euler pitch angle

$\psi$	Euler yaw angle
--------	-----------------

**Lowercase Letters**

$d$	Moment arm
$e_{ss}$	Steady-state error
$g$	Acceleration of gravity
$m$	Mass
$r_D$	Rotor chord Length
$t$	Time
$v$	Linear velocity
$x_{(.)}, y_{(.)}, z_{(.)}$	x-, y- and z-axis of any arbitrary right-hand orthogonal axis system (.)

**Uppercase Letters**

$A$	Reference area
$C_{(.)}$	Cosine of (.)
$C_D$	Drag coefficient
$I_{(.)(.)}$	Moment of inertia around the principle inertial axis (.)()
$\dot{N}, \dot{E}, \dot{D}$	Velocities in the x-, y- and z-axis in the inertial reference frame
$K_d$	Derivative gain
$K_i$	Integral gain
$K_p$	Proportional gain
$K_V$	Velocity error constant
$L, M, N$	Moments around the x-, y- and z-axis in the body reference frame
$L^{(.)}, M^{(.)}, N^{(.)}$	Moments around the x-, y- and z-axis in the body reference frame due to (.)
$O_{(.)}$	Origin of reference frame (.)
$P, Q, R$	Angular rates around the x-, y- and z-axis in the body reference frame
$\dot{P}, \dot{Q}, \dot{R}$	Angular Acceleration around the x-, y- and z-axis in the body reference frame
$R_{LD}$	Rotor lift to drag ratio
$S_{(.)}$	Sine of (.)
$T_{(.)}$	Instantaneous thrust of motor (.)
$T_{(.)R}$	Reference thrust of motor (.)
$\dot{T}_{(.)}$	Rate of change in thrust of motor (.)
$U, V, W$	Velocities along the x-, y- and z-axis in the body reference frame

$\dot{U}, \dot{V}, \dot{W}$	Acceleration along the x-, y- and z-axis in the body reference frame
$\bar{V}$	Velocity magnitude of the vehicle
$X, Y, Z$	Forces along the x-, y- and z-axis in the body reference frame
$X^{(.)}, Y^{(.)}, Z^{(.)}$	Forces along the x-, y- and z-axis in the body reference frame due to (.)
$X_{(.)}, Y_{(.)}, Z_{(.)}$	Direction of the x-, y- and z-axis in reference frame (.)

**Subscripts**

$B$	Body reference frame
$I$	Inertial reference frame
$W$	Wind reference frame
$xx, yy, zz$	Principle inertial x-, y- and z-axis

**Superscripts**

$D$	Moments or forces induced by aerodynamic drag
$G$	Moments or forces induced by gravity
$T$	Moments or forces induced by the actuators

# Acknowledgements

I want to acknowledge the following people and organizations for their support and partaking in the project. I would not have been able to successfully complete the thesis if it was not for them:

- First and foremost my family, for their continual loving support and their eagerness to listen to my project's progress and my concerns. I would like to thank my father, Paul Möller, my mother, Magda Möller and my sister Liza Möller.
- My study leader, Japie Engelbrecht, for his insight and guidance throughout the project. Thank you for being understanding, being accessible to your students and for being thoroughly involved in the project from beginning until completion. Without your input, I would not have been able to complete the thesis.
- The friends that I made during the course of the thesis: especially Suandrie McLaren, Louise Steyn, Shaun Irwin, Du Toit Hannekom and Johan Botha, who ensured that we have fun even when working very late hours.
- The lab engineers of 2012, Anton Runhaar and Lionel Basson, for their initial support in getting me started on the hardware for the project.
- The new lab engineers of 2012 and 2013, Wiaan Beeton and Nico Alberts, for their support and hardware development during the course of the thesis. Without their input I would not have been able to complete the project successfully.
- The safety pilot of the ESL, Micheal Basson, for helping with flight tests and always ensuring that the quadrotor is safe during flight tests.
- Jolette Roodt, for her proficiency in proof reading and text editing of this document.
- Last but not least, I would like to thank Armscor for providing the needed funding for the project. Without their funding I would not have been able to do a thesis on such an interesting topic.

# Chapter 1

## Introduction

### 1.1 Project Background

Unmanned Aerial Vehicles (UAVs) are becoming more common and relevant in society, gaining popularity in recreational, civil and military applications. Along with the expanding roles of UAVs, the need is growing to automate them to perform dull, dirty or dangerous tasks.

This project originates from joint ongoing research and development between the Electronic Systems Laboratory (ESL) from the University of Stellenbosch and the Institute for Maritime Technology. Research was started in 2007, with the goal of developing a countermeasure system to defend naval ships against hostile anti-ship missiles. The system makes use of a transponder to increase the echo strength of the missile radar, effectively making the target appear larger than it is. The transponder must, however, be carried out to sea to divert the missile safely from the ship's position. It is proposed that a UAV carry the transponder off board.

This UAV is recoverable, with a typical flight mission consisting of four consecutive phases, namely take-off, decoy, return and landing. A vehicle was designed and built from scratch by engineers in the ESL. The aircraft was dubbed SLADe, the name of the project, which is short for Surface Launched Aerial Decoy electric. The vehicle was based on a ducted fan concept using two counter-rotating fixed-pitch propellers in a cylindrical fuselage. The UAV is driven by two electric motors, ensuring reliable and prompt deployment. Control was performed by flap-like control surfaces, vectoring the thrust of the propellers. SLADe can be seen in Figure 1.1.



**Figure 1.1 – SLADe Ducted Fan Concept**

This concept, however, appeared to have limited agility and payload carrying capacity, which led to a completely re-designed vehicle. The re-designed SLADe is a multirotor, using four electric-driven rotors for actuation. This kind of UAV is more commonly known as a quadrotor. The re-designed SLADe is shown in Figure 1.2.



**Figure 1.2 – SLADe Quadrotor Concept**

The quadrotor concept has the payload carrying capacity and the maneuverability to perform the decoy mission. Currently SLADe has the ability to autonomously take off, perform the decoy and return to the ship. SLADe can furthermore perform an autonomous landing on a stationary ship, but SLADe does not have the ability to autonomously land on a moving ship.

Landing on a moving ship poses a complex problem, as all six degrees of freedom of the ship can have a notable impact on the landing procedure. The ship could be rolling, pitching, yawing, heaving and translating while the aircraft needs to touch down. The problem is complicated further by the ever-changing wind and sea conditions around the ship, which can be challenging to estimate and predict.

This project will focus on developing a method to autonomously land SLADe on a moving vehicle such as a ship. The goal is narrowed down to specifically land SLADe on a translating-only ship or platform. Other former students of the ESL, namely Phillip Bellstead and Dewald de Swart, focused on landing a helicopter on a rolling, pitching and heaving deck. The research and development done in this project is, however, not limited to this specific application, being applicable to wherever a rotary wing aircraft needs to land autonomously on a translating object.

## 1.2 Problem Statement

This section elaborates on the project goal, by establishing the resources available, quantifying relevant parameters and defining the project scope. In a nutshell, an autonomous landing system capable of landing a rotary wing aircraft on a translating platform must be designed. After design, the system must be simulated, implemented and practically verified.

The aircraft that will be used is the SLADe quadrotor UAV. The autonomous landing system will build upon the existing firmware and control architecture. A ship is a resource that is

not available for the project. A substitute is proposed - a platform that will be mounted on a trailer, towed by a motor vehicle. This setup will adequately mimic the translational movement of a ship. The platform will therefore move in the latitudinal and longitudinal directions. For all practical purposes, the height and orientation of the platform will be constant throughout the landing procedure.

A platform is available in the ESL that can be fastened to the trailer. The platform is square with a side length of  $2\text{ m}$ . Given the physical size of the aircraft and the platform, a maximum positional landing error of  $0.5\text{ m}$  is allowed. The aircraft might otherwise fall off the platform upon touchdown.

The speed of the translating platform is assumed to be constant during the autonomous landing procedure. This is based on the assumption that the bandwidth of the ship's translational speed is much lower than the aircraft's. According to the South African Navy, the cruising speed of the SAS Amatola, a valour class frigate, is  $15\text{ kts}$ , or  $27.8\text{ km/h}$  [1], which is assumed to be the speed at which the aircraft will attempt to land on the ship. The speed of the translating platform is thus established to be  $30\text{ km/h}$ .

A maximum wind speed of  $5\text{ m/s}$  is assumed to be present, and is the final parameter to be quantified. Control system design and simulation will be done with this in mind. Practical flight tests will always be planned to take place in calm or no wind conditions, but provision is needed for immediate weather changes. However, prior to this project, SLADe has proven to be capable of flying in wind speeds of up to  $7\text{ m/s}$ .

With regard to the automated landing system, The following aspects will be investigated:

1. An automated landing strategy that will guide the vehicle through the landing process.
2. The sensor configuration needed to autonomously land on a translating platform.
3. SLADe's existing control system, to evaluate whether it is adequate to be used for the translating landing strategy.

Out of the above-mentioned aspects, control system analysis and design will receive the most attention. An in-depth analysis of SLADe's control system will be performed, followed by simulation and practical validation. This project's scope does not include an intensive study of sensor configurations and state estimation techniques. This is the scope and topic of the thesis of Shaun Irwin, another student currently in the ESL.

### 1.3 Literature Study

The literature study conducted during the course of the project is presented in this section. Three key aspects were investigated: automated landing systems and strategies, which will determine how the automated landing procedure is performed; sensor configurations, as the vehicle requires measurements or estimates of its own states and the states of the platform; and mathematical models of quadrotors, to make it possible to accurately design control systems and perform simulations.

#### 1.3.1 Automated Landing Systems And Strategies

Approaches and strategies that currently exist for this application are investigated. During the literature study it was observed that state machines are used in all automated landing strategies that were found. A closer look is taken into the states that some strategies use to perform automated landings. Their capabilities, limitations and applicability to this project are discussed.

In 2003 Saripalli et al. used a state machine approach to autonomously land a model helicopter on a stationary target [2]. There are three states: search, track and land. The helicopter searches for the landing target, then attempts to track the target by aligning itself with it. The state machine finally transitions to the landing state, commanding the helicopter to descend onto the platform. States in this state machine can be recurring. If the target is lost, the state machine re-enters the search state, or if the helicopter becomes misaligned during the landing state, it will re-enter the tracking state.

This automated landing strategy is practical in the sense that states can be recurring. The possibility of touching down on the landing location is increased. The criteria for determining whether a state should be redone must, however, be chosen carefully, as the aircraft might never complete the entire automated landing process. The landing criteria for this project require accurate tracking of the landing location ( $0.5\text{ m}$  landing accuracy) at a speed of  $30\text{ km/h}$ . A recurring state machine is therefore applicable, as there will be unforeseen disturbances affecting the aircraft and platform. The disturbances will potentially force the quadrotor to redo a state to meet the landing criteria, or terminate the landing procedure altogether. A recurring state technique will be used.

Later in 2007 Saripalli et al. did research on landing a model helicopter on a moving target [3]. Successful experiments were conducted where the helicopter landed on an imaginary platform moving at  $4\text{ m/s}$ . A state machine approach was again followed, with the state machine having four states: detection, tracking, motion planning and landing. The target is detected, then tracked by the helicopter. During tracking, the helicopter will match the speed of the platform and hover directly above the landing location. An optimal landing trajectory is then calculated by using the kinematic equations of the helicopter. This trajectory will ensure that the aircraft will touch down within a user-specified distance from the helicopter's current position. The helicopter finally lands on the platform by tracking this trajectory.

The disadvantage of this landing strategy is that the helicopter will touch down, regardless of its tracking performance during descent. If the helicopter is disturbed by external factors such as wind, it will continue descending and it might miss the landing location. The strategy is logical in the sense that the vehicle firstly tracks the platform before it starts descending, ensuring steady-state behavior relative to the moving platform before descending. Steady-state behavior before descending is also required for this project, so that transient movements are kept to a minimum while the vehicle is landing.

The state machine used by Wenzel et al. for landing a quadrotor on a translating platform consists of three states: track, approach and land [4]. The quadrotor firstly tracks a location that is a user-defined offset from the landing location. After 10 seconds of tracking, the aircraft approaches a tracking location directly above the designated landing mark, but slightly lower than the tracking height tracked in the track state. When the aircraft has reached the approaching point it descends and touches down on the platform.

A possible drawback of the strategy used by Wenzel et al. is that the criteria for advancing from the track state to the approach state is time dependent, and not position dependent. Thus even if the vehicle is not following the tracking location properly, it will approach the platform. However, the idea of tracking an offset of the landing location before coming into close proximity of the platform is a relevant concept to this project. Transients from catching up to the platform, or from matching the platform's speed, can then be negated at a safe distance away from the platform.

Voos et al. use a state machine with two states to land a quadrotor on a translating platform, appropriately called approach and land [5]. The quadrotor will approach the platform from an arbitrary position, tracking a safe height directly above the landing location. But unlike in Wenzel et al., the landing state is initiated by positional error instead of time dependence.

When the aircraft is horizontally within  $0.5\text{ m}$  of the landing location, it will start to descend at a constant rate.

The strategy used by Voos et al. is robust in that it uses the position error between the aircraft and the landing location to determine when it should descend. The state machine used for the automated landing procedure in this project will also be driven by positional measurements instead of time dependent advancement.

An autonomous landing strategy incorporating four states and path planning is used by Wu et al. to land a model helicopter on a moving target [6]. This research was, however, only proven successful in simulation, and practical experiments were not conducted. The strategy consists of four states: searching, pursuit, initial descent and final descent. Throughout all the states, the landing path is optimized and planned by use of linear programming. The optimization algorithm takes into account the dynamic constraints of both aircraft and platform. The helicopter determines the position of the platform in real-time and then starts to track the landing location. It descends initially to a predetermined height where the aircraft performs safety checks and gathers data to perform motion prediction. When deemed safe to land, the aircraft descends at a constant rate until touchdown.

This system is advantageous in that it firstly verifies that it is safe to land on the moving platform, giving greater confidence that the aircraft will touch down on the designated location. Safety checks are considered to be used for this project whereas trajectory generation is not considered, due to the added complexity. As stated in §1.2, focus will be placed on control system analysis and design.

Friis et al. followed a similar strategy, where their quadrotor tracked a moving platform from above the landing location before descending [7]. Their research, however, focused more on two different control system architectures to track the platform. Linear quadradic and classic Proportional Integral Differential (PID) control theories were applied and practically tested, both with great success. Wenzel et al. also implemented a consecutive PID loop control architecture to track a moving platform [4]. Similarly, a combination of Proportional and Proportional Differential controllers were used to control the attitude, velocity and position of the quadrotor used by Voos et al. [5].

As will be seen later, SLADe uses a consecutive PID loop control architecture. The research done by Friis et al., Wenzel et al. and Voos et al. confirms that this project can build upon SLADe's current control architecture to track the moving platform. Other control architectures are also used, like Fuzzy Logic. Olivares-Mendez et al. used a Fuzzy Logic controller to control the altitude of a model helicopter during descent of a stationary automated landing [8]. Fuzzy logic controllers will, however, not be considered for this project.

Focus is now shifted towards the pre-existing automated landing strategy that SLADe uses to land on a stationary platform. The state machine comprises of 3 states: descent, leveling and shutdown. Before the state machine becomes active the quadrotor is commanded by the Ground Station Officer, who can send instantaneous commands to SLADe through a wireless link, to descend to an initial landing altitude. When the aircraft is at the pre-landing altitude it must reach a steady-state. When viewed as an acceptable steady-state by the Ground Station Officer, the vehicle is commanded by the officer to land.

The state machine becomes active, entering the descent state, commanding the vehicle to descend at a constant rate. When the vehicle is  $1\text{ m}$  above the ground, where the In-Ground Effect (IGE) becomes significant, the state machine enters the leveling state, where control is changed from positional tracking to attitudinal tracking. In this state the vehicle attempts to maintain a level attitude. This control change is done to ensure that the vehicle does not aggressively roll and pitch in attempting to eliminate the disturbance from the IGE. The vehicle finally makes contact with the ground, where its accelerometers will measure a spike

in specific force. When the spike is measured, the shutdown state is entered, shutting down the rotors.

This landing strategy requires human intervention, where the Ground Station Officer acts as the safety mechanism to ensure that the vehicle is in a steady-state before descending, which in principle does not make the process completely autonomous. A disadvantage is that no safety checks are performed as the vehicle is descending towards the ground. The vehicle might therefore not accurately touch down on the designated landing mark. What is advantageous about this strategy, is that the vehicle is commanded to descend at a constant rate. This will ensure that the vehicle does touch down, even in the event of altitude measurement errors. This descend rate was also chosen in such a way that the aircraft can break through the IGE quickly enough, so that its effect is decreased.

A Master's student who graduated in the ESL also performed autonomous landing on a moving platform. Swart performed an autonomous landing with a model helicopter on a heaving platform [9]. In short, the heaving platform was characterized by algorithms executed on the helicopter and possible landing windows were identified into the future. These landing windows consisted of time slots where the heaving deck was near the maximum stationary part of the sine wave heaving behavior. When a window was identified, the helicopter would descend at a constant rate. This rate was again chosen in such a way as to negate the disturbance of the IGE on the helicopter.

Swart, Voos et al. and the stationary landing strategy currently incorporated on SLADe all use a commanded constant descent rate to land on a platform. This proves that descending at a constant rate is sufficient to touch down safely on the platform. No trajectory or profile generation, such as those used by Saripalli et al. and Wu et al., will thus be considered for descending onto the platform.

The state machine that will be used in this project can now be better defined from the choices made through this section of the literature study. The state machine will guide the vehicle from an arbitrary location to land on a platform that is moving horizontally with a constant velocity. The state machine will perform safety checks throughout the entire landing process, ensuring that the aircraft is tracking the reference given to achieve the required landing accuracy. The state machine will use safety checks to determine whether a state should be redone or whether the aircraft must abort the landing procedure entirely.

The state machine will have a state where the vehicle will firstly track an offset from the designated landing location. When stable and transients have diminished, the aircraft will be guided to directly above the landing location and finally descend at a constant rate until touchdown. The landing strategy will be explained in detail in Chapter 3

The second part of the literature study concentrates on the different sensing strategies or methods that were found in the literature to measure the states of a platform.

### 1.3.2 Sensor Configurations For Automated Landings

To enable autonomous landing of a UAV on a moving platform, the flight control system requires real-time knowledge of the positions, velocities and attitudes of both the vehicle and the platform. The pre-existing flight control system on SLADe already includes a kinematics estimator which it uses to estimate the vehicle states from on-board sensor measurements. The focus of this subsection is therefore to investigate sensor configurations and algorithms for sensing and/or estimating the states of the platform.

Several vision-based autonomous landing systems were found. Garret et al., Wenzel et al., Saripalli et al., Swart and Herisse et al., to name but a few, all use cameras and landmarks to determine relative states between the platform and the UAV. The differences between these vision-based systems are the landmarks used to track the landing location and the

techniques used to estimate the relative states from the camera frames. Vision systems are used to greatly increase the number of landing possibilities, not requiring a platform to be instrumented beforehand. Some of these vision-based system are looked at more closely.

Garret et al. use a sensor configuration for autonomously landing a model helicopter on a heaving, pitching and rolling platform [10]. The configuration requires no instrumentation of the platform, except for a narrowband light source positioned in the middle, which acts as a landmark. This system can measure the relative position of the platform from the aircraft and determine the orientation of the platform every  $40\text{ ms}$ . An on-board downward-facing camera system provides the relative position measurements, and a Light Detection And Ranging (LiDAR) aided system determines the orientation of the platform. Flight tests were conducted where the UAV hovered above a heaving, pitching and rolling platform, estimating the platform's states with success.

This system will not always be functional in this project as it will only work when the UAV and the platform are in close proximity to each other. In this project, the quadrotor will approach the platform from an arbitrary distance, which could be too far for the vision system to operate properly. An option would be to outfit the platform with a GPS sensor to guide the aircraft to the platform until the vision system has the platform in sight. This will require instrumentation of the platform, and sending the platform's position to the aircraft through a wireless data link.

Swart uses a sensor configuration encapsulating GPS and monocular vision to autonomously land on a heaving, pitching and rolling platform. The GPS will guide the UAV until the platform is in sight of the on-board downward-facing camera. The position and attitude states of the platform relative to the helicopter are then accurately estimated from a known landmark by the vision system.

Wenzel et al. used a forward-facing vision based system to estimate the position of a translating platform relative to a small quadrotor [4]. A camera system such as this one seems applicable to this project, possibly being able to detect the platform from a greater horizontal distance. The platform is not instrumented, except for four infrared Light-Emitting Diodes (LEDs). Successful flight tests were performed with the quadrotor autonomously landing on a translating platform.

However, the flight tests performed by Wenzel et al. were indoors and at low speeds ( $40\text{ cm/s}$ ). Infrared LEDs cannot be used as landmarks in daylight, and the platform used in this project will be translating considerably faster ( $30\text{ km/h}$ ). If the initial speed difference between the platform and the vehicle is large it can lead to an extensive distance between them, making it difficult for a visual system to estimate states. According to Hu et al., detecting targets with forward-facing cameras can be challenging, with this technique commonly being applied using infrared cameras [11]. Forward Looking Infrared (FLIR) Thermal Imaging is used for target detection in military and civil applications. Lobo et al., however, state that the tracking of targets in FLIR sequences is a hard problem because of the variability of the appearance of targets due to atmospheric conditions [12].

Furthermore, bearing in mind that this project is aimed at landing autonomously out at sea, weather conditions can have fatal effects on vision systems. Fog and rain can cause zero-visibility conditions, rendering a vision system ineffective. Non-vision based sensor configurations are explored.

A solution is proposed by Pervan et al. for automated shipboard landings in zero-visibility conditions [13]. They propose the use of a Carrier-Phase Differential Global Position System (DGPS). This DGPS provides accurate relative positional measurements of the platform relative to the aircraft. In 2009 Boskovic et al. proposed a landing system for autonomously landing a fixed-wing aircraft on an aircraft carrier using a similar system, called the Ship Relative Global Positioning System (SRGPS). In 2013 a Northrop Grumman X-47B UAV

successfully performed the first ever automated landing on an aircraft carrier using this system [14].

A similar system was used by Hardesty et al., who were part of Boeing's Unmanned Little Bird programme [15]. In 2006 the Unmanned Little Bird program saw more than 100 successful autonomous translational platform and six autonomous ship-deck landings of a full-size helicopter [15]. Their sensor configuration comprised mostly of Novatel's SPAN system, which is Novatel's solution for continual 3D positioning, velocity and attitude determination, even when satellite reception may be compromised for short periods of time [16].

The SPAN system by Novatel comprises two subsystems, an Inertial Navigation System (INS) and a Global Navigation Satellite System (GNSS). The GNSS used in the Unmanned Little Bird program provided centimeter-level accuracy relative position measurements between a GPS sensor on the unmanned helicopter and a sensor on the platform. The mode in which this GNSS works is dubbed Novatel Align<sup>TM</sup>, which is their solution for precise relative positioning. The method in which this GNSS functions meets all the needs of this project.

To accurately describe a translating platform, its latitude, longitude, height, speed and heading must be known. If using Novatel Align<sup>TM</sup>, the relative latitude, longitude and height of the platform to the aircraft will be known, with centimeter-level accuracy. The speed of the platform and its heading can also be provided by the sensor mounted on the platform.

The disadvantage of this system is that it is reliant on a wireless data link between the sensor on the aircraft and the platform. If this link fails, the relative measurements will not be available. This system does, however, not have the limitation of a vision-based system, being able to function in zero-visibility conditions, and to provide accurate position measurements far from the platform.

A system capable of this functionality is available in the ESL, and was decided to be used for this project. A wireless data link already exists between the quadrotor and the base station. The base station will be mounted to the platform, providing the ability to send information to and from the vehicle. The third part of the literature study focuses on mathematical quadrotor models.

### 1.3.3 Mathematical Models

This section concludes the literature study by reviewing mathematical models of quadrotors and aspects relevant to automated landing. The models are needed to design a control system and to simulate the automated landing strategy. The higher the fidelity of these models, the more confidence is gained that the automated landing procedure will progress as simulated. Models of quadrotors will firstly be investigated, followed by a review of models describing an effect relevant to this project, the IGE.

A mathematical model of the aircraft is needed to perform control system design and simulation. The higher the fidelity of the model, the more confidence is gained that the aircraft will behave as it did in simulation. Intricate models can, however, make it difficult to theoretically design a control system. The goal is thus to find a model that sufficiently describes the quadrotor, making a trade-off between the complexity of the model and practicality for control design.

The quadrotor model describes the forces and moments acting on the aircraft due to interaction with its environment. These forces and moments originate from three sources: gravity, actuators and aerodynamics. A review is done of models that have been used by others and the pre-existing model used for the SLADe project to describe these forces and moments.

The force gravity exerts on the aircraft is modeled similarly for all quadrotor models encountered during the literature study, where gravity is defined as a point force acting on the center of mass of the aircraft. Not surprisingly, the mathematical model of SLADe models gravity in the same fashion. The gravitational model will thus be maintained.

A model of the forces that originate from the actuators of a quadrotor is investigated next. Quadrotors use four actuators, each consisting of an electric motor mechanically connected to a propeller. Each actuator produces a force and a moment that are used by the control system to move the vehicle in a controlled manner. Balas states that it should be determined whether the lag dynamics of the motor and propeller combination can be neglected or whether they should be incorporated into the model [17].

Rotor lag dynamics describe the delay between when a thrust value is commanded for one of the motors to when the motor has actually achieved the new thrust value. This lag manifests due to the inertia of the rotor and the electrical characteristics of the motor driving the rotor. The rotor lag dynamics were ignored by Powers et al. in the modeling of a small quadrotor, with the aircraft successfully flying [18]. This was not the case of the model used by Bresciani, where the electrical and mechanical lag dynamics of the motor and rotor combination were modeled in detail and incorporated into the design [19].

Not modeling the rotor lag dynamics could misrepresent the performance and practicality of the control system that was designed from using the model. This was noted by Balas when simulation results were favorable, but the vehicle became completely unstable during practical tests. It is thus better to investigate and model the rotor lag dynamics.

SLADe does indeed have a rotor lag model for simulation and control system design. The model consists of a first-order differential equation, describing the lag dynamics of the rotors. The time constant for the rotor model was determined experimentally and has proven to adequately describe the rotor dynamics in simulation. The rotor model will be maintained as it has proven to be effective from previous work.

Models of aerodynamic effects on quadrotors are investigated and complete the review of quadrotor models. Aerodynamic modeling is the area in which some models are much more advanced than others. The model presented by Garcia Carrillo et al. has no modeled aerodynamics, being a minimalistic model that describes the aircraft sufficiently for near-hover control [20]. This approach will be effective in instances where the aircraft will be hovering stationarily or translating at very low speeds. This project, however, requires the vehicle to translate at speeds in excess of  $30 \text{ km/h}$ . This model will not describe the aircraft aerodynamics sufficiently.

A model presented by Elruby et al. incorporates simple drag theory by approximating the vehicle as a flat plate, with an effective frontal area, moving through the air [21]. The frontal area is different for each translational axis, better describing the drag induced during forward or vertical flight. The model that has been used for the SLADe project is similar, describing the drag experienced by a flat plate with an effective frontal area and drag coefficient.

The current aerodynamic model used in the SLADe project has been used to design and simulate a control system that successfully saw the aircraft translating at a speed of  $79 \text{ km/h}$ . The current aerodynamic model for SLADe is therefore sufficient to describe its motion adequately for the speeds that will be attained during this project. The need for a more accurate model may not be necessary.

It is thus established that no changes will be made to the mathematical model of SLADe, as it appears to be sufficient for control system design and simulation for this project. The investigation is nevertheless being continued, leaving room for other effects which might become apparent during the course of the project.

Bristeau et al. performed extensive aerodynamic modeling of the rotor blades, incorporating flapping dynamics of the rotors [22]. By using blade element theory, formulas were obtained

for the forces and moments acting on the center of each rotor disk. This requires accurate determination of the flexibility of the rotors, since it may be easy to obtain misleading results otherwise. Rotor-flapping models will, however, not be considered, unless this becomes an apparent phenomenon during the project.

Powers et al. performed modeling on determining the thrust obtained from the rotor as a function of the velocity induced by the rotor on the air, the relative velocity between the rotor and the surrounding air, and the angle of attack [18]. It was noted that the thrust produced by the rotor decreases notably with increased wind speed in a given direction. The mathematics describing this thrust requires experimental determination of parameters and coefficients on a test rig. Their model was coherent with the test rig, but not with results obtained in free flight. Similar behavior has been noted with SLADe from past experiments. However, the behavior was not characterized as the simulation model approximates the load on the rotors sufficiently.

Attention is now shifted towards other aerodynamic effects that play a role while landing a rotorcraft. An aerodynamic effect that is relevant to automated landings is the IGE. According to Schmaus et al., IGE is a term used to describe the changes in performance that rotorcraft and fixed-wing aircraft experience as they approach the ground [23]. Schmaus et al. further state that the IGE is often viewed as an increase in thrust for a constant power, which can be beneficial in some applications, but not for an automated landing where the aircraft dynamics are changing. When a rotorcraft is subject to IGE conditions, the wake of a helicopter rotor interacts with the ground and causes significant perturbation to the flow near the rotor blades, as well as the rest of the craft [24]. IGE therefore effectively disturbs the aircraft, which is not favorable while the aircraft is translating near the platform.

The IGE, to its full extent, is not applicable to this project. The platform which is available in the ESL, and which the aircraft will be landing on, is perforated. The perforated surface allows the wake from the rotors to partially pass through the platform. The effect from coming close to the platform is therefore decreased, but not completely absent. As a consequence, it might be worthwhile to investigate this phenomena separately.

A mathematical IGE model for a helicopter is proposed by Johnson [25]. The equation which describes the lift that is obtained when the helicopter is less than a rotor length above the ground, indicates that more lift is generated here than when the helicopter is far from the ground. Powers et al. investigated this model, specifically using a quadrotor, and noted that although it follows the same trend, the distance at which the equation becomes valid is different. Through experiments, Powers et al. observed that the IGE became apparent at a distance of 5 rotor lengths above the ground [18], which in the case of SLADe is 2.5 m.

Throughout the literature study, a model could not be obtained for the IGE for a quadrotor above a perforated platform. Experiments will therefore be performed to determine whether this effect is prominent and if it can be characterized. From the research done by Powers et al., an indication is given of the altitude at which IGE becomes apparent. This will be closely monitored.

In closing this section, the modeling decisions made are summarized. The current mathematical model for SLADe will be maintained. This model has proven in the past to sufficiently describe the aircraft while flying at speeds that will be seen in this project. An effect that will nonetheless be closely monitored is the IGE, which will be observed during practical flight tests.

## 1.4 Project Overview

This section briefly exemplifies the work carried out and the events that took place from the commencement of the project up until the project goal was achieved. The project in-

volved theoretical and practical work, like control system design, writing firmware, hardware integration and performing goal orientated flight tests.

A review of research and development that was done by others on automated landing and quadrotors was the logical first step. In tandem with the investigation, intense study and documentation of the abilities, control system architecture and firmware structure of SLADe were performed.

Control system analysis then started, to determine whether the control system of the aircraft is adequate to achieve the project goal. This was done by conducting linear and non-linear software-only simulations. It was ascertained that the control system required modification, as the control law was not functional for the project and the bandwidth of the system was too low to adequately track a moving platform. The horizontal control system of the quadrotor was partially re-designed and the horizontal velocity control law was augmented with velocity feed-forward.

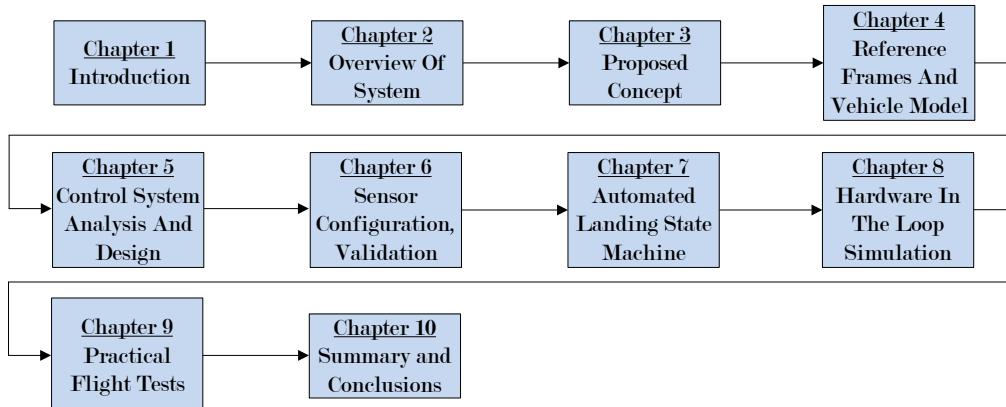
A period of modifying and testing of the Novatel DGPS, which plays a key role in providing measurements for the automated landing, were done while control system design was taking place. Integrating the Novatel Align<sup>TM</sup> feature required software modifications to the firmware of the quadrotor and the Ground Station. Hardware challenges were also presented, which were solved by implementing a high-speed wireless link in the Novatel DGPS.

Hardware-In-the-Loop (HIL) simulations were performed in preparation of flight tests, while a trailer was instrumented and tested for use as the translating platform. This involved more practical testing of the sensor configuration used for the automated landing procedure.

Flight testing started after the Novatel Align<sup>TM</sup> feature was functional. A series of tests were performed with each test trialling a new addition to the aircraft or the automated landing strategy. The project goal was attained during the last flight test.

## 1.5 Thesis Layout

The layout of the thesis is presented in this section. The chapters are not necessarily in the chronological order of how the project unfolded, but are appropriately grouped to communicate information effectively. Refer to Figure 1.3 for a diagrammatic summary of the thesis outline.



**Figure 1.3 – Thesis Layout**

Chapter 1 gave the project background and motivated the need for the project goal. A detailed definition of the project goal followed and relevant parameters were quantified. A literature review was then presented, followed by an overview of the project plan.

Chapter 2 will present an overview of the quadrotor UAV system as it stood when the project commenced. The aircraft and Ground Station hardware are reviewed, supported by a broad view of the structure of the avionics and the Ground Station. The control system structure on SLADe is also introduced on a very high level.

Chapter 3 presents the proposed concept for autonomous landing on a moving platform. The states that will guide the vehicle through the automated landing procedure are proposed, followed by the recommended sensor configuration to allow the aircraft to land itself on a moving platform. The chapter closes by proposing how the final flight demonstration test will be executed and coordinated, as extensive planning is needed.

Chapter 4 defines the reference frames and formulates the equations of motion for the vehicle and the platform. The axis systems used throughout the thesis are defined first, followed by definition of the kinetic and kinematic relations. The forces and moments that originate from aerodynamic effects and actuators of a quadrotor aircraft are then presented, followed lastly by the mathematical models used to simulate sensor noise and wind in the project.

Chapter 5 presents the analysis and evaluation of the pre-existing flight control system, presents the upgrades and re-designs required to enable autonomous landing on a moving platform, and presents the results of the Software-In-the-Loop (SIL) simulations. The entire flight control system is analyzed to expose possible areas in need of modification and improvement for the project. Relevant flight control systems are then re-designed and thoroughly tested by performing non-linear SIL simulations.

Chapter 6 presents the pre-existing sensor hardware and software used to estimate the vehicle states. The hardware and software modifications that were done to enable the system to automatically land on a moving platform are then presented, finally followed by the practical tests that were performed to verify that they meet the requirements.

Chapter 7 presents the design of the automated landing state machine, which guides the vehicle through the landing procedure. The states of the proposed state machine are explained for a general autolanding on a moving platform. The autolanding state machine that will specifically be used for the flight demonstration test is then presented. The chapter closes by introducing the user interface that was designed to manage the state machine.

Chapter 8 presents the Hardware-In-the-Loop (HIL) simulation execution and results. The pre-existing HIL setup is introduced, followed by the modifications that were needed to perform HIL simulation of automated landings on a moving platform. The simulation results are finally presented and discussed.

Chapter 9 presents the practical flight test campaign and the analysis of the flight test data. In total five flight tests were conducted, ending with an automated landing on a moving platform.

Chapter 10 presents the summary of the research performed during the course of the thesis and makes recommendations for future work on the project.

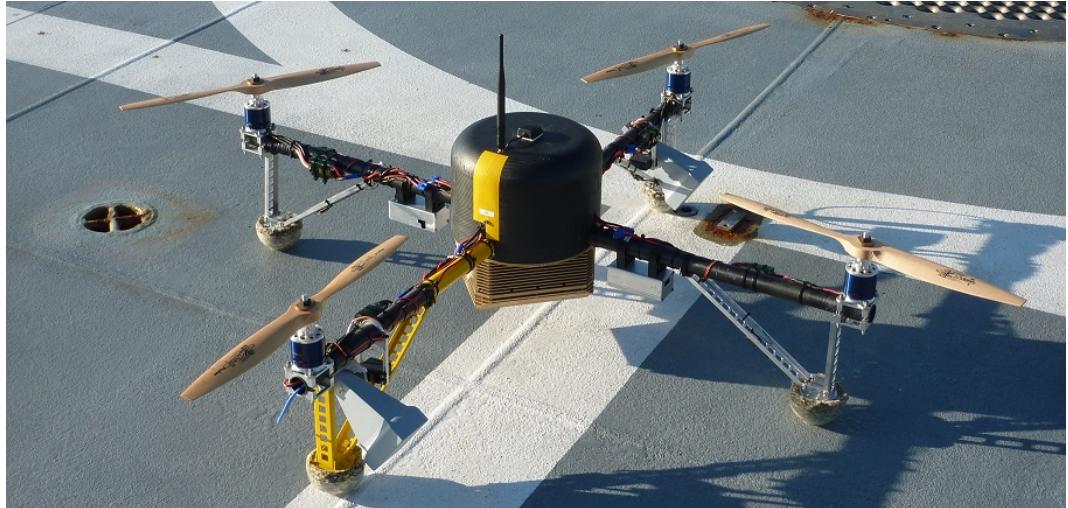
## Chapter 2

# Overview of Quadrotor Unmanned Aerial System

An overview of the unmanned aerial system used for the project is presented in this chapter. The chapter begins by introducing the three constituents of the system. The system is comprised of the aircraft, the Ground Station and the Safety Pilot. Each constituent is briefly reviewed, followed by a more detailed presentation of the aircraft and its capabilities. Lower-level details like the hardware and software of the aircraft and the Ground Station are then discussed. The chapter finishes by presenting, on a high level, the existing flight control system of the quadrotor.

### 2.1 The Unmanned Aerial System

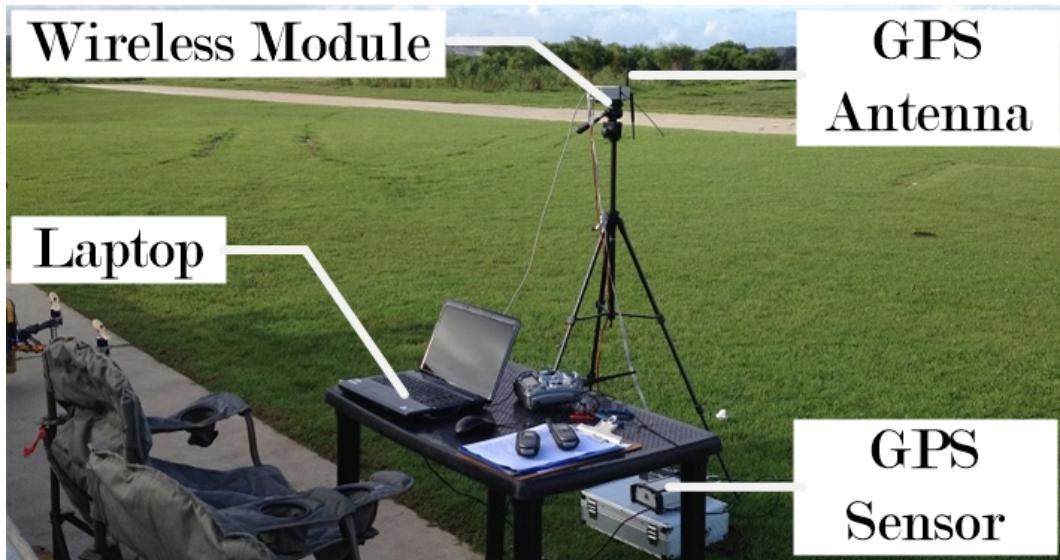
The aircraft used in this project is known as a quadrotor and can be seen in Figure 2.1. A quadrotor has four rotors as actuators, each connected to a motor. This makes the vehicle mechanically simpler than a conventional helicopter. By varying the speed of each rotor in a controlled manner, the vehicle can be made to fly.



**Figure 2.1 – The Quadrotor**

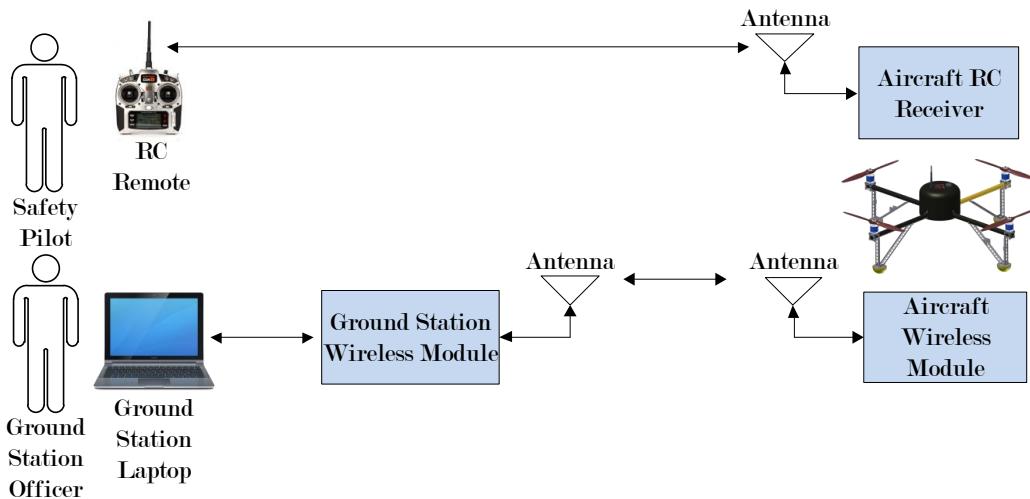
This quadrotor is considered, in terms of payload, to be large by today's standards, weighing 13 kg with its payload. The quadrotor has a motor-to-motor distance across the center of 1185 mm. A wireless module for communicating with the Ground Station is mounted on the aircraft.

The Ground Station is manned by the Ground Station Officer, who sends commands to the vehicle to perform autonomous missions and who monitors telemetry data from the aircraft. The Ground Station Officer uses a laptop with a wireless module for communication with the quadrotor. Refer to Figure 2.2 for an image of the ground station setup.



**Figure 2.2 – The Ground Station**

The Safety Pilot is the final constituent of the system and is always present at flight tests. A standard Radio Control (RC) remote is used by the Safety Pilot to control the aircraft when not in autonomous flight. The aircraft can therefore be controlled by the Ground Station Officer or the Safety Pilot. The command architecture of the unmanned aerial system is illustrated in Figure 2.3.

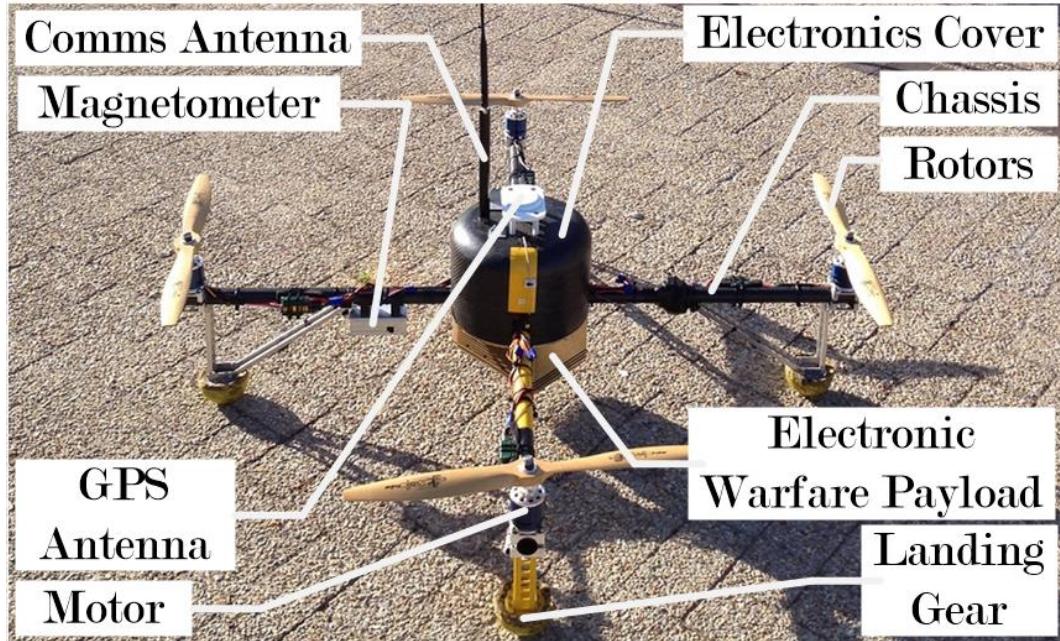


**Figure 2.3 – Command Structure**

The Safety Pilot also serves as a countermeasure against catastrophic failure, with authority to take control of the aircraft at will if unexpected behavior is observed during autonomous flight.

## 2.2 The Quadrotor

A more detailed look is now taken at the quadrotor, describing the aircraft and its abilities in more detail. Design of the quadrotor started in early 2009 with assembly commencing near the end of the same year. The quadrotor, with annotations of the components, can be seen in Figure 2.4.



**Figure 2.4 – Quadrotor Details**

The chassis base plates, arms and electronics cover are made from woven carbon fiber. The landing gear as well as all frame fixtures are made from aluminum. This mechanical design ensure rigidity and saves weight. The part of the landing gear that comes into contact with the ground is connected to the aluminum struts by a dense foam, which is protected by plastic hubs. Impact from touching down on the ground is damped by the dense foam.

The electronic systems on the aircraft, called the avionics, are situated in and above the chassis base plates, and are protected by the carbon fiber cover while in flight. The four motors are brushless Direct Current (DC) motors, each with a maximum power output of 2 kW. Every motor is driven by its own Electronic Speed Control (ESC), which is commanded by the avionics to adjust the motor speed.

Power is provided to the motors and all other on-board electronics by lithium polymer batteries. The batteries can deliver high currents and are exceptionally energy-dense, containing as much as 505 joules of energy per gram of weight [26]. Eight batteries in total are used to power the motors, with each motor powered by two batteries connected in series. The avionics are powered by a small single battery. More technical details of the power system components on the vehicle are exhibited in Table A.1 in Appendix A.

The power system of the quadrotor enables it to fly for approximately 4 minutes while carrying the payload. The aircraft has exceptional payload carrying capacity, being able to carry almost 100% of its own weight. The payload that the aircraft carries weighs approximately  $6\text{ kg}$ . The aircraft is able to safely fly in wind gusts of up to  $7\text{ m/s}$  and has attained a maximum airspeed of  $22\text{ m/s}$  from previous flight envelope testing.

Throughout development of the quadrotor, its autonomous abilities were expanded. Four autonomous missions can be performed: Loitering, Automated Take-Off and Landing (ATOL), Decoy and Return Path Tracking (RPT).

Loitering is the most basic, which is appropriately referred to as hover mode. In this mode the aircraft will hold its position, fighting against any disturbances. From the Ground Station the aircraft can be commanded to move a user-defined distance forward, back, left, right, up or down. Similarly, its yaw angle can be changed to a user-defined heading.

ATOL can be performed from a stationary landing pad, with the location of the landing pad defined as the location where the aircraft is launched. The Decoy is the string of actions that are performed while the vehicle guides the missile away from the ship. RPT is a string of actions that returns the aircraft safely to the base after performing the Decoy. The hardware and software that enable the aircraft to perform these tasks autonomously are reviewed next.

## 2.3 Unmanned Aerial System Hardware And Software

This section provides more detail on the existing system, elaborating on the hardware and software architecture of the aircraft and the Ground Station. The hardware components of both subsystems will be introduced, followed by illustrations of how these components are interfaced with each other.

### 2.3.1 Quadrotor Hardware And Software

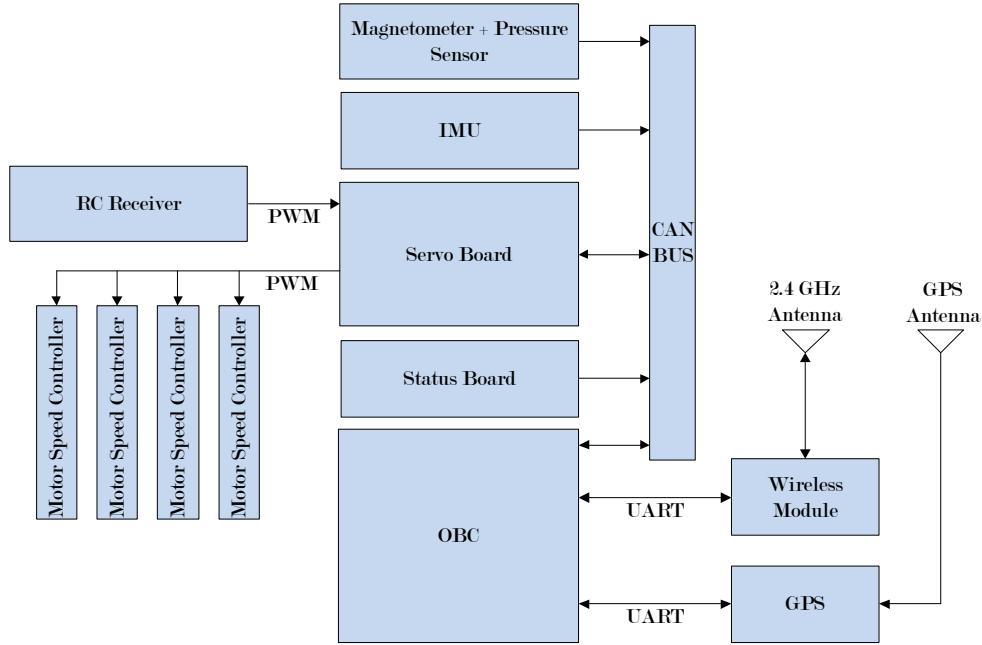
The quadrotor is equipped with an avionics package. This package consists of sensor, processing and communication hardware. The sensor hardware provides measurements that are used by the navigation and control algorithms to perform a mission. The processing hardware is responsible for executing the algorithms and other supporting functions. The communication hardware is responsible for sending information between all the components of the avionics as well as between the aircraft and the Ground Station. The avionics architecture is illustrated by a block diagram in Figure 2.5, which may be referred to as the hardware components are introduced.

#### 2.3.1.1 Sensors

To be able to control the aircraft, the control system requires measurements or an estimation of the aircraft's position, velocity, orientation, acceleration and angular rates. Position and velocity measurements are obtained from a GPS, while the orientation is determined from a combination of measurements from gyroscopes, accelerometers and a magnetometer. Other miscellaneous sensors provide information on the current state of all the batteries on the aircraft. Each sensor unit is briefly explained.

##### Inertial Measurement Unit (IMU)

The IMU is a piece of hardware that is equipped with a 3-axis gyroscope and 3-axis accelerometer. The gyroscopes provide measurements of the rate of angular change around all three rotation axes of the aircraft. The 3-axis accelerometer provides specific force measurements along all three translational axes. The IMU is a sensor that can provide high frequency updates, being utilized in the quadrotor to provide measurements at  $50\text{ Hz}$ . Refer to Table A.2 in Appendix A for more technical information regarding the IMU.



**Figure 2.5 – Avionics Architecture**

### Global Positioning System (GPS)

GPS is used to measure the position and velocity of the vehicle. The GPS does not, however, provide measurements at a high rate like the IMU, providing measurements instead at a rate of 10 Hz. Refer to Table A.3 in Appendix A for details on the GPS sensor and antenna used on the aircraft.

### Magnetometer

The magnetometer is used to measure the vehicle's orientation relative to the earth's magnetic field. An algorithm (Kinematics State Estimator) on the aircraft uses a combination of measurements from the magnetometer, accelerometers and gyroscopes to estimate the orientation of the vehicle relative to the earth. The unit also has a barometric sensor which is useful for measuring the change in atmospheric pressure as the height of the aircraft changes. The pressure sensor is not used, and altitude information is obtained from the GPS measurements only. Like the IMU, the magnetometer provides measurements at 50 Hz.

### Status Sensors

The status sensors are all located on a single unit called the Status Board. The Status Board is outfitted with sensors for measuring currents and voltages. The current sensors measure the current draw of each battery connected to a motor, while the voltage sensors measure the voltage.

#### 2.3.1.2 Processing

The processing hardware situated on the aircraft is called the On-Board Computer (OBC). The OBC contains two processors, each capable of 30 million instructions per second. The primary processor carries most of the processing load and runs the firmware that contains the navigation, control and all other supporting algorithms.

The secondary processor handles GPS packet processing, extracting the GPS measurements from the raw data obtained from the GPS sensor. Logging of measurements and other

important parameters are also done on the OBC, with each parameter being logged at 25 Hz.

### 2.3.1.3 Communication

The communication hardware consists of a wireless link, a unit called the Servo Board and a Controller Area Network (CAN) bus.

The wireless module is connected to both processors on the OBC and operates on a 2.4 GHz band. All instructions are sent to the vehicle and all telemetry is sent by the vehicle to the Ground Station through this link.

The CAN bus is used as the communication interface between the sensors and the OBC, sending measurements and parameters between all the avionics.

The Servo Board is connected to the Safety Pilot's RC receiver and the speed controllers of the motors. The RC receiver is wirelessly connected to the RC remote of the Safety Pilot. The Servo Board receives Pulse Width Modulation (PWM) signals from the RC receiver and converts the analog PWM values to digital for use on the processors. The digital values are then transported from the Servo Board to the OBC by the CAN bus. Commands from the OBC are converted from digital values to analog PWM signals by the Servo board and are sent to the speed controllers of the motors.

### 2.3.2 Ground Station Hardware and Software

The Ground Station is composed of fewer hardware components than the aircraft. Its hardware can, however, also be divided into processing, sensing and communication types. The Ground Station hardware architecture and data flow between the hardware components can be seen in Figure 2.6.

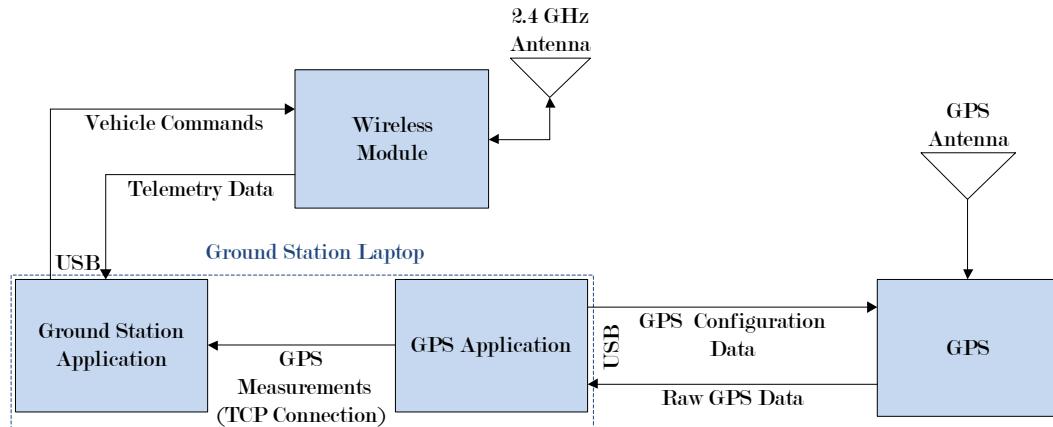


Figure 2.6 – Ground Station Architecture

#### 2.3.2.1 Sensors

Like the aircraft, the Ground Station is equipped with a GPS sensor. This sensor provides position and velocity measurements to the Ground Station. The measurements are primarily used to let the aircraft know where the Ground Station is located. This is done by sending the measurements to the aircraft through the wireless link between the aircraft and the

Ground Station. Refer to Table A.3 in Appendix A for details on the GPS sensor and GPS antenna used for the Ground Station.

### 2.3.2.2 Processing

Processing is performed on a laptop, which runs two applications making up the Ground Station. The applications are appropriately named the Ground Station Application (GSA) and the GPS Application.

The function of the GSA is to send commands from the Ground Station Officer to the aircraft and display important telemetry that the Ground Station Officer must monitor. The GSA is written in Qt Creator, which is a freeware software package created by Nokia. This software package simplifies the process of adding a graphical user interface to the firmware necessary to make the Ground Station function. The language that is used by Qt Creator is a variation of C++.

A screenshot of the GSA can be seen in Figure 2.7. The screenshot shows the existing command interface for ATOL on a stationary landing pad. The command interface allows state transition telecommands to be issued to the vehicle, control parameters to be uploaded, and individual control loops to be enabled and disabled.

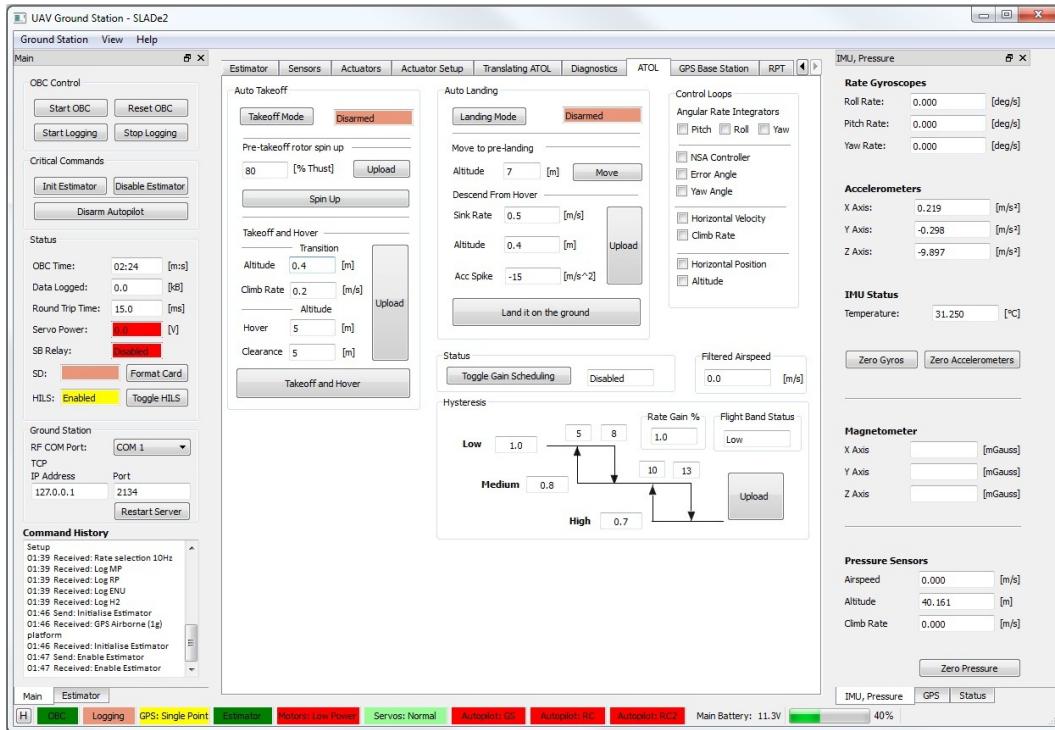


Figure 2.7 – Ground Station Application Screenshot

The GPS Application is used to initialize and configure the Ground Station GPS sensor for operation. It will extract relevant measurements from the raw data from the GPS sensor and send it to the GSA. The GPS Application is written in Borland, which is similar to Qt Creator, in that it uses a variation of C++ and simplifies adding a graphical user interface.

### 2.3.2.3 Communication

The Ground Station is outfitted with the same wireless module as the aircraft. This module offers bidirectional data communication between the aircraft and the Ground Station.

Bidirectional data communication also takes place between the GPS Application on the laptop and the GPS sensor. The GPS Application sends data to the GPS sensor to configure it for operation and receives raw data from the GPS sensor, out of which the GPS Application extracts the measurements.

Finally, the GPS Application and the GSA are connected in terms of software by a Transmission Control Protocol (TCP) connection on the laptop. Through the TCP connection the GPS Application sends GPS measurements to the GSA. The GSA can then send the base's GPS measurements to the aircraft through the wireless link if needed.

### 2.3.3 Differential GPS

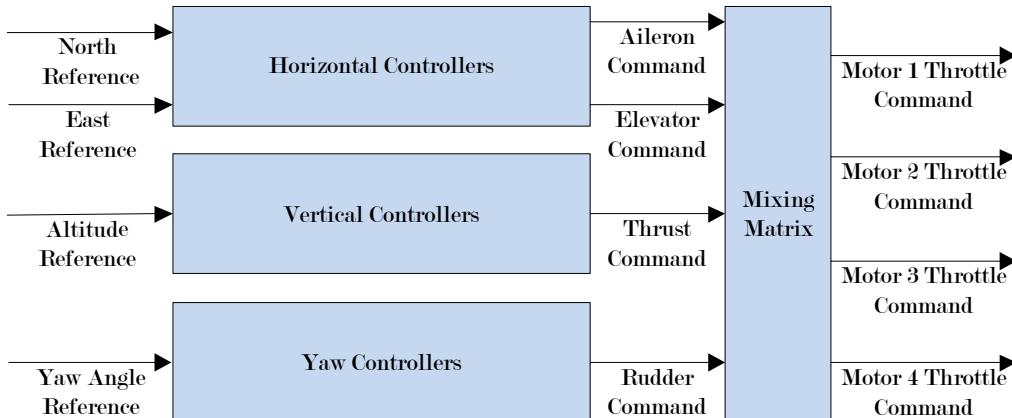
Differential GPS (DGPS) is a method of enhancing the position and velocity measurement accuracy of a normal GPS. The hardware must, however, be capable of DGPS operation. The GPS sensors used for the aircraft and the Ground Station in this project do have this capability.

Both of the sensors are manufactured by Novatel, who offers different DGPS modes of operation. The existing flight control system uses the Novatel system in Real-Time-Kinematic (RTK) mode.

This mode is able to provide position measurements with an accuracy of 20 mm to the sensor on the aircraft, provided that the sensor of the Ground Station remains stationary. This involves sending correction data from the Ground Station sensor to the aircraft sensor through the wireless link. A more in-depth explanation of the Novatel DGPS is presented in Chapter 3.

## 2.4 Quadrotor Control System

The chapter closes with a brief overview of the control system on the quadrotor. The control system is a suite of decoupled control subsystems which are each responsible for controlling a different axis of the aircraft. Each axis has a combination of proportional, integral and differential control laws in consecutive loops to finally control the position of the aircraft in space. A high-level block diagram of the control system can be seen in Figure 2.8.



**Figure 2.8 – High-Level Control System Overview**

On the left side of the diagram, the references can be seen. A north, east and altitude reference can be given to the aircraft so that it changes its position in space. The horizontal control subsystem is responsible for controlling the horizontal position (north and east) of the aircraft, while the vertical control subsystem is solely responsible for controlling the altitude of the vehicle. The yaw control subsystem accepts a yaw angle reference and changes the heading of the vehicle appropriately.

The high-level outputs of the control subsystems are virtual aileron, elevator, thrust and rudder commands. These commands are called virtual commands, as the vehicle does not physically have any control surfaces for actuation. However, the four motors can be actuated in a way that they appear to have the same effect on the quadrotor as ailerons, elevators and a rudder have on a fixed wing aircraft.

The virtual elevator command will induce a pitching moment on the vehicle, the virtual aileron command a rolling moment and the virtual rudder command a yawing moment. Also, much like a normal rotary wing aircraft, the throttle command will affect the vehicle's vertical motion.

Converting the virtual actuator commands to individual actuator commands for each motor requires the use of a mixing matrix. The mixing matrix is determined according to the dynamics of the quadrotor, actuating different combinations of the four motors to achieve a pitching, rolling and yawing moment. A more in-depth control system review and analysis is presented in Chapter 5.

## Chapter 3

# Automatic Landing Strategy

In this chapter the strategy for achieving the project goal is presented. The first section elaborates on the overall concept of the landing strategy, discussing the procedure through which the aircraft will be guided. The proposed sensor configuration for the landing strategy is then made known, briefly explaining what the benefits of this configuration are and what has to be done for its implementation. The execution procedure of the practical test that is done to demonstrate the automatic landing strategy is finally explained.

### 3.1 Proposed Concept for Autolanding on a Moving Platform

This section presents the proposed procedure for autolanding on the translating platform. After considering all the autolanding strategies that were investigated during the literature review, an autolanding procedure is proposed with the following states:

1. Tracking
2. Homing
3. Descending
4. Shutdown

Each of the proposed states is motivated and supported by a visual representation of the procedure in Figure 3.1.

In Figure 3.1 a block is annotated as 'Start Of Procedure'. This is assumed to be the initial condition of the autolanding process. The aircraft is hovering at an arbitrary height and horizontal location behind the platform. At this point, the platform is already translating at a constant velocity of  $V_{Platform}$ . The first state, the Tracking state, starts as soon as the aircraft is commanded to land by the Ground Station.

During the Tracking state, the aircraft will attempt to close the initial distance between the translating platform and itself. The quadrotor will therefore have to attain a higher speed than that of the platform. When the aircraft is near the platform, it will decelerate to match the platform's speed. Transients in the aircraft's movement will naturally arise from the deceleration. The transients need to be negated at a safe distance from the platform. An offset of the landing location is therefore firstly tracked during the Tracking state, a technique similar to what Wenzel et al used in their automated landing strategy. In Figure 3.1 this offset tracking location is defined as a distance  $X$  behind the translating platform and a height  $Z$  above the platform.

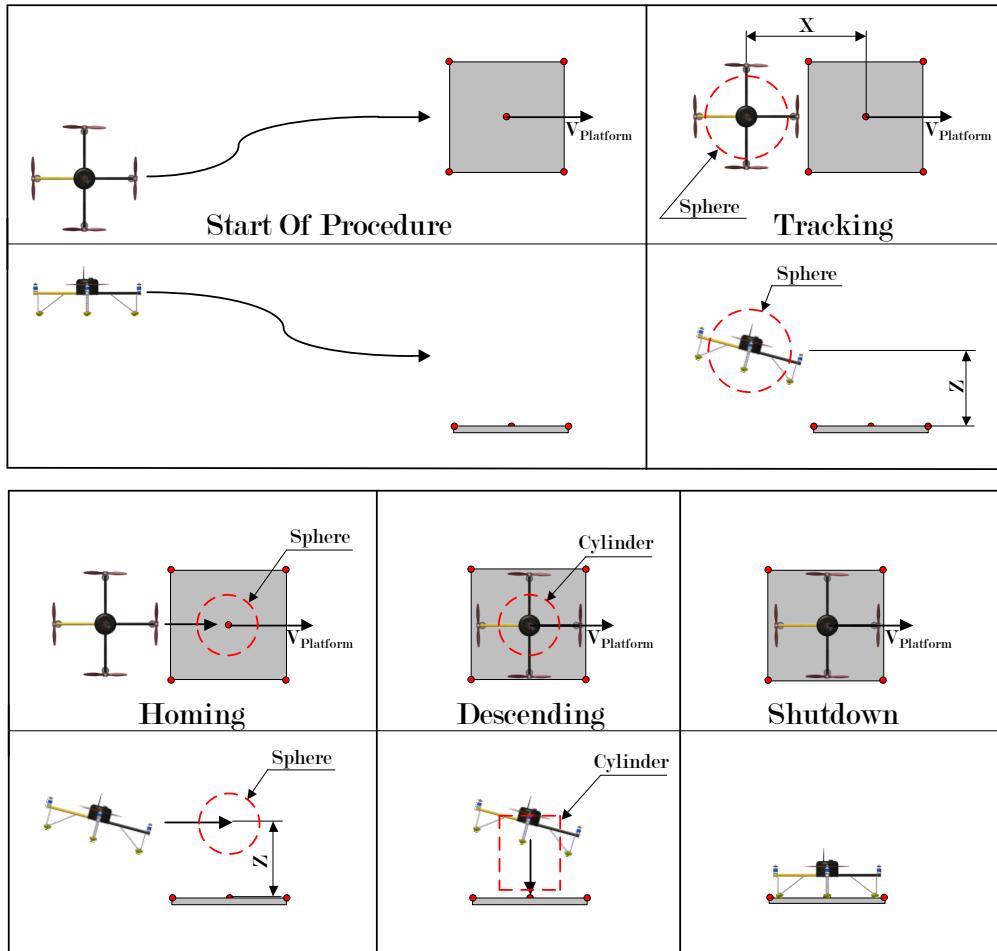


Figure 3.1 – The States Of The Landing Procedure

The criteria for advancing to the next state, the Homing state, is position-based as seen in Voos et al, and also time-based, as seen in Wenzel et al. The position error between the aircraft and the offset tracking location must continually remain below a predetermined absolute value for a specified time frame. Geometrically this translates to the aircraft remaining in a sphere with a predetermined radius for a continuous amount of time. This is a safety check that is done to ensure that the aircraft is tracking the platform with steady-state-like behavior before continuing with the landing process.

Once the aircraft has passed the safety check, the Homing state starts. The aircraft then advances slowly, until it has proceeded to a position directly above the landing location. The height of the aircraft above the platform will remain the same. The speed of the advancement will be slow relative to the platform, to keep transients from the acceleration and deceleration during the state to a minimum.

To advance to the next state, the Descending state, the same safety criteria are set as for the Tracking state. The aircraft must remain within a sphere for a continuous amount of time above the landing location. Once the safety test is passed, it can be assumed that the aircraft is again in a steady-state relative to the translating platform. The critical state then starts, namely the Descending state.

During the Descending state, the aircraft starts to descend at a constant rate. This is the vital state where the vehicle is not allowed to drift horizontally by more than 0.5 m from the landing location. A safety check is implemented, where the vehicle is not allowed to drift out of a horizontal circle with a 0.5 m radius while it is descending. This can geometrically be represented as a cylinder, as shown in Figure 3.1. If the aircraft drifts out of this cylinder during the descending state, the landing procedure will be aborted.

If, however, the aircraft remains within the cylinder while descending, it will touch down on the platform. When the quadrotor makes contact with the platform, a spike will be measured by the accelerometers of the avionics. This spike initiates the final state, namely the Shutdown state.

In the Shutdown state, the rotors of the aircraft will stop rotating and the control system will be disarmed. The autolanding procedure is then completed.

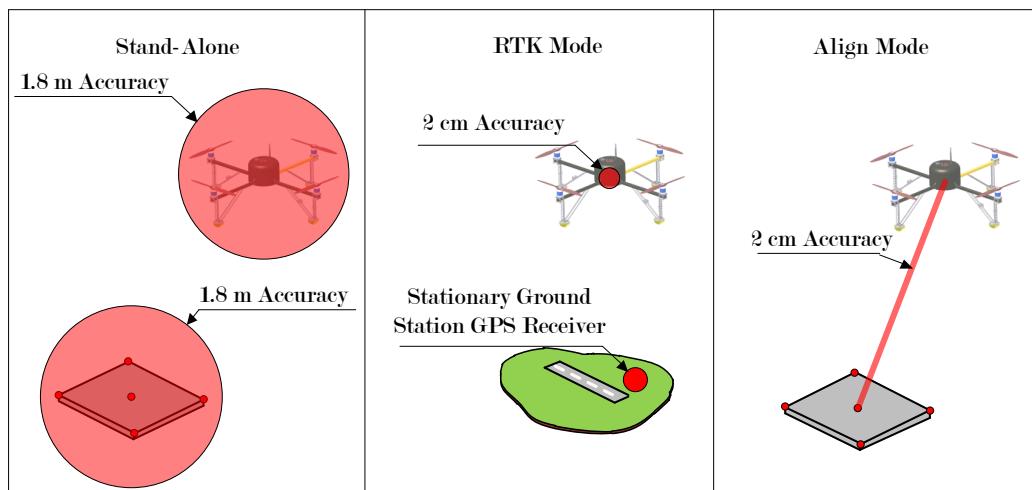
To practically implement the states of the autolanding process on the aircraft, a state machine is used. The state machine, which controls the states of the autolanding procedure, is presented in Chapter 7. Chapter 7 also explains the states in more detail. As with the state machine used by Saripalli et al, there will be recurring states to improve the safety of the aircraft and its surroundings, and to improve the chances of completing the landing process successfully. States will therefore be recurring if a safety check is failed during a state. However, if the state is a critical state, like the Descending state where impact is imminent with the platform, the autolanding process will be aborted if a safety check is failed.

### 3.2 Proposed Sensor Configuration

To land on a translating platform, the aircraft's flight control system requires real-time feedback of the states of the platform. Fully describing the movement of a platform translating at a constant speed requires measurement of its velocity and position. The position measurement must be accurate, as the aircraft must land within 0.5 m of the center of the platform.

During the literature review presented in Chapter 1, Novatel's Align<sup>TM</sup> solution was identified as a viable option for accurate measurement of the relative position between the platform and the aircraft. This requires two Novatel Global Positioning System (GPS) sensors to function as a Differential Global Positioning System (DGPS). Two Novatel sensors were fortunately available in the Electronic Systems Laboratory (ESL) and could be used for the project.

Novatel's sensors can function in different modes to provide the desired functionality or accuracy for different applications. Refer to Figure 3.2 where three different modes of operation are illustrated.



**Figure 3.2 – GPS Modes Of Operation (Not To Scale)**

The two Novatel sensors can operate as two stand-alone GPS units. A GPS sensor can be fixed to the platform and the platform's velocity and position measurements can then be sent to the aircraft. However, when used in this configuration the accuracy of position measurement for each unit is at best 1.8  $m$  Root Mean Square (RMS), according to Novatel. This accuracy is not sufficient for the project's requirements, as the quadrotor must land within 0.5  $m$  of the center of the platform.

The two Novatel sensors have been used previously for the SLADe project, although not functioning in Align<sup>TM</sup> mode. The existing flight control system uses the Novatel DGPS in Real-Time Kinematic (RTK) mode. This DGPS mode can provide an absolute position measurement accuracy of 2  $cm$  for the sensor fixed to the aircraft. RTK requires that the other GPS sensor, which is positioned at the Ground Station, be stationary, as it is used to generate correction and observation data of the GPS drift experienced in the region. This correction data is sent to the GPS sensor on the aircraft through the existing wireless link, enabling the aircraft to know where it is on the earth within 2  $cm$ .

The Novatel Align<sup>TM</sup> feature provides accurate enough position measurements even if the two sensors are in motion. When operating in Align<sup>TM</sup> mode, the absolute position measurements of the vehicle and the platform experience almost identical GPS position drift. While the two sensors are communicating, they can provide accurate relative data, but 1.8  $m$  accurate absolute position measurements. Align<sup>TM</sup> provides a viable option for landing on a translating platform, as the relative position measurements between the aircraft and the platform will be accurate to within 2  $cm$ . The relative position measurements can then be used as the position error between the platform and the aircraft.

It must, however, be thoroughly examined whether the Novatel Align<sup>TM</sup> technology is capable and robust enough to provide credible relative position measurements. Several experiments will therefore need to be performed. This examination is done in Chapter 6. Software and hardware modifications that were needed to change the DGPS mode from RTK to Align<sup>TM</sup> are also presented in Chapter 6.

### 3.3 Flight Demonstration

This section closes the chapter and presents the setup and procedure of the flight test that will practically verify whether the aircraft can automatically land on a translating platform. The equipment needed is discussed first, followed by the sequence of events of the flight test.

The quadrotor, Ground Station, platform and Safety Pilot are the critical constituents of the flight test. The platform will be fixed to a trailer, which will be towed by a motor vehicle. The motor vehicle will accommodate the Ground Station equipment and the Ground Station Officer. This requires fitting a GPS antenna to the platform and fitting all communication equipment to the motor vehicle.

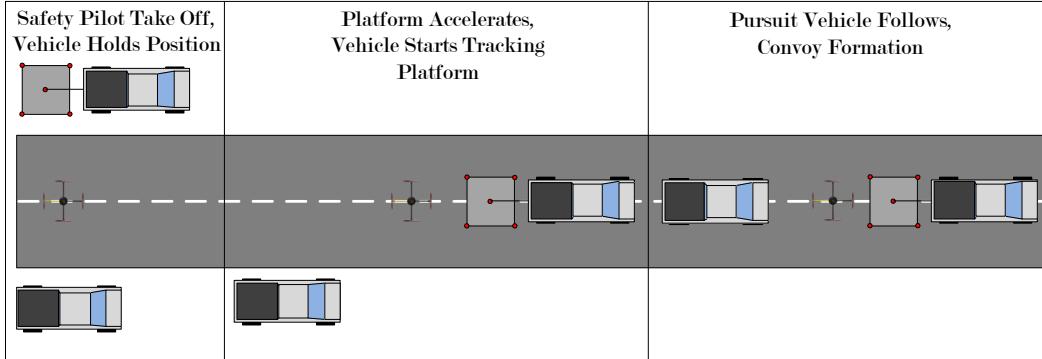
The Safety Pilot must be near the aircraft and have it in sight at all times. He must be able to hear the rotors of the quadrotor and be able to easily determine the orientation of the aircraft. From far a quadrotor can look very similar from all sides, making it difficult for a pilot to estimate its heading and thus control the aircraft. A second motor vehicle will therefore be needed to ensure that the Safety Pilot is near the aircraft during the entire test. This motor vehicle will be the pursuit vehicle.

The pursuit vehicle will accommodate the Safety Pilot and the Safety Officer. The Safety Officer will have a wireless voice connection with the Ground Station Officer, so that the status of the test can be communicated to the Safety Pilot.

It is proposed that the landing procedure be started with the vehicle stationary at an arbitrary latitude, longitude and altitude behind the already translating platform, with the vehicle and the platform having approximately the same heading.

To practically achieve these initial conditions, the Safety Pilot is required to lift off. After the Safety Pilot has lifted off the Ground Station will take control of the vehicle. At this point, the vehicle will hold its position and the translating platform will start to move. As soon as the platform has reached the velocity at which the landing must be performed, the automated landing procedure will be started. The state machine is active between this stage and when the vehicle has touched down on the platform.

Refer to Figure 3.3, illustrating how all the vehicles will be coordinated throughout the test.



**Figure 3.3 – Coordination Of All Vehicles During Test Execution**

The test begins with the pursuit and towing motor vehicles stationary next to the road, while the Safety Pilot lifts the aircraft off the ground. The Ground Station takes control and the vehicle holds its position. The towing vehicle starts accelerating up to the speed at which the test is to be performed. Once the platform is translating at the required speed, the Ground Station Officer commands the aircraft to start tracking the platform. When the aircraft starts accelerating, the pursuit vehicle follows so that the Safety Pilot can monitor the aircraft. The vehicles stay in this convoy formation until the automated landing is completed.

## Chapter 4

# Mathematical Aircraft Model

The mathematical models of the aircraft and other aspects, such as wind and sensor noise, are derived in this chapter. The models, as they are expressed in this chapter, are used for non-linear simulations. In Chapter 5 these non-linear models are simplified and then used for control system design.

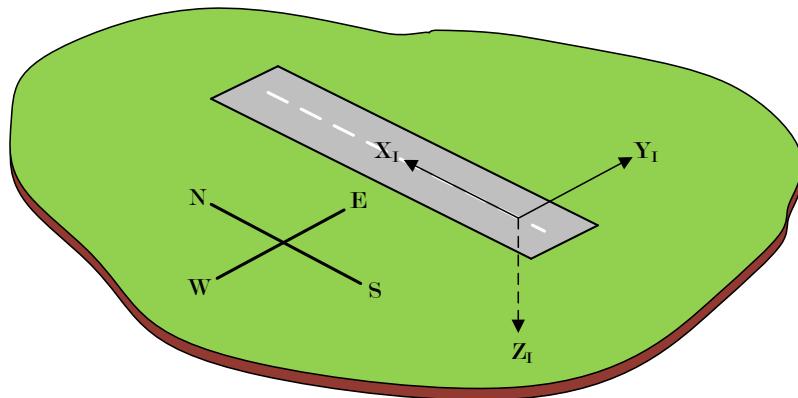
The chapter will start by explaining the different axis systems that will be used to express the state of motion of the aircraft, followed by the derivation of the equations of motion. The forces and moments that act on the body of the aircraft will then be derived. The chapter closes by elaborating on the sensor noise and wind models that are used during non-linear simulation in the thesis.

### 4.1 Axis Systems and Notation

All the axis systems that will be used throughout the thesis are defined in this section. The equations of motion can then be applied to express the aircraft's motion within these axis systems. The inertial axis is defined first, followed by the aircraft's body axis. The wind axis is then defined, finally followed by the notation used to express the forces, moments, velocities and angular velocities of the aircraft. All of the axis systems defined are right-handed orthogonal axis systems.

#### 4.1.1 Inertial Axes

Small-scale Unmanned Aerial Vehicles (UAV's) typically make use of the North-East-Down (NED) inertial axis system, seen in Figure 4.1. The NED axis system assumes that the earth is flat and that it is not rotating. The origin of this inertial axis system is chosen at a convenient place, such as the take-off location of the aircraft (illustrated in Figure 4.1 as the one end of the runway). As can be seen, the  $X_I$  axis points in the northern direction while the  $Y_I$  and  $Z_I$  axes points east and into the earth respectively.



**Figure 4.1 – Inertial Axis System**

### 4.1.2 Body Axes

The body axes, seen in Figure 4.2, are fixed to the aircraft and originate from its center of mass. The  $X_B$  axis is aligned with the front arm of the quadrotor, which points forward, and the  $Y_B$  axis is perpendicular to that arm, in this case pointing along the right arm. The  $Z_B$  axis is normal to the  $X_B$  and  $Y_B$  axes, pointing in the same direction in which air flows through the rotors.

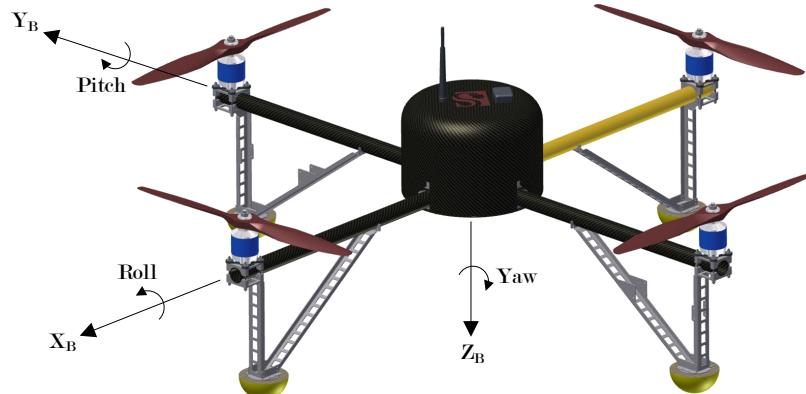


Figure 4.2 – Body Axis System

### 4.1.3 Wind Axes

The origin of the wind axis system, like the body axis system, is fixed to the aircraft's center of mass. The  $X_W$  axis is, however, not aligned with the  $X_B$  axis, but is aligned with the velocity vector of the aircraft and pointing in the velocity vector's direction. The  $Z_W$  axis is normal to the  $X_W$  axis, also pointing in the direction in which air flows through the rotors. The  $Y_W$  axis completes the right-handed orthogonal system, aligned normally to the  $X_W$  and  $Z_W$  axes. The wind axes can be seen in Figure 4.3, where  $\bar{v}$  is the velocity vector of the aircraft.

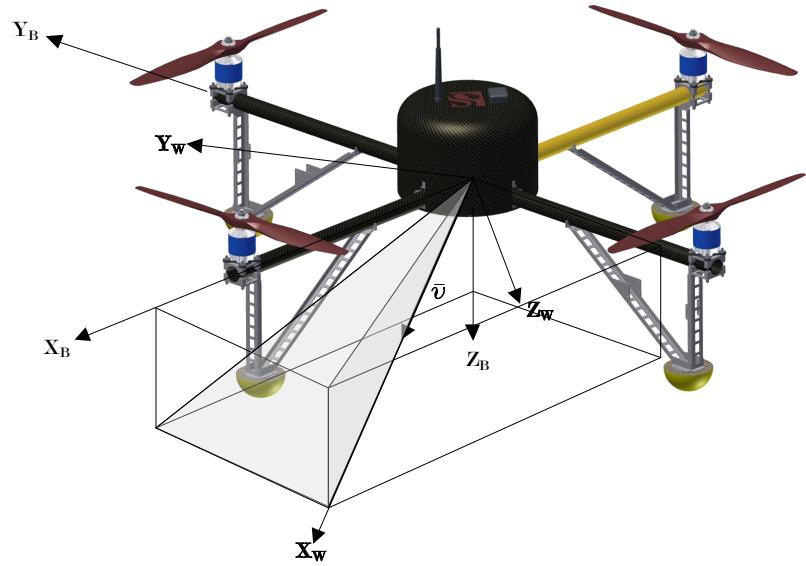


Figure 4.3 – Wind Axis System

#### 4.1.4 Notation

The following symbols will be used in the modeling of the vehicle's motion. The physical interpretation of these variables and their positive directions are shown in Figure 4.4.

$X, Y, Z$	The force magnitude and direction along the respective body axes.
$L, M, N$	The Moment magnitude and direction around the respective body axes.
$U, V, W$	The one-dimensional velocity magnitude and direction along each body axis.
$P, Q, R$	The magnitude and direction of the angular velocity around each body axis.

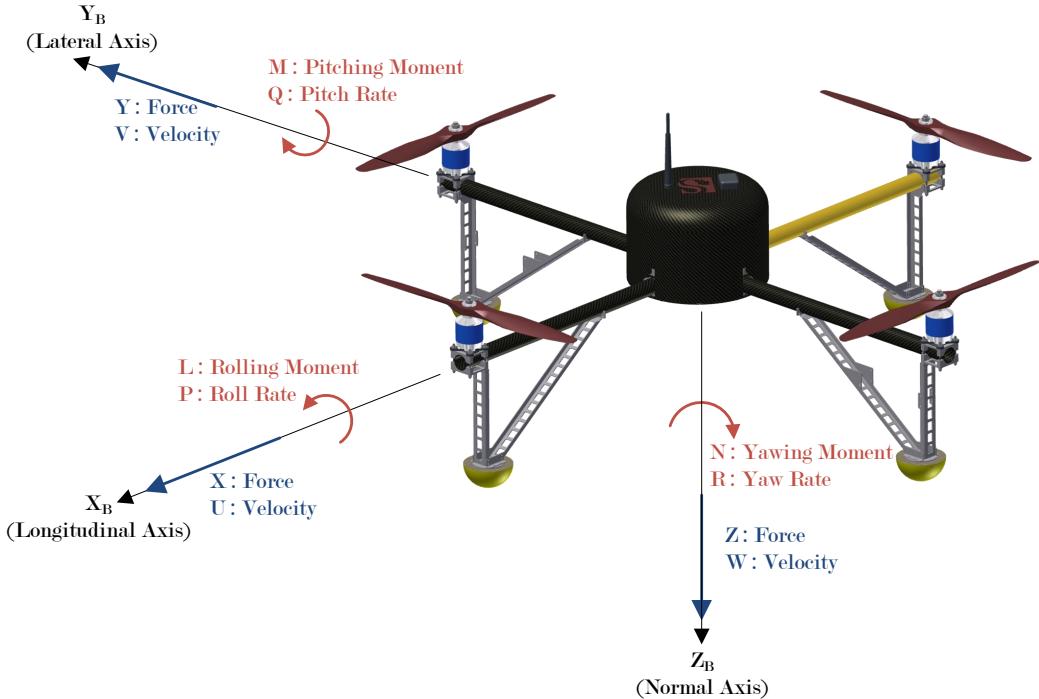


Figure 4.4 – Vehicle Notation Used

The notation of the velocity vector is also defined. Consider Figure 4.5. It is convenient to express the velocity vector in polar coordinates as a speed  $\bar{V}$ , an angle of attack  $\alpha$ , and an angle of side slip  $\beta$ ,

$$\bar{V} = \sqrt{U^2 + V^2 + W^2} \quad (4.1.1)$$

$$\alpha = \tan^{-1} \left( \frac{W}{U} \right) \quad (4.1.2)$$

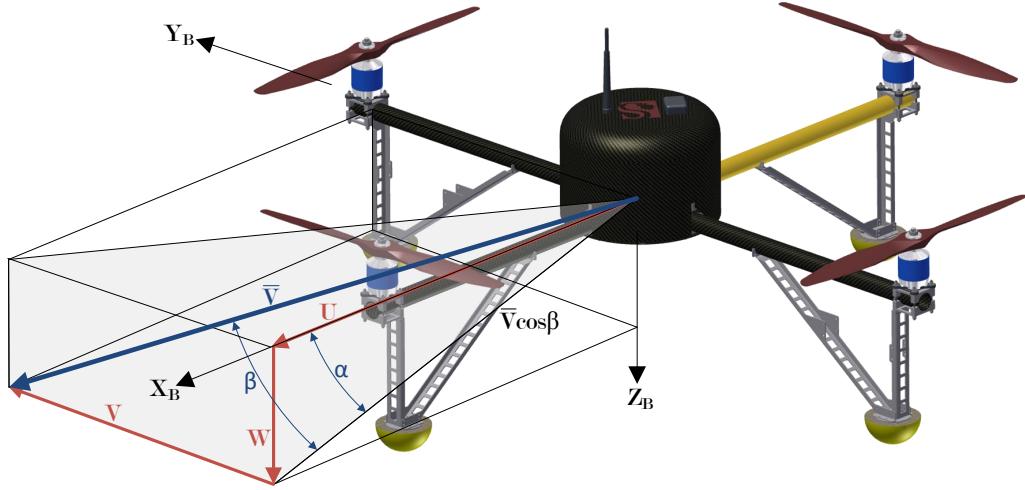
$$\beta = \tan^{-1} \left( \frac{V}{\bar{V}} \right) \quad (4.1.3)$$

The inverse relationships can be expressed as

$$U = \bar{V} \cos(\alpha) \cos(\beta) \quad (4.1.4)$$

$$V = \bar{V} \sin(\beta) \quad (4.1.5)$$

$$W = \bar{V} \sin(\alpha) \cos(\beta) \quad (4.1.6)$$



**Figure 4.5 – Velocity Vector Notation**

## 4.2 Kinetics and Kinematics

The equations of motion are derived in this section. To express the motion of the aircraft mathematically, the aircraft is modeled as a six-degree-of-freedom rigid body. The six degrees of freedom comprise of the aircraft's three translational degrees of freedom and its three rotational degrees of freedom.

The aircraft in this project can be expressed as a rigid body as the perturbations due to deformation experienced by the airframe are negligible. This is, however, not always the case, especially with aircraft with large airframes.

### 4.2.1 Kinetics

The kinetic equations relate the forces and moments acting on the vehicle to its kinematic state, such as its position, velocity and acceleration. Using Newton's second law of motion, as shown by Blakelock, the kinetic equations can be derived [27].

$$X = m(\dot{U} - VR + WQ) \quad (4.2.1)$$

$$Y = m(\dot{V} + UR - WP) \quad (4.2.2)$$

$$Z = m(\dot{W} - UQ + VP) \quad (4.2.3)$$

$$L = \dot{P}I_{xx} + QR(I_{zz} - I_{yy}) \quad (4.2.4)$$

$$M = \dot{Q}I_{yy} + PR(I_{xx} - I_{zz}) \quad (4.2.5)$$

$$N = \dot{R}I_{zz} + PQ(I_{yy} - I_{xx}) \quad (4.2.6)$$

In Equations 4.2.1 to 4.2.6,  $m$  is the mass of the aircraft and  $I_{xx}$ ,  $I_{yy}$  and  $I_{zz}$  are the moments of inertia around the aircraft's  $X_B$ ,  $Y_B$  and  $Z_B$  axes respectively. These equations are derived on the following assumptions:

1. The aircraft has a constant mass.
2. The aircraft is a rigid body.
3. The  $X_B$  and  $Z_B$  axes lie in the plane of symmetry of the aircraft, which implies that  $I_{xy}$  and  $I_{yz}$  are zero.
4. That  $I_{xz}$  is negligibly small.

For all practical purposes, the assumptions are valid for the vehicle used in this project.

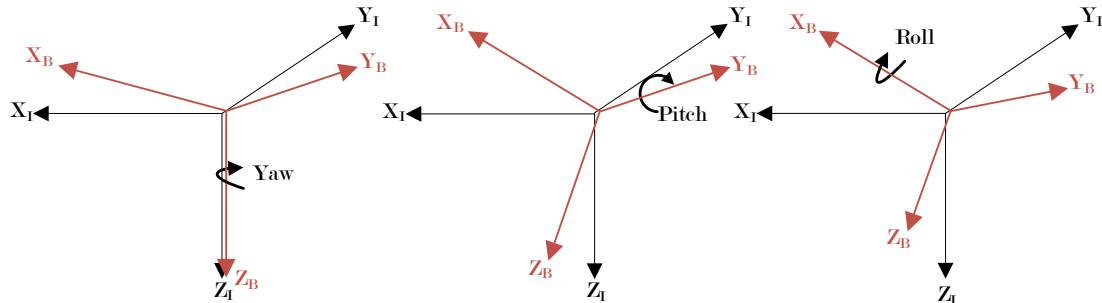
### 4.2.2 Kinematics

The kinematic equations are the mathematical relationships that relate the attitude and position of the vehicle to its translational velocity and angular rates. The transformation matrices that relate one axis system to another are also derived from these equations.

#### 4.2.2.1 Euler Angles

In order to describe the motion of the aircraft relative to the inertial axis system, it is necessary to specify the orientation of the body axis system relative to the inertial axis system. This can be done through three angles known as Euler angles. Euler angles can be specified in different ways. The way in which one Euler parameterization differs from another is the order in which the Euler angles are applied to determine the orientation of the vehicle. For this project Euler 3-2-1 attitude parameterization will be used. An illustration of the application of these Euler angles can be found in Figure 4.6. The sequence in which the three Euler angles are applied is as follows:

1. Yaw the original body axis system of the vehicle through the heading angle  $\Psi$ .
2. Pitch the resulting body axis through the pitch angle  $\Theta$ .
3. Roll the resulting body axis system through the roll angle  $\Phi$ .



**Figure 4.6 – Euler Angle Representation**

By applying the Euler angles in this order, the orientation of the vehicle relative to the inertial axes can be described.

### 4.2.2.2 Attitude Dynamics

The next step is to relate the Euler angles to other kinematic states. The rate of change of the Euler angles can be related to the body axes angular rates of the aircraft. Equation 4.2.7 describes the relationship between the body angular rates ( $P, Q, R$ ) and the time rate of change of the Euler angles for Euler 3-2-1 parameterization,

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin\phi\sin\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi\sec\theta & \cos\phi\sec\theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \quad \theta \neq \frac{\pi}{2} \quad (4.2.7)$$

### 4.2.2.3 Position Dynamics

The final kinematic equations that are needed to describe the state of the vehicle in the inertial axis system are presented here. The velocity vector of the vehicle in the body axis system needs to be expressed in the inertial axis system. Therefore some form of axis transformation is needed. Transformation matrices can be used for this purpose. Consider Figure 4.7,

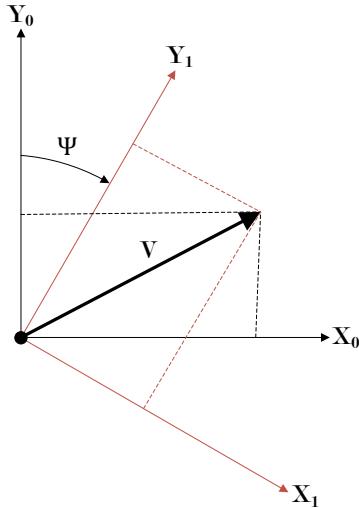


Figure 4.7 – Axis Yaw Rotation

where  $V$  is a velocity vector in an arbitrary original axis system and  $\psi$  the yaw angle between the original axis system and some other axis system. The velocity vector can be expressed in the original axis system as

$$V = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (4.2.8)$$

Through simple geometry it is straightforward to show that the coordinates of  $V$  in the other axis system are related through the transformation matrix in Equation 4.2.9

$$\begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad (4.2.9)$$

The result from the yaw transformation can similarly be pitched through a pitch angle  $\theta$  by using the following transformation matrix

$$\begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ z_1 \end{bmatrix} \quad (4.2.10)$$

Finally the result from the pitch rotation can be rolled through the roll angle  $\phi$  by using the final transformation matrix

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_2 \\ y_2 \\ z_2 \end{bmatrix} \quad (4.2.11)$$

The above three transformation matrices can be multiplied together to yield a single matrix that will execute axis system transformation through all three Euler angles,

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = \begin{bmatrix} C_\psi C_\theta & S_\psi C_\theta & -S_\theta \\ C_\psi S_\theta S_\phi - S_\psi C_\phi & S_\psi S_\theta S_\phi + C_\psi C_\phi & C_\theta S_\phi \\ C_\psi S_\theta C_\phi + S_\psi S_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} \quad \begin{aligned} C_\# &= \cos(\#), \\ S_\# &= \sin(\#) \end{aligned} \quad (4.2.12)$$

The transformation matrix in Equation 4.2.12 is known as the Direct Cosine Matrix (DCM). This matrix can be used to transform a velocity vector in the inertial axis system to the body axis system:

$$\begin{bmatrix} U \\ V \\ W \end{bmatrix} = \begin{bmatrix} C_\psi C_\theta & S_\psi C_\theta & -S_\theta \\ C_\psi S_\theta S_\phi - S_\psi C_\phi & S_\psi S_\theta S_\phi + C_\psi C_\phi & C_\theta S_\phi \\ C_\psi S_\theta C_\phi + S_\psi S_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} \quad \begin{aligned} C_\# &= \cos(\#), \\ S_\# &= \sin(\#) \end{aligned} \quad (4.2.13)$$

It can be shown that the DCM matrix is orthogonal and therefore its inverse is the transpose of the transformation matrix [7]. As a result, converting the velocity vector from body axis to inertial axis can be easily done by taking the transpose of the DCM in Equation 4.2.12 to yield Equation 4.2.14:

$$\begin{bmatrix} \dot{N} \\ \dot{E} \\ \dot{D} \end{bmatrix} = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} U \\ V \\ W \end{bmatrix} \quad \begin{aligned} C_\# &= \cos(\#), \\ S_\# &= \sin(\#) \end{aligned} \quad (4.2.14)$$

The motion of the six-degree-of-freedom rigid body of the aircraft is now sufficiently described for simulation and control system design.

### 4.3 Forces and Moments

With the motion of the aircraft described, the mathematical expressions are now derived that relate the forces and moments acting on the aircraft due to its state of motion.

For most airborne vehicles, forces and moments arise from three categories,

1. Actuators
2. Aerodynamics
3. Gravity

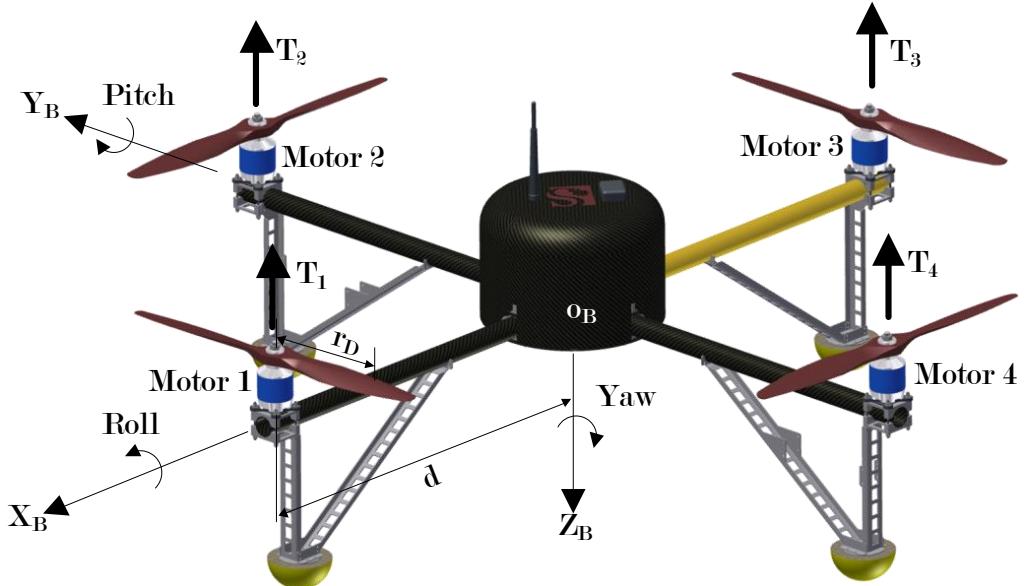
#### 4.3.1 Actuators

A quadrotor has only its four motors as actuators. The quadrotor uses variation of speed of all four of its rotors to generate the forces and moments necessary to control the aircraft.

An important parameter is how quickly the motors can respond to new thrust commands. According to Bouabdallah et al, it should be determined whether the time constant of the thrust response is short enough to be neglected [28]. This time constant from the rotor lag dynamics has a profound effect on the bandwidth of the vehicle.

This is also one of the factors that limit the size of quadrotors. The larger the quadrotor, the more thrust is required. More thrust requires larger diameter propellers, which are heavier and have a larger moment of inertia. According to Pounds et al, swash plates can be incorporated to compensate for this delay, but will compromise the robustness of a quadrotor due to increased mechanical complexity [29].

Consider Figure 4.8 which depicts the vertical forces that the motors generate,



**Figure 4.8 – Actuator Forces**

where  $d$  is the distance of the motor shaft from the center of the body,  $r_D$  is the chord length where the rotor drag force, assumed to be a point force, is exerted,  $R_{LD}$  is the rotor's lift to drag ratio and  $T_1$  to  $T_4$  are the thrust forces generated by the individual rotors.

The thrust forces that the rotors generate are modeled as point forces on the ends of each of the arms, as the motors are rigidly fixed to the arms. These generated forces cause moments around the body axes of the vehicle. The forces and moments are expressed in Equations 4.3.1 to 4.3.4,

$$Z^T = -(T_1 + T_2 + T_3 + T_4) \quad (4.3.1)$$

$$L^T = d(T_4 - T_2) \quad (4.3.2)$$

$$M^T = d(T_1 - T_3) \quad (4.3.3)$$

$$N^T = r_D(-T_1 + T_2 - T_3 + T_4)/R_{LD} \quad (4.3.4)$$

It should be noted that the above equations are for instantaneous thrust values. There are lag dynamics associated with the thrust values, as the rotors have inertia and are constantly experiencing drag.

The rotor lag dynamics can sufficiently be modeled as a first-order differential equation. From data recorded during thrust tests, the step response of the specific rotor and motor combination used, has a first order appearance. A thrust response from one of the experiments can be seen in Appendix B. The lag dynamics is captured in the following equations

$$T_1 = -\dot{T}_1\tau + T_{1R} \quad (4.3.5)$$

$$T_2 = -\dot{T}_2\tau + T_{2R} \quad (4.3.6)$$

$$T_3 = -\dot{T}_3\tau + T_{3R} \quad (4.3.7)$$

$$T_4 = -\dot{T}_4\tau + T_{4R} \quad (4.3.8)$$

where  $T_{\#R}$  is the reference thrust command and  $\tau$  is the time constant of the actual thrust response.

The time constant was determined by fixing a motor and propeller to a load cell. A reference step thrust value was commanded and the actual thrust that the motor exerts on the load cell was measured. The time constant could then be determined from the recorded step response. A segment of the results from such a test can be seen in Appendix B. The time constant was determined to be 0.125 seconds.

### 4.3.2 Aerodynamics

The mathematical model of the vehicle encapsulates aerodynamics that only arise from dynamic drag due to a frontal area. The parameters describing this drag were determined during previous work. The equation for the drag experienced by an object moving through a fluid is

$$F_D = \frac{1}{2}\rho v^2 C_D A \quad (4.3.9)$$

where the symbols represent the following variables:

$\rho$	The density of the fluid around the object
$v$	The linear velocity of the object relative to the fluid
$C_D$	The drag coefficient
$A$	The reference area

This equation is used with the assumption that the aircraft has a blunt form factor and the fluid has a large enough Reynolds number to produce turbulence behind the object.

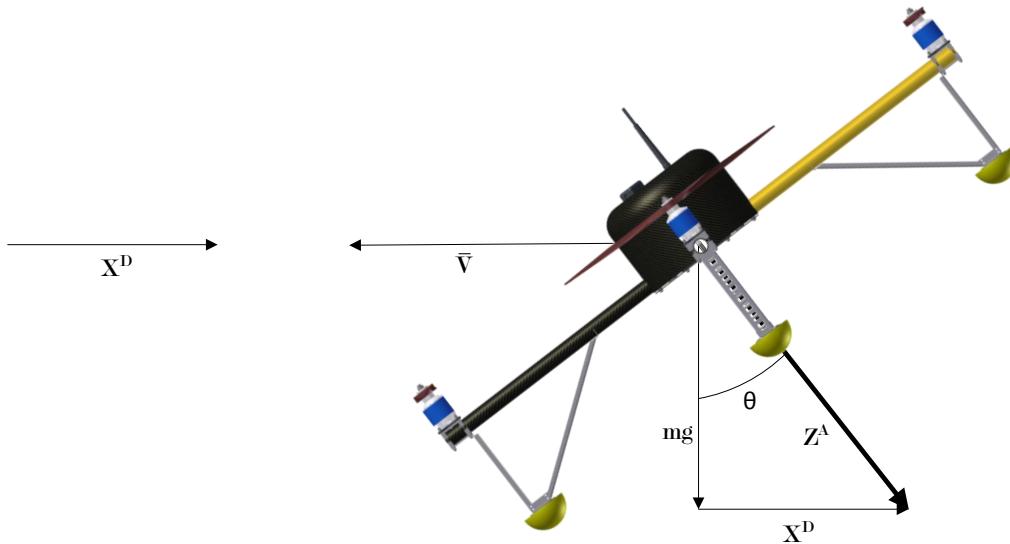
From the airflow over the vehicle, drag forces will be exerted in the x, y and z body axes against the direction of movement.

$$X^D = \frac{1}{2} \rho V_{BW_x}^2 A_x C_D \quad (4.3.10)$$

$$Y^D = \frac{1}{2} \rho V_{BW_y}^2 A_y C_D \quad (4.3.11)$$

$$Z^D = \frac{1}{2} \rho V_{BW_z}^2 A_z C_D \quad (4.3.12)$$

The drag coefficients and reference area for Equation 4.3.10 to 4.3.12 were determined experimentally during research prior to this project. The vehicle was flown, in no-wind conditions, at a constant velocity and height, which translates to a constant pitch and roll angle. The angle and the force gravity exerts on the vehicle were known and the drag force could be calculated. Refer to Figure 4.9, where the vehicle is flying at a pitch angle  $\theta$  and speed  $\bar{V}$ .



**Figure 4.9 – How The Drag Reference Area And Coefficients Were Determined**

The drag force and the velocity are now known. By using Equation 4.3.10 the reference area was determined by choosing 1 as the drag coefficient. The effective drag area, when viewing the vehicle along the  $X_B$ ,  $Y_B$  and  $Z_B$  axis, was determined to be  $0.3 \text{ m}^2$ .

### 4.3.3 Gravity

The force that gravity exerts on a vehicle can be modelled purely as the gravitational force due to the mass of the vehicle. In the inertial axis system the gravity vector points downward at all times and can be expressed as

$$\begin{bmatrix} F_x^G \\ F_y^G \\ F_z^G \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} mg \quad (4.3.13)$$

The inertial gravity vector can be converted to the body axis system through use of the Direction Cosine Matrix. The result is Equation 4.3.14:

$$\begin{bmatrix} X^G \\ Y^G \\ Z^G \end{bmatrix} = \begin{bmatrix} -\sin\theta \\ \cos\theta\sin\phi \\ \cos\theta\cos\phi \end{bmatrix} mg \quad (4.3.14)$$

Lastly, the gravity vector does not induce any moments on the aircraft due to the gravity vector acting through the center of mass. Therefore,

$$L^G = M^G = N^G = 0 \quad (4.3.15)$$

## 4.4 Sensor Models

The sensors on board the quadrotor measure some of its kinetic states. In Table 4.1 the kinetic states and the sensors that measure them can be seen,

**Table 4.1** – Kinetic States Measured By On-Board Sensors

State	Symbol	Sensor
Pitch Rate	Q	Pitch Gyroscope
Roll Rate	P	Roll Gyroscope
Yaw Rate	R	Yaw Gyroscope
Specific Force along $X_B$	$\dot{U}$	X Accelerometer
Specific Force along $Y_B$	$\dot{V}$	Y Accelerometer
Specific Force along $Z_B$	$\dot{W}$	Z Accelerometer
Speed along $X_I$	$\dot{N}$	GPS N Speed
Speed along $Y_I$	$\dot{E}$	GPS E Speed
Speed along $Z_I$	$\dot{D}$	GPS D Speed
Position along $X_I$	N	GPS N Position
Position along $Y_I$	E	GPS E Position
Position along $Z_I$	D	GPS D Position

For linear simulation, steady-state measurement errors were considered for some sensors, especially sensors that play a vital role in the automatic landing strategy. These primarily

consisted of measurement errors from the GPS N and GPS E speed measurements. Characterization of these errors was done experimentally and can be seen in Chapter 6.

For non-linear Software-In-the-Loop (SIL) and Hardware-In-the-Loop (HIL) simulation purposes, noise was introduced into the system by use of Simulink's Band-Limited White Noise blocks (BLWN). These blocks were characterized during prior work in the ESL on the IMU and GPS units in use on the vehicles in the lab.

For each gyroscope, accelerometer and GPS speed sensor, a BLWN block is used with a specific Power Spectral Density (PSD) height and sample time. Refer to Table B.1 in Appendix B for the PSD and sample time values for each of the sensors. Different seeds are used for every sensor.

The GPS N, E and D position measurement noise is characterized by use of BLWN blocks and filters. Each N, E and D measurement error is generated by a BLWN block, which is then filtered by first-order filters to simulate low frequency drift. An exception is the GPS D measurement, which had a high-frequency component added to the low-frequency GPS position drift. The structure of the GPS N, E and D noise simulation block can be seen in Appendix B.

## 4.5 Wind Model

The wind model is the final mathematical model to be introduced. The model, which is used in non-linear simulations, incorporates constant wind and wind gusts. The wind components which disturb the aircraft are generated in the inertial axis system. Through the kinematic relationships these components are resolved into the wind and body axis systems. The structure of the wind model can be seen in Figure 4.10,

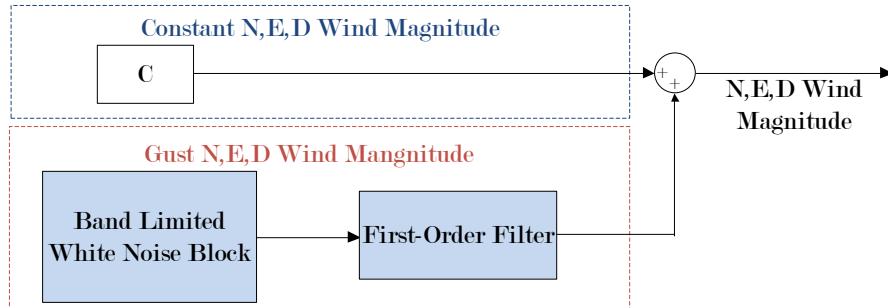


Figure 4.10 – The Wind Model

The wind gusts are simulated by using a BLWN block and a first-order filter. The first-order filter has a 5 second time constant which smoothes the output from the BLWN block.

## Chapter 5

# Control System Analysis and Design

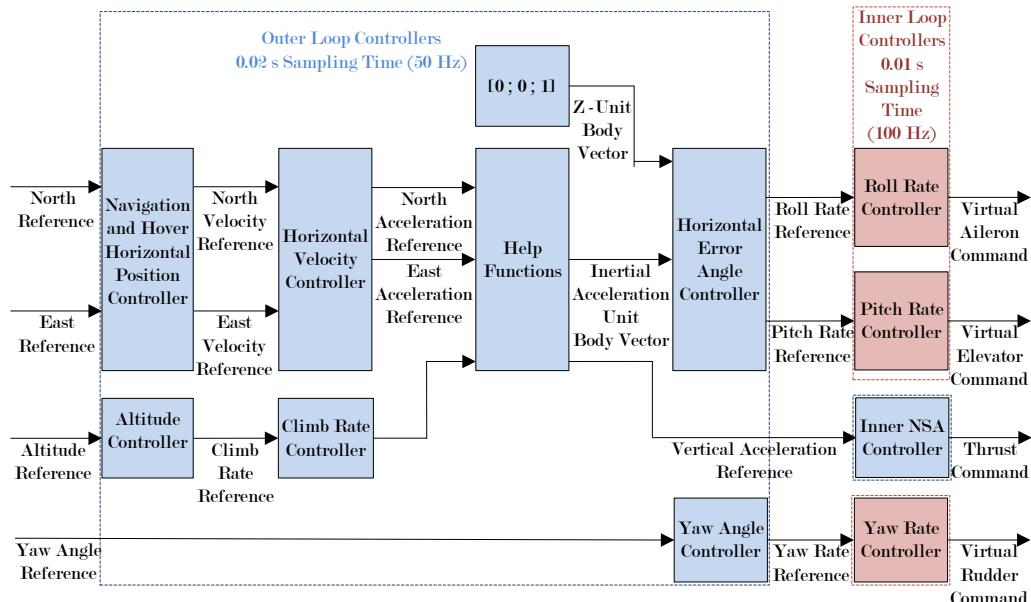
In this chapter, control system analysis and design is performed. The chapter is divided into two major sections. In the first section, the pre-existing flight control system on the quadrotor is analyzed to understand its architecture, to evaluate its performance, and to identify possible areas of improvement.

In the second section, the flight control system is re-designed and upgraded specifically for the task of automated landing on a moving platform. The chapter concludes with non-linear Software-In-the-Loop (SIL) simulation and an evaluation of both the pre-existing and the re-designed flight control system.

### 5.1 Analysis of Pre-Existing Control System For Autoland On A Moving Platform

In this section, the pre-existing flight control system is presented, its architecture is analyzed, its performance is evaluated, and possible areas of improvement are identified. First, an overview of the entire control system is given, and then each individual control loop is considered in detail. The detail analysis is broken up into three sections, analyzing the horizontal, yaw and vertical control systems separately.

The flight control system of the quadrotor is a suite of decoupled control subsystems which are each responsible for controlling a different axis of the vehicle. A top-level block diagram of the flight control system is shown in Figure 5.1.

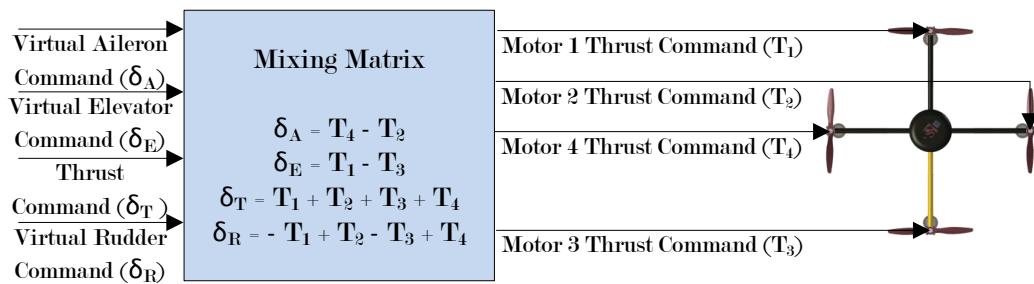


**Figure 5.1 – Top-Level Block Diagram of the Flight Control System**

Each axis has a combination of proportional, integral and derivative control laws in consecutive loops to finally control the position of the vehicle in space.

The outputs of the flight control system are virtual aileron, elevator and rudder commands. The commands are called virtual commands, because a quadrotor vehicle does not physically have ailerons, elevators and a rudder like a fixed-wing aircraft does. However, the four motors may be actuated in a way that they appear to have the same effect on the quadrotor as ailerons, elevators and a rudder have on a fixed-wing aircraft.

A mixing matrix is required to take these virtual commands and convert them to separate thrust values for each of the four motors. The motors will then generate the moments and forces required to move the vehicle. An illustration of this concept can be seen in Figure 5.2.



**Figure 5.2 – Mixing Matrix**

As an example, the virtual aileron command,  $\delta_A$ , will cause actuation of motors four and two. For a positive  $\delta_A$ , the thrust  $T_4$  that motor four generates, will increase, and the thrust  $T_2$  that motor two generates, will decrease. This will induce a rolling moment, similar to what an aileron does on a fixed-wing aircraft.

The control systems responsible for control of the different axes of the aircraft are now individually investigated. Very limited documentation was available on the design and implementation of the pre-existing control system of the aircraft, as it was derived and implemented during time-constrained periods of the SLADe project. All analysis and documentation in this chapter is therefore performed after attentively studying the current C code on the On-Board Computer (OBC).

### 5.1.1 Horizontal Control System

A detailed analysis of the horizontal control system is firstly performed. As shown in Figure 5.1, the horizontal control system consists of the horizontal rate, angle, velocity and position controllers.

The inner loop controllers of the horizontal control system, namely the roll rate and pitch rate controllers, run at a sampling rate of 100 Hz, while all the outer loop controllers run at a sampling rate of 50 Hz. The angular rate controllers receive their references from the tilt angle controller. The tilt angle controller receives its reference from the horizontal velocity controllers, and finally the velocity controllers receive their references from the horizontal position controllers.

Because the vehicle is symmetric around its centre of mass, the longitudinal and lateral control loops are identical. Only the analysis of the longitudinal controllers will therefore be presented, since it is representative of both the longitudinal and lateral controllers.

### 5.1.1.1 Pitch Rate Dynamics

A model of the vehicle's longitudinal dynamics is required before analyzing the longitudinal control system. The following linear, near-hover model is derived.

The state variables are chosen as  $M$  and  $Q$ , where  $M$  is the pitching moment and  $Q$  the pitch rate of the vehicle. A virtual elevator actuator  $\delta_E$  is also defined, which will cause a pitching moment when actuated:

$$\delta_E = d(T_1 - T_3) \quad (5.1.1)$$

In near-hover flight, the relationship between angular acceleration in pitch  $\dot{Q}$  and applied pitching moment  $M$  is governed by Newton's law for rotational motion and can be simplified to:

$$I_{yy}\dot{Q} = M \quad (5.1.2)$$

The pitching moment  $M$  is produced by the differential thrust of motors one and three, which is controlled through the virtual elevator actuator  $\delta_E$ :

$$M = d(T_1 - T_3) = \delta_E \quad (5.1.3)$$

$T_1$  and  $T_3$  are the respective motor thrusts, and  $d$  is the moment arm of the motors relative to the centre of mass of the quadrotor.

As Equation 5.1.1 is just the superposition of the thrust developed by rotors one and three,  $\delta_E$  will have the same lag dynamics as the rotors, which was derived in Chapter 4. Defining  $\delta_{E_R}$  as the virtual elevator reference command yields Equation 5.1.4, which describes the actuator lag dynamics:

$$\dot{\delta}_E = -\frac{1}{\tau}\delta_E + \frac{1}{\tau}\delta_{E_R} \quad (5.1.4)$$

The differential equation in 5.1.2 and Equation 5.1.3 can be represented in state space form as:

$$\begin{bmatrix} \dot{M} \\ \dot{Q} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} & 0 \\ \frac{1}{I_{yy}} & 0 \end{bmatrix} \begin{bmatrix} M \\ Q \end{bmatrix} + \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \delta_{E_R} \quad (5.1.5)$$

$$y = [0 \quad 1] \begin{bmatrix} M \\ Q \end{bmatrix} \quad (5.1.6)$$

and the transfer function from virtual elevator reference command  $\delta_{E_R}$  to pitch rate  $Q$  can be derived (see Appendix B for full derivation),

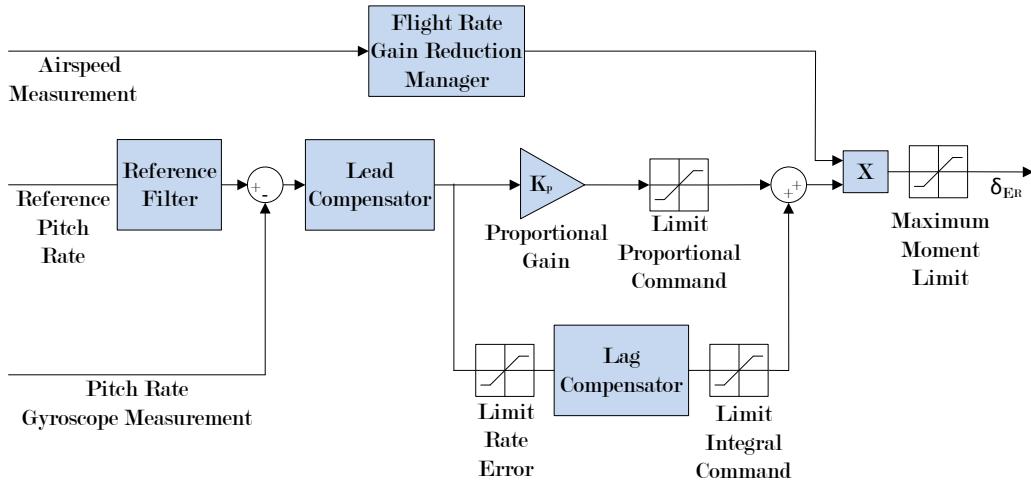
$$G(s) = \frac{\frac{1}{\tau I_{yy}}}{s(s + \frac{1}{\tau})} \quad (5.1.7)$$

From Equation 5.1.7 it can be seen that the plant has a pure integrator and a pole at  $s = -\frac{1}{\tau}$ .

Now that the pitch rate dynamics of the vehicle have been modeled, the pitch rate controller can be analyzed.

### 5.1.1.2 Pitch/Roll Rate Controllers

The architecture of the pitch and roll rate controllers were determined from the C code currently on the On-Board Computer (OBC) and can be seen in Figure 5.3. It is not known why this is the chosen structure of the controller and what methodology was followed during its design. The controller consists of a lead compensator, proportional gain and lag compensator.

**Figure 5.3 – Pre-Existing Pitch Rate Controller Architecture**

A non-linear component forms part of the controller and is called the Flight Rate Gain Reduction Manager. From previous research it was seen that the gain of the pitch rate plant increases as the airspeed of the vehicle increases. The non-linear component reduces the gain of the controllers during periods of fast flight. The gain reduction is scheduled as a function of the airspeed measurement. When the vehicle is near hover, the gain factor is unity, and as the vehicle gains airspeed the gain factor is reduced.

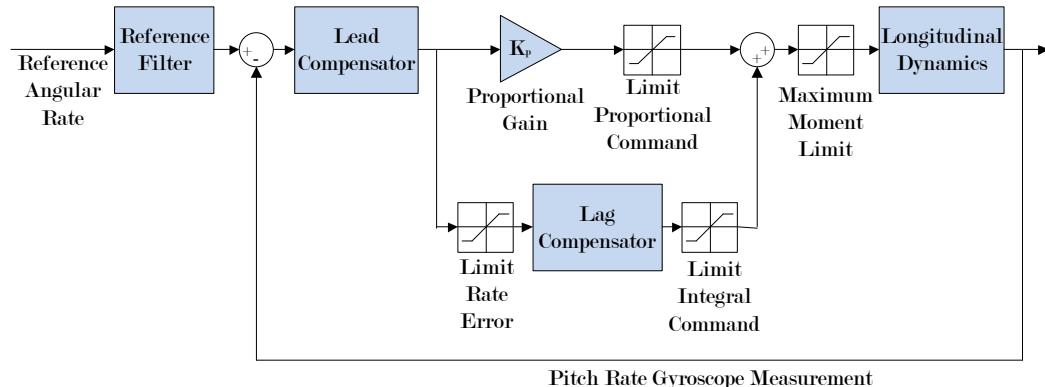
This non-linear form of control effectively keeps the bandwidth of the pitch rate system constant during hover and fast-forward flight. As the Flight Rate Gain Reduction parameters were determined empirically, they will not form part of this analysis.

The limiters presented in Table 5.1 also form part of the control structure.

**Table 5.1 – Pre-Existing Pitch Rate Controller Limits and Their Functions**

Limit	Function	Value
Maximum Moment Limit	Prevent actuator saturation	24.5 Nm
Proportional Command Limit	Limit proportional command	12.5 Nm
Integral Command Limit	Limit the maximum disturbance rejection	12 Nm
Rate Error Limit	Constrain integrator wind-up speed	14 deg/s

The simplified block diagram used for the analysis, which omits the non-linear gain reduction component, can be seen in Figure 5.4.

**Figure 5.4 – Pitch Rate Controller Block Diagram Used For Analysis**

Before the analysis is started, an intuitive look is taken at the pitch rate controller. The proportional term is most likely used to obtain the desired bandwidth, whereas the lead compensator is used to increase the phase of the system, thus improving the transient characteristics. The lag compensator will improve steady-state tracking and compensate for unmodelled dynamics and disturbances. The combination of these terms approximates a Proportional Integral Derivative (PID) controller.

The control structure is studied next. The likely design procedure that was followed can be deduced from the final structure of the controller. It appears that the proportional gain was firstly determined to obtain the desired bandwidth. The lag compensator was then added, to compensate for any unmodelled dynamics or disturbances. The lag compensator could also have been used to replace a pure integrator, which could have affected the phase margin at the gain crossover frequency too drastically. The initial design of the controller could therefore have been a pure Proportional Integral (PI) controller.

The lead compensator was most likely added after the initial (proportional and lag compensator) design. The addition could be to improve the transient response of the system, or to maintain the transient response with an increased proportional gain, for a scenario where a higher system bandwidth was desired.

An advantage of this structure is that the control authority that different control terms have can be managed. Consider Figure 5.5 where three different control architectures are depicted.

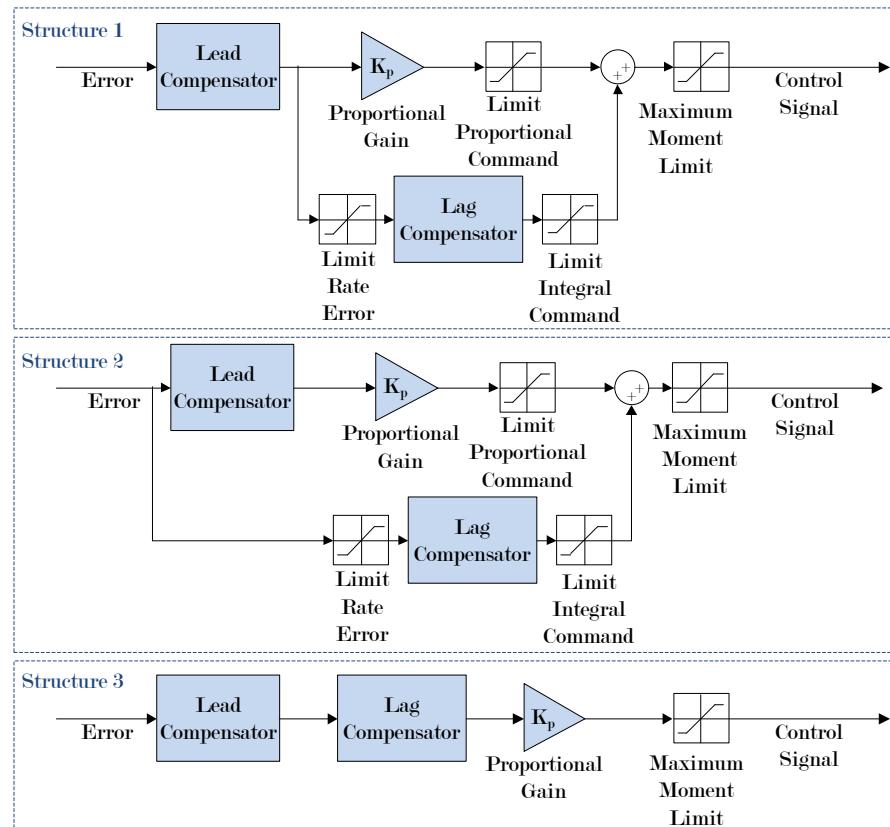


Figure 5.5 – Different Control Architectures for Pitch Rate Controller

All three architectures will be able to control the plant in approximately the same fashion, but limiters can be implemented very specifically in the first two structures. In the third structure there is less freedom to implement limiters.

System type and steady-state characteristics are studied next. The loop transfer function of the block diagram in Figure 5.4, when expressed from the reference input to the output, is shown in Equation 5.1.8:

$$G(s)_{PitchRateOL} = \frac{592s^2 + 6852s + 6457}{s^4 + 94s^3 + 690s^2 + 27s} \quad (5.1.8)$$

One free integrator is present, making the system type one for tracking. This integrator arises from the linear longitudinal dynamics of the vehicle. According to this linear model the vehicle will track a reference pitch rate step input with a zero steady-state error. The system will track a ramp input with a finite steady-state error. The position error constant of the system is therefore zero. The velocity error constant is calculated to be:

$$K_V = \frac{6457}{27} = 240 \quad (5.1.9)$$

which translates to a steady-state error of:

$$e_{ss} = \frac{1}{K_V} = \frac{1}{240} = 0.0042 \text{ rad/s} \quad (5.1.10)$$

The velocity error constant is also computed without the lag compensator in the system, to illustrate its role in reducing the steady-state error:

$$K_V = \frac{6207}{685.7} = 9.05 \quad (5.1.11)$$

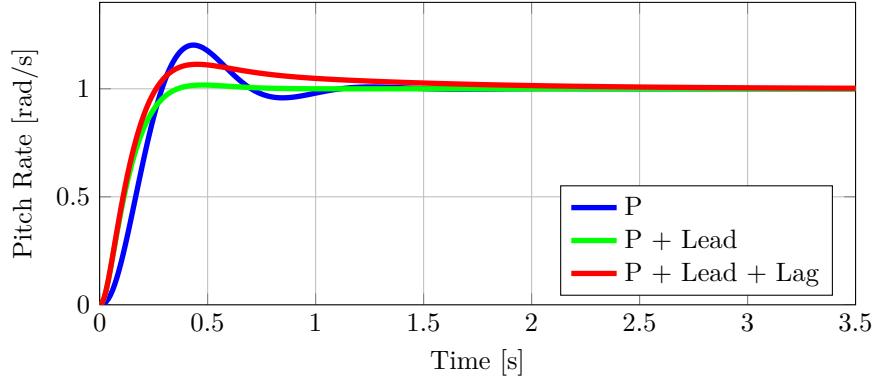
which translates to a steady-state error of:

$$e_{ss} = \frac{1}{K_V} = \frac{1}{9.05} = 0.1105 \text{ rad/s} \quad (5.1.12)$$

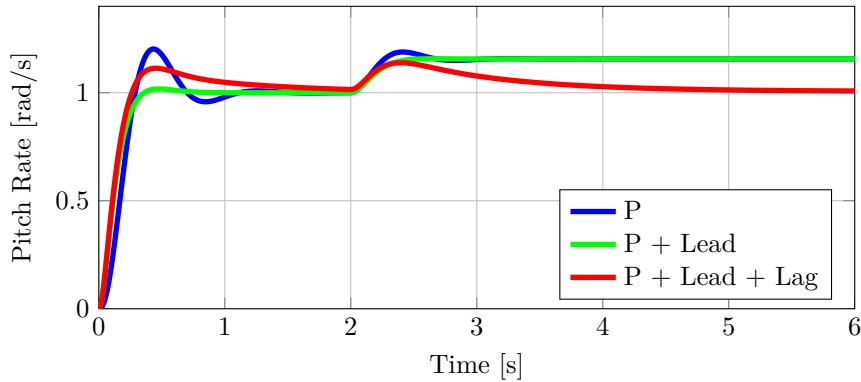
The lag compensator therefore decreases the steady-state error due to a ramp input by a factor of 26.

As this control loop is the innermost control loop, it should respond as fast as possible. If a disturbance is present in the system, the innermost control loops will negate these disturbances the fastest. It therefore means that the bandwidth of this control loop should be as high as the dynamics of the vehicle allow, but with a safe phase and gain margin to guarantee stability. The physical factors that limit the bandwidth of the pitch rate dynamics are the inertia of the rotors, the electrical properties of the motors, the moment of inertia of the vehicle chassis and the sampling rate of the control loop. From Table 5.1, the limits show that the proportional command and the integral command almost have equal control authority, indicating that the system is very dependent on integral control to compensate for disturbances and unmodelled dynamics.

Figure 5.6 depicts what the step response of the system will look like if each of the components of this controller is omitted. This is done to analyze how each component affects the transient and steady-state characteristics of the response. No limiters were included in the simulation of the step response. When only using proportional control, some notable overshoot and oscillation are present, indicating that the bandwidth is at a point where the system is underdamped. When using the proportional controller and the lead compensator, a more damped response is observed, clearly showing that the lead compensator improves the phase margin. When the lag compensator is included in the controller, some overshoot is observed.

**Figure 5.6** – Step Response of Pre-Existing Pitch Rate System

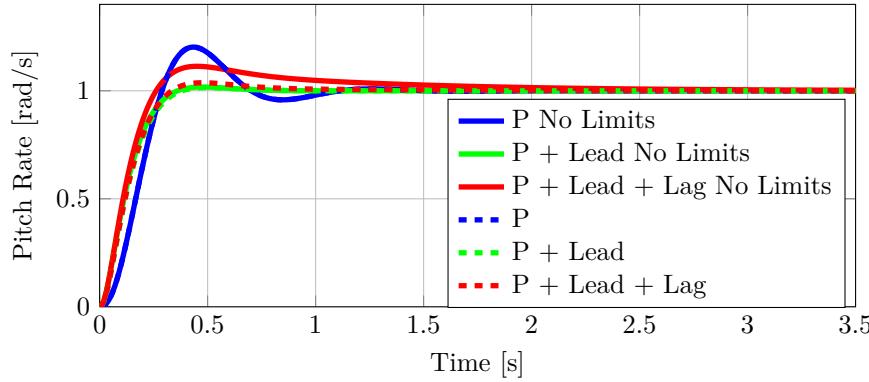
When all control terms are taken into account, a rise time of 0.18 seconds and a 5% settling time of 0.97 seconds are seen. The time constant of the system is 0.138 seconds and a maximum overshoot of 11% is observed. The overshoot addition from the lag compensator can be explained with the help of Figure 5.7. The system responds to a reference unit step input at time zero and is also disturbed with a unit step input two seconds later.

**Figure 5.7** – Disturbance Response of Pre-Existing Pitch Rate System

It can be seen how the proportional gain and lead compensator negate some of the disturbance, but a steady-state error still exists. This error is, however, rapidly negated by the lag compensator, indicating that the lag compensator has a high cutoff frequency. The overshoot caused by the lag compensator during the transient portion of the step response is therefore a result of the lag compensator having a high cutoff frequency. The high cutoff frequency is most probably needed for rejecting disturbances and compensating for unmodelled dynamics in the plant model. It is not known how the lag compensator's cutoff frequency was determined. The gain and cutoff frequency were likely determined experimentally, as it can be challenging to quantify frequencies of external disturbances and uncertainties in the plant model.

The step response which incorporates the limiters listed in Table 5.1 is plotted in Figure 5.8. The effect of the limiter in front of the lag compensator can be seen clearly.

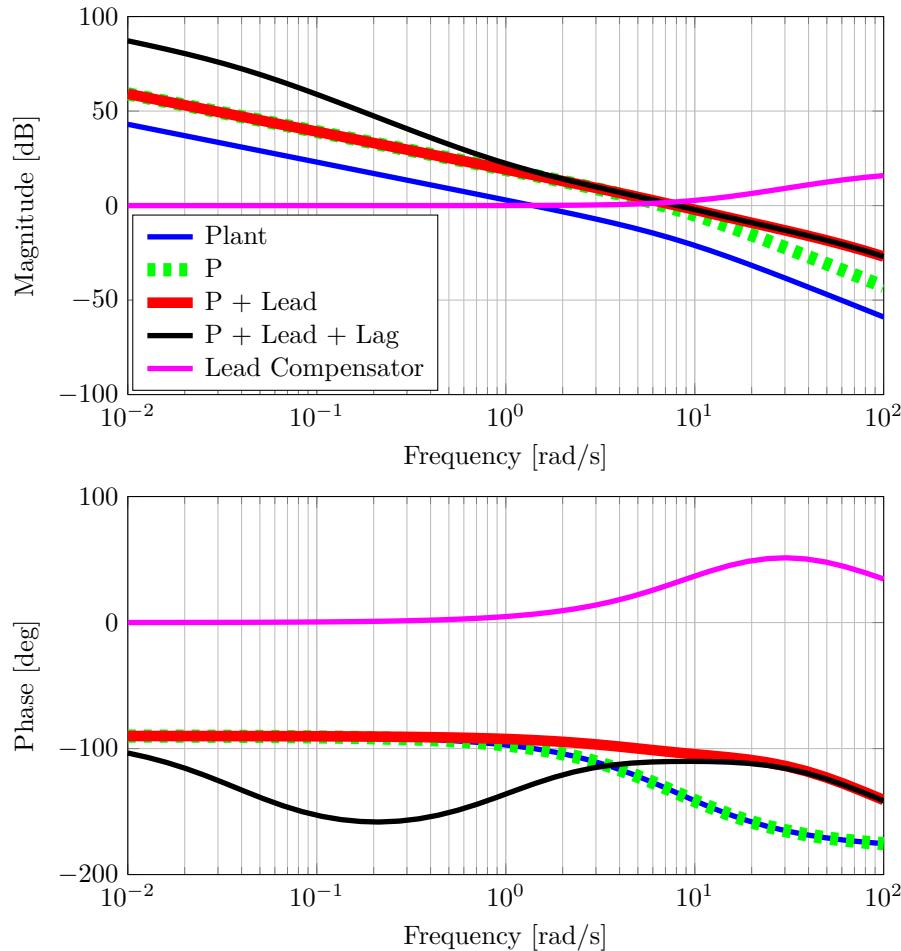
The overshoot due to the lag compensator's contribution is less, showing a maximum overshoot of 3.8% instead of the 11% overshoot seen in Figure 5.6. This suggests that the limiter is used to constrain the wind-up of the integrator for large pitch rate errors, and is a clever



**Figure 5.8 – Step Response of Pre-Existing Pitch Rate System With Limiters Included**

method to ensure that the integrator functions more during steady-state behavior, which will be the behavior that constant moment disturbances exhibit.

The open-loop Bode plots of the pitch rate system is plotted in Figure 5.9, and are investigated next.

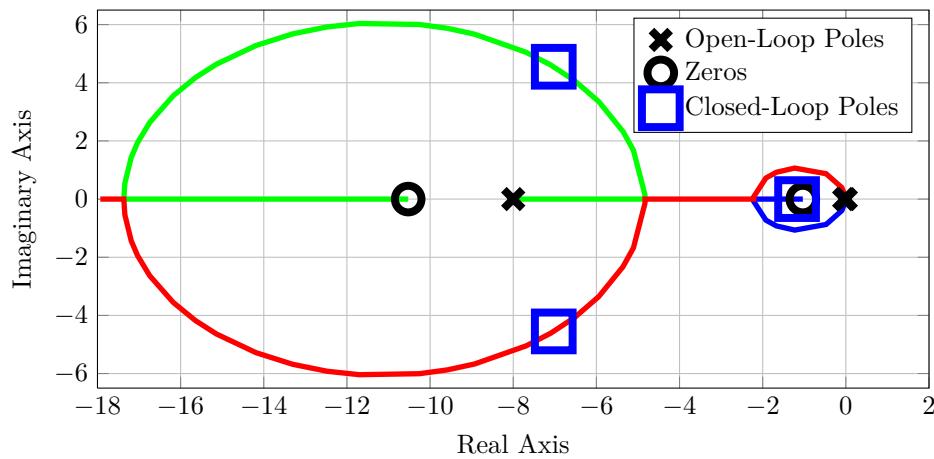


**Figure 5.9 – Open-Loop Bode Plot of Pre-Existing Pitch Rate System**

It can be seen how the proportional gain plays the largest role in changing the gain crossover frequency of the system from  $1 \text{ rad/s}$  to  $10 \text{ rad/s}$ . The lead compensator slightly affects the gain crossover frequency and amplifies frequencies higher than the gain crossover frequency. The lag compensator does not affect the gain crossover frequency and amplifies frequencies lower than the gain crossover frequency. The lag compensator does not have an infinite gain for DC frequencies and thus acts as a leaky integrator. The Bode plot further supports the speculated design strategy that was used. The lead compensator was designed last to add phase and was added after the proportional gain was used to increase the bandwidth. The lead compensator adds  $35^\circ$  of phase to the phase margin, whereas the lag compensator removes approximately  $5^\circ$  of phase. The final phase margin is  $70^\circ$ , which corresponds to a optimally damped system.

The Bode plot of the lead compensator is also plotted. It is seen that the lead compensator does not add its maximum phase at the gain crossover frequency. The maximum phase that the lead compensator can add to the system is  $50^\circ$ , but at a frequency of  $30 \text{ rad/s}$ . It is possible that the designers deliberately designed the lead compensator in this non-standard manner, so that the magnitude plot of the lead compensator would amplify frequencies higher than the gain crossover frequency more. This lead compensator also changes the phase curve in such a way that the curve has a flatter appearance directly before and after the gain crossover frequency. This will ensure that the phase margin will remain practically constant for slight changes in the gain crossover frequency.

The root locus was lastly investigated and is shown in Figure 5.10. The open-loop and closed-loop poles are also drawn.



**Figure 5.10 – Root Locus of Pre-Existing Pitch Rate System**

An open-loop pole is located at the origin, which is the integrator that arises from the pitch rate plant dynamics. Similarly, a pole is located at  $s = -8$ , which also arises from the pitch rate plant dynamics, more specifically the lag dynamics of the rotors. The lead compensator and lag compensator each introduced another open-loop pole, located at  $s = -85.70$  and  $s = -0.04$  respectively. The lead compensator's pole is not considered to have an effect on the behavior of the system. Two zeros are also introduced by the lead compensator and lag compensator, located at  $s = -10.53$  and  $s = -1.04$  respectively.

Looking at the closed-loop poles, pole/zero cancellation almost takes place at  $s = -1.04$ , leaving the dominant behavior of the system to the complex pole pair at  $s = -7.03 \pm 4.36i$  and the zero at  $s = -10.53$ . The pole/zero cancellation is due to the addition of the lag compensator, which by definition should not affect the root locus remarkably.

This concludes the analysis of the pitch rate system. It can be said with certainty that the design goal was to increase system bandwidth and to negate disturbances aggressively, while still maintaining a high enough low-frequency gain to counter constant disturbances. The bandwidth is as high as the dynamics of the system allow, in order to reject dynamic disturbances, while still maintaining an acceptable phase margin of  $70^\circ$ . The system also has a high enough low-frequency gain to find the differential thrust trim and to counter constant disturbance torques.

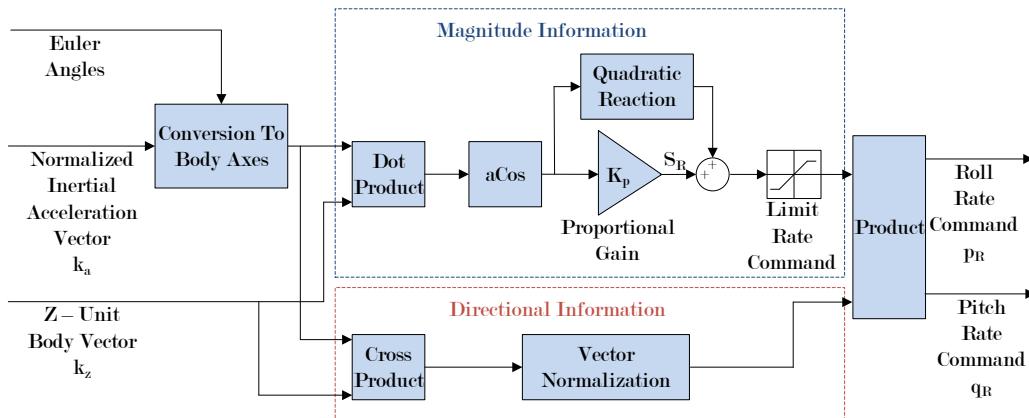
The architecture of the pre-existing pitch rate and roll rate controllers would not have been the first design choice, as intuitively it would have made sense to attempt to use a PI controller at first. Yet it should still be questioned whether the lag compensator performs notably better than a pure integrator with an appropriately chosen zero frequency. It is nevertheless clear why a lead compensator was incorporated to achieve a higher bandwidth while maintaining an acceptable transient response.

The control loop is well designed and utilizes the dynamics of the vehicle as well as it can for this application, while still maintaining a safe phase margin. High-frequency and low-frequency disturbances are well rejected, and the system is well damped.

### 5.1.1.3 Tilt Angle Controller

The tilt angle controller, shown in Figure 5.11, controls the tilt angle of the vehicle by commanding pitch and roll rates. This controller receives an inertial acceleration vector reference input from the horizontal velocity controller and the climb rate controller, and outputs pitch rate and roll rate references to the pitch rate controller and the roll rate controller.

In addition to the proportional gain, the controller also contains two non-linear components, namely a quadratic reaction gain and a rate command limiter. The quadratic reaction gain is used in conjunction with the proportional gain to improve the angle tracking performance. The rate command limiter is introduced to prevent actuator saturation. Without the rate command limiter, a too-large angular rate can be commanded, which requires a larger moment. This will in turn cause the pitch rate controller and roll rate controller to actuate the rotors aggressively.

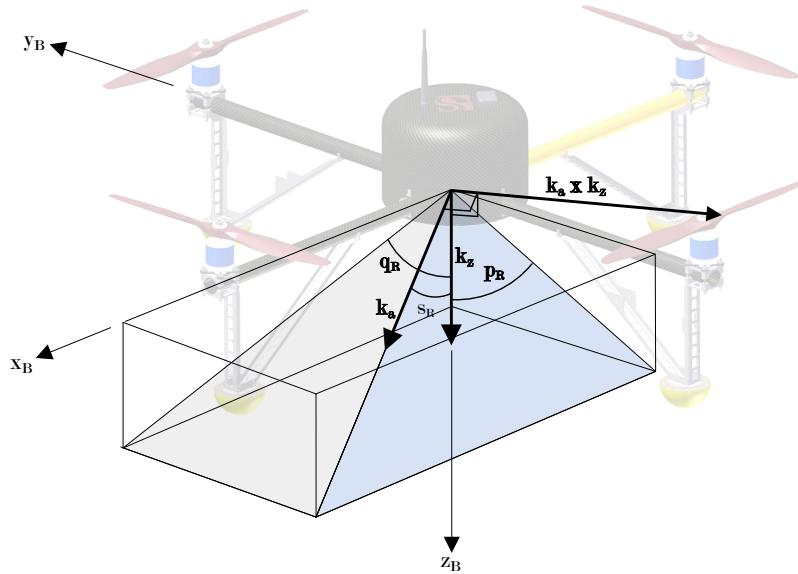


**Figure 5.11 – Pre-Existing Tilt Angle Controller Architecture**

A fundamental concept of the tilt angle controller is how the controller calculates the magnitude and direction of the roll and pitch rate references. The controller does this by making

use of the dot and cross product between two vectors. This is done to make the tilt angle control independent of the yaw angle of the vehicle.

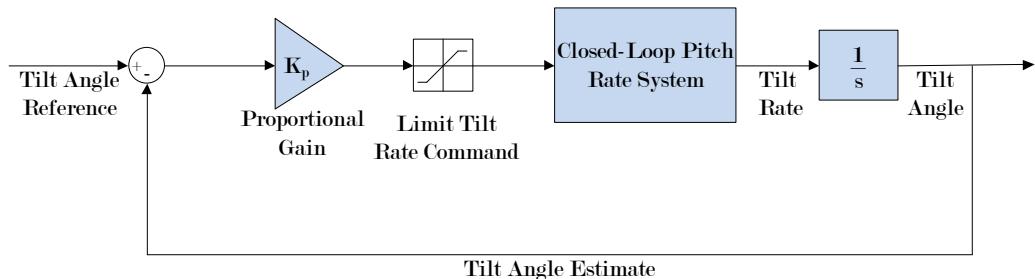
Consider Figure 5.12, which depicts the two vectors that the controller uses, and Figure 5.11 which shows the controller architecture. The normalized reference acceleration vector, commanded by the horizontal velocity and climb rate controller, is converted to the body axis system to yield unit acceleration vector  $k_a$ . The other vector shown is  $k_z$ , a constant unit vector in the body z-axis direction.



**Figure 5.12 – Tilt Angle Calculation by the Tilt Angle Controller**

By taking the arc cosine of the dot product of  $k_a$  and  $k_z$ , and multiplying the result by the proportional gain, the total tilt rate  $s_R$  is calculated. The cross product of  $k_a$  and  $k_z$  is also computed to yield the direction of  $s_R$  in body axes, resulting in the reference roll rate  $p_R$  and pitch rate  $q_R$  commands. The goal of the tilt angle controller is thus to align the body z-axis of the vehicle (also the thrust axis) with the desired inertial acceleration vector.

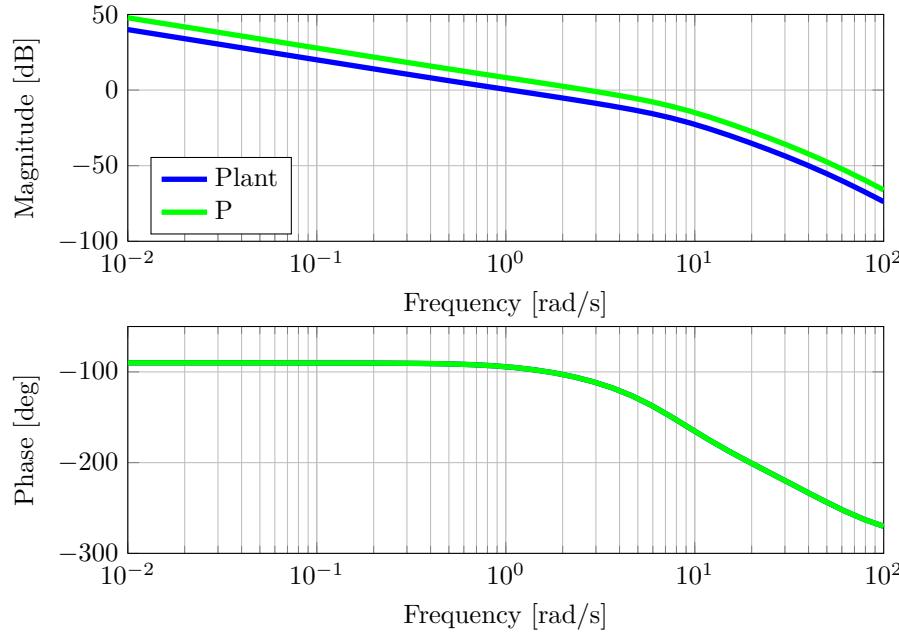
To simplify the analysis, and without loss of generality, the tilt angle controller will only be analyzed in the longitudinal direction with a yaw angle of zero. This is equivalent to controlling only the pitch angle. The quadratic reaction gain will also not form part of the analysis, yielding the simplified controller in Figure 5.13.



**Figure 5.13 – Tilt Angle Controller Block Diagram Used For Analysis**

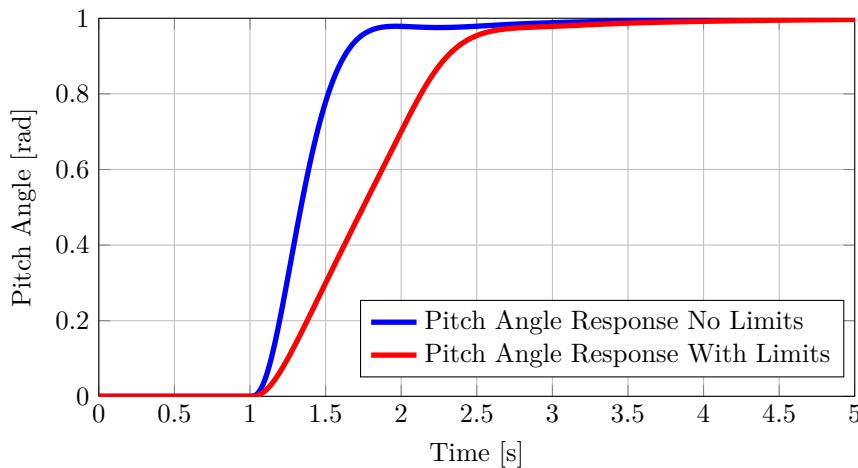
The open-loop transfer function of the simplified block diagram has one free integrator, which originates from the relationship between pitch rate and pitch angle. This makes the system type one for tracking a reference signal. The system will therefore track a step input with a zero steady-state error and a ramp input with a finite steady-state error.

Plotting the tilt angle plant open-loop Bode plot and the open-loop Bode plot where the proportional gain is taken into account shows the phase margin and the gain crossover frequency of the system.



**Figure 5.14** – Open-Loop Bode Plot of Pre-Existing Tilt Angle System

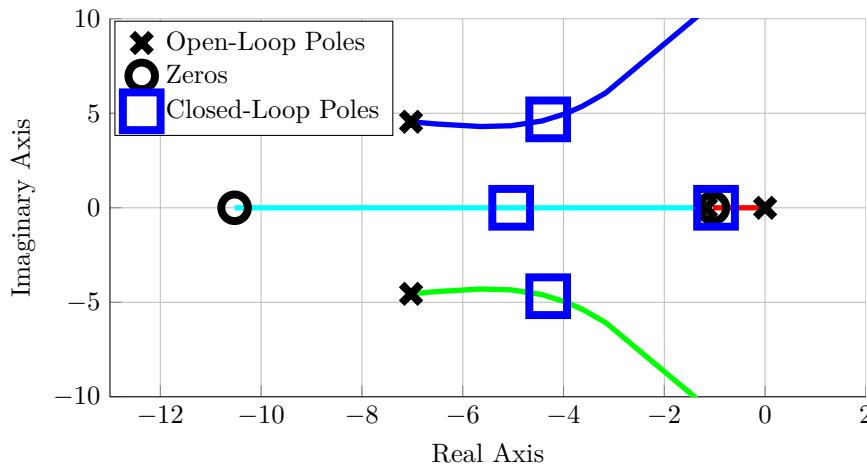
It can be seen how the magnitude plot was moved up to yield a gain crossover frequency of  $2.8 \text{ rad/s}$  which corresponds to a phase margin of  $70^\circ$ . The step response of the tilt angle system to a unit step input can be seen in Figure 5.15.



**Figure 5.15** – Step Response of Pre-Existing Tilt Angle System

Two curves are plotted. The blue curve does not incorporate limiters, showing the response of the linear tilt angle system. The red curve incorporated limiters and, as expected, needs more time to settle. Both curves show an overdamped response, almost approximating a first-order step response. A rise time of 0.48 seconds and a 5% settling time of 0.73 seconds is seen for the linear response. No overshoot is observed and the linear tilt angle system has a time constant of 0.41 seconds.

The root locus of the tilt angle controller is considered next. The root locus can be seen in Figure 5.16, plotted with the open-loop poles and zeros, as well as the dominant closed-loop poles.



**Figure 5.16** – Root Locus of Pre-Existing Tilt Angle System

Two other closed-loop poles and one zero, considered to be less dominant, are not shown. These closed-loop poles are located at  $s = -43$  and  $s = -78$  and the zero is located at  $s = -200$ . The lead compensator's zero is located on the root locus at  $s = -10.53$ . The only new dominant pole to be added from the pitch rate system is the pole located at the origin, which originates from the natural relationship between the pitch rate and pitch angle.

The overdamped step response can be explained by the pole at  $s = -5$ , which gives the system a first-order step response appearance along with the dominant pole pair. Pole/zero cancellation effectively takes place at  $s = -1$ , which is due to the lag compensator in the pitch rate controller, so that the root locus is not affected remarkably.

This concludes the analysis of the tilt angle controller. This controller does not utilize either integral control or derivative control. Presumably the original designers decided that the transient response already had acceptable speed-of-response and damping with pure proportional control, and therefore did not see the need to add derivative control.

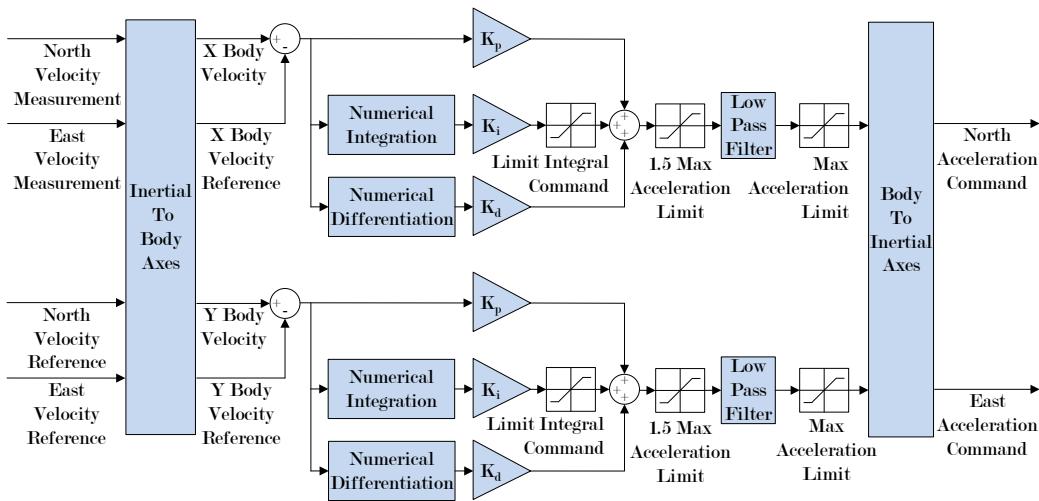
Integral control is most likely also not needed, as there are integral control terms in the control loops before and after the tilt angle controller. Lag compensators are implemented in the angular rate control loops and integral control is implemented in the horizontal velocity controller, as will be seen in the next section. The horizontal velocity controller's integral term enables it to control the tilt angle to achieve the desired velocity.

This controller seems to be fast enough and is definitely damped well enough. The control structure seems adequate as a fast response was obtained with favorable transient behavior and a safe phase margin. Exact steady-state tracking is most probably not a concern with this controller, as there is no sense in controlling the tilt angle precisely if the horizontal velocity controller can control the speed of the vehicle as desired.

### 5.1.1.4 Horizontal Velocity Controller

The horizontal velocity controller, shown in Figure 5.17, is a decoupled system of two PID controllers that control the longitudinal and lateral velocities of the vehicle by commanding longitudinal and lateral accelerations. The velocity reference is received in the inertial axis system and then converted to the body axis system. Reference acceleration commands are then generated by two PID controllers. The acceleration commands are finally low pass filtered and converted back to inertial values.

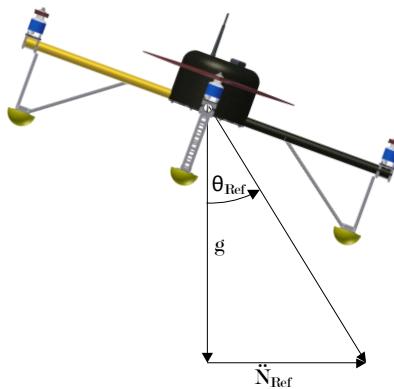
Limiters are used to limit the maximum acceleration command and the maximum integral command. The acceleration limit effectively limits the maximum pitch and roll angles that the vehicle can reach, whereas the integrator limiter combats integrator wind-up. The low pass filter is used to filter the acceleration command which could contain noise due to the derivative control term.



**Figure 5.17 – Pre-Existing Horizontal Velocity Controller Architecture**

To simplify the analysis, and without loss of generality, it will be assumed that the vehicle is flying in a northern direction with a yaw angle of zero. The nose of the vehicle, which is aligned with the x-axis of the aircraft, will be pointing north. The horizontal velocity controller will therefore only function along the x-axis of the vehicle, allowing analysis of only one of the PID controllers.

To enable the application of linear analysis techniques, the relationship between the tilt angle  $\theta$  and the horizontal acceleration  $\dot{N}$  will be linearized. Consider Figure 5.18.



**Figure 5.18 – Acceleration To Angle Relationship**

The relationship between the tilt angle  $\theta$  and the northern acceleration  $\ddot{N}$  is then linearized as follows:

$$\ddot{N} = g \tan(\theta) \quad (5.1.13)$$

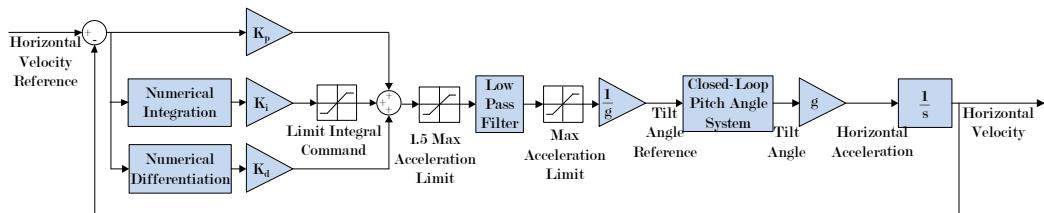
$$\ddot{N} \approx g\theta \quad (5.1.14)$$

and likewise the relationship that translates a commanded horizontal acceleration reference  $\ddot{N}_{REF}$  to a commanded tilt angle reference  $\theta_{REF}$  will be linearized as:

$$\tan(\theta_{REF}) = \frac{1}{g} \ddot{N}_{REF} \quad (5.1.15)$$

$$\theta_{REF} \approx \frac{1}{g} \ddot{N}_{REF} \quad (5.1.16)$$

The linearization performed then yields the simplified block diagram in Figure 5.19 for analyzing the horizontal velocity controller.

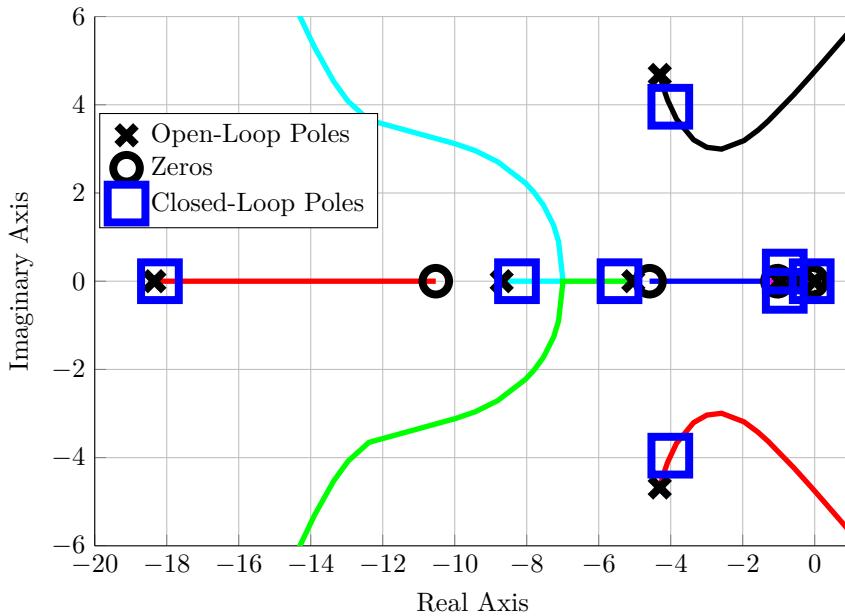


**Figure 5.19 – Horizontal Velocity Controller Block Diagram Used For Analysis**

Intuitively, the proportional gain is used to provide the desired closed-loop bandwidth, whereas the derivative term is used to increase the phase margin. The integral term is needed to negate horizontal force disturbances that can originate from wind and unmodelled aerodynamics of the vehicle.

The system contains a free integrator from the relationship between acceleration and velocity. Another free integrator is present in the system from the integral term. The system is therefore a type two system, which will follow up to a ramp input with zero steady-state error.

The root locus is examined next. Shown in Figure 5.20 is the root locus, with the open-loop poles and dominant closed-loop poles drawn.

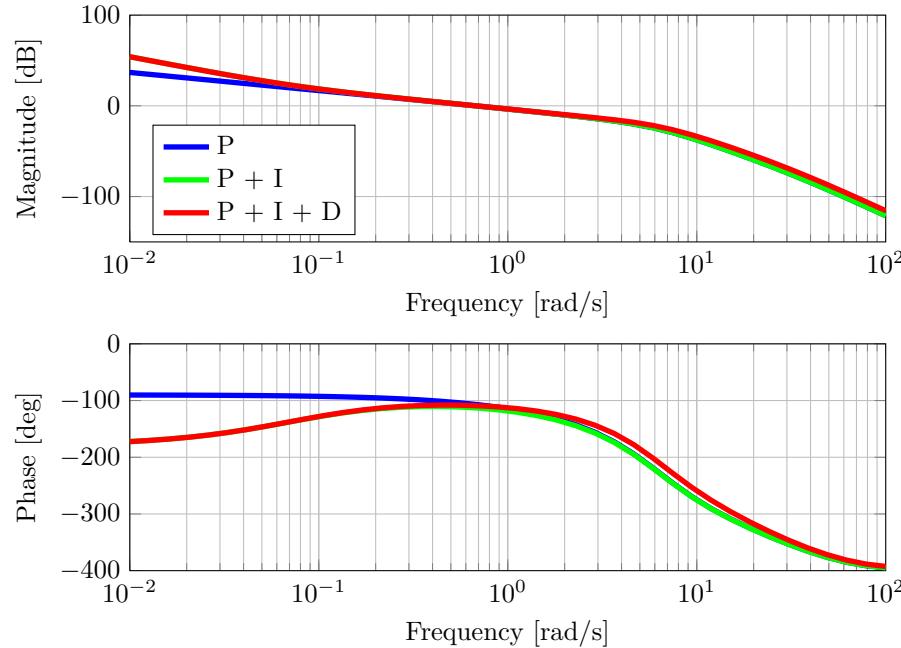


**Figure 5.20 – Root Locus of the Pre-Existing Horizontal Velocity System**

Two new zeros, located at  $s = -4.59$  and  $s = -0.07$ , are introduced into the horizontal system by the PID controller. Two open-loop poles also originate from the PID controller: one pole at the origin, which is the integrator from the integral term, and another pole at  $s = -8.69$ , which comes from the derivative term. Another pole is also introduced at the origin, which originates from the natural relationship between acceleration and velocity.

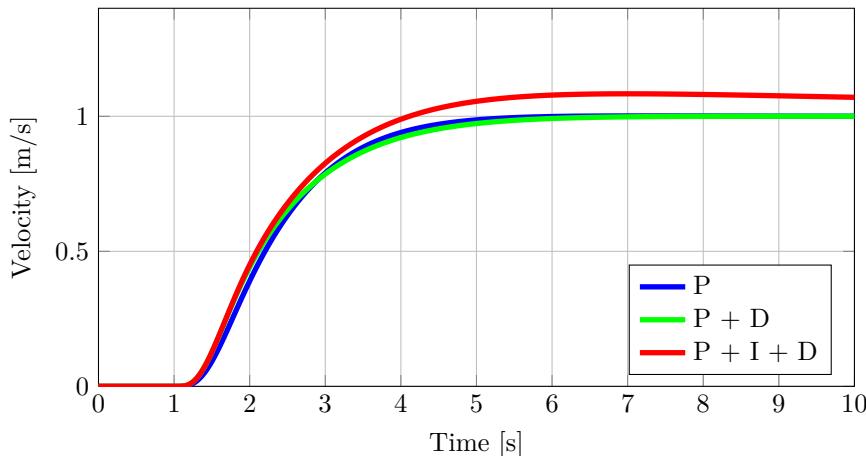
It is difficult to predict the response of the closed-loop system by looking at the root locus, as it is a high order system. The dominant system behavior will arise from the complex pole pairs at  $s = -0.83 \pm 0.22i$  and  $s = -4.01 \pm 3.96i$ , and the host of poles and zeros on the real axis. A cluster of three closed-loop poles is located near the origin. One of these poles undergoes pole/zero cancellation by the one zero near the origin. This leaves the complex pole pair at  $s = -0.83 \pm 0.22i$  to be the most dominant of all the constituents. These two poles are near the real axis, indicating that they will contribute a considerable amount of damping to the system. A step response which is almost like a first-order step response can therefore be expected.

The open-loop Bode plot is shown in Figure 5.21, with each term of the PID controller omitted so that their effect can be seen. The final result is a phase margin of  $70^\circ$  at a gain crossover frequency of  $0.68 \text{ rad/s}$ . The derivative term barely increases the phase margin, possibly indicating that the bandwidth of this system could be increased.



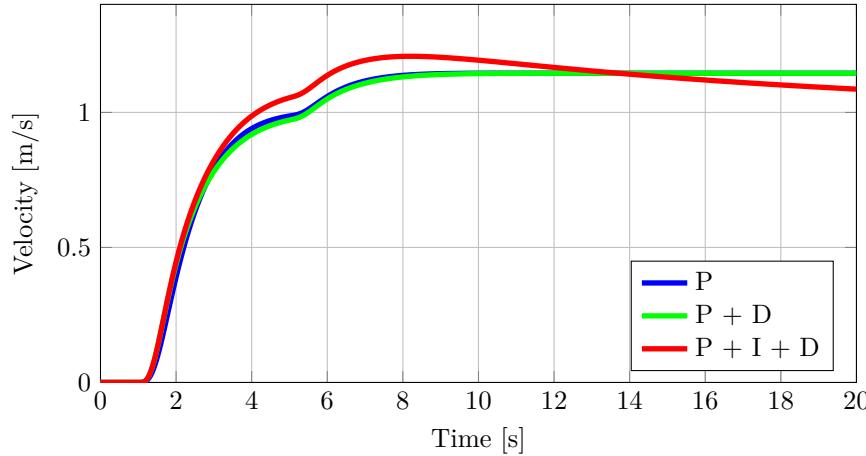
**Figure 5.21** – Open-Loop Bode Plot of the Pre-Existing Horizontal Velocity System

The step response in Figure 5.22, of a linear model of the system without any limiters included, illustrates how each term of the PID controller affects the transient and steady-state characteristics of the horizontal velocity controller:



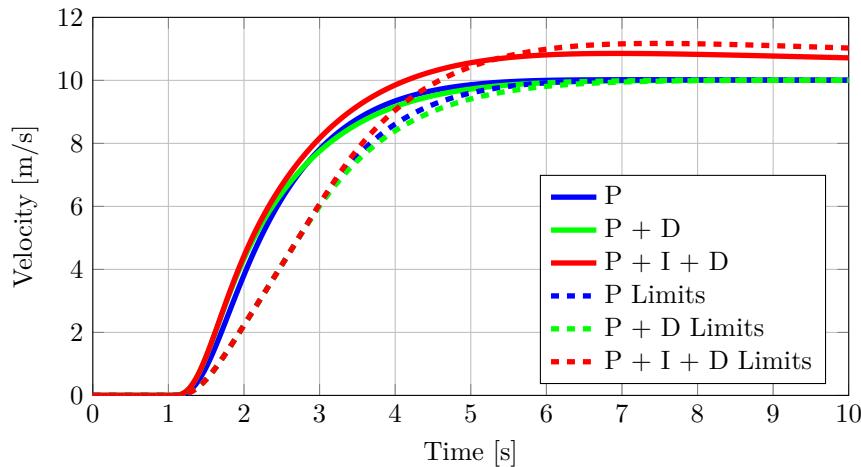
**Figure 5.22** – Step Response of Pre-Existing Horizontal Velocity System

The integrator appears to be very slow and the derivative term does not have a noticeable effect. The rise time of the system is 1.90 seconds and a 5% settling time of 9.71 seconds is observed. The time constant is 1.39 seconds and a maximum overshoot of 8% is seen. The 5% settling time is notably long, and is due to the overshoot caused by the integral term of the controller. The integrator's slow response can be seen in Figure 5.23, where the system is disturbed at  $t = 5$  seconds with an acceleration step of  $0.1 \text{ m/s}^2$ . This acceleration would manifest as a result of a force physically disturbing the aircraft.



**Figure 5.23** – Disturbance Response of Pre-Existing Horizontal Velocity System

The proportional and derivative terms negate some of the disturbance, but the integrator slowly neutralizes the disturbance completely. The integrator will therefore only be effective at eliminating constant wind disturbances and constant drag force disturbance during long periods of translational movement. A step response which includes the limiters in the system is shown in Figure 5.24. The system is given a reference step command of 10 m/s, so that the interaction of the limiters is more noticeable.



**Figure 5.24** – Step Response of Pre-Existing Horizontal Velocity System With Limiters

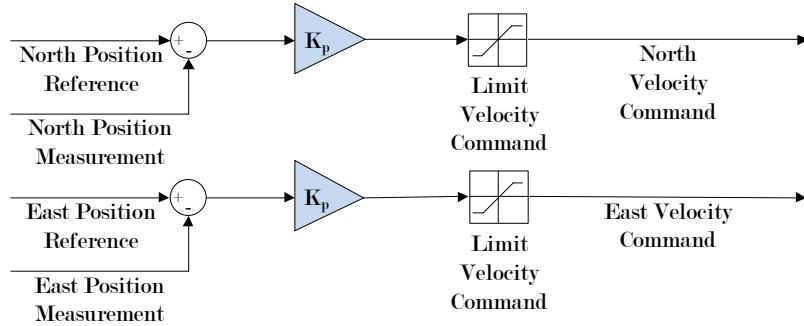
As expected, the system with the limiters responds more slowly than the system without the limiters. A slight increase in overshoot is noticed when the limiters are taken into account, and is most probably due to the slower response allowing more time for the integrator to wind up.

This concludes the analysis of the horizontal velocity controller. The integral term seems to be very slow in negating disturbances in the system, and is therefore presumably only intended for very low frequency disturbances. The derivative control term does not provide any notable increase in the phase margin and does thus not benefit the transient response. Therefore, a derivative term that adds usable phase to the system should allow for the use

of a higher proportional gain, while still maintaining the transient response characteristics. For the first time in the horizontal control system, there appears to be a control loop that does not utilize the aircraft's dynamics to its full potential. This controller can very likely be improved for the task of landing on a moving platform.

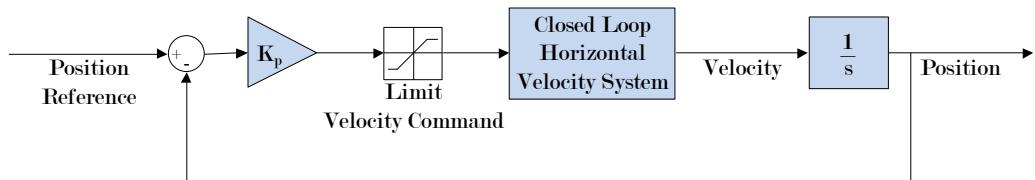
#### 5.1.1.5 Horizontal Position Controller

The horizontal position controllers are analyzed in this section and are the outermost loop controllers in the horizontal control system. The horizontal position controllers can be seen in Figure 5.25 and control the north and east components of the vehicle's position by commanding the north and east components of the vehicle's velocity.



**Figure 5.25** – Pre-Existing Horizontal Position Controller Architecture

These controllers are based on a pure proportional control law, and unlike all the other horizontal controllers which function in the body axis of the aircraft, function in the inertial axis only. The position controller is described as a hover controller and is used for low speed flight and hover. A limiter is used to limit the velocity command to  $3.5 \text{ m/s}$  for large position errors, limiting the flight envelope of the aircraft considerably. Once again only the north position controller will be analyzed, but the analysis is equally applicable to the east position controller. The block diagram used for the analysis can be seen in Figure 5.26.



**Figure 5.26** – Horizontal Position Controller Block Diagram Used For Analysis

Due to the natural integrator originating from the relationship between position and velocity, the system has one free integrator. The system will thus track a step input with zero steady-state error (assuming no measurement errors of the aircraft's velocity) and a ramp input with a finite steady-state error. The velocity error constant for a ramp input with a slope of  $1 \text{ m/s}$  is

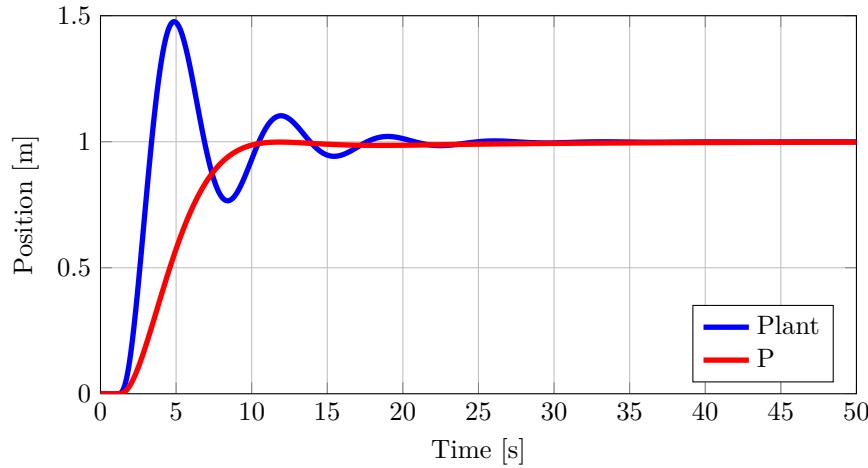
$$K_V = \frac{1.904e23}{7.934e23} = 0.239 \quad (5.1.17)$$

which translates to a steady-state position error of

$$e_{ss} = \frac{1}{K_V} = 4.2m \quad (5.1.18)$$

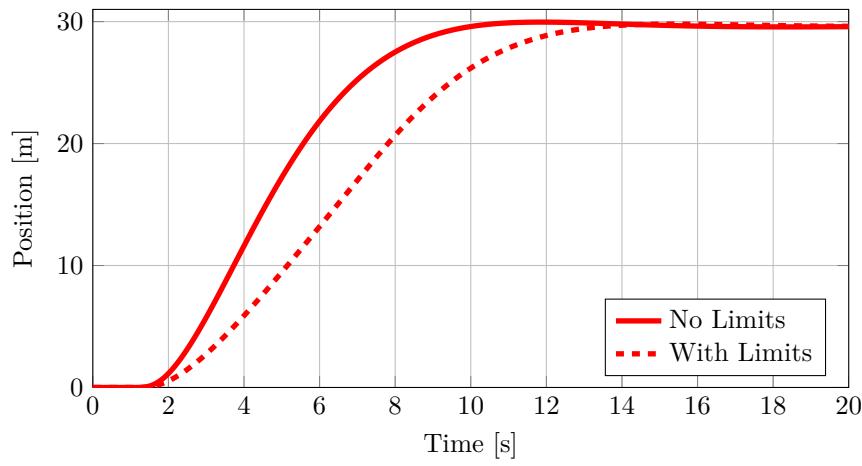
Equation 5.1.18 dictates that the vehicle will not be able to track the position of a moving platform (which can be seen as a position ramp command), clearly indicating that the control strategy needs to be modified for this project.

The step response of the system is looked at next and is plotted in Figure 5.27. The linear system, which does not include any limiters, was given a unity step reference at  $t = 1$  second.



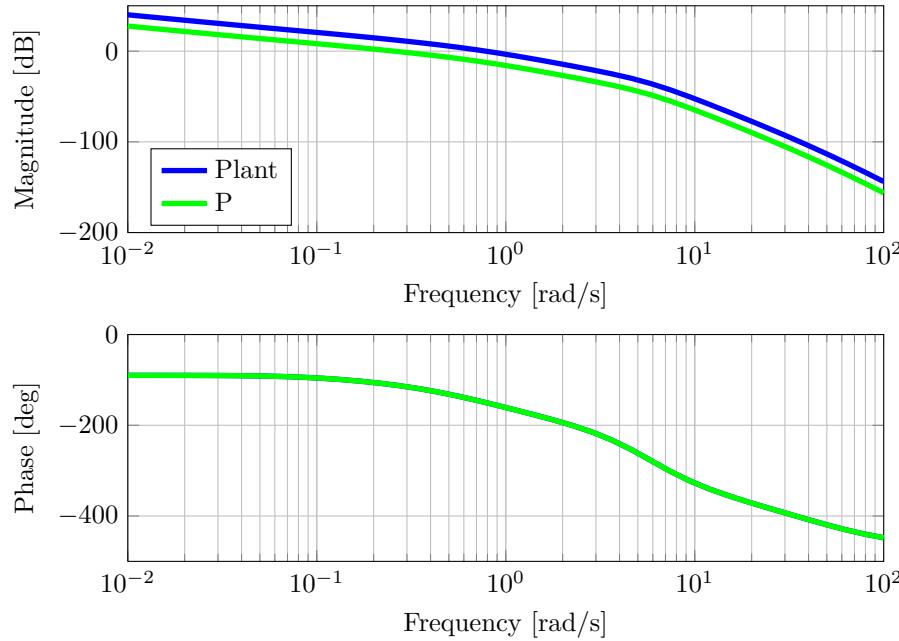
**Figure 5.27** – Step Response of Pre-Existing Horizontal Position System

The plant appears to be underdamped, making the response oscillate. The proportional gain decreased the bandwidth of the system, making it more damped. The time constant is 4.37 seconds and no overshoot is seen. A rise time of 5.27 seconds and a 5% settling time of 7.67 seconds are observed. A step response, including all the limiters in the system, is plotted in Figure 5.28, where a reference position step command of 30 m was issued to make the interaction of the limiters apparent.



**Figure 5.28** – Step Response of Pre-Existing Horizontal Position System With Limiters

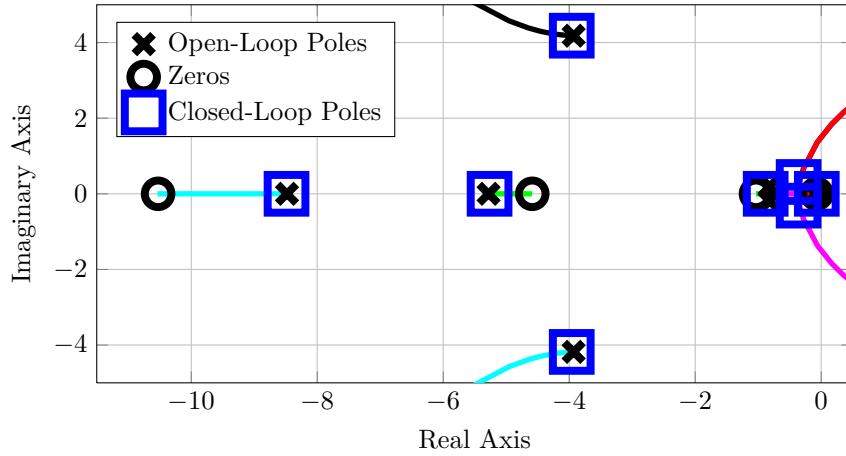
As expected, the system without limiters settles faster. With limiters the system takes approximately 3.75 seconds longer to settle. The open-loop Bode plot is investigated next and is plotted in Figure 5.29.



**Figure 5.29** – Open-Loop Bode Plot of Pre-Existing Horizontal Position System

The proportional gain decreases the gain crossover frequency of the plant to yield a phase margin of  $70^\circ$  at a gain crossover frequency of  $0.2 \text{ rad/s}$ , indicating that the system is optimally damped.

The root locus is finally investigated. The root locus, open-loop poles and zeros, and closed-loop poles are plotted in Figure 5.30.



**Figure 5.30** – Root Locus of Pre-Existing Horizontal Position System

The root locus has been augmented with only one pole, located at the origin, compared to the root locus of the horizontal velocity system. This integrator originates from the natural relationship between velocity and position. Poles and zeros which are considered less dominant and which lie far to the left of the root locus are not shown. Predicting the response is difficult as the system is of a high order. However, looking at the cluster of poles and zeros near the origin can give some indication. Four poles are located near the origin, two of which nearly undergo pole/zero cancellation. A complex pole pair remains at  $s = -0.36 \pm 0.31i$ , which will dictate the dominant behavior of the system. As these poles are near the real and imaginary axes, a low-frequency response with a considerable amount of damping can be expected.

This concludes the analysis of the horizontal position controller. This strategy has worked for reference position steps, but cannot be used for tracking a moving platform with precision. The architecture of the controller will therefore need to be modified for this project. No form of derivative control was implemented, as the designers must have believed that the transient response and the current bandwidth are satisfactory. Similarly, no integral control is implemented, as disturbance rejection was not most probably not considered a problem. However, for this project, not only the architecture of the controller will have to change, but the speed-of-response must possibly also be increased to be able to land within the accurate bounds required for this project.

### 5.1.2 Yaw Control System

The system responsible for yaw control is investigated next. The yaw control system consists of a yaw rate controller and a yaw angle controller. The inner yaw rate controller runs at a sample rate of 100 Hz and receives its reference from the yaw angle controller, which runs at a sampling rate of 50 Hz. The yaw angle of the vehicle can be commanded independently of the translation direction of the vehicle, as explained in Section 5.1.1.3.

#### 5.1.2.1 Yaw Dynamics

A linear model of the yaw dynamics is firstly derived for use in the analysis. The state variables for the yaw dynamics are chosen as  $N$  and  $R$ , where  $N$  is the yawing moment and  $R$  is the yaw rate of the vehicle. A virtual rudder actuator named  $\delta_R$  is also defined, which will cause a yawing moment when actuated:

$$\delta_R = \frac{r_D}{R_{LD}}(-T_1 + T_2 - T_3 + T_4) \quad (5.1.19)$$

$R_{LD}$  is the rotor's lift to drag ratio and  $r_D$  the chord length where the rotor drag force is exerted. As Equation 5.1.19 is just the superposition of the thrust developed by all the rotors,  $\delta_R$  will have the same lag dynamics as a single rotor, which was derived in Chapter 4:

$$\dot{\delta}_R = -\delta_R \tau + \delta_{R_R} \quad (5.1.20)$$

The final equation forming part of the yaw dynamics is the relationship between angular acceleration to applied moment:

$$N = I_{zz} \dot{R} \quad (5.1.21)$$

From Equation 5.1.20 and 5.1.21 the state space equation can be derived:

$$\begin{bmatrix} \dot{N} \\ \dot{R} \end{bmatrix} = \begin{bmatrix} -\frac{1}{\tau} & 0 \\ \frac{1}{I_{zz}} & 0 \end{bmatrix} \begin{bmatrix} N \\ R \end{bmatrix} + \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \delta_{R_R} \quad (5.1.22)$$

$$y = [0 \quad 1] \begin{bmatrix} N \\ R \end{bmatrix} \quad (5.1.23)$$

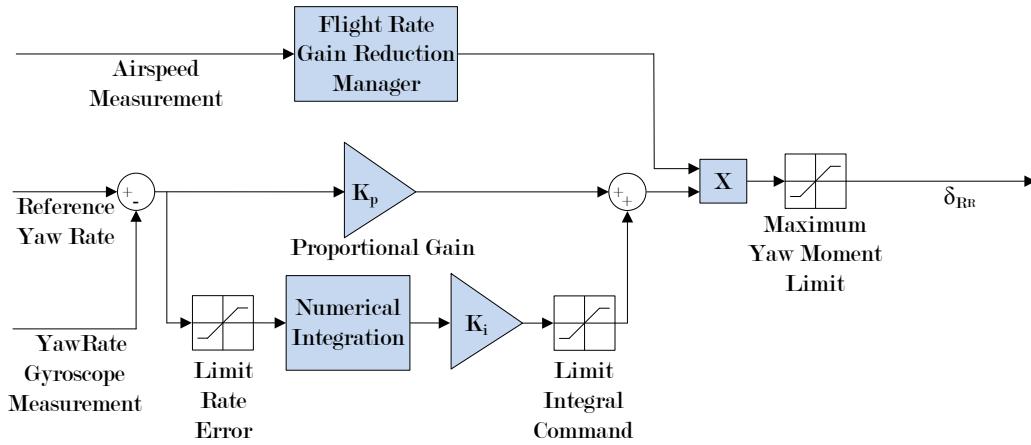
This state space equation is converted in Appendix C to a transfer function,

$$G(s) = \frac{\frac{1}{\tau I_{zz}}}{s + \frac{1}{\tau}} \quad (5.1.24)$$

Equation 5.1.24 states that the plant has a pure integrator and a pole at  $s = -\frac{1}{\tau}$ . Now that the vehicle yaw dynamics have been modelled, the yaw rate controller can be analyzed.

### 5.1.2.2 Yaw Rate Controller

The architecture of the yaw rate controller can be seen in Figure 5.31, as it was determined from the C code implemented on the OBC. The yaw rate controller uses a PI control law, as well as the non-linear component that was seen in the analysis of the roll and pitch rate controllers, namely the Flight Rate Gain Reduction Manager. It is unknown why this is the architecture of this controller and what design procedure was followed.



**Figure 5.31 – Pre-Existing Yaw Rate Controller Architecture**

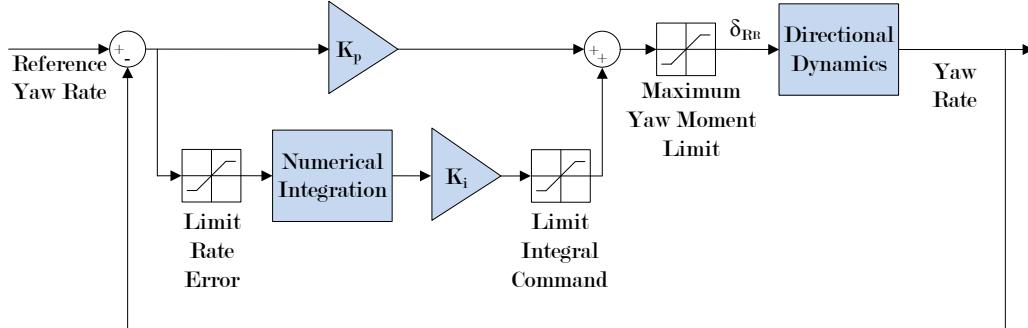
The function of the non-linear component is repeated for convenience. From previous research it was seen that the gain of the yaw rate plant increases as the airspeed of the vehicle increases. The Flight Rate Gain Reduction Manager reduces the gain of the yaw rate controller during periods of fast flight. The gain reduction is scheduled as a function of the airspeed measurement. When the vehicle is near hover, the gain factor is unity, and as the vehicle gains airspeed the gain factor is reduced. By doing this, the Flight Rate Gain Reduction Manager effectively keeps the bandwidth of the yaw rate system constant during hover and fast forward flight. The Flight Rate Gain Reduction Manager will, however, not form part of this analysis, as its parameters were determined empirically.

The limits exhibited in Table 5.2 also form part of the controller's architecture.

**Table 5.2 – Pre-Exisiting Yaw Rate Controller Limits and Their Functions**

Limit	Function	Value
Maximum Moment Limit	Prevent actuator saturation	1.25 Nm
Integral Command Limit	Limit the maximum disturbance rejection	0.58 Nm
Rate Error Limit	Constrain integrator wind-up speed	14 deg/s

For the purpose of the yaw rate controller analysis, the block diagram in Figure 5.32 will be used.



**Figure 5.32 – Yaw Rate Controller Block Diagram Used for Analysis**

Upon inspection of the controller it can be assumed that the proportional gain is used to obtain the desired bandwidth and transient characteristics. The integrator is used to improve steady-state tracking and to compensate for disturbances and unmodelled dynamics. Unmodelled dynamics like the differential yaw moment caused by the rotors, and constant yaw torques induced by external factors like wind, are mitigated by the integral term.

The control structure is analyzed next and the design procedure that was followed can be deduced. The proportional gain was most probably determined first in order to achieve the desired bandwidth. The integrator was then added to improve steady-state characteristics. This structure, like the roll rate controller and pitch rate controller, enables control authority to be assigned to the integral and proportional terms by use of limiters.

The integral term can command a maximum yaw moment of  $0.58 \text{ Nm}$ , thereby limiting the magnitude of the disturbance rejection that can be done. The yaw rate error that is fed into the integrator is also limited to ensure that the integral term does not wind up unnecessarily quickly, and to ensure that the integrator only integrates low-magnitude yaw rate errors. Logically, the total yaw moment command is also limited to ensure that the actuators are not saturated.

The proportional term will have full control authority in the absence of disturbances, being able to command the full  $1.25 \text{ Nm}$  moment if the integral term's command is  $0 \text{ Nm}$ . This, however, changes as the integral term starts providing disturbance rejection. When the integral term is providing its maximum moment command of  $0.58 \text{ Nm}$ , the proportional term can only contribute  $0.68 \text{ Nm}$  to the total yaw moment command.

The system type and steady-state characteristics are investigated next. The loop transfer function of the block diagram in Figure 5.32, when expressed from the reference yaw rate to the output yaw rate, is shown in Equation 5.1.25:

$$G(s)_{YawRateOL} = \frac{18.72s + 2.619}{s^3 + 8s^2} \quad (5.1.25)$$

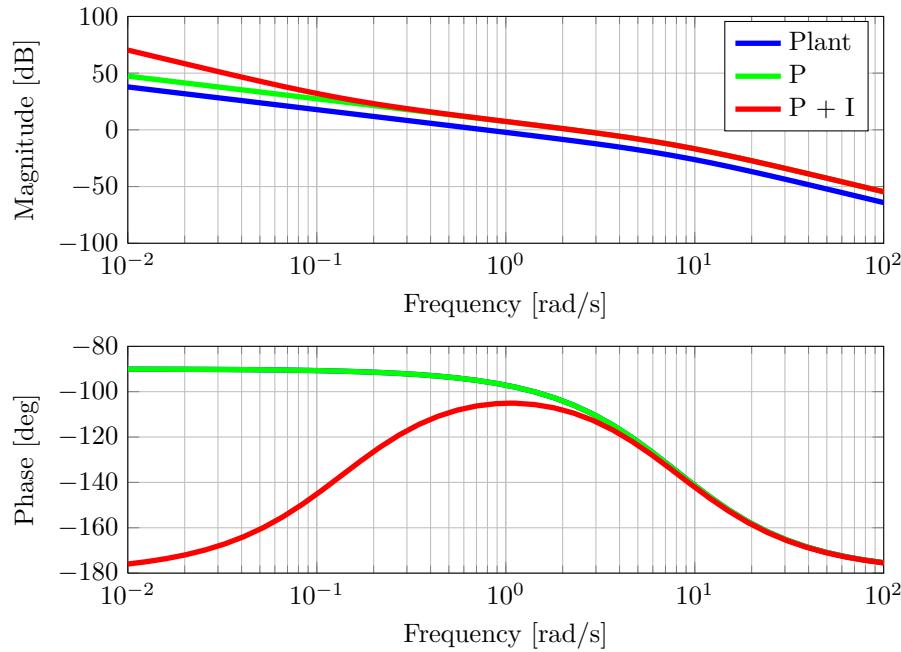
Two free integrators are present, making the system type two for tracking. The one integrator arises from the linear yaw dynamics of the vehicle and the other integrator from the controller's integral term. The vehicle will therefore track a ramp yaw rate input with a steady-state error of zero. The position error constant and the velocity error constant of the system are therefore zero.

Like the roll rate control loop and the pitch rate control loop, the yaw rate control loop is the innermost loop and should respond as fast as possible. Disturbances in the yaw system will be negated by this control loop the fastest. The bandwidth of this controller should therefore be as high as the dynamics of the vehicle allow, but with a safe gain and phase margin. The physical factors that limit the bandwidth of the yaw rate dynamics are the inertia of the rotors, the electrical properties of the motors, the inertia of the vehicle chassis and the sampling rate of the control loop.

In the case of the yaw dynamics, the yaw moment that can be generated by the rotors is less than the pitch and roll moment. Around the body x-axis and the body y-axis, differential motor thrusts generate the moments required to pitch and roll the aircraft, while around the body z-axis the yaw moment is generated by aerodynamic drag described by the lift to drag ratio ( $R_{LD}$ ) of the rotors. This drag force is exerted at the chord length  $r_D$  from the center of the rotor disks. The moment of inertia of the vehicle chassis is also higher around the yaw axis (body z) than the roll (body x) and pitch (body y) axes. It can therefore be expected that the attainable bandwidth of this control loop will be lower than the bandwidth of the pitch rate control loop and roll rate control loop.

Like the pitch rate and roll rate control loops, the yaw rate controller should also be able to compensate for steady-state errors. Strong disturbance rejection should therefore also be done in this loop by the integral term. Strong disturbance rejection is apparent in that the maximum contribution to the yaw moment command is  $0.58 \text{ Nm}$  from the integral term, slightly less than the  $0.67 \text{ Nm}$  that can be commanded from the proportional term.

The open-loop Bode plot of the yaw rate system is looked at first, and can be seen in Figure 5.33.



**Figure 5.33 – Open-Loop Bode Plot of Pre-Existing Yaw Rate System**

The proportional gain increased the gain crossover frequency from  $0.77 \text{ rad/s}$  to  $2.25 \text{ rad/s}$  at which the phase margin is  $70^\circ$ . The integral term amplified the steady-state gain as expected and did not change the gain crossover frequency.

The gain crossover frequency of the yaw rate control loop ( $2.25 \text{ rad/s}$ ) is four times less than the gain crossover frequency of the pitch and roll rate control loops ( $10 \text{ rad/s}$ ). The gain crossover frequency of the yaw rate control loop could have been increased by a higher proportional gain and adding a lead compensator to maintain a  $70^\circ$  phase margin. The original designers of the control system must, however, have been satisfied with the performance of the controller.

The step response due to each term of the controller can be seen in Figure 5.34. The linear model was used, without any limiters.

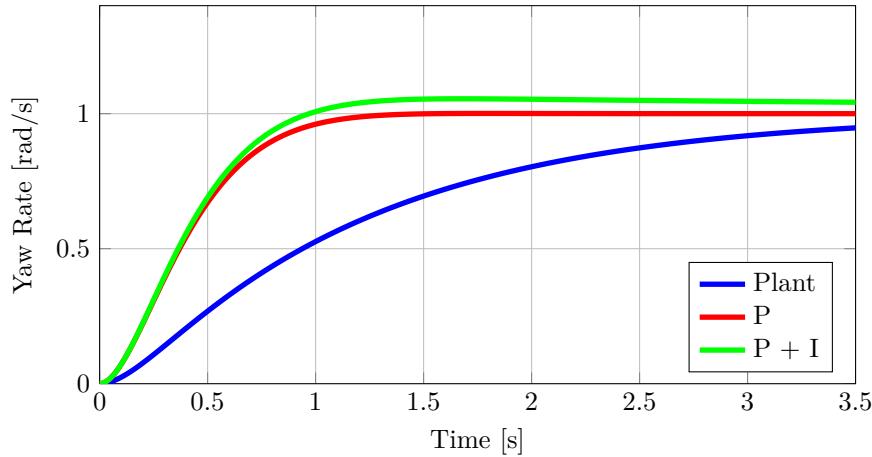


Figure 5.34 – Step Response of Pre-Existing Yaw Rate System

The proportional term settles with zero steady-state error. Some overshoot is observed when the integral term is added, and it settles slowly. A rise time of 0.61 seconds and a rather long 5% settling time of 2.41 seconds is seen. The long settling time is a result of the overshoot observed, of which the maximum is 5.5%.

Disturbance rejection of the system is investigated next. A step response is plotted in Figure 5.35, where the system is disturbed by a constant moment of  $0.5 \text{ Nm}$  at  $t = 3$  seconds.

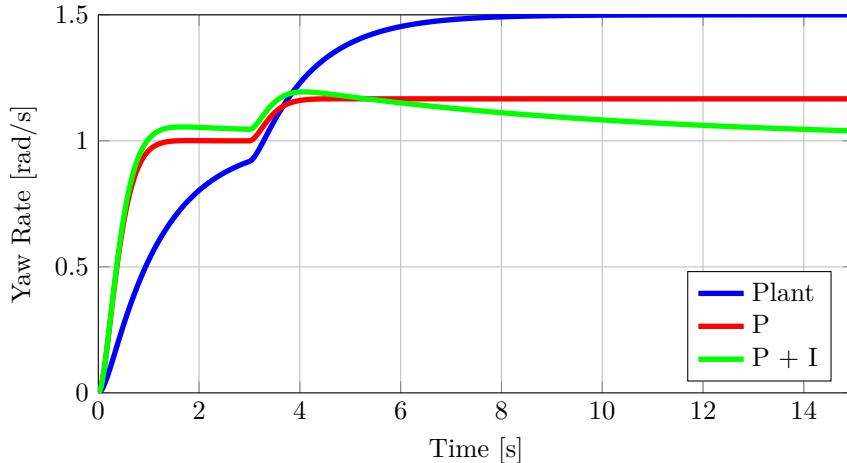
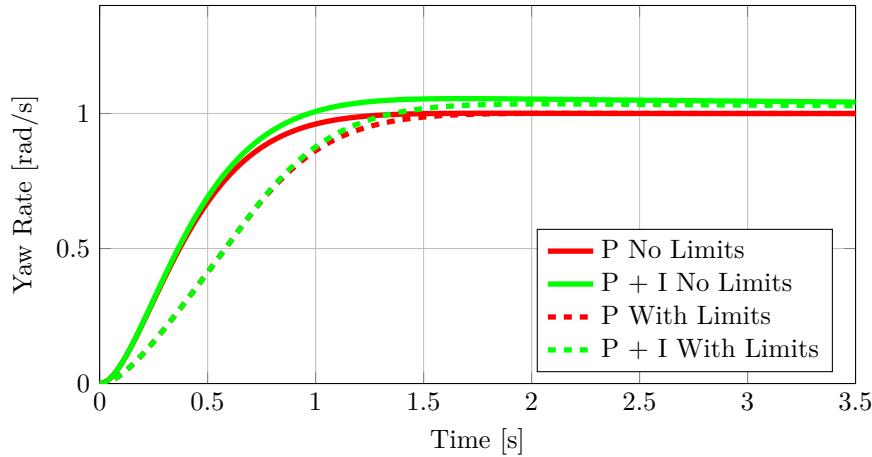


Figure 5.35 – Disturbance Response of Pre-Existing Yaw Rate System

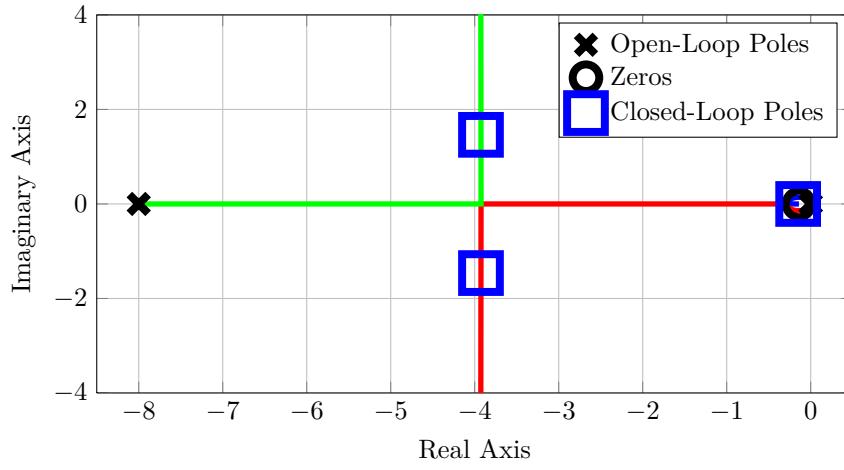
The proportional term negates some of the disturbance, while the integrator very slowly compensates for the constant yaw moment. A closer look is also taken at the effect of the limiters. The step response in Figure 5.36 illustrates the working of the limiters of the yaw rate controller.



**Figure 5.36** – Step Response of Pre-Existing Yaw Rate System With Limiters

When the limiters are taken into account, the systems settles slower and less overshoot is seen from the integral term. The decrease in overshoot is due to the limit in front of the integrator, which limits the yaw rate error to  $\pm 14^\circ/s$ .

The root locus is lastly looked at, and is plotted in Figure 5.37 with the open-loop poles and zeros, and the closed-loop poles.



**Figure 5.37** – Root Locus of Pre-Existing Yaw Rate System

Two open-loop poles are located at the origin. One originates from the yaw rate dynamics and the other from the integral term of the PI controller. A zero is also plotted at  $s = -0.13$ , which also originates from the the PI controller. The remaining pole, located at  $s = -8$ , originates from the lag dynamics of the rotors. Looking at the closed-loop poles, pole/zero

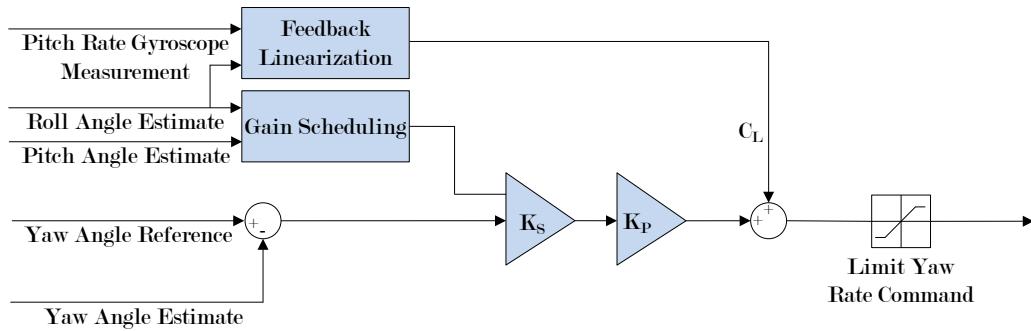
cancellation takes place at  $s = -0.13$ , leaving the dominant behavior of the yaw system to come from the complex pole pair at  $s = -3.93 \pm 1.46i$ .

This concludes the analysis of the yaw rate controller. This controller uses the dynamics of the system to its full potential while maintaining a  $70^\circ$  phase margin. A lead compensator or derivative control could have allowed for a higher attainable bandwidth, but the designers were most probably satisfied with the controller as it is. It is also possible that this control loop can physically not utilize more actuation, as a small yawing moment can require large thrust changes from the rotors. A higher bandwidth is therefore possibly not practically attainable.

The integral term negates disturbances very slowly. This poses the question of whether the integral term can respond to disturbances quickly enough. It will work sufficiently for finding the differential moment trim due to physical inconsistencies between the four rotors, but it will respond too slowly to external disturbances like varying wind or abrupt aerodynamic changes.

### 5.1.2.3 Yaw Angle Controller

The yaw angle controller architecture is shown in Figure 5.38, as it was determined from the embedded C code on the OBC. The controller controls the yaw angle of the vehicle by commanding a reference yaw rate to the yaw rate controller. The yaw angle controller receives its reference from the Ground Station or from the navigation algorithms on the aircraft.



**Figure 5.38 – Pre-Existing Yaw Angle Controller Architecture**

The yaw angle controller is a proportional controller with two non-linear components. The one non-linear component is gain scheduling and the other component feedback gain linearization. The gain scheduling is a function of the roll and pitch angles of the aircraft. Equation 5.1.26 expresses  $K_S$ , the gain scheduling parameter:

$$K_S = \frac{\cos(\theta)}{\cos(\phi)} \quad (5.1.26)$$

$K_S$  is unity when the roll angle  $\phi$  and pitch angle  $\theta$  are equal in magnitude. When the roll angle magnitude is more than the pitch angle's,  $K_S$  is more than unity. When the roll angle magnitude is less than the pitch angle magnitude,  $K_S$  is less than unity. It is not known why the gain scheduling is done with this relationship, but it seems that this implementation was done to compensate for a difference in the moment of inertia around the body x-axis and body y-axis. Therefore, if the vehicle is pitched more than it is rolled, the bandwidth

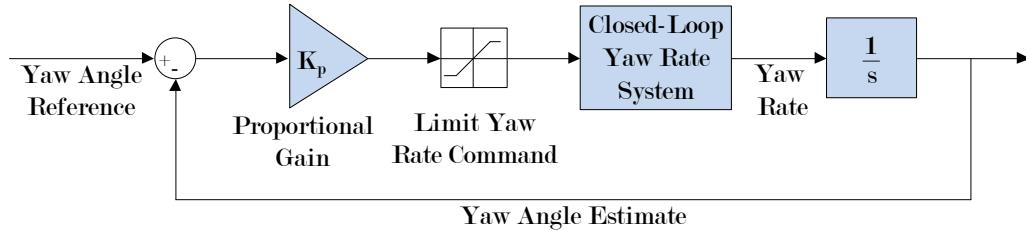
of the yaw angle system is decreased. If the vehicle is rolled more than it is pitched, the bandwidth of the yaw angle system is increased. This difference in the moment of inertia is due to the electronic warfare payload antennae used on the aircraft, which are mounted on the body x-axis of the aircraft. The antennae make the moment of inertia around the body y-axis more than around the body x-axis.

The gain scheduling will not form part of this analysis, as the aircraft will not have a different moment of inertia around the body x-axis and body y-axis. The antennae will not be mounted on the aircraft during this project, making the moment of inertia equal around the two axes.

The feedback gain linearization is a function of the pitch rate and the roll angle. The linearized term  $C_L$  is added to the yaw rate command. The linearized term is determined by Equation 5.1.27:

$$C_L = -\tan(\phi)Q \quad (5.1.27)$$

This non-linear component will also not form part of the linear analysis. The simplified block diagram that is used for the analysis is shown in Figure 5.39.



**Figure 5.39 – Yaw Angle Controller Block Diagram Used For Analysis**

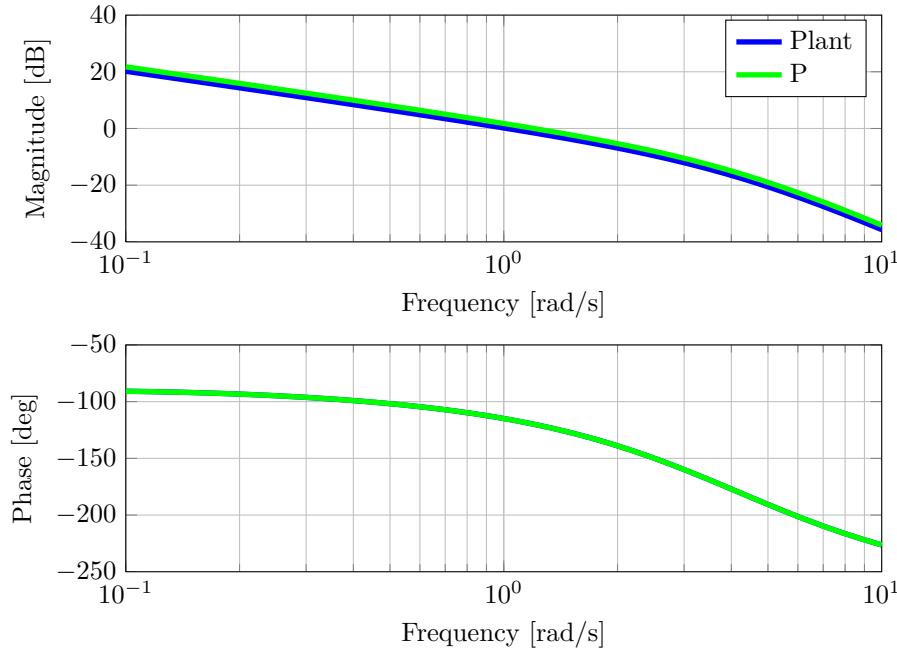
The yaw rate command from the yaw angle controller is limited to  $\pm 1.047 \text{ rad/s}$ , or  $60^\circ/\text{s}$ , ensuring that the aircraft does not yaw excessively fast for large yaw angle errors. Due to the natural integrator originating from the relationship between the yaw angle and yaw rate, the system has one free integrator. The system is thus type one for tracking, and will track a step input with zero steady-state error and a ramp input with a finite steady-state error. The velocity error constant for tracking a ramp yaw angle reference of  $1 \text{ rad/s}$  is

$$K_V = \frac{3.143}{2.619} = 1.2 \quad (5.1.28)$$

indicating that the steady-state error due to a unit ramp input is

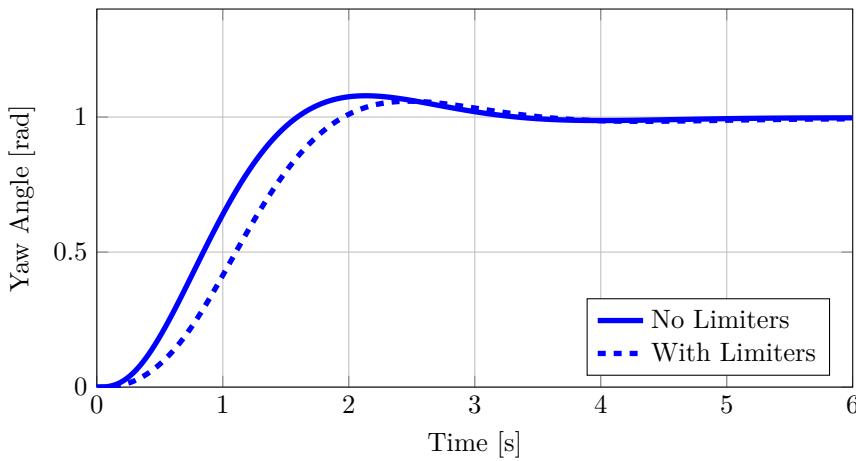
$$e_{ss} = \frac{1}{K_V} = 0.83 \text{ rad} \quad (5.1.29)$$

The Bode plot is investigated next and is plotted in Figure 5.40. Plotting the unity gain open-loop Bode plot and the open-loop Bode plot taking the proportional gain into account shows the phase margin and the gain crossover frequency of the system.



**Figure 5.40 – Open-Loop Bode Plot of Pre-Existing Yaw Angle System**

The magnitude plot was moved up slightly by the proportional gain, changing the gain crossover frequency from  $1.02 \text{ rad/s}$  at unity gain to yield a gain crossover frequency of  $1.20 \text{ rad/s}$ . The increased gain crossover frequency corresponds to a phase margin of  $60^\circ$ , indicating that the system is not optimally damped. The unit step response of the linear system, which does not take any limiters into account, and the non-linear system which does, can be seen in Figure 5.41,



**Figure 5.41 – Step Response of Pre-Existing Yaw Angle System**

An underdamped response is seen, with the linear system showing a peak overshoot of approximately 8%. The system settles within 5% of its final value after a time of 2.52 seconds. A rise time of 0.99 seconds is seen and the time constant of the system is also 0.99 seconds. As expected, the system with the limiters responds slower, lagging slightly

behind the linear system. The non-linear system, however, displays less overshoot, showing a maximum overshoot of 6%. The root locus is lastly investigated and is shown in Figure 5.42. The open-loop poles, open-loop zeros and closed-loop poles are drawn.

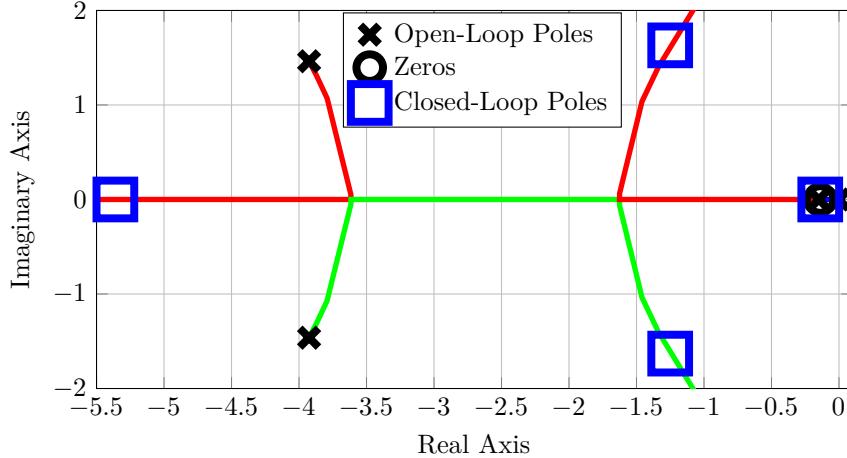


Figure 5.42 – Root Locus of Pre-Existing Yaw Angle System

A pole is introduced into the system which is located at the origin. This integrator originates from the natural relationship between the yaw rate and yaw angle. Pole/Zero cancellation effectively takes place on the real axis at  $s = -0.14$ . The dominant behavior of the system comes from the complex pole pair as  $s = -1.25 \pm 1.63i$  and the pole on the real axis at  $s = -5.36$ .

This concludes the analysis of the yaw angle system. This is the first control loop that is seen on the aircraft that was not designed with a phase margin of  $70^\circ$ . The original designers must have felt that the bandwidth of this controller is sufficiently fast and that the transient response is satisfactory. Another factor that might have played a role in deeming the speed-of-response of this control system satisfactory is the dynamics of the aircraft. Generating a yaw moment requires more control effort than generating a pitching or a rolling moment. Increasing the bandwidth of this system or improving its transient characteristics might have required too much actuation from the rotors, compromising the available control effort to stabilize the aircraft.

The yaw control system does appear to be adequate to achieve the goal of this project. In Section 1.2 it was stated that the moving platform will have a constant orientation, which implies that the heading will be fixed. The heading of the aircraft will therefore also be constant, only requiring that the yaw control system reject disturbances. The outer yaw angle control loop can track a constant yaw angle with zero steady-state error, but cannot reject disturbances, as it has no integrator. However, if disturbances are present, small deviations in the heading of the aircraft will not have a noticeable effect on its translating motion, as its translational control is independent of its yaw angle.

### 5.1.3 Vertical Control System

The control system responsible for vertical control is investigated in this section. The entire vertical control system runs at a sample rate of  $50\text{ Hz}$  and consists of the Normal Specific Acceleration (NSA) controller, the climb rate controller and the altitude controller. The inner loop controller is the NSA controller which receives its reference from the climb rate controller. The climb rate controller finally receives its reference from the altitude controller.

The altitude controller is commanded by the Ground Station or by navigation algorithms. Before the vertical controllers can be analyzed, the heave dynamics are derived.

### 5.1.3.1 Heave Dynamics

The state variable of the heave dynamics is chosen as  $T$ , where  $T$  is the total thrust generated by the four rotors. A thrust command  $\delta_T$  is also defined, which will cause a vertical force if actuated:

$$\delta_T = T_1 + T_2 + T_3 + T_4 \quad (5.1.30)$$

As Equation 5.1.30 is a superposition of the thrust generated by rotors one to four,  $\delta_T$  will have the same lag dynamics derived in Chapter 4 for a single rotor:

$$\dot{\delta}_T = -\delta_T \tau + \delta_{T_R} \quad (5.1.31)$$

The final equation forming part of the heave dynamics is Newton's second law, which describes the acceleration obtained by a force imbalance. Expressing this law using body axis notation yields Equation 5.1.32:

$$Z = m\ddot{W} \quad (5.1.32)$$

Equations 5.1.31 and 5.1.32 can be represented in state space form as:

$$[\dot{T}] = [-\frac{1}{\tau}] [T] + [\frac{1}{\tau}] \delta_{T_R} \quad (5.1.33)$$

$$y = [-\frac{1}{m}] [T] \quad (5.1.34)$$

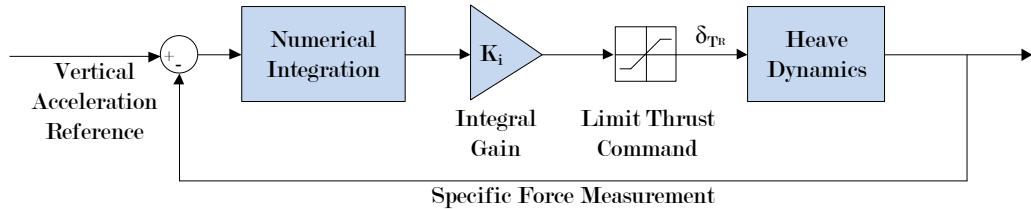
The following transfer function from thrust reference command  $\delta_T$  to normal specific acceleration  $C$  can be derived (see Appendix C for full derivation):

$$G(s) = \frac{C(s)}{\delta_T(s)} = \frac{-\frac{1}{\tau m}}{(s + \frac{1}{\tau})} \quad (5.1.35)$$

This transfer function describes the normal specific acceleration of the vehicle in body axes due to the thrust applied. The negative gain of the transfer function indicates that, according to the axis definition used, a negative vertical acceleration will be induced by the thrust of the rotors. The plant has a single pole located at  $s = -\frac{1}{\tau}$  and no zeros. Now that the heave dynamics are known, the analysis of the vertical control system can be done.

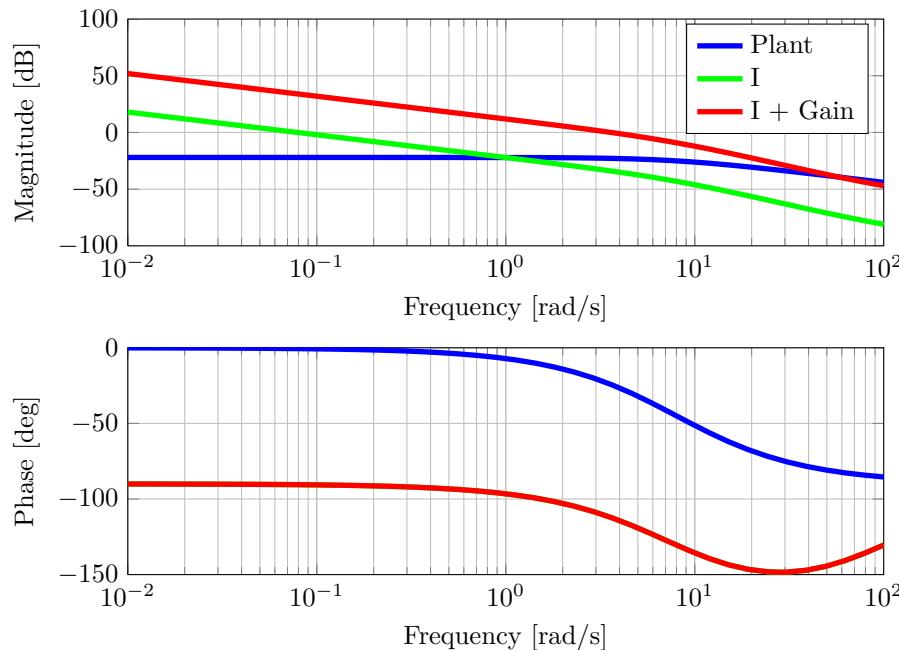
### 5.1.3.2 NSA Controller

The NSA controller controls the specific acceleration normal to the horizontal plain of the aircraft. The controller, which can be seen in Figure 5.43, uses a pure integral control law. Numerical integration of the vertical acceleration error is performed and then multiplied by the integral gain.



**Figure 5.43** – Pre-Existing NSA Controller Architecture

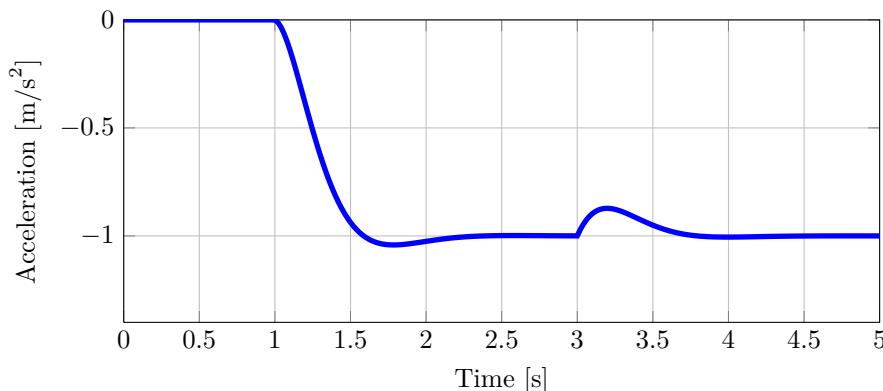
A limiter is used to limit the maximum and minimum thrust commands. The maximum limit will ensure that the NSA controller does not saturate the actuators, which will compromise functioning of the pitch, roll and yaw rate controllers. The lower limit will ensure that the aircraft does not fall to the ground too fast. The maximum and minimum thrust limits are defined as 188 N and 92 N respectively. The open-loop Bode plot of the heave dynamics, the dynamics with the integrator added and the dynamics with the integrator and the integral gain added can be seen in Figure 5.44. The entire phase plot is moved down  $90^\circ$  and



**Figure 5.44** – Open-Loop Bode Plot of Pre-Existing NSA System

low frequencies are amplified by the addition of the integrator. Adding the integral gain moves the magnitude plot up, finally yielding a phase margin of approximately  $68^\circ$  at a gain crossover frequency of  $3.6 \text{ rad/s}$ . The step response is looked at next.

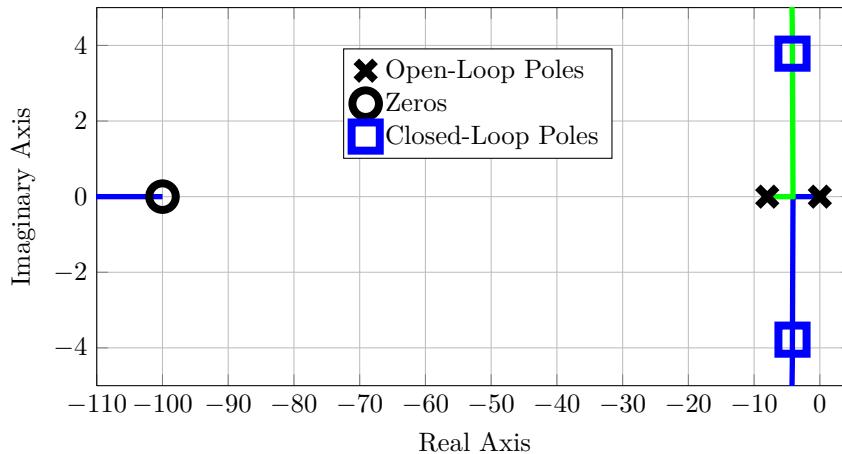
Consider the step response in Figure 5.45. The step response of the linear system is plotted, without the influence of any limiters. The vehicle is commanded to accelerate upward at a rate of  $1 \text{ m/s}^2$  at  $t = 1$  second and disturbed by a constant negative vertical force of  $20 \text{ N}$  at  $t = 3$  seconds.



**Figure 5.45** – Step Response and Disturbance Response of Pre-Existing NSA System

The system's response is the fastest of all the vertical controllers, with a rise time of 0.38 seconds and a 5% settling time of 0.51 seconds. A maximum overshoot of 4.2% is seen and the time constant of the system is 0.30 seconds. Strong disturbance rejection can be noted at  $t = 3$  seconds, where the controller negates the disturbance. This disturbance will primarily arise from external forces like wind and loss in vertical thrust due to rolling, pitching and yawing actuation.

The root locus of the controller is plotted in Figure 5.46 with the location of the open-loop and closed-loop poles indicated.



**Figure 5.46 – Root Locus of Pre-Existing NSA System**

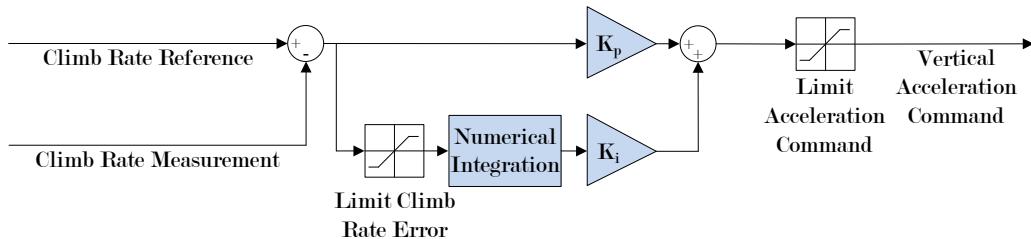
The plant open-loop pole is seen at  $s = -8$  which originates from the lag dynamics of the rotors. Another open-loop pole is located at the origin due to the integral control term. A zero is also plotted, which arises due to the numerical integration performed on the OBC, and has a negligible effect on the system dynamics.

Looking at the closed-loop poles, the integral gain has moved the two open-loop poles to  $s = -4.16 \pm 3.78i$  along the root locus, which gives the system its behavior. An underdamped step response should be seen.

This concludes the analysis of the NSA controller. It is clear that this controller was designed with strong disturbance rejection in mind and to track the reference acceleration precisely. The designers must have agreed that pure integral control is sufficiently fast and that no proportional or derivative terms were needed to increase the bandwidth. The system is well damped with the fast-acting integrator.

### 5.1.3.3 Climb Rate Controller

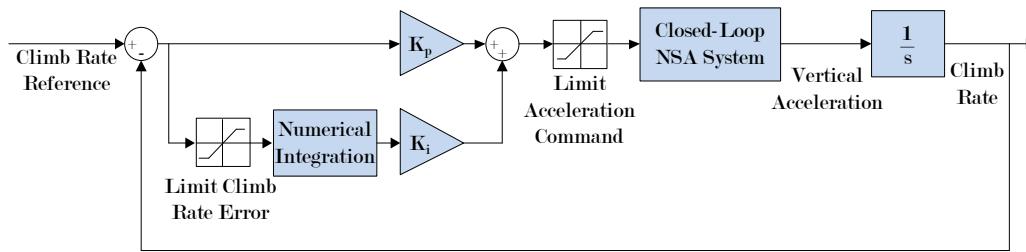
The architecture of the climb rate controller can be seen in Figure 5.47 as determined from the code on the OBC. The climb rate controller commands the NSA controller to control the climb rate of the aircraft.



**Figure 5.47 – Pre-Existing Climb Rate Controller Architecture**

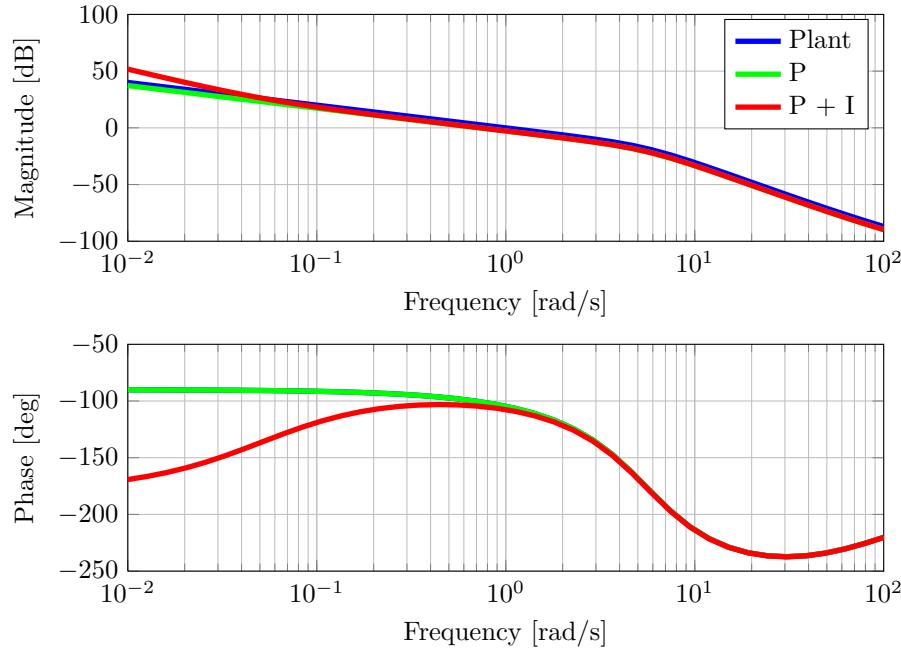
The climb rate controller uses a PI control law. At first glance it can be assumed that the proportional gain is used to achieve the required bandwidth, while the integrator and integral gain are used for disturbance rejection and possibly to compensate for bias errors in the accelerometer measurement, which will affect the NSA controller adversely.

Two limiters form part of this control strategy. The first limiter limits the climb rate error that is fed into the integrator. The original designers presumably included this limit to avoid fast wind-up of the integrator for large climb rate errors. Similarly, the acceleration command is also limited for large climb rate errors, prohibiting the NSA controller from continually limiting the admissible commanded thrust of the rotors. This limiter limits the acceleration reference command to  $\pm 1.5 \text{ m/s}^2$ . Figure 5.48 shows the block diagram that is used for the analysis.



**Figure 5.48** – Climb Rate Controller Block Diagram Used For Analysis

The open-loop Bode plot of the climb rate controller can be seen in Figure 5.49, indicating the effect that each control term has on the plot.



**Figure 5.49** – Open-Loop Bode Plot of Pre-Existing Climb Rate System

The proportional gain is used to slightly decrease the gain crossover frequency from unity gain. The integrator and integral gain do not affect the gain crossover frequency notably, and amplify low frequencies. The final result is a system with a phase margin of  $75^\circ$  at a gain crossover frequency of  $0.73 \text{ rad/s}$ . It is not known how the integral gain was determined, as it is not known over what frequency range it was intended to function. The step response is looked at next.

The step response to a climb rate reference of  $1 \text{ m/s}$  given at  $t = 1$  second can be seen in Figure 5.50. The linear model of the climb rate system was used, which does not include any limiters.

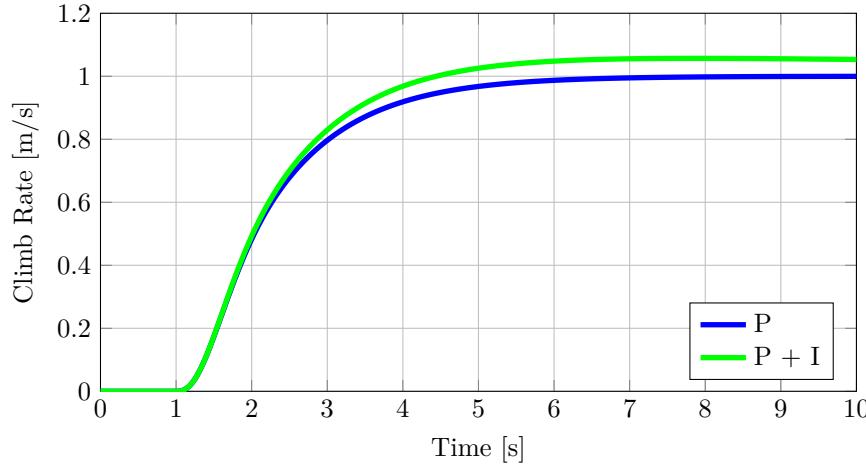


Figure 5.50 – Step Response of Pre-Existing Climb Rate System

A rise time of 2.01 seconds is seen, with a maximum overshoot of 5.7%. The 5% settling time is 9.7 seconds due to the overshoot introduced by the integrator. The integrator term seems to be very slow, but is needed for disturbance rejection.

Consider Figure 5.50 where the controller is given a climb rate step reference of  $1 \text{ m/s}$  at  $t = 1$  second and then disturbed by a constant acceleration of  $0.1 \text{ m/s}^2$  at  $t = 7$  seconds. This acceleration could arise from an external force like wind or from a measurement error by the accelerometer responsible for the NSA controller's measurement.

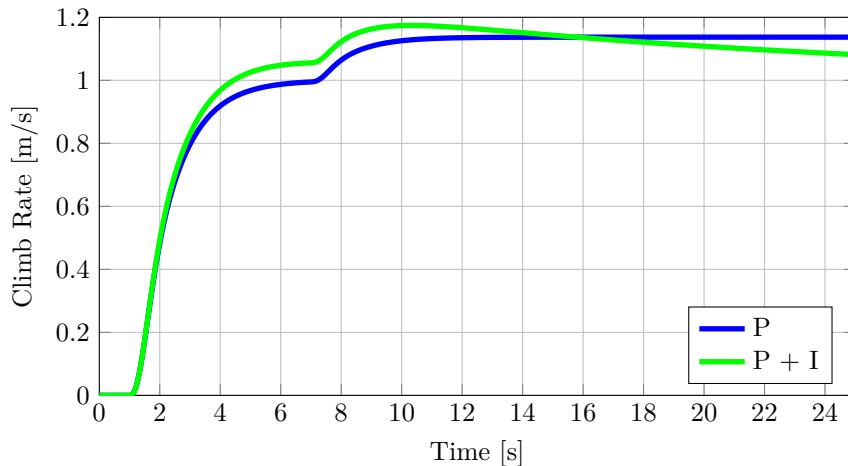
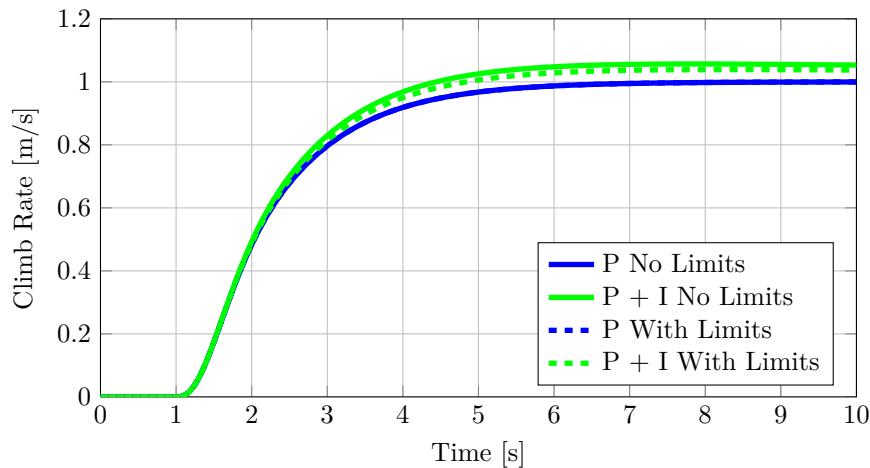


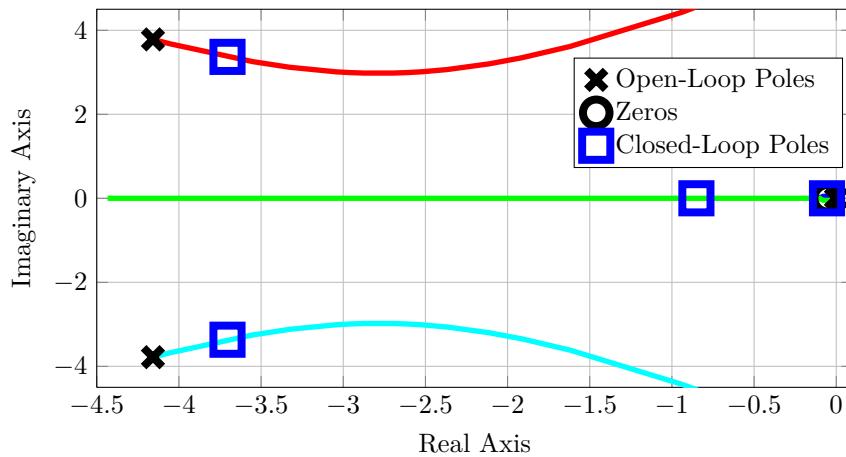
Figure 5.51 – Disturbance Response of Pre-Existing Climb Rate System

The integrator negates the constant disturbance, albeit slowly. This poses the question of whether this integrator is able to reject fast-changing external disturbances like wind and loss in thrust due to rolling and pitching. It seems that the integrator would only be effective at eliminating bias errors in the vertical specific force measurement of the vehicle or to compensate for constant drag forces during extended periods of descending or ascending. The limits are also introduced into the system and the step response is replotted in Figure 5.52.



**Figure 5.52** – Step Response of Pre-Existing Climb Rate System With Limiters

Slightly less overshoot is seen, with a maximum overshoot of 3.7%. This is due to the limit that acts on the error value being fed into the integrator, allowing limited integration of the climb rate error. The root locus of the climb rate system is finally investigated and is plotted in Figure 5.53. The open-loop poles, zeros and closed-loop poles are shown.



**Figure 5.53** – Root Locus of Pre-Existing Climb Rate System

The open-loop complex pole pair comes from the closed-loop NSA system. Two open-loop poles are also located on the origin. One of the poles arises from the natural relationship

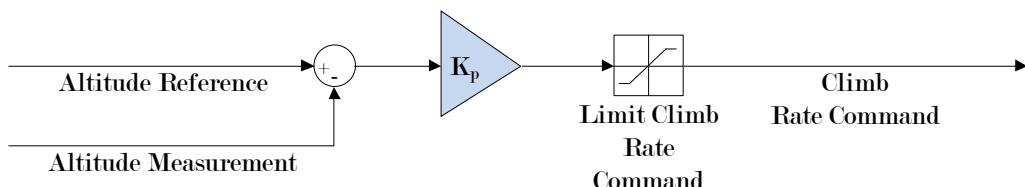
between acceleration and velocity (climb rate) and the other from the integral term of the climb rate controller. The PI controller also added a zero to the system, located at  $s = -0.05$ .

Looking at the closed-loop poles, pole/zero cancellation almost takes place at  $s = -0.05$ , leaving the dominant behavior of the system to arise from the complex pole pair now at  $s = -3.70 \pm 3.37i$  and the pole on the real axis at  $s = -0.85$ . Due to the pole near the origin being dominant, a step response which is almost like a first-order step response is expected.

This concludes the analysis of the climb rate controller. It can be deduced that the aim of this controller is to control the climb rate of the vehicle in the presence of disturbances. The integral term is, however, very slow, affirming that the integrator would only be sufficient at negating low-frequency disturbances, like a bias in the specific force measurement of the NSA controller. This controller does seem to be sufficient for the task of landing on a moving platform. The vehicle will for the most part be tracking a constant height above the platform, which this controller will be capable of doing. Only in the final stages of the automated landing procedure will the aircraft descend at a constant rate onto the platform. This controller is adequate to track the constant descend rate.

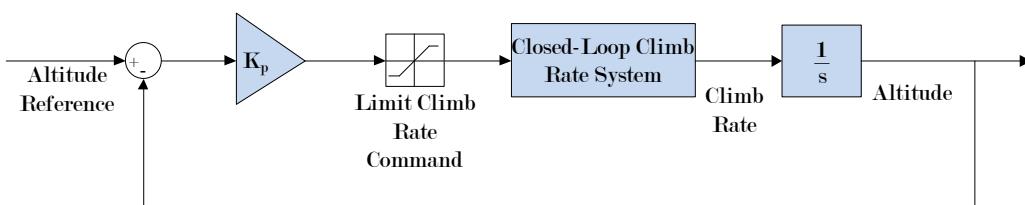
#### 5.1.3.4 Altitude Controller

The altitude controller is finally investigated and concludes the control system analysis of the pre-existing flight control system of the vehicle. The altitude controller utilizes a pure proportional control law as shown in Figure 5.54.



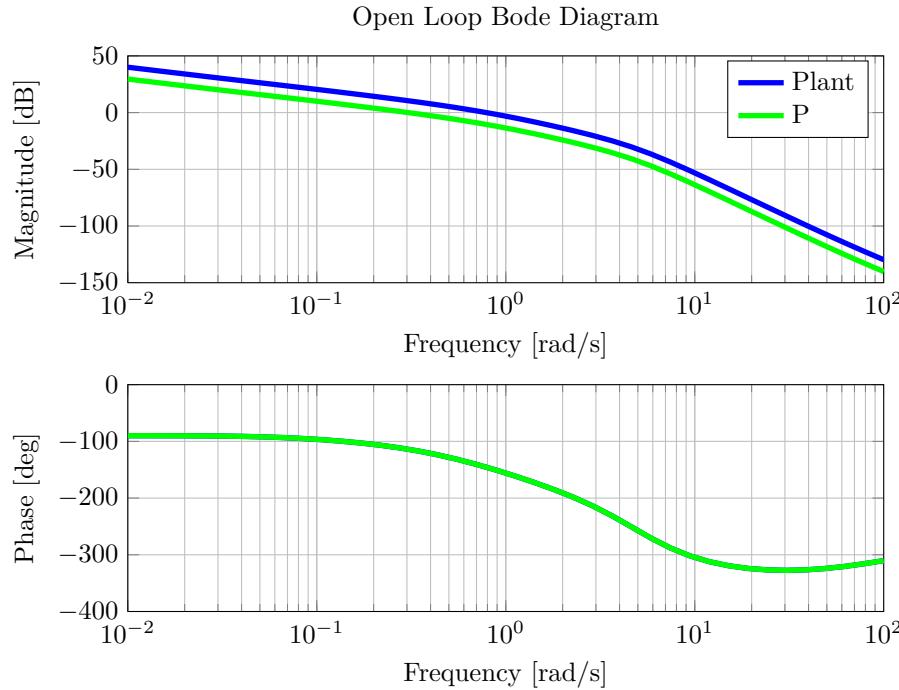
**Figure 5.54** – Pre-Existing Altitude Controller Architecture

The control strategy incorporates a limiter which limits the maximum and minimum climb rate command for large altitude errors. The controller will therefore command the vertical speed of the aircraft within practical bounds. The block diagram used for the analysis of the altitude controller can be seen in Figure 5.55.



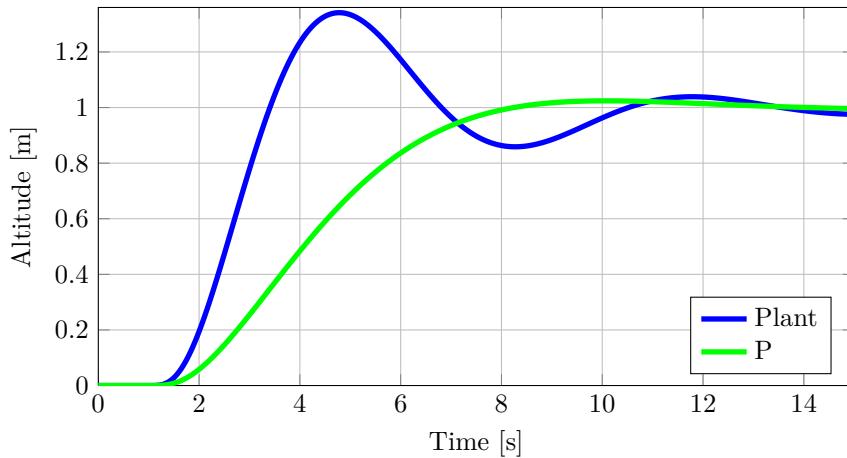
**Figure 5.55** – Altitude Controller Block Diagram Used For Analysis

An integrator is introduced into the system, which originates from the natural relationship between velocity (climb rate) and position (altitude). The open-loop Bode plot is investigated next and is plotted in Figure 5.56.



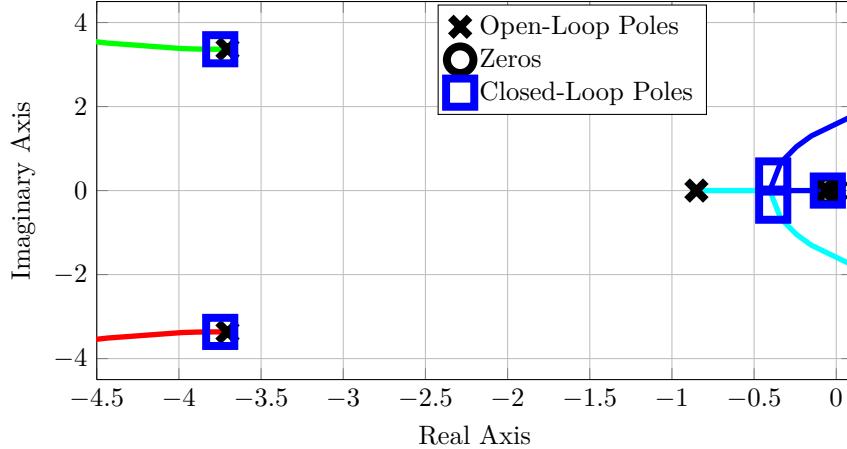
**Figure 5.56** – Open-Loop Bode Plot of Pre-Existing Altitude System

The Bode plot shows how the magnitude plot was moved down by using the proportional gain, changing the gain crossover frequency from  $0.79 \text{ rad/s}$  to  $0.30 \text{ rad/s}$ . The phase plot remains unchanged and the phase margin was increased from  $35^\circ$  to  $66^\circ$  by the decrease in gain crossover frequency. The step response is looked at next and is plotted in Figure 5.57, where a step altitude reference of  $1 \text{ m}$  is given at  $t = 1 \text{ second}$ .

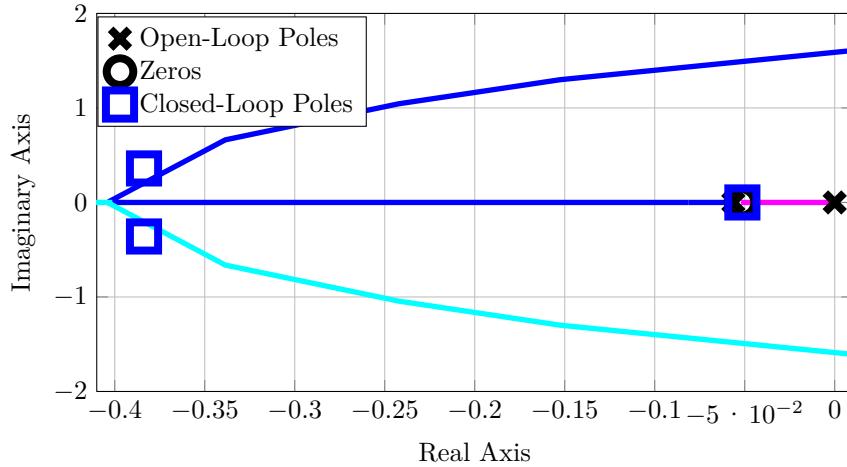


**Figure 5.57** – Step Response of Pre-Existing Altitude System

Notable oscillation and overshoot of almost 30% are seen when only the plant is used. When the proportional gain is taken into account, a maximum overshoot of 2.4% is seen. Furthermore, a rise time of 4.33 seconds and a 5% settling time of 6.21 seconds are observed. The root locus is finally investigated and is plotted in Figures 5.58 and 5.59, where Figure 5.59 is a closer look at the cluster of poles and zeros near the origin.



**Figure 5.58** – Root Locus of Pre-Existing Altitude System



**Figure 5.59** – Root Locus (Near the Origin) of Pre-Existing Altitude System

All open-loop poles and zeros come from the climb rate system, with the only exception being a new pole at the origin. Looking at the closed-loop poles, pole/zero cancellation effectively takes place at  $s = -0.05$ . The dominant behavior of the system comes from the two complex pole pairs at  $s = -0.38 \pm 0.36i$  and  $s = -3.75 \pm 3.36i$ . The expected response of the system will therefore be a combination of two underdamped second-order responses.

This concludes the analysis of the altitude controller. The speed-of-response and tracking performance of this controller are sufficient for the task of automated landing on the moving platform. The controller will need to maintain a constant height throughout the landing procedure, which it is adequately capable of doing. Slight variation in the height might

occur due to disturbances, but practically speaking, will not have an impact on the landing procedure.

## 5.2 Control System Re-Design For Autoland On a Moving Platform

In this section, the flight control system is re-designed and upgraded specifically for the task of automated landing on a moving platform. The design goals to be considered are determined first, and then the design of the control system follows. Before the design goals are determined, the yaw control system, vertical control system and the horizontal control system are briefly discussed regarding their adequacy for the project. The yaw control system is discussed first.

Consider the proposed automated landing strategy in Chapter 3. Throughout the entire automated landing procedure the aircraft will maintain a constant yaw angle, which will be the same as the yaw angle of the moving platform. The yaw control system will therefore only be required to follow a constant yaw angle reference, and to perform disturbance rejection. The control system analysis indicated that the speed-of-response of the yaw control system is sufficient to track a constant yaw angle.

For the yaw control system to perform disturbance rejection, integrators are required. The yaw control system has an integrator in the yaw rate control system, but not in the yaw angle control system. Therefore the yaw angle could have small steady-state errors. However, the translational movement of the aircraft is unaffected by the yaw angle, as seen in Section 5.1.1.3. Perturbations in the yaw angle due to disturbances or inadequate tracking will thus not affect the translational control of the aircraft. This provides further confidence that the yaw control system is adequate. A re-design of the yaw control system will therefore not be performed.

The adequacy of the vertical control system is considered next. The aircraft will be commanded to maintain a constant height of 3 m above the translating platform throughout all but the final state of the landing procedure. In the landing state, the altitude controller will be disarmed and the aircraft will be commanded to descend at a constant rate onto the moving platform. Like the yaw angle controller, the altitude controller will receive a constant reference to maintain, only requiring the controller to track the reference and to reject disturbances.

The altitude controller does not have an integrator and is not able to perform disturbance rejection. This is not considered to be a problem, as minor changes in the aircraft's altitude above the platform, originating from disturbances, will not drastically affect the landing capability of the aircraft. The altitude controller is deemed adequate for the project.

However, during the landing stage, more precise control is required from the climb rate controller, as the aircraft needs to descend at a constant rate. The climb rate controller does make use of an integrator, making the aircraft capable of following a climb rate reference in the presence of disturbances. The climb rate controller is therefore deemed acceptable for use in the project, and a re-design of the vertical control system will not be performed.

On the other hand, the horizontal control system is not adequate for automated landing on a moving platform. The controller is not capable of following a moving object with a steady-state error of zero, and the analysis showed that the horizontal velocity controller was possibly designed conservatively, which indicates potential room for improvement. A complete re-design of the horizontal control system will be performed. The design goals for the re-design are considered next.

### 5.2.1 Design Goals

Design goals need to be established before re-design of the horizontal control system is performed. The following design goals are set, followed by brief explanations for their choice and the overall approach to the re-design of the system:

1. The horizontal position of the aircraft must be controlled to track the position of the platform moving at a constant velocity, with zero steady-state error. (The position reference represented by a platform moving at a constant velocity is equivalent to a position ramp reference).
2. The horizontal velocity of the aircraft must be controlled to match the horizontal velocity of the platform.
3. The horizontal position control must compensate for the possibility that there is a bias error in the platform's velocity measurement.
4. The transient response of the aircraft horizontal position control must be fast enough to track deviations in the platform's motion from ideal constant velocity motion.
5. The transient response of the aircraft horizontal position control must be sufficiently damped so that overshoot does not cause the horizontal position of the aircraft to drift outside of the allowed bounds for landing on the platform.
6. The aircraft horizontal position and velocity control must provide disturbance rejection for constant wind and wind gusts.
7. The flight control system must be robust to uncertainty in the aircraft dynamics.
8. The flight control system must not exceed the control authority of the motors during the landing procedure.

The approach to satisfying design goals 1, 2 and 3 is discussed first. The horizontal position of the vehicle must be controlled to track the horizontal position of the platform, which is translating at a constant velocity. The constant-velocity translating position of the platform over time is equivalent to giving a position ramp reference input to the horizontal position control loop. Since it was shown in Section 5.1.1 that the horizontal position control loop is a type one system for tracking, the vehicle position will track the platform's position with a finite steady-state error.

The finite steady-state error in position is related to the slope of the position ramp reference by a proportional gain. Since the slope of the position ramp reference is the velocity of the platform, the steady-state error is effectively proportional to the velocity of the platform. The steady-state position tracking error, in terms of the actual velocity of platform  $V_{Platform}$ , was calculated by applying the final value theorem to the horizontal position control loop:

$$e_{ss} = 4.17(V_{Platform}) \quad (5.2.1)$$

However, from design goal 1, it is required that the vehicle position should track the moving platform's position with zero steady-state error. One solution is to add an integrator to the horizontal position controller to increase the type of the horizontal position control system. Yet there are disadvantages associated with this approach: The integrator will take time to determine the position compensation to add, and the integrator will decrease the stability margin of the system, thus forcing the controller to be designed for a lowered bandwidth. The integrator will also contribute to overshoot when tracking reference position commands.

Another option is to add a feed-forward velocity reference. The measured velocity of the platform will be given directly as a feed-forward velocity reference ( $V_{Feedforward_R}$ ) for the horizontal velocity controller and will be added to the velocity reference generated by the horizontal position controller, which is a proportional controller at this stage. By again applying the final value theorem, but now in terms of the velocity feed-forward reference  $V_{Feedforward_R}$  and the actual velocity of the platform  $V_{Platform}$ , Equation 5.2.2 is determined:

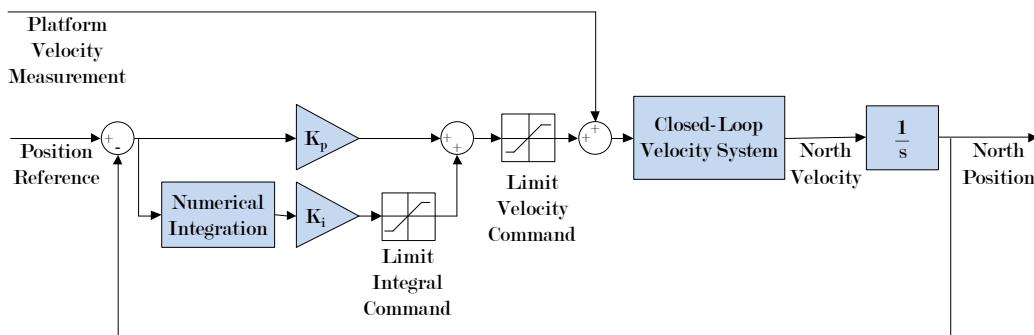
$$e_{ss} = 4.17(V_{Platform} - V_{Feedforward_R}) \quad (5.2.2)$$

If the measured platform velocity is provided as a feed-forward velocity reference to the horizontal velocity control loop, and the velocity control loop is able to control the vehicle velocity to track a constant velocity reference with zero steady-state error, then the proportional horizontal position controller will be able to achieve a steady-state position tracking error of zero.

However, adding a feed-forward velocity reference assumes that the vehicle and platform velocity measurements are ideal and contain no bias errors. In practice, both the vehicle and the platform velocity measurements may contain bias errors, which would then result in a finite steady-state tracking error in position.

An integrator must therefore be added to the horizontal position control loop to compensate for the velocity measurement errors, so that the position error can still be controlled to zero. The advantage of using both the feed-forward velocity reference for the velocity control loop and the integrator for the position control loop, is that the integrator now only has to compensate for the velocity measurement error, and does not have to determine the relative velocity of the platform by integrating the relative position error over time. This means that most of the velocity reference will be contributed by the feed-forward component, and the integrator will only contribute a small part of the velocity reference.

Adding an integrator to the horizontal position control loop and augmenting the position controller with a feed-forward velocity reference will satisfy design goals 1, 2 and 3. This control scheme will guarantee that the vehicle will track the platform with zero position error at a steady-state. The control structure that is proposed for the horizontal position controller can be seen in Figure 5.60.



**Figure 5.60** – Proposed Position Controller Architecture

Design goal 5 is considered next. When the vehicle is near the translating platform, the room for movement becomes limited. If the vehicle exhibits large overshoots while tracking the platform's position, it could collide with the platform, or miss the landing location entirely.

This means that the transients that are present while the vehicle is near the platform must be well damped, and should settle quickly. To incorporate enough damping, the system will be designed to be optimally damped, which in the frequency domain corresponds to a phase margin of  $70^\circ$ .

Satisfying design goals 4 and 6 are considered next. For the most part, the platform will be moving at a constant speed. Slight variation in the speed of the platform will, however, be present, requiring a fast response from the vehicle to stay within the required error bounds. Other factors will also cause transients to arise between the relative position of the platform and the aircraft. Transients in the vehicle's motion could be caused by variations in the reference commands that originate from state transitions in the landing state machine, and deviations of the platform's motion from ideal constant-velocity motion. Transients can furthermore be caused by disturbances from external forces like wind and turbulence.

The speed-of-response of the aircraft should therefore be increased. Improved speed-of-response will improve disturbance rejection, and ensure that transients are negated as quickly as possible. Steady-state errors will also be reduced with an increased speed-of-response, requiring less compensation from integrators.

The focus of the design should therefore be to make the transient response as fast and well-damped as possible, and to reject disturbances as quickly as possible. This involves decreasing the rise time and settling time of the response for all controllers. In the frequency domain, this translates to increasing the bandwidth of the system. Increasing the bandwidth should, however, be done in such a way that the stability of the system is not compromised and that noise on the control signal is not adversely amplified.

A limitation which must be taken into consideration is the physical limitation of the actuators of the vehicle. During the control system analysis, several limiters were shown and their purpose explained. It is decided that these limiters will be maintained and the new control system will function within these limits, as they have proven in the past that they ensure controllability of the aircraft.

With these design goals identified, re-design of the control system can be done. It is first evaluated whether there is scope for improving the overall speed-of-response of the horizontal position control. The open-loop crossover frequencies, and the resulting closed-loop bandwidths, for each of the consecutive horizontal control loops are summarized in Table 5.3. Each successive loop is a factor slower than the previous loop.

**Table 5.3 – Horizontal Control System Frequency Characteristics**

Control Loop	Crossover Frequency [rad/s]	Bandwidth (at -3 dB) [rad/s]	Factor Slower Than Previous Loop
Rate Control Loop	10.00	10.46	NA
Tilt Angle Control Loop	2.80	4.45	2.35
Velocity Control Loop	0.68	0.96	4.64
Position Control Loop	0.20	0.42	2.28

The tilt control loop is 2.35 times slower than the rate control loop, and the position loop is 2.28 times slower than the velocity control loop. In contrast to these two factors, the horizontal velocity control loop is 4.64 times slower than the tilt angle control loop. This means that the frequency separation factor between the horizontal velocity control loop and the tilt angle control loop is almost twice the frequency separation factor between all the

other control loops. This is an indication that the velocity control loop might have been designed conservatively and that its bandwidth can possibly be increased.

Increasing the bandwidth of the horizontal velocity control loop will increase the attainable bandwidth of the horizontal position control loop as well, resulting in an overall increase in the speed-of-response of the horizontal control system.

A re-design of the horizontal control system is now performed, starting with the innermost loop controller, namely the pitch/roll rate controller. The design of the controllers will be done in the frequency domain by use of Bode plots, as Bode plots easily provide information on the phase margin and gain margin of a system. Design calculations, given design requirements for the phase margin and gain crossover frequency, can become tedious when using root locus design methods.

### 5.2.2 Horizontal Roll/Pitch Rate Controller Re-Design

It can be said with certainty that high system bandwidth and aggressive disturbance rejection were the design goals of the pre-existing pitch/roll rate controllers. In Section 5.1.1.2 it is shown that the bandwidth was increased by implementing a lead compensator, and disturbance rejection was done by a lag compensator with high control authority. For the re-design of the controller, the following design requirements are to be met or to be improved upon, which are characteristics of the current pitch/roll rate controller on the vehicle:

1. A phase margin of  $70^\circ$ .
2. A gain crossover frequency of  $10 \text{ rad/s}$  or higher.
3. Add an integrating term with a zero frequency that is similar to the zero frequency of the pre-existing lag compensator ( $6.4 \text{ rad/s}$ ) and maintain the maximum contribution allowed by the integral term to the control signal.
4. The actuator commands must obey the same actuator command limits that were used for the pre-existing control system.

It is decided not to change the zero frequency of  $6.4 \text{ rad/s}$  of the pre-existing pitch/roll rate controller's integral term, which defines the frequency range over which the integrator will contribute to the control signal. This frequency is most probably determined experimentally, and is characterized by uncertainties in the dynamic model of the aircraft. These uncertainties could originate from moment imbalances caused by the rotors and unmodelled aerodynamic effects during flight.

Furthermore, it is decided to maintain the structure of the pitch/roll rate controllers. In Section 5.1.1.2 the analysis showed that this structure allows control authority to be individually assigned to the proportional and integral control terms by use of limiters. As one of the design requirements is to maintain these limiters, the pre-existing control architecture must be used, which will see the integral and proportional terms in parallel.

The re-design can be approached from a classic PID perspective or a PID approximation in the form of lead and lag compensation. It is clear from the Bode plot of the pitch rate plant, shown in Figure 5.9, that using a proportional controller alone can deliver the required gain crossover frequency, but not with the required phase margin. Similarly, the required phase margin can be achieved, but at the expense of a lower gain crossover frequency. Some form of derivative control must therefore be implemented to increase the phase margin. A lead compensator is considered instead of a pure derivative term. Noise on the process variable, which in this case is the pitch rate measurement, can cause very large control signals when differentiated. If pure derivative control is implemented, a linear low pass filter would need to be added in front of the differentiator.

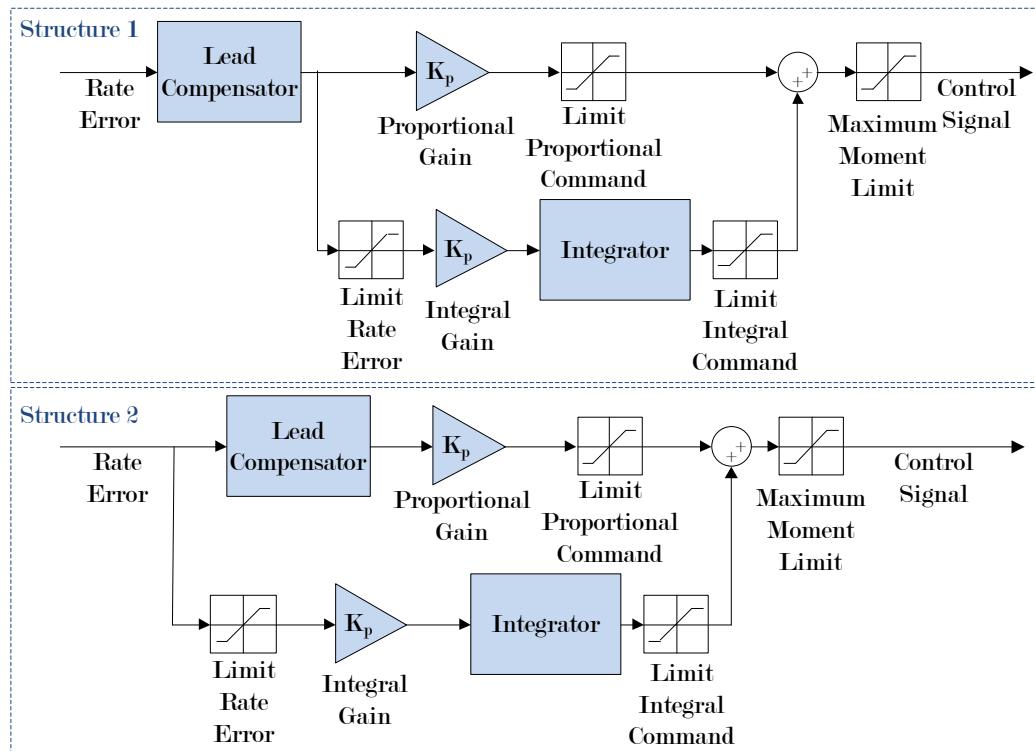
The desired gain crossover frequency is determined next. The pre-existing pitch/roll rate controller has a gain crossover frequency of  $10 \text{ rad/s}$ , and required a considerable amount of phase addition to maintain a phase margin of  $70^\circ$ . Drastically increasing the gain crossover frequency might therefore not be possible, as more phase will need to be added. It was therefore decided to design for a conservative increase in gain crossover frequency to  $12 \text{ rad/s}$ . The design process for the re-design is listed below:

### Design Steps

1. Implement a pure integrator for integral control with a zero frequency of  $6.4 \text{ rad/s}$ .
2. Adjust the proportional gain.
2. Design a lead compensator for  $70^\circ$  of phase at  $12 \text{ rad/s}$ .
3. Adjust the lead compensator's DC gain to achieve a gain crossover frequency of  $12 \text{ rad/s}$ .

The re-design differs slightly from the conventional methods of designing compensators, due to the given requirements and information available for the design.

Two architectures are proposed for the re-design of the pitch/roll rate controller, and are exhibited in Figure 5.61. One architecture proposes that the lead compensator be placed before the integral term, whereas the other proposes that the lead compensator be placed after the error signal splits between the integral and proportional terms.

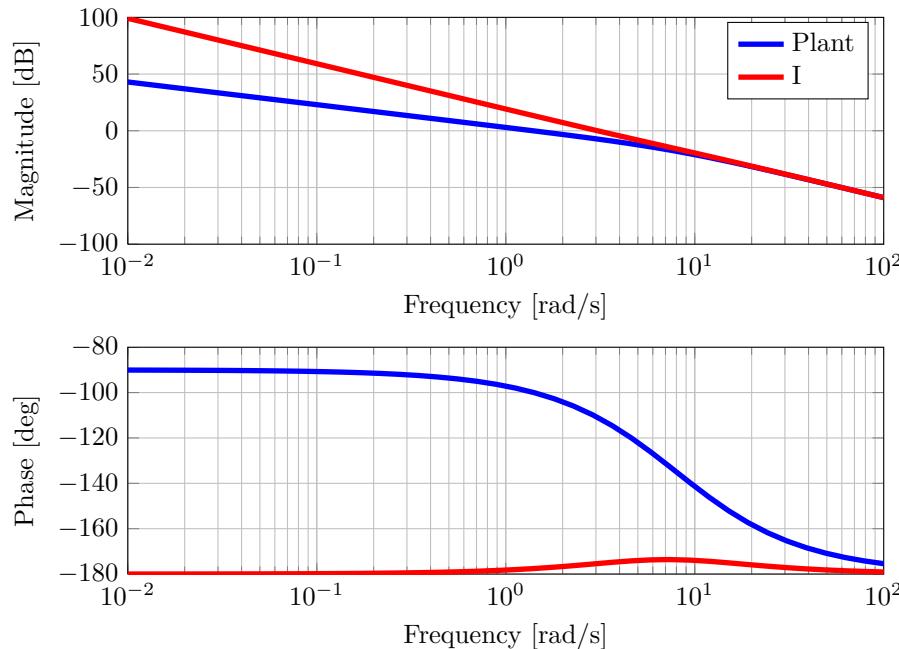


**Figure 5.61 – Proposed Pitch/Roll Rate Control Loop Architecture**

The architecture which might be more favorable is the first architecture, which is the same as the architecture implemented on the pre-existing pitch/roll rate controllers. With Structure 1 the integrator will be able to integrate the predicted error, due to the error signal being

modified by the lead compensator. This can make the integrator respond faster than it would have with the unmodified error signal. Structure 1 is therefore chosen and the design process is continued using this structure. The second structure will be investigated later.

The integrator is implemented first, and from the design requirements it is known that the integrator needs to have a zero frequency of  $6.4 \text{ rad/s}$ , which will be the integral gain. The integrator and integral gain are added in their own branch of the control signal, so that limiters can be used to limit the contribution from the integral control term. The current state of the Bode plot can be seen in Figure 5.62.



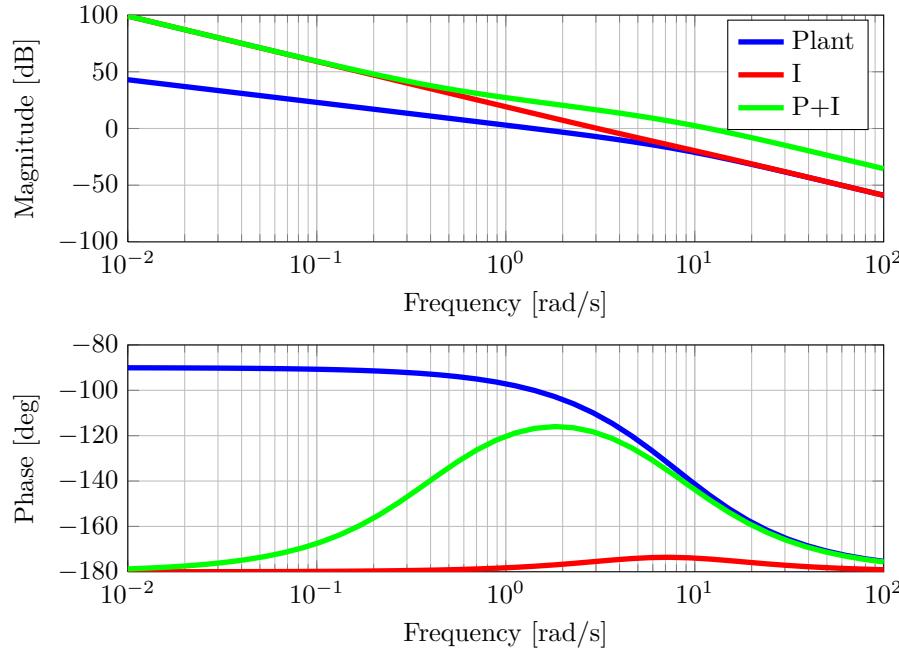
**Figure 5.62 – Open-Loop Bode Plot of Pitch/Roll Rate System With Integral Control Law**

The integrator and integral gain degraded the phase margin considerably. The next step is to adjust the proportional gain. The design process deviates at this stage from the conventional way of a lead compensator-based controller design. Normally, at this stage, the designer would investigate the phase margin and determine how much phase the lead compensator must contribute and at what frequency. After the lead compensator has been designed, the proportional gain is adjusted to achieve the required gain crossover frequency.

However, with this control architecture the phase plot will change as the proportional gain is changed. It therefore makes sense to change the proportional gain at this stage, before it is determined how much phase is to be added by the lead compensator. The proportional gain is increased by a factor of 15 from unity, yielding a gain crossover frequency of approximately  $12 \text{ rad/s}$ . The resultant Bode plot is plotted in Figure 5.63.

The phase graph has changed, and it can now be approximately determined how much phase needs to be added to the system. After the gain adjustment, the phase margin is  $31.6^\circ$  at a frequency of  $12 \text{ rad/s}$ . Therefore the lead compensator will have to contribute  $38.4^\circ$  of phase to the system at a frequency of  $12 \text{ rad/s}$ . Two lead compensator designs are proposed and compared.

As the magnitude plot of the lead compensator might affect the gain crossover frequency drastically, a choice is made for the first lead compensator to add more phase at a higher

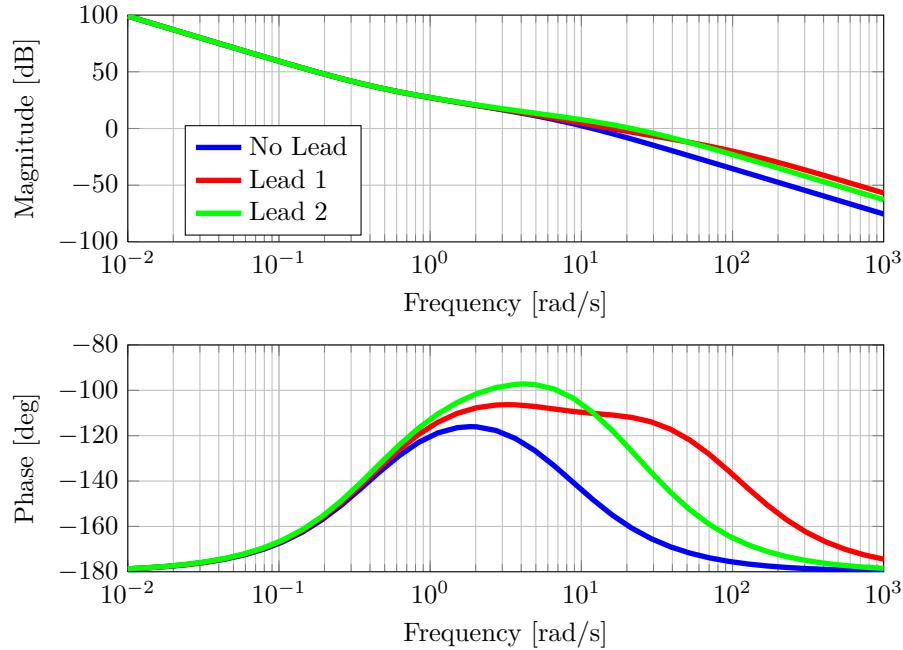


**Figure 5.63** – Open-Loop Bode Plot of Pitch/Roll Rate System With PI Control Laws

frequency. The peak of the magnitude plot of the lead compensator will then be located to the right of the desired gain crossover frequency, and should then not affect the gain crossover frequency drastically. This method also has another advantage: the lead compensator can be designed so that the phase margin stays near constant before and after the gain crossover frequency. Minor uncertainties in the plant model, which will change the gain crossover frequency, will not affect the transient response of the system as the phase margin remains constant. The lead compensator was designed, and adds  $52^\circ$  of phase at a frequency of  $35 \text{ rad/s}$ . The lead compensator is added in series with the parallel proportional and integral terms, as Structure 1 illustrates in Figure 5.61.

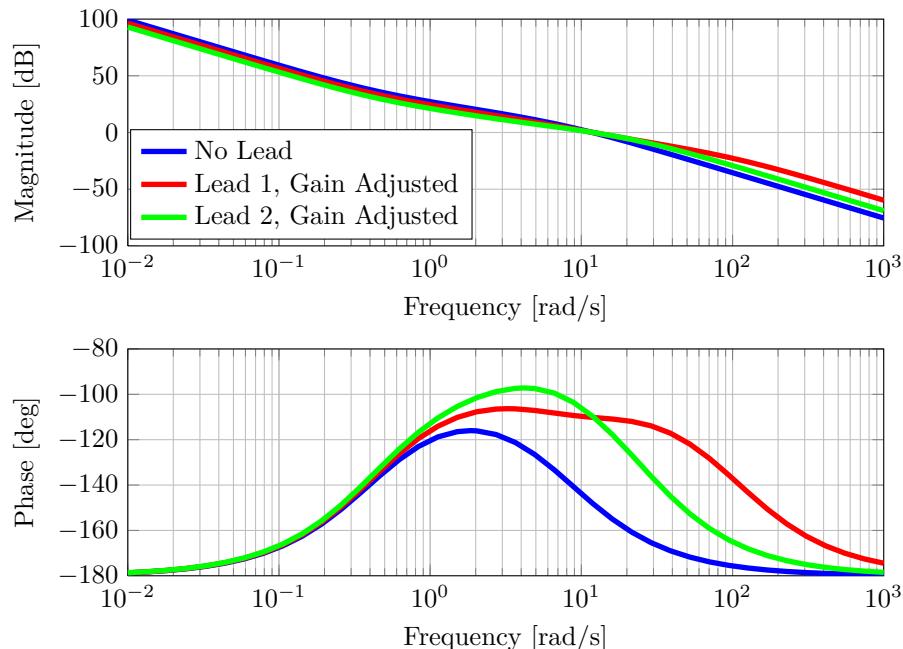
The second lead compensator design adds  $38.4^\circ$  of phase, which is exactly the phase that is required, at the desired gain crossover frequency of  $12 \text{ rad/s}$ . With this method it is expected that the lead compensator's magnitude curve will affect the gain crossover frequency. The Bode plot of the system without a lead compensator and the system with the two re-designed compensators are plotted in Figure 5.64. Both re-designs yield a  $70^\circ$  phase margin at a frequency of  $12 \text{ rad/s}$ .

The first lead compensator design, which added more phase than needed at a higher frequency, does indeed affect the gain crossover frequency less than the second lead compensator, and gives the phase graph almost-constant behavior before and after the desired gain crossover frequency. High frequencies are, however, amplified more than in the second lead compensator design. Therefore, if high-frequency process noise is present, this noise could contribute to prominent control signals. The second compensator design increases the gain crossover frequency from  $12 \text{ rad/s}$  to  $21.5 \text{ rad/s}$ , but does not amplify high-frequency components as much as the first lead compensator.



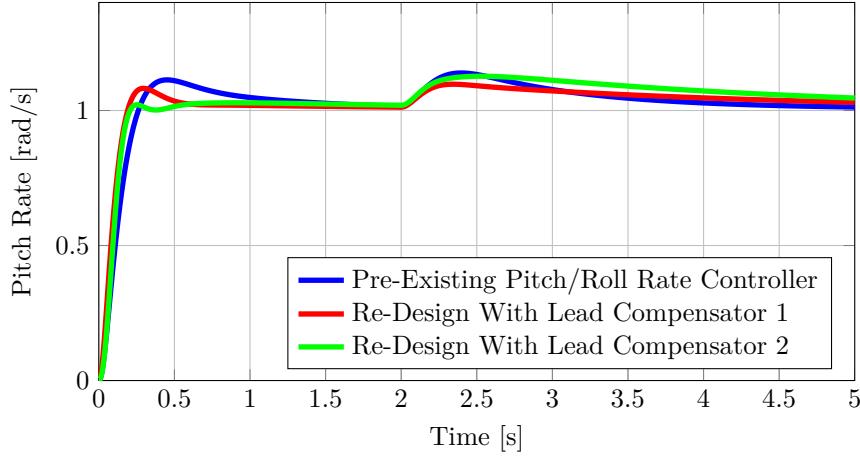
**Figure 5.64** – Open-Loop Bode Plot of Pitch/Roll Rate System With Lead Compensators

The final step in the design is to adjust the DC gain of the lead compensators, so that the gain crossover frequencies of both designs return to the desired gain crossover frequency. The gain is adjusted and the final result is the Bode plot in Figure 5.65.



**Figure 5.65** – Open-Loop Bode Plot of Re-Designed Pitch/Roll Rate System

The step response of the pre-existing system and the re-designed pitch/roll rate controller can be seen in Figure 5.66. No limiters were included when simulating the step response. A unit step reference is given at time  $t = 1$  second and a unit step disturbance is given at time  $t = 2$  seconds.



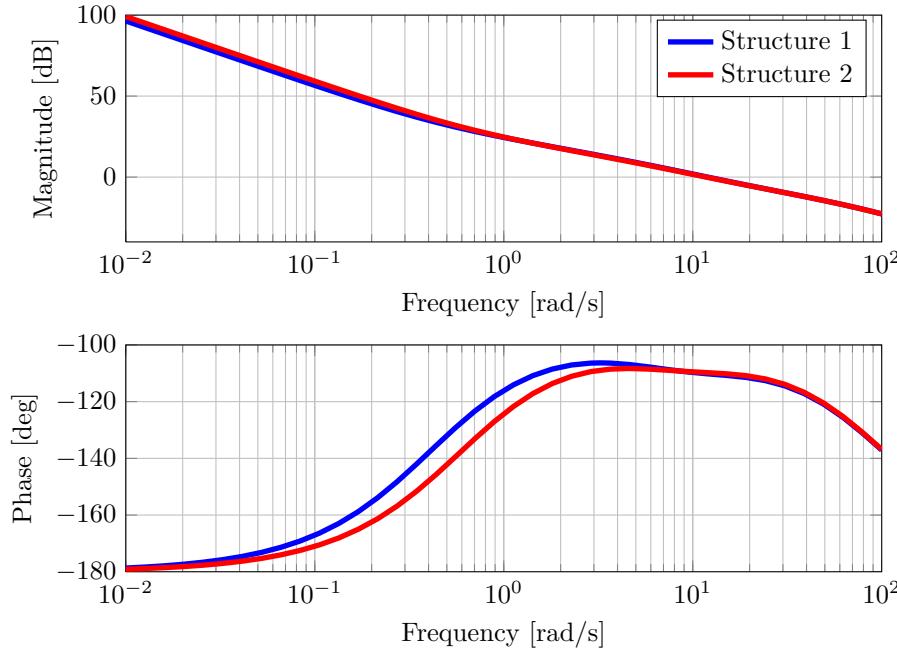
**Figure 5.66** – Step Response and Disturbance Response of Re-Designed Pitch/Roll Rate Systems

The settling time of both re-designed controllers are a slight improvement on the pre-existing control system. The re-designed controller with the first compensator outperforms the other two controllers with regard to settling time and disturbance rejection. Disturbance rejection transients are negated slower by the re-design using the second lead compensator.

Actuator saturation was also investigated. Saturation of the actuators was checked by determining the minimum step size for the pitch rate reference command that would cause the pre-existing pitch rate control system to saturate the actuators during the transient response. The same size pitch/roll rate step command was then given to the two re-designed pitch/roll rate control systems. The two re-designed pitch/roll rate control systems responded by saturating the actuators marginally longer than the pre-existing pitch/roll rate control system. The actuator saturation of the re-designed pitch rate controllers were therefore deemed to be acceptable.

The two new control structures presented in Figure 5.61 are lastly compared. The lead compensator from the first re-design was put in parallel with the integrator term, as seen in Structure 2 in Figure 5.61. The Bode plots of the two different structures are plotted in Figure 5.67.

The gain crossover frequency and phase margin remain unchanged. However, with Structure 2 the phase margin has worsened for lower frequencies, but the low-frequency gain is increased. The comparison shows that there is no notable benefit in using either one of the structures.



**Figure 5.67** – Open-Loop Bode Plot of Proposed Control Structures for Pitch/Roll Rate Controller Re-Design

This concludes the re-design of the pitch/roll rate control system. The re-designed controllers will not be implemented, as the marginal improvement over the current control system does not justify reimplementation. Attempting to achieve a higher gain crossover frequency than  $12 \text{ rad/s}$  will require more phase addition, which will inevitably amplify high-frequency noise, which can become a problem in practice.

The pre-existing pitch/roll rate controller's speed-of-response, damping, disturbance rejection and steady-state tracking are deemed acceptable for the requirements of the project.

### 5.2.3 Tilt Angle Controller

The tilt angle controller is re-designed next. The following design parameters are set:

1. A phase margin of  $70^\circ$ .
2. A gain crossover frequency of  $2.8 \text{ rad/s}$  or higher.
3. The pitch/roll rate reference commanded by the tilt angle controller must obey the same pitch/roll rate reference limits that were used for the pre-existing tilt angle controller.

It is firstly called into question why no integral control or derivative control was implemented in the pre-existing tilt angle controller. The possible reasons for this are discussed.

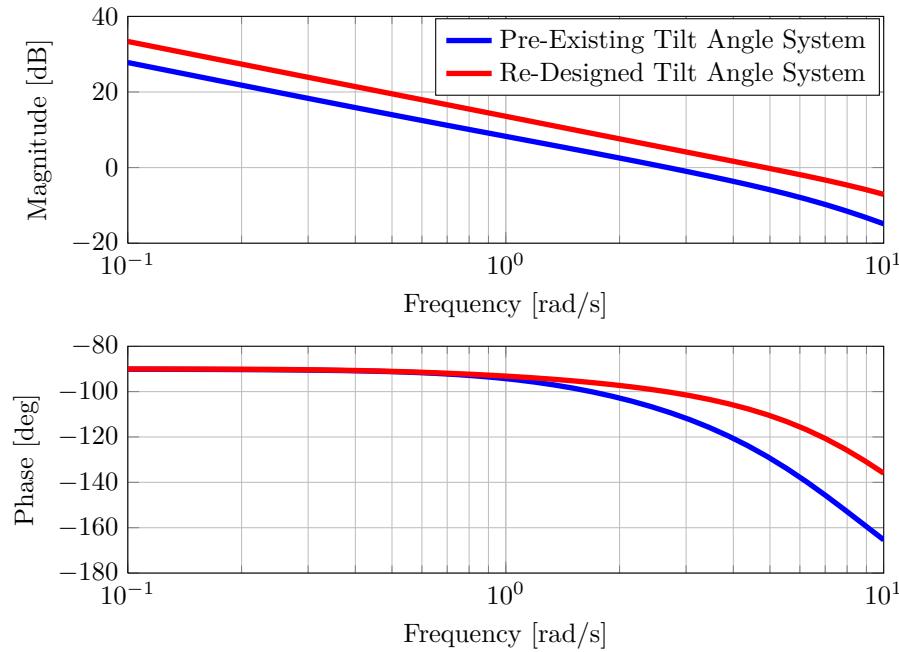
The system is a type one system for tracking, implying that the steady-state error will be zero for a reference step input. To be able to reject disturbances and to compensate for unmodelled dynamics, the system must at least be a type two system, which requires an integral control term. However, increasing the system type might not be necessary for disturbance rejection, as the velocity control loop makes use of integral control.

The integrator in the velocity loop will continue to increase or decrease the tilt angle reference command until the velocity error is reduced to zero. There is therefore no need or desire to control the tilt angle to track the tilt angle reference with zero steady-state error. The compromise associated with this approach is that an integrator in the velocity loop will take longer to compensate for an error in the tilt angle tracking than an integrator in the tilt angle control loop itself would.

Another reason for not adding an integrator is that any uncertainties in the plant model are compensated for by the lag compensator in the pitch/roll rate loop. The pitch/roll rate control loop has very strong compensation for plant uncertainties, as was seen in Section 5.1.1.2. The only dynamics which the closed pitch/roll rate loop is augmented with are the integrator relating pitch rate and pitch angle. The added dynamics are dynamics that are well known and easily modelled.

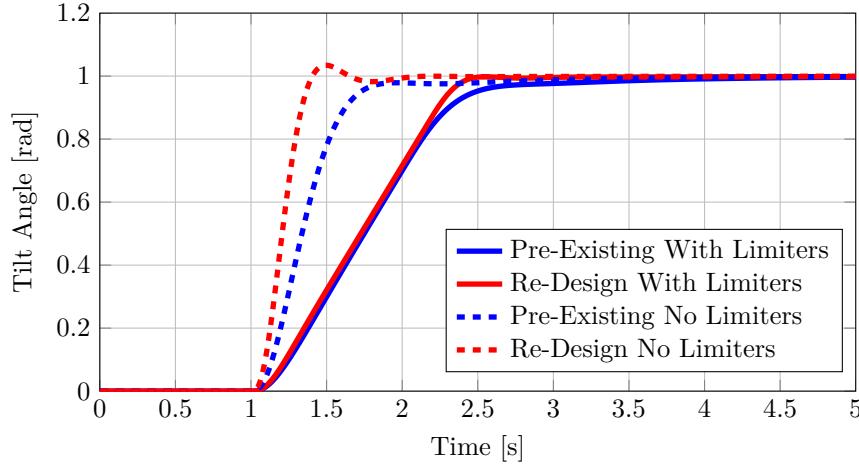
Adding an integrator will decrease the phase margin. To improve the phase margin, derivative control will be needed. The complication with implementing derivative control in this loop is that derivative control is being performed in the horizontal velocity and the pitch/roll rate loops. This means that there will be three successive loops, each being sensitive to process noise and each contributing more process noise.

From the above reasoning, it is clear that the best choice will be to use a pure proportional controller. A re-design, based on the pre-existing pitch/roll rate control loop, will not be performed, as the achievable gain crossover frequency with a phase margin of  $70^\circ$  is  $2.8 \text{ rad/s}$ . A design of the tilt angle controller can, however, be performed on the re-designed pitch/roll rate controller, as this controller has a gain crossover frequency of  $12 \text{ rad/s}$  versus the pre-existing pitch/roll rate controller's  $10 \text{ rad/s}$ . This will lead to the tilt angle controller being able to achieve a higher gain crossover frequency with the required phase margin. Consider Figure 5.68.



**Figure 5.68 – Bode Plot of Re-Designed Tilt Angle Controller**

The re-designed pitch/roll rate controller allowed for the proportional gain of the tilt angle controller to be increased from 2.45 to 4.65. The gain crossover frequency is increased from  $2.8 \text{ rad/s}$  to  $4.85 \text{ rad/s}$ , while maintaining a  $70^\circ$  phase margin. The step responses of the pre-existing tilt angle controller and the re-designed tilt angle system are plotted in Figure 5.69.



**Figure 5.69** – Step Response of Pre-Existing and Re-Designed Tilt Angle Systems

When looking at the linear step responses, when no limiters are used, a maximum overshoot of 3.5% is seen in the re-designed system's response, in contrast to undershoot being observed by the pre-existing system. The rise time has decreased from 0.48 seconds to 0.24 seconds, and the time constant has changed from 0.41 seconds to 0.24 seconds. The 5% settling time has also decreased from 0.72 seconds to 0.36 seconds.

However, when the limiters are taken into account, the increase in the speed-of-response is marginal. The two responses have the same time constant and almost the same rise time and 5% settling time.

The activity of the pitch/roll rate reference command limiter of the tilt angle controller was lastly investigated. The re-designed system limited the pitch/roll rate reference commands for approximately 10% longer than the pre-existing system, given the same step reference tilt angle command. The increase in the aggressiveness of the tilt angle controller is deemed acceptable.

The re-design of the tilt angle controller is completed. Like the pitch/roll rate controller re-design, the increase in speed-of-response of the re-designed tilt angle controller does not justify reimplementation, especially when the limiters are taken into account. The pre-existing tilt angle controller has sufficient speed-of-response and damping for use in the project.

#### 5.2.4 Horizontal Velocity Controller

The next controller in the horizontal control system to be re-designed is the horizontal velocity controller. Table 5.3 illustrates that the horizontal velocity control loop is 4.64 times slower than the pre-existing tilt angle control loop. In contrast, the frequency separation factor is almost halve of that (2.35) between the pitch/roll rate control loop and the tilt angle control loop. Similarly, the frequency separation factor between the horizontal position control loop and the horizontal velocity control loop is 2.28.

Furthermore, in Section 5.1.1.4, the analysis of the pre-existing horizontal velocity controller indicated that the derivative term of the controller contributes a negligible amount of phase. The large frequency separation factor and the negligible phase contributed by the derivative control term are indicators that the horizontal velocity control loop is conservatively designed. In theory it should therefore be possible to increase the bandwidth of the controller for the project. Increasing the horizontal velocity controller's bandwidth will naturally lead to an opportunity to increase the horizontal position control loop's bandwidth.

To increase the bandwidth of the system, a design goal is set of at least halving the frequency separation factor of 4.64, but still maintaining a phase margin of  $70^\circ$ . The frequency separation factor between each consecutive control loop in the horizontal control system will then be similar. It is not known what the exact gain crossover frequency is that should be designed for to halve this factor. However, halving this factor is analogous to doubling the bandwidth of the horizontal velocity control loop.

In a first attempt, an assumption is made that the crossover-frequency-to-closed-loop-bandwidth relationship is linear. This is not true, but it provides a good starting point for determining the gain crossover frequency that should be designed for. Therefore, if the bandwidth of the horizontal position control loop must be doubled, the gain crossover frequency must be doubled. Doubling the pre-existing gain crossover frequency sets a design goal for achieving a new gain crossover frequency of  $1.38 \text{ rad/s}$ .

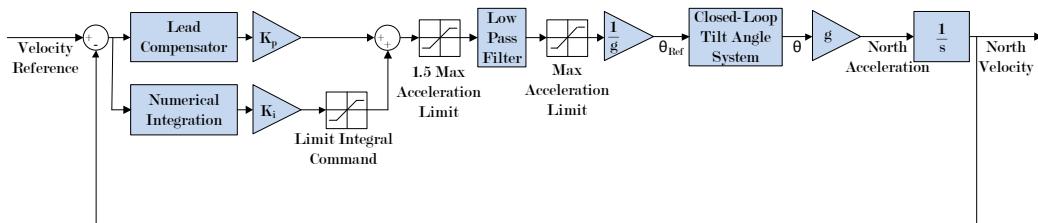
However, rather than designing for a specific gain crossover frequency of  $1.38 \text{ rad/s}$ , it is decided to design for a gain crossover frequency between  $1.2 \text{ rad/s}$  and  $1.4 \text{ rad/s}$ . Three designs will be performed in this gain crossover frequency range, each varying in aggressiveness. The following design parameters are set:

1. A phase margin of  $70^\circ$ .
2. A gain crossover frequency between  $1.2$  and  $1.4 \text{ rad/s}$ .
3. Maintain the zero frequency and limiters of the pre-existing horizontal velocity controller's integral term.
4. The acceleration reference commanded by the horizontal velocity controller must obey the same acceleration reference limits that were used for the pre-existing horizontal velocity controller.

The design process for each of the three re-designs is the same:

1. Add an integrator with the same integral gain and limits as the pre-existing design.
2. Design a lead compensator to provide a phase margin of  $70^\circ$ .
3. Perform a final proportional gain adjustment if needed.

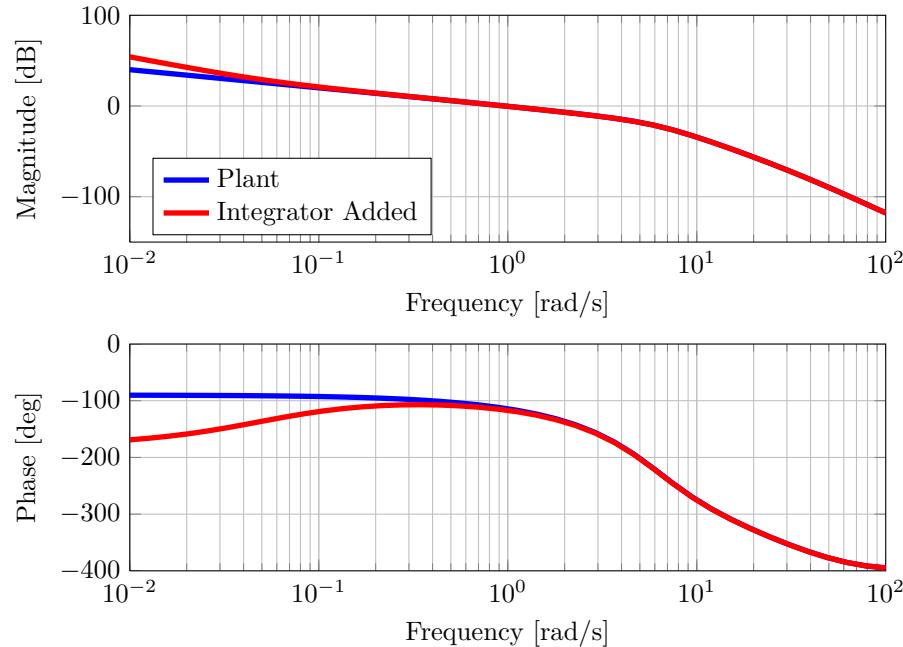
The proposed architecture of the controller is illustrated in Figure 5.70.



**Figure 5.70 – Proposed Horizontal Velocity System Architecture**

The re-design that was finally implemented on the vehicle was the third design. The third design will be documented here and the reason for choosing this design will be made clear later in this section. The other two designs are not presented here, as the only difference between the three re-designs is the different gains associated with different crossover frequencies, and different lead compensator designs.

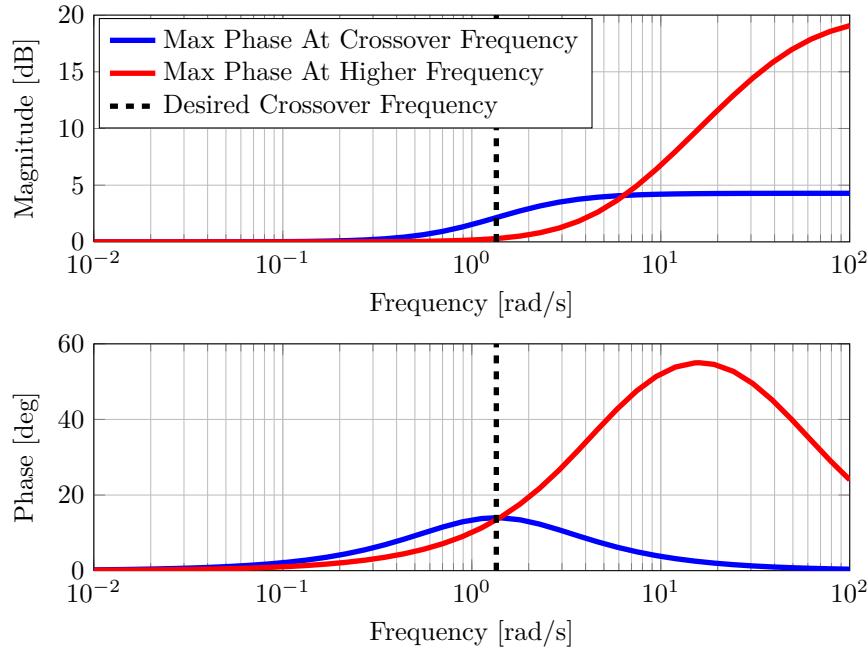
In the first design step the integrator is added. The integral gain and the limit of the integrator was kept as it was in the pre-existing horizontal velocity control system. The Bode plot of the plant and the plant and integral term are plotted in Figure 5.71. The integrator is added in its own branch of the error signal, as shown in Figure 5.70.



**Figure 5.71** – Open-Loop Bode Plot of Horizontal Velocity Plant And Integral Term

The phase margin has not been degraded for frequencies higher than  $1 \text{ rad/s}$ . This is due to a small integral gain, giving the integrator a low zero frequency. The next step was the design of the lead compensator. The phase margin is investigated and is found to be  $56^\circ$  at a frequency of  $1.38 \text{ rad/s}$ , which is within the desired gain crossover frequency range of  $1.2 \text{ rad/s}$  to  $1.4 \text{ rad/s}$ .

The lead compensator should add  $14^\circ$  of phase to achieve a phase margin of  $70^\circ$ . It is well known that adding a lead compensator to a system will slightly increase the gain crossover frequency, if the compensator's maximum phase is at the desired gain crossover frequency. Consider Figure 5.72, where the Bode plots of two lead compensators are plotted. The blue curve represents a lead compensator that adds  $14^\circ$  of phase at the desired gain crossover frequency, indicated on the graph as a vertical black dotted line. The red curve is a lead compensator that adds  $55^\circ$  of phase at a frequency of  $16 \text{ rad/s}$ .

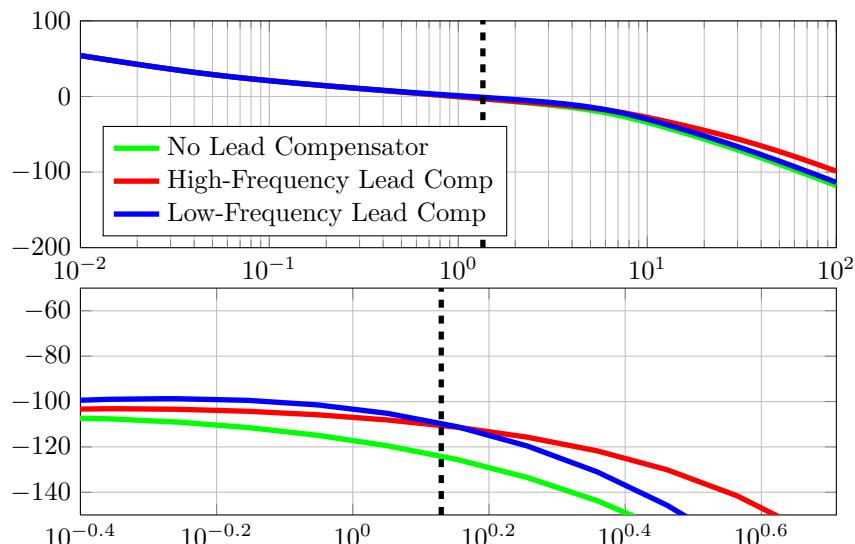


**Figure 5.72** – Open-Loop Bode Plot Comparison of Low- and High-Frequency Lead Compensators

On the magnitude plot, at the desired gain crossover frequency, the blue curve's magnitude plot contributes 2.5 dB, whereas the red curve contributes 0.2 dB. The magnitude addition from the blue curve will in turn increase the gain crossover frequency, requiring a gain adjustment.

Another advantage of the red curve is that more phase is added at higher frequencies. This will make the phase margin of the horizontal velocity system more robust to changes in the gain crossover frequency. The change in gain crossover frequency can originate from unmodelled dynamics in the plant.

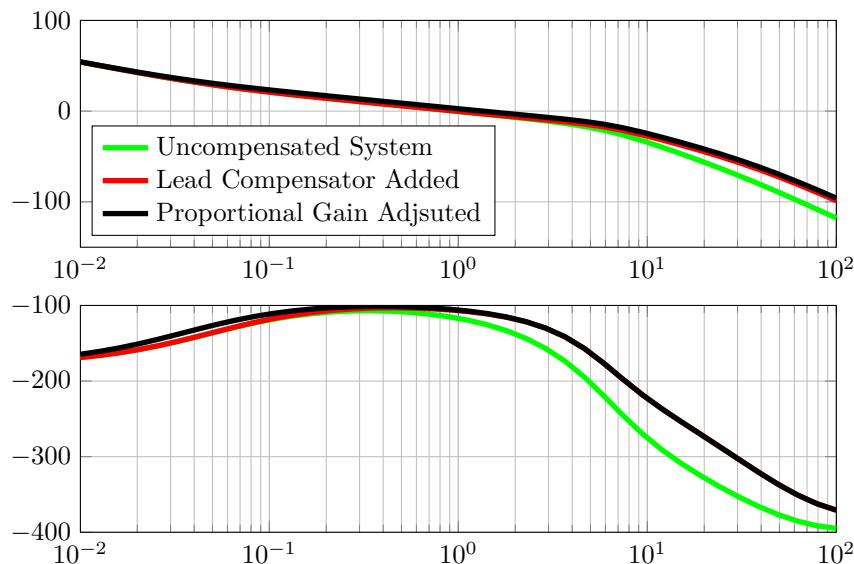
Consider Figure 5.73. Note that the frequency range on the x-axis is not the same for the magnitude plot and the phase plot. The desired gain crossover frequency is again indicated by the vertical black dotted line. The uncompensated horizontal velocity system is represented by the green line and the red curve represents the horizontal velocity compensated by the red compensator in Figure 5.72. Similarly the blue curve represents the compensated horizontal velocity system using the blue compensator in Figure 5.72.



**Figure 5.73** – Open-Loop Bode Plot of Horizontal Velocity System with Low- and High-Frequency Lead Compensators

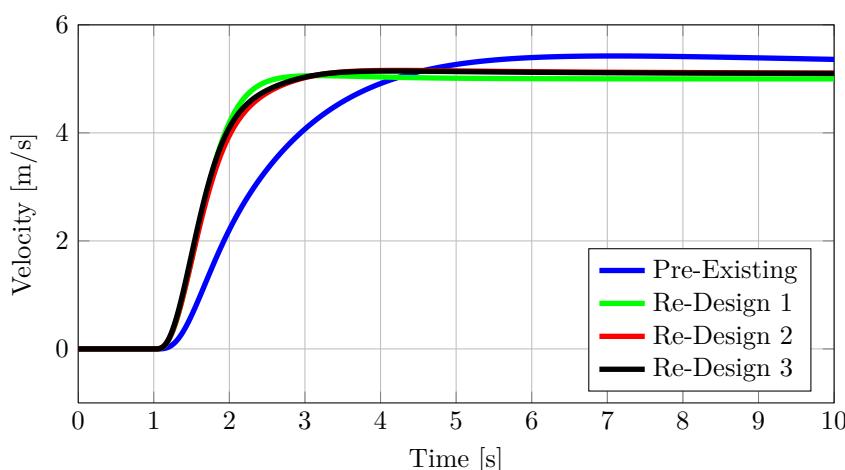
On the phase plot it can be seen that the red curve's phase margin will be less sensitive to variations in the gain crossover frequency, whereas the blue curve's phase margin starts to decline more sharply for frequencies higher than the gain crossover frequency. The compromise with this approach is apparent in the magnitude plot, where high frequencies are amplified by the red curve. Therefore, when using this method to implement a lead compensator, the system will be more sensitive to high-frequency process noise.

The lead compensator that added  $55^\circ$  of phase at a frequency of  $16 \text{ rad/s}$  was chosen for implementation, due to its added robustness to the phase margin from changes in the gain crossover frequency. After the implementation of the lead compensator, a final gain adjustment was made, which did not affect the phase plot. The Bode plot on the final state of the system is shown in Figure 5.74. The final result is a gain crossover frequency of  $1.34 \text{ rad/s}$  and a phase margin of  $71^\circ$ .



**Figure 5.74 – Open-Loop Bode Plot Of Re-Designed Horizontal Velocity System**

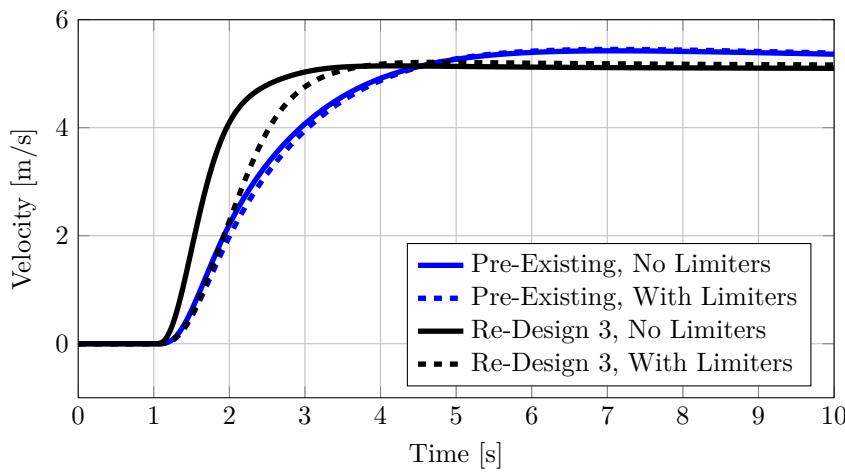
The step response of the re-designed system was simulated with the linear model, which excludes all the limiters. The step response of the pre-existing and the three re-designed controllers can be seen in Figure 5.75.



**Figure 5.75 – Step Response of Pre-Existing and Re-Designed Horizontal Velocity Systems**

The three re-designed controllers all have less overshoot and settle faster than the pre-existing system. Design 3, which is the design chosen for implementation, is compared to the pre-existing step response. The overshoot of the pre-existing horizontal velocity system is 8.5%, compared to 2.8% of the re-designed system. The 5% settling time has decreased from 9.71 seconds to 1.45 seconds. The rise time has decreased from 1.90 seconds to 0.93 seconds. Lastly, the time constant of the horizontal velocity system has decreased from 1.42 seconds to 0.75 seconds.

If the bandwidth of the system is increased, it will naturally require more aggressive actuation. This will lead to the acceleration limit of the vehicle being reached more often. The additional time in which this limit is reached is investigated. The limiters were introduced into the system and the step response to a reference step command of 5 m/s is plotted in Figure 5.76. A reference step command of 5 m/s is given so that the activity of the limiters can become apparent. The linear system response and the system response with the limiters are shown for the pre-existing system and re-designed system that is to be implemented.



**Figure 5.76 – Step Response of Pre-Existing and Final Re-Designed Horizontal Velocity Systems, With Limiters**

Both systems are limited by the acceleration limit of  $4 \text{ m/s}^2$  immediately after the step command is issued. The limiter stops limiting the pre-existing system approximately a second later, whereas the re-designed system is limited for another 0.5 seconds. The limiter that regulates the maximum pitch and roll moments was also investigated, to ensure that the actuators are not saturating. A negligible difference was seen in actuator acceleration between the two systems. The re-designed controller is therefore deemed acceptable.

Table 5.4 summarizes the controller gains and the resulting frequency domain characteristics for each of the three re-designed velocity controllers.

**Table 5.4 – Pre-Existing and Re-Designed Horizontal Velocity System Parameters**

Parameter	Pre-Existing	Design 1	Design 2	Design 3
Proportional Gain	0.69	1.5	1.34	1.4
Lead Compensator Phase [°]	NA	55.0	56.5	55.0
Lead Compensator Frequency [rad/s]	NA	22	18	16
Integral Gain	0.05	0.05	0.05	0.05
Phase Margin [°]	70	67	70	71
Crossover Frequency [rad/s]	0.68	1.38	1.28	1.34
Bandwidth (-3 dB) [rad/s]	0.96	2.38	2.02	2.16

The summary shows that Design 1 has the highest bandwidth, being the most aggressive design; Design 2 has the lowest bandwidth; and Design 3 is in the middle. This was the reason for choosing Design 3 to be implemented, as it is a compromise between aggressive actuation and speed-of-response.

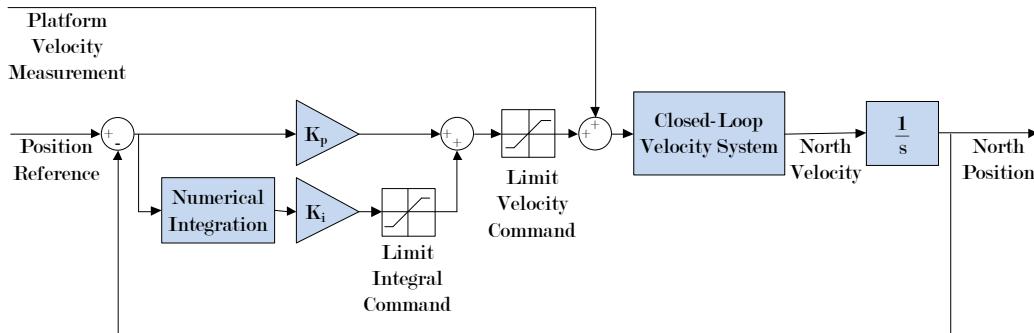
The resulting frequency separation factor is lastly investigated to ensure that the design goal has been achieved. The tilt angle control loop bandwidth is  $4.45 \text{ rad/s}$  and the re-designed horizontal velocity control loop's bandwidth is  $2.16 \text{ rad/s}$ , making the frequency separation factor between the two control loops 2.05. This separation factor is less than half of the frequency separation factor before the system was re-designed. The design goal has therefore been achieved. From the achieved frequency separation factor, it can also be noted that the assumption that was made in the beginning of this section, namely that the crossover-frequency-to-closed-loop-bandwidth relationship is linear, is not true. This assumption did, however, provide an adequate estimate for which gain crossover frequency must be designed.

This concludes the re-design of the horizontal velocity controller. Further analysis of the horizontal velocity control loop will be done in the non-linear SIL simulation, in Section 5.3.

### 5.2.5 Horizontal Position Controller

Finally, a re-design of the horizontal position control loop is performed. The design of the position controller will build upon the three re-designs that were performed for the horizontal velocity controllers.

As mentioned in the beginning of this section, an integrator will be added in this control loop, to compensate for errors in the velocity measurements of the vehicle and the platform. The control architecture for the position controller will therefore be changed from a proportional controller to a PI controller. Furthermore, the controller will be augmented with a feed-forward velocity reference, which will be the platform's velocity measurement. The new control structure is shown in Figure 5.77.



**Figure 5.77 – Proposed Horizontal Position Controller Architecture**

The design goal is to achieve a bandwidth that is as high as possible, while maintaining a phase margin of  $70^\circ$ . The design process will start by augmenting the controller with an integrator. After the integrator has been added, the proportional gain will be adjusted to achieve a phase margin of  $70^\circ$ .

An attempt was made to characterize the combined error in the velocity measurement of the GPS (Global Positioning System) sensors of the aircraft and the platform. The experiments

performed to characterize the error will be presented in Chapter 6. The magnitude of the velocity measurement error will be used to choose the limits of the north and east position integrators, so that the integrators are not allowed to integrate freely. To prevent integrator wind-up, the integrators are limited so that they can only contribute enough to the control signal to compensate for the maximum expected velocity measurement error, and no more. This limit was determined to be  $\pm 0.142 \text{ m/s}$ .

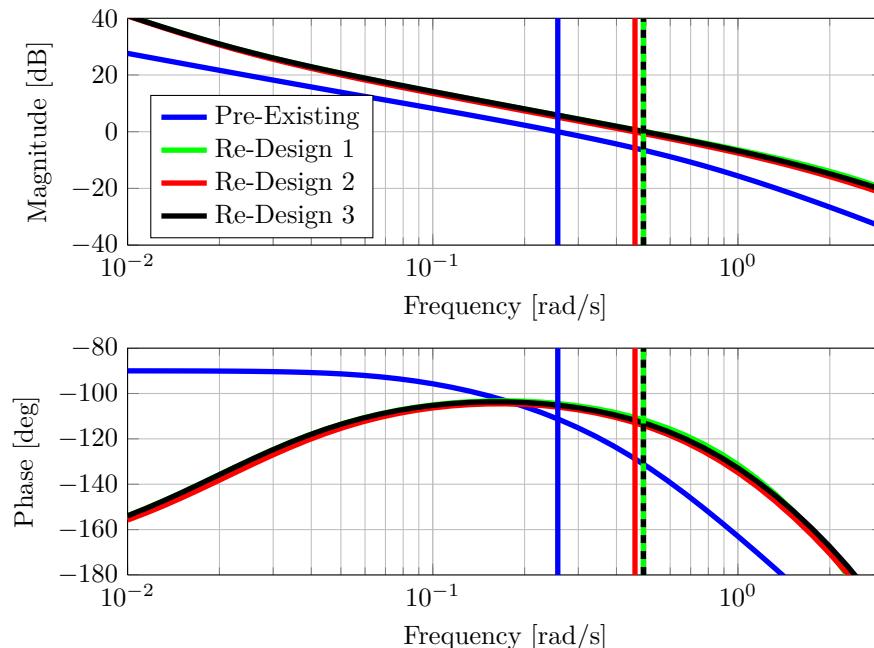
The integral gain must also be determined. The gain can be difficult to determine, as it is not known how fast the integrator needs to respond to compensate for errors in the velocity measurements. A logical approach was followed to make the integral gain as high as possible, without affecting the transient response of the position system. This was done by examining the step response after an integral gain adjustment. This integral gain was determined to be 0.01. The re-designed system was then augmented with the integral gain and integrator.

The final step is to adjust the proportional gain to obtain a gain crossover frequency with a phase margin of  $70^\circ$ . The proportional gain was determined using all three re-designed horizontal velocity controllers. The proportional gain could effectively be doubled from the pre-existing system's proportional gain of 0.24, while maintaining the desired phase margin. Characteristics of the three re-designed horizontal position controllers and the pre-existing horizontal position controller are exhibited in Table 5.5.

**Table 5.5** – Re-Designed and Pre-Existing Horizontal Position System Parameters

Parameter	Pre-Existing	Design 1	Design 2	Design 3
Proportional Gain	0.24	0.49	0.45	0.49
Integral Gain	NA	0.01	0.01	0.01
Integral Command Limit [m/s]	NA	0.142	0.142	0.142
Crossover Frequency [rad/s]	0.2	0.49	0.46	0.49
Phase Margin [ $^\circ$ ]	70	67	69	70
Bandwidth [rad/s]	0.42	0.86	0.75	0.82

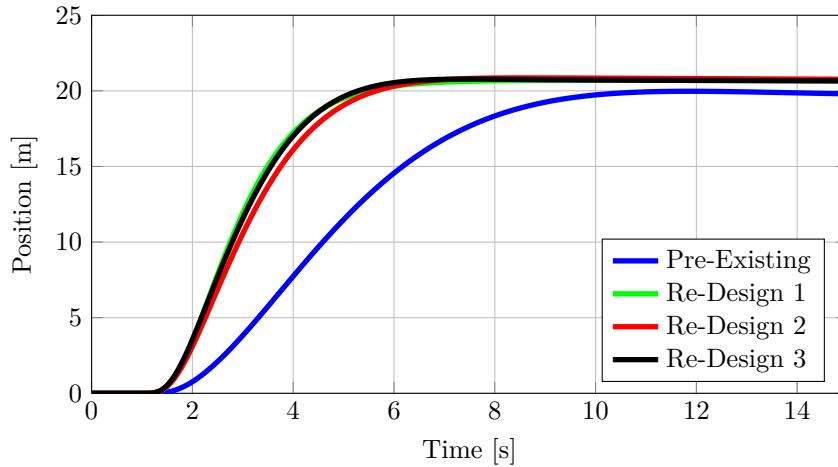
The bandwidth of the pre-existing position system is  $0.42 \text{ rad/s}$ . All three of the new designs yield a bandwidth that is approximately double that of the pre-existing horizontal position controller's bandwidth. The open-loop Bode plot of the pre-existing and all the re-designed position controllers can be seen in Figure 5.78.



**Figure 5.78** – Open-Loop Bode Plot of Pre-Existing and Re-Designed Horizontal Position Systems

The crossover frequencies of the pre-existing system and the three re-designed systems are indicated by the vertical lines. It is noticeable on the Bode plots of the three re-designed systems that low frequencies and high frequencies are amplified, when compared to the pre-existing horizontal position system. Low frequencies are amplified by the addition of the integral term to the position controller and high frequencies are amplified by the addition of the lead compensator in the re-designed horizontal velocity control loop.

The step response is looked at next. The pre-existing system and three re-designed controllers were given a unit step position reference at time  $t = 1$  second. The response can be seen in Figure 5.79. No limiters were included in order to maintain the linearity of the models.



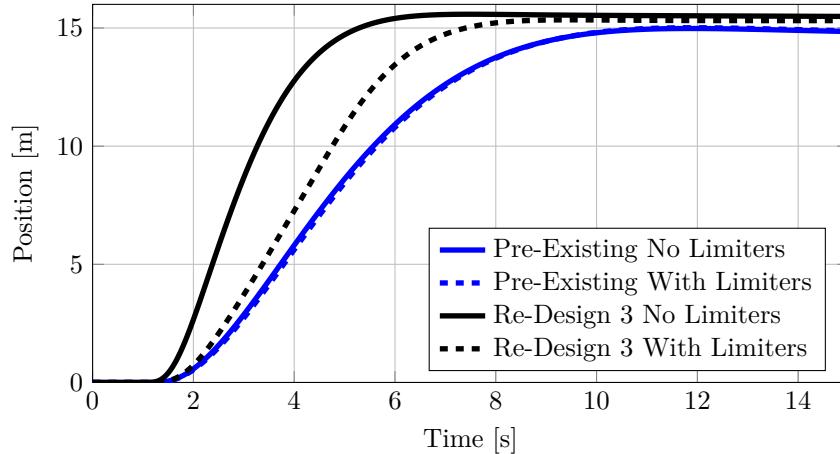
**Figure 5.79** – Step Response of Pre-Existing and Re-Designed Horizontal Position Systems

Slight overshoot is observed with the three re-designed controllers, which presumably originates from the addition of the integrator in the horizontal position controller. The pre-existing system has no overshoot whereas re-design 3 has a maximum overshoot of 3.5%. The 5% settling time of the pre-existing position controller is 7.66 seconds, which has decreased to 3.67 seconds by re-design 3. The rise time also decreased, from 5.27 seconds for the pre-existing system to 2.50 seconds for re-design 3.

The step response is also investigated when the limiters are included. The limiter of  $\pm 0.142 \text{ m/s}$  for the newly added integral term and the limiter of  $\pm 3.5 \text{ m/s}$  for the velocity reference command are included. The step response is plotted in Figure 5.80 for a reference position step of 15 m. A large reference position step command is given so that the functionality of the limiters can become obvious.

The step responses of the pre-existing system and re-design 3 are compared. When the step reference command is issued, at time  $t = 1$  second, both the pre-existing system and re-design 3 are limited by the velocity reference limit. Re-design 3, however, is limited for a longer duration of time, as the controller is more aggressive. The actuator commands were also investigated, where a slight difference in the aggressiveness of required actuation was seen. This is not considered a problem and the position controller re-design is deemed acceptable.

This concludes the re-design of the horizontal control system. The control system was upgraded for the purpose of automated landing on a translating platform. The speed-of-response of the horizontal control system has been increased for improved disturbance rejection, to enable landing within the tight bounds of the project goal. Action was taken by



**Figure 5.80** – Step Response of Pre-Existing and Re-Designed Horizontal Position Systems, With Limiters

increasing the speed-of-response of the horizontal velocity and horizontal position controllers to approximately double the speed-of-response of the pre-existing systems.

The steady-state requirements were also investigated and the horizontal control system has been modified appropriately. The position controller was augmented with an integrator to compensate for vehicle and platform velocity measurement errors and a velocity feed-forward signal was added to the horizontal velocity controller's reference.

To summarize, the following upgrades were performed:

1. The speed-of-response of the horizontal velocity control loop was increased.
2. The speed-of-response of the horizontal position control loop was increased.
3. A velocity feed-forward signal was added to the horizontal velocity controller's reference.
4. An integrator was added to the horizontal position controller.

In the next section, non-linear simulations will be performed to continue the evaluation of the re-designed controllers. The non-linear simulation will further affirm whether the design goals have been achieved.

### 5.3 Non-Linear Software Simulation

In this section, the performance of the re-designed horizontal velocity and horizontal position controllers are verified more accurately. The section will start by introducing the components of the non-linear simulation. The software simulation results will then be presented.

During the design process, linear simulations were performed to verify the performance of the re-designed horizontal velocity and position controllers. The linear simulations isolated the controller from the rest of the dynamics of the system. By performing non-linear simulations, the other dynamics are taken into account.

The non-linear aircraft model derived in Chapter 4 is used in the simulation. This model represents the aircraft more accurately than the linearized dynamics used to design the controllers. Furthermore, the non-linear simulation will indicate what effect the new horizontal

controllers have on the controllers functioning in the other axes of the vehicle, as these effects are not modelled in linear simulation. The disturbance effect that the re-designed horizontal control system has on the vertical control system and the yaw control system can therefore be investigated. If the controllers are deemed acceptable after non-linear simulation, more confidence is gained in their design, and they have a better chance of working as intended in the real-world system.

### 5.3.1 Non-Linear Simulation Setup

The non-linear simulation is referred to as a Software-In-the-Loop (SIL) simulation. The SIL simulation makes use of the same simulation environment as the Hardware-In-the-Loop (HIL) simulation, which will be presented in Chapter 8. The major difference is that in HIL simulation, the flight software is executed as embedded C code on the actual OBC, while in SIL simulation the flight software C code is called and executed directly by Matlab on the desktop computer. This saves time when testing and debugging the code, as no compiling and reprogramming of the OBC need to be done, and all of the flight control variables are available for inspection in the Matlab environment.

The pre-existing SIL simulation already contained models for the vehicle dynamics, sensor noise and wind, and was modified to add a model of the Extended Kalman Filter (EKF) used to estimate the kinematic states of the vehicles in the ESL. The EKF was implemented prior to testing the controllers, as the EKF states are used as the measurements for the controllers. The pre-existing SIL simulation did not have an EKF estimating the vehicle's states. The states were directly given to the controllers from the six-degree-of-freedom model in the Simulink environment.

It is important that the EKF is functional in the SIL simulation, as the EKF can possibly add or remove noise to the process variables of the controllers. Controllers that increase the bandwidth of the system can be negatively affected by this noise. Therefore, having the EKF in the SIL simulation will make it apparent if this is a problem.

The Simulink block diagram of the Software-In-the-Loop simulation is shown in Figure 5.81. The SIL block diagram is similar to the HIL block diagram, with minor differences.

The green Vehicle Model block contains non-linear simulation models of the vehicle dynamics and wind effects. The red Autopilot Model block contains the sensor models, and also calls and executes the C code for the flight control laws and the EKF state estimator. The sensor measurements are modelled by adding simulated measurement noise to the simulated kinematic states of the vehicle. The blue Visual Output with Moving Base block in the top right corner connects to a pre-existing visualization environment that renders a three-dimensional representation of the vehicle and its environment. The visualization environment is shown in Figure 5.82, depicting the vehicle.

The re-designed controllers were tested and compared to the pre-existing controllers by performing position steps in the non-linear simulation. Two consecutive position steps were performed during the same simulation session. The first step was 5 m in a northern direction. Twenty seconds later a 5 m step was performed in an eastern direction, allowing enough time for the vehicle to settle from the previous step. The vehicle was commanded to maintain a yaw angle of zero throughout the simulation. The simulation was repeated for all three of the new controller designs and for the pre-existing controller design.

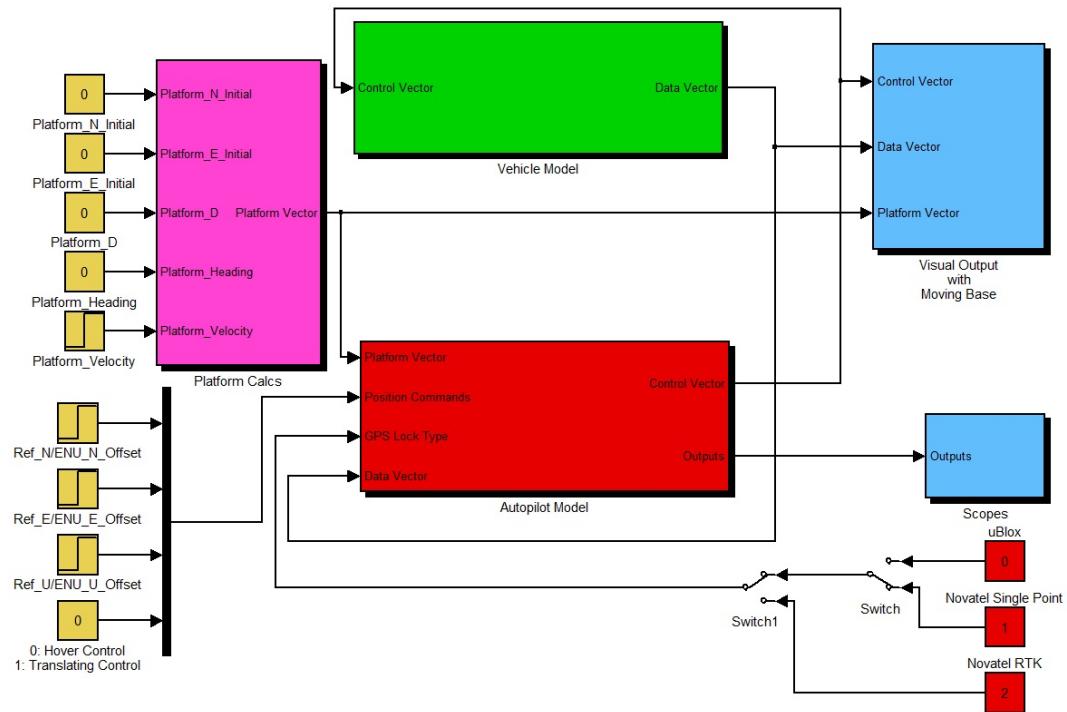


Figure 5.81 – The Simulink Block Diagram Used For SIL and HIL Simulations

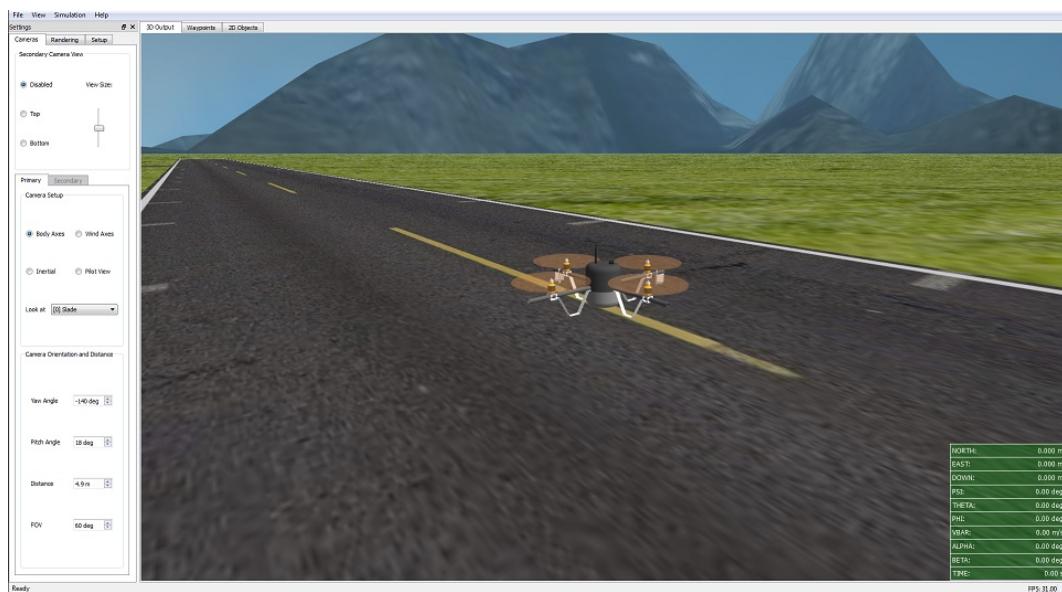


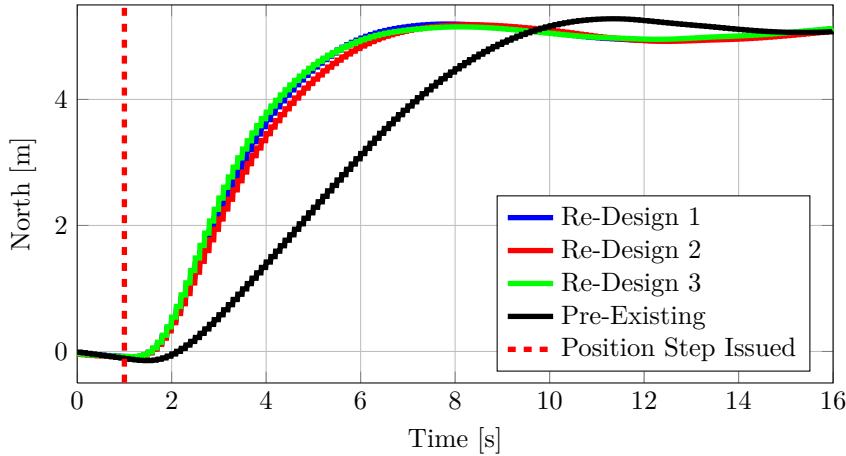
Figure 5.82 – The Three Dimensional Visualization Environment

### 5.3.2 Simulation Results

The simulation results are investigated to determine, among other things, the increased speed-of-response and to make sure that there is adequate damping in the system. These

characteristics are determined by investigating step responses.

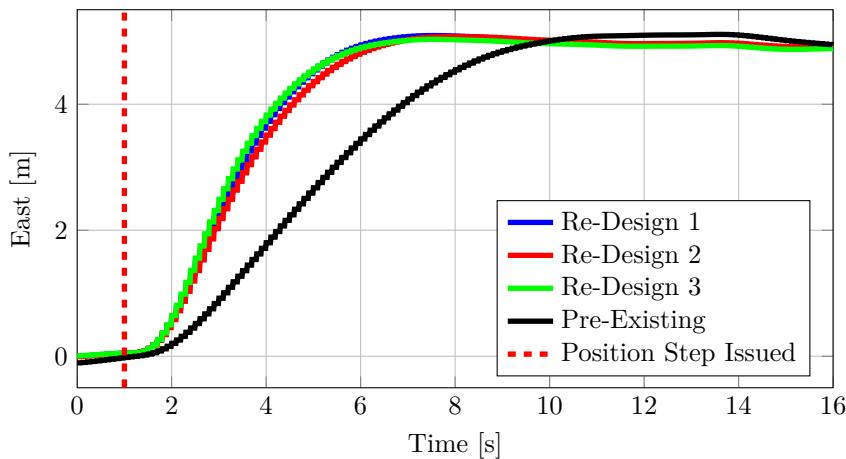
The north position step response is considered first and is shown in Figure 5.83. The 5 m position step reference was commanded at  $t = 1$  second, indicated by the vertical red dashed line.



**Figure 5.83 – Non-Linear Simulation North Position Step Response**

The pre-existing control system has a rise time of 5.1 seconds and a 5% settling time of 11 seconds. The three re-designs have very similar responses, and exhibit almost double the speed-of-response compared to the pre-existing controller. Design 2 has the slowest response of the new controllers, with a rise time of 3.3 seconds, and a 5% settling time of 4.8 seconds. The step responses also correlated sufficiently with the curve in Figure 5.80 with the limiters included, which has a rise time of 3.1 seconds and a 5% settling time of 4.7 seconds. Lastly, the step response indicates that enough damping is present in the system, as very little overshoot is observed.

The east position step response is considered next and is shown in Figure 5.84. This response should be very similar to the northern response as the vehicle is almost symmetric in the longitudinal and lateral axes. The position step command is again issued at a time of  $t = 1$  second.

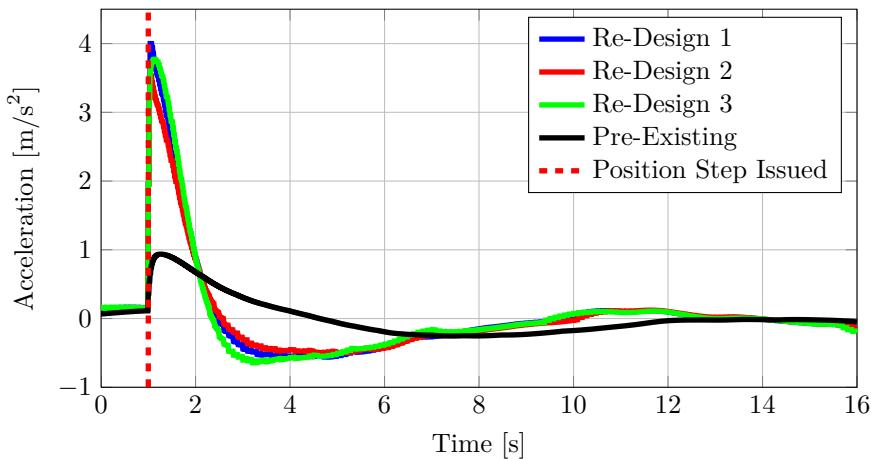


**Figure 5.84 – Non-Linear Simulation East Position Step Response**

Slightly less overshoot is observed in the east position response, compared to the north position response. The pre-existing control system has similar overshoot, and a similar speed-of-response, with a rise time of 5.4 seconds and a 5% settling time of approximately 11 seconds. The rise time is similar to the north response at 5.4 seconds. The three new designs again have very similar responses, and again exhibit almost double the speed-of-response compared to the pre-existing controller. Design 2 again has the slowest response of the new controllers, with a rise time of 3.3 seconds, and a 5% settling time of 4.8 seconds. As with the north controllers, adequate damping is observed, only yielding slight overshoot.

Limiters are also investigated, specifically the acceleration limiter, which dictates the maximum tilt angle that the aircraft can maintain. The acceleration limiter should not be limiting continuously after a reference position step command is issued. This is an indication that the aircraft is at risk of saturating its actuators.

The acceleration command during a horizontal position step of 5 m is inspected, to check the magnitude of the acceleration command relative to the acceleration limits. The vehicle is limited to accelerate translationally at a rate of  $4 \text{ m/s}^2$ . The acceleration command response for a north position step is plotted in Figure 5.85.

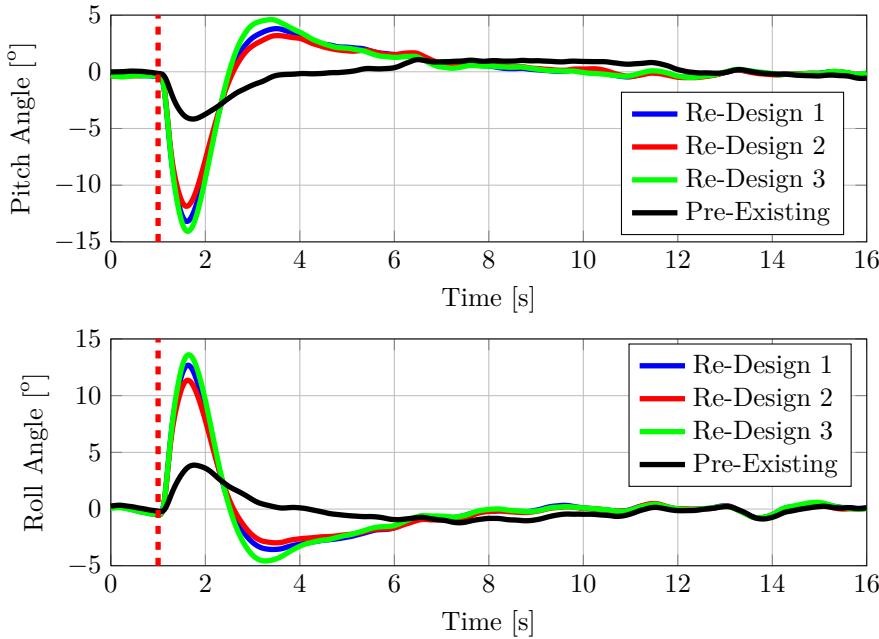


**Figure 5.85 – Non-Linear Simulation Acceleration Command During North Position Step**

The 5 m step command is issued one second into the simulation. While using the pre-existing control system, a maximum acceleration of  $1 \text{ m/s}^2$  is commanded. The three new designs command a higher acceleration, requiring more actuation. Design 1 seems to be the most aggressive, reaching the acceleration command limit for 30 ms. Design 2 appears to be the least aggressive, reaching a maximum acceleration command of  $3.6 \text{ m/s}^2$ .

The maximum commanded acceleration effectively translates to a maximum angle command. This maximum obtainable angle is  $\pm 22^\circ$ . The pitch and roll angles should also be investigated. The pitch and roll angles are both plotted in Figure 5.86. Immediately after the step command is issued at time  $t = 1$  second, the pitch and roll angles reach their highest value as the vehicle attempts to accelerate. While using the pre-existing control system, a maximum negative pitch angle of  $-4.2^\circ$  was recorded. Design 3 of the new controllers drove the vehicle to a maximum negative pitch angle of  $-14.1^\circ$ . The roll angle response exhibits similar behavior.

The re-designed horizontal velocity and position controllers definitely demand more aggressive actuation. Fortunately the required actuation is not so aggressive that the actuators are

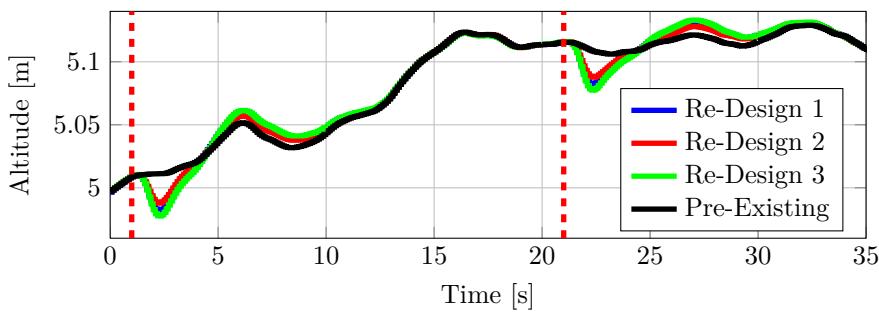


**Figure 5.86** – Non-Linear Simulation Pitch And Roll Angles During Position Steps

saturated and that the controllability of the controllers in the other axes of the aircraft is compromised.

It is, however, necessary to investigate the cross-coupling disturbances on the other control systems, due to the more aggressive horizontal controllers. For instance, the aggressive rolling and pitching of the aircraft will cause more abrupt loss in vertical thrust. This will disturb the vertical control system. It is investigated whether the vertical control system and the yaw control system are capable of handling the additional disturbance induced on them by the horizontal controllers.

The cross-coupling disturbance on the vertical control system is considered first. Figure 5.87 shows the altitude response of the vehicle to horizontal position steps. The instances where the north and east step commands are issued are also shown, represented by the vertical lines at  $t = 1$  second and  $t = 21$  seconds respectively.



**Figure 5.87** – Non-Linear Simulation Disturbance on Heave Control System

During the simulation, the height drifted by approximately 15 cm. This variation is due to Global Positioning System (GPS) drift, which is modelled in the simulation. However,

a brief drop in altitude is observed directly after each of the two position step commands is issued. This is due to the vehicle losing vertical thrust as it pitches and rolls during the north and east position steps. This behavior is not seen when using the pre-existing control system, as the pre-existing system pitches and rolls less aggressively and therefore loses less vertical thrust. Fortunately, the drop in altitude is not alarming, as it is negligible at 3 cm.

Finally, the cross-coupling disturbance on the yaw control system is investigated. During the simulated test, the vehicle was commanded to maintain a yaw angle of zero degrees. How well the vehicle regulates its heading is therefore an indication of how well the yaw control system handles the cross-coupling disturbance produced by the horizontal controllers. Dashed red lines indicate instances where the north and east step commands were issued at  $t = 1$  and  $t = 21$  seconds.

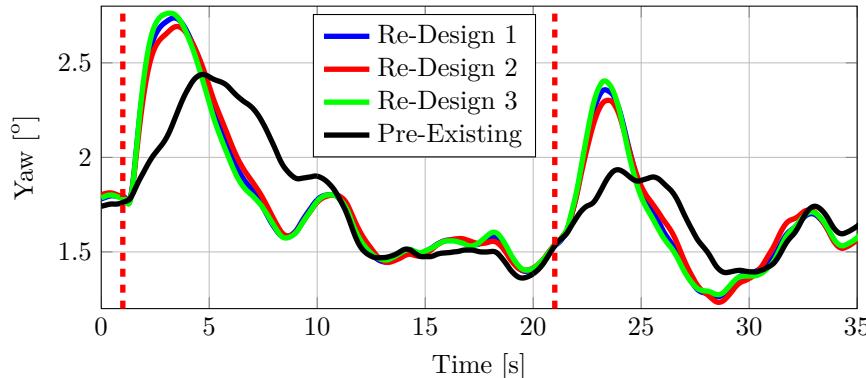


Figure 5.88 – Non-Linear Simulation Disturbance on Yaw Angle

Immediately after each of the position step commands are issued, a disturbance in the yaw angle can be seen. The disturbance in the yaw angle is approximately  $0.5^\circ$  more when using the new horizontal controllers compared to when using the pre-existing controller. The percentage increase in deviation of the yaw angle, when compared to the pre-existing system, is 40%. The increase in cross-coupling disturbance on the yaw control system is marginal and is therefore also deemed acceptable.

In conclusion, the non-linear simulation results further confirm that the re-designed horizontal velocity and position controllers function as they were designed. Their increased aggressiveness is not considered to be a problem, and their effect on the control systems controlling the other axes of the aircraft is considered negligible.

## 5.4 Discussion

The entire control system of the aircraft has been analyzed and shortcomings in the control system for the goal of landing on a translating platform were identified. The performance of the control systems responsible for vertical and yaw control of the aircraft was deemed acceptable. The analysis of the horizontal control system did, however, indicate that the horizontal velocity controller was conservatively designed, and that the control architecture of the horizontal position controller required modification for tracking a translating platform. A complete re-design of the horizontal control system followed.

Design goals were then established, before the re-design of the horizontal control system commenced. The design goals stated that the re-designed horizontal control system must track the platform with zero steady-state error and that the velocity of the vehicle must

match the velocity of the platform. There must also be compensation for velocity measurement errors in the velocity measurements of both the platform and the vehicle.

Action was taken by augmenting the horizontal position controller with an integrator, to compensate for velocity measurement errors, and a feed-forward velocity reference, which allowed the vehicle's velocity to match the platform's velocity. The augmentation of the horizontal position system also allowed for a zero steady-state tracking error.

Other design goals were also established. The speed-of-response should be fast enough to aid in disturbance rejection and to allow tracking of the landing location during non-ideal conditions, such as variation in the speed of the platform. Yet the system must still be damped well enough, so that overshoot of the landing location is not excessive.

The problem was approached by increasing the bandwidth of the horizontal velocity and horizontal position controllers, while maintaining a phase margin of  $70^\circ$ . A phase margin of  $70^\circ$  will ensure that there is enough damping in the system to combat overshoot. The bandwidth was increased by designing and implementing a lead compensator in the horizontal velocity controller, which also allowed for a higher attainable bandwidth for the horizontal position controller. The result was a speed-of-response of the re-designed horizontal velocity and horizontal position controllers that has double the speed-of-response of the pre-existing controllers. The re-designed systems also had adequate damping, thanks to the design requirement of maintaining a  $70^\circ$  phase margin.

The horizontal velocity and horizontal position controllers were also designed to be more robust, which was another design goal. Robustness of the horizontal velocity and horizontal position controllers was ensured by implementing a lead compensator that adds more phase than required at a higher crossover frequency. This design technique ensured that the phase margin of the system is less sensitive to variation in the gain crossover frequency of the system.

After the re-design was performed, which made use of linear transfer functions, non-linear SIL simulations were performed. The non-linear simulation proved that the controllers were now more aggressive, but not too aggressive for the actuation capabilities of the vehicle. The cross-coupling effect that the increased speed-of-response of the horizontal control system has on the vertical control system and the yaw control system were also investigated during non-linear simulation. The increased cross-coupling disturbance on these control systems was determined to be negligible.

The next step will be to perform HIL simulations with the new controllers, and to finally perform a flight test aimed at practically validating the performance of the re-designed controllers.

## Chapter 6

# Vehicle and Platform Sensing

This chapter concentrates on sensor-related investigations, modifications and tests that were critical to the project. The chapter starts by discussing the sensing requirements of the aircraft and the platform, specifically for the task of automatically landing on a translating platform. Some brief background is then provided on how a Differential Global Positioning System (DGPS) works, as a DGPS is the essence of the position measurement method used for the project.

The pre-existing DGPS is then introduced in detail followed by a section which documents the hardware and software modifications performed to the pre-existing DGPS. The main purpose of the modifications was to change the mode in which the DGPS functions. The new DGPS mode had to be thoroughly tried and tested.

An extensive practical test campaign, designed to test the performance of the new DGPS mode, then follows. An overview of the campaign is presented and branches out into a brief presentation of the methodology of each test, and the corresponding test's results. A discussion which closes the campaign, summarizes the findings of the practical tests. The practical test campaign highlighted factors which could be fatal during the automatic landing procedure.

The chapter ends by introducing a safety mechanism that was designed to address the safety factors identified during the practical tests. Sensor propagation of the measurements obtained from the DGPS is also performed.

### 6.1 Vehicle and Platform Sensing Requirements

The sensing requirements of the vehicle and the platform, for landing on a translating platform, are established first. Recall that the literature study indicated that the best choice, for this application, is to use a Global Positioning System (GPS)-dependent sensor to provide the measurements required for the automated landing system. The sensing requirements of the vehicle are discussed first, followed by the sensing requirements of the platform.

At the beginning of the project the vehicle already has knowledge of all its states. Its position, velocity, specific force and angular rates are measured, whereas its tilt and yaw angles are estimated from these measurements by the on-board Extended Kalman Filter. There is thus no additional sensor information required by the vehicle to describe its states. The accuracy of the position and velocity measurements of the vehicle is very important if the goal is to land on a translating platform. The aircraft's position and velocity measurements are obtained from a GPS sensor on the aircraft.

The platform that the aircraft has to land on is a translating platform that has a constant speed and heading. Position and velocity measurements are the only required information

to describe the platform's states. As with the aircraft, using a GPS sensor will allow measurement of the platform's position and velocity. But again, the accuracy of the position measurements and velocity measurements is very important.

The accuracy of the measurements obtained from the GPS sensor on the aircraft and the GPS sensor on the platform is important, because the aircraft must land on the platform with a maximum position error of  $0.5\text{ m}$ . If perfect position measurements are available, the control system can control the vehicle to within the error bounds. However, if the position measurement contains errors, the control system has no way of controlling the vehicle to land within the tight bounds. Thus, the position measurement error of the aircraft and the position measurement error of the platform combined may not exceed  $\pm 0.5\text{ m}$ .

The only addition that will therefore be needed is a GPS sensor that is fitted to the platform. The measurements from the GPS sensor on the platform can then be given to the aircraft, so that its control system has knowledge of the platform's position and velocity. Fortunately, GPSs exist which are able to provide accurate enough position and velocity measurements for the project. A system like this is referred to as a DGPS.

## 6.2 GPS and DGPS Background

In order to better understand the GPS that will be used in this project, some GPS and DGPS background is needed. A GPS sensor can be used as a stand-alone sensor, or it can be used in conjunction with other GPS sensors to function as a DGPS. A DGPS has definite advantages as opposed to using a stand-alone GPS sensor.

There is a multitude of different DGPS modes in which the two or more GPS sensors can function, with each mode having its specific advantages and disadvantages. It all depends on the application for which they will be used. This project will make use of a DGPS that is manufactured by Novatel and requires two GPS sensors to function. One of the GPS sensors is referred to as the Master sensor and the other is referred to as the Rover sensor.

The DGPS requires that both Master sensor and Rover sensor track and collect satellite data simultaneously from common satellites. Both of these sensors will then experience the same bias measurement errors. When the distance between the sensors is relatively small, the bias errors are considered to be nearly the same. The bias errors can then be effectively canceled by differential corrections, hence the name Differential Global Positioning System.

The technical challenge to using this DGPS is that communication is required between the Master sensor and the Rover sensor. The communication is required to send satellite data and information for correcting bias errors between the two sensors. Some DGPS modes require one-way communication and other DGPS modes require two-way communication, depending on the measurements that the user want to obtain from the sensors. The DGPS mode that was implemented in the quadrotor system prior to the start of the project required one-way communication, and is discussed in the next section.

## 6.3 Pre-Existing DGPS System

The DGPS implemented in the quadrotor system prior to the start of the project is introduced in this section. The hardware that is used is made known, which is also the hardware that is available for use in this project. The measurements that are obtained from this DGPS are discussed and the expected accuracy of the measurements is explained.

Two GPS sensors which are manufactured by Novatel and are capable of DGPS operation are available for this project. The two GPS sensors provide measurements in the form of packets. These packets are requested upon start-up of the sensors. Once the packets have been requested, the sensors will continue transmitting the requested packets until

the sensor is switched off or restarted. Some of the packets are synchronous and some are asynchronous, meaning that some packets can be expected at fixed time frames while others are only transmitted once certain information is known to the GPS sensor. Normal GPS packets are always available, while some packets are only available during certain DGPS modes of operation. There are also different position measurement accuracies that are available during DGPS operation. The abbreviations of the position accuracy types and their corresponding expected measurement error are exhibited in Table 6.1.

**Table 6.1** – Novatel Position Types and Their Associated Accuracies

Abbreviation	Position Type	Expected Accuracy
Single	Single Point Position	At Best 1.8 m RMS
PSRDiff	Pseudorange Differential Position	1 m SEP <sup>1</sup>
L1 Float	Floating L1 Ambiguity Position	20 cm RMS
L1 Int	Integer L1 Ambiguity Position	1 cm + 1 ppm RMS

The position accuracy of the first entry in Table 6.1 is known as Single Point. This position accuracy is obtainable from a stand-alone Novatel GPS sensor, and can provide a position measurement which is at best 1.8 *m* Root Mean Square (RMS) from the actual position of the sensor. When the sensor is used in conjunction with one or more GPS sensors, improved position measurement accuracies can be obtained through DGPS modes. The best accuracy obtainable is abbreviated as L1 Int, and can provide a position measurement accuracy of 1 *cm* + 1 *ppm* RMS of the actual position of the sensor/s.

The two GPS sensors that are available are known as the Novatel OEM-V1 and Novatel Propak-V3. In the Electronic Systems Laboratory (ESL), the OEM-V1 is used as the Rover sensor and the Propak-V3 as the Master sensor. The OEM-V1 is mounted on the vehicles and the Propak-V3 is used by the Ground Station. The hardware is specifically capable of using Novatel's RT-2 L1TE technology, which enables use of a DGPS mode known as Real-Time-Kinematic (RTK) mode. Usually very accurate DGPSs work on the L1 and L2 carrier frequencies, but L1TE technology works on the L1 carrier frequency only. The baseline, which is the maximum admissible distance between the sensors, is limited to 3 *km*, however. This is Novatel's budget precision GPS solution for L1 Int absolute Rover position accuracy. However, it is important to note that the Master sensor must remain stationary during RTK DGPS operation.

Prior to the start of the project, the DGPS was integrated to function in RTK mode. The stationary Master sensor was located next to the Ground Station and generated bias correction data, which was sent through the wireless telecommand and telecommunications link from the Ground Station to the aircraft. This mode of DGPS requires only one-way communication from the Master sensor to the Rover sensor. RTK mode was used in a previous project to perform automated landing on a stationary landing platform.

In RTK mode, two packets are available that are relevant to the states of the aircraft: an absolute position measurement packet called BestPos, and a velocity measurement packet called BestVel. BestPos is a packet that is available during normal GPS operation as well, but increases in accuracy during RTK DGPS operation. Therefore the accuracy will be Single Point during normal GPS operation, and will become L1 Int accuracy during RTK DGPS operation. BestVel will also increase in accuracy as BestPos increases.

---

<sup>1</sup>Spherical Error Probable (SEP) - For a 50th percentile, half the position measurements would fall within a sphere of this radius.

During RTK operation, the L1 Int precision BestPos packet is only available to the Rover sensor. The Master sensor has to be stationary on the ground to generate bias corrections that are sent to the Rover. Both BestPos and BestVel are synchronous packets, which implies that they are transmitted at a fixed sampling rate. The sampling rate is selectable on the Rover sensor up to 10 Hz. RTK DGPS operation is, however, not a viable sensor solution for this project, as explained in the next section.

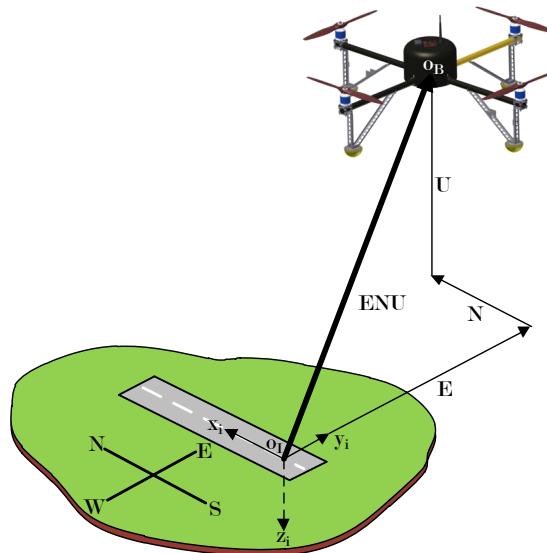
## 6.4 Proposed Sensing Method

For this project a single Master and Rover sensor are available. Consequently the Master sensor will be fixed to the platform. Using the Master sensor on the translating platform will render the RTK DGPS mode unusable.

Fortunately, Novatel does offer a DGPS mode that provides accurate position measurement data if both Master and Rover sensors are on moving objects. This mode is known as Novatel Align<sup>TM</sup> and is Novatel's solution for precise relative positioning measurements with centimeter-level accuracy. By contrast, RTK DGPS mode is Novatel's solution for precise absolute positioning measurement with centimeter-level accuracy. Align<sup>TM</sup> DGPS mode allows for Single Point precision absolute position measurements for both the Rover and the Master sensors, but L1 Int precise relative positioning measurements.

For this application, Align<sup>TM</sup> seems more beneficial than RTK, especially from a cost and practicality perspective. Maintaining the accuracy obtained by the RTK DGPS mode between the platform and vehicle requires a second Rover sensor, which is mounted on the translating platform. The stationary Master sensor can then send corrections to the Rover sensor on the vehicle and the Rover sensor on the platform. This solution is costly, as a third sensor is required. Another complication is the communication structure that is needed between the three sensors. The stationary Master sensor must always be within communication distance of both Rover sensors. Thus if a ship is out at sea, far away from land where a stationary Master sensor generates correction data, the communication structure becomes complicated and costly. When using Novatel Align<sup>TM</sup>, no stationary Master antenna is required and communication is only needed between the Master sensor and Rover sensor, which will be near each other during landing operations.

When the two GPS sensors are functioning in Align<sup>TM</sup> DGPS mode, a new packet can be requested from both sensors. The packet is called the East-North-Up (ENU) vector. This vector provides the position of the Rover relative to the Master in inertial coordinates. Refer to Figure 6.1 to see how the ENU vector is defined if the Master sensor is at the origin of the inertial axis system.



**Figure 6.1 – ENU Vector Definition**

It is important to note that although the ENU vector is defined in the same directions as the NED axes, it does not use the same sign convention as the inertial axis defined for the project in Chapter 4.

Unfortunately, the OEM-V1 GPS sensor used on the vehicle is end-of-life hardware, and is no longer supported by Novatel. To allow Align<sup>TM</sup> to function, Novatel technical support suggested that the Propak-V3 be used as the Rover sensor and the OEM-V1 be used as the Master sensor. The Master sensor will therefore be located on the aircraft and the Rover sensor on the platform. This does not require any hardware modifications, but it does require that the sensor software be configured differently. Therefore the ENU vector will be defined from the aircraft to the platform.

The vehicle cannot, however, solely depend on the ENU vector for position measurements. The vehicle requires its own absolute position measurements for the on-board kinematic state estimator to function. When using Align<sup>TM</sup> DGPS mode the Master sensor, which is mounted on the vehicle, can still receive its own absolute position measurement. The packet that contains the measurement is known as MasterPos and will provide the absolute latitude, longitude and height above the Mean Sea Level (MSL) ellipsoid. A similar packet is available to the Rover sensor. The Rover sensor is mounted on the platform and a packet called RoverPos will contain the platform's absolute position data. Recall that the absolute position packets will have Single Point precision when using Align<sup>TM</sup>.

When using the Align<sup>TM</sup> DGPS mode, the BestPos and BestVel packets are also still available to both the Master and the Rover sensors. BestPos and BestVel are normal GPS packets and are therefore always available. The accuracy of BestPos will be Single Point precision.

The proposed strategy is to use MasterPos as the position measurement for normal flight control, and the ENU vector while the landing procedure is in progress. Position measurement will therefore be relative while the landing procedure is in progress and absolute for all other flight control purposes.

A disadvantage of using Align<sup>TM</sup> DGPS mode is that the ENU vector, MasterPos and RoverPos, are asynchronous packets. This implies that these DGPS packets are not transmitted at a fixed sampling rate, but rather at a variable sampling rate that is dependent on the instantaneous quality of the DGPS communication link. The sampling rate can vary from 0 to 10 Hz.

A useful attribute of the Align<sup>TM</sup> DGPS mode is that some packets are common to both the Rover and Master sensors. All packets containing position and velocity measurements that are available to the vehicle (Master) and the platform (Rover) are summarized in Tables 6.2 and 6.3.

**Table 6.2 – Packets Available to the Vehicle (Master)**

packet Name	packet Description	Accuracy	Synch/Asynch
MasterPos	Align <sup>TM</sup> Vehicle Absolute Position	Single	Asynchronous
RoverPos	Align <sup>TM</sup> Platform Absolute Position	Single	Asynchronous
ENU	Align <sup>TM</sup> Relative Position Vector	L1 Int	Asynchronous
BestPos	GPS Vehicle Absolute Position	Single	Synchronous
BestVel	GPS Vehicle Velocity Data	NA	Synchronous

**Table 6.3** – Packets Available to the Platform (Rover)

packet Name	packet Description	Accuracy	Synch/Asynch
MasterPos	Align <sup>TM</sup> Vehicle Absolute Position	Single	Asynchronous
RoverPos	Align <sup>TM</sup> Platform Absolute Position	Single	Asynchronous
ENU	Align <sup>TM</sup> Relative Position Vector	L1 Int	Asynchronous
BestPos	GPS Platform Absolute Position	Single	Synchronous
BestVel	GPS Platform Velocity Data	NA	Synchronous

Note from Tables 6.2 and 6.3 that the platform’s velocity is not a packet that is common to both the Rover and Master sensors, and that the platform velocity measurement is therefore not available to the vehicle directly from its GPS sensor. Yet, the vehicle requires the platform velocity measurement if velocity feed-forward is used to support the feedback control. The velocity and heading information of the platform will therefore have to be sent to the vehicle through the telecommand and telemetry link.

In conclusion, it was decided to use Novatel’s Align<sup>TM</sup> DGPS mode as the sensing solution for the project. The relevant position measurement data provides centimeter-level accuracy, which is accurate enough for position measurements for the control system. However, extensive experimental tests must be performed to verify that each of the packets that are available is accurate and reliable enough to be used. Hardware and software implementation challenges, which are required to change from RTK DGPS mode to Align<sup>TM</sup> DGPS mode, are presented in the following section.

## 6.5 Re-Designed GPS and DGPS Hardware and Software Architecture

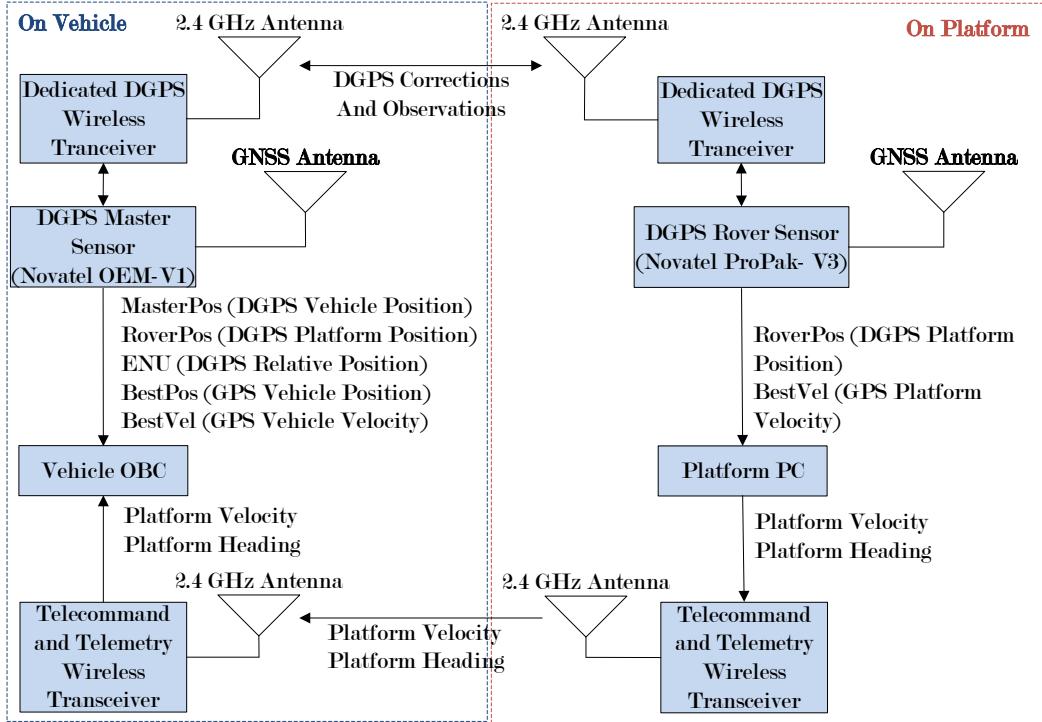
To implement the new DGPS mode, hardware and software modifications had to be made. This posed new challenges for the project, especially from a communications perspective.

### 6.5.1 Hardware Modifications

To allow the Master sensor and Rover sensor to function in Align<sup>TM</sup> DGPS mode, two-way communication is required. By contrast, one-way communication is required between the sensors if they are functioning in RTK DGPS mode. Furthermore, to maintain a sampling rate of 10 Hz for all the Align<sup>TM</sup> packets, Novatel recommends that a communications link that communicates at a baud rate of 230400 b/s be installed between the Master sensor and Rover sensor. The existing communications link that is used between the vehicle and Ground Station communicates at a baud rate of 9600 b/s. This link is normally used for vehicle telecommands and telemetry, but is shared to also transmit DGPS corrections when the RTK mode is used. Unfortunately, the link lacks the bandwidth that is needed for Align<sup>TM</sup> DGPS communication and already carries a considerable communications load. Other options for a communication link must be explored.

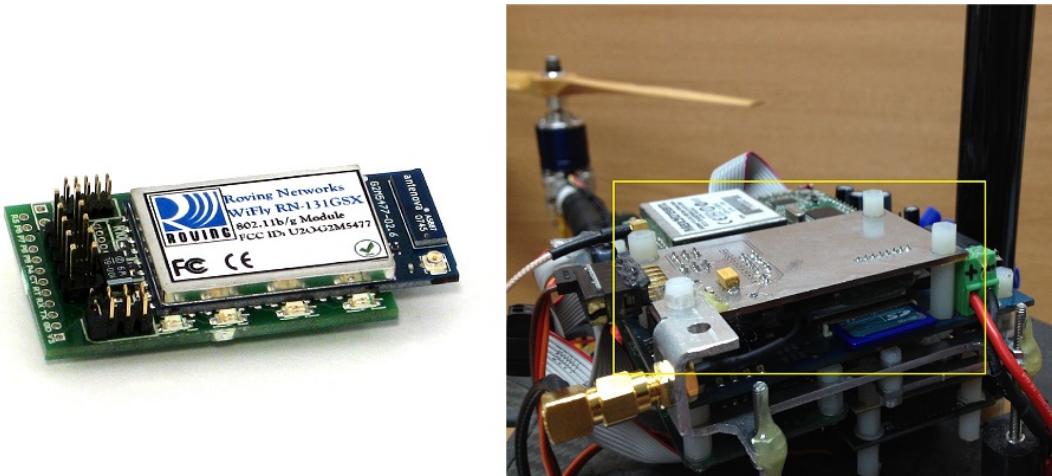
The two options are to replace the existing shared link with a high-performance link or to add an additional dedicated link to handle DGPS communication only. It was observed that the existing link has a good track record in the ESL, and is robust and reliable. This link will therefore stay in use and a secondary wireless link will be implemented. See Figure 6.2 where the final communication structure can be seen to allow for vehicle and platform position and velocity sensing in Align<sup>TM</sup> DGPS mode. The diagram also displays how the

vehicle will receive all the information so that the vehicle has knowledge of all the platform's states.



**Figure 6.2 – Platform and Vehicle Position and Velocity Sensing Communication Structure**

Several wireless link options were considered, and Roving Networks' Wifly wireless modules were chosen. The reason for using these modules is that they have a very low power consumption, are lightweight and most importantly have a very small form factor. This makes it possible to add the module to the existing avionics of the vehicle. These modules are capable of communicating at a speed of  $54\text{ Mb/s}$  and can interface with the Novatel sensor at a baud rate of  $230400\text{ b/s}$ . The Wifly unit is shown in Figure 6.3. Also shown in Figure 6.3, enclosed by the yellow rectangle, is the breakout board that was designed to interface the Wifly unit with the existing avionics.



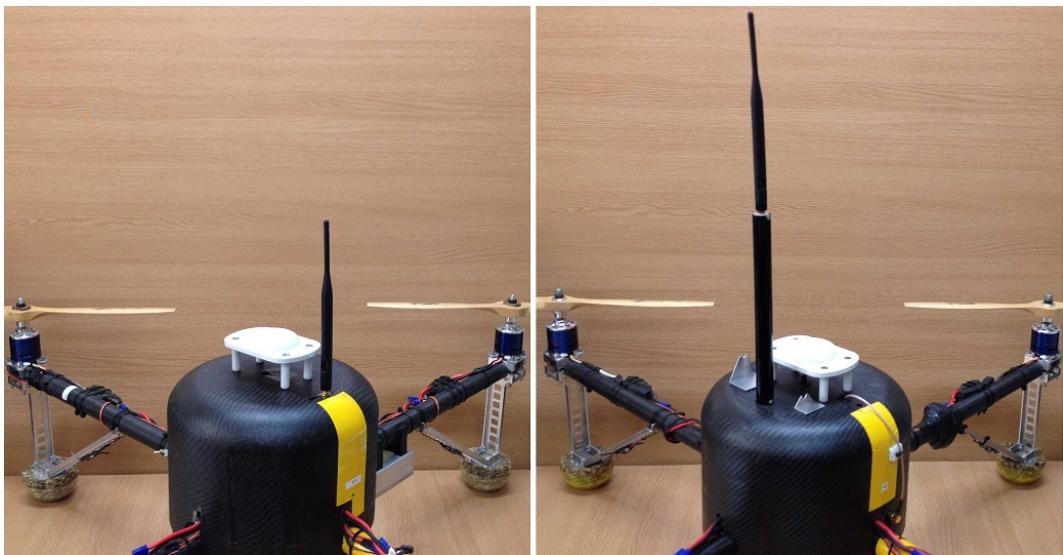
**Figure 6.3 – The Wifly Module Alone and Installed on the Breakout Board**

A test was performed to verify the performance of the Wifly modules. The performance could be affected by two factors: the distance between the DGPS sensors and radio interference from other communication sources on the vehicle or Ground Station. The transceiver used for telecommand and telemetry communication, the RC transmitter and Wifly module all use frequencies in the vicinity of  $2.4\text{ GHz}$  to communicate. The telecommand and telemetry transceiver and the RC transmitter incorporate frequency hopping for robustness, while the Wifly unit communicates on a single selectable frequency. This makes the Wifly module more susceptible to interference from other transmitters.

The test was done by sending known data from one Wifly module to another. The data was then immediately returned from the receiving Wifly unit. It could then be determined how many packets were lost and what the delay was between sending to receiving the packets. These tests showed that the packet loss encountered over distances of more than  $20\text{ m}$  was unacceptable at a communication speed of  $54\text{ Mb/s}$ . Furthermore it was noticed that the telecommand and telemetry transceiver had a noticeable negative effect on the performance of the Wifly units, whereas the RC transmitter did not have any measurable impact.

The best results that were obtained were at a communication speed of  $1\text{ Mb/s}$ , which is the slowest speed at which the Wifly units can operate. This speed was still sufficient for DGPS communication and proved to be the most robust, showing the least packet loss. The DGPS communications link functioned adequately at distances of up to  $80\text{ m}$ . It was also determined that the best location for the antenna of the Wifly units was right below the telecommand and telemetry transceiver's antenna. This was determined by trial and error.

The final antenna configuration can be seen in Figure 6.4. The original antenna configuration is on the left, showing only the telecommand and telemetry antenna. The new antenna assembly is on the right, showing the telecommand and telemetry antenna above the Wifly antenna. The Wifly antenna is mounted inside a PVC pipe using machined fixtures. The Computer Aided Drawings of the antenna tube and fixtures can be seen in Appendix D. A similar configuration was adapted on the antenna assembly of the telecommand and telemetry and Wifly module on the Ground Station.

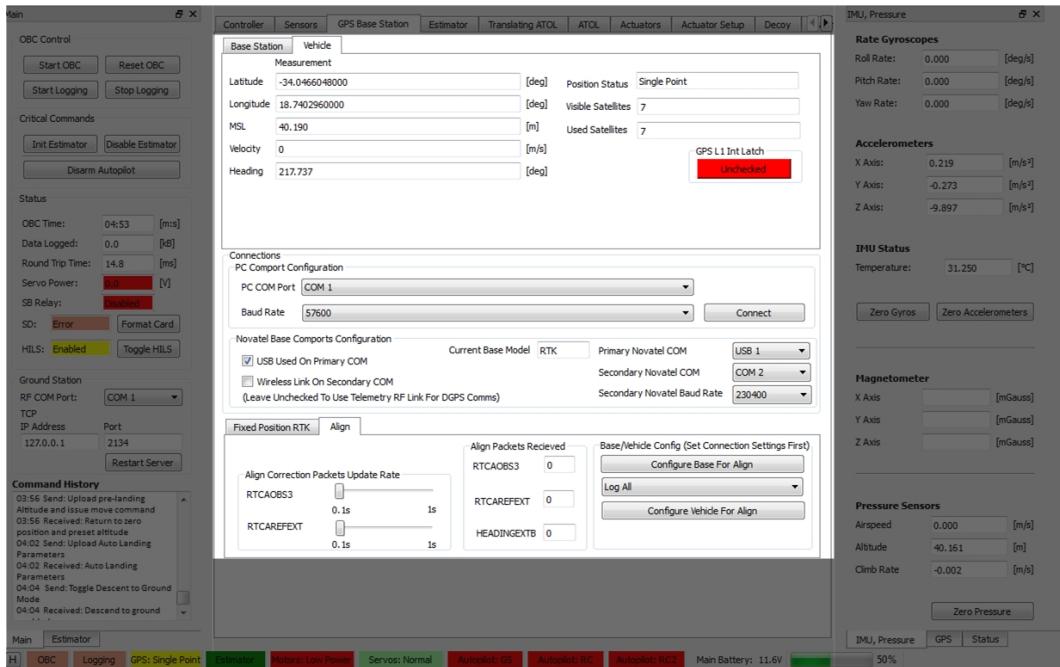


**Figure 6.4 –** The Wifly Module Positioned Below the Telecommand and Telemetry Antenna

### 6.5.2 Software Modifications

To enable Align<sup>TM</sup> DGPS mode to function, changes had to be made to the firmware on the vehicle and the Ground Station. The software changes had to be made for two reasons: To configure the DGPS sensors to function in Align<sup>TM</sup> mode, and to recognize and extract the new Align<sup>TM</sup> packets. All the software was modified so that the RTK mode could still be used if required.

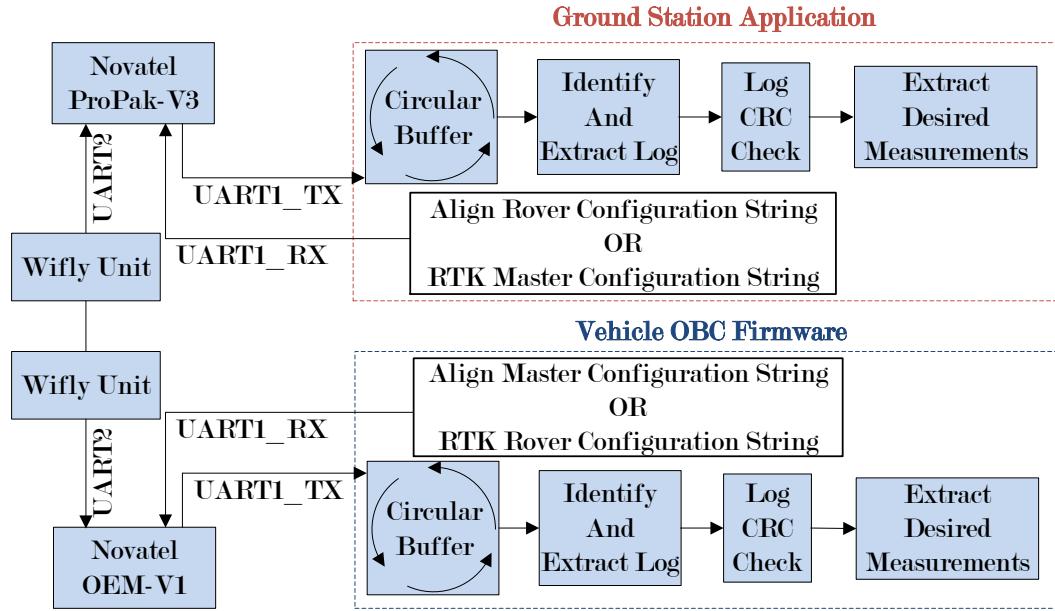
Previously, the Ground Station consisted of two applications. The primary application handled communication with the vehicle, and the secondary application handled configuration and packet extraction of the GPS sensor on the Ground Station. The Ground Station software was modified and now consists of one application which performs the task of both applications. The secondary application is incorporated into the primary application. The tab of the base station, which effectively replaces the secondary application and handles GPS and DGPS configuration, is displayed in Figure 6.5.



**Figure 6.5** – The Tab Used for Configuring the Vehicle and Ground Station for Align<sup>TM</sup> or RTK Operation

After the sensors have been configured at start-up, a circular buffer becomes active on the Ground Station and the OBC. The data stream from the GPS sensors is then parsed and the desired measurements are extracted for use. A simple schematic of this software structure is presented in Figure 6.6.

An average sampling rate of 5 Hz was obtained for the Align<sup>TM</sup> packets after all the hardware and software implementations. The sampling time varied between 0 Hz and 8 Hz, but rarely dropped below 3 Hz. It was decided that this sampling rate is sufficient, and that practical testing of the position and velocity measurements could be started.



**Figure 6.6 – The Software Structure Used to Extract GPS Measurements on the Ground Station and OBC**

## 6.6 Sensing Validation Tests

In this section, practical tests that were conducted on sensors that are considered crucial for the automated landing strategy are presented. Several practical tests were designed and performed, each with a specific goal. A test plan is firstly presented, followed by a brief overview of how each test was executed and the results from the corresponding test. A discussion on the results obtained is then presented.

### 6.6.1 Sensing Validation Test Plan

The overall goal of the tests is to ensure the reliability and integrity of the measurements obtained from the GPS sensors functioning in Align<sup>TM</sup> DGPS mode. A single test was also conducted to investigate the effect the metal trailer and platform has on the magnetometer of the aircraft.

The chronological order in which the tests were performed is shown in Table 6.4. The table illustrates the train of thought through the sensing validation test campaign. Exhibited in the table is the name of each test that was performed, alongside the specific goal of that test.

**Table 6.4 – Sensor Test Campaign Strategy**

Test Name	Test Goal
Static Test	Test RoverPos, MasterPos & ENU accuracy when stationary
DGPS Communication Failure Test	Test Align <sup>TM</sup> packets during DGPS communication failure
Dynamic Test - Master Moving	Test ENU vector accuracy while Master sensor is moving
Dynamic Test - Rover Moving	Test ENU vector accuracy while Rover sensor is moving
Dynamic Test - Both Moving	Test ENU vector accuracy both sensors are moving
Dynamic Test - RC Piloted Flight	Test ENU vector accuracy while aircraft is airborne
Dynamic Test - Autopiloted Flight	Test ENU vector for autopilot use
Velocity Measurement Error Test	Quantify Master and Rover velocity measurement errors
Platform Velocity Send Rate Test	Determine attainable platform velocity update rate
Magnetic Interference Tests	Magnetic disturbance of platform on magnetometer

Three Align<sup>TM</sup> packets are investigated for possible use, namely MasterPos, RoverPos and the ENU vector. These packets are looked into during the first test. The test will indicate whether the accuracies that Novatel states for the packets are what is observed in practice.

A DGPS communication failure test is then performed, so that the expected behavior of the Align<sup>TM</sup> packets during such an event is understood. When the behavior of the packets is understood, an appropriate contingency plan can be designed and put in place.

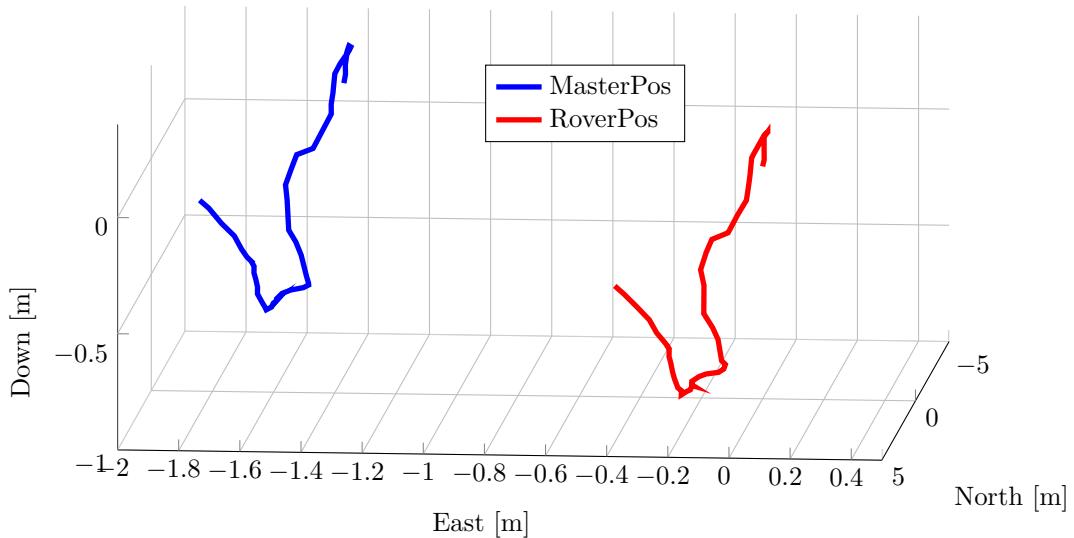
A series of tests then follows, which concentrates on the performance of the ENU vector when the Master and Rover sensors are moving. Four instances are investigated: An instance where the Master or Rover sensor is moving, an instance when both sensors are moving simultaneously, and finally where the vehicle is in RC piloted flight and under autopiloted flight.

The final test is conducted to investigate the disturbance that the metal trailer and platform have on the magnetometer of the aircraft. The earth's magnetic field will be slightly distorted around the metal trailer, which can lead to false orientation measurements. The yaw angle estimate of the aircraft will be most affected by this disturbance and could make the yaw control system malfunction during the landing procedure.

### 6.6.2 Static Test

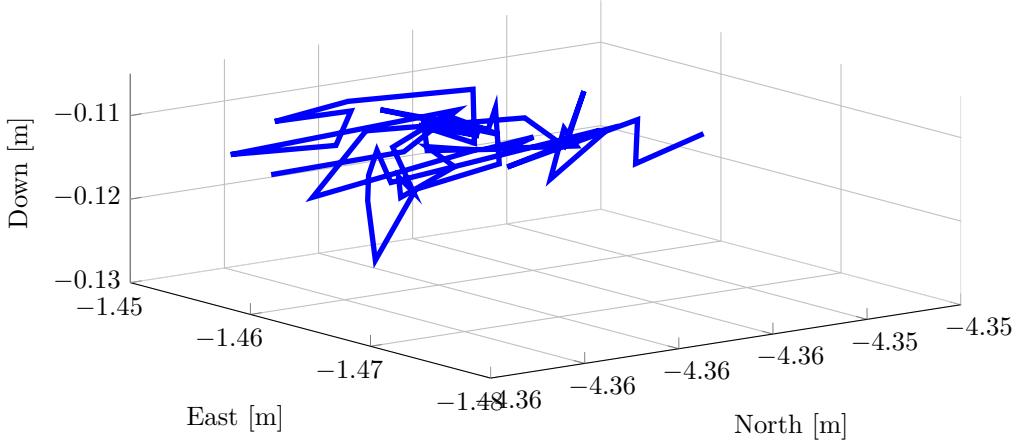
A stationary test was performed to analyze the bias errors experienced by MasterPos, RoverPos and the ENU vector. The test will indicate what the absolute accuracy of the MasterPos and RoverPos packets are and what the accuracy of the relative position is between the Master sensor and the Rover sensor.

The sensors were left stationary for a duration of 6 minutes to gather position data. The drift experienced by MasterPos and RoverPos can be seen in Figure 6.7.



**Figure 6.7 – MasterPos and RoverPos Bias Error**

In Figure 6.7 MasterPos and RoverPos experienced approximately the same drift. Both sensors' absolute position measurements drifted approximately 0.5 m north, 0.5 m east and 1 m vertically. By contrast, the ENU vector experienced significantly less drift. The ENU vector drift for the the same test can be seen in Figure 6.8.



**Figure 6.8 – ENU Bias Error**

The ENU vector drifted approximately 1 *cm* north, 2 *cm* east and 2.5 *cm* vertically. This is considerably less than the drift experienced by MasterPos and RoverPos.

As MasterPos and RoverPos experienced the same drift, their relative position measurements remained nearly unchanged. The cancellation of the bias errors is portrayed in the comparison of MasterPos, RoverPos and the ENU vector. The test proved that MasterPos and RoverPos are Single Point accuracy packets and the ENU vector is a L1 Int accurate packet.

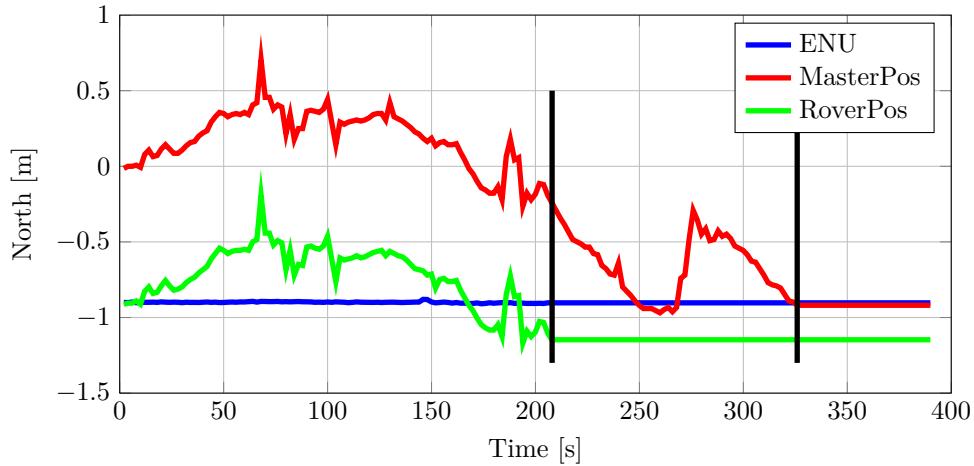
### 6.6.3 DGPS Communication Failure Test

As DGPS functionality is dependent on the communications link between the Master and Rover sensors, the scenario of a DGPS communication link failure should be considered. It is especially important to know what will happen to MasterPos and the ENU vector, as the automated landing strategy will use these two packets during different stages in flight.

The Master sensor and the Rover sensor were again left stationary for the duration of the test. A serial cable was used to simulate an ideal communication link between the two GPS sensors. The cable connected directly to both the Master and Rover sensors' serial ports. Halfway through the test, the cable was unplugged and communication was disrupted between the sensors. Recall that MasterPos, RoverPos and the ENU vector are extractable from both the Rover and the Master sensor. It is, however, only necessary to consider what happens to these packets on the Master sensor, as it is the source of absolute position measurements and the source of relative position measurements between the platform and the aircraft, which will be used by the flight control system.

What was observed was that the ENU vector and RoverPos immediately become stagnant. MasterPos on the other hand, continued functioning for an arbitrary amount of time and then also became stagnant. The response of the packets for this test can be seen in Figure 6.9.

The first vertical black line, at approximately  $t = 210$  seconds, is the instance where the cable was unplugged, and RoverPos and the ENU vector became stagnant. The second vertical black line at  $t = 326$  seconds is the moment where MasterPos also became stagnant. The test was repeated several times and the same results were seen.



**Figure 6.9 – DGPS Communication Failure Test.**

Based on these results, it was decided to use BestPos instead of MasterPos as the absolute position measurement for the vehicle's kinematic state estimator. This makes more sense from a safety perspective. BestPos is safer to use due to it being a GPS packet. A GPS packet will always be available to the sensor and rids the absolute position measurement of DGPS communication dependability. Therefore, if DGPS communication is disrupted during flight, there will always be a position measurement packet available to revert to. Furthermore, BestPos can be requested at a constant rate of 10 Hz, whereas the sample rate of MasterPos is varying and again dependent on the quality of DGPS communication. In terms of accuracy, there is no gain or loss as BestPos and MasterPos are both Single Point position accuracy packets.

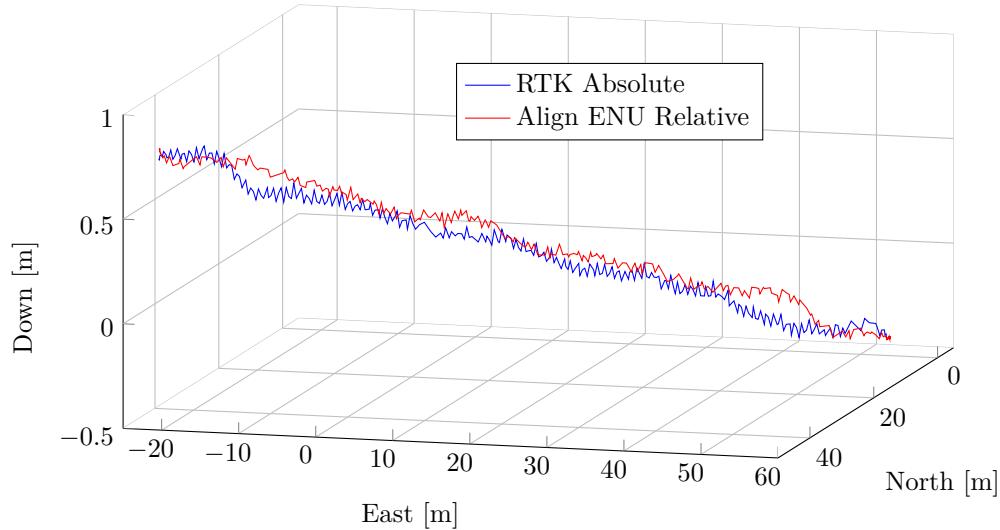
The results from the test make it apparent that a method for determining a DGPS communication failure must be implemented for safety. The ENU vector will still be used as the position measurement during the automated landing process, and will become stagnant during a communication failure. This occurrence will be fatal to the aircraft. Some form of contingency will have to be designed and implemented.

#### 6.6.4 Dynamic Test - Master Sensor Moving

The first dynamic test was performed by moving the Master sensor, which is mounted on the aircraft. The Rover sensor remained stationary for the entire test. The aim was to validate the accuracy of the ENU vector while the Master sensor is in motion.

A ground truth for the test was established by first performing the test in RTK DGPS mode. This data is taken as control for the test. The test is then repeated but with the GPS sensors functioning in Align<sup>TM</sup> DGPS mode. The relative measurements of the ENU vector can then be mapped to absolute measurements by adding the ENU vector to the absolute position of the stationary Rover antenna. The measurements obtained from these two methods can then be compared.

A straight line was marked on a sports field. The vehicle was lifted above head height and carried from a designated starting point to a designated end point. The results of the test are shown in Figure 6.10.



**Figure 6.10 – Master Sensor Movement Test Result**

The measured ENU vector in Align<sup>TM</sup> mode and the measured position in RTK mode both show how the vehicle was carried from the same starting point to the same end point. The length of the line on the field was 88.9 m and the elevation of the field decreased by 1.3 m from the start point to the end point. The results indicate sufficient correlation between the accuracy of the RTK absolute position measurements and the Align<sup>TM</sup> ENU measurements.

The test also served as a range check for the dedicated DGPS communication link. Throughout the test the communication quality was adequate enough for the ENU vector to be transmitted from the GPS sensors at an average rate of 5 Hz.

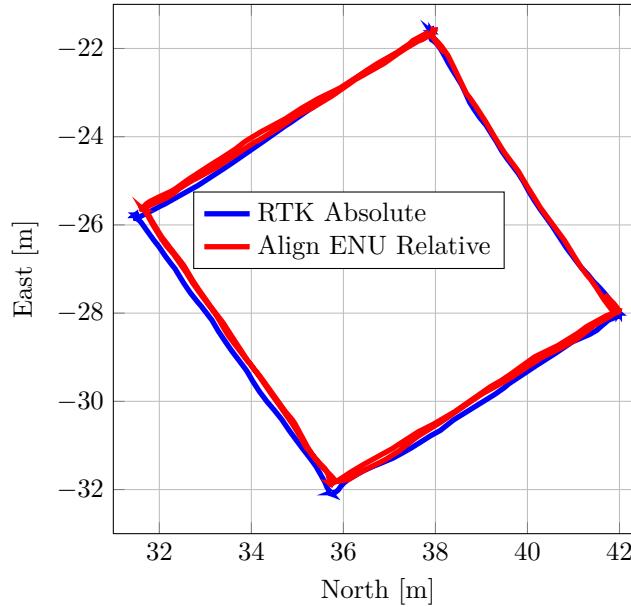
### 6.6.5 Dynamic Test - Rover Sensor Moving

The second dynamic test was performed with the Rover sensor in motion and the Master sensor stationary. The aim of the test was again to determine the accuracy of the ENU vector, but this time while the Rover sensor is in motion.

The test was performed by moving the Rover sensor in a pre-determined square around the Ground Station setup, while the aircraft which housed the Master sensor remained stationary on the ground.

Again a ground truth was determined from absolute measurements from RTK DGPS mode so that there is data to compare the ENU vector with. Recall that the Rover sensor must be stationary to perform the test in RTK DGPS mode. The Master antenna, which is fixed to the vehicle, was carried above head height around the square to obtain the ground truth measurements in RTK DGPS mode.

During the Align<sup>TM</sup> DGPS test the vehicle remained completely stationary and the Rover antenna was carried at head height around the square. Figure 6.11 shows where the Align<sup>TM</sup> ENU measurements are mapped to the absolute RTK measurements.

**Figure 6.11** – Rover Sensor Movement Test Result

Sufficient correlation is again seen between the Align<sup>TM</sup> ENU vector and the RTK BestPos measurements with only slight offsets. These offsets are very likely due to the RTK and Align<sup>TM</sup> test not being repeated perfectly.

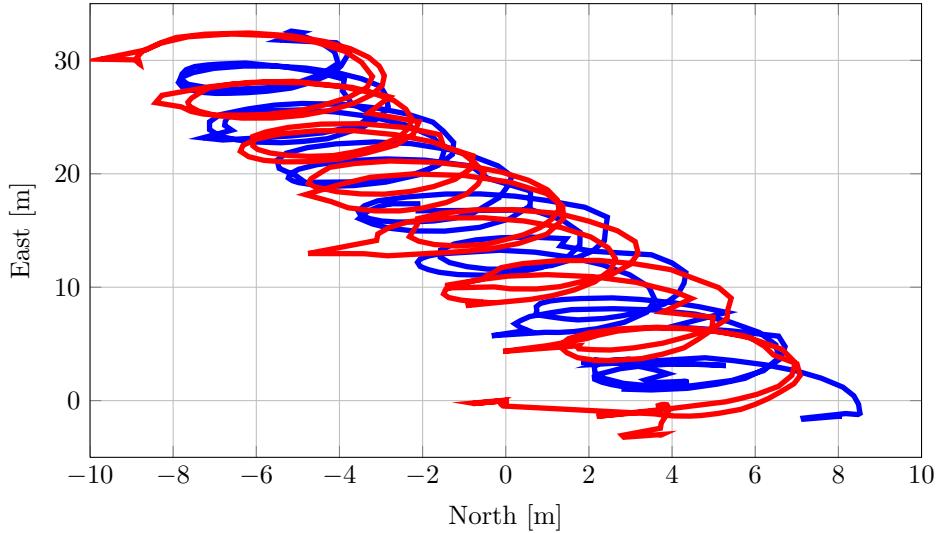
#### 6.6.6 Dynamic Test - Both Sensors Moving

For the third test both sensors were moved simultaneously. The test was performed to verify whether the ENU vector retains its accuracy while both Master sensor and Rover sensor are in motion.

A movable test rig was constructed and can be seen in Figure 6.12. A trolley housed both the Rover and the Master antennae at a fixed distance from each other. The Master and Rover antennae locations are indicated with yellow squares and are separated by a distance of 4.589 m. This distance is the ground truth for the test.

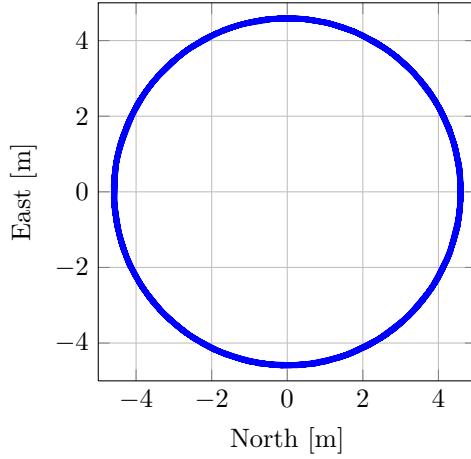
**Figure 6.12** – Movable Align<sup>TM</sup> DGPS Test Rig

The trolley was translated and rotated at the same time. MasterPos, RoverPos and the ENU vector were logged. In this test their accuracies cannot be compared, but the absolute measurements obtained indicate how much movement was done during the test. The absolute measurements obtained by MasterPos and RoverPos are shown in Figure 6.13.



**Figure 6.13** – Logged MasterPos and RoverPos Measurements During Translating and Rotating Test

While the trolley is translated and rotated, the ENU vector should form a circle. This is due to the Rover sensor moving around the Master sensor at a fixed distance, which is the ground truth distance of 4.589 m. The logged data of the ENU vector is plotted in Figure 6.14.



**Figure 6.14** – Logged ENU Measurements During Translating and Rotating Test

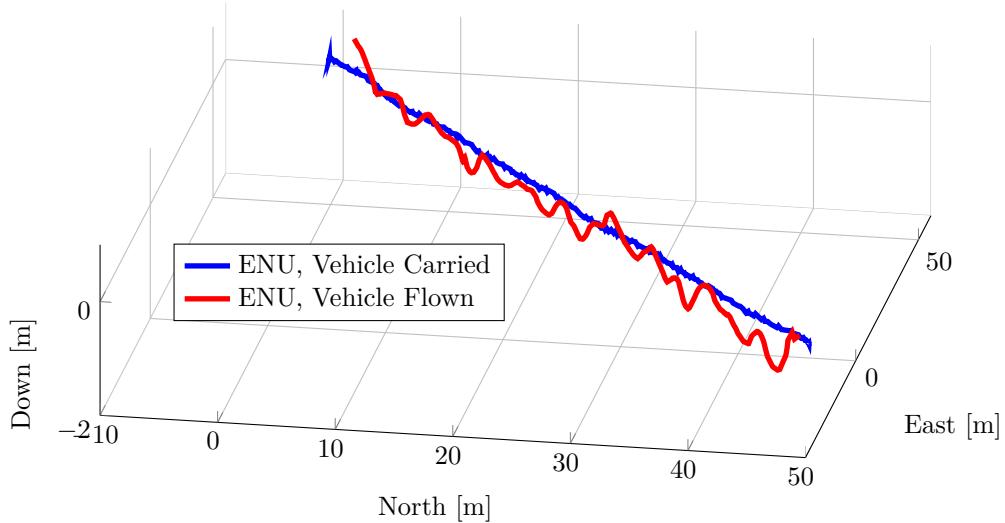
Figure 6.14 shows that an almost perfect circle is formed by the relative measurements of the ENU vector. The radius of the circle must be the ground truth distance. The radius of the circle indeed agreed within 2 cm with the ground truth distance between the two GPS

sensors. The test results prove that the ENU vector retains its accuracy while both GPS sensors are in motion.

### 6.6.7 Dynamic Test - RC Piloted Flight

The second-last dynamic position measurement test consisted of an RC piloted flight. The RC piloted test was done to verify that the dedicated wireless DGPS communications link and Align<sup>TM</sup> DGPS mode function normally while the rotors are active and the vehicle is airborne. The RC piloted flight used a similar trajectory as the first dynamic test, where the vehicle was carried down a straight line on the sports field.

The ground truth dataset for the RC piloted flight test was taken from the first dynamic test that was performed. The ENU vector measurements obtained from walking along the straight line on the sports field can be compared to the ENU data obtained from the RC piloted flight. The safety pilot attempted to fly the vehicle at head height above the line at walking pace. Figure 6.15 shows the test results.



**Figure 6.15 – ENU Measurements When the Vehicle Was Flown and Carried**

The ENU vector appeared to be functioning normally while the vehicle was airborne. More variation in the height of the ENU vector can be seen, and is a result of the RC pilot attempting to regulate the height of the aircraft. Acceptable correlation is seen between the ground truth dataset and the data obtained during the test.

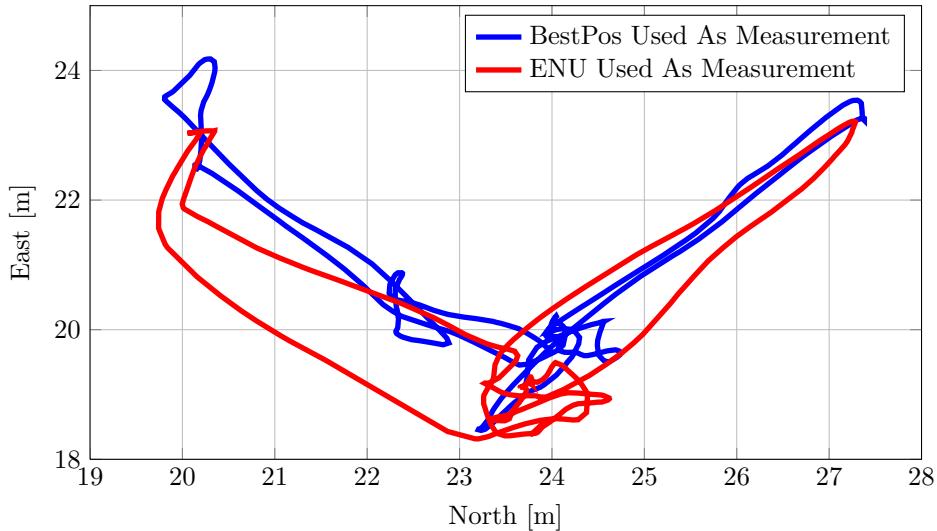
The average update rate of the ENU vector was still 5 Hz, indicating that the active motors and rotors had no effect on the quality of the dedicated DGPS communication link.

### 6.6.8 Dynamic Test - Autopiloted Flight

The results from all the static and dynamic tests confirmed that the ENU vector provides reliable and accurate enough relative position data between the Master sensor and the Rover sensor. The results provided enough confidence to start testing the autopilot of the aircraft. The test will indicate whether the control system on the vehicle operated correctly when using the ENU vector as the position feedback sensor measurement to control the vehicle. The pre-existing control system was used during this test.

The vehicle was commanded to step forward, back, left and right in 5 m increments. The step commands were first performed using the normal BestPos GPS packet as the position measurement and then repeated using the ENU vector as the position measurement. The relative measurements of the ENU vector measurements were then mapped to absolute positions using the absolute position of the Rover sensor on the Ground Station.

The aircraft successfully executed the test, being able to control its position while using the ENU vector as position measurement. Plots of the absolute measurements and the mapped ENU measurements during the two flight tests are shown in Figure 6.16.



**Figure 6.16** – Measurement Data From The ENU Vector During The Step Command Test

During the test the wind was fairly strong, with speeds of 5 m/s recorded. The disturbances due to wind can be seen in the plots. The data shows a correlation between the forward, back, left and right position steps that were performed. The vehicle never drifted considerably while under autopilot control, indicating that the ENU measurements were reliable. It is lastly worth noting that the ENU vector maintained an average update rate of 5 Hz, increasing the confidence in the quality of the dedicated DGPS communication link.

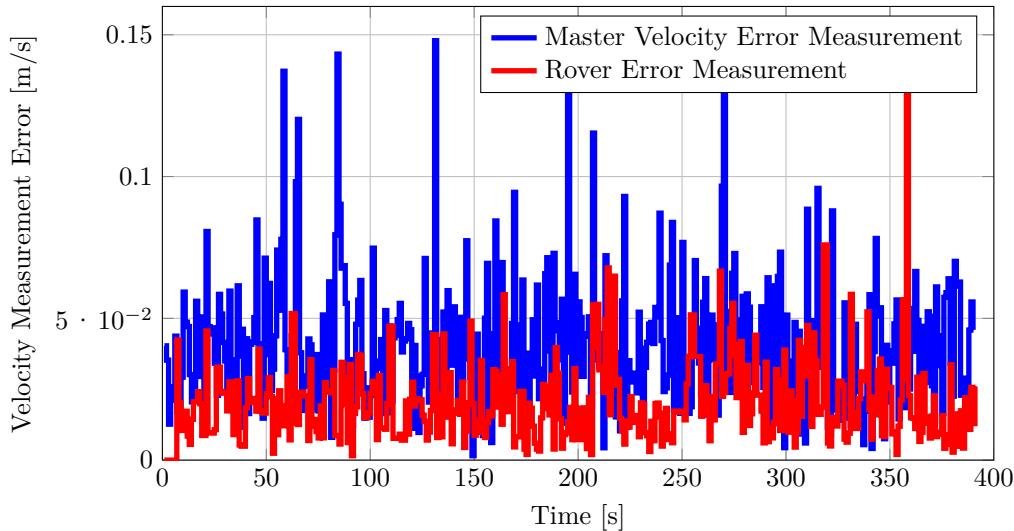
### 6.6.9 Velocity Measurement Error Test

The BestVel packet provides the GPS sensor's horizontal speed, vertical speed and heading with respect to true north. From past experience in the ESL, the BestVel packet has proven to be robust and reliable. The velocity measurement error must, however, be quantified, as a parameter in the re-designed horizontal position controller requires this parameter.

Recall from Chapter 5 that it was decided that integrators would be added to the position controller. These integrators will compensate for any errors in the velocity measurements of the vehicle and the platform. The appropriate choices for the saturation limits of these integrators can be determined from the test. In Section 5.2.1, where the control system design goals are established, it is explained that the steady-state position error, when the vehicle is tracking the moving platform, is a function of the velocity error measurements of both the vehicle and the platform. When integrators are added to the position controller, this steady-state error can be controlled to zero. The goal of these tests is to determine the maximum velocity error measurement that the position integrators must compensate for.

The solution is to quantify the velocity measurement error of the Master and Rover sensors, and then to design the limits of the integrator for a worst-case scenario. Novatel states that a velocity accuracy of  $0.03 \text{ m/s}$  can be expected from the BestVel packet [30]. However, tests were still performed to verify how well the BestVel velocity measurements agree with the specifications provided by Novatel.

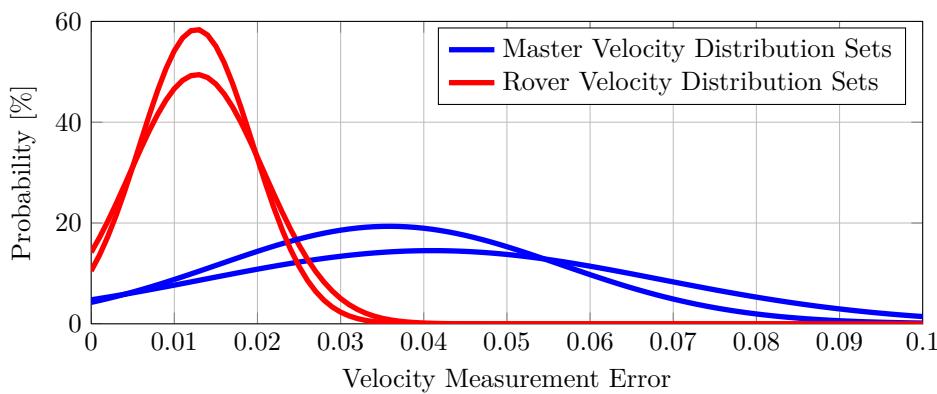
A convenient and simplistic test where a ground truth of the speed measurement can be known is a stationary test. It is known that the GPS sensor's speed should be zero and any non-zero speed measurement is not genuine. Consider Figure 6.17, which shows the horizontal speed measurements of the stationary Rover sensor and the stationary Master sensor over 400 seconds. The speed measurements were logged at a rate of  $10 \text{ Hz}$ .



**Figure 6.17 – Master And Rover Sensor Speed Measurement Error**

What is seen is that the Master's speed measurement errors are generally greater than the Rover's measurement errors. This is possibly due to the Rover sensor using a high performance Global Navigation Satellite System (GNSS) antenna, whereas the Master sensor is using a standard performance GNSS antenna. The sensor units are also different Novatel products, which could contribute to the difference in speed measurement errors.

It is assumed that the speed measurement errors are random and can be represented by a normal Gaussian distribution, as was done by Chalko [31]. Figure 6.18 shows the Master and Rover probability density functions for two separate data sets.



**Figure 6.18 – Velocity Measurement Error PDF**

In total, 5 data sets were recorded and analysed. Average characteristics were determined from the probability density functions of the 5 data sets of the Rover and the Master sensors. Refer to Table 6.5 for these characteristics and their corresponding values.

**Table 6.5 – Average Rover And Master PDF Characteristics**

Characteristic	Value [m/s]
Rover Average Mean Value [ $\mu$ ]	0.015
Master Average Mean Value [ $\mu$ ]	0.035
Rover Average Standard Deviation [ $\sigma$ ]	0.009
Master Average Standard Deviation [ $\sigma$ ]	0.021
Rover Maximum Error Expected [ $3\sigma + \mu$ ]	0.044
Master Maximum Error Expected [ $3\sigma + \mu$ ]	0.098

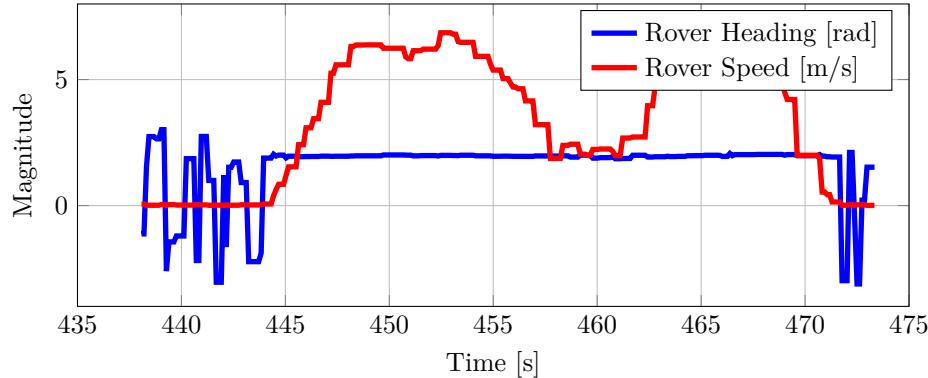
The worst-case scenario is therefore when both the Rover and Master velocity measurement errors are at their maximum. This is the value that must be compensated for by the position controller's integrators. The sum of the Rover and Master maximum expected speed measurement error is  $0.142 \text{ m/s}$ , and this value will therefore be used for the integrator limits.

The heading measurement obtained by BestVel of the Rover sensor, which will be mounted on the platform, should also be investigated. The heading measurement is used to convert the horizontal speed to inertial axes, thus providing the north and east feed-forward velocity references for the velocity controller. The heading measurement has a variable cross-track error and is a function of the horizontal speed of the sensor.

Novatel provides an equation for determining the heading error as a function of the speed of the Rover antenna [30]. The equation is derived under the assumption that the maximum speed measurement error is  $0.03 \text{ m/s}$ . The direction error function  $d(\text{speed})$  is then expressed as

$$d(\text{speed}) = \tan^{-1}(0.03/\text{speed}) \quad (6.6.1)$$

Thus, while the sensor is stationary, the error will be very large. As the horizontal speed increases, the direction error decreases. A test was conducted where the Rover sensor was fixed to the roof of an automobile. The automobile was then driven in a straight line. A segment of the results of the test is plotted in Figure 6.19.



**Figure 6.19 – GPS Heading Measurement Test Results**

The results show that the heading measurement is sporadic when the speed of the Rover antenna is near zero. As the speed increases, the heading measurement stabilises. It can be concluded from this test that the heading measurement of the Rover should only be used once the speed of the Rover is above at least  $1 \text{ m/s}$ . According to Equation 6.6.1 a speed of  $1 \text{ m/s}$  will translate to a heading measurement error of  $1.72^\circ$ .

The automated landing test will, however, not be performed at a speed of  $1 \text{ m/s}$ , but at speeds of up to  $8.3 \text{ m/s}$ . The expected directional error in the heading measurement of the platform will thus be less than  $1.72^\circ$ . As such, the error will be very small and is considered to not affect the control system of the aircraft noticeably.

### 6.6.10 Platform Velocity Send Rate Test

The rate at which the platform's velocity measurements are sent to the aircraft must also be determined. The faster the velocity measurement can be sent to the aircraft, the faster the velocity control loop will be able to respond to changes in the platform's speed. The telecommand and telemetry link is the medium through which this information is sent to the the aircraft.

However, the link already carries a considerable communications load and will be the limiting factor which determines the maximum attainable send rate. A test was devised to determine what send rate will be bearable for the telecommand and telemetry link. The Ground Station was modified to allow for a configurable send rate of the platform's velocity to the aircraft. Selectable send rates of  $1 \text{ Hz}$ ,  $2 \text{ Hz}$ ,  $5 \text{ Hz}$  and  $10 \text{ Hz}$  were available.

The test was repeated for each of the selectable send rates. A send rate of  $1 \text{ Hz}$  and  $2 \text{ Hz}$  did not affect the telemetry link at all, while very little packet loss was noticed if a send rate of  $5 \text{ Hz}$  was selected. Using a send rate of  $10 \text{ Hz}$  caused considerable packet loss and started affecting the communication speed of telemetry being sent to the Ground Station from the aircraft.

Given the results of the test, a decision was made to use a platform velocity send rate of  $5 \text{ Hz}$ . A send rate of  $5 \text{ Hz}$  is considered acceptable, as the platform's speed should remain near-constant throughout the automated landing test. If the automated landing test was to occur with large variation in the platform's speed, a send rate of  $5 \text{ Hz}$  would not have been acceptable. If considerable packet loss had been seen during practical flight tests, this rate could be lowered on demand to  $1 \text{ Hz}$  or  $2 \text{ Hz}$  from the Ground Station.

### 6.6.11 Magnetic Interference Tests

The vehicle is equipped with a magnetometer, which is used to measure the yaw angle of the vehicle with respect to true north. The metal trailer and platform will have an effect on the magnetic field near the landing location, thus disturbing normal operation of the magnetometer.

A test was performed in an attempt to quantify the disturbance due to the metal trailer on the magnetometer measurements. The test was conducted by placing the vehicle on the center of the platform and then driving around with the platform. This was the worst-case scenario, as this was the closest that the vehicle would be to the platform during the automated landing procedure. The body x-axis of the vehicle was aligned with the body x-axis of the trailer. The heading measured by the magnetometer on the vehicle was therefore expected to agree with the heading measured by the Rover sensor on the platform.

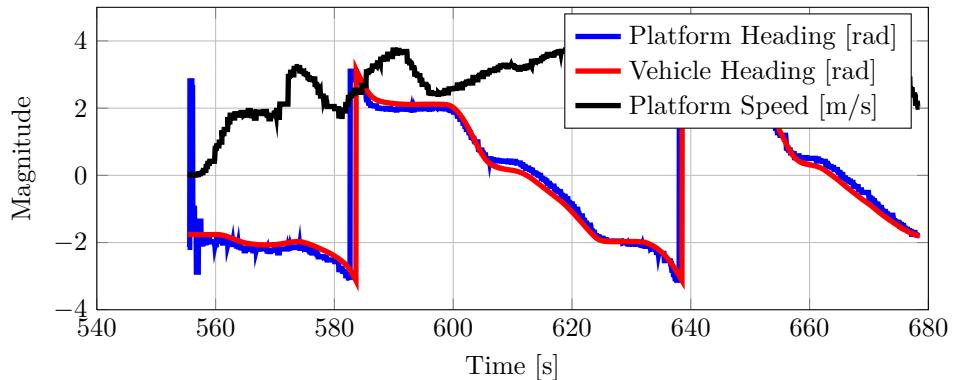
A control dataset was very difficult to obtain. The ideal test would be to hold the heading of the vehicle constant while the platform is brought into close proximity of the vehicle. Attempts to obtain a control dataset was unsuccessful. However, the test continued in an attempt to quantify the heading measurement difference between the heading of the aircraft

and the heading of the platform. In Figure 6.20 the trailer can be seen outfitted with the platform and instrumented with the Rover antenna. More information on assembly of the moving platform will be presented in Section 9.1. The vehicle was stationed on the platform.



**Figure 6.20 – The Vehicle Stationed on the Platform**

The trailer was towed around a parking lot to obtain data. The aircraft was not allowed to rotate or translate relative to the platform. The body x-axes of the vehicle and the trailer thus remained aligned throughout the test. The measured headings of the vehicle and the platform can be seen in Figure 6.21. The speed of the platform is also plotted to support the validity of the platform's GPS heading measurement.



**Figure 6.21 – Magnetometer Influence Test**

The results from this test show that there is a varying heading measurement difference of  $5^\circ$  to  $10^\circ$  between the platform and the vehicle. The error does not seem significant, but could still have an effect on the control systems on the aircraft.

The results from this test are thus inconclusive. In Chapter 9 a flight test was performed with one of its goals being to get a better idea of the trailer and platform's effect on the magnetometer.

### 6.6.12 Test Campaign Discussion

The test results confirmed that the ENU vector is reliable and that the obtained relative position measurements between the Master sensor and the Rover sensor are accurate. The ENU vector maintained its accuracy throughout all the tests conducted. The autopiloted flight test indicated that the ENU vector can be successfully utilized by the control system of the aircraft to control its position in space. The ENU vector's accuracy is deemed good enough for the task of automated landing on a translating platform, and the reliability is acceptable, given that there is sufficient communication between the Master sensor and the Rover sensor.

A communication failure between the Master sensor and Rover sensor can be catastrophic, as the ENU vector, MasterPos and RoverPos will then stop giving measurement updates. This consequently made the sensing strategy change. The ENU vector will still be used for the automated landing procedure, but MasterPos, which would have been used for the absolute position measurement of the aircraft, has been substituted by BestPos. BestPos does not require DGPS communication and will always be available. Fortunately, there is no gain or loss in accuracy due to the packet substitution.

The communication link between the Master sensor and Rover sensor has proven to function well in practice. The dedicated DGPS communication link proved to work at distances of up to 90 m, and the communication quality is sufficient to allow for updates of the ENU vector at an average rate of 5 Hz. The ENU vector's update frequency is considered acceptable for the goal of the project, but the update frequency can be improved by propagating the ENU vector between updates. The DGPS communication range is also deemed acceptable, as the Master sensor and the Rover sensor will be closer to each other than 90 m during the automated landing test.

The velocity measurement error in the velocity measurement of the Master sensor and the Rover sensor has also been quantified, and can now be compensated for by the integrator in the re-designed horizontal position controller. The rate at which the platform's velocity information is sent to the aircraft has also been determined. The telecommand and telemetry link between the Ground Station and the aircraft has enough head-room in its bandwidth to deliver velocity measurement of the platform to the aircraft at a rate of 5 Hz. The platform's velocity update rate is deemed acceptable, as excessive variation in the platform's velocity is not expected.

A contingency plan must, however, be implemented for the scenario of a DGPS communication failure. The plan is designed and implemented in the next section. Propagation of the ENU vector is also implemented in the next section, in order to artificially increase the update rate of the vector.

## 6.7 Sensing Safety and Propagation

### 6.7.1 Sensing Safety

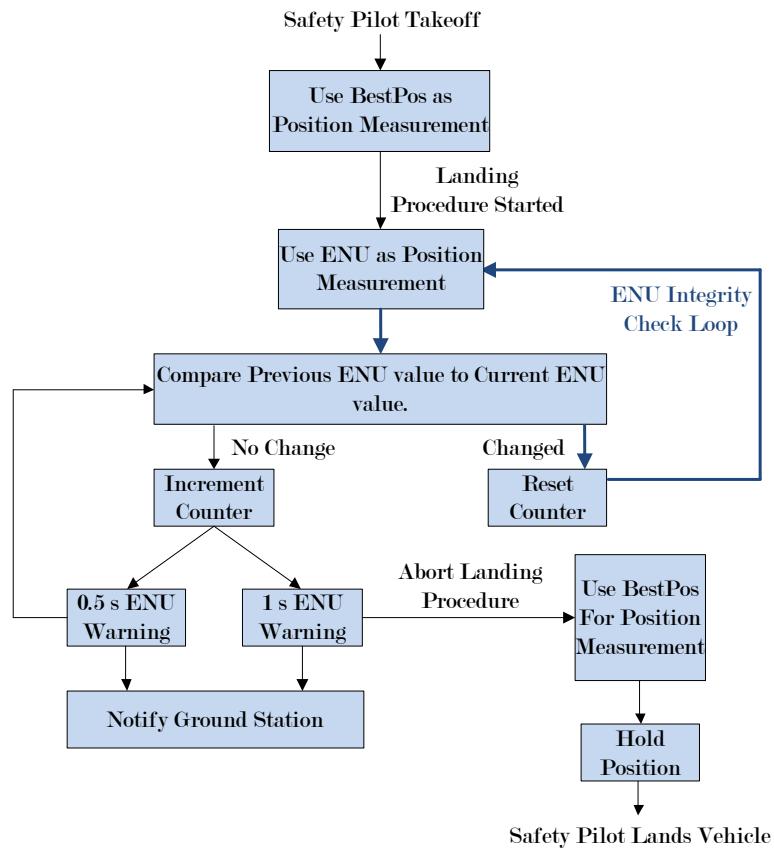
The results of the DGPS communication failure test showed that the outcome of such a failure will be fatal to the aircraft. The only DGPS packet that will be used is the ENU vector, and from the DGPS communication failure test it is known that the ENU vector will immediately become stagnant if DGPS communication has failed. A mechanism should therefore be implemented to detect a DGPS communication failure, and to take appropriate action to accommodate the failure. It is firstly taken into consideration how to detect a DGPS communication failure, and then the strategy that is used to accommodate the failure is presented.

A logical solution is to monitor the elapsed time between ENU vector updates. A counter was implemented in the main loop of the OBC. The main loop is executed at 50 Hz, allowing

the counter to be incremented 50 times a second. The counter is incremented every time the ENU vector has not been updated from the previous execution of the main loop. From this counter, it can be determined for how long the ENU vector has not been updated.

At the same time, the sample rate of the ENU vector is calculated. This is done by monitoring how many times the ENU vector has been updated over a one-second window. The sample rate is continuously sent to the Ground Station to monitor DGPS health and functionality.

A quantitative bound was chosen next, to determine when the ENU vector has become unsafe to use. It was decided that an update rate of less than 1 Hz is unacceptable. Therefore, if more than a second has elapsed since the last ENU vector update, the vector is declared unfit for use. A warning is also sent to the Ground Station if the ENU vector has not changed its value for 0.5 seconds. This warning, along with the knowledge of the sample rate, will give a sense of the health of the DGPS functionality. Figure 6.22 shows a block diagram of the proposed safety mechanism that will determine whether a DGPS communication failure has occurred.



**Figure 6.22 – DGPS Communication Failure Detection and Safety Mechanism**

Consider the proposed flight demonstration in Section 3.3 and the state machine in Figure 6.22. The flight demonstration will start with the safety pilot taking off to an appropriate height. The autopilot is then engaged by the Ground Station and the aircraft will be commanded to hold its position. While holding position, BestPos will be used as position

measurement. As the aircraft hovers, the trailer and platform start moving, and as soon as the platform has reached the desired speed, the automated landing procedure is started by a command from the Ground Station.

From this point until the end of the landing procedure, the ENU vector is used as position measurement. The state machine in Figure 6.22 becomes active and starts performing the ENU integrity check. As mentioned, the time elapsed between ENU vector updates is quantified by a counter in the main loop of the OBC. Every time the main loop is executed, a check is performed to see whether the ENU vector value has changed from the previous execution. If the ENU value has changed, the counter is reset.

However, if the value of the ENU vector from the previous main loop execution is the same as the ENU value during the current execution, the counter is incremented. If the counter has been incremented 25 times, a warning is sent to the Ground Station to indicate that the ENU vector has not been updated for 0.5 seconds. The ENU integrity loop continues comparing the ENU value between main loop executions.

If the vector continues to remain the same, the counter will eventually reach a value of 50. This indicates that the ENU vector has not been updated for one second. An update rate of less than 1 Hz has therefore been achieved and the ENU vector is declared unfit for use. This is the product of a DGPS communication failure.

The Ground Station is notified and the aircraft automatically aborts the landing procedure. As the ENU vector can no longer be used as position measurement, the aircraft reverts to BestPos for position information. Recall that BestPos will always be available, and is a synchronous packet that has a fixed sampling rate of 10 Hz.

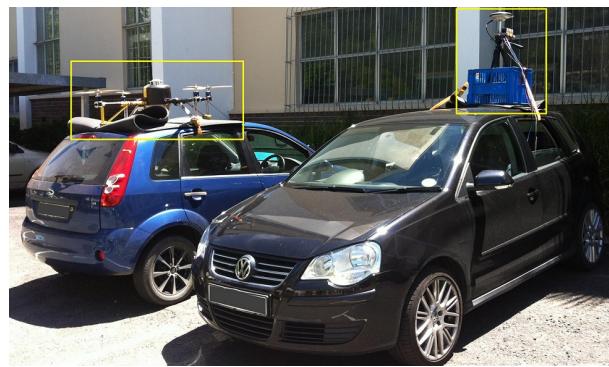
The aircraft will automatically come to a hover when the landing procedure is aborted. The safety pilot will retake control of the aircraft and then land it.

### 6.7.2 Sensor Propagation

The ENU vector is an asynchronous packet with a sample rate that varies between 0 Hz and 10 Hz. To provide relative position estimates at a higher frequency, the ENU vector is propagated between measurement updates.

Propagation is done by performing numerical integration of the difference between the velocity measurement of the Rover and the Master sensors. Numerical integration is performed on the OBC of the vehicle at a rate of 50 Hz. The vehicle receives measurement updates of its own velocity at 10 Hz, which is the maximum update rate that the Master sensor can provide. The Rover sensor also provides velocity updates at a rate of 10 Hz, but the vehicle only receives the Rover's velocity at a rate of 5 Hz. This is due to the saturated telemetry link, as was seen from the test performed in Section 6.6.10.

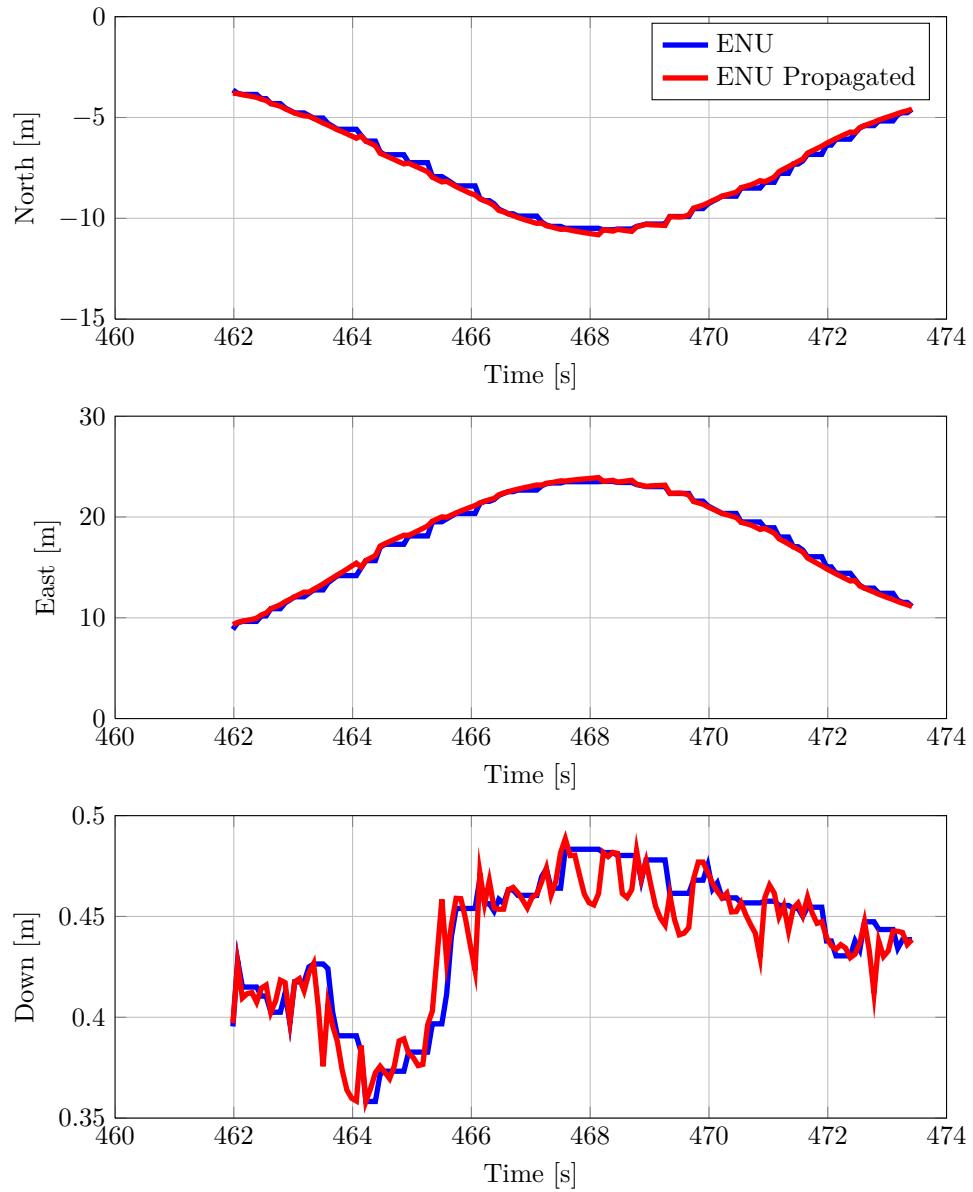
To verify the functionality of the ENU vector propagation, a practical test was performed. Figure 6.23 shows the test setup.



**Figure 6.23 – ENU Vector Propagation Test Setup**

The Master sensor and the Rover sensor were fixed to the roof of two different automobiles. The Master sensor, located on the aircraft, can be seen on the automobile on the left and the Rover sensor, which will be located on the platform, on the automobile on the right.

A segment of the results from the test is shown in Figure 6.24. The original ENU vector and the propagated ENU vector are plotted.



**Figure 6.24** – ENU Vector And Propagated ENU Vector

Propagation in the north and east directions yields an accurate higher-frequency update rate over the un-propagated ENU measurement. However, propagation in the Up direction appears to yield a notably degraded accuracy during propagation. This is, however, not the case, as the change in vertical position was small during the test.

It was, however, decided not to propagate the Up component of the ENU vector. The Up component of the ENU vector will remain constant for all practical purposes during the landing procedure, as the vehicle will track the platform from a constant height. A higher-frequency update rate of this measurement is therefore not critical.

Only during the last stage, which is the descending stage, will the Up component change notably. When the landing procedure is in this stage, the altitude controller will be disarmed. The ENU Up measurement is thus not used during this stage.

### 6.7.3 Sensing Safety and Propagation Discussion

This section has introduced a safety mechanism for a DGPS communication failure and an algorithm that provides high-frequency ENU vector updates. The safety mechanism will ensure the safety of the aircraft during a DGPS communication failure. A state machine was designed which consistently monitors the ENU vector, so that the aircraft knows when a DGPS communication failure has occurred. When the state machine has determined that the ENU vector is unfit for use, the aircraft will abort the automated landing procedure, and it will automatically come to a hover.

The strategy will require extensive Hardware-In-the-Loop simulation to ensure that the proposed state machine of the safety mechanism functions as intended. A test flight will also have to be conducted, to verify the response of the aircraft when the landing procedure is aborted. The results of a test flight which simulates an abort situation is presented in Section 9.4.

Propagation of the ENU vector is also implemented between updates of the vector. As the vector is only updated at an average rate of 5 Hz, propagation is done between updates to provide a higher frequency update rate. Propagation is performed by numerical integration of the difference between the velocity measurement of the Master sensor and the Rover sensor. A test was conducted to verify whether propagation of the ENU vector does indeed benefit the relative position measurements. It was found that propagating the vector in the east and north directions yielded an accurate high-frequency representation of the actual vector, while propagation in the Up direction degraded the accuracy of the vector. Propagation is therefore only implemented on the east and north components of the ENU vector.

## Chapter 7

# Automated Landing State Machine

This chapter will elaborate on the state machine that was designed to guide the aircraft through the autolanding procedure. The state machine ensures the safety of the aircraft and of its surroundings, and improves the probability of a successful autonomous landing during the autolanding procedure.

The chapter starts by providing a brief overview of the proposed autolanding state machine that will be used during the Landing phase of a typical flight mission. Recall from Chapter 1 that a typical flight mission consists of four consecutive phases, namely Take-Off, Decoy, Return and Landing.

A more detailed presentation is then given on each state during the Landing phase, explaining the actions that will be performed and the motivation for the actions in that state.

What follows is a section that explains differences between the proposed autolanding state machine used for typical flight missions and the autolanding state machine that will be used for the flight demonstration of the project. The chapter closes by presenting changes that were made to the Ground Station to monitor and configure the autolanding state machine.

### 7.1 Proposed Autolanding State Machine

The Return algorithms will guide the aircraft to be finally located behind the landing platform after it has performed a Decoy. At this point in the flight mission the Landing phase starts, and the autolanding state machine becomes active.

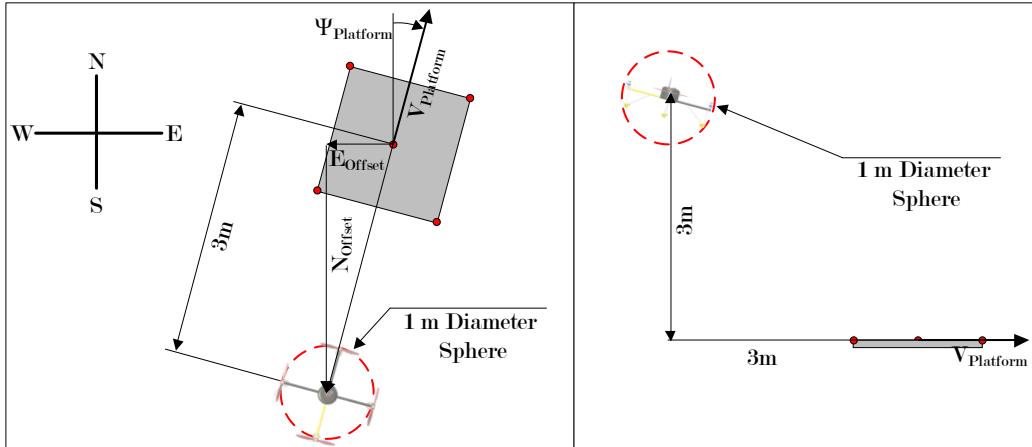
As soon as the autolanding state machine becomes active, the vehicle switches to using the East-North-Up (ENU) vector for position feedback instead of BestPos, which is used throughout the other phases of the flight mission. The ENU vector is then directly used as the error for the position controllers. The proposed autolanding state machine has four states, which are appropriately named:

1. Tracking
2. Homing
3. Descending
4. Shutdown

The states are designed to guide the aircraft safely through the autolanding procedure. A series of safety checks is performed to ensure that the aircraft is in a steady-state, and that no transients are present while critical landing actions are performed. A detailed presentation of each of the four states follows, starting with the Tracking state.

### 7.1.1 Tracking State

During the Tracking state the vehicle accelerates and catches up with the landing platform, and tracks a position 3 m behind and 3 m above the landing location. This is a dynamic offset in the inertial axis system, and is calculated by using the heading  $\Psi_{Platform}$  of the platform, as shown in Figure 7.1.



**Figure 7.1 – The Error Bounds During the Tracking State**

The tracking location is chosen 3 m above and 3 m behind the landing location, so that any transients due to the vehicle decelerating to match the velocity of the platform are allowed to settle at a safe distance from the platform. As soon as the vehicle has entered a virtual sphere surrounding the tracking location, a counter is started.

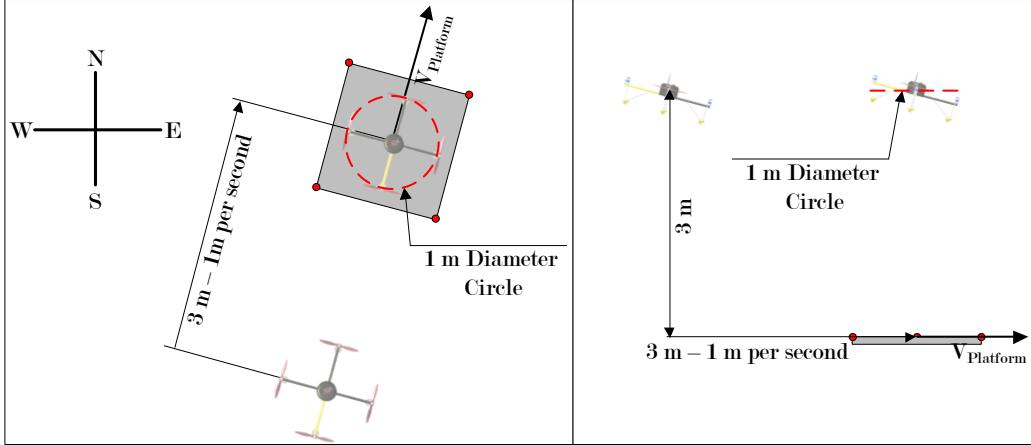
Given the physical size of the aircraft and the bandwidth of its position control loops, the diameter of this sphere is chosen to be 1 m. This diameter should allow enough freedom for the position control loops to reject disturbances and movement within this sphere will not compromise the landing accuracy of the procedure.

When the transients from the deceleration have settled, then the vehicle should remain in the sphere. However, other factors may drive the vehicle out of the sphere, such as external disturbances due to wind, or if the platform speed changes faster than the bandwidth of the vehicle. If the vehicle exits the sphere at any time, the counter is zeroed. The counter is only incremented again once the vehicle re-enters the sphere. When the counter has been incremented for a continuous time frame of 3 seconds, the state machine advances to the Homing state.

### 7.1.2 Homing State

During the Homing state, the vehicle homes in and tracks a position 3 m directly above the landing location. The Homing state is illustrated in Figure 7.2.

To avoid inducing transients, the horizontal position reference is slowly moved to match the landing location's horizontal position. A speed of 1 m/s is chosen to home in. This is done by ramping the north and east offsets to zero with time. Slowly moving the reference also has another advantage: Less control effort is required for tracking, allowing more control effort to be utilized for combating disturbances.



**Figure 7.2 – The Homing Action and the Error Bounds During the Homing State**

As soon as the north and east offsets have become zero, two actions are taken. A second counter is started and the position controller's integrators are enabled. The counter is started to check that the vehicle remains in a horizontal circle with a diameter of 1 m. The vehicle has to stay in the circle for a continuous duration of 3 seconds. If the vehicle exits the circle before the counter has reached 3 seconds, the counter is zeroed and only restarted upon re-entry of the circle. This is again done to perform a logical check to ensure that the vehicle is in a steady-state condition, and not moving dramatically relative to the landing location.

Two reasons lie behind the motivation to enable the position integrators at this time. The first is that any unnecessary integrator wind-up is avoided, as the vehicle will already be very close to the tracking location when the integrators are enabled. The second is that very precise positional control is required from this state to touchdown. If there is an error present in the feed forward velocity measurement, then the integrators will compensate to minimize the position error.

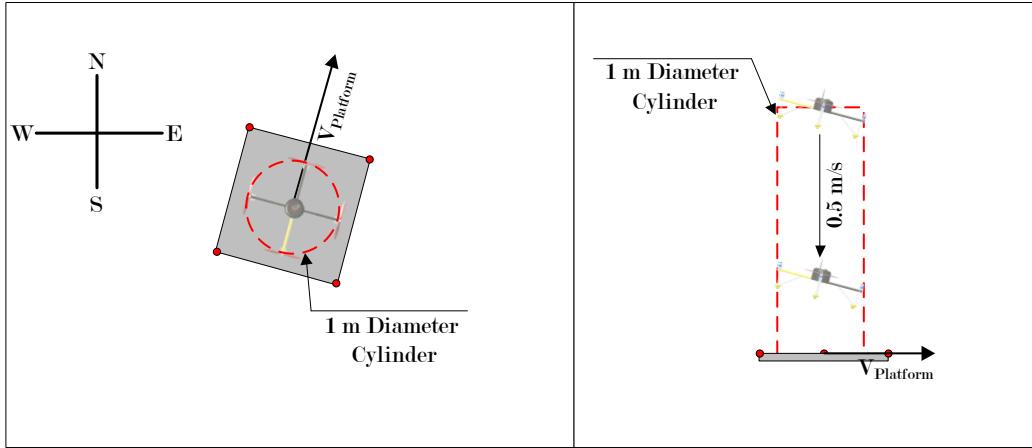
When the counter has been incremented to a value which corresponds to 3 seconds, the Descending state will become active.

### 7.1.3 Descending State

The Descending state is the crucial state where the vehicle will make contact with the translating platform. The vehicle descends at a constant rate to the platform while maintaining its horizontal position relative to the moving platform. The checks performed in the Tracking and the Homing states were done to ensure that the vehicle is now in a steady condition, so that the probability of missing the landing location while descending is decreased. Refer to Figure 7.3 for a graphical representation of the Descending state.

The moment the Descending state is entered, the altitude controller is disarmed. The vehicle is commanded to descend at a constant rate of 0.5 m/s. This ensures that the vehicle will eventually make contact with the platform, if it remains within the error bounds. While descending onto the platform the vehicle has to remain within a 1 m diameter circle, which can geometrically be seen as a cylinder. The cylinder allows for a maximum absolute horizontal error of 0.5 m.

If the aircraft remains within the cylinder throughout the Descending state, it will make contact with the landing platform. However, if the aircraft drifts outside of the cylinder



**Figure 7.3 – The Error Bounds During the Descending State**

while descending, the state machine will revert to the Homing state. Once the aircraft has again passed the error criteria for the Homing state, it can attempt to descend again.

#### 7.1.4 Shutdown State

The Shutdown state is entered when an accelerometer spike is measured as the vehicle makes contact with the platform. From previous work on stationary automated take-off and landing, a vertical accelerometer spike of less than  $-15 \text{ m/s}^2$  was measured when descending at a rate of  $0.5 \text{ m/s}$ . This rate is fast enough that the vehicle can break through the ground effect caused by the rotors, but not too fast to cause damage due to impact. However, the ground effect is less of a concern for this project, as the platform that the vehicle lands on has a perforated surface. The perforated platform on the trailer can be seen in Section 9.6. When a specific force spike of less than  $-15 \text{ m/s}^2$  is measured, the vehicle will automatically disarm its entire flight control system. Consequently, the rotors will also stop spinning. The autolanding procedure is then completed and it is safe for personnel to approach the aircraft.

## 7.2 Autolanding State Machine For Flight Demonstration

The practical flight demonstration test that will be done will not use exactly the same autolanding state machine as the proposed autolanding state machine. The reason is that the test will attempt to isolate the landing phase of the flight mission, so that no other phases of the mission have an influence on the results of the test. The Take-Off, Decoy and Return phases will therefore not be incorporated into the flight demonstration test for the project. The flight demonstration test requires some modifications to the autolanding state machine, making the test a special case of the proposed autolanding procedure. The modification required specifically for the practical flight demonstration test is documented in this section.

The autolanding state machine that will be used to demonstrate the system's capability has one emergency state and five normal states. The five normal states are appropriately called:

1. Standby
2. Tracking
3. Homing
4. Descending
5. Shutdown

During all five normal states, the re-designed horizontal position and horizontal velocity controllers will be used, but in the emergency state the flight control will revert to the pre-existing, less aggressive controllers. BestPos is used as position measurement during the Standby state and the emergency state, whereas the ENU vector will be used during the four other normal states for accurate relative position measurements.

The operation of this special case of the autolanding state machine will be explained in detail with reference to Figure 7.4, which shows the states and flow of the state machine.

Before an in depth discussion of each state, the emergency stop feature is elaborated upon. The emergency stop feature is implemented as a safety measure for two possible scenarios. The one scenario is the loss of the DGPS relative position measurements, possibly due to failure or degradation of the wireless communications link between the Master and the Rover sensors. The other scenario is if any unexpected behavior is observed during the autolanding procedure. The emergency stop can then be signaled by the officer working the Ground Station.

### 7.2.1 Emergency Stop Feature

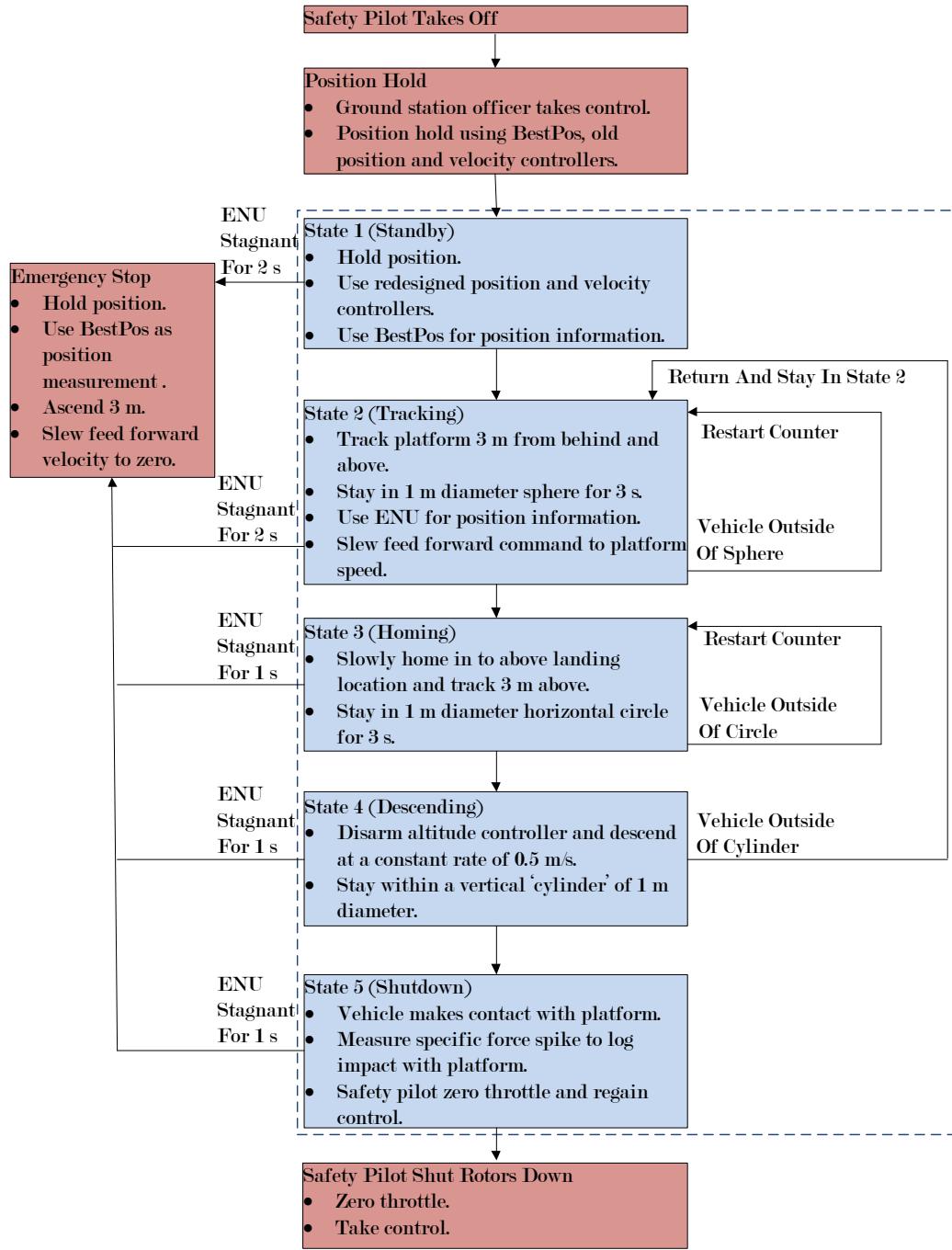
During an emergency stop, the vehicle reverts to normal BestPos GPS position feedback, ascends 3 m, and slews to a standstill, where it hovers and maintains its position, waiting for the safety pilot to take over.

As mentioned, a scenario which requires an emergency stop is the scenario of a DGPS communication failure. The practical sensor tests in Chapter 6 showed that the ENU vector can become stagnant if DGPS communication between the Master sensor and the Rover sensor is disrupted during operation. The algorithm that determines whether this has occurred is also explained in Chapter 6.

When the algorithm has detected that the ENU vector has become unsafe to use, the emergency stop will be triggered. When the emergency stop is signaled, either by the algorithm monitoring the ENU vector's health or by the Ground Station Officer, the following events will occur:

- The vehicle will revert to BestPos for position measurement
- The position and velocity controllers will revert to the pre-existing, less aggressive controllers
- The feed forward velocity command will be zeroed
- The vehicle will immediately attempt to hold its horizontal position
- The vehicle will be commanded to immediately ascend 3 m

The safety pilot can then take over manual control of the vehicle and land. The reason for reverting to BestPos as position measurement is that it is a stand-alone GPS packet which is more reliable than a communication-dependent DGPS packet.



**Figure 7.4 – Diagrammatic Representation of the Sequence of Events During the Flight Demonstration Test**

### 7.2.2 Standby State

The Standby state is used only for the practical flight demonstration test, and effectively replaces the Return phase of a typical flight mission. A scenario will therefore need to be artificially created, which puts the aircraft in the air behind the already moving platform.

To create such a scenario, the landing procedure will commence by the safety pilot lifting off and aligning the heading of the vehicle approximately with the heading that the translating platform will maintain during the course of the landing. When the safety pilot has lifted off to a safe height, the Standby state will be activated from the Ground Station, and the aircraft will hold its position.

When the Standby state is active, the vehicle starts using the re-designed horizontal position and horizontal velocity controllers, and uses BestPos for position measurement. In this state, the vehicle will hold its position regardless of what the position or speed of the translating platform is.

The translating platform will then start to accelerate away from the vehicle. As soon as the platform's speed exceeds  $1 \text{ m/s}$ , the vehicle will start to track the heading of the platform while still remaining stationary. The platform must be moving at least  $1 \text{ m/s}$  in order to trust the heading measurement of the Rover sensor, as was experimentally determined in Chapter 6. When the platform has reached the desired speed at which the autolanding procedure will take place, the Ground Station Officer will issue the command to start the autolanding procedure. The state machine will then advance to the Tracking state.

A scenario will then effectively have been created, which is equivalent to a typical flight mission, where the Return algorithms have rendezvoused the aircraft with the ship.

Caution must be used during this state, as the distance between the platform and the vehicle should not become too large. The communication link used for DGPS communication might be disrupted if the baseline distance between the vehicle and the platform exceeds  $80 \text{ m}$ . If communication is disrupted, the vehicle will automatically perform an emergency stop as detailed in Section 7.2.1.

### 7.2.3 Tracking State

During the Standby state, a considerable distance can arise between the hovering aircraft and the accelerating platform. This can cause aggressive acceleration from the aircraft during the Tracking state, in an attempt to catch up to the already translating platform. If this acceleration is too aggressive, the actuators might saturate. A way of limiting this acceleration must be implemented for the demonstration test. This is achieved by limiting the velocity command from the position controller and slewing the feed forward velocity command.

Slewing the velocity feed forward command is not necessary during a typical flight mission, as the Return algorithm will have guided the aircraft to rendezvous with the ship while maintaining some velocity in the same direction as the ship. Less acceleration will therefore be required from the aircraft to catch up to the landing platform. During the flight demonstration test, the aircraft will not have an initial speed and will be stationary, requiring aggressive acceleration to catch up to the translating platform.

The horizontal velocity that the vehicle must attempt to maintain is computed from two terms. The one term is the velocity command given to the horizontal velocity controller by the horizontal position controller. The second term is the feed forward velocity command, also given to the horizontal velocity controller. These two terms should not vary too drastically, as they will demand aggressive acceleration from the vehicle. The vehicle's acceleration is limited to  $4 \text{ m/s}^2$ , which in effect translates to a maximum tilt angle of approximately  $22^\circ$ .

Constantly maintaining the maximum acceleration is not desired, as strenuous actuation is required. The time that the vehicle spends accelerating at  $4 \text{ m/s}^2$  should therefore be kept to a minimum. A decision was made to implement a slewed feed forward velocity limit.

The velocity reference command generated by the position controller is hard-limited to  $3.5 \text{ m/s}$  for large position errors. Given the proportional gain of the velocity controller of 1.4, the maximum acceleration command contributed by the position controller will be  $4.9 \text{ m/s}^2$ . This command is higher than the  $4 \text{ m/s}^2$  acceleration limit of the acceleration command generated by the velocity controller. It is therefore decided that the feed forward term should not be varied too fast and will consequently not be allowed vary by more than  $1 \text{ m/s}$  per second. This will, in effect command an acceleration of  $5.9 \text{ m/s}^2$  under the most demanding circumstances. This will drive the pitch and roll angles to their limits, but not continuously. The slew limit will function for both increases and decreases in the feed forward velocity command. A practical flight test was performed in Chapter 9 which tested the slew rate of the feed forward velocity command.

#### 7.2.4 Descending State

Exceeding the admissible position error during the descending state could lead to a catastrophic failure, due to one of the feet of the vehicle missing the platform. It is therefore decided that if the vehicle fails the error criteria for the flight demonstration test, it will revert to the Tracking state and not be allowed to attempt another landing.

For this special case of the autolanding procedure, when the aircraft exceeds the admissible error while descending, the platform should be brought to a smooth stop. The vehicle will consequently also stop in mid-air,  $3 \text{ m}$  behind and  $3 \text{ m}$  above the platform. Control will then be given to the safety pilot by the Ground Station Officer and the safety pilot will land the vehicle.

During a normal flight mission, in the event of such a scenario, the autolanding state machine will revert to the Homing state and the aircraft will be allowed to attempt another descent, as a safety pilot may not always be present during a typical flight mission.

#### 7.2.5 Shutdown State

During a typical flight mission, the aircraft will shut its rotors down automatically when it measures a specific force spike of less than  $-15 \text{ m/s}^2$ . However, upon touchdown for the flight demonstration test, the safety pilot will zero the throttle on the RC remote and take control. The vehicle rotors will then stop rotating. The responsibility of shutting down the rotors is given to the safety pilot for the sake of safety in testing the autolanding concept.

The Shutdown state will, however, still be entered upon measurement of a large enough vertical specific force spike. The state is thus a dormant state, purely indicating that the vehicle knows that it has touched down on the platform.

For the flight demonstration test, when the Shutdown state is entered, the vehicle will continue to attempt to descend at  $0.5 \text{ m/s}$ . This will make the rotors gradually rotate at a lower rotation speed until the integrators in the various control loops start to wind up. The integrators in the horizontal velocity and angular rate control loops will start to wind up due to unmodelled dynamics as a result of contact with the platform, which could eventually cause the vehicle to turn over. However, simulations show that this will only happen several seconds after the vehicle has touched down.

The safety pilot will be commanded to instantly zero the throttle and take control of the vehicle as soon as it has made contact with the platform. Integrator wind-up while the vehicle is on the platform is thus not expected to be a problem. To prove that the vehicle could shut itself down in future, the accelerometer spike is logged and a notification is logged on the vehicle that the Shutdown state has become active.

### 7.3 State Machine User Interface

The Ground Station Application was modified by the addition of a user interface that aids in managing and monitoring the autoland state machine. The user interface made it easier to understand the status of the autoland test and to control the aircraft in the rush of the moment. The Ground Station Application was augmented with a single tab which displays relevant telemetry of the state machine and allows the Ground Station Officer to send configuration parameters to the state machine.

All state machine parameters can be adjusted from the Ground Station by using the added user interface. The functionality was implemented to be able to modify parameters on the day of a flight test, without requiring to reprogram the vehicle. The state machine indicators and adjustable parameters can be seen in Figure 7.5.



Figure 7.5 – The State Machine Status Indicators and Adjustable Parameters

The current state of the state machine is indicated by the series of diagrams at the bottom of the highlighted block. The diagram that represents the current state of the state machine is highlighted in green. The Control Loops box indicates which control loops of the flight control system are armed, and the General Parameters box allows live uploading of the state machine parameters.

The adjustable state machine parameters include the amount of time that the error bounds must be met for the Tracking and Homing states, as well as the diameters of the sphere used in the Tracking state and the cylinder used in the Descending state. The descent rate can also be modified and uploaded before a test.

The offset of the landing location relative to the GPS antenna mounted on the platform can be uploaded at any time using the Platform Antenna Offset block, allowing for the location of the antenna on the platform to be changed on the day of the flight test. The emergency stop button is highlighted in red, with the button that initiates the autolanding process just above it.

This concludes the in-depth overview of the state machine and the reasoning behind each chosen state. The state machine will be thoroughly tested with Hardware-In-the-Loop (HIL) simulation in Chapter 8 and practical flight tests in Chapter 9.

## Chapter 8

# HIL Simulation

The focus of this chapter is on the Hardware-In-the-Loop (HIL) simulations that were performed to prepare for the actual practical flight tests. HIL simulation is performed to ensure that the Ground Station and the vehicle communicate as intended, that the On-Board Computer (OBC) interprets sensor data correctly, and that the OBC code is executed as expected.

The chapter will begin by presenting an overview of the pre-existing HIL setup. The pre-existing HIL setup did not have the capability to simulate an autoland on a translating platform and consequently required modification. Modifications done to the setup will then be presented, followed by the establishment of the conditions in which the HIL simulations are performed. The chapter closes by presenting results from the HIL simulations performed during the project.

### 8.1 Pre-Existing HIL Setup

A diagrammatic representation of the HIL setup used in the Electronic Systems Laboratory (ESL) is shown in Figure 8.1.

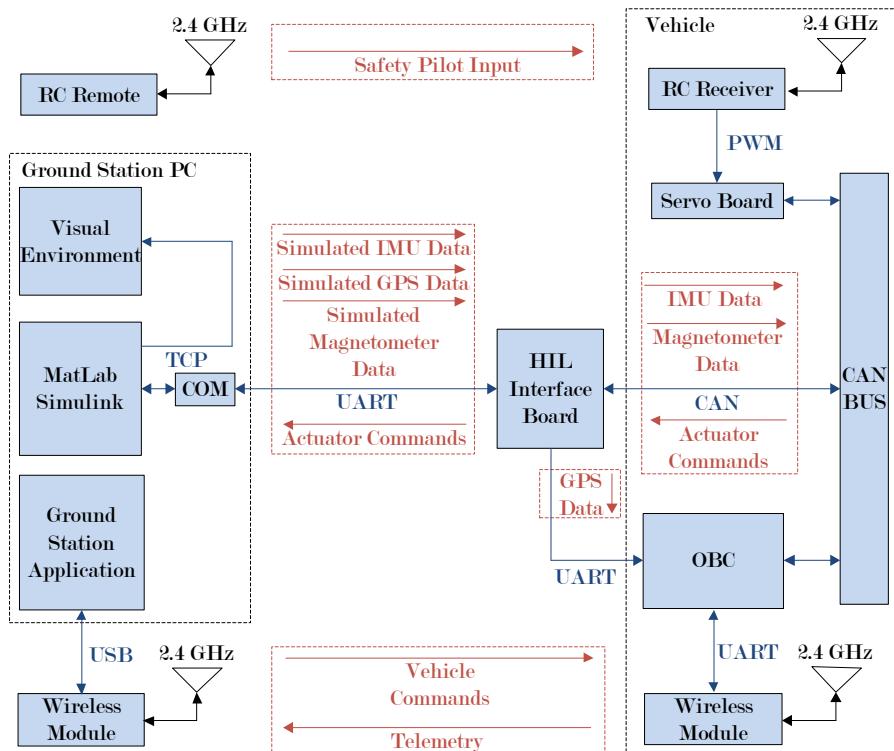


Figure 8.1 – Pre-Existing HIL Setup

The HIL setup consists of the Ground Station Personal Computer (PC), the Ground Station telecommand and telemetry wireless link, the Remote Control (RC) transmitter, the HIL interface board and the vehicle.

The HIL interface board is at the core of the HIL setup. It is the communication medium between the Matlab Simulink simulation environment and the hardware of the vehicle. The Inertial Measurement Unit (IMU), magnetometer and Global Positioning System (GPS) measurements are generated in Simulink and sent to the vehicle. These measurements are divided into two different data paths by the HIL interface board before being sent to the avionics of the vehicle.

The IMU and magnetometer measurements are sent to the Controller Area Network (CAN) bus, which connects most of the avionics on the vehicle. The GPS measurements are sent to the OBC via the Universal Asynchronous Receiver and Transmitter (UART), which connects to the Novatel sensor on board the vehicle. The actuator commands are sent from the OBC over the CAN bus to the Simulink environment to actuate the vehicle model in simulation.

During HIL simulation the telecommand and telemetry wireless link and the RC transmitter communicate with the vehicle as intended during actual flight. Telemetry data is sent from the OBC to the PC, and is then displayed on the Ground Station Application (GSA). The GSA in turn sends commands and important parameters to the OBC.

The pre-existing HIL setup was capable of fabricating Novatel Real-Time-Kinematic (RTK) packets. These packets are created in Simulink in the same binary format provided by a Novatel sensor. To simulate the new positional sensing method, discussed in Chapter 6, the HIL setup must be augmented.

The Align<sup>TM</sup> East-North-Up (ENU) vector must be generated in Simulink. To generate the ENU vector, positional information of the vehicle and the platform is required. Vehicle position information is generated in Simulink, but platform position generation does not exist in the pre-existing HIL setup. Fabricating platform position measurement requires software additions.

## 8.2 HIL Setup Modifications And Additions

A program was written solely to generate position and velocity data for the platform. This program was called the Virtual Platform Application (VPA) and was written in QtCreator, the same program that the GSA was created in. The VPA was programmed with the idea that it may be used for HIL purposes, but also for practical virtual platform landings. A practical virtual platform landing was performed using the VPA and will be discussed in Chapter 9.

The VPA utilises a thread of the Central Processing Unit of the Ground Station PC. This is done so that the VPA executes platform movement as real-time as possible. QtCreator has access to native Windows functions, which made it simple to accurately time the execution of the virtual platform's movement. The Windows *Sleep*<sup>1</sup> function was used to execute platform propagation at a fixed sample time. In Chapter 6 it was seen that the ENU vector can be sampled at a average rate of 5 Hz. The VPA was therefore written to also provide platform position measurements at a rate of 5 Hz.

The VPA was connected to both the GSA and the Simulink environment by use of Transmission Control Protocol (TCP). The local host Internet Protocol (IP) address on the Ground Station PC was employing two different ports. The final HIL communications structure is shown in Figure 8.2.

---

<sup>1</sup>The native *Sleep* function accepts an argument in milliseconds, e.g. *Sleep(200)*. Windows will pause a thread for the given time in milliseconds, then continue execution.

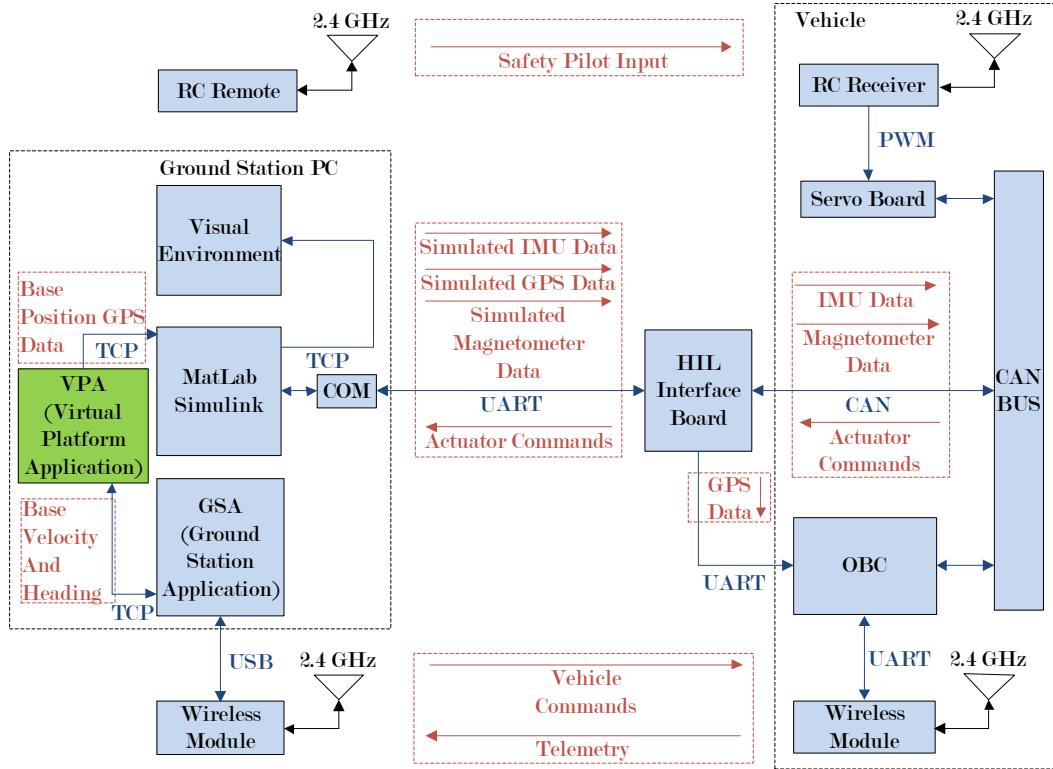


Figure 8.2 – Final HIL Setup

The VPA connects to Simulink to fabricate latitude, longitude and altitude measurements for the Rover sensor. By using the latitude, longitude and altitude of the vehicle and platform, the ENU vector could be fabricated in Simulink and sent to the vehicle in Novatel's binary packet form. The binary structure of the ENU packet can be seen in Appendix A.

The VPA connects to the GSA for two reasons, one being so that the virtual platform parameters can be adjusted from the GSA during HIL simulation and practical virtual platform landings. The other reason is so that velocity and heading information of the platform can be sent to the vehicle over the telecommand and telemetry wireless link, as it would during an actual test. Figure 8.3 shows the VPA enclosed in the red rectangle and the tab of the GSA that configures the VPA parameters enclosed in the yellow rectangle.

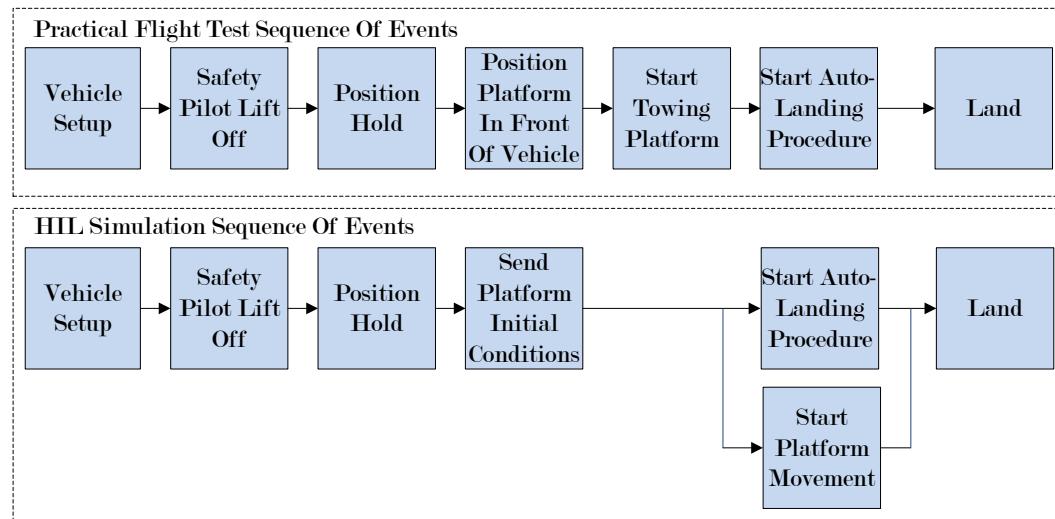


Figure 8.3 – Ground Station Application and Virtual Platform Application

From the GSA, with the aircraft airborne in the HIL simulation, the initial virtual platform parameters can be specified. These parameters are the distance at which the platform will start in front of the vehicle, the platform velocity, the platform height and the feed forward velocity error. The heading of the platform is the same as the heading of the vehicle at the time of issuing the initial platform parameters.

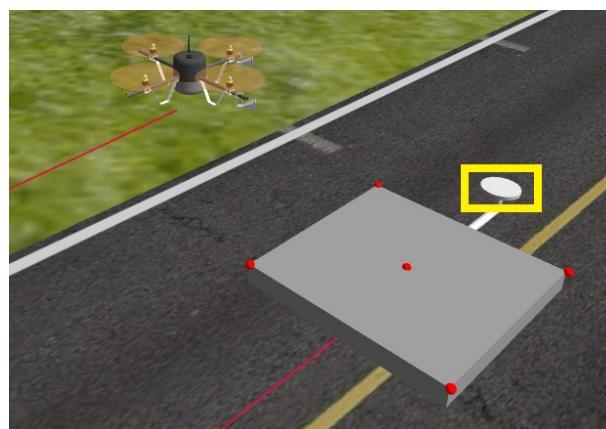
As soon as the autolanding procedure is started, the VPA will start to propagate the platform's movement. The VPA then sends the platform's latitude, longitude and MSL to Simulink. The ENU vector is then calculated in Simulink and packed into the binary structure of the ENU packet. The ENU packet is finally sent to the vehicle.

The HIL procedure is therefore designed to be as analogous as possible to an actual flight test. Figure 8.4 illustrates how the events unfold during HIL simulation and practical testing.



**Figure 8.4 – HIL Simulation and Flight Demonstration Test Execution Sequence**

A three-dimensional model, shown in Figure 8.5, was created to represent the platform in the visual environment. The model was created to scale to represent the actual platform that is used for the project. The Novatel antenna location, indicated by the yellow rectangle, was also modelled to accurately represent the offset between the landing location and the GPS antenna. More information on the platform that was used can be found in Chapter 6.



**Figure 8.5 – The Three-Dimensional Platform Model**

### 8.3 HIL Simulation Conditions

The possible conditions during an actual flight test are discussed. Intuitively, the most influential factors that could affect the landing capability of the vehicle are external disturbances like wind and variation in the speed of the platform.

The simulation incorporates constant wind and wind gusts. From flight tests done before the start of the project, wind speeds of up to  $5 \text{ m/s}$  were deemed acceptable for the vehicle to fly safely. Flight tests would therefore not be conducted in winds higher than  $5 \text{ m/s}$ , which provides an indication of worst-case wind speeds that can be used in HIL simulations. Constant winds and wind gusts were modelled, both blowing in the same constant direction. Constant winds of  $3 \text{ m/s}$  with wind gusts varying around  $\pm 2.5 \text{ m/s}$  were implemented. The wind gusts were modelled by use of a Band-Limited White Noise Blocks (BLWN) with a low pass filter.

The wind was modelled to blow in a north-eastern direction. In simulation, the platform moved approximately in a northern direction. The wind will therefore have both a crosswind and an in-track component in simulation.

HIL simulations will be conducted with platforms translating at  $3 \text{ m/s}$ ,  $5 \text{ m/s}$  and  $7 \text{ m/s}$ . The platform will be translating at a constant velocity and heading.

The platform model used in the project is an existing platform model that has been used in the ESL. The platform models contact between the two objects by using a spring damper system. Friction is modelled between the landing gear of the vehicle and the platform.

Differential Global Positioning System (DGPS) delay should also be incorporated into the HIL setup and plays a vital role in the stability of the position controllers. A means of quantifying the DGPS delay when using Novatel Align<sup>TM</sup> posed a challenge. Neumann et al performed tests on a RT-2 RTK Novatel DGPS in 1996, and found that a typical position measurement latency of  $100 \text{ ms}$  can be expected [32]. Information on delays for a Novatel DGPS that functions in Align<sup>TM</sup> mode could not be found. Unfortunately, time constraints for the project also did not allow for measuring this delay.

An assumption had to be made to quantify the delay when using Align<sup>TM</sup>. While functioning in Align<sup>TM</sup> mode, the DGPS communicates between the Rover sensor and the Master sensor at a baud rate that is an order of magnitude higher compared to when the system is functioning in RTK mode. It is therefore assumed that this increased baud rate will decrease the delay relative to when RTK is used. The assumption was made that the delay is half of the delay experienced when using RTK mode. A delay of  $50 \text{ ms}$  was used in the HIL simulation.

### 8.4 Simulation Results

The results from HIL simulations, which consisted of simulating the entire autolanding procedure, are presented in this section. The sequence of events of every HIL simulation was the same, as illustrated in Figure 8.4. Several HIL simulations were conducted throughout the course of the project, but the results presented in this chapter were all obtained under the same conditions so that comparisons could be made. The platform moved in approximately the same direction in every simulation, with its heading varying by approximately  $2^\circ$  between simulations. The wind was blowing in the same direction in every simulation.

Eighteen consecutive simulations were conducted. Nine were conducted with wind and nine without wind. The results of only one simulation, performed with the platform moving at  $3 \text{ m/s}$  in wind, is presented in this chapter. The results for the other simulations are all displayed in Appendix E. Shown in Figure 8.6 is the absolute horizontal error from the beginning of the autolanding process until touchdown.

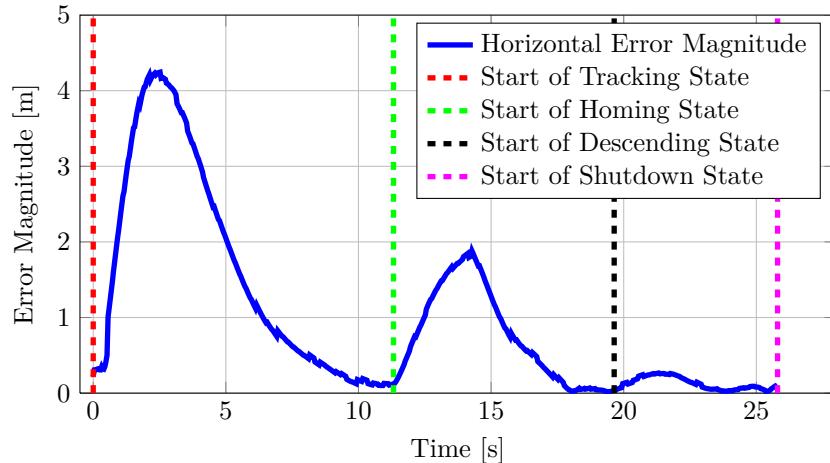


Figure 8.6 – HIL Simulation Absolute Horizontal Position Error During 3 m/s Simulation

Four dashed lines are also plotted. The dashed lines indicate the states of the state machine, described in Chapter 7. The red dashed line indicates at what time the vehicle entered the Tracking state, where the vehicle has to track the landing location from 3 m behind and 3 m above. Immediately after the red dashed line, the position error grows rapidly as the vehicle is stationary and the platform is moving at 3 m/s. As the vehicle accelerates, it gains speed and catches up to the platform. The vehicle then follows the reference point of 3 m behind the platform. As soon as the absolute error has remained under 0.5 m for 3 seconds, the next state is entered.

A green dashed line indicates the start of the Homing state. In this state, the vehicle homes in at a speed of 1 m/s to above the landing location. The error again starts to increase as the position reference is moved to above the landing location at a rate of 1 m/s. The vehicle has to increase its speed to finally reach the new position reference. When the absolute error has remained under 0.5 m for three seconds, the Descending state is entered.

The black dashed line indicates the start of the Descending state, where the vehicle starts to descend onto the platform. The altitude controller is disarmed and the vehicle descends at a constant rate of 0.5 m/s. If the vehicle has stayed within the 1 m diameter cylinder throughout the Descending state, it will make contact with the platform.

The final state, entered when the vehicle has touched down on the platform, is indicated by the magenta dashed line. An accelerometer spike with a magnitude exceeding 15 m/s<sup>2</sup> is measured, causing the state machine to enter the Shutdown state. The vehicle finally touches down on the platform with a absolute position error of 0.11 m.

Plotted in Figure 8.7 is the ENU Up measurement for the same HIL simulation.

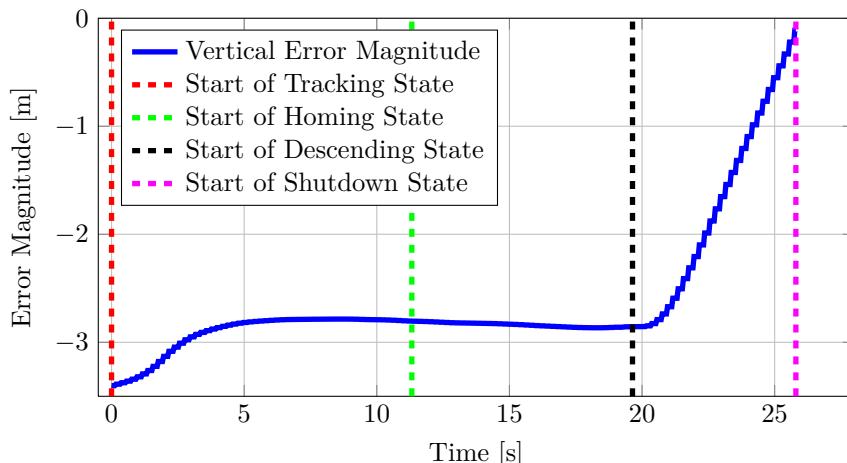
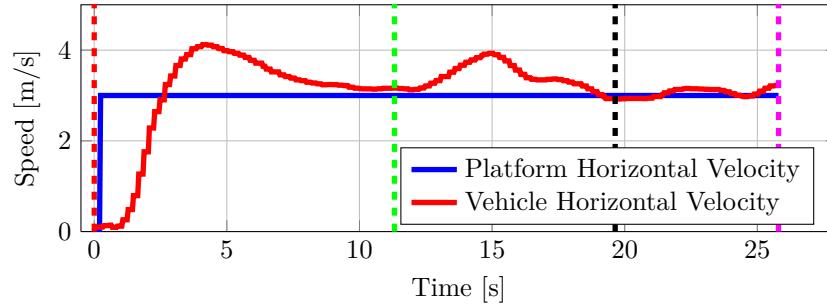


Figure 8.7 – HIL Simulation ENU Vector Up Component During 3 m/s Simulation

As soon as the Tracking state is entered, the vehicle descends from the arbitrary height it was holding during the Standby state. The vehicle descends to 3 m above the platform. It stays approximately 3 m above the platform until the Descending state is entered. The vehicle then descends at a rate of 0.5 m/s until it makes contact with the platform.

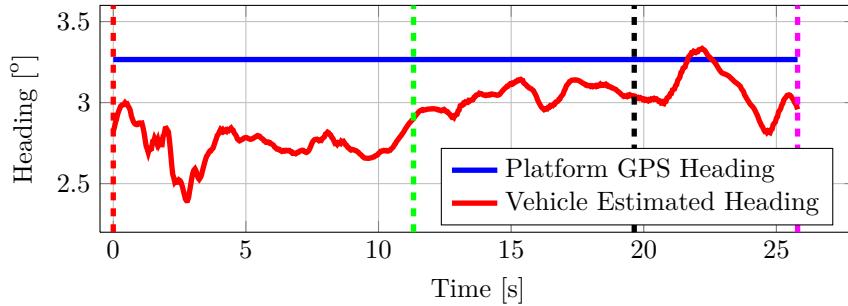
Plotted in Figure 8.8 is the velocity of the vehicle and the platform during the autolanding procedure. The dashed vertical lines again indicate when the different states of the autolanding state machine are entered.



**Figure 8.8 – HIL Simulation Vehicle and Platform Velocity During 3 m/s Simulation**

The velocity of the vehicle increases from zero to above 4 m/s as it gains on the position of the platform during the Tracking state. Approximately 3 seconds before it enters the Homing state, the vehicle matches the velocity of the platform. The velocity increases again while the vehicle homes in to above the landing location. From this point onwards, the vehicle approximately matches the velocity of the platform to the end of the landing procedure. The vehicle touches down on the platform with a velocity difference of 0.2 m/s.

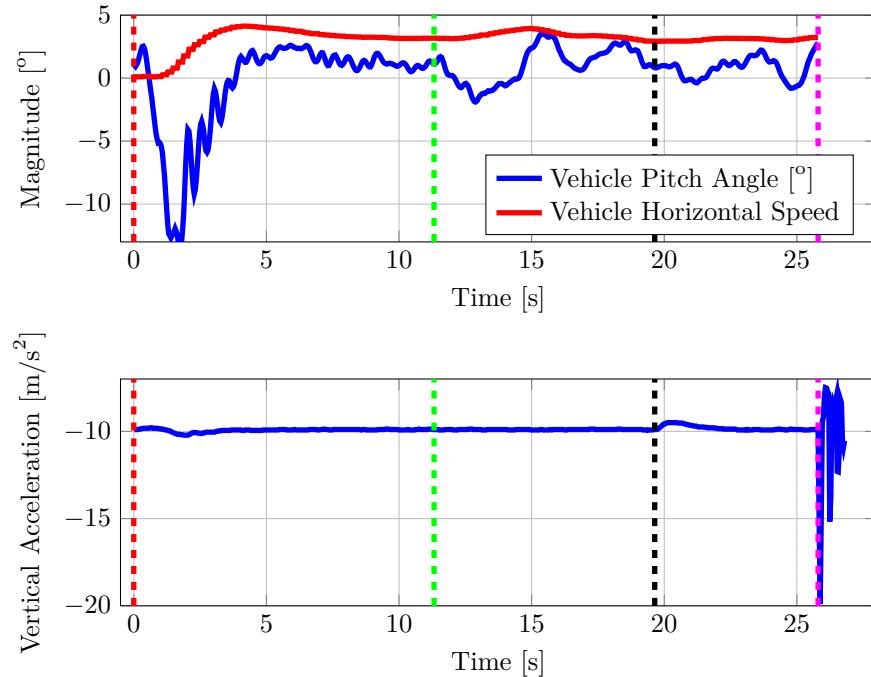
Shown in Figure 8.9 is the platform heading and the vehicle heading relative to north.



**Figure 8.9 – HIL Simulation Vehicle and Platform Heading During 3 m/s Simulation**

The platform's heading remains completely constant and the vehicle's heading varies by slightly more than 0.5° throughout the autolanding procedure. The vehicle finally touches down with its heading within 0.25° of the platform's heading.

Shown in Figure 8.10 is the pitch angle of the vehicle and the accelerometer spike measured when the vehicle makes contact with the platform. The velocity is plotted with the pitch angle of the vehicle for convenience.

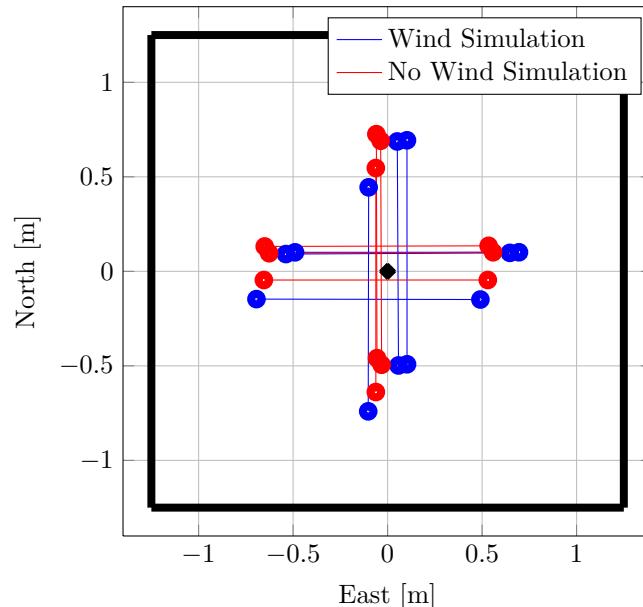


**Figure 8.10** – HIL Simulation Vehicle Pitch Angle and Vertical Accelerometer Spike During 3 m/s Simulation

The pitch angle almost reaches  $-13^\circ$ , as the vehicle initially pitches when it accelerates from a standstill. The vehicle finally touches down with a pitch angle of  $2.75^\circ$ . The pitch angle is small enough not to be considered a problem during touchdown.

A specific force spike of  $-20 \text{ m/s}^2$  is measured when the vehicle makes contact with the platform. This spike is sufficiently large for the vehicle to know it has touched down on the platform, and for the flag to be raised to enter the Shutdown state.

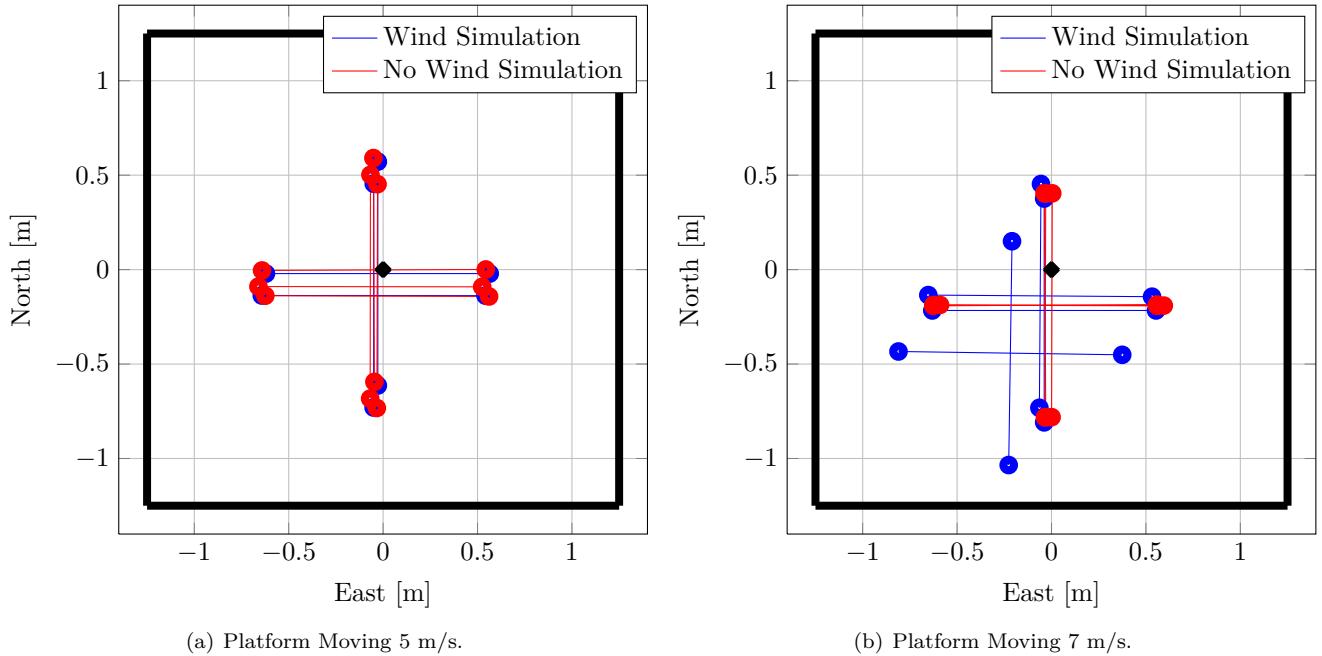
As previously mentioned, 18 consecutive simulations were conducted. Six of the simulations were done with a platform speed of 3 m/s, where three were performed with wind and three without wind. The touchdown location is plotted for the six simulations conducted at 3 m/s in Figure 8.11.



**Figure 8.11** – The Touchdown Locations Relative to the Center of the Platform During 3 m/s Simulation

The platform and the vehicle are represented to scale in Figure 8.11. The vehicle is represented for each of the simulations by a cross and four markers, indicating where the landing gear of the vehicle would have made contact with the platform. The vehicle is plotted with its heading relative to the platform's heading. The three simulations where wind was included in the simulation depict a scattered touchdown, whereas the simulation that did not include wind displays more repeatable behavior.

Six of the remaining simulations were performed at  $5\text{ m/s}$  and six at  $7\text{ m/s}$  where three of each were with wind and three without wind. Shown on the left of Figure 8.12 is the touchdown locations for the simulations performed at  $5\text{ m/s}$  and on the right simulations performed at  $7\text{ m/s}$ .



**Figure 8.12 –** Touchdown Locations During  $5\text{ m/s}$  and  $7\text{ m/s}$  HIL Simulations

One of the simulations at  $5\text{ m/s}$  did not pass the state machine criteria during the Descending state, thus not touching down. The simulations conducted at  $5\text{ m/s}$  depict repeatable behaviour. When wind was included in the  $7\text{ m/s}$  simulations scattered touchdown locations were observed, whereas if no wind was modelled, the results were again repeatable.

## 8.5 Discussion

To summarise the results obtained during the 18 HIL simulations, a table is generated. Table 8.1 shows the relevant factors and parameters important to the final touchdown during each simulation. A brief discussion of each factor is provided.

The horizontal error is the absolute horizontal error seen at touchdown. This error must be less than  $0.5\text{ m}$  for the vehicle to safely touch down on the platform. A trend can be seen that the landing accuracy decreases as the platform speed increases. The simulations where wind was modelled also deteriorate the touchdown accuracy.

**Table 8.1** – Relevant Parameters At Touchdown For Each HIL Simulation

HIL Simulation Conditions	Horizontal Error [m]	Pitch Angle [ $^{\circ}$ ]	Specific Force Spike [ $m/s^2$ ]	Velocity Difference At Touchdown [m/s]
3 m/s, No Wind	0.077	0.18	-19.67	0.004
3 m/s, No Wind	0.145	0.96	-23.11	0.004
3 m/s, No Wind	0.104	0.36	-23.48	0.055
<b>Average</b>	<b>0.109</b>	<b>0.50</b>	<b>-22.09</b>	<b>0.021</b>
3 m/s, Wind	0.109	2.75	-19.89	0.222
3 m/s, Wind	0.179	1.03	-23.73	0.196
3 m/s, Wind	0.144	2.04	-22.09	0.004
<b>Average</b>	<b>0.144</b>	<b>1.94</b>	<b>-21.90</b>	<b>0.141</b>
5 m/s, No Wind	0.049	-1.10	-17.39	0.077
5 m/s, No Wind	0.113	-1.23	-21.10	0.105
5 m/s, No Wind	0.144	-1.03	-19.62	0.101
<b>Average</b>	<b>0.102</b>	<b>-1.12</b>	<b>-19.40</b>	<b>0.094</b>
5 m/s, Wind	0.146	1.20	-20.96	0.176
5 m/s, Wind	NA	NA	NA	NA
5 m/s, Wind	0.035	0.96	-20.61	0.174
<b>Average</b>	<b>0.091</b>	<b>1.08</b>	<b>-20.79</b>	<b>0.175</b>
7 m/s, No Wind	0.189	1.57	-16.80	0.131
7 m/s, No Wind	0.193	0.72	-18.16	0.139
7 m/s, No Wind	0.191	0.04	-18.30	0.119
<b>Average</b>	<b>0.191</b>	<b>0.77</b>	<b>-17.75</b>	<b>0.130</b>
7 m/s, Wind	0.493	1.48	-17.37	0.056
7 m/s, Wind	0.151	2.46	-17.07	0.051
7 m/s, Wind	0.220	-0.77	-19.23	0.152
<b>Average</b>	<b>0.288</b>	<b>1.57</b>	<b>-17.89</b>	<b>0.086</b>
<b>Grand Average</b>	<b>0.158</b>	<b>1.17</b>	<b>20.09</b>	<b>0.104</b>

The pitch angle is shown to indicate at what pitch the vehicle will touch down. If the pitch angle is too large, the effects seen during touchdown could be unfavorable. A pitch angle that is too large might require flaring the vehicle just before touching down. The pitch angle does not appear to be a problem, however, as a maximum pitch angle of  $2.75^{\circ}$  was recorded at touchdown.

The specific force spike measured at touchdown is used to enter the final state of the state machine. A specific force magnitude of at least  $15 \text{ m/s}^2$  must be measured to enter the Shutdown state. The spike was large enough during all successful landings to enter this state. Nevertheless, a trend can be seen that the spike decreases in magnitude as the platform was moving faster.

The velocity difference at touchdown is also considered an important factor. The velocity difference at touchdown is the difference in speed between the vehicle and the platform at touchdown. If this value becomes too large, the vehicle might slide off of the platform after touching down. No trend can be seen in the data, and the velocity difference at touchdown is not considered dangerous. A maximum velocity difference at touchdown of  $0.22 \text{ m/s}$  is recorded.

Overall, the HIL simulations prove that the vehicle can safely land on the platform at speeds of up to  $7 \text{ m/s}$ . The strategy is to firstly test whether the vehicle can successfully land on the platform at a speed of  $3 \text{ m/s}$ . Attempts can then be made at landing at  $5 \text{ m/s}$ , and if again successful, at  $7 \text{ m/s}$ . An attempt can then finally be made to land the vehicle while the platform is translating at  $30 \text{ km/h}$ , which is equivalent to  $8.33 \text{ m/s}$  and is the goal of the project.

## Chapter 9

# Practical Flight Tests

This chapter presents a series of flight tests that were performed during the course of the project, including the final flight test, where the vehicle autonomously landed on a translating platform. The chapter starts by explaining how the moving platform was assembled and instrumented, as the platform is used in all the flight tests presented in this chapter.

A flight test campaign was devised and an overview of this campaign follows. The campaign highlights the train of thought throughout the execution sequence of the flight tests. The order in which the practical flight tests were performed is presented, as well as the goal of each test.

The flight tests are finally presented. The goal, execution and results for each flight test will be presented and discussed. The chapter closes with a general discussion.

### 9.1 The Translating Platform

This section briefly presents how the translating platform was assembled and instrumented. The assembly consists of a perforated platform, a trailer and the Global Positioning System (GPS) antenna of the Rover sensor.

A perforated platform is available for use in the Electronic Systems Laboratory. The platform measures 2 m by 2 m and is made from aluminum tubes and a perforated aluminum grid. The platform on its own is shown in Figure 9.1.



**Figure 9.1** – The Perforated Platform On Which the Vehicle Will Land

A white border can be seen around the perforated platform. The border is made of rectangular plastic tubing, and was fixed to the platform by use of cable ties, so that the quadrotor would not slip off of the platform while the platform accelerates or decelerates. The borders will also save the quadrotor if it lands on the moving platform with a velocity difference between itself and the platform. If the velocity difference is large enough upon touchdown, the vehicle could slip and fall off of the platform.

A trailer was sourced next, to which the perforated platform could be mounted. A trailer which was suited to the application was hired from a trailer company. The trailer has high side walls, which allows for the perforated platform to be mounted high relative to the axle of the trailer. Mounting the platform high above the axle of the trailer has two benefits: The In-Ground Effect (IGE) is decreased, and the wake turbulence of the motor vehicle towing the trailer will have less impact on the aircraft while it is descending. The platform was fixed to the trailer using cable ties. The assembly can be seen in Figure 9.2.



**Figure 9.2 – The Moving Platform Assembly**

The IGE is decreased, as air can flow freely through the perforated platform and through the side walls of the trailer structure. The turbulent wake of the motor vehicle towing the platform will have less of an effect on the descending aircraft, as the platform is not mounted directly behind it, but above it.

The final step was instrumenting the platform with the Rover GPS antenna. The antenna had to be mounted with an offset to the center of the platform. The antenna could not be mounted in the center, as the aircraft would descend onto it and damage it upon impact. The GPS antenna could also not be mounted underneath the platform, as this would have compromised the visibility of satellites to the antenna.

The Rover GPS antenna was finally mounted in front of the platform and above the trailer's link to the motor vehicle. The location of the Rover GPS antenna is indicated by the yellow rectangle in Figure 9.2.

The assembly could possibly have adverse effects on some of the sensors of the autolanding system, specifically the magnetometer of the aircraft and the Rover GPS sensor. Investigations were conducted in Chapter 6, and are continued in this chapter to determine whether this could be a problem.

## 9.2 Overview of Flight Test Campaign

A flight test campaign was devised which would test each addition to the autolanding system in isolation. This ensured that the build-up to the final flight test was solid, increasing confidence in achieving the project goal. The addition of the Novatel Align<sup>TM</sup> Differential Global Positioning System (DGPS) and the dedicated wireless link implemented for DGPS communication have already been thoroughly tested in Chapter 6.

Five successive practical tests were performed to isolate each addition to the firmware and hardware of the vehicle and platform, and ensure that it performs as intended. The chronological order in which the flight tests were performed can be seen in Table 9.1.

**Table 9.1 – Flight Test Campaign Summary**

Flight Test	Objective	Risk
1	Test re-designed horizontal velocity and position controllers	Low
2	Test acceleration extremes and emergency stop	Medium
3	Midair autolanding on a virtual platform	Medium
4	Autolanding on a stationary platform	High
5	Autolanding on a moving platform	High

The objective of each flight test is briefly stated as well as the risk involved. The risk is an indication of the probability of damage to the vehicle and its surroundings during the flight test. The flight test campaign was constructed so that the risk would be low for the first flight test, and increase as the campaign progresses. Debugging of new additions to the hardware and software of the autolanding system could then be done in the first flight tests, where the risk would be lower than in the final flight tests.

The first flight test was performed to test the re-designed horizontal position and horizontal velocity controllers. The controllers were re-designed in Chapter 5. The stability of the controllers were tested and their response was compared to the response observed in simulation. The results were also compared to the pre-existing control system, to ensure that the re-designed controllers are effective.

The second flight test tested the acceleration extremes that the vehicle would be subject to during the final flight test. The response and stability of the aircraft were investigated under strenuous acceleration and deceleration conditions. A new implementation during this test was the emergency stop feature.

The third flight test was the first flight test that incorporated the state machine for the autolanding procedure. Design of the state machine is documented in Chapter 7. The aircraft attempted to land on a virtual translating platform, thus not coming into contact with any other objects.

In the fourth flight test the vehicle attempted to land on a stationary platform. The platform used was the platform that would be used during the final flight test. The test was the second-last test, done to gain enough confidence in the autolanding system to perform the final flight test.

In the final flight test the aircraft attempted to autonomously land on a moving platform.

### 9.3 Flight Test 1 (Coetzenburg, 6 November 2013)

The goal of the first flight test was to determine whether the re-designed, more aggressive horizontal velocity and position controllers function as simulated. The test was performed at Coetzenburg sports fields in Stellenbosch. Wind conditions during the test were not ideal, with constant winds of 3 m/s, and wind gusts of up to 6 m/s. The wind was blowing in a north-western direction.



**Figure 9.3 – The Vehicle Pitching During a Forward Position Step**

Re-design of the horizontal controllers are documented in Chapter 5. The bandwidth of the velocity and position control loops has been approximately doubled from the pre-existing controllers, making them more aggressive. This was done to decrease the rise and settling time of the position response and to increase the disturbance rejection. The performance and stability of the re-designed controllers needed to be verified practically, which was done by performing horizontal position steps.

During this test the vehicle used the Align<sup>TM</sup> East-North-Up (ENU) vector as position measurement and BestVel as velocity measurement. This was done to gain confidence in using the ENU vector as the position feedback measurement for the flight tests that would follow. The Rover sensor, which is mounted to the platform, was kept stationary. Consequently the platform was also kept stationary for this test.

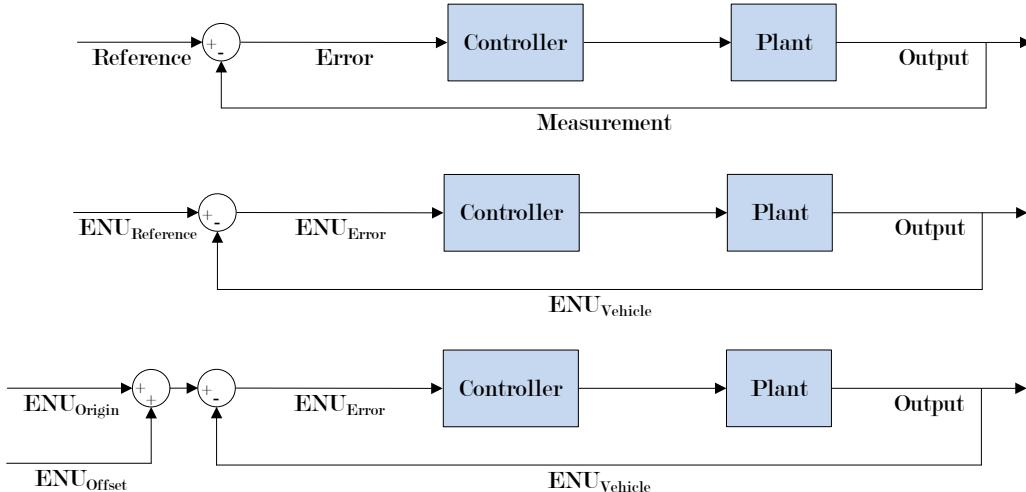
As the ENU vector provides relative position information, a means had to be found to provide relative position references, and to convert them to relative position errors for the position controllers. The ENU vector cannot be used directly as the north, east and down errors of the position controllers, because then the vehicle would control itself to the same position as the Rover GPS sensor.

The problem was approached by defining a reference command called  $ENU_{Reference}$ , to which the vehicle must control itself. In Figure 9.4 three feedback control loops are shown. The first control loop can be seen as the architecture of a general feedback control loop, while the bottom two control loops use the ENU vector as feedback measurement.

Refer to the second feedback control loop. When using the ENU vector for feedback control, the error signal defined as  $ENU_{Error}$  can be expressed similarly to the error signal of the general feedback control loop,

$$ENU_{Error} = ENU_{Vehicle} - ENU_{Reference} \quad (9.3.1)$$

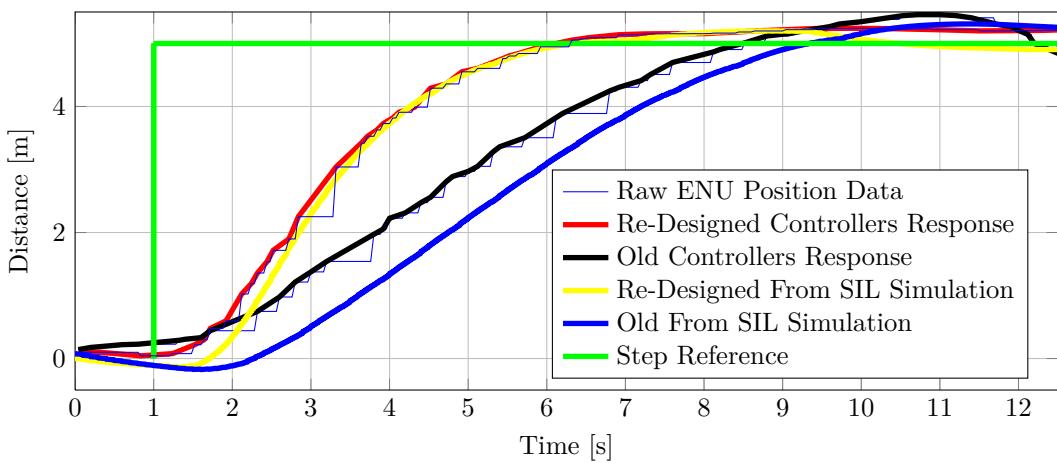
where  $ENU_{Vehicle}$  is the measurement of the instantaneous relative position of the vehicle to the Rover GPS sensor on the platform. The  $ENU_{Reference}$  command can be further

**Figure 9.4** – How The ENU Vector Is Used During Position Steps

broken up into a  $ENU_{Origin}$  and a  $ENU_{Offset}$  term, as seen in the third feedback control loop in Figure 9.4. The  $ENU_{Origin}$  term is a constant, and is the position relative to the Rover GPS sensor on the platform that the position steps must be performed from. The  $ENU_{Offset}$  is a variable term, and is used to command the position steps.

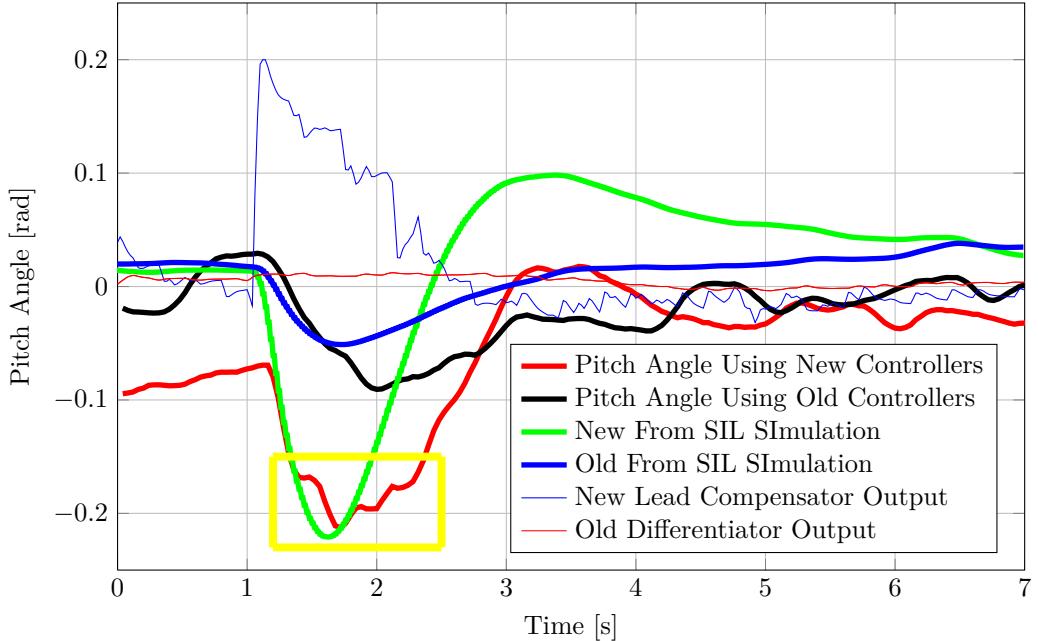
For example, if the  $ENU_{Origin}$  term is zero, the aircraft will perform the position steps from the same location as the Rover GPS sensor on the platform, which is not practical. If, however, the  $ENU_{Origin}$  term is some arbitrary value, like 4 m east, 3 m north and 5 m up, the aircraft will perform the position steps from a location 4 m north, 3 m east and 5 m up relative to the Rover GPS sensor on the platform.

The test was performed by executing 5 m horizontal position steps in the vehicle's body axis. The vehicle was commanded to step 5 m forward, back, left and right. As the vehicle is symmetric about both the lateral and longitudinal axes, the same transient and steady-state behavior should be seen in all directions. Figure 9.5 shows a forward step response while using the re-designed controllers, compared to a forward step response while using the pre-existing velocity and position controllers.

**Figure 9.5** – Forward Step Response Using Pre-Existing and Re-Designed Horizontal Controllers

From Figure 9.5, it can be seen that the rise time has decreased from 5.5 seconds to 3.1 seconds. The 5% settling time has decreased from 10.9 seconds to 4.4 seconds. The re-designed controllers appear to be functioning as simulated in the non-linear Software-In-The-Loop (SIL) simulations performed in Chapter 5.

However, some additional oscillations were observed in the pitch and roll angles of the vehicle when pitching and rolling aggressively during the first part of the transient response of a position step. The additional oscillations are indicated by the yellow square in Figure 9.6, showing the pitch angle for a forward step while using the pre-existing and re-designed controllers respectively.



**Figure 9.6 – Pitch Angle During Forward Step for Pre-Existing and Re-Designed Horizontal Controllers**

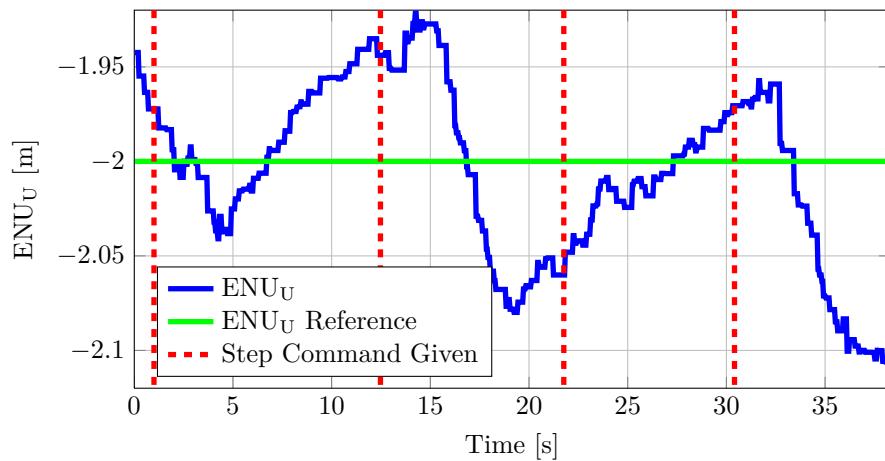
Looking closely at the pitch angle response when the pre-existing controllers are used, similar oscillations can be seen, but they are less pronounced. Noise on the velocity measurement was considered to be the cause. If noise is present, it will be amplified by the lead compensator in the velocity control loop. The velocity control loop in turn commands the tilt angle control loop, which will attempt to control the vehicle's tilt angle to follow the noise-contaminated reference. Yet the oscillations appear to be deterministic rather than stochastic, which indicates that noise might not be the cause of the phenomenon.

The offset of the GPS antenna from the vehicle's Centre of Gravity (CG) was also considered. The Master GPS antenna is fixed approximately 250 mm above the CG of the vehicle. This will induce a velocity measurement while the vehicle is pitching and rolling, even if the aircraft's CG is stationary. Additional linear and non-linear simulations were conducted, which accurately modelled the antenna offset from the CG. The simulation results did not, however, provide convincing evidence that this is the cause of the problem. It is also worth noting that the pitch angles seen in practice corresponds well with the pitch angles recorded from non-linear SIL simulation results.

The origin of the additional oscillations could not be determined, as time was running out to perform succeeding flight tests. It was decided that flight testing would continue, even

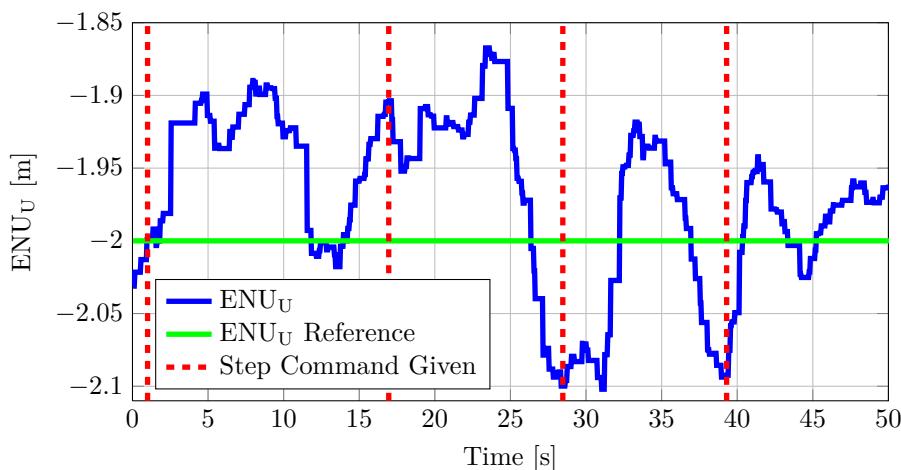
though the additional oscillations were present, as the phenomenon did not appear to be an instability problem. Flight testing will continue using the re-designed horizontal controllers.

The cross-coupling effect that the more aggressive horizontal controllers have on the vertical control system must also be checked. More aggressive pitching and rolling will require more prompt vertical thrust compensation from the vertical control system. Figure 9.7 shows the ENU Up measurement compared to the ENU Up reference for a certain time period during the flight test. The measurements show the response of the vertical position control to commanded horizontal position steps using the re-designed, more aggressive horizontal controllers. The red dashed lines indicate instances where 5 m horizontal steps were commanded. Enough time was given between steps, allowing the vehicle to reach a steady-state.



**Figure 9.7** – Vertical Disturbance Rejection Due To Horizontal Position Step, Using Re-Designed Controllers

While stepping forward, back, left and right, the vehicle's height varied by 0.15 m. Figure 9.8 shows the response of the vertical position control to commanded horizontal steps using the pre-existing, less aggressive horizontal controllers.



**Figure 9.8** – Vertical Disturbance Rejection Due To Horizontal Position Step, Using Pre-Existing Controllers

Looking at Figure 9.8, the vehicle's height varied by 0.23 m while stepping forward, back, left and right. The variation in height when using the re-designed horizontal controllers does not differ significantly from the variation in height when using the pre-existing controllers. The vertical control system therefore does not appear to be more adversely affected by the cross-coupling from the re-designed horizontal controllers.

In conclusion, the test results illustrate that the vehicle is stable with the re-designed velocity and position controllers. The controllers reduce the rise and settling time of the position response as designed. The additional oscillations observed in the pitch and roll angles of the vehicle do not appear to be a stability problem. It was thus deemed acceptable to continue the project using these controllers.

#### 9.4 Flight Test 2 (Coetzenburg, 6 November 2013)

The goal of the second flight test was to test the flight control system at the acceleration extremes that the vehicle would be required to perform at while executing an autolanding. The test was also done at Coetzenburg sports fields, on the same day as flight test 1. Wind conditions were similarly unfavorable, with constant winds of 3 m/s and wind gusts of up to 6 m/s. The wind was blowing in a north-western direction.



**Figure 9.9** – The Vehicle Decelerating Aggressively During an Emergency Stop

New implementations in the second flight test included a slewed limit on the velocity feed forward command and the integration of the emergency stop feature, which would be active during succeeding flight tests. These implementations are documented in Chapter 7. The ENU vector was again used for position measurements, along with BestVel for velocity measurements.

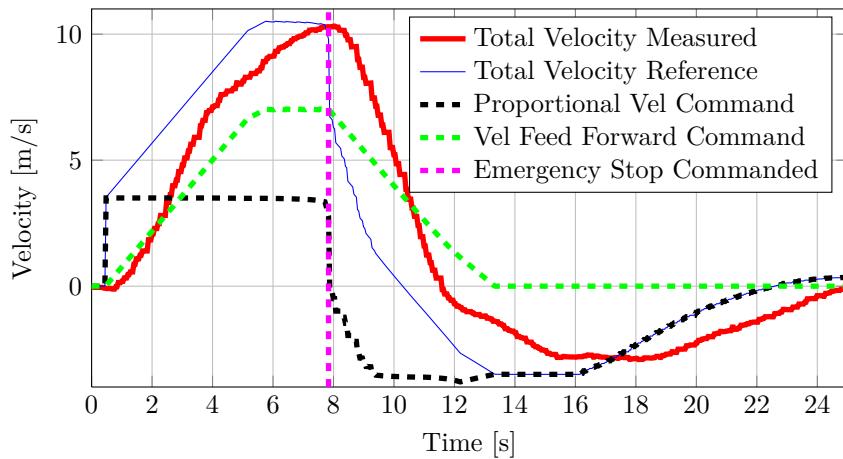
The flight test had to be designed to test the acceleration extremes that the vehicle could be subjected to during the autolanding procedure. The one extreme is where the vehicle has to accelerate from a standstill to start tracking the platform. The other extreme is where the vehicle attempts to come to an immediate standstill when the emergency stop is triggered.

The test was executed by performing a large position step, with added false feed forward velocity, followed by an emergency stop. This simulates the landing procedure starting with the platform far ahead of the vehicle, translating at an arbitrary velocity. As a result,

a maximum acceleration command would be induced. When the vehicle has reached the reference position, it would be traveling at its maximum speed. Performing an emergency stop at this point would realistically illustrate how the vehicle behaves under demanding deceleration conditions. A position step of 50 m and false feed forward velocity of 7 m/s were chosen for the test. The position step reference was also issued by using ENU offsets as in the first flight test, where the platform and the Rover GPS sensor were kept stationary.

The maximum commanded velocity, given the 50 m position step reference and the 7 m/s feed forward velocity reference, will be 10.5 m/s. This command is the sum of the feed forward command and the velocity commanded by the position controller. The position controller is limited to commanding a velocity of 3.5 m/s for large position errors. The 10.5 m/s velocity command is thus the sum of the 7 m/s feed forward velocity and 3.5 m/s commanded by the position controller.

The test was performed twice. Illustrated in Figure 9.10 are the velocity references and measurement in the direction of the 50 m step during one test. Results from the second test can be found in Appendix C. The 50 m position step command is issued shortly after  $t = 0$  seconds, whereas the emergency stop was commanded approximately 8 seconds into the test.

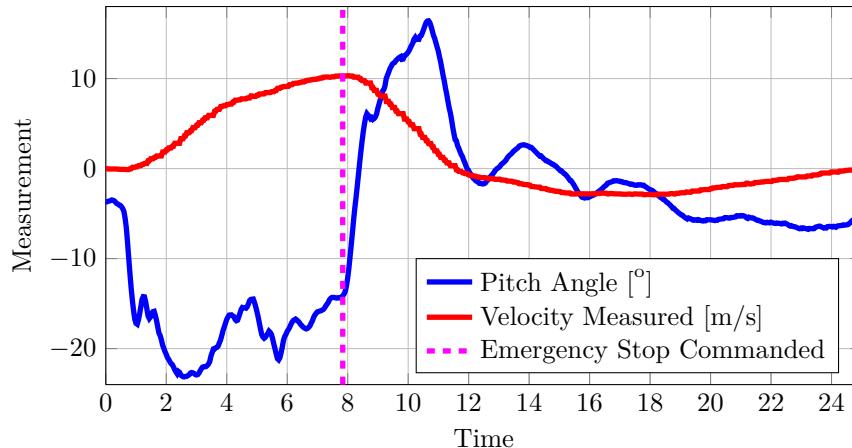


**Figure 9.10 – Forward Velocity References and Measurement During the Second Flight Test**

First consider the proportional and feed forward velocity commands. The proportional velocity command immediately limits to 3.5 m/s when the position step is issued, whereas the feed forward velocity command is slewed from 0 m/s up to 7 m/s. After the emergency stop is commanded, the proportional velocity command limits at -3.5 m/s and the feed forward velocity command is again slewed, from 7 m/s to 0 m/s. The proportional velocity command remains negative for about 14 seconds as the aircraft overshoots the position where the emergency stop was commanded.

Now consider the total velocity reference and the velocity measured. The vehicle accelerates aggressively, reaching a maximum velocity of 10.3 m/s before the emergency stop is commanded. The vehicle then decelerates aggressively in an attempt to come to a standstill. The vehicle's velocity response shows that it slightly lags the velocity reference, but not so much that it is considered a problem. The response is stable, indicating that the flight control system is capable of accelerating and decelerating aggressively while maintaining stability.

The pitch angle response during maximum acceleration and maximum deceleration is also investigated. The vehicle's pitch and roll angle commands are limited to  $\pm 22.5^\circ$ , and correspond to the maximum acceleration that the vehicle can maintain. Figure 9.11 shows the measured pitch angle response during the acceleration tests. The measured velocity is also plotted as reference.



**Figure 9.11** – Recorded Pitch Angle During The Second Flight Test.

The pitch angle limit is reached during forward acceleration. The pitch angle saturates at its negative limit for a duration of 1.1 seconds, reaching a maximum negative pitch angle of  $23^\circ$ . After the emergency stop has been commanded a maximum positive pitch angle of  $16.4^\circ$  is seen, indicating that the pitch angle does not saturate during the emergency stop deceleration.

To conclude this section, the results indicate that the vehicle is capable of performing adequately during the most demanding acceleration and deceleration conditions that will be seen while performing an autoland. The actuators are capable of aggressive acceleration and deceleration for the time periods required, ensuring that the vehicle remains stable.

## 9.5 Flight Test 3 (Helderberg Radio Flyers, 8 November 2013)

The goal of the third flight test was to verify the correct operation of the autolanding state machine, documented in Chapter 7. The test location was Helderberg Radio Flyers (HRF) outside of Somerset West. Constant winds of  $3 \text{ m/s}$  with wind gusts of  $5 \text{ m/s}$  were present, blowing in a north-eastern direction. The state machine had to be tested while minimizing the risk of damage to the vehicle. The problem was resolved by landing on a translating virtual platform.



**Figure 9.12** – The Vehicle After Touchdown on the Virtual Platform

The virtual platform translates with a constant heading and height above ground level. The vehicle will then in effect ‘land’ on the virtual platform without coming into contact with anything. The Virtual Platform Application (VPA), presented in Chapter 8, was used to fabricate data for the virtual platform from the Ground Station Personal Computer (PC). Fabricated data is sent as a packet to the vehicle through the wireless telecommand and telemetry link.

The virtual platform’s position was simulated by adding ENU offsets to ENU origin values, similar to what was done in the first flight test. Table 9.2 exhibits what each packet of virtual platform information contained.

**Table 9.2** – The Packet Sent to the Vehicle During Virtual Platform Landings

Fabricate Measurement	Unit
ENUN,Offset	[m]
ENU <sub>E</sub> ,Offset	[m]
ENU <sub>U</sub> ,Offset	[m]
Virtual Platform Heading	[rad]
Virtual Platform Velocity	[m/s]

The rate at which the information of the virtual platform is sent to the vehicle was determined beforehand (see Chapter 6) and was further validated during this flight test. Recall that the telecommand and telemetry link becomes saturated when sending the virtual platform’s position and velocity at 10 Hz. A rate of 1 Hz did not saturate the link, but was too low and would not represent the behavior of the platform very well. A rate of 5 Hz was found to be a good compromise, and did not saturate the telecommand and telemetry link excessively, while still providing measurements of the platform’s velocity and heading at a sufficient update rate.

A variable delay in sending and receiving the platform’s measurement was always observed, regardless of the rate at which the platform measurements were sent. Time constraints did not allow for finding a solution to the problem. This did, however, impact the virtual landing tests undesirably, as will be seen in the test results.

The position of the virtual platform was propagated by the On-Board Computer (OBC) of the vehicle between receiving the virtual platform’s position and velocity updates from the VPA. This basically means that the reference ENU vector that the vehicle must control itself to, is propagated to smoothen the virtual platform’s movement. This propagation is not to be confused with the propagation of the measured ENU vector, as presented in Chapter 6.

The propagation of the virtual platform’s position was done in an attempt to better represent the virtual platform’s movement, as the send rate of the fabricated virtual platform’s measurements is only 5 Hz and subject to variable delays. The propagation was performed by numerical integration of the virtual platform’s velocity. This involved propagating the ENU offsets between platform position updates. For example, the north offset is propagated as follows on the OBC:

$$ENUN,Offset,Propagated = ENUN,Offset + V_{N,VP} \cdot T_s \quad (9.5.1)$$

where  $T_s$  is the sample time of the OBC,  $V_{N,VP}$  the virtual platform’s north velocity and  $ENUN,Offset$  the offset representing the virtual platform’s position.

Propagation of the measured ENU vector, as presented in Chapter 6, is always performed on the OBC of the vehicle and is done to provide a higher update rate of the ENU vector.

Recall that the measured ENU vector is also only updated at 5 Hz, due to the quality of DGPS communication between the Master and the Rover sensors. The measured ENU vector's north component will normally be propagated by use of Equation 9.5.2, as explained in Chapter 6:

$$ENU_{N,Propagated} = ENU_N + (V_{N,Master} - V_{N,Rover}) \cdot T_S \quad (9.5.2)$$

The propagation is performed by numerical integration of the difference in the velocity of the Rover sensor, mounted on the actual platform, and the Master sensor which is mounted on the vehicle. Noise from the velocity measurements of both the Master and the Rover sensors will thus contaminate the propagation of the measured ENU vector. A better solution would have been to implement a proper estimation algorithm for both estimation of the ENU offsets and the measured ENU vector, but time constraints for the project did not allow for this.

Yet this specific flight test is a special case, as it is known that the actual platform is stationary, and that the Rover sensor's velocity measurement should always be zero. The Rover sensor's velocity was thus hard-coded to zero, so that non-zero velocity measurements would not be included during propagation calculations. The ENU north component was propagated by only using  $V_{N,Master}$ , the Master sensor's measured north velocity:

$$ENU_{N,Propagated} = ENU_N + V_{N,Master} \cdot T_S \quad (9.5.3)$$

By performing these propagations, the estimated states of both the virtual platform and the vehicle are available to the flight control system at a higher sampling rate. Four virtual landing tests were performed, each with different virtual platform and state machine parameters. The parameters for each of the tests can be seen in Table 9.3.

**Table 9.3** – Parameters Used During the Virtual Platform Landing Flight Test

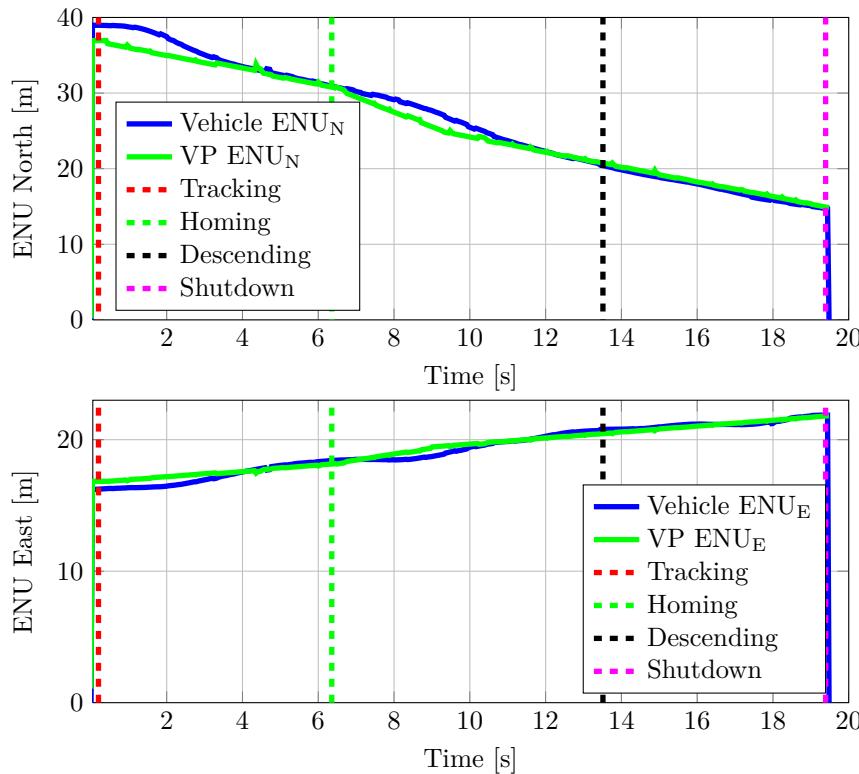
Virtual Landing Attempt	Tracking Time [s]	Tracking Bound [m]	Homing Time [s]	Homing Bound [m]	Virtual Platform Speed [m/s]
Attempt 1	3	3	3	3	1
Attempt 2	3	2	3	2	1
Attempt 3	3	2	3	2	2
Attempt 4	3	1	3	1	1

It should be noted that the speed at which the virtual platform is moving is notably lower than the 30 km/h speed at which the actual platform should be moving. Furthermore, the error bounds were enlarged for some of the tests. Recall that the error bounds must be 1 m during the Tracking state and the Homing state, in order to land within 0.5 m of the middle of the platform. The motivation behind choosing the low virtual platform speeds and the enlarged error bounds was due to two factors.

The one being that the distance between the two GPS sensors can become too large, and as a result DGPS communication could be compromised. The actual platform is stationary for this flight test. The Ground Station setup is located on the actual platform and is thus not moving. The transceivers used for telemetry communication and dedicated DGPS communication are therefore also stationary, as they form part of the Ground Station. The distance between the vehicle and the platform can therefore become large, as both are not moving together for this test. If the distance between the Ground Station and the vehicle was to exceed 80 m, communication would be disrupted. The second reason for choosing

the low translating speeds is due to the presence of variable time delays in the telecommand and telemetry link, of which the effect will become apparent shortly.

The first two tests resulted in successful virtual platform landings, but the last two tests did not, due to the fact that all of the state machine's error checks were not passed. The results from a successful virtual landing are shown in Figure 9.13. The results are from the second virtual platform landing, where the virtual platform was translating at 1 m/s and the error bounds were 2 m for the Tracking and the Homing states.

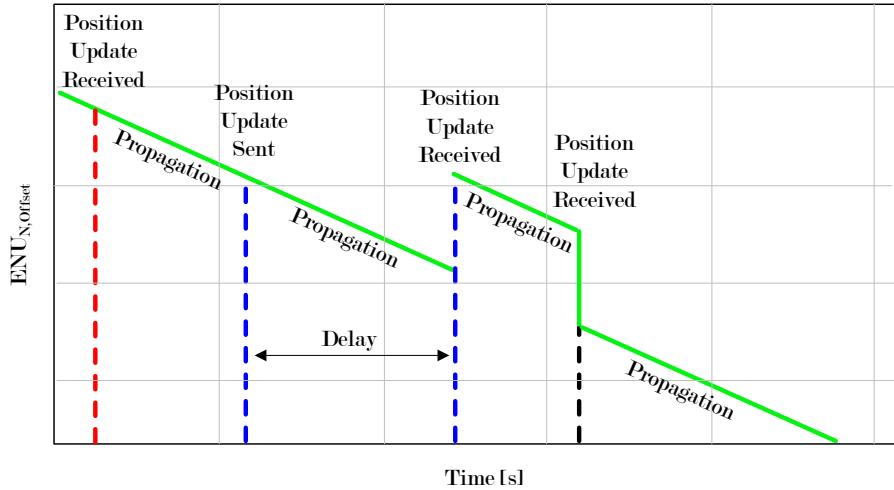


**Figure 9.13 – Tracking of the Virtual Platform During Second Attempt**

In the top graph, the north component of the reference ENU vector is compared to the north components of the actual ENU vector measured by the vehicle. Similarly, in the bottom graph the east ENU reference vector components are compared to the actual east ENU components measured by the vehicle. What can effectively be seen is how the vehicle tracks the virtual platform, relative to the Rover sensor on the stationary platform. Four dashed lines are also plotted, indicating the start of the different states of the state machine.

Toothed edges and irregularities can be seen on the green curve, that represents the reference ENU trajectory that the vehicle needs to follow. The toothed edges are a result of the variable delays between when the VPA sends virtual platform information and when it is received by the vehicle. The phenomenon is explained with reference to Figure 9.14.

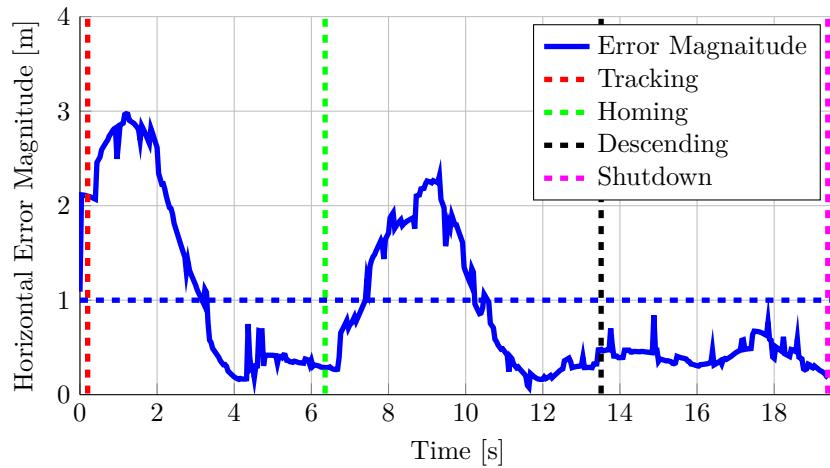
The vehicle propagates the virtual platform's position after each position update received through the telemetry link. When the VPA sends a new update, and the update is delayed by the telemetry link, a toothed edge is created. The toothed edge will only disappear once an update has been received without a notable delay between being sent by the VPA and received by the vehicle. This undesirable effect did not make it possible to successfully



**Figure 9.14** – Illustration of the Phenomenon in the Virtual Platform Position Due to Telemetry Link Delays

perform virtual platform landings at speeds higher than  $1 \text{ m/s}$ , with error bounds smaller than  $2 \text{ m}$  for the Tracking and Homing states.

Figure 9.15 shows the absolute horizontal error during the landing procedure. Due to delays in the telemetry link while sending the platform information, the toothed edges in Figure 9.13 manifest as sharp spikes in Figure 9.15, which can easily prevent the vehicle from passing the state machine's error checks, even though the vehicle is tracking the virtual platform adequately.



**Figure 9.15** – Virtual Landing Absolute Horizontal Position Error During Second Attempt

The state machine lines are plotted with an additional horizontal blue dashed line. The blue dashed line is the error bound within which the vehicle needs to remain in order to advance to the next state of the state machine. For this test, the error bounds were  $2 \text{ m}$ . The absolute error is thus not allowed to exceed  $1 \text{ m}$ .

Between the start of the Tracking and Homing state, the error grows and then decreases as the vehicle catches up with the tracking location, 3 m behind the landing position. As soon as the vehicle has passed the check of not going outside of the 2 m diameter sphere for 3 seconds, the Homing state starts. The error again increases as the reference position is homed to directly above the landing location. Once the vehicle has passed the state machine's check while in the Homing state, the Descending state is entered. The vehicle starts to descend and does not deviate from the landing location by more than 1 m. The Shutdown state is finally entered when the vehicle reaches the height of the virtual platform.

Figure 9.16 shows the vertical tracking of the vehicle during the same virtual platform landing. The plot shows the vehicle's measured height above the virtual platform compared to the reference height commanded by the autolanding state machine.

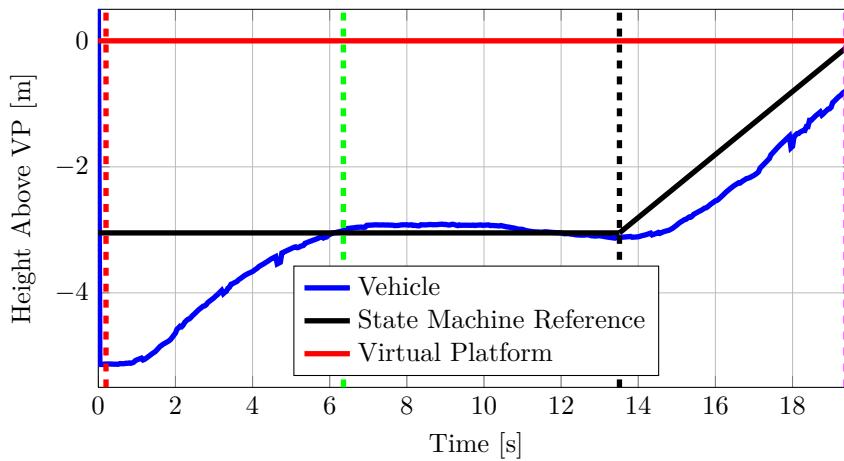


Figure 9.16 – Virtual Landing Vertical Reference and Position During Second Attempt

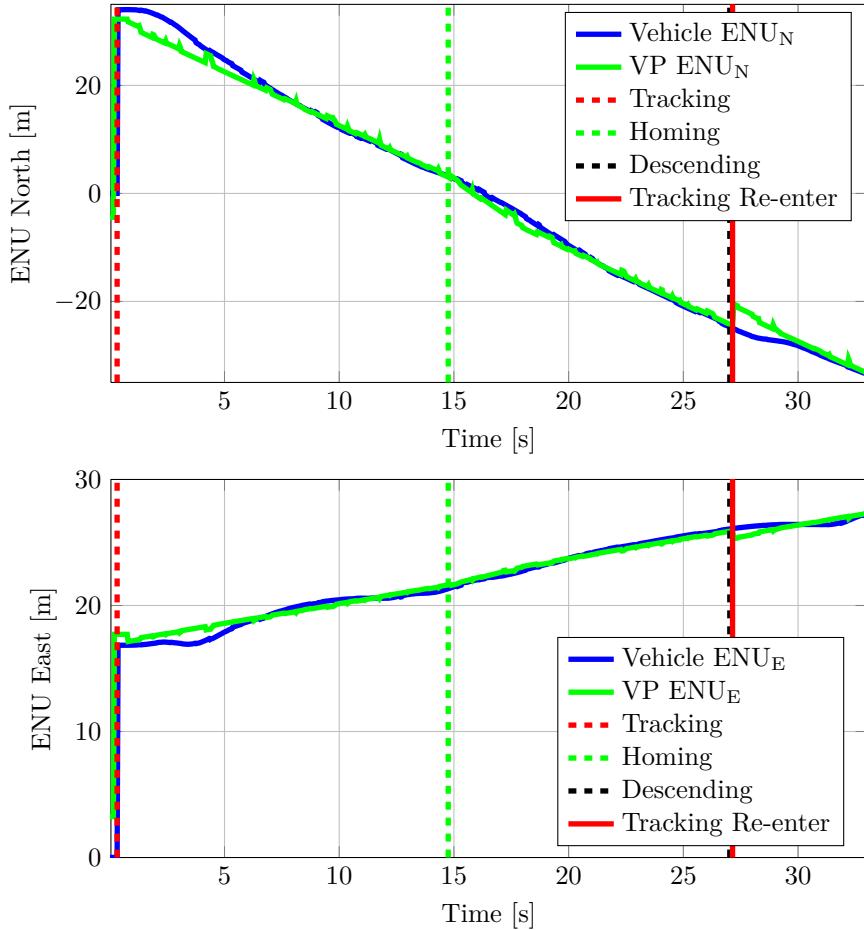
As soon as the Tracking state starts, the vehicle descends from an initial arbitrary height to 3 m above the virtual platform. After the Homing state is entered, the vehicle descends slightly as it increases its pitch angle to home to above the landing location. After the Descending state is entered, the vehicle descends at 0.5 m/s until it makes contact with the virtual platform, entering the Shutdown state.

The results of the third virtual platform landing attempt, where the vehicle did not successfully advance through all the states of the state machine, are now analyzed. The virtual platform speed was 2 m/s and the error bounds for the Tracking and Homing states were both 2 m. Figure 9.17 shows how the vehicle tracked the platform during the third virtual platform landing attempt.

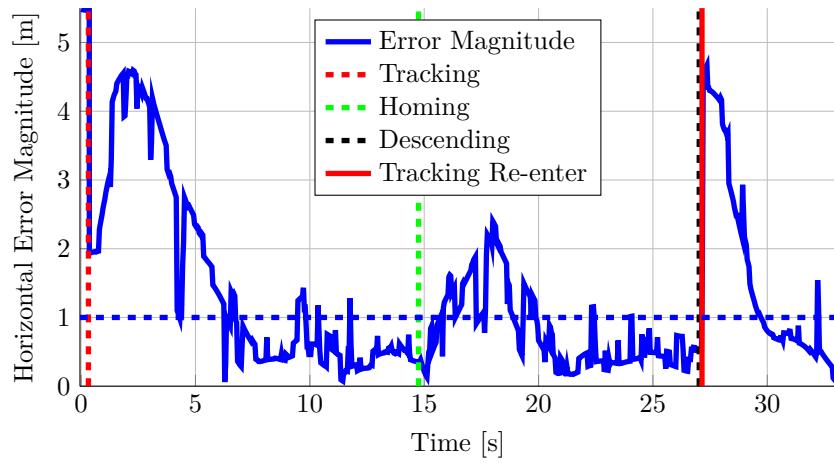
Toothed edges are again seen on the north and east reference ENU components as a result of delays in the telemetry wireless link. The state machine entered the Descending state and the vehicle started to descend, but almost immediately failed the error check due to a spike in the absolute horizontal error. The state machine reverted to and remained in the Tracking state as it should.

The absolute error for the third test is plotted in Figure 9.18. The state machine re-enters the Tracking state shortly after entering the Descending state. The error then grows to above 4 m when the reference location is moved to 3 m behind and 3 m above the landing location.

The third flight test as a whole proves that the state machine functions correctly and that the safety measures put in place operate as expected. The phenomenon caused by the delay



**Figure 9.17** – Tracking of the Virtual Platform During Third Autolanding Attempt



**Figure 9.18** – Virtual Landing Absolute Horizontal Position Error During Third Attempt

in the telecommand and telemetry link unfortunately impacted the ability of the vehicle to pass the states of the state machine under certain conditions. However, the telemetry delay

would not play a role in an actual translating landing, as the platform's position would not be sent to the vehicle through this link. Position information will be extracted directly from the Master GPS sensor, unaffected by the delay of the telemetry link. The platform's velocity and heading will still be sent to the vehicle, at a rate of 5 Hz, as this update rate was proven to not saturate the telecommand and telemetry link in this flight test.

During the actual translating platform landing, the platform velocity and heading will unfortunately still be subject to delays. The presence of delays in these measurements is not expected to impact the landing capability of the vehicle notably, as they would remain reasonably constant during the autolanding procedure.

If Align<sup>TM</sup> was not used for position measurements for the project, the telemetry link would surely have been the bottleneck preventing the achievement of the project goal. Platform position information would have been sent to the vehicle through the telemetry link, and consequently the delays would have impacted the autolanding capability of the system. This flight test proves that the vehicle would not have been able to land within a circle of 1 m diameter at 30 km/h, if all the platform information had been transmitted through the telecommand and telemetry link. The results of the other two virtual landing attempts that were not presented here, can be found in Appendix F.

## 9.6 Flight Test 4 (Coetzenburg, 29 November 2013)

The objective of the fourth flight test was to perform an autolanding on a stationary platform. The flight test took place at Coetzenburg sports fields. Wind conditions were not favorable, with constant winds of 3 m/s and gusts of 7 m/s. The wind was blowing in a north-western direction, with the vehicle and platform also aligned in a north-western direction.



**Figure 9.19** – The Vehicle Homing in on the Platform During a Stationary Autonomous Landing

In essence, no new additions were made to the firmware of the vehicle for this test. This would, however, be the first flight test where the aircraft would land on the actual platform that would be used for the final flight test. The metal trailer and platform could have unforeseen effects on the magnetometer of the vehicle and the DGPS. In Chapter 6, tests were performed to check whether the metal trailer has a prominent effect on the magnetometer of the vehicle, but the test was inconclusive. Some of the goals of this flight test were

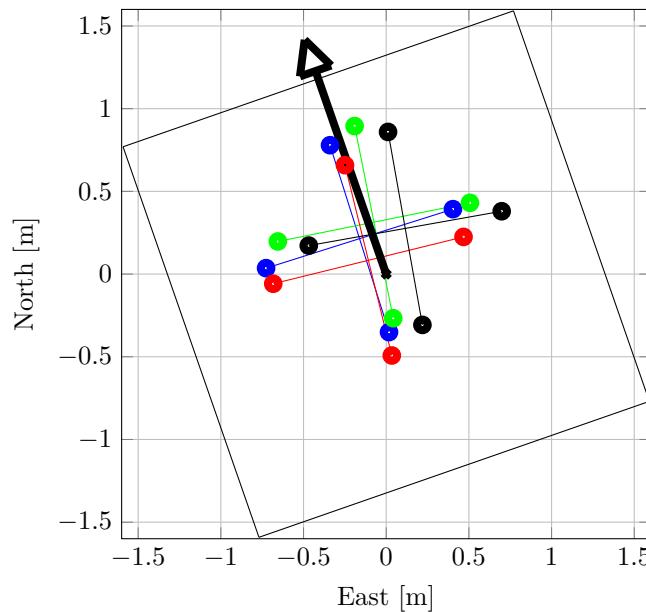
therefore to verify the acceptable functioning of the vehicle magnetometer while the vehicle is descending onto the platform, and to verify the correct functioning of the DGPS.

At the same time, the landing capability of the vehicle would be tested to determine whether the vehicle can land within the tight bounds while the platform is stationary. The test was constructed so that minimal changes had to be made to the firmware between this test and the final test, which would be the autoland of the vehicle on the actual moving platform. The test was executed five times. The vehicle passed the state machine safety checks and landed successfully four times.

As the Rover sensor would be stationary for this test, the GPS heading measurement of the Rover sensor could not be used. This posed a problem, as the antenna offsets from the landing location are calculated by use of the heading of the platform, as presented in Chapter 8. The problem was solved by placing the vehicle on the landing location in the middle of the platform prior to flight, and saving the ENU north and east offsets from the Rover sensor.

The vehicle also requires the platform's heading in order to align its own heading with the platform's heading. It was decided that the vehicle would maintain the heading it has when the Ground Station takes control of the vehicle. The safety pilot was therefore asked to align the vehicle heading with the platform heading prior to the Ground Station taking control.

Figure 9.20 shows the touchdown locations of the vehicle on the stationary platform for the four successful landings. The platform heading is indicated with the black arrow, with the origin of the arrow coinciding with the center of the platform. The platform and the vehicle are both drawn to scale.



**Figure 9.20 – The Touchdown Locations On the Platform**

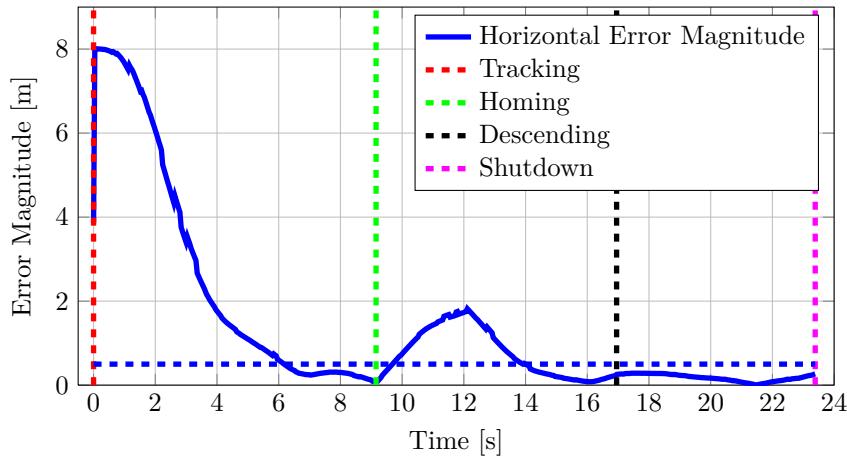
Three of the landings show that the vehicle landed with a small offset slightly northwest of the center of the platform. This is most likely due to the wind conditions at the time of the flight test. These results prove that the vehicle has the ability to land within a 1 m diameter circle of the designated landing position. Table 9.4 shows the position error at touchdown for each successful attempt.

**Table 9.4** – Stationary Platform Landing Accuracy

Landing	North Error [m]	East Error [m]	STD North [cm]	STD East [cm]	Absolute Horizontal Error [m]
Landing 1	0.214	-0.161	<1	<1	0.268
Landing 2	0.314	-0.074	<1	<1	0.323
Landing 3	0.083	-0.108	<1	<1	0.136
Landing 4	0.276	0.115	<1	<1	0.299
<b>Average</b>	<b>0.222</b>	<b>-0.057</b>	<b>NA</b>	<b>NA</b>	<b>0.257</b>

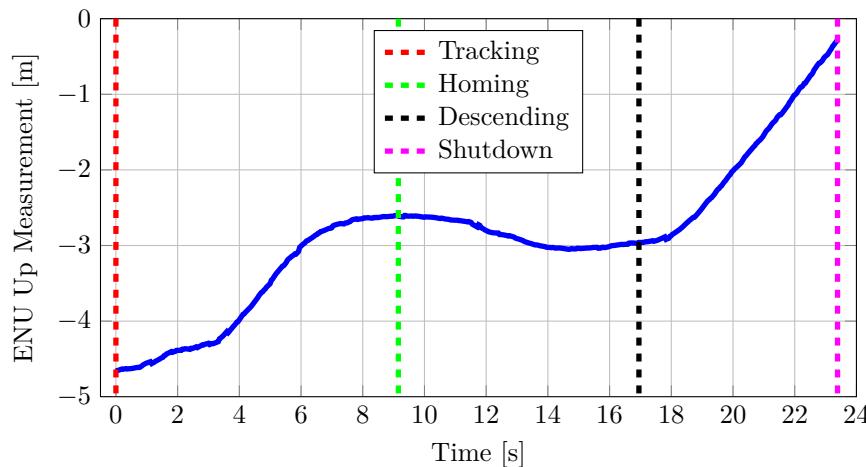
Two important parameters are also listed, namely the standard deviations of the ENU north and east components. These standard deviations are supplied by the Novatel DGPS and provide confidence in the position measurements. The standard deviations provided throughout the test were always less than 1 cm, indicating that the measurements were reliable. The vehicle touched down at worst 0.32 m from the landing location, and yielded an average landing accuracy of 0.26 m. It is therefore safe to assume that the DGPS was functioning normally.

The absolute horizontal error is investigated next. The horizontal error is plotted in Figure 9.21 for the first stationary landing.

**Figure 9.21** – Stationary Autolanding Absolute Horizontal Position Error During the First Attempt

The dashed lines are again plotted to indicate the start of the different states of the state machine. A dashed horizontal blue line is also plotted, indicating the error bounds for the test. The absolute error should not exceed 0.5 m for the state machine to advance through the states. After the Tracking state has started, the vehicle closes in on the platform, tracking the reference location behind the landing mark. Approximately 3 seconds after the absolute error curve crosses the error bound line, the Homing state is entered. The absolute error grows and then decreases as the vehicle tracks the reference which homes the vehicle in. Again, 3 seconds after the absolute error curve crosses the error bound line, the Descending state is entered. The vehicle tracks the landing location from above, staying within the error bounds until the Shutdown state.

The Up component of the ENU vector for the same attempt is plotted in Figure 9.22.



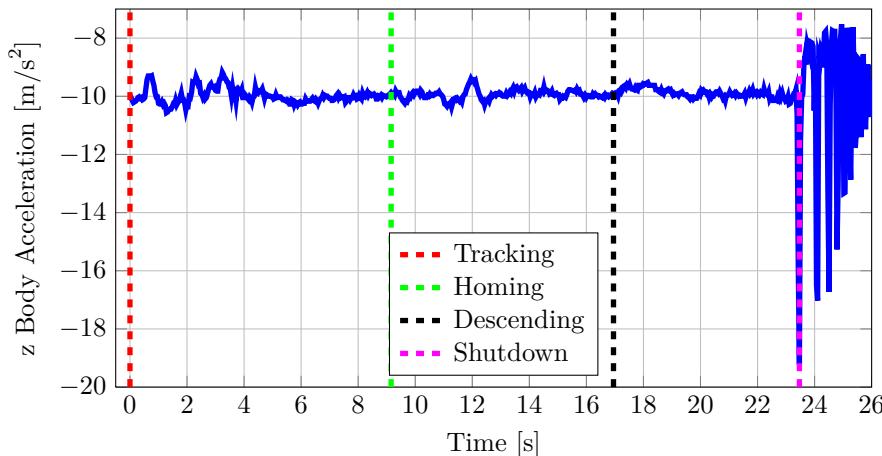
**Figure 9.22** – The ENU Up Component During The First Stationary Landing.

When the Tracking state starts, the vehicle is 4.5 m above the Rover antenna. The vehicle descends, attempting to stay 3 m above the Rover antenna during the Tracking and Homing states. As soon as the Descending state is entered, the vehicle starts to descend at 0.5 m/s, eventually making contact with the platform and entering the Shutdown state.

The specific force spike measured at touchdown plays a vital role in the state machine, notifying the vehicle that it has made contact with the platform. As stated in Chapter 7, the Shutdown state is a dormant state. Although it is a dormant state, if the state machine enters the state it proves that the vehicle would have been able to shut itself down.

In HIL simulation, and from previous work done on stationary autolanding, descending at a constant rate of 0.5 m/s provided a large enough specific force spike during impact to notify the vehicle that it has made contact with the ground. The setup in this case is different, as the platform the vehicle is landing on is not as rigid as the ground. Having a less rigid structure to land on could influence the spike.

The accelerometer measurement in the body z-axis of the vehicle, recorded during the first stationary landing, is plotted in Figure 9.23.



**Figure 9.23** – Recorded Body z-Axis Accelerometer Measurements For the First Stationary Autolanding

A maximum specific force value of  $-19.22 \text{ m/s}^2$  was recorded, which is sufficient to exceed the required  $-15 \text{ m/s}^2$  to enter the Shutdown state. Prominent underdamped behavior is observed after the initial spike. This behavior indicates that the platform along with the structure of the vehicle has a low damping ratio, dissipating energy over a relatively long time period.

The specific force spikes measured during the four successful stationary autolandings are listed in Table 9.5.

**Table 9.5** – Maximum Accelerometer Spike Measured In z Body Axis.

Landing	Spike Recorded [m/s <sup>2</sup> ]
Landing 1	-19.22
Landing 2	-22.34
Landing 3	-23.45
Landing 4	-20.29
<b>Average</b>	<b>-21.33</b>

Throughout the four successful landing attempts, the accelerometer spikes were large enough to enter the Shutdown state, yielding an average specific force spike of  $-21.33 \text{ m/s}^2$ .

Recall that the effect the trailer structure has on the magnetometer is also investigated in this test. The effect could unfortunately not be determined from the recorded heading data, as there is no ground truth to compare the heading measurement to. However, the vehicle's heading did not change notably when it came into close proximity with the platform. This would seem to indicate that the magnetometer measurement was not influenced significantly by proximity to the platform.

As a whole the flight test was a success, confirming that the platform and the metal trailer do not have an adverse impact on the operation of the DGPS or the magnetometer of the vehicle. Instrumentation of the platform can be deemed acceptable, as the offsets from the Rover GPS sensor functioned correctly to indicate to the vehicle where the center of the platform is.

The state machine functioned as expected, and shows that no modifications are necessary prior to the final flight test. The test also demonstrated that the vehicle has the capability to land with the required accuracy on a stationary landing platform, paving the way for an accurate landing on a moving platform. Given the current state of the autolanding system, the flight test provided confidence that the chance of success of landing on a moving platform is high.

## 9.7 Flight Test 5 (Wingfield Aerodrome, 4 December 2013)

The fifth flight test was the final flight test, with the goal of performing an autonomous landing on a translating platform. The test location was Wingfield Aerodrome, a decommissioned airfield located in Cape Town. The airfield was visited a few days prior to the final test, to search for a stretch of straight road that is long enough to perform the autonomous landing. In addition the road must not have any obstacles, such as trees and buildings, above or near the side of it, as the aircraft might collide with the obstacles during the flight test.



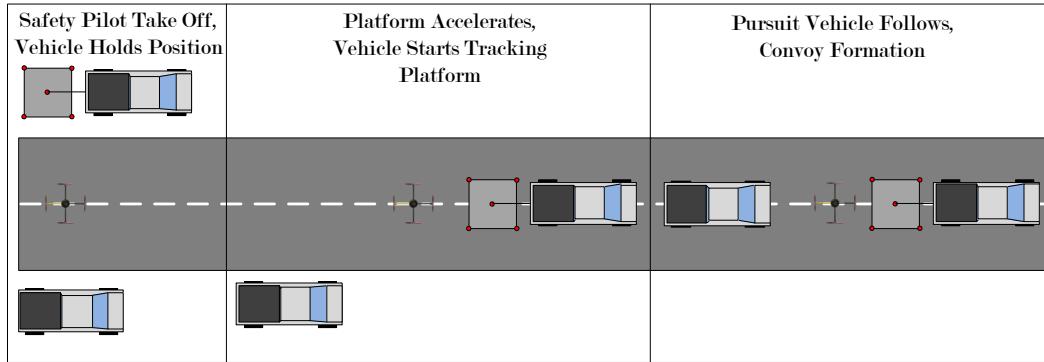
**Figure 9.24 – The Vehicle Tracking the Platform on the Trailer**

An isolated straight and level road of 400 m long was found and was available for executing the flight test. Longer straight roads were located on the airfield, but had obstacles close to the road. In HIL simulation, it was determined that conservative road lengths of 150, 250 and 490 m would be needed to perform autolandings at platform speeds of 3, 5 and 7 m/s respectively. These lengths are required to provide enough time for the state machine to advance through all the states, if all goes as planned, with some room for error. This posed a problem, as the available road at Wingfield is only 400 m long, forcing a deviation from the proposed sequence of events in Section 3.3 for the flight demonstration test.

On the same day as the inspection of the airfield, the DGPS was tested with the platform and the vehicle stationary. This was done to ensure that DGPS communication would not be disrupted by any other communication sources in the vicinity. The results were satisfactory and the go-ahead was given to perform the flight test. In the course of the following days, the weather was monitored and the 4th of December 2013 was chosen for performing the final flight test.

Recall from Section 3.3 how the flight demonstration test was planned. For the sake of convenience, the flight demonstration strategy will be briefly repeated here. Two motor vehicles assisted in the test: a vehicle towing the platform and a pursuit vehicle. The vehicle towing the platform accommodated a driver, the Ground Station and the Ground Station Officer. The pursuit vehicle accommodated a driver, the safety pilot and the safety officer. The safety officer communicated the expected behavior of the aircraft to the safety pilot during the test. Figure 9.25 illustrates how all the vehicles were coordinated during the test.

The test begins with both the pursuit and towing vehicles stationary next to the road, while the safety pilot pilots the vehicle off the ground. The Ground Station takes control and commands the vehicle to hover and hold its position. The towing vehicle starts accelerating up to the speed at which the test is to be performed. Once the platform is translating at the intended speed, the Ground Station Officer commands the aircraft to start the autolanding procedure. When the aircraft starts accelerating, the pursuit vehicle follows to monitor the aircraft. The vehicles stay in this convoy formation until the autolanding procedure is completed.



**Figure 9.25** – Coordination Of All Vehicles During the Flight Demonstration Test

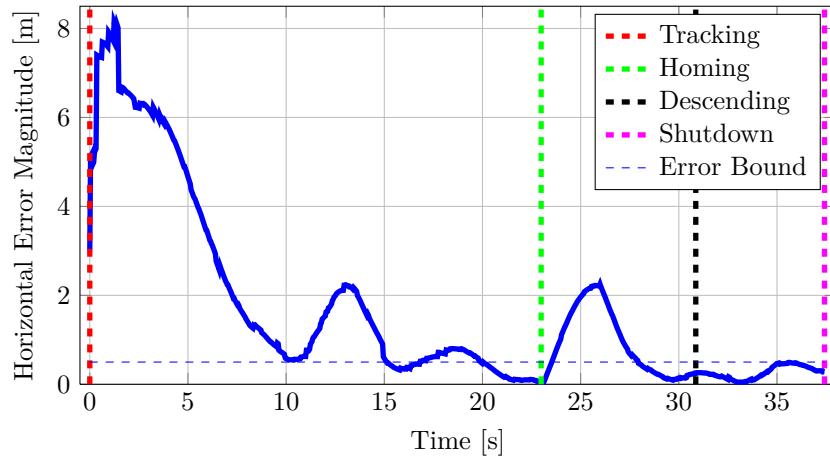
The towing vehicle used to tow the trailer was a diesel-powered Toyota Hilux double cab pick-up truck. The Hilux was conveniently able to maintain a speed of roughly  $10 \text{ km/h}$  while the engine was idling in first gear. Similarly, the Hilux could idle in second, third and fourth gear at  $20, 30$  and  $40 \text{ km/h}$  respectively. This allowed the platform to translate at constant speeds, eliminating the need for the driver to regulate platform speed.

Four autonomous translating landings were attempted, with the platform translating at  $10, 20, 30$  and  $40 \text{ km/h}$ . This is equivalent to  $2.8, 5.6, 8.3$  and  $11.1 \text{ m/s}$  respectively. This posed a problem, as the available road at Wingfield is only  $400 \text{ m}$  long, forcing a deviation from the planned sequence of events shown in Figure 9.25 for the  $30$  and  $40 \text{ km/h}$  landing attempts. Instead of the vehicle remaining stationary while the towing vehicle accelerates up to speed, the aircraft and the towing vehicle will start to accelerate at approximately the same time. This will shorten the distance the vehicle needs to travel to catch up to the platform, thus requiring a shorter road.

### 9.7.1 Autolanding Results At 10, 20 and 30 km/h

Fortunately, wind conditions were favorable, with no constant wind and maximum wind gusts of only  $1 \text{ m/s}$  being present. No new additions were made to the firmware or hardware from the preceding flight test.

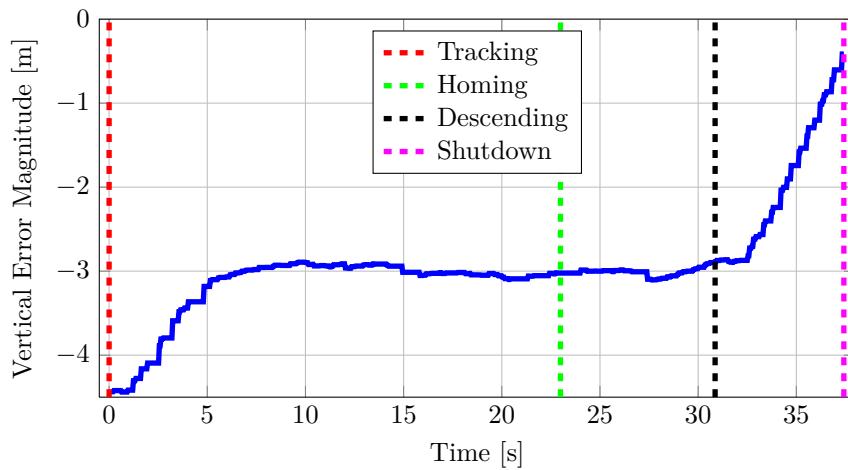
The vehicle successfully landed on the translating platform at speeds of  $10, 20$  and  $30 \text{ km/h}$ . The figures represented in this section are from the autolanding performed at  $30 \text{ km/h}$  or  $8.3 \text{ m/s}$ . The results for the other three attempts can be found in Appendix F. The absolute horizontal error recorded during the flight test is shown in Figure 9.26.



**Figure 9.26** – Absolute Horizontal Position Error During 30 km/h Autolanding

The state machine lines are plotted, indicating the start of each state. The error bound is also plotted, indicating the maximum value that the absolute error is permitted to be in order to land within a 1 m diameter circle. The vehicle starts approximately 8 m behind the platform when the Tracking state starts. The error decreases as the vehicle starts to accelerate, catching up to the platform. Twenty seconds into the test, the error is less than 0.5 m. Three seconds later, the Homing state is entered, where the error again grows as the vehicle has to accelerate to home in to a position above the center of the moving platform. At 28 seconds, the absolute error is again less than 0.5 m, entering the Descending state 3 seconds later. The error remains below 0.5 m for the entire Descending state, entering the Shutdown state 37 seconds from the start of the landing procedure.

The vertical position error recorded during the flight test is shown in Figure 9.27. The ENU Up component is plotted, effectively indicating the vertical error between the vehicle and the platform.



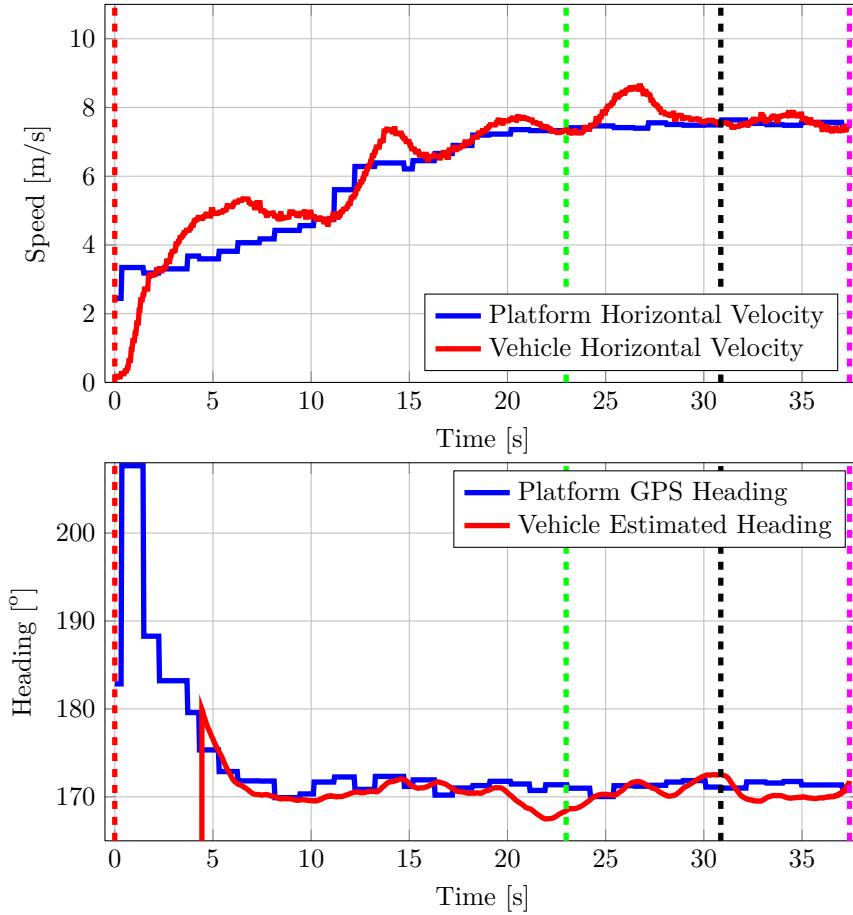
**Figure 9.27** – The Vertical Error While Landing At 30 km/h.

The landing procedure starts with the vehicle 4.2 m above the landing location. The vehicle descends during the Tracking state, maintaining a height of 3 m above the landing location until the Descending state is entered. During the Descending state, the vehicle descends at a rate of 0.5 m/s, finally making contact with the platform.

The top figure of Figure 9.28 displays the measured velocities of the platform and the vehicle throughout the landing procedure. The bottom figure shows the heading measurements of both the platform and the vehicle.

It can be seen that the platform's velocity has not yet reached the target speed of 8.3 m/s when the landing procedure starts. As explained previously, this was done to avoid running out of usable road. During the Tracking state, the vehicle is already close to the platform, thus not requiring a high velocity to catch up to the platform. Eighteen seconds into the landing procedure, the vehicle matches the velocity of the platform. After the Homing state is entered, the vehicle speed increases slightly while the vehicle homes in, matching the velocity of the platform again shortly afterwards. The Descending state starts and the vehicle approximately matches the velocity of the platform until the Shutdown state.

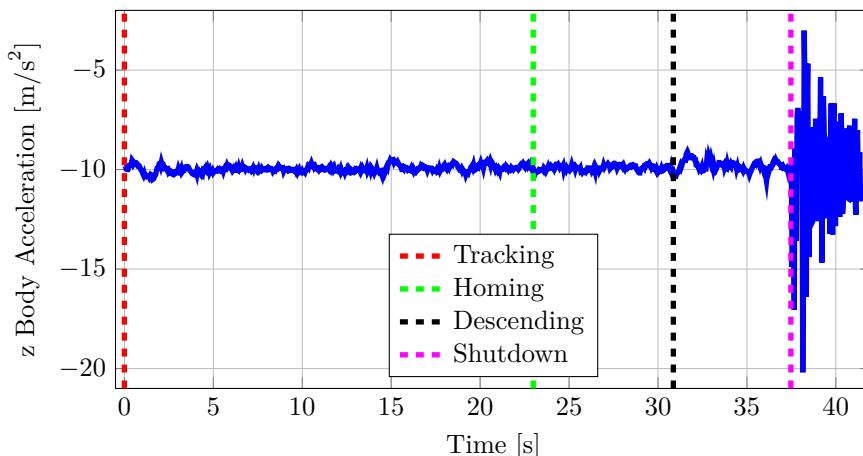
The platform heading can be seen changing from 210° to approximately 170° during the first 10 seconds of the landing procedure. This is ascribable to the towing vehicle aligning itself with the center of the straight road. After the platform is aligned with the direction of the road, the heading remains constant. Five seconds into the autolanding procedure, the



**Figure 9.28** – Platform and Vehicle Velocity and Heading During 30 km/h Autolanding

vehicle aligns its heading with the platform's heading. The vehicle then tracks the heading of the platform until it lands on the platform.

Finally, the recorded measurements of the accelerometer aligned in the body z-axis of the vehicle are considered. The measurements are visually inspected to check whether this sensor can be used to adequately determine if the vehicle has touched down on the platform. The time history of the vehicle's body z-axis accelerometer measurement is plotted in Figure 9.29.



**Figure 9.29** – Body z-Axis Specific Force During 30 km/h Autolanding

The accelerometer measurement remains close to the specific force induced by gravity on the sensor for the entire flight, until contact is made with the platform. The first spike that is measured is  $-17 \text{ m/s}^2$ , which exceeds the required  $-15 \text{ m/s}^2$  to enter the Shutdown state. Several specific force spikes are observed after the initial spike, due to the vehicle and the platform oscillating after the safety pilot shuts the rotors down. Spikes continue to be measured as bumps in the road disturb the trailer, which in turn disturbs the platform and the vehicle.

### 9.7.2 Autolanding Results At 40 km/h

As mentioned, an autolanding attempt was also performed while the trailer was moving  $40 \text{ km/h}$ . However, this attempt was not successful. The reason for this failed attempt is investigated.

The autolanding procedure was appearing to be going as planned, with the state machine advancing through all its states. The aircraft successfully passed all the error checks of the Tracking and Homing states. However, when the aircraft started to descend it became clear that it would touch down short of the moving platform. Figure 9.30 is a photograph of the vehicle missing the landing location upon making contact with the platform.



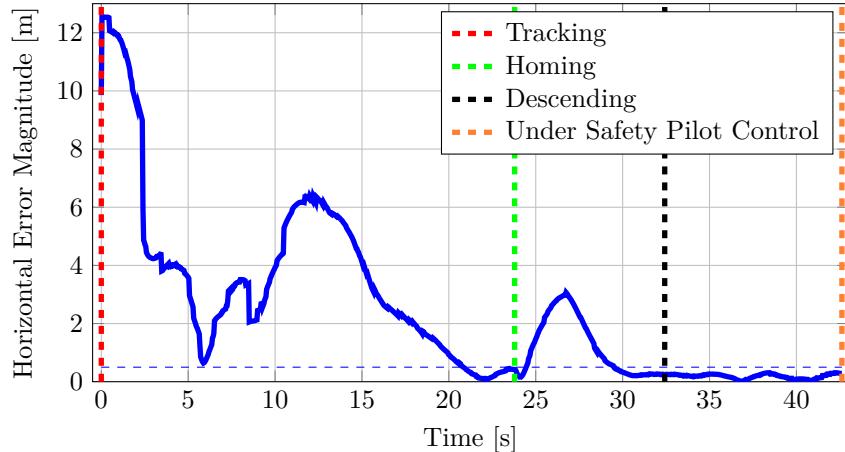
**Figure 9.30** – The Vehicle Missing the Platform During the 40 km/h Landing Attempt

Consequently, the aircraft would have fallen off of the platform. This forced the safety pilot to immediately take control of the aircraft, as the aircraft did not abort the landing automatically. After the test, the logged data was thoroughly investigated, and the apparent cause was found.

While the test was in progress, two of the 16 satellites visible to the Novatel sensors had been lost. This had reduced the accuracy of the ENU vector's relative position measurements, which led to the vehicle misinterpreting the landing location.

To prove that the vehicle believed it had remained within the error bounds, the measured horizontal position error recorded during the flight test is inspected, and is shown in Figure 9.31.

The vehicle passes all the states of the state machine and remains within the error bounds during the Descending state. An orange dashed line indicates the instant the safety pilot took control of the vehicle, corresponding to the moment shown in Figure 9.30. This leads to

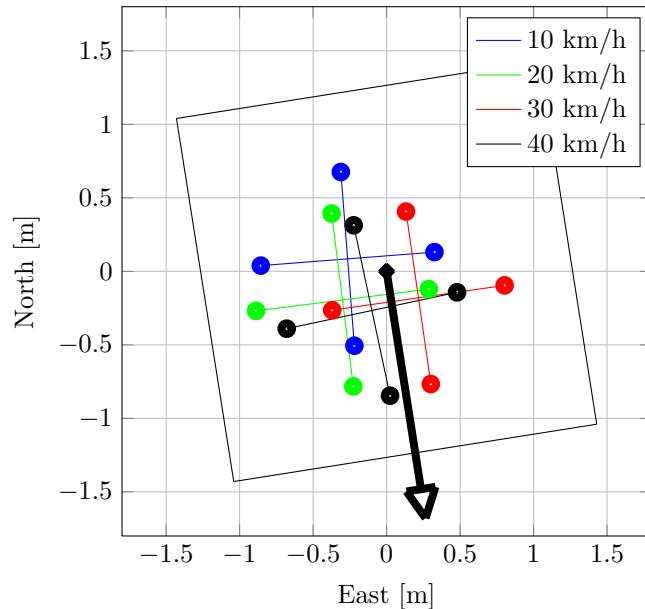


**Figure 9.31** – Absolute Horizontal Position Error During 40 km/h Autolanding Attempt

the question whether the vehicle would have been able to land at 40 *km/h* on the designated landing mark if DGPS accuracy had not been compromised.

### 9.7.3 Final Flight Test Summary

Shown in Figure 9.32, drawn to scale, are the touchdown locations on the platform for all four autolanding attempts. The 40 km/h attempt has been included, to illustrate how accurately the vehicle believed it landed relative to the designated landing location.



**Figure 9.32** – Actual Landing Locations for 10, 20 and 30 km/h Autolandings, As Well As Believed Landing Location for 40 km/h Autolanding Attempt

There does not seem to be a correlation between the magnitude or direction of the landing error and the speed of the moving platform. The absolute measured horizontal error and

the standard deviations reported by the Novatel DGPS are shown in Table 9.6 for all four attempts. The vehicle believed it landed the nearest to the designated landing point during the  $40 \text{ km/h}$  attempt.

**Table 9.6** – Translating Platform Landing Accuracy

Landing	North Error [m]	East Error [m]	STD North [cm]	STD East [cm]	Absolute Horizontal Error [m]
Landing 1 (10 km/h)	0.085	-0.265	<1	<1	0.279
Landing 2 (20 km/h)	-0.195	-0.300	<1	<1	0.359
Landing 3 (30 km/h)	-0.180	0.216	<1	<1	0.282
Landing 4 (40 km/h)	-0.266	-0.100	27	23	0.285
<b>Average</b>	<b>-0.139</b>	<b>-0.112</b>	<b>NA</b>	<b>NA</b>	<b>0.301</b>

On average, the vehicle touched down within  $0.3 \text{ m}$  of the landing location. The reported standard deviation of the ENU vector's north and east components remained less than  $1 \text{ cm}$  for the first three autolanding attempts. During the  $40 \text{ km/h}$  attempt, the standard deviation increased to  $27 \text{ cm}$  north and  $23 \text{ cm}$  east, indicating that the ENU measurements were less reliable during this last attempt. In future, a safety check must be included to monitor the reported standard deviations, commanding the vehicle to abort the autolanding procedure if measurement accuracy becomes unacceptable.

The final flight test can be declared a success. The results illustrate that the vehicle successfully landed on the translating platform at  $30 \text{ km/h}$ .

## 9.8 Discussion

The first flight test proved that the control system designed in Chapter 5 functioned as simulated and increased the bandwidth of the velocity and position controllers. The second flight test demonstrated that the vehicle is capable of accelerating and decelerating demandingly, which was required at the final flight test. The third flight test showed that the state machine, designed in Chapter 7, functions as expected. The fourth flight test indicated that the vehicle has the capability to autonomously land on a stationary platform within a  $1 \text{ m}$  diameter circle. The final flight test demonstrated that the vehicle has the capability to land on a translating platform at a speed of  $30 \text{ km/h}$ , within a  $1 \text{ m}$  diameter circle.



**Figure 9.33** – The Vehicle Tracking the Platform

Overall, the flight test campaign can be described as a success, as it experimentally verified and practically demonstrated with flight tests that the autolanding system meets the project requirements and that the project goals have been achieved.

## Chapter 10

# Summary and Recommendations

This final chapter provides a summary of all the work performed during the course of the project, followed by the main conclusions that were reached. The chapter closes with recommendations for improving the system to support future research.

### 10.1 Summary and Conclusions

This thesis presented the successful design, implementation and practical testing of an automated landing system that autonomously landed a quadrotor Unmanned Aerial Vehicle on a moving platform. The autolanding system was practically tested and autonomously landed the quadrotor within a 1  $m$  diameter circle of the designated landing mark on a platform translating at 30  $km/h$ . The problem of designing an autolanding system capable of doing this was solved by upgrading the horizontal flight control system, modifying the Differential Global Positioning System (DGPS) and designing a state machine that safely guided the quadrotor through the landing procedure. A brief summary of the work presented and performed during each chapter is given here.

In the second chapter, an overview was given of the pre-existing quadrotor unmanned aerial system. The system consists of three constituents, namely the aircraft, the Ground Station and the safety pilot. The pre-existing quadrotor unmanned aerial system had full autonomous flight capability and a centimeter-level accuracy *absolute* position measurement DGPS. The DGPS allowed for centimeter-level accurate position sensing of the vehicle's Global Positioning System (GPS) sensor, but required that the second GPS sensor remain *stationary*.

In the third chapter, the strategy for attaining the project goal was devised. An autolanding state machine that has four states was proposed to guide the aircraft through the autolanding procedure. The four states are named Tracking, Homing, Descending and Shutdown. The position sensing strategy of the quadrotor and the platform was also presented. The strategy entailed modifying the pre-existing DGPS of the autolanding system, to provide centimeter-level accuracy *relative* position measurements instead of absolute position measurements. The modification to the DGPS would allow the second GPS sensor to be placed on the *moving* landing pad. The strategy for executing the final flight demonstration was then explained, and proposed that a platform mounted to a trailer being towed by a motor vehicle be used to mimic a moving ship deck.

The fourth chapter presents the pre-existing mathematical models used in the thesis, which were incorporated in non-linear simulations. The axis systems used were firstly defined, and the corresponding notation to describe the aircraft's movement was introduced. The kinetic and kinematic relations were then presented, followed by the equations describing the forces and moments that act on the quadrotor due to aerodynamics, gravity and the actuators. Lastly, the sensor models and wind models were presented.

In the fifth chapter, the entire flight control system was analyzed and the horizontal flight control system was re-designed. Using linear vehicle models, the flight control systems responsible for the horizontal translation, vertical translation and yaw rotation control were analyzed separately and critically evaluated for the task of autolanding on a moving platform. The horizontal velocity controllers were found to be conservatively designed, and the architecture of the horizontal position controllers were not capable of following a moving platform with zero steady-state error. A complete re-design of the horizontal control system followed, which doubled the bandwidth of the horizontal position controllers and augmented the horizontal position controllers with feed-forward velocity references and integrators to compensate for velocity measurement errors. The chapter closed by performing non-linear Software-In-the-Loop simulations to validate the performance of the re-designed controllers, and to determine their effect on the vertical and yaw flight control systems. The re-designed controllers were found to perform as designed and did not adversely affect the other flight control systems.

In the sixth chapter, the position sensing method of the autolanding system was implemented and practically verified through experiments. The pre-existing DGPS was modified to provide centimeter-level accurate relative position measurements between the platform and the vehicle instead of absolute position measurements. This modification required sourcing and installation of a wireless communication link that is dedicated to DGPS communication. A series of practical tests then followed, and confirmed that centimeter-level accurate relative position measurements are reliably obtainable. The quality of the dedicated DGPS communication link allowed for relative position updates at a rate of 5 Hz, which was deemed adequate for the project. The update rate was artificially increased by integrating the velocity difference of the vehicle and the platform with the elapsed time between updates. An algorithm was also implemented that would indicate if the DGPS has been compromised in the event of a DGPS communication failure.

The seventh chapter described the proposed autolanding state machine in detail. The state machine performs safety checks and guides the vehicle through the landing procedure to increase the chances of landing within the admissible error bounds of the project. The state machine has four states, namely Tracking, Homing, Descending and Shutdown. During the Tracking state, the vehicle tracks the platform from 3 m behind and 3 m above so that any transients from approaching the platform are diminished at a safe distance. In the Homing state, the vehicle slowly homes in to 3 m directly above the landing location. In the Descending state, the vehicle descends at a constant rate of 0.5 m/s onto the platform. When the vehicle makes contact with the platform, it will measure an specific force spike, which will let the state machine advance to the Shutdown state. In the Shutdown state, the flight control system is disarmed and the rotors will stop rotating.

In the eighth chapter, non-linear Hardware-In-the-Loop (HIL) simulations were performed. The pre-existing HIL system was augmented with the re-designed horizontal controllers, the autolanding state machine and modelling of the modified DGPS. The conditions in which the HIL simulations would take place were then determined, so that each simulation could be compared. Eighteen consecutive simulations were performed in wind and no-wind conditions while the platform was moving at 3 m/s, 5 m/s and 7 m/s. The simulation results indicated that the vehicle touched down on the moving platform with an average position error of 0.144 m, 0.091 m and 0.288 m respectively for platform speeds of 3 m/s, 5 m/s and 7 m/s, while wind with a speed of 5 m/s was present. The results from the simulations were deemed acceptable.

The ninth chapter documents the execution and results of the flight test campaign that was conducted to test each addition to the autolanding system. The campaign consisted of five flight tests, with the goal of the first flight test being to test the re-designed horizontal controllers. The vehicle was commanded to follow horizontal position step references, and

the results showed that the position control loop's bandwidth was significantly increased. However, somewhat increased oscillations were observed in the transient response of the pitch and roll angles of the aircraft. Time constraints did not allow for the source to be isolated, but the phenomenon did not appear to be a stability problem, but more likely the result of a disturbance. It was decided that flight testing would continue. The second flight test tested the stability of the aircraft at the acceleration extremes that it must be able to endure during an autolanding. The aircraft was aggressively accelerated up to a top speed of  $10.5 \text{ m/s}$  and then immediately aggressively decelerated by performing an emergency stop. The aircraft remained stable under the demanding acceleration conditions and did not saturate its actuators.

Three flight tests then followed with all three performing autolandings using the autolanding state machine. The state machine was safely tested by first landing in mid-air on a virtual platform. The state machine successfully advanced through all states and showed that the safety checks that were put in place also functioned as intended. In the next flight test, the aircraft successfully auto-landed on a stationary platform. The autolanding system proved that it could land on a stationary platform within the tight error bounds required for the project. Five stationary autolandings were attempted, with four resulting in successful landings, and one automatically (and correctly) aborted landing. The average position error with which the aircraft touched down was  $0.257 \text{ m}$ . The final flight test followed, where the vehicle attempted to perform autolandings on the moving platform at speeds of  $10 \text{ km/h}$ ,  $20 \text{ km/h}$ ,  $30 \text{ km/h}$  and  $40 \text{ km/h}$ . The aircraft successfully landed on the translating platform at speeds of  $10 \text{ km/h}$ ,  $20 \text{ km/h}$  and  $30 \text{ km/h}$ , with an average position error at touchdown of  $0.307 \text{ m}$ . The autolanding attempt while the platform was traveling at  $40 \text{ km/h}$  passed all the safety checks of the autolanding state machine, and the vehicle touched down on the moving platform, but with a position error that was larger than the admissible position error. Upon inspection of the logged flight data, it was found that the DGPS accuracy was compromised during the autolanding attempt. The state machine therefore thought it was guiding the vehicle to the correct landing location, but in reality it was not. Subsequently, an additional safety check was proposed that would check the reported accuracy of the DGPS measurements, and would abort the landing if it exceeded a safe limit.

## 10.2 Recommendations

The following recommendations are made for improving the system to support future research:

- The non-linear model of the vehicle does not include accurate modelling of the aerodynamics of the rotors and the chassis of the quadrotor. A more detailed model is required to be able to more accurately represent the aircraft during non-linear simulation, especially when it is not near hover flight. An in-depth study on the aerodynamics involved is recommended.
- A more robust wireless communication link is required between the DGPS sensors. For this project, the robustness and quality of the communication link was sufficient to provide accurate relative position measurements at an update rate of  $5 \text{ Hz}$ . The current communication link does not utilize frequency hopping and functions on a single channel. For future research, a link that does utilize frequency hopping is recommended, which will lead to better communication quality and thus a higher measurement update rate.
- A safety check must be implemented in the autolanding state machine that actively checks the reported position measurement accuracy from the GPS sensors. As was seen during the autolanding attempt while the platform was moving at a speed of  $40 \text{ km/h}$ ,

the reported accuracy of the relative position measurements between the platform and the vehicle decreased. This was not actively checked by the state machine, and the vehicle consequently passed all state machine safety checks. However, in reality the aircraft did not land within the specified position bounds. The safety check would then be used by the autolanding state machine to abort the landing procedure.

- It is recommended that the quadrotor be equipped with a newer model Novatel GPS sensor, as the current GPS sensor is end-of-life hardware and is no longer supported by Novatel. The implication for the project was that the Master and Rover sensors had to be programmed differently, and anomalous behavior of the DGPS was noted at times. On several occasions, the DGPS would fail to get initial lock on satellites in the vicinity. Only after a hard reset of both the Master and Rover GPS sensors was DGPS lock achieved.
- Some constituents of the avionics of the vehicle require more programming space. The current microprocessor used for flight control, state estimation and navigation is utilizing all of its program memory. Consequently, there is not enough available program memory left to add additional functionality to the autolanding state machine.

## Appendix A

# Additional Technical Information

In this appendix, technical information on some of the hardware of the quadrotor UAV is given.

### A.1 Power System

The technical details of the power system components on the quadrotor can be seen in Table A.1.

**Table A.1 – Quadrotor Power System Technical Details**

Component	Description.
Motors	Hyperion ZS4025 16 Turn 320kV.
ESC	Castle Creations Phoenix ICE HV 80A 50V.
Batteries	Hyperion 3300 - 4S 35C Lipo.
Propellers	JZ Zinger 20x6 Wood.

The motors are made by Hyperion, and are capable of a maximum power output of 2200 W. The rotors are JZ Zinger propellers, of which each has a diameter of 20 inches and a pitch of 6 inches. They are made from beech wood. The batteries used by the quadrotor are also made by Hyperion. Eight batteries in total are used, each with a capacity of 3300 *mah*, a nominal voltage of 14.8 V and they are capable of continuously discharging at a rate of 115 A. The electronic speed controllers which drive the motors are made by Castle Creations, and are capable of continuously delivering 50 A to the motors at a voltage of 50 V.

### A.2 IMU

The IMU is made by Analog Devices and is used to measure the inertial specific acceleration that the quadrotor experiences, as well as its angular rates. Refer to Table A.2 for more information regarding the 3-axis gyroscope and accelerometer.

**Table A.2 – IMU Sensor Information**

<b>Component</b>	<b>Range</b>	<b>Resolution</b>
3-axis Gyroscope	$\pm 75^\circ/s, \pm 150^\circ/s, \pm 300^\circ/s$	14-bit
3-axis Accelerometer	$\pm 10 g$	14-bit
Bandwidth	350 Hz	NA
Operating Temperature	-40°C to +85°C	NA

### A.3 GPS

The GPS used in the UAV system is manufactured by Novatel. A GPS sensor is used by both the platform and the quadrotor for position and velocity measurements. Refer to Table A.3 for more specific details on the components.

**Table A.3 – DGPS Sensor Information**

<b>Component</b>	<b>Name</b>	<b>Model</b>
Rover Receiver	Novatel OEM-V1	OEM-V1G
Rover Antenna	Novatel Compact GNSS Antenna	ANT-A71GLA4
Master Receiver	Novatel Propak 3	ProPak-V3
Master Antenna	Novatel High-Performance GNSS Antenna	GPS-701-GG

The quadrotor uses a Novatel OEM-V1 sensor with a compact GNSS antenna to save weight. The platform uses the Novatel Propak-3 with a high-performance GNSS antenna, as weight is not a factor. The binary structure of the Novatel packets relevant to the project can be seen in the following tables.

**Table A.4 – BestPos Packet Binary Format**

<b>Field</b>	<b>Data Description</b>	<b>Format</b>	<b>Binary Bytes</b>	<b>Binary Offset</b>
1	BestPos Log Header	-	H	0
2	Solution Status	Enum	4	H
3	Position Type	Enum	4	H+4
4	Latitude	Double	8	H+8
5	Longitude	Double	8	H+16
6	Height Above Mean Sea Level [m]	Double	8	H+24
7	Undulation [m]	Float	4	H+32
8	Datum ID Number	Enum	4	H+36
9	Latitude Standard Deviation	Float	4	H+40
10	Longitude Standard Deviation	Float	4	H+44
11	Height Standard Deviation	Float	4	H+48
12	Base Station ID	Char[4]	4	H+52
13	Differential Age In Seconds	Float	4	H+56
14	Solution Age In Seconds	Float	4	H+60
15	Number Of Satellites Tracked	Uchar	1	H+64
16	Number Of Satellites Used	Uchar	1	H+65
17	Number Of GPS + GLONASS L1 Used	Uchar	1	H+66
18	Number Of GPS + GLONASS L1 and L2 Used	Uchar	1	H+67
19	Reserved	Uchar	1	H+68
20	Extended Solution Status	Hex	1	H+69
21	Reserved	Hex	1	H+70
22	Signals Used Mask	Hex	1	H+71
23	32-bit CRC	Hex	1	H+72
24	Sentence Terminator (ASCII Only)	-	-	-

The first packet seen in Table A.4 is the normal GPS position measurement packet called BestPos. The packet is a synchronous packet that is available irrespective of DGPS communication, and can provide position measurement updates at a rate of 10 Hz.

The second packet relevant to the project is the normal GPS velocity measurement packet that can be seen in Table A.5. The packet is also a synchronous packet which is always available at an update rate of 10 Hz.

**Table A.5 – BestVel Packet Binary Format**

Field	Data Description	Format	Binary Bytes	Binary Offset
1	BestVel Log Header	-	H	0
2	Solution Status	Enum	4	H
3	Velocity Type	Enum	4	H+4
4	Latency Of The Velocity Time Tag	Float	4	H+8
5	Differential Age In Seconds	Float	4	H+12
6	Horizontal Ground Speed [m/s]	Double	8	H+16
7	Heading w.r.t True North	Double	8	H+24
8	Vertical Speed [m/s]	Double	8	H+32
9	Reserved	Float	4	H+40
10	32-bit CRC	Hex	4	H+44
11	Sentence Terminator (ASCII Only)	-	-	-

The third packet, which is only available during Align<sup>TM</sup> DGPS operation, is shown in Table A.6. This is the ENU vector, and it is an asynchronous packet that is only available when there is communication of sufficient quality between the Master and Rover GPS sensors. The update rate of the vector varies between 0 Hz and 10 Hz, depending on the quality of the communication link between the Master and Rover GPS sensors.

**Table A.6 – ENU Packet Binary Format**

Field	Data Description	Format	Binary Bytes	Binary Offset
1	ENU Log Header	-	H	0
2	Solution Status	Enum	4	H
3	Position Type	Enum	4	H+4
4	East Baseline [m]	Double	8	H+8
5	North Baseline [m]	Double	8	H+16
6	Up Baseline [m]	Double	8	H+24
7	East Baseline Standard Deviation [m]	Float	4	H+32
8	North Baseline Standard Deviation [m]	FLoat	4	H+36
9	Up Baseline Standard Deviation [m]	Float	4	H+40
10	Rover Receiver ID	Char[4]	4	H+44
11	Master Receiver ID	Char[4]	4	H+48
12	Number Of Satellites Tracked	Uchar	1	H+52
13	Number Of Satellites Used	Uchar	1	H+53
14	Number Of Satellites Above Mask Angle	Uchar	1	H+54
15	Number Of Satellites Above Mask Angle WItL L2	Uchar	1	H+55
16	Reserved	Uchar	1	H+56
17	Extended Solution Status	Hex	1	H+57
18	Reserved	Hex	1	H+58
19	Signals Used Mask	Hex	1	H+59
20	32-bit CRC	Hex	1	H+60
21	Sentence Terminator (ASCII Only)	-	-	-

Another packet which is only available during Align<sup>TM</sup> DGPS operation is MasterPos. MasterPos is also an asynchronous packet with an update rate that can vary between 0 Hz and 10 Hz, depending on the quality of the communication link between the Master and Rover GPS sensors.

**Table A.7** – MasterPos Packet Binary Format

Field	Data Description	Format	Binary Bytes	Binary Offset
1	MasterPos Log Header	-	H	0
2	Solution Status	Enum	4	H
3	Position Type	Enum	4	H+4
4	Master WGS84 Latitude	Double	8	H+8
5	Master WGS84 Longitude	Double	8	H+16
6	Master Height Above Mean Sea Level [m]	Double	8	H+24
7	Undulation [m]	Float	4	H+32
8	Datum ID Number	Enum	4	H+36
9	Latitude Standard Deviation [m]	Float	4	H+40
10	Longitude Standard Deviation [m]	Float	4	H+44
11	Height Standard Deviation [m]	Float	4	H+48
12	Receiver ID	Char[4]	4	H+52
13	Reserved	Float	4	H+56
14	Reserved	Float	4	H+60
15	Number Of Satellites Tracked	Uchar	1	H+64
16	Number Of Satellites Used	Uchar	1	H+65
17	Number Of Satellites Above Mask Angle	Uchar	1	H+66
18	Number Of Satellites Above Mask Angle WIth L2	Uchar	1	H+67
19	Reserved	Uchar	1	H+68
20	Reserved	Uchar	1	H+69
21	Reserved	Uchar	1	H+70
22	Reserved	Uchar	1	H+71
23	32-bit CRC	Hex	1	H+72
24	Sentence Terminator (ASCII Only)	-	-	-

The final packet which is only available during Align<sup>TM</sup> DGPS operation is RoverPos. RoverPos is an asynchronous packet with an update rate that can vary between 0 Hz and 10 Hz, depending on the quality of the communication link between the Master and Rover GPS sensors.

**Table A.8** – RoverPos Packet Binary Format

Field	Data Description	Format	Binary Bytes	Binary Offset
1	RoverPos Log Header	-	H	0
2	Solution Status	Enum	4	H
3	Position Type	Enum	4	H+4

4	Rover WGS84 Latitude	Double	8	H+8
5	Rover WGS84 Longitude	Double	8	H+16
6	Rover Height Above Mean Sea Level [m]	Double	8	H+24
7	Undulation [m]	Float	4	H+32
8	Datum ID Number	Enum	4	H+36
9	Latitude Standard Deviation [m]	Float	4	H+40
10	Longitude Standard Deviation [m]	Float	4	H+44
11	Height Standard Deviation [m]	Float	4	H+48
12	Receiver ID	Char[4]	4	H+52
13	Reserved	Float	4	H+56
14	Reserved	Float	4	H+60
15	Number Of Satellites Tracked	Uchar	1	H+64
16	Number Of Satellites Used	Uchar	1	H+65
17	Number Of Satellites Above Mask Angle	Uchar	1	H+66
18	Number Of Satellites Above Mask Angle WIth L2	Uchar	1	H+67
19	Reserved	Uchar	1	H+68
20	Reserved	Uchar	1	H+69
21	Reserved	Uchar	1	H+70
22	Reserved	Uchar	1	H+71
23	32-bit CRC	Hex	1	H+72
24	Sentence Terminator (ASCII Only)	-	-	-

## Appendix B

# Additional Modelling Information

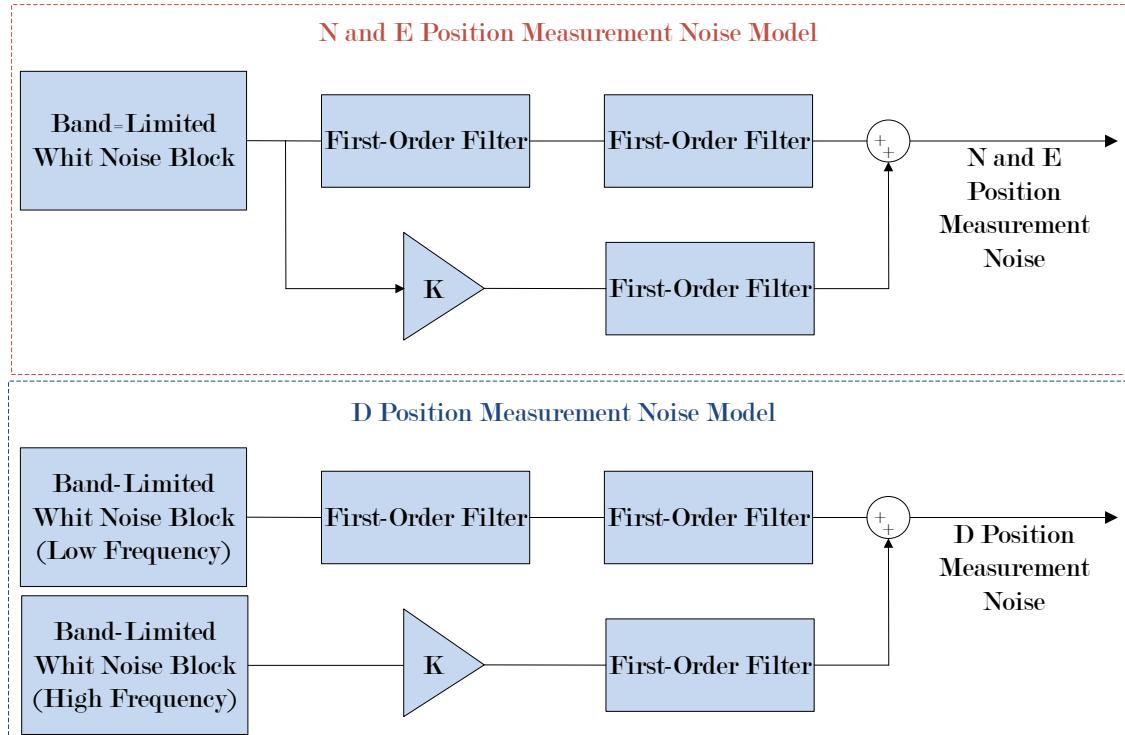
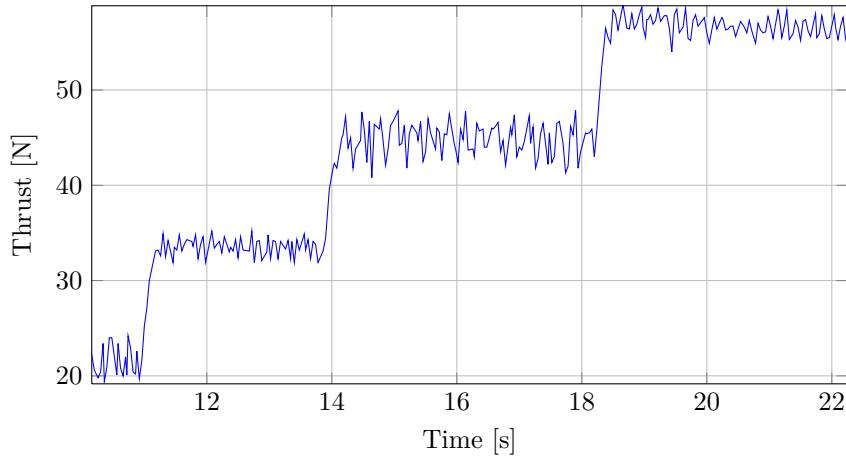
This appendix provides additional information on modelling of the quadrotor. The parameters used to simulate sensor noise during non-linear simulation are presented first, followed by the step responses from the aircraft's rotors. These parameters were used during Software-In-the-Loop (SIL) and Hardware-In-the-Loop (HIL) simulation. The Band-Limited White Noise (BLWN) block Power Spectral Density (PSD) parameter and the sample times are summarized in Table B.1.

**Table B.1** – Gyroscope, Accelerometer and GPS Simulation Noise Parameters

Sensor	Symbol	BLWN Block PSD	Block Sample Time [s]
Pitch Gyroscope	Q	$1.523 * 10^{-6}$	0.020
Roll Gyroscope	P	$1.523 * 10^{-6}$	0.020
Yaw Gyroscope	R	$1.523 * 10^{-6}$	0.020
X Accelerometer	$\dot{U}$	$2.000 * 10^{-4}$	0.020
Y Accelerometer	$\dot{V}$	$2.000 * 10^{-4}$	0.020
Z Accelerometer	$\dot{W}$	$2.000 * 10^{-4}$	0.020
GPS N Speed	$\dot{N}$	$1.000 * 10^{-7}$	0.002
GPS E Speed	$\dot{E}$	$1.000 * 10^{-7}$	0.002
GPS D Speed	$\dot{D}$	$8.000 * 10^{-8}$	0.002
GPS N Position	N	$1.000 * 10^{-4}$	0.002
GPS E Position	E	$1.000 * 10^{-4}$	0.002
GPS D Position (LF)	D	$2.000 * 10^{-8}$	0.002
GPS D Position (HF)	D	4.000	0.002

All sensor measurements are modelled purely by a BLWN block with a specific sample time and PSD. The GPS N,E and D position measurements, however, incorporate first-order filters to more accurately simulate GPS position drift. A block diagram illustrating how the noise is generated for the N, E and D position measurements is shown in Figure B.1.

The rotor lag dynamics were characterized during previous work. By commanding a thrust step reference to a rotor mounted to a load cell, the time constant of the lag dynamics could be determined. The step responses of a single rotor's thrust can be seen in Figure B.2. The

**Figure B.1** – GPS Position Sensing Noise Model**Figure B.2** – Rotor Step Response

three step commands were given in the range that the rotor will be operating in during flight.

Assuming that the response is a first-order response, the time constant could be determined from the results. Closer inspection of the data indicated that the time constant is approximately 0.125 seconds.

## Appendix C

# Control System Design

In this appendix, additional information that supports the control system design section is presented. The equations that convert the linear state space equations of the horizontal dynamics, the yaw dynamics and the vertical dynamics to transfer functions are shown.

### C.1 Horizontal Dynamics State Space To Transfer Function

$$G(s) = C(sI - A)^{-1}B + D \quad (\text{C.1.1})$$

$$G(s) = [0 \ 1] \left( \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} -\frac{1}{\tau} & 0 \\ \frac{1}{\tau} & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.1.2})$$

$$G(s) = [0 \ 1] \left( \begin{bmatrix} s + \frac{1}{\tau} & 0 \\ -\frac{1}{\tau} & s \end{bmatrix} \right)^{-1} \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.1.3})$$

$$G(s) = [0 \ 1] \left( \frac{1}{s(s + \frac{1}{\tau})} \begin{bmatrix} s & 0 \\ \frac{1}{\tau} & s + \frac{1}{\tau} \end{bmatrix} \right) \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.1.4})$$

$$G(s) = [0 \ 1] \left( \frac{1}{s(s + \frac{1}{\tau})} \begin{bmatrix} s & 0 \\ \frac{1}{\tau} & s + \frac{1}{\tau} \end{bmatrix} \right) \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.1.5})$$

$$G(s) = \frac{\frac{1}{\tau I_{yy}}}{s(s + \frac{1}{\tau})} \quad (\text{C.1.6})$$

### C.2 Yaw Dynamics State Space To Transfer Function

$$G(s) = C(sI - A)^{-1}B + D \quad (\text{C.2.1})$$

$$G(s) = [0 \ 1] \left( \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \begin{bmatrix} -\frac{1}{\tau} & 0 \\ \frac{1}{\tau} & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.2.2})$$

$$G(s) = [0 \ 1] \left( \begin{bmatrix} s + \frac{1}{\tau} & 0 \\ -\frac{1}{\tau} & s \end{bmatrix} \right)^{-1} \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.2.3})$$

$$G(s) = [0 \ 1] \left( \frac{1}{s(s + \frac{1}{\tau})} \begin{bmatrix} s & 0 \\ \frac{1}{\tau} & s + \frac{1}{\tau} \end{bmatrix} \right) \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.2.4})$$

$$G(s) = [0 \quad 1] \left( \frac{1}{s(s + \frac{1}{\tau})} \begin{bmatrix} s & 0 \\ \frac{1}{I_{zz}} & s + \frac{1}{\tau} \end{bmatrix} \right) \begin{bmatrix} \frac{1}{\tau} \\ 0 \end{bmatrix} \quad (\text{C.2.5})$$

$$G(s) = \frac{\frac{1}{\tau I_{zz}}}{s(s + \frac{1}{\tau})} \quad (\text{C.2.6})$$

### C.3 Vertical Dynamics State Space To Transfer Function

$$G(s) = C(sI - A)^{-1}B + D \quad (\text{C.3.1})$$

$$G(s) = \left[ -\frac{1}{m} \right] \left( [s] - \left[ -\frac{1}{\tau} \right] \right)^{-1} \left[ \frac{1}{\tau} \right] \quad (\text{C.3.2})$$

$$G(s) = \left( -\frac{1}{m} \right) \left( s + \frac{1}{\tau} \right)^{-1} \left( \frac{1}{\tau} \right) \quad (\text{C.3.3})$$

$$G(s) = \frac{-\frac{1}{\tau m}}{\left( s + \frac{1}{\tau} \right)} \quad (\text{C.3.4})$$

## Appendix D

# Antenna Tube Design

This appendix contains the drawings used to manufacture the antenna tube which housed both the dedicated DGPS communication antenna and the telemetry antenna. The three parts of the assembly can be manufactured from these drawings.

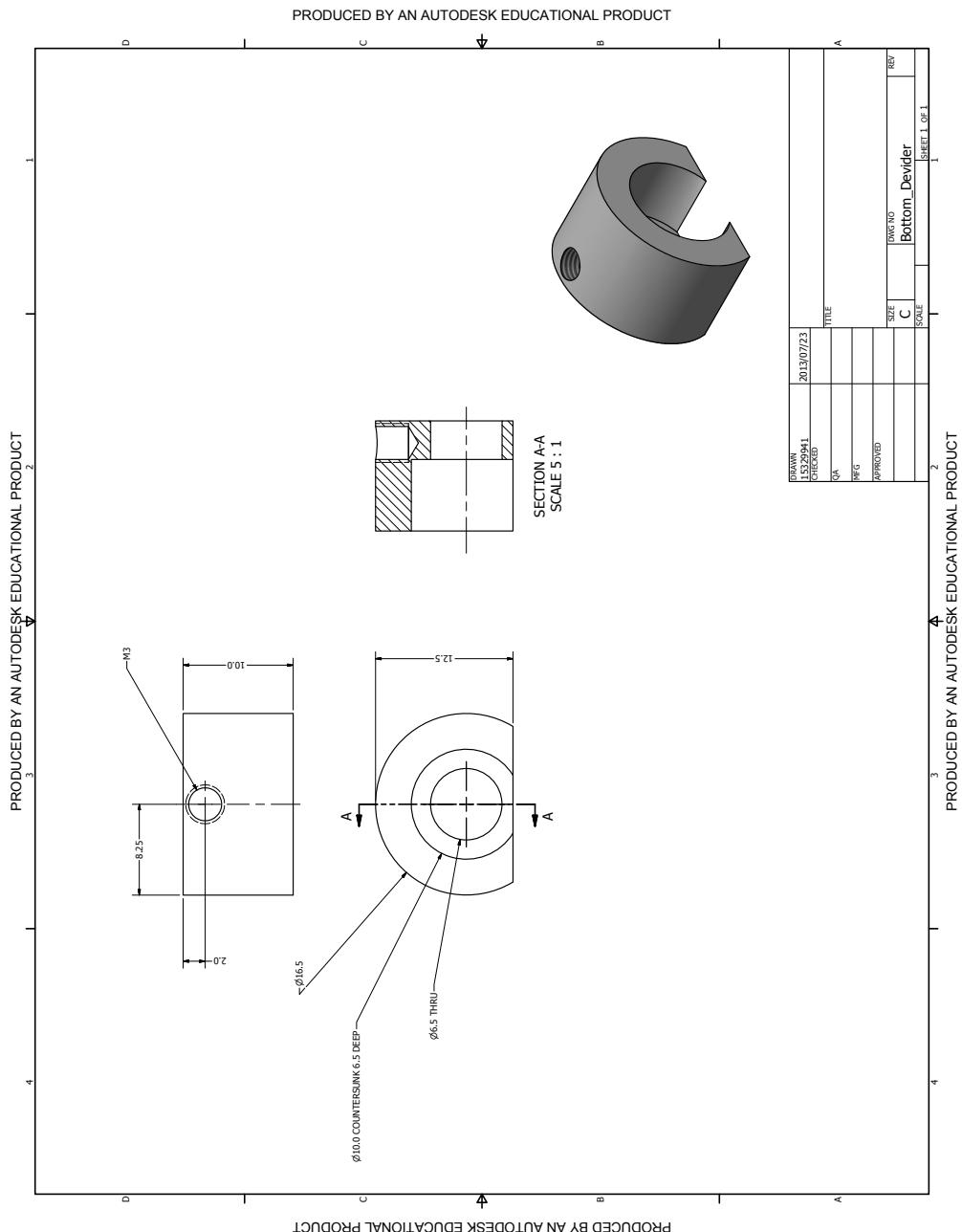


Figure D.1 – DGPS Communication Antenna Holder Drawing

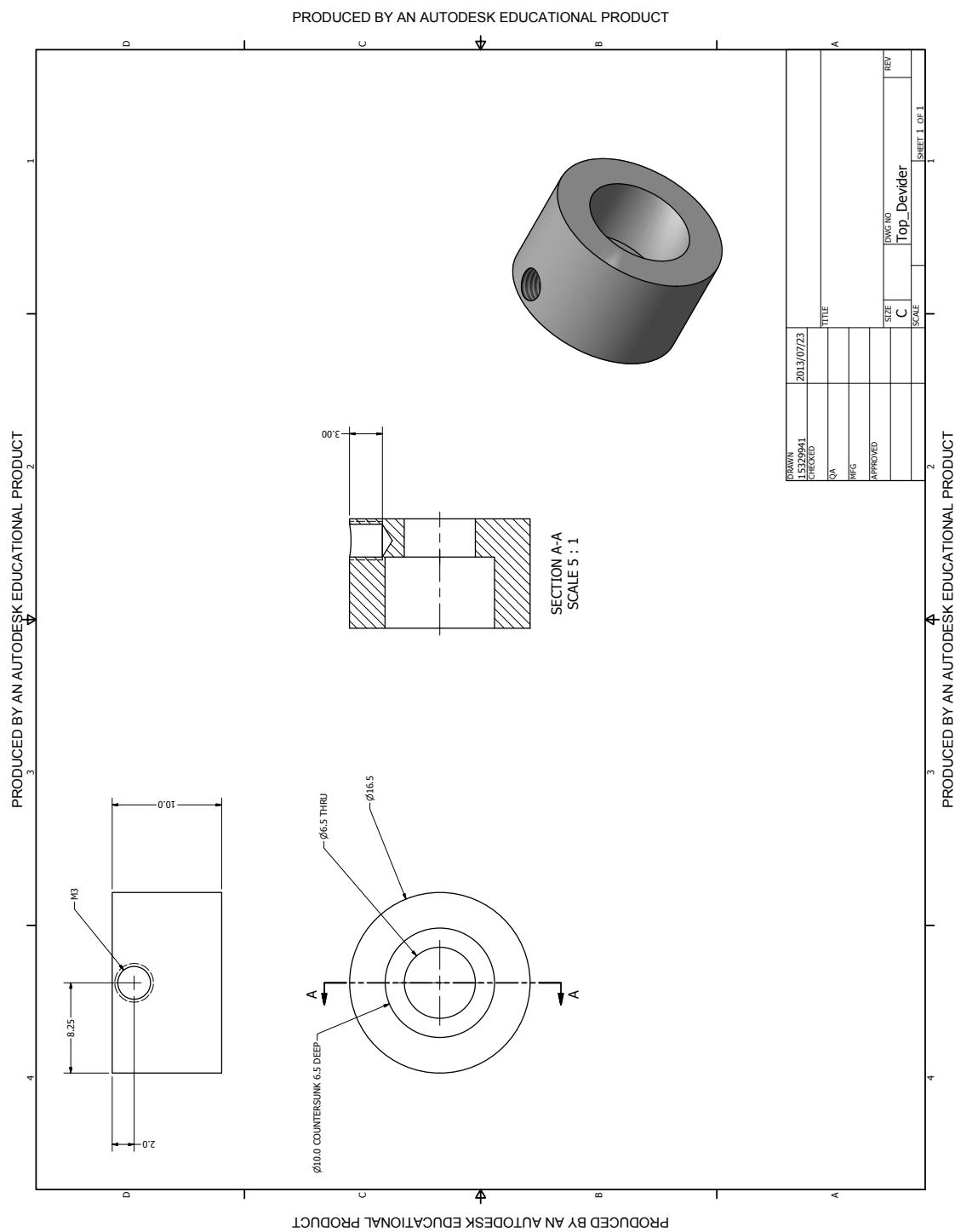


Figure D.2 – Telemetry Communication Antenna Holder Drawing

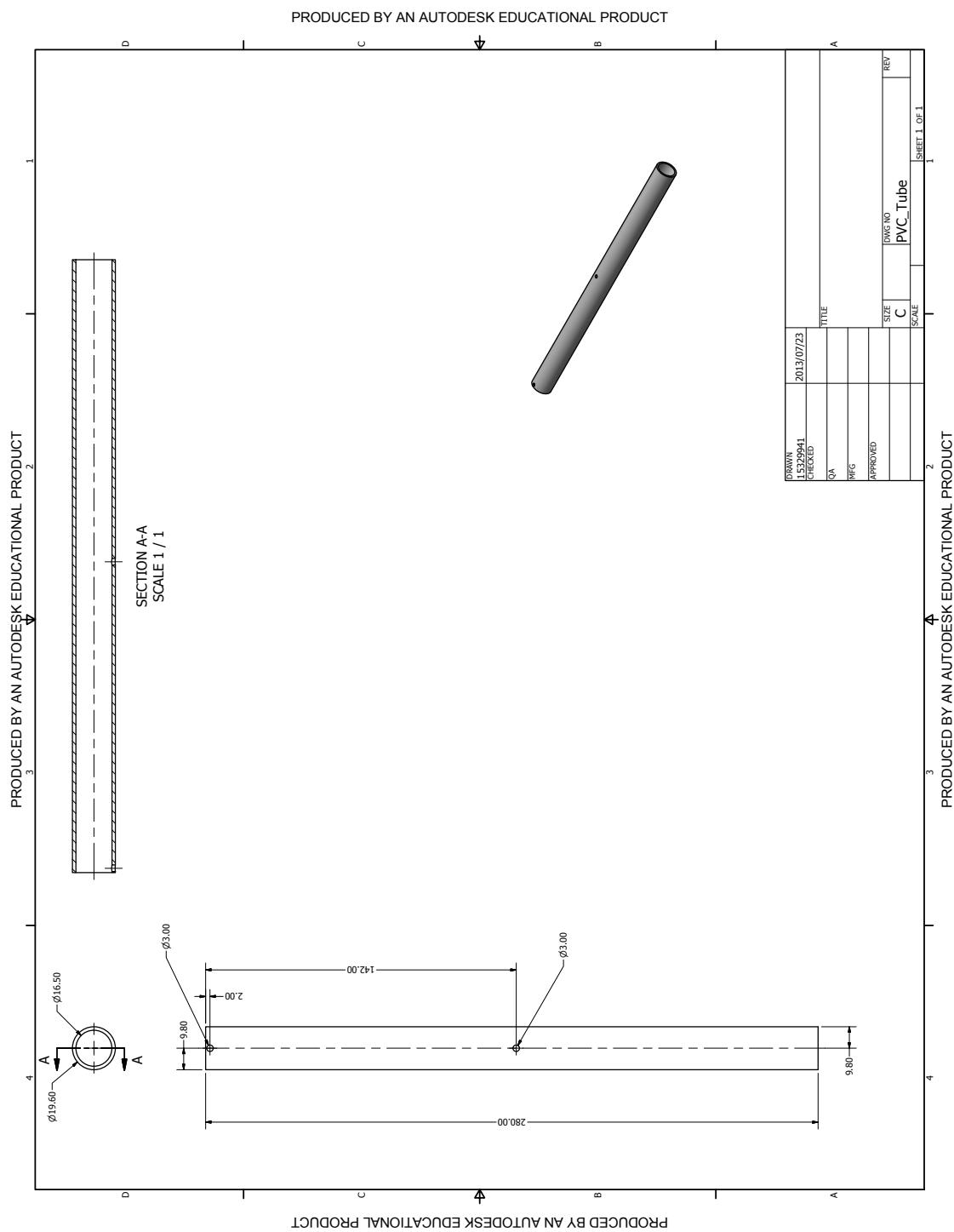


Figure D.3 – Antenna PVC Tube

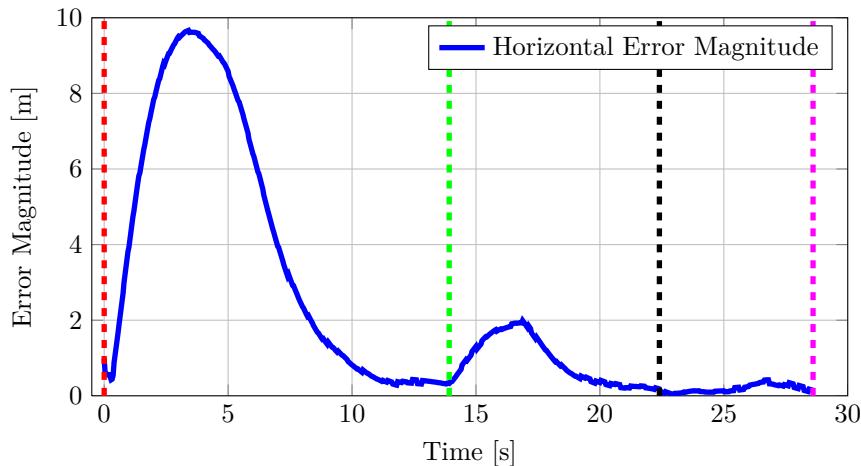
## Appendix E

# Additional HIL Simulation Results

This appendix presents additional results from the Hardware-In-the-Loop (HIL) simulations that were performed. Eighteen consecutive simulations were performed: six with the platform moving at  $3 \text{ m/s}$ , six with the platform moving at  $5 \text{ m/s}$  and six with the platform moving at  $7 \text{ m/s}$ . The results from a HIL simulation while the platform was moving at  $3 \text{ m/s}$  is shown in Chapter 8, while results from instances where the platform was moving at  $5 \text{ m/s}$  and  $7 \text{ m/s}$  are presented here. For the sake of convenience, the graphs are interpreted here exactly as the HIL results were in Chapter 8. The results from a HIL simulation performed while the platform moved at  $5 \text{ m/s}$  are presented first.

### E.1 5 m/s Simulation

The graphs illustrated in this section correspond to a HIL simulation performed with a platform moving at  $5 \text{ m/s}$ . Constant wind and wind gusts were modelled in the simulation. The absolute horizontal error magnitude is plotted in Figure E.1 during the HIL simulation.



**Figure E.1 – HIL 5 m/s Absolute Horizontal Position Error**

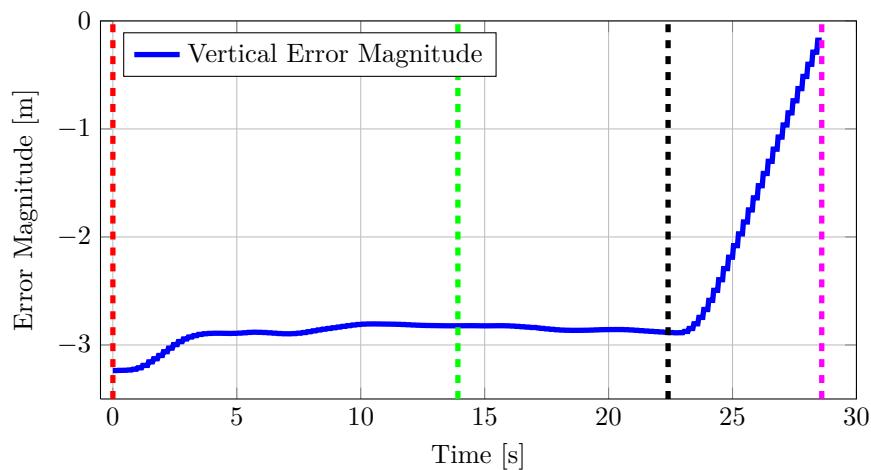
The vertical dashed lines indicate the states of the state machine, described in Chapter 7. The red dashed line indicates at what time the vehicle entered the Tracking state, where the vehicle has to track the landing location from  $3 \text{ m}$  behind and  $3 \text{ m}$  above. Immediately

after the red dashed line, the position error grows rapidly as the vehicle is stationary and the platform is moving at  $5 \text{ m/s}$ . As the vehicle accelerates, it gains speed and catches up to the platform. The vehicle then follows the reference point of  $3 \text{ m}$  behind the platform. As soon as the absolute error has remained under  $0.5 \text{ m}$  for 3 seconds, the next state is entered.

A green dashed line indicates the start of the Homing state. In this state, the vehicle homes in at a speed of  $1 \text{ m/s}$  to above the landing location. The error again starts to increase as the position reference is moved to above the landing location at a rate of  $1 \text{ m/s}$ . The vehicle has to increase its speed to finally reach the new position reference. When the absolute error has remained under  $0.5 \text{ m}$  for 3 seconds, the Descending state is entered.

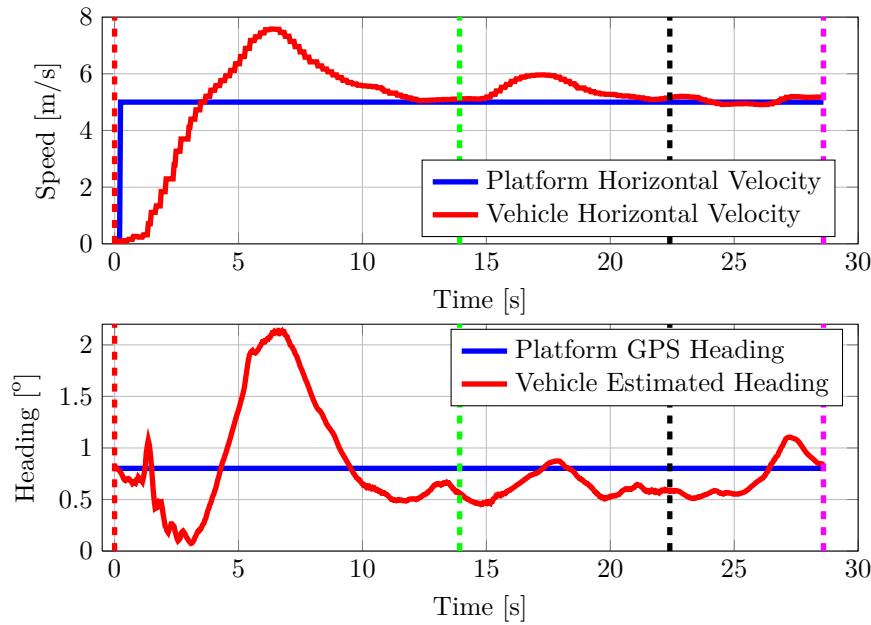
The black dashed line indicates the start of the Descending state, where the vehicle starts to descend onto the platform. The altitude controller is disarmed and the vehicle descends at a constant rate of  $0.5 \text{ m/s}$ . If the vehicle has stayed within the  $1 \text{ m}$  diameter cylinder throughout the Descending state, it will make contact with the platform. The final state, entered when the vehicle has touched down on the platform, is indicated by the magenta dashed line. An accelerometer spike with a magnitude exceeding  $15 \text{ m/s}^2$  is measured, causing the state machine to enter the Shutdown state. The vehicle finally touches down on the platform with a absolute position error of  $0.176 \text{ m}$ .

The vertical error during the HIL simulation while the platform is moving at  $5 \text{ m/s}$  is plotted in Figure E.2.



**Figure E.2 – HIL 5 m/s Simulation ENU Vector Up Component**

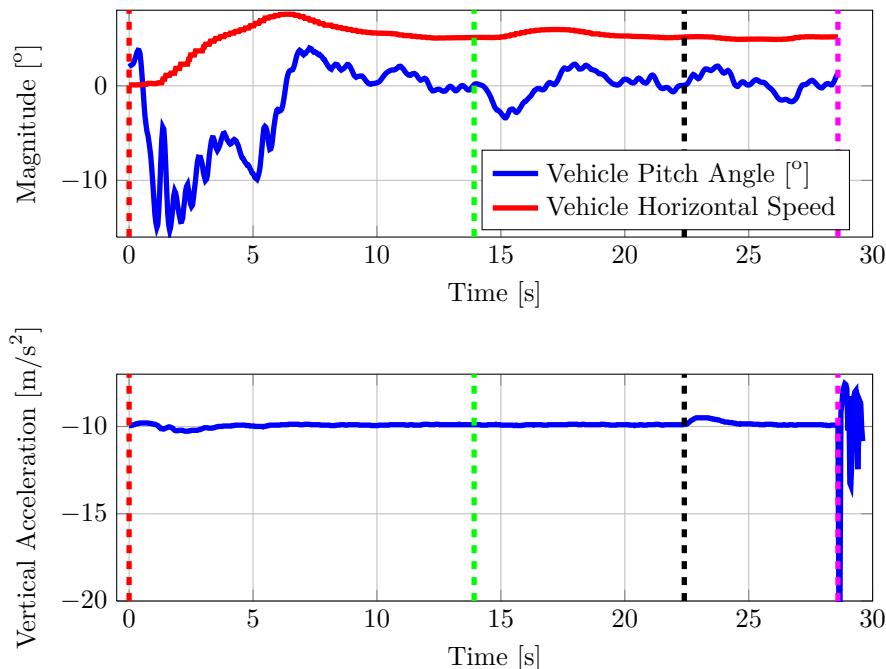
As soon as the Tracking state is entered, the vehicle descends from the arbitrary height it was holding during the Standby state. The vehicle descends to  $3 \text{ m}$  above the platform. It stays approximately  $3 \text{ m}$  above the platform until the Descending state is entered. The vehicle then descends at a rate of  $0.5 \text{ m/s}$  until it makes contact with the platform. Plotted in Figure E.3 is the velocity and heading of the vehicle and the platform during the HIL simulation, where the platform moved at  $5 \text{ m/s}$ . The dashed vertical lines again indicate when the different states of the autolanding state machine are entered.



**Figure E.3 – HIL 5 m/s Simulation Vehicle and Platform Velocity, Heading**

The velocity of the vehicle increases from zero to above 7 m/s as it gains on the position of the platform during the Tracking state. Approximately 3 seconds before it enters the Homing state, the vehicle matches the velocity of the platform. The velocity increases again while the vehicle homes in to above the landing location. From this point onwards, the vehicle approximately matches the velocity of the platform to the end of the landing procedure. The vehicle touches down on the platform with a velocity difference of 0.15 m/s. The platform's heading remains completely constant and the vehicle's heading varies by slightly more than 0.5° throughout the autolanding procedure. The vehicle finally touches down with its heading within 0.15° of the platform's heading.

Shown in Figure E.4 is the pitch angle of the vehicle and the accelerometer spike measured when the vehicle makes contact with the platform. The velocity is plotted with the pitch angle of the vehicle for convenience.

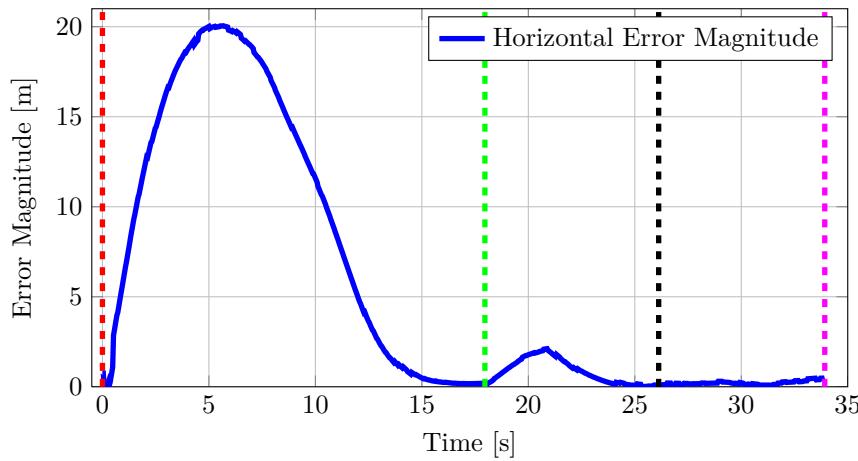


**Figure E.4 – HIL 5 m/s Simulation Vehicle Pitch Angle, Vertical Accelerometer Spike**

The pitch angle almost reaches  $-15^\circ$ , as the vehicle initially pitches when it accelerates from a standstill. The vehicle finally touches down with a pitch angle of  $1.20^\circ$ . The pitch angle is small enough not to be considered a problem during touchdown. A specific force spike of  $-20 \text{ m/s}^2$  is measured when the vehicle makes contact with the platform. This spike is sufficiently large so that the vehicle knows it has touched down on the platform and to raise the flag to enter the Shutdown state.

## E.2 7 m/s Simulation

The graphs illustrated in this section correspond to a HIL simulation performed with a platform moving at  $7 \text{ m/s}$ . Constant wind and wind gusts were modelled in the simulation. The absolute position error to the landing location on the platform is plotted in Figure E.5.



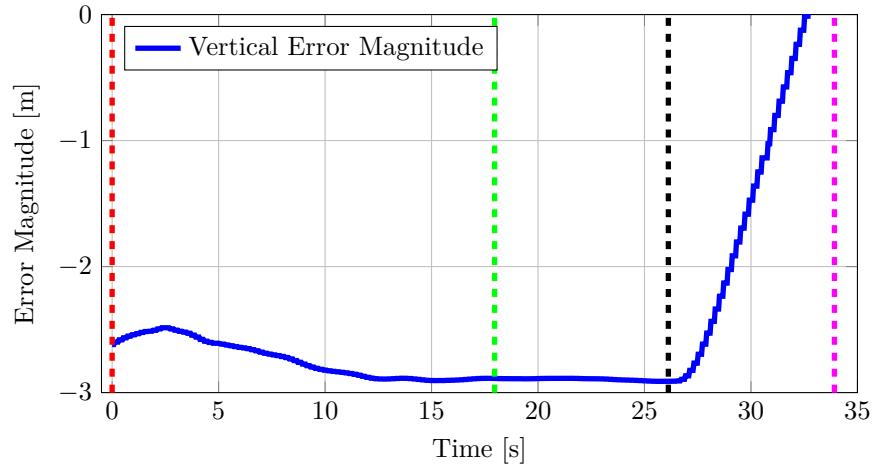
**Figure E.5 – HIL 7 m/s Absolute Horizontal Position Error**

The vertical dashed lines indicate the states of the state machine, described in Chapter 7. The red dashed line indicates at what time the vehicle entered the Tracking state, where the vehicle has to track the landing location from  $3 \text{ m}$  behind and  $3 \text{ m}$  above. Immediately after the red dashed line, the position error grows rapidly, as the vehicle is stationary and the platform is moving at  $7 \text{ m/s}$ . As the vehicle accelerates, it gains speed and catches up to the platform. The vehicle then follows the reference point of  $3 \text{ m}$  behind the platform. As soon as the absolute error has remained under  $0.5 \text{ m}$  for 3 seconds, the next state is entered.

A green dashed line indicates the start of the Homing state. In this state, the vehicle homes in at a speed of  $1 \text{ m/s}$  to above the landing location. The error again starts to increase as the position reference is moved to above the landing location at a rate of  $1 \text{ m/s}$ . The vehicle has to increase its speed to finally reach the new position reference. When the absolute error has remained under  $0.5 \text{ m}$  for 3 seconds, the Descending state is entered.

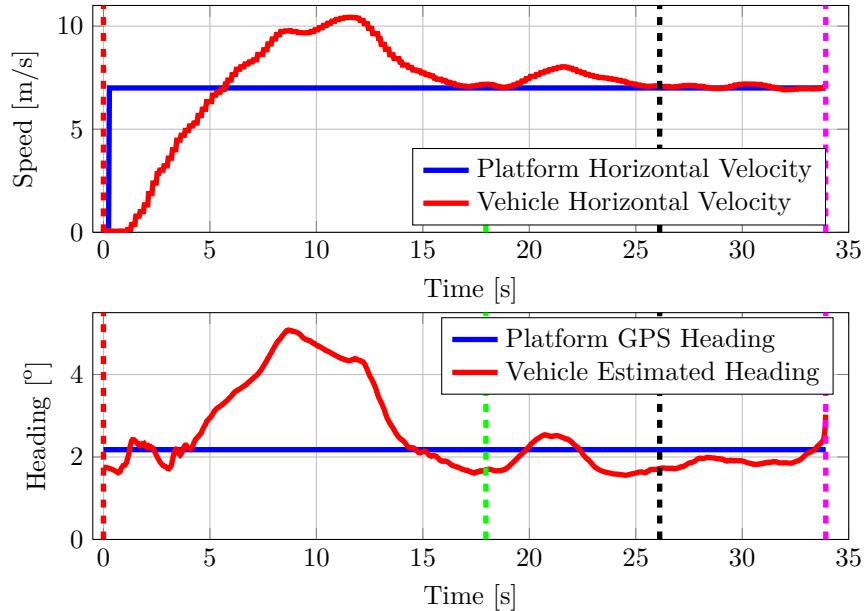
The black dashed line indicates the start of the Descending state, where the vehicle starts to descend onto the platform. The altitude controller is disarmed and the vehicle descends at a constant rate of  $0.5 \text{ m/s}$ . If the vehicle has stayed within the  $1 \text{ m}$  diameter cylinder throughout the Descending state, it will make contact with the platform. The final state, entered when the vehicle has touched down on the platform, is indicated by the magenta dashed line. An accelerometer spike with a magnitude exceeding  $15 \text{ m/s}^2$  is measured,

causing the state machine to enter the Shutdown state. The vehicle finally touches down on the platform with a absolute position error of 0.220 m.



**Figure E.6 – HIL 7 m/s Simulation ENU Vector Up Component**

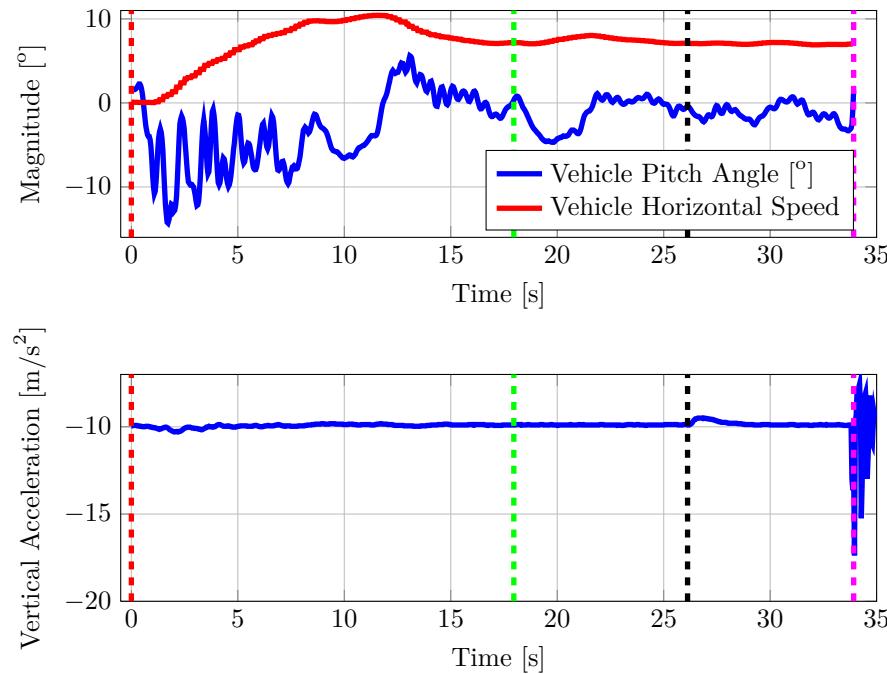
As soon as the Tracking state is entered, the vehicle descends from the arbitrary height it was holding during the Standby state. The vehicle descends to 3 m above the platform. It stays approximately 3 m above the platform until the Descending state is entered. The vehicle then descends at a rate of 0.5 m/s until it makes contact with the platform. Plotted in Figure E.7 is the velocity and heading of the vehicle and the platform during the HIL simulation where the platform moved at 7 m/s. The dashed vertical lines again indicate when the different states of the autolanding state machine are entered.



**Figure E.7 – HIL 7 m/s Simulation Vehicle and Platform Velocity, Heading**

The velocity of the vehicle increases from zero to  $11 \text{ m/s}$  as it gains on the position of the platform during the Tracking state. Approximately 3 seconds before it enters the Homing state, the vehicle matches the velocity of the platform. The velocity increases again while the vehicle homes in to above the landing location. From this point onwards, the vehicle approximately matches the velocity of the platform to the end of the landing procedure. The vehicle touches down on the platform with a velocity difference of  $0.152 \text{ m/s}$ . The platform's heading remains completely constant and the vehicle's heading varies by slightly more than  $0.5^\circ$  throughout the autolanding procedure. The vehicle finally touches down with its heading within  $3^\circ$  of the platform's heading.

Shown in Figure E.4 is the pitch angle of the vehicle and the accelerometer spike measured when the vehicle makes contact with the platform. The velocity is plotted with the pitch angle of the vehicle for convenience.



**Figure E.8 – HIL 7 m/s Simulation Vehicle Pitch Angle, Vertical Accelerometer Spike**

The pitch angle almost reaches  $-13^\circ$ , as the vehicle initially pitches when it accelerates from a standstill. The vehicle finally touches down with a pitch angle of  $0.77^\circ$ . The pitch angle is small enough not to be considered a problem during touchdown. A specific force spike of  $-19.23 \text{ m/s}^2$  is measured when the vehicle makes contact with the platform. This spike is sufficiently large so that the vehicle knows it has touched down on the platform and to raise the flag to enter the Shutdown state.

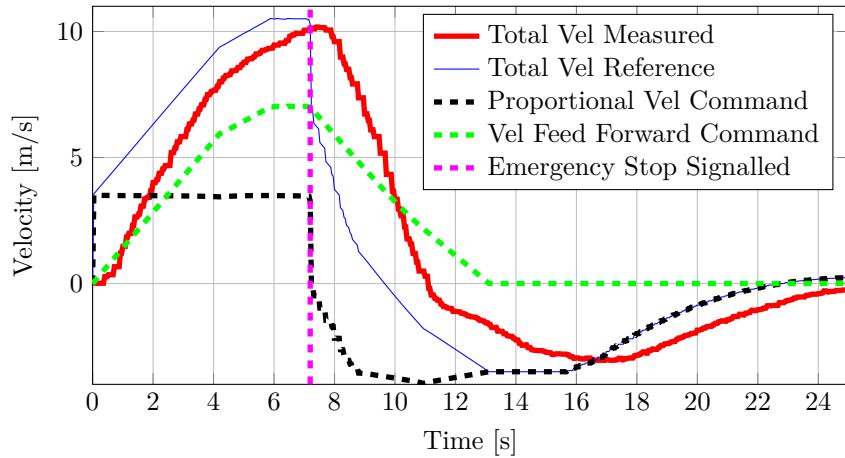
## Appendix F

# Additional Flight Test Results

In this appendix, additional results of the practical flight tests are presented. Five flight tests were conducted, and of these, four were performed more than once. The additional results of flight test 2 to flight test 5 are presented here. For the sake of convenience, the graphs are interpreted here in the same manner as in Chapter 9.

### F.1 Flight Test 2 Attempt 2

The figures in this section exhibit results from the second attempt at operating the vehicle at acceleration extremes that will be seen during an autolanding on a moving platform. Illustrated in Figure F.1 are the velocity references and measurement in the direction of the 50 m step during the second attempt. The 50 m position step command was issued shortly after  $t = 0$  seconds, whereas the emergency stop was commanded approximately 7.3 seconds into the test.

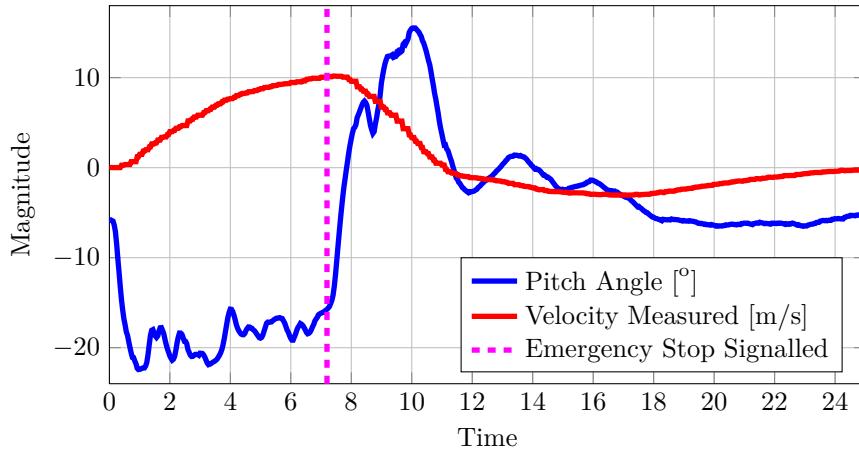


**Figure F.1** – Forward Velocity Reference and Response During the Second Attempt at the Second Test

First consider the proportional and feed forward velocity commands. The proportional velocity command immediately limits to  $3.5 \text{ m/s}$  when the position step is issued, whereas the feed forward velocity command is slewed from  $0 \text{ m/s}$  up to  $7 \text{ m/s}$ . After the emergency stop is commanded, the proportional velocity command limits at  $-3.5 \text{ m/s}$  and the feed forward velocity command is again slewed, from  $7 \text{ m/s}$  to  $0 \text{ m/s}$ . Now consider the total

velocity reference and the velocity measured. The vehicle accelerates aggressively, reaching a maximum velocity of  $10.5 \text{ m/s}$  before the emergency stop is commanded. The vehicle then decelerates aggressively in an attempt to come to a standstill.

The pitch angle response during maximum acceleration and maximum deceleration is also investigated. The vehicle's pitch and roll angle commands are limited to  $\pm 22.5^\circ$ , and correspond to the maximum acceleration that the vehicle can maintain. Figure F.2 shows the measured pitch angle response during the acceleration tests. The measured velocity is also plotted as reference.



**Figure F.2** – Recorded Pitch Angle During the Second Attempt at the Second Flight Test

The pitch angle limit is reached during forward acceleration. The pitch angle saturates at its negative limit for a duration of 0.3 seconds, reaching a maximum negative pitch angle of  $22.4^\circ$ . After the emergency stop has been commanded a maximum positive pitch angle of  $15.5^\circ$  is seen, indicating that the pitch angle does not saturate during the emergency stop deceleration.

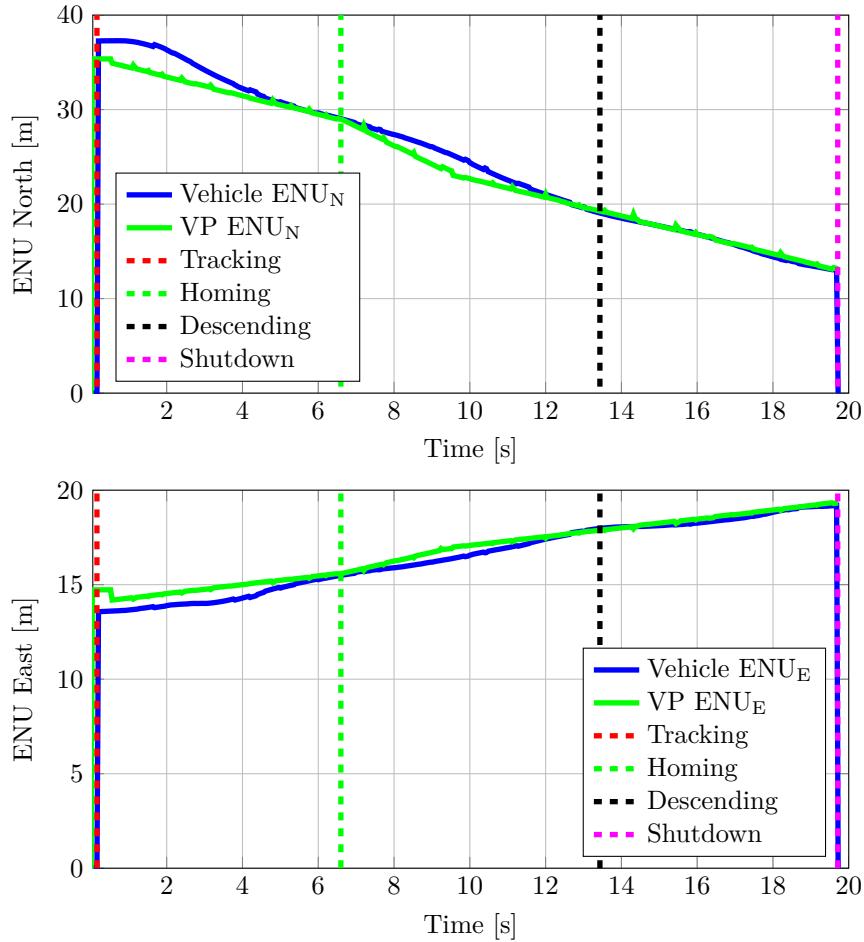
## F.2 Flight Test 3

The figures in this section exhibit results from the first and fourth attempts during flight test 3, where autolandings were attempted on a virtual platform.

### F.2.1 Attempt 1

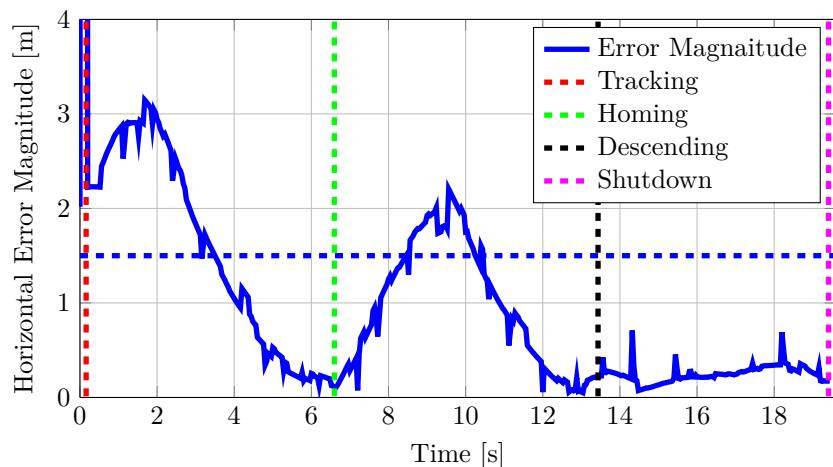
The first attempt at a virtual platform landing was successful. The platform was translating at  $1 \text{ m/s}$  and the error bounds were  $3 \text{ m}$  for the Tracking and the Homing states. In the top graph of Figure F.3, the north component of the reference ENU vector is compared to the north components of the actual ENU vector measured by the vehicle. Similarly, in the bottom graph the east ENU reference vector components are compared to the actual east ENU components measured by the vehicle.

What can effectively be seen is how the vehicle tracks the virtual platform relative to the Rover sensor on the stationary platform. Four dashed lines are also plotted, indicating the start of the different states of the state machine. Toothed edges and irregularities can be seen on the green curve, which represents the reference ENU trajectory that the vehicle needs to follow.



**Figure F.3** – Tracking Of The Virtual Platform During First Attempt of Third Flight Test

Figure F.4 shows the absolute horizontal error during the landing procedure. Due to delays in the telemetry link while sending the platform information, the toothed edges in Figure F.3 manifest as sharp spikes in Figure F.4, which can easily prevent the vehicle from passing the state machine's error checks, even though the vehicle is tracking the virtual platform adequately.

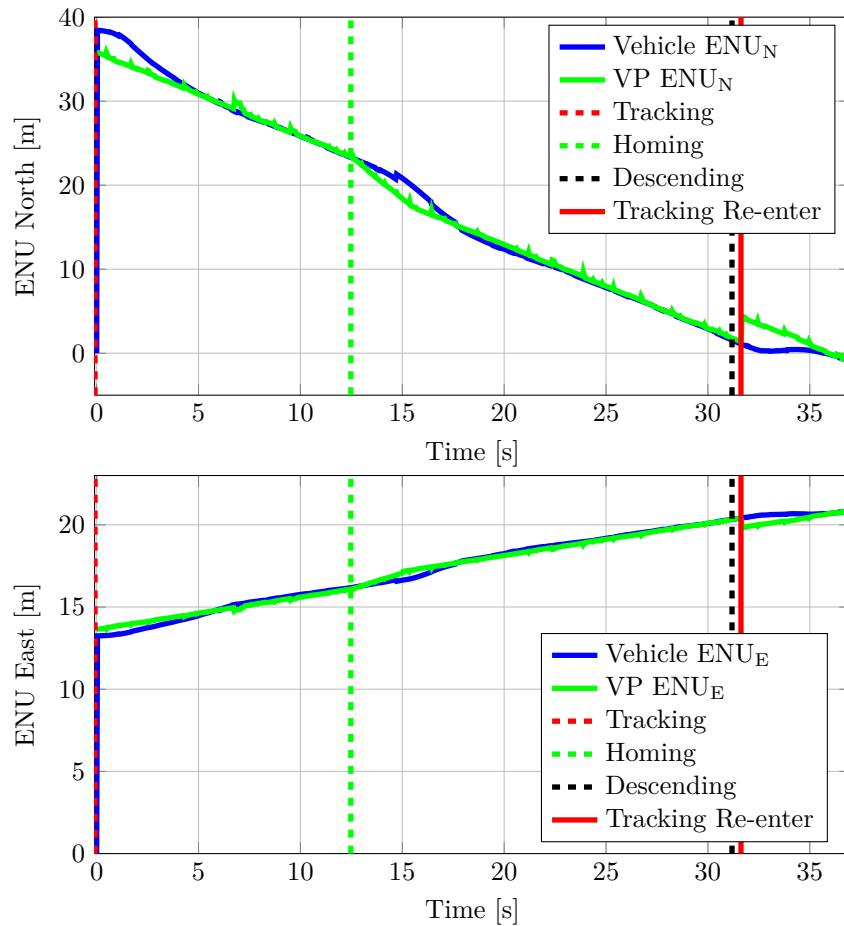


**Figure F.4** – Virtual Landing Absolute Horizontal Position Error During First Attempt at Third Flight Test

The state machine lines are plotted with an additional blue dashed line. The blue dashed line is the error bound that the vehicle needs to remain within in order to advance to the next state of the state machine. For this test the error bounds were 3 m. The absolute error is thus not allowed to exceed 1.5 m. Between the start of the Tracking and Homing state, the error grows and then decreases as the vehicle catches up with the tracking location, 3 m behind the landing position. As soon as the vehicle has passed the check of not going outside of the 3 m diameter sphere for 3 seconds, the Homing state starts. The error again increases as the reference position is homed to directly above the landing location. Once the vehicle has passed the state machine's check while in the Homing state, the Descending state is entered. The vehicle starts to descend and does not deviate from the landing location by more than 1.5 m. The Shutdown state is finally entered when the vehicle reaches the height of the virtual platform.

### F.2.2 Attempt 4

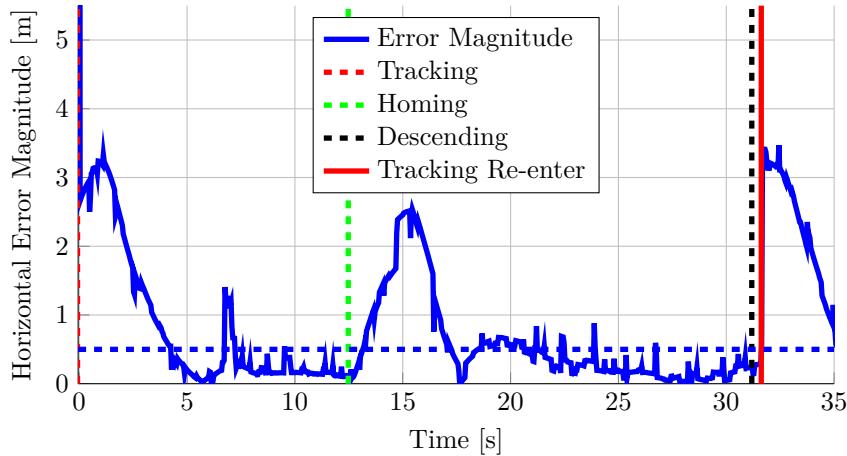
The fourth attempt at a virtual platform landing was not successful. The platform was translating at 1 m/s and the error bounds were 1 m for the Tracking and the Homing states. In the top graph of Figure F.5, the north component of the reference ENU vector is compared to the north components of the actual ENU vector measured by the vehicle. Similarly, in the bottom graph the east ENU reference vector components are compared to the actual east.



**Figure F.5 – Tracking Of The Virtual Platform During Fourth Attempt at Third Flight Test**

Toothed edges are again seen on the north and east reference ENU components as a result of delays in the telemetry wireless link. The state machine entered the Descending state where it started to descend, but almost immediately failed the error check due to a spike in the absolute horizontal error. The state machine reverted to and remained in the Tracking state.

The absolute error for the fourth test is plotted in Figure F.6. The state machine re-enters the Tracking state shortly after entering the Descending state. The error then grows to just above 3 m, when the reference location is moved to 3 m behind and 3 m above the landing location.



**Figure F.6** – Virtual Landing Absolute Horizontal Position Error During Fourth Attempt at Third Flight Test

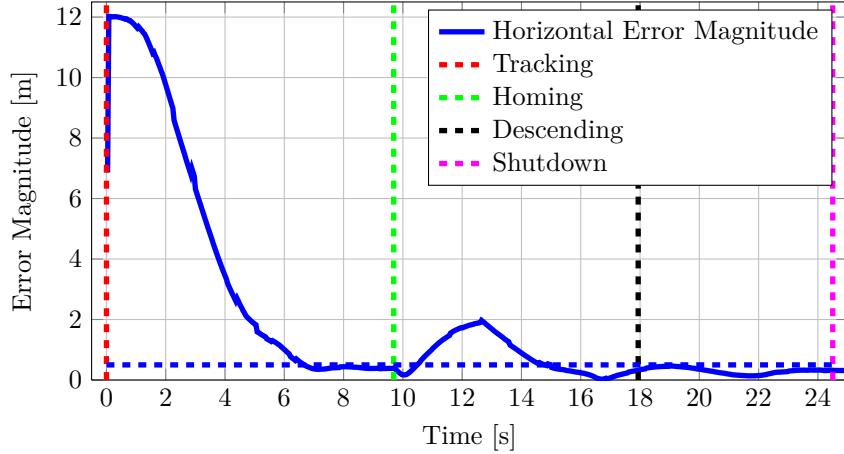
### F.3 Flight Test 4

The figures in this appendix exhibit results from the fourth flight test, where an autonomous landing was performed on a stationary platform. The flight test was performed five times. The first attempt was successful and is documented in Chapter 9. The second, third and fourth attempts are documented here. They were also successful autolandings on a stationary platform. The fifth attempt was not successful, and its results are presented at the end of this section.

#### F.3.1 Attempt 2

These results are from the second attempt, where the vehicle successfully landed on the stationary platform. The absolute horizontal error during the stationary autolanding is plotted in Figure F.7.

The dashed lines are plotted to indicate the start of the different states of the state machine. A dashed horizontal blue line is also plotted, indicating the error bounds for the test. The absolute error should not exceed 0.5 m for the state machine to advance through the states. After the Tracking state has started, the vehicle closes in on the platform, tracking the reference location behind the landing mark. Approximately 3 seconds after the absolute error curve crosses the error bound line, the Homing state is entered. The absolute error grows and then decreases as the vehicle tracks the reference which homes the vehicle in. Again, 3 seconds after the absolute error curve crosses the error bound line, the Descending

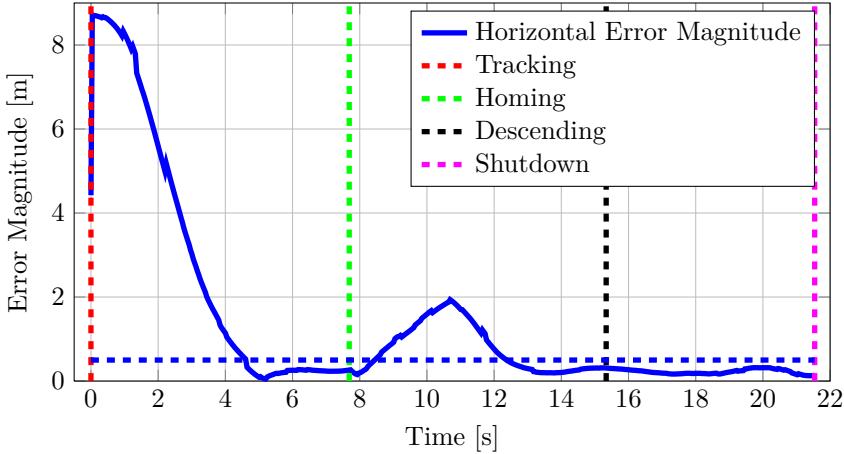


**Figure F.7** – Absolute Horizontal Error, Flight Test 4, Attempt 2

state is entered. The vehicle tracks the landing location from above, staying within the error bounds until the Shutdown state.

### F.3.2 Attempt 3

These results are from the third attempt, where the vehicle successfully landed on the stationary platform. The absolute horizontal error during the stationary autolanding is plotted in Figure F.8.



**Figure F.8** – Absolute Horizontal Error, Flight Test 4, Attempt 3

The dashed lines are plotted to indicate the start of the different states of the state machine. A dashed horizontal blue line is also plotted, indicating the error bounds for the test. The absolute error should not exceed  $0.5\text{ m}$  for the state machine to advance through the states. After the Tracking state has started, the vehicle closes in on the platform, tracking the reference location behind the landing mark. Approximately 3 seconds after the absolute error curve crosses the error bound line, the Homing state is entered. The absolute error grows and then decreases as the vehicle tracks the reference which homes the vehicle in. Again, 3 seconds after the absolute error curve crosses the error bound line, the Descending

state is entered. The vehicle tracks the landing location from above, staying within the error bounds until the Shutdown state.

### F.3.3 Attempt 4

These results are from the third attempt, where the vehicle successfully landed on the stationary platform. The absolute horizontal error during the stationary autolanding is plotted in Figure F.9.

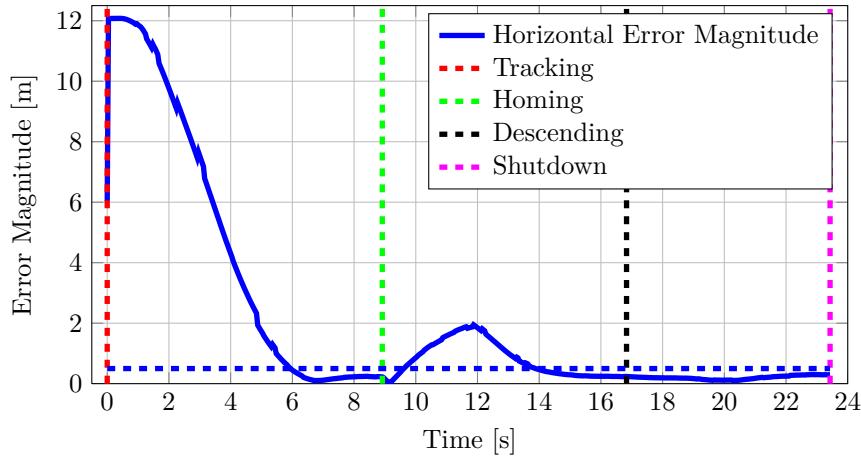


Figure F.9 – Absolute Horizontal Error, Flight Test 4, Attempt 4

The dashed lines are plotted to indicate the start of the different states of the state machine. A dashed horizontal blue line is also plotted, indicating the error bounds for the test. The absolute error should not exceed  $0.5\text{ m}$  for the state machine to advance through the states. After the Tracking state has started, the vehicle closes in on the platform, tracking the reference location behind the landing mark. Approximately 3 seconds after the absolute error curve crosses the error bound line, the Homing state is entered. The absolute error grows and then decreases as the vehicle tracks the reference which homes the vehicle in. Again, 3 seconds after the absolute error curve crosses the error bound line, the Descending state is entered. The vehicle tracks the landing location from above, staying within the error bounds until the Shutdown state.

### F.3.4 Attempt 5

These results are from the fifth attempt, where the vehicle did not pass all the checks of the state machine. The state machine reverted to the Tracking state shortly after entering the Descending state. The probable cause of this is due to external disturbances like wind.

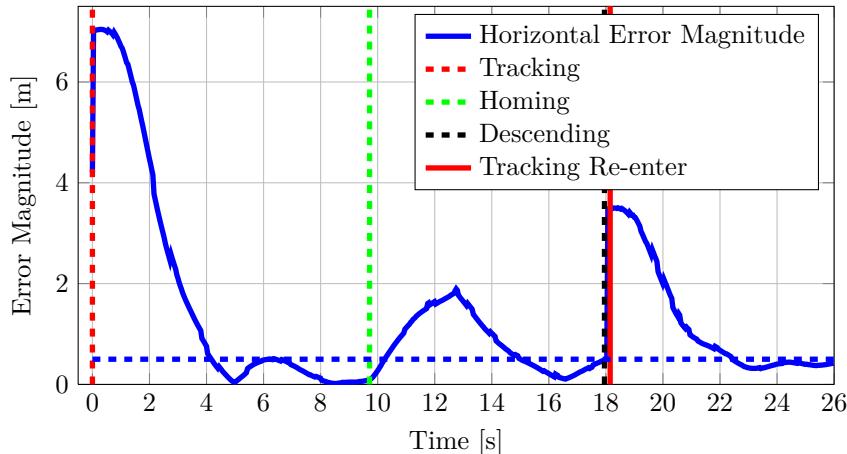


Figure F.10 – Absolute Horizontal Error, Flight Test 4, Attempt 5

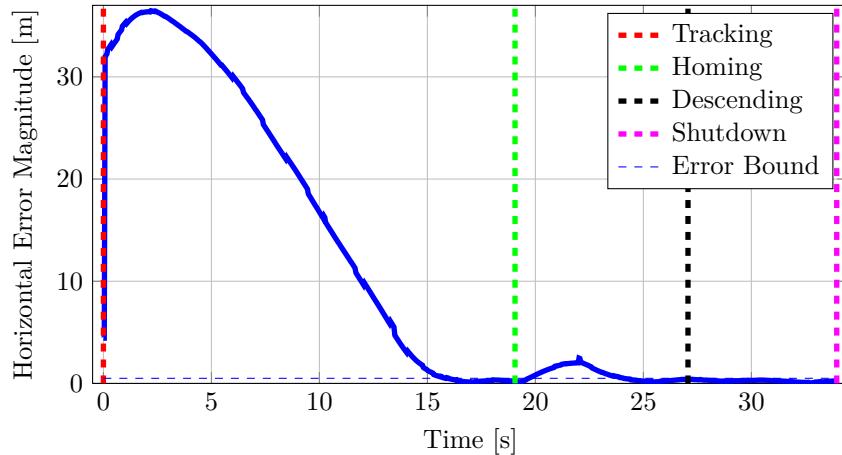
The dashed lines are plotted to indicate the start of the different states of the state machine. A dashed horizontal blue line is also plotted, indicating the error bounds for the test. The absolute error should not exceed  $0.5\text{ m}$  for the state machine to advance through the states. After the Tracking state has started, the vehicle closes in on the platform, tracking the reference location behind the landing mark. Approximately 3 seconds after the absolute error curve crosses the error bound line, the Homing state is entered. The absolute error grows and then decreases as the vehicle tracks the reference which homes the vehicle in. Again, 3 seconds after the absolute error curve crosses the error bound line, the Descending state is entered. However, the aircraft does not stay horizontally within  $0.5\text{ m}$  of the landing mark, almost immediately drifting out of the admissible error bounds after the Descending state is entered. The state machine re-enters the Tracking state and the absolute error grows to almost  $4\text{ m}$  as the reference position is moved  $3\text{ m}$  back and  $3\text{ m}$  up from the landing location.

## F.4 Flight Test 5

The figures in this section exhibit additional test results from the fifth flight test, where the vehicle attempted to autonomously land on the moving platform. Results are shown from autolanding attempts while the platform moved at speeds of  $10\text{ km/h}$ ,  $20\text{ km/h}$  and  $40\text{ km/h}$ . The attempts at  $10\text{ km/h}$  and  $20\text{ km/h}$  were successful, whereas the attempt at  $40\text{ km/h}$  was unsuccessful.

### F.4.1 Attempt 1

Results from the first attempt at landing on a translating platform are presented here. The vehicle successfully landed on the translating platform while the platform was moving at a speed of  $10\text{ km/h}$ . The absolute horizontal error recorded during the first attempt is shown in Figure F.11.

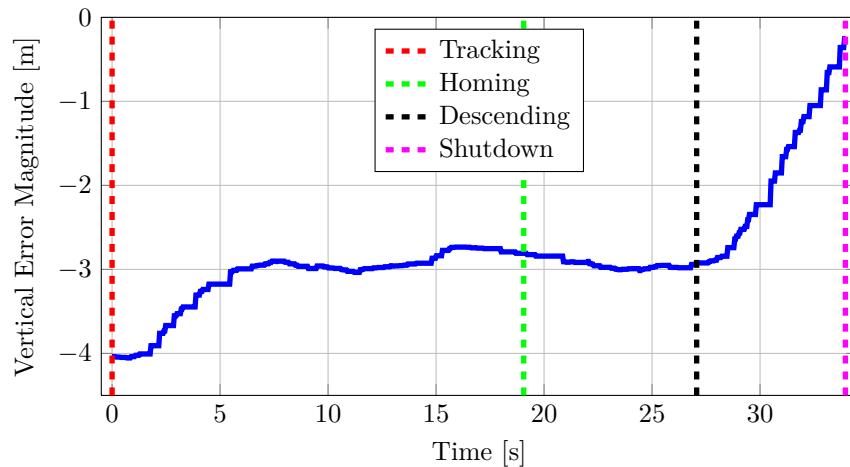


**Figure F.11** – Absolute Horizontal Position Error During  $10\text{ km/h}$  Automated Landing

The state machine lines are plotted, indicating the start of each state. The error bound is also plotted, indicating the maximum value that the absolute error is permitted to be to land in a  $1\text{ m}$  diameter circle. The vehicle starts approximately  $32\text{ m}$  behind the platform when the Tracking state starts. The error decreases as the vehicle starts to accelerate, catching up to the platform. Seventeen seconds into the test, the error is less than  $0.5\text{ m}$ . Three

seconds later the Homing state is entered, where the error again grows as the vehicle has to accelerate to home in to a position above the center of the moving platform. At 25 seconds, the absolute error is again less than  $0.5\text{ m}$ , entering the Descending state 3 seconds later. The error remains below  $0.5\text{ m}$  for the entire Descending state, entering the Shutdown state 37 seconds from the start of the landing procedure.

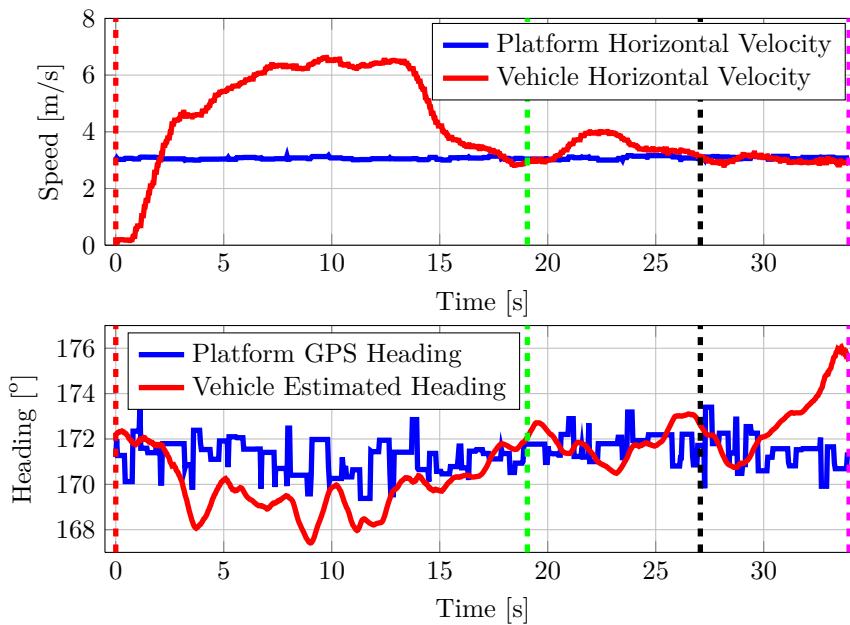
The vertical position error recorded during the flight test is shown in Figure F.12. The ENU Up component is plotted, effectively indicating the vertical error between the vehicle and the platform.



**Figure F.12** – Vertical Error During  $10\text{ km/h}$  Automated Landing

The landing procedure starts with the vehicle  $4\text{ m}$  above the landing location. The vehicle descends during the Tracking state, maintaining a height of  $3\text{ m}$  above the landing location until the Descending state is entered. During the Descending state, the vehicle descends at a rate of  $0.5\text{ m/s}$ , finally making contact with the platform.

The top figure of Figure F.13 displays the measured velocities of the platform and the vehicle throughout the landing procedure. The bottom figure shows the heading measurements of both the platform and the vehicle.



**Figure F.13** – Platform and Vehicle Velocity, Heading, During  $10\text{ km/h}$  Automated Landing

It can be seen that the platform's velocity has reached the target speed of  $10 \text{ km/h}$  when the landing procedure starts. Seventeen seconds into the landing procedure, the vehicle matches the velocity of the platform. After the Homing state is entered, the vehicle speed increases slightly while the vehicle homes in, matching the velocity of the platform again shortly afterwards. The Descending state starts and the vehicle approximately matches the velocity of the platform until the Shutdown state. The platform heading can be seen remaining fairly constant throughout the landing procedure. The aircraft tracks the heading of the platform throughout the autolanding procedure, with its heading not deviating by more than  $3^\circ$  from the platform's heading.

#### F.4.2 Attempt 2

Results from the second attempt at landing on a translating platform are presented here. The vehicle successfully landed on the translating platform while the platform was moving at a speed of  $20 \text{ km/h}$ . The absolute horizontal error recorded during the second attempt is shown in Figure F.14.

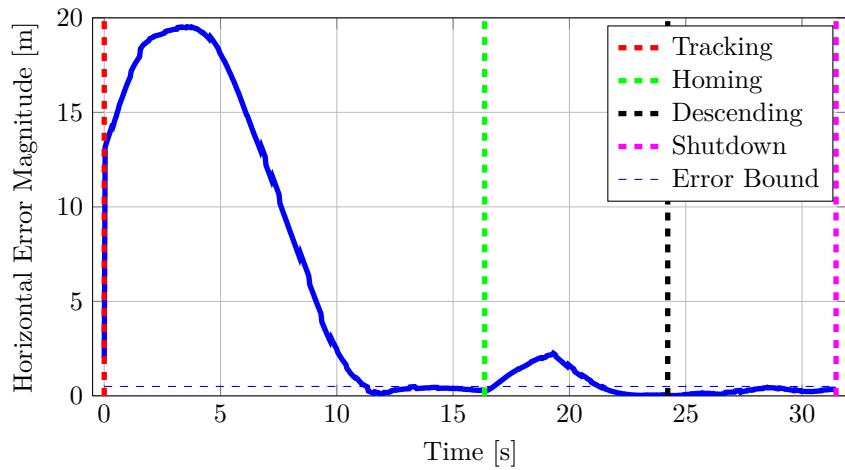
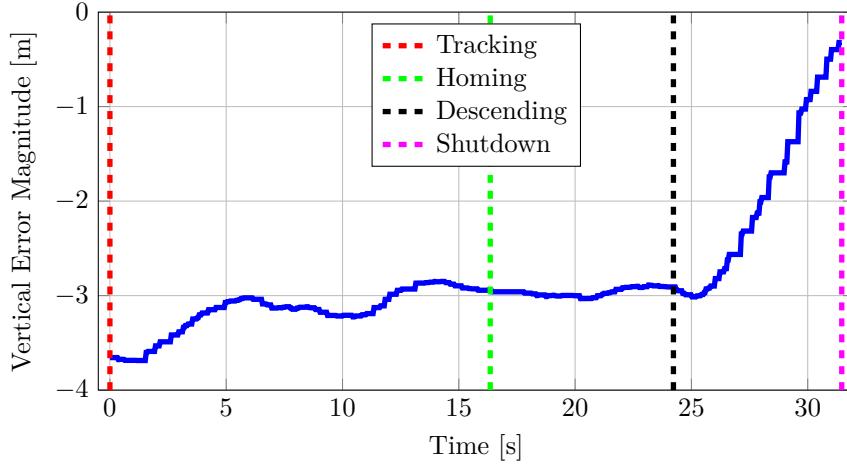


Figure F.14 – Absolute Horizontal Position Error During  $20 \text{ km/h}$  Automated Landing

The state machine lines are plotted, indicating the start of each state. The error bound is also plotted, indicating the maximum value that the absolute error is permitted to be to land in a  $1 \text{ m}$  diameter circle. The vehicle starts approximately  $14 \text{ m}$  behind the platform when the Tracking state starts. The error decreases as the vehicle starts to accelerate, catching up to the platform. Thirteen seconds into the test, the error is less than  $0.5 \text{ m}$ . Three seconds later the Homing state is entered, where the error again grows as the vehicle has to accelerate to home in to a position above the center of the moving platform. At 21 seconds, the absolute error is again less than  $0.5 \text{ m}$ , entering the Descending state 3 seconds later. The error remains below  $0.5 \text{ m}$  for the entire Descending state, entering the Shutdown state 32 seconds from the start of the landing procedure.

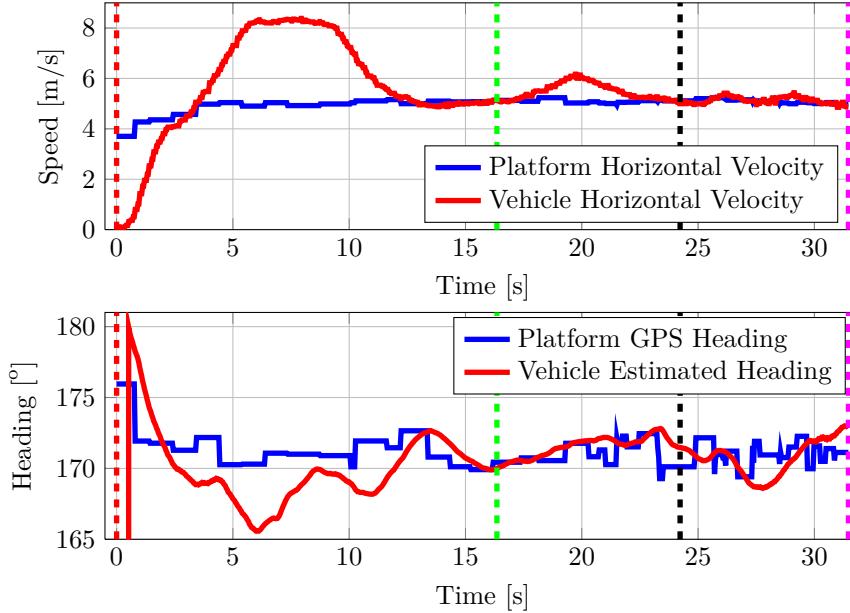
The vertical position error recorded during the flight test is shown in Figure F.15. The ENU Up component is plotted, effectively indicating the vertical error between the vehicle and the platform.



**Figure F.15** – Vertical Error During 20 km/h Automated Landing

The landing procedure starts with the vehicle 3.7 m above the landing location. The vehicle descends during the Tracking state, maintaining a height of 3 m above the landing location until the Descending state is entered. During the Descending state, the vehicle descends at a rate of 0.5 m/s, finally making contact with the platform.

The top figure of Figure F.16 displays the measured velocities of the platform and the vehicle throughout the landing procedure. The bottom figure shows the heading measurements of both the platform and the vehicle.



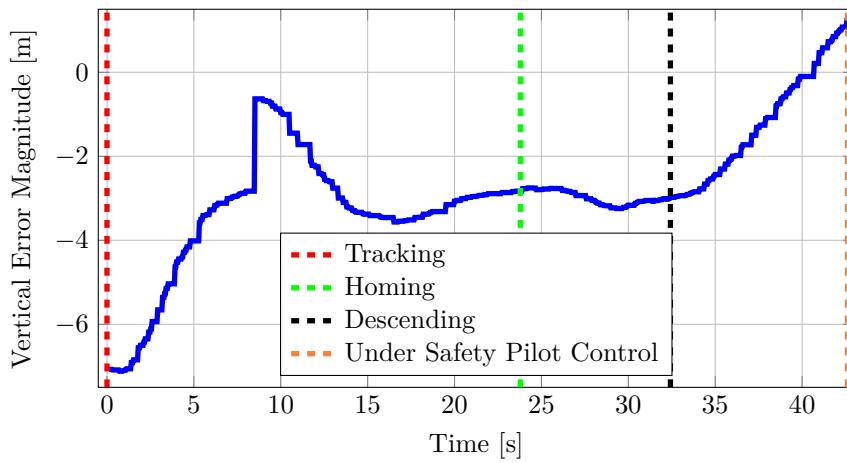
**Figure F.16** – Platform and Vehicle Velocity, Heading, During 20 km/h Automated Landing

It can be seen that the platform's velocity has almost reached the target speed of 20 km/h when the landing procedure starts. Thirteen seconds into the landing procedure, the vehicle matches the velocity of the platform. After the Homing state is entered, the vehicle speed

increases slightly while the vehicle homes in, matching the velocity of the platform again shortly afterwards. The Descending state starts and the vehicle approximately matches the velocity of the platform until the Shutdown state. The platform heading can be seen remaining fairly constant through the landing procedure. The aircraft tracks the heading of the platform throughout the autoland procedure, with its heading not deviating by more than  $5^\circ$  of the platform's heading.

#### F.4.3 Attempt 4

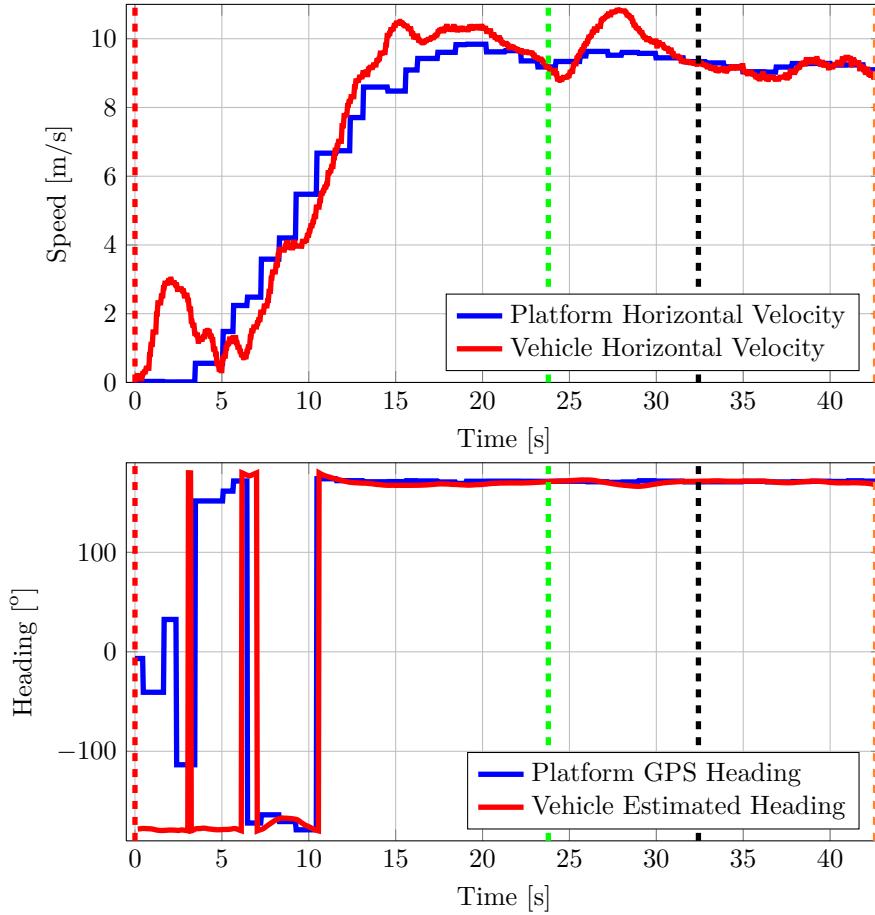
Results from the the fourth attempt at landing on a translating platform are presented here. The vehicle attempted to land on the translating platform while the platform was moving at a speed of  $40 \text{ km/h}$ . The autoland was not successful, as the accuracy of the position measurements of the DGPS was compromised during the test. The vertical position error recorded during the flight test is shown in Figure F.17. The ENU Up component is plotted, effectively indicating the vertical error between the vehicle and the platform.



**Figure F.17** – Vertical Error During  $40 \text{ km/h}$  Automated Landing Attempt

The landing procedure starts with the vehicle  $7 \text{ m}$  above the landing location. The vehicle descends during the Tracking state, where a jump in the vertical error measurement is seen at the time of 8 seconds into the test. From this point onwards, the accuracy of the relative position measurements was compromised. The aircraft attempts to maintain a height of what it thinks is  $3 \text{ m}$  above the landing location until the Descending state is entered. During the Descending state, the vehicle descends at a rate of  $0.5 \text{ m/s}$  until the safety pilot has to take control of the aircraft.

The top figure of Figure F.18 displays the measured velocities of the platform and the vehicle throughout the landing procedure. The bottom figure shows the heading measurements of both the platform and the vehicle.



**Figure F.18** – Platform and Vehicle Velocity, Heading, During 40 km/h Automated Landing Attempt

It can be seen that the platform's velocity has not reached the target speed of 40 km/h when the landing procedure starts. As explained, this was done to avoid running out of usable road during the 30 km/h and 40 km/h autoland attempts. During the Tracking state, the vehicle is already close to the platform, thus not requiring a high velocity to catch up to the platform. Eighteen seconds into the landing procedure, the vehicle matches the velocity of the platform. After the Homing state is entered, the vehicle speed increases slightly while the vehicle homes in, matching the velocity of the platform again shortly afterwards. The Descending state starts and the vehicle approximately matches the velocity of the platform until the safety pilot has to take control of the aircraft.

The platform heading can be seen acting sporadically while the speed of the platform is low. As the platform's speed increases the heading measurement stabilizes and the aircraft starts tracking the heading of the platform. The aircraft tracks the heading of the platform throughout the autoland procedure, with its heading not deviating by more than 5° from the platform's heading.

## Bibliography

- [1] Navy, S.A.: Equipment, Valour Class Frigates @ONLINE. 2014.  
Available at: <http://www.navy.mil.za/equipment/valour.htm>
- [2] Saripalli, S., Montgomery, J.F. and Sukhatme, G.S.: Visually-Guided Landing of an Unmanned Aerial Vehicle. 2003.  
Available at: <http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1206795>
- [3] Saripalli, S. and Sukhatme, G.S.: Landing a Helicopter on a Moving Target. In: *International Conference on Robotics and Automation Proceedings*. IEEE, April 2007. ISBN 1-4244-0602-1. ISSN 1050-4729.  
Available at: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4209384>
- [4] Wenzel, K.E., Masselli, A. and Zell, A.: Automatic Take Off, Tracking and Landing of a Miniature UAV on a Moving Carrier Vehicle. *Journal of Intelligent Robotic Systems*, 2010.  
Available at: <http://www.springerlink.com/index/10.1007/s10846-010-9473-0>
- [5] Voos, H. and Bou-Ammar, H.: Nonlinear tracking and landing controller for quadrotor aerial robots. In: *International Conference on Control Applications*. IEEE, 2010. ISBN 9781424453634.  
Available at: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5611204>
- [6] Wu, C., Song, D., Qi, J. and Han, J.: Autonomous Landing of an Unmanned Helicopter on a Moving Platform based on LP Path Planning. In: *8th International Conference on Intelligent Unmanned Systems Proceedings*, ICIUS. 2012. ISBN 9789810742256.
- [7] Friis, J., Nielsen, E., Andersen, R.F. and Jochumsen, A.: Autonomous Landing on a Moving Platform. 2009.
- [8] Olivares-Mendez, M.A., Mondragon, I.F., Campoy, P. and Martinez, C.: Fuzzy Controller for UAV-Landing Task Using 3D-Position Visual Estimation, 2010.
- [9] Swart, A.D.: Monocular Vision Assisted Autonomous Landing of a Helicopter on a Moving Deck. Master thesis, Stellenbosch University, 2013.  
Available at: <http://scholar.sun.ac.za/handle/10019.1/80134>
- [10] Garratt, M., Pota, H., Lambert, A., Eckersley-Maslin, S. and Farabet, C.: Visual Tracking and LIDAR Relative Positioning for Automated Launch and Recovery of an Unmanned Rotorcraft from Ships at Sea. *Naval Engineers Journal*, June 2009. ISSN 00281425.  
Available at: <http://www.ingentaconnect.com/content/asne/nej/2009/00000121/00000002/art00014>

- [11] Hu, X., Yu, L. and Feng, J.: Improved Image Target Detection Based on Component Analysis. *International Journal of Digital Content Technology and its Applications*, April 2012.
- [12] Yilmaz, A., Shafique, K., Lobo, N., Li, X., Olson, T. and Mubarak, S.: Target -Tracking In FLIR Imagery Using Mean-Shift and Global Motion Compensation, 2001.  
Available at: [http://crcv.ucf.edu/papers/yilmaz\\_cvbvs\\_2001.pdf](http://crcv.ucf.edu/papers/yilmaz_cvbvs_2001.pdf)
- [13] Pervan, B., Chan, F.-C., Gebre-Egziabher, D., Pullen, S., Enge, P. and Colby, G.: Performance Analysis of Carrier-Phase DGPS Navigation for Shipboard Landing of Aircraft. *Journal of The Institute of Navigation*, 2003.  
Available at: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.194.5925&rep=rep1&type=pdf>
- [14] DefenseMediaNetwork: Details of the X-47Bs First Autonomous Carrier Landing @ONLINE. 2014.  
Available at: <http://www.defensemedianetwork.com/stories/details-of-the-x-47bs-first-automated-carrier-landing/>
- [15] Hardesty, M., Kennedy, S., Dixon, S., Berka, T., Graham, J. and Caldwell, D.: Hardesty Development of Navigation and Automated Flight Control System Solutions for Maritime VTOL UAS Operations. In: *Association for Unmanned Vehicle Systems International*. 2006.  
Available at: [http://www.novatel.com/assets/Documents/Papers/Little\\_Bird\\_AUVSI\\_Conf\\_Paper.pdf](http://www.novatel.com/assets/Documents/Papers/Little_Bird_AUVSI_Conf_Paper.pdf)
- [16] Novatel: Tightly Coupled GNSS+INS Technology for Continuous 3D Position, Velocity and Attitude. 2014.  
Available at: <http://www.novatel.com/assets/Documents/Papers/SPANBrochure.pdf>
- [17] Balas, C.: Modelling and Linear Control Of a Quadrotor. Master Thesis, Cranfield University, 2007.  
Available at: [http://www.icrobotics.co.uk/wiki/images/6/66/Quadcopter\\_simulink\\_paper.pdf](http://www.icrobotics.co.uk/wiki/images/6/66/Quadcopter_simulink_paper.pdf)
- [18] Powers, C., Mellinger, D., Kushleyev, A., Kothmann, B. and Kumar, V.: Influence of Aerodynamics and Proximity Effects in Quadrotor Flight. 2013.  
Available at: [http://www.seas.upenn.edu/~dmel/ISER2012\\_powers.pdf](http://www.seas.upenn.edu/~dmel/ISER2012_powers.pdf)
- [19] Bresciani, T.: Modelling , Identification and Control of a Quadrotor Helicopter. Master Thesis, Lund University, 2008.  
Available at: <http://www.control.lth.se/documents/2008/5823.pdf>
- [20] Carrillo, G.L.R., Lopez, D.A.E., Lozano, R. and Pegard, C.: Quad Rotorcraft Control. 2013.  
Available at: <http://link.springer.com/10.1007/978-1-4471-4399-4>
- [21] Elraby, A.Y., El-khatib, M.M., El-Amary, N.H. and Hashad, A.I.: Dynamic Modelling and Control Of Quadrotor Vehicle. 2010.
- [22] Bristeau, P.-j., Martin, P., Salaün, E. and Petit, N.: The Role of Propeller Aerodynamics in the Model of a Quadrotor UAV. In: *European Control Conference Proceedings*. 2009. ISBN 9789633113691.  
Available at: <http://cas.ensmp.fr/~petit/papers/ecc09/PJB.pdf>

- [23] Schmaus, J., Berry, B., Gross, W. and Koliais, P.: Experimental Study of Rotor Performance in Deep Ground Effect with Application to a Human-Powered Helicopter. *American Helicopter Society 68th Annual Forum*, 2012.
- [24] Ganesh, B. and Komerath, N.: Unsteady Aerodynamics of Rotorcraft in Ground Effect. *American Institute of Aeronautics and Astronautics*, June 2004.  
Available at: <http://www.adl.gatech.edu/archives/adlp04062502.pdf>
- [25] Johnson, W.: *Helicopter Theory*. Dover Publications, 1994. ISBN 9780486682303.  
Available at: <http://books.google.co.za/books?id=0qBD9bhRShUC>
- [26] Moller, P.D.S.: Optimization of a Quadrotor Frame and Flight Time. 2011.
- [27] Blakelock, J.B.: *Automatic Control of Aircraft and Missiles*. 2nd edn. Wiley, 1991. ISBN 978-0471506515.  
Available at: <http://books.google.co.za/books?id=ubcczZUDCsMC>
- [28] Bouabdallah, S., Noth, A. and Siegwan, R.: PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor. In: *International Conference on Intellinet Robots and Systems Proceedings*. IEEE, 2004.  
Available at: <http://infoscience.epfl.ch/record/97531/files/330.pdf>
- [29] Pounds, P. and Mahony, R.: Design Principles of Large Quadrotors for Practical Applications. In: *International Conference on Robotics and Automation Proceedings*. IEEE, May 2009. ISBN 978-1-4244-2788-8.  
Available at: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5152390>
- [30] Novatel: OEM6 Family Firmware Reference Manual. 2012.  
Available at: [www.novatel.com/assets/Documents/Manuals/om-20000128.pdf](http://www.novatel.com/assets/Documents/Manuals/om-20000128.pdf)
- [31] Chalko, T.: Estimating Accuracy of GPS Doppler Speed Measurement using Speed Dilution of Precision (SDOP) parameter. 2009.
- [32] Neumann, J.B., Manz, A., Ford, T.J. and Mulyk, O.: Test Results from a New 2 cm Real Time Kinematic GPS Positioning System, 1996.  
Available at: <http://www.novatel.com/support/search/items/TechnicalPapers>