

Instrumentation Design, System Identification, and LQR/LQG Control of a
Small Scale Helicopter

by

Alejandro Martínez-González

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Mechanical Engineering
University of Alberta

© Alejandro Martínez-González, 2014

Abstract

The interest of Unmanned Aerial Vehicles (UAV) for civil application has increased dramatically in recent years. With improvement in computer hardware, lightweight sensors, and light material, the cost of UAVs has reduced significantly. UAVs with vertical takeoff and landing capabilities have been the focus of much development. A mathematical model for a scale RC helicopter including the rotor dynamics is presented in this work. The main rotor lift slope C_{L_α} , the main rotor aerodynamic drag coefficient C_{D_o} , and the moments of inertia about the center of gravity are determined by the use of two sets of HIL testbeds. The avionics instrumentation, communication and Ground Station system are described. Finally, the model linearization and the Linear Quadratic Regulator and Linear Quadratic Gaussian compensator in output feedback configuration with reference inputs are presented. Simulation results show that the proposed compensator stabilized and control the linearized dynamics of the helicopter.

Acknowledgments

I am indebted to my supervisor, Dr. C. Robert Koch for his invaluable support, assistance and guidance along my work.

I want to thanks Dr. Farbod Fahimi to bring me the opportunity to participate in this research and for his support and comprehension.

I want to be grateful in a special way to Dr. Larry W. Kostiuk, chair of Mechanical Department for his invaluable support, without which it had not been possible to finish my thesis.

I cannot forget all the enjoyable talks I had with my lab mates Masoud Mashkournia, Kahashayar Ebrahimi, and Ahmad. I really appreciate your ideas and comments. Special thanks to my lab mate Sepeher Pourrezai Kahaligh for all his ideas, knowledge, suggestions and comments that he shared with me along this work.

I would like to thanks all people in the machine Work Shop, particularly to Bernie Faulkner for his invaluable tips and support regarding all the hardware he made for us. To Rick Conrad from the Electronic Work Shop, for share his facilities and tips regarding the soldering of the avionic box and other related stuff. To Virginia Makofka, David Dubyk, and Wayne Pittman from the IT, that support me with all the software and systems requirements needed for the present research. Special thanks to Travis the pilot for his free help to fly the helicopter and for his tips and suggestion.

Finally and the most important to me, I would like to express all my love and gratitude to my wife Laura, to my son Diego and to my daughter Alinn for all the sacrifice you have done and for all the time I took away from you. I promise you I will recover all that time and more. Los amo, Ale.

Contents

List of Tables	vii
List of Figures	viii
Abbreviations	xiii
Nomenclature	xvi
Glossary	xx
1 Introduction	1
2 Helicopter's Mathematical Model	5
2.1 Reference Frame definitions	5
2.2 Rigid-body General motion equations	16
2.3 Helicopter Dynamic Model	20
2.4 Helicopter Aerodynamic model	23
2.4.1 Main Rotor Thrust	25
2.4.2 Main Rotor Drag Torque	31
2.4.3 Roll and Pitch Moments	34
2.4.4 Main Rotor in-plane Force	39
2.4.5 Tail Rotor Thrust	44
2.4.6 Tail Drag Torque	48
2.4.7 Tail In-plane Force	49
2.4.8 Aerodynamic Drag Force	51
2.5 Complete Helicopter Model	55
2.6 Helicopter Model Summary	61
3 MOI Testbed and HIL Force Testbed	62
3.1 MOI Testbed	62
3.1.1 Moment of Inertia Testbed Mathematical Model	63
3.1.2 Helicopter Mass Moment of Inertia Estimation	67
3.2 HIL Force Testbed	69
3.2.1 Force Testbed Equations	72
3.2.2 Servo Motors and Load Cell Calibration	75
3.2.3 Lower Swash-plate Kinematics	76

3.2.4	HIL experiments and Parameters ID	81
4	Helicopter's Instrumentation	87
4.1	ECICH, Avionic Box	89
4.1.1	Inertial Measurement Unit	91
4.1.2	Servo Switch Control Module	93
4.1.3	Communication Module	96
4.1.4	Voltage and Current Monitor Module	102
4.1.5	Laser Range Finder Module	113
4.1.6	Step Down Voltage Converter	117
4.1.7	Computer Module	119
4.2	Ground Station	125
4.3	ECICH Data Validation	132
4.4	Data Flight	156
5	System Linearization and Optimal LQR/LQG Controller	164
5.1	Model linearization	164
5.1.1	Linearization About Hover	166
5.2	LQR/LQG Controller	169
5.2.1	Optimal Linear Quadratic Regulator	172
5.2.2	State Observer and LQG Compensator	174
5.2.3	LQR/LQG Simulation Results	183
6	Conclusions	202
A	Transformations, WGS84 Datum and Model Coefficients	206
A.1	Direction Cosine Matrix	206
A.2	WGS84 Geodetic data	211
A.3	Affine-Form Coefficients	212
B	MOI Test Bed and HIL Force Bed	218
B.1	Test Bed Model	218
B.2	Pulley's and Rectangular shapes MOI computation and uncertainty analysis	220
B.2.1	Moment of inertia	220
B.2.2	Error propagation	221
B.3	Servo and Load Cell Calibration Outcomes	222
B.4	Homogenous Transformations and Solution	227
B.5	Linkage, Frame Dimensions and Angle Relations	230
C	ECICH and GS Technical Information	233
C.1	Electrical Diagram	233
C.2	BMS Plots	236
C.3	μ Controller program	239
C.4	MATLAB [®] Programs and Simulink Block Diagrams	256
C.4.1	SSC and RPM Programs	256

C.4.2	IMU Programs	262
C.4.3	Range Sensor Programs	267
C.4.4	BMS Programs	269
C.4.5	GS Programs	270
C.5	Ground Station Program	274
C.6	Mechanical Data	305
C.7	Data Flight Graphics	313
D	Helicopter Nonlinear Equations, Model Linearization and LQR Controller	319
D.1	Helicopter nonlinear equations.	319
D.2	Jacobian derivatives	328
D.3	Riccati Equation Derivation	352
	Bibliography	355

List of Tables

2.1	Air Star Evolution Helicopter's Parameters	60
3.1	MOI testbed parameters.	67
3.2	Helicopter's MOI.	68
4.1	IMU Characteristics and Settings Summary.	94
4.2	SSC Switch Table Settings.	96
4.3	ECICH to GS frame	103
4.4	GS to ECICH frame	103
4.5	Most common commercial LiPo batteries	107
4.6	Communication protocol frames between PC/104™ and GS.	112
4.7	Horizontal Position Statistical Summary	138
4.8	Statistical Summary based on the Mahalanobis Distance	140
4.9	Enclosed areas by the ellipses.	141
4.10	Three Dimension Space Statistical Summary	143
4.11	Magnitude Drift Error and Drift Error Speed.	149
5.1	Hover equilibrium values for hover	168
5.2	Eigenvalues of the open-loop system (5.5)	170
5.3	Selected maximum value of the states and the command inputs	174
A.1	WGS84 Four Defining Parameters	211
A.2	WGS84 Ellipsoid Geometric Constants	211
B.1	Measurement Instruments Characteristics.	221
B.2	MOI bed physical characteristics	223
B.3	Servo Calibration Outcome Summary.	224
B.4	Load Cells Calibration Outcome Summary.	225
B.5	Linkages and frame dimension for the lower swash-plate	230
B.6	Functional relation coefficients	231

List of Figures

1.1	The bombing balloon.	1
2.1	Inertial and NED frames.	9
2.2	Aircraft's frames	10
2.3	Euler angles rotations.	12
2.4	Helicopter forces and moments	18
2.5	Element of blade on point p.	24
2.6	Velocity of the element p	25
2.7	Vertical and induced velocity	28
2.8	Induced roll moment	34
2.9	Induced pitch moment	35
2.10	Tail rotor and body frames	45
2.11	Drag classification	53
2.12	Vertical Fin Shape.	53
2.13	Zero-Lift Drag Coefficient for a circular cylinder	54
3.1	MOI testbed general layout.	63
3.2	Pitch moment estimation.	65
3.3	Helicopter mounted for roll moment estimation. Partial view.	66
3.4	Helicopter mounted for roll moment estimation. Full view.	66
3.5	Mass Moment Of Inertia Setup and Simulink [®] diagram.	67
3.6	Position and velocity data for the pitch moment experiment	69
3.7	Position and velocity data for the roll moment experiment	70
3.8	Position and velocity data for the yaw moment experiment	71
3.9	Block diagram of the Force Bed Setup	72
3.10	Some aspects of the Force Bed HIL	73
3.11	Force testbed free body diagram	74
3.12	Schematic diagram of the swash-plate mechanism.	77
3.13	$\delta_{\theta_{col}}$ vs. z_c with digital inclinometer	81
3.14	$\delta_{\theta_{col}}$ vs. z_c with incremental encoder	82
3.15	HIL force testbed xPC Target [™] main program block diagram	82
3.16	Throttle input and RPM signals used in several test.	83
3.17	HIL force testbed thrust test outcomes	85
3.18	Main rotor thrust and drag force from the HIL force testbed	86

4.1	ECICH Block Diagram	90
4.2	Different aspects of the ECICH box.	91
4.3	IMU Hardware block diagram.	92
4.4	IMU Software block diagram.	92
4.5	SSC block diagram.	95
4.6	XStream [®] RF modems accumulative spectrum.	97
4.7	Dipole antenna radiation pattern.	99
4.8	Communication protocol general layout.	99
4.9	Payload blocks of the ECICH frame.	100
4.10	Battery Monitor System Block Diagram.	108
4.11	Voltages and currents plots from flight No. 5.	109
4.12	Voltage Vs. Current Characteristic	110
4.13	Protocol frames between GS and μC	111
4.14	Range Sensor Scanning Area	113
4.15	LRF Area Set and Pitch Compensation.	114
4.16	LRF Communication Protocol Frame.	115
4.17	Details of the LRF enclosure attached to the Avionic Box.	116
4.18	Block Terminal Connections for Servos, RPM and ESC.	118
4.19	ECICH interruptors sets.	118
4.20	PC104 Program Block Diagram	121
4.21	Detailed Connection of ECICH and Helicopter' Systems	125
4.22	GS Main Screen General Layout.	127
4.23	First Data Set Histogram and Scatter pastern.	134
4.24	Second Data Set Histogram and Scatter pastern.	135
4.25	Third Data Set Histogram and Scatter pastern.	136
4.26	Stationary Position PDF and CDF.	139
4.27	Position Scatter Pattern for the three data set.	142
4.28	3D Position Scatter Pattern for the first data set.	144
4.29	3D Position Scatter Pattern for the first data set.	145
4.30	3D Position Scatter Pattern for the first data set.	146
4.31	Error Position Drift Vs. Time for data set one.	148
4.32	Three minutes data chunk after 3 hours.	150
4.33	Three minutes data chunk after 17 hours.	150
4.34	Car ride track trajectory.	152
4.35	Car ride track trajectory zoom.	152
4.36	ECICH velocities validation.	153
4.37	Magnitude Velocity Histogram.	154
4.38	RPM Speed and RPM Histogram.	154
4.39	LRF Range Output and LRF Histogram.	154
4.40	BMS Voltage Output and BMS Histogram.	155
4.41	BMS Current Output and BMS Histogram.	155
4.42	Fifth Flight Euler angles.	156
4.43	Fifth Flight Body angular rates.	157
4.44	Fifth Flight Accelerations coming out from the IMU.	157
4.45	Fifth Flight Frame \mathbf{C}_N velocities.	158

4.46	Fifth Flight GPS position.	159
4.47	Fifth Flight Throttle and main rotor RPM.	159
4.48	Fifth Flight Pilot servo commands.	160
4.49	Main rotor collective angle	160
4.50	Cyclic lateral command	161
4.51	Cyclic longitudinal command	161
4.52	Fifth Flight Helicopter GPS altitude.	162
4.53	Fifth Flight Helicopter trajectory in the North-East plane.	162
4.54	Fifth Flight Helicopter trajectory on space.	163
5.1	Pole map of the Open-Loop linearized dynamics of the Air Star Evolution helicopter.	170
5.2	u vs. command open-loop impulse response	171
5.3	v vs. command open-loop impulse response	171
5.4	w vs. command open-loop impulse response	172
5.5	Block diagram of the LQR implementation.	172
5.6	Closed-loop control inputs due to an impulse in body speed u	175
5.7	Closed-loop control inputs due to an impulse in body speed v	175
5.8	Closed-loop control inputs due to an impulse in body speed w	176
5.9	Closed-loop control inputs due to an impulse in roll rate p	176
5.10	Closed-loop control inputs due to an impulse in pitch rate q	177
5.11	Closed-loop control inputs due to an impulse in heading rate r	177
5.12	Closed-loop control inputs due to an impulse in roll angle ϕ	178
5.13	Closed-loop control inputs due to an impulse in pitch angle θ	178
5.14	Closed-loop control inputs due to an impulse in heading angle ψ	179
5.15	Block diagram of the LQR/LQG compensator.	180
5.16	Response of u , v and w to a set point in u	184
5.17	Response of p , q and r to a set point in u	185
5.18	Response of u , v and w to a set point in v	185
5.19	Response of p , q and r to a set point in v	186
5.20	Response of ϕ , θ and ψ to a set point in v	186
5.21	Response of u , v and w to a set point in w	187
5.22	Response of p , q and r to a set point in w	187
5.23	Response of ϕ , θ and ψ to a set point in ψ	188
5.24	Response of p , q and r to a set point in ψ	188
5.25	Response of u , v and w to a set point in ψ	189
5.26	Body velocities response to a set point in u added with noise	190
5.27	Angular velocities response to a set point in u added with noise	190
5.28	Euler angles response to a set point in u added with noise	191
5.29	Body velocities response to a set point in v added with noise	191
5.30	Angular velocities response to a set point in v added with noise	192
5.31	Euler angles response to a set point in v added with noise	192
5.32	Body velocities response to a set point in w added with noise	193
5.33	Angular velocities response to a set point in w added with noise	193
5.34	Euler angles response to a set point in ψ added with noise	194

5.35	Angular velocities response to a set point in ψ added with noise	194
5.36	Body velocities response to a set point in ψ added with noise . .	195
5.37	Comparison between u and flight velocity u	196
5.38	Comparison between v and flight velocity v	197
5.39	Comparison between w and flight velocity w	198
5.40	Comparison between p and flight velocity p	198
5.41	Comparison between q and flight velocity q	199
5.42	Comparison between r and flight velocity r	199
5.43	Comparison between ϕ and flight roll angle ϕ	200
5.44	Comparison between θ and flight pitch angle θ	200
5.45	Comparison between ψ and flight heading angle ψ	201
A.1	DCM inner products angle computation	210
B.1	Calibration servo response	224
B.2	Longitudinal load cell responses.	225
B.3	Lateral and horizontal load cell responses.	226
B.4	Blade $\delta_{\theta_{col}}$ response and z_c input vs. time	230
B.5	Servo angles vs. z_c	231
B.6	Blade collective pitch angle $\delta_{\theta_{col}}$ vs. servo angles	232
C.1	ECICH Electrical Diagram page 1 of 3	233
C.2	ECICH Electrical Diagram page 2 of 3	234
C.3	ECICH Electrical Diagram page 3 of 3	235
C.4	Five flights ECICH's Battery Voltages Plots	236
C.5	Five flights ECICH's Battery Current Plots	236
C.6	Five flights Rotor's Battery Voltages Plots	237
C.7	Five flights Servos' Battery Voltages Plots	238
C.8	Five flights Servos' Battery Current Plots	238
C.9	Simulink [®] PWM Computation from Servo Switch Card Block.	256
C.10	Simulink [®] Pilot's Commands Storage Block.	256
C.11	Simulink [®] PC104 Controller Block.	257
C.12	Simulink [®] CRC computation and Packet Transmission	258
C.13	Simulink [®] Servo Switch Card Data Request Block.	259
C.14	Simulink [®] IMU Data Block.	262
C.15	Simulink [®] IMU Data Storage Block.	263
C.16	Simulink [®] Crossbow Data Request Block.	263
C.17	Simulink [®] Storage Range Sensor Altitude Data Block.	267
C.18	Simulink [®] Range Computation Block.	267
C.19	Simulink [®] Voltage Current Monitor Block.	269
C.20	Simulink [®] Ground Station Data Transmission Block.	270
C.21	Simulink [®] Ground Station Command Execution Block.	271
C.22	ECICH General Layout.	305
C.23	PC/104 [™] Dimensions.	306
C.24	NAV440 [™] Dimensions.	307

C.25 SSC Connector Dimensions.	308
C.26 Radio Modem OEM module Dimensions.	308
C.27 PCB Dimensions.	309
C.28 PCB Top Side tracks. Not at scale.	310
C.29 PCB Bottom Side tracks. Not at scale.	311
C.30 PCB Components Layout. Not at scale.	312
C.31 Euler Angels, body angular rates and accelerations and NED velocities	313
C.32 GPS position, Throttle command, main rotor RPM and pilot servo commands from second flight.	314
C.33 Helicopter Trajectory and altitude from second flight.	315
C.34 Euler angles, body angular rates, body accelerations and NED velocities data from fourth flight.	316
C.35 GPS position, Throttle command, main rotor RPM and pilot servo commands from fourth flight.	317
C.36 Helicopter Trajectory and altitude from fourth flight.	318

Abbreviations and Acronyms

Abbreviation	Description
A/D	Analog to Digital Converter
ADI	Attitude Director Indicator
ASCII	American Standard Code for Information Interchange
BFSK	Binary Frequency Shift Keying
BLOS	Beyond line of sight
BMS	Battery Monitor System
bps	Bits Per Second
CCITT	Consultative Committee on International Telephone and Telegraph
CDF, cdf	Cumulative Density Function
CF	Compact Flash Card or Memory
cg	Center of Gravity
CDMA	Code Division Multiple Access
CEP	Circular Error Probable
CR	Carriage Return
CRC	Cyclic Redundancy Check
DCM	Direct Cosine Matrix
DGPS	Differential Global Position System
DOF	Degree Of Freedom
DSP	Digital Signal Processor
E-W	East-West
ECEF	Earth-Centered-Earth-Fixed frame
ECICH	Electronic Control, Instrumentation and Communication Hardware
EEP	Ellipse or Ellipsoid Error Probable
EEPROM	Electrical Erasable Programmable Read Only Memory
EKF	Extended Kalman Filter
EOF	End Of Frame
ESC	Electronic Speed Controller
FHSS	Frequency-Hopping Spread Spectrum
FIR	Finite Impulse Response
FPGA	Field-Programmable Gate Array
GS	Ground Station

Abbreviation	Description
GPS	Global Position System
Hex, hex	Hexadecimal
HIL	Hardware-In-the-Loop
IC	Integrated Circuit
LF	Line Feed
LiPo	Lithium-Polymer batteries
lla	latitude-longitude-altitude
LOS	Line Of Sight
LQG	Linear Quadratic Gaussian Compensator
LQR	Linear Quadratic Regulator
LRF	Laser Range Finder
MRA	Miniature Radar Altimeter
μ C	micro-Controller
MEMS	Micro Electro-Mechanical Systems
MOI	Moment Of Inertia
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
N-S	North-South
NED	North-East-Down frame or Earth-Surface Local Tangential frame
OEM	Original Equipment Manufacturer
PCB	Printed Circuit Board
PDF, pdf	Probability Density Function
PWM	Pulse Width Modulation
RAM	Random Access Memory
RC	Radio Control
RCDI	Rate of Climb and Descent Indicator
RF	Radio Frequency
rms	Root Mean Square value
RPM	Revolutions Per Minute
RS-232	Electrical Interface for Asynchronous Serial Communication
SDC	DC to DC Step Down Voltage Converter
S/N	Signal to Noise ratio
SPI	Serial Peripheral Interface
SSC	Servo Switch Controller
SSP	Synchronous Serial Port
Sync	Synchronization
TPT	Tip-Path-Plane
UAS	Unmanned Aerial Systems
UAV	Unmanned Aerial Vehicles
VGA	Video Graphics Array
VLSI	Very Large Scale of Integration
VSI	Vertical Speed Indicator

Abbreviation	Description
WAAS	Wide Area Augmentation System
WGS84	World Geodetic System 84

Nomenclature

Symbol	Unit	Description
A		Any capital letter in boldface is a matrix or tensor
\mathcal{R}		Main blade Aspect Ratio
\mathcal{R}_t		Tail blade Aspect Ratio
a		Any lowercase letter in boldface is a vector
${}^b\mathbf{a}$		Vector expressed in the frame indicated by the leading superscript
a	m	Earth's Semi-major Axis
b	m	Main rotor blade span
b_t	m	Tail rotor blade span
c	m	Cord of the main blade
\mathcal{C}_b		Body frame
\mathcal{C}_{bl}		Main rotor or blade frame
\mathcal{C}_E		Earth-Centered-Earth-Fixed frame
\mathcal{C}_i		Inertial frame
\mathcal{C}_{lla}		longitude-latitude-altitude frame
\mathcal{C}_N		North-East-Down or surface tangential frame
\mathcal{C}_t		Tail frame
\mathcal{C}_v		Vehicle-Carried frame
C_{D_i}		Main rotor induced drag coefficient
$C_{D_{i_t}}$		Tail rotor induced drag coefficient
C_{D_o}		Main rotor zero-lift drag coefficient
$C_{D_{o_t}}$		Tail rotor zero-lift drag coefficient
C_L		Main blade Lift coefficient
C_{L_α}	$\frac{1}{\text{rad}}$	main blade Lift Slope
C_{L_t}		Tail blade Lift coefficient
C_{L_α}	$\frac{1}{\text{rad}}$	Tail blade Lift Slope
\mathbf{C}_i^b		Transformation matrix from inertial to body frame
\mathbf{C}_b^i		Transformation matrix from body to inertial frame
\mathbf{C}_{bl}^b		Transformation matrix from blade to body frame

Symbol	Unit	Description
\mathbf{C}_E^N		Transformation matrix from ECEF to NED frame
\mathbf{C}_N^E		Transformation matrix from NED to ECEF frame
\mathbf{C}_{lla}^E		Transformation matrix from lla to ECEF frame
\mathbf{C}_{EEP}		Ellipsoid Error Probable Rotation matrix
$\mathbf{C}_\omega^{\dot{\phi}}$		Transformation matrix from ω to Euler rates vector $\dot{\phi}$
$\frac{{}^b d(\cdot)}{dt}$		Time derivative taken in the frame indicated by the leading superscript
\mathbf{e}_n		Orthonormal basis
e		Main blade Oswald number
e_t		Tail blade Oswald number
\mathbf{F}_D	N	Aerodynamic drag
h	m	Altitude above the reference ellipsoide
\mathbf{h}_r	$\frac{\text{kg}\cdot\text{m}^2}{\text{s}}$	Main rotor angular momentum
\mathbf{h}_t	$\frac{\text{kg}\cdot\text{m}^2}{\text{s}}$	Tail rotor angular momentum
\mathbf{H}_r	N	Main rotor in-plane force
\mathbf{H}_t	N	Tail rotor in-plane force
\mathbf{I}		Inertia Tensor
\mathbf{i}		Unit vector in the direction of x axis
I_x	$\text{kg}\cdot\text{m}^2$	Helicopter's Roll moment. About x_b axis
I_y	$\text{kg}\cdot\text{m}^2$	Helicopter's Pitch moment. About y_b axis
I_z	$\text{kg}\cdot\text{m}^2$	Helicopter's Yaw moment. About z_b axis
$I_{x_{bl}}$	$\text{kg}\cdot\text{m}^2$	Main blades's mass MOI about x_{bl} axis
$I_{y_{bl}}$	$\text{kg}\cdot\text{m}^2$	Main blades's mass MOI about y_{bl} axis
$I_{z_{bl}}$	$\text{kg}\cdot\text{m}^2$	Main blades's mass MOI about z_{bl} axis
\mathbf{j}		Unit vector in the direction of y axis
\mathbf{k}		Unit vector in the direction of z axis
m	kg	Helicopter's mass excluding the main rotor blades
m_r	kg	Main rotor blades' mass
\mathbf{M}_D	N·m	Moment due to the aerodynamic drag
\mathbf{M}_f	N·m	Moment due to the fly bar
\mathbf{M}_{H_t}	N·m	Moment due to the tail rotor in-plane force
\mathbf{M}_{H_r}	N·m	Moment due to the main rotor in-plane force
\mathbf{M}_t	N·m	Moment due to the tail rotor thrust
\mathbf{M}_θ	N·m	Pitch moment
\mathbf{M}_ϕ	N·m	Roll moment
p, q, r	$\frac{\text{rad}}{\text{s}}$	Roll, Pitch, and Yaw Rate
$\dot{p}, \dot{q}, \dot{r}$	$\frac{\text{rad}}{\text{s}^2}$	Roll, Pitch, and Yaw Acceleration
\mathbf{Q}_r	N·m	Drag torque due to the main rotor

Symbol	Unit	Description
\mathbf{Q}_t	N·m	Drag torque due to the tail rotor
R	m	Main rotor disc
R_h	m	Main rotor hinge offset or length
S		Main rotor solidity factor
S_t		Tail rotor solidity factor
\mathbf{T}_r	N	Main rotor Thrust
\mathbf{T}_t	N	Tail rotor Thrust
u, v, w	$\frac{\text{m}}{\text{s}}$	Body frame linear velocities along $x_b, y_b,$ and z_b
u_w, v_w, w_w	$\frac{\text{m}}{\text{s}}$	Component of wind velocities along $x_i, y_i,$ and z_i
$\dot{u}, \dot{v}, \dot{w}$	$\frac{\text{m}}{\text{s}^2}$	Body frame linear accelerations along $x_b, y_b,$ and z_b
v_N, v_E, v_D	$\frac{\text{m}}{\text{s}}$	NED frame velocities
\mathbf{W}	N	Helicopter total weight
x_b, y_b, z_b	m	Body frame axes coordinates
x_{bl}, y_{bl}, z_{bl}	m	Main rotor or blade frame axes coordinates
x_E, y_E, z_E	m	ECEF frame axes coordinates
x_i, y_i, z_i	m	Inertial frame axes coordinates
x_{tcg}	m	Distance in the x_b axis from the helicopter's cg to the tail rotor
y_{tcg}	m	Distance in the y_b axis from the helicopter's cg to the tail rotor
z_{tcg}	m	Distance in the z_b axis from the helicopter's cg to the tail rotor
z_{rcg}	m	Distance in the x_b axis from the helicopter's cg to the main rotor
Greek		
α	rad	Angle of attack of the main blades
α_1	N·m	
α_2	N·m·s	
α_3	N·m	
α_4	N·m·s	
ζ	rad	Latitude
η	rad	Angle of the oncoming air stream respect to the horizontal
λ	rad	Longitude
λ_i		Normalized induced velocity
Ξ		Covariance matrix
ξ		Cumulative probability
ρ	$\frac{\text{Kg}}{\text{m}^3}$	Air density
σ		Standard deviation
σ^2		Variance

Symbol	Unit	Description
ϕ, θ, ψ	rad	Euler angles: roll, pitch and yaw
$\dot{\phi}, \dot{\theta}, \dot{\psi}$	$\frac{\text{rad}}{\text{s}}$	Euler rate angles
φ	rad	Euler angel vector
χ	m	Inertial position vector
χ^2		Chi square distribution
Ψ	rad	Main blades azimuth angle
ω	$\frac{\text{rad}}{\text{s}}$	Angular velocities vector
ϖ	rad	Earth's rotation angle
Ω	$\frac{\text{rad}}{\text{s}}$	Main rotor angular speed

Glossary

- BFSK** Binary Frequency Shift Keying, is a digital modulation technique in which the information is transmitted in the changes of frequency of a carrier signal, usually an analog sinusoidal signal. The amount of information that can be transited depends of the number of bits employed to form the symbols, and the communication channel bandwidth., 95
- CDMA** Code Division Multiple Access is a communication technique in which the communication channel is divided to allow several simultaneous transmission over the same communication channel without interference. Each peer of transceiver has a unique Code to Access the channel, therefore any other transmission with different code is interpreted as noise and rejected by the receiver., 95
- CRC** Cyclic Redundance Check is an error detection technique used in communication protocols. It consist of add to the transmission frame a fix length of bits, between 8 to 32, depending on the polynomial or algorithm. The basic CRC are capable to detect contiguous errors induced by noise but cannot correct them. More advanced CRC technique are capable not only to detect error, but correct them without retransmission request, saving time in the total transmission time., 98, 100–102, 111, 114, 122
- CR** Carriage Return is a widely used ASCII character, with an hexadecimal value of 0x0D, used in communication protocols and printers. It is use to indicate to the printer to return the carriage to its home position. This character in combination with the Line Feed character is what is know as an “Enter” in the computers keyboards., 102, 112, 114

- EEPROM** Electrical Erasable Programable Read Only Memory is a memory integrated circuit that can be erased and programmed by using specific voltage levels applied during certain time intervals. It is a non-volatile memory and usually can retain data over a periods for 10 to 100 years with out refreshing. Most of them are capable to perform over 10,000 erasing-witting cycles or more., 91, 93, 121
- EOF** End OF File is a widely use ASCII character with an hexadecimal value of 0x1A that indicate the end of transmitted file in communication protocols, 98, 101, 102, 112, 114, 115
- FHSS** Frequency-Hopping Spread Spectrum is a combination of two communication methods, the frequency hopping, and the Spread spectrum. In the frequency hopping, the transmission frequencies are changed rapidly in a psudorandom pattern known from both transceivers. The spread spectrum is a technique in which the information is added with more bandwidth but keeping the same transition power. In this way, the signal to noise ratio, S/N is reduced making difficult to recover without the proper psudorandom pattern., 95
- FPGA** Field Programable Gate Array is an integrated circuit that contains hundreds of logic gates and possesses reconfigurable connections. This allows the chip to be programmed to implement simple logic functions like flip-flops up to complex arithmetic logic units an even small μ processors whit one or two multiplier-accumulation units., 88
- Hex** Hexadecimal numbers. Is a 16 digits base numeration system, form 0, . . . , 9, A, . . . , F, intended to codify the binary numbers for easy interpretation. An 8 bits binary number, known as byte, is divided in two sets of 4 bits known as a nibbles. Each nibble, the upper and the lower is represented by a couple of hexadecimal numbers., 101, 102, 111, 112, 114, 115, 122, 123

- LF** Line Feed is a widely used ASCII character, with an hexadecimal value of 0x0A, used in communication protocols and printers. It is used to indicate to the printer that a new line needs to be fed. This character in combination with the carriage return character is what is known as the “Enter” key in the computer keyboards., 102, 112, 114, 115
- MOSFET** Metal Oxide Semiconductor Field Effect Transistor. Is a construction technique to manufacture transistors in which a thin oxide-silicon layer is deposited to isolate the gate from the substrate and induce the channel between the drain and source terminal., 117
- PCB** Printed Circuit Board. Normally is a FR-4 thin board laminated with copper on one or two sides or in some cases with several inner layers of copper, and is used to provide mechanical support and electrically connect electronics components between them., 90, 118
- PWM** Pulse Width Modulation. Is a modulation technique in which the information to be transmitted is contained in the pulse width of a train of square pulses. The main idea is to change the duty cycle of square pulses proportional to the information., 93, 104, 118, 120–122
- RS-232** RS-232 refers to the electrical interface characteristics of the signals intended to be used in data serial communication between data terminal equipments and/or data communication devices. It is regulated by the Consultative Committee on International Telephone and Telegraph (CCITT)., 88, 97, 112
- SPI** Serial Peripheral Interface refers to a communication interface between μ Controllers and μ Processors in which two or three physical connections are needed. It is a high speed communication, several MBps, and is intended for short distance, normally within the PCB layout., 88
- SPP** Synchronous Serial Port refers to a specific type of μ Processor or μ Controller port in which a Clock signal is needed to transmit data between μ Controller, μ Processors, or peripheral IC., 88
- VGA** Video Graphics Array is a graphic video display system for personal computers and provides a graphic resolution up to 720×400 pixels., 119

- class 6** Refers to the classification of calibration weights given by the National Institute of Standards and Technology (NIST), Handbook 44., 62, 67, 75
- collective** The collective pitch angle is the angle applied to the main rotor blades that causes an increase or decrease in the total lift by changing the angle of attack of the blades. This angle is applied evenly to the main rotor blades and is independent of the cyclic position of the blades. In other words, this angle is the same during the whole rotation of the blades., 25
- cyclic** The cyclic pitch angle is a change in the angle of attack of the blades that happens in a specific position on the cycle of rotation. This causes that the helicopter to bank (roll motion) or to pitch (nose up or down) depending of the cyclic lateral or longitudinal command., 25
- dB_i** Are antenna gain units referred to the ideal isotropic antenna which radiates electromagnetic waves energy uniformly in all direction., 95

Chapter 1

Introduction

The interest in Unmanned Aerial Vehicles (UAV)s has been present before the invention of airplanes. In 1786 the Montgolfire brothers launch an unmanned hot air balloon [1]. The idea of unmanned aerial vehicle took form for military applications like the “The Bombing Balloon” in 1849 and pilot less aircraft “Aerial Target” in World War I in 1916 [2]. The aerial torpedo was developed

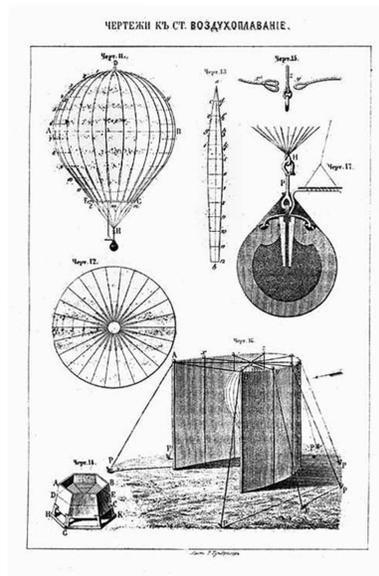


Figure 1.1: *The bombing balloon was the first UAV ever registered on a document in 1849, [2] on The Press, of Vienna. The document said: “Venice is to be bombarded by balloons, as the lagunes prevent the approaching of artillery. Five balloons, each twenty-three feet in diameter, are in construction at Treviso. In a favorable wind the balloons will be launched and directed as near to Venice as possible, and on their being brought to vertical positions over the town, they will be fired by electro magnetism by means of a long isolated copper wire with a large galvanic battery placed on the shore. The bomb falls perpendicularly, and explodes on reaching the ground.” Image source from: http://www.ctie.monash.edu/hargrave/images/balloonbombs1848_500.jpg*

shortly after World War I in 1930, and the glide bomb known as “Buzz Bomb” controlled by radio waves was used at the end of the World War II [3]. The first UAV was used for surveillance in Vietnam War, and UAVs were used by the Israeli Air force in the early stage of UAV development [4].

Although UAVs were initially restricted to military use, with the rapid improvements in the semiconductors and micro-machine industry, improvements in mechanical systems and light materials, the use of Unmanned Aerial Systems (UAS) has become popular and affordable for the general public. This has provided the motivation for civilian applications. As a result, many researchers and private companies are using mid and small scale Radio Controlled (RC) aircraft, helicopters and quadrotors for civil application within Line Of Sight (LOS) distance.

Although the dynamic behavior and parameters of full scale aircraft and helicopters are well established, they cannot be applied directly to small scale RC counterparts due to their rapid dynamics. In particular, small scale helicopters are configured differently to have fly bar stabilizer, a hinge-less rotor mechanism, and relative low or even negligible flapping dynamics. They can perform extreme flight maneuvers like inverted flight, 360° pitch rotations, 90° roll angle flight, due to their high power to weight ratio, and lack of pilot. Modeling and parameter identification of small scale helicopters is detailed in [5] – [21]. However, the current small scale dynamic models of helicopters can be improved by adding the rotor dynamics to the model. In the present work, the proposed model in [22] that includes the rotor dynamics is presented and derived in detail. This model is used as a basis for subsequent model linearization, parameter identification and controller design.

Despite that many UAVs are currently operational under military bases, flight Beyond Line of Sight (BLOS) remains a challenge even in military application [3]. A complete and well established infrastructure including human and material resources is required. Strategies for launching, radar tracking, airport coordination, radio and satellite telemetry, collision avoidance systems and vehicle recovery are needed to minimize vehicles loses and avoid collateral damage [3]. Due to the lack of all this infrastructure for civil applications, countries like Canada have banned flight BLOS for RC vehicles and UAVs [23].

Although there is an increasing market for civil applications, it is necessary to solve all of these problems before flight BLOS for civil applications will be allowed. Telemetry, radio link and ground station infrastructure is emphasized in this work to address part of these problems. A detailed explanation of the telemetry parameters that are needed to monitor the helicopter like system voltage and current, main rotor RPM, Euler angles, body velocities and many others are presented. The communication protocol is detailed and the ground station is thoroughly explained. Finally all the hardware and software involved in the avionics box is presented.

The thesis is organized as follows:

In chapter two, the mathematical model of the Air Star Evolution helicopter [22] is presented in detail to provide a basis for understanding the complex dynamics of helicopters. The derivation begins with a description of the reference frames and the helicopter frames in section 2.1, followed by a brief discussion of the rigid-body motion equations and the involved forces/-moments in section 2.2. Then the dynamics of the helicopter using the total angular momentum is derived in section 2.3. In section 2.4 the aerodynamics forces of the helicopter are introduced using the blade element theory and the momentum theory. The main rotor thrust, main rotor torque, roll and pitch moments, main rotor in-plane force, tail rotor thrust, tail drag torque, tail in-plane force and vertical fin aerodynamic force are derived. Finally the complete non-linear model of the small scale helicopter is presented.

The setup to estimate the moments of inertia of the helicopter, the main rotor lift slope $C_{L\alpha}$ and the main rotor zero-lift drag coefficient C_{D_o} is described in chapter 3. The equations of the Moment Of Inertia (MOI) testbed and required instrumentation is described in section 3.1. The experiments to determine the corresponding moment I_x , I_y and I_z in the body frame are also described. Then, in section 3.2 the Hardware-In-The-Loop (HIL) force testbed is detailed. The HIL testbed equations and the servo motor calibration are presented. An analysis of the lower Swash-Plate kinematics that follows [24] is done for the Air Star Evolution four servomotor swash-plate. The chapter ends with a description of the experiments done to estimate the main rotor lift slope and zero-lift drag coefficient ($C_{L\alpha}$ and C_{D_o}).

The helicopter instrumentation, the communication protocol and the ground station program are described in chapter 4. All the components that are integrated in the avionics box are presented. Section 4.1 begins with a general layout description of the avionics box. In the following subsections, the Inertial Measurement Unit (IMU), the servo switch controller, the communication module, the voltage and current monitor system, the Laser Range Finder module, the step down voltage converter system and the computer module are described. The ground station components and the communication protocol are described in detail in section 4.2. Experiments to validate the information coming from the avionics are in section 4.3. An analysis of the GPS accuracy and drift is done along with other experiment to validate the IMU and other systems of the avionics box. Finally data collected from different helicopter flights is described in section 4.4.

Model linearization of the non-linear dynamics of the helicopter described in chapter 2, is detailed in chapter 5. Then the implementation and simulation of an optimal linear Quadratic Regulator (LQR) and a Linear Quadratic Compensator (LQG) is described. In section 5.1, model linearization about hover using the small perturbation method is presented. The LQR/LQG controller is presented in section 5.2. First, the LQR is computed followed by the design of a full state Kalman observer, based on the separation principle. By combining the LQR and the Kalman observer the implementation of an output feedback controller is realized. Finally three sets of simulations are presented and discussed.

Chapter 2

Helicopter's Mathematical Model

The non-linear dynamic model of the Air Star Evolution RC helicopter given by [22] is derived in detail in this chapter. Equations of motion, aerodynamic model and main rotor effect will be described thoroughly. To get a better understanding in the key parameters that affect the helicopter's behavior, the equations of motion, aerodynamics and the main rotor will be described in detail.

To derive a mathematical model of the helicopter and find its parameters it is necessary to define a consistent set of reference frames in which Newton-Euler equations are derived. Section 2.1 defines all the necessary reference frames, as well as the helicopter's reference frames and conventions used throughout the thesis. In Section 2.2 the rigid-body general motion equations and the external forces and moment acting on the helicopter are stated. Section 2.3 defines the dynamic model used for the helicopter. Section 2.4 details the aerodynamic forces including the effect of the main rotor dynamics. Finally, in section 2.5 the complete affine non-linear model given by [22] is presented along with a summary of its parameters.

2.1 Reference Frame definitions

Due to the fact that Earth is spinning around its axis orbiting the sun, it is necessary to define an inertial reference frame in which Newton-Euler equations can be derived. This reference frame must be both non-rotating and non-accelerating. Nevertheless, the entire universe is in expansion, the time scales

in which this happens is extremely large compared to our 15 to 30 minutes flight, and even more. As a result of this, it is possible to consider all the stars as fixed points in the universe. Therefore, it is possible to define the inertial frame \mathbf{C}_i , with orthonormal basis ${}^i\mathbf{e}_k$, for $k = 1, 2, 3$, as a frame attached in some point in the universe.

As a convention, for any given frame, the orthonormal basis ${}^\varsigma\mathbf{e}_k$ will be defined as follows: ${}^\varsigma\hat{\mathbf{e}}_1 = {}^i\mathbf{i}_\varsigma$ for x axis, ${}^\varsigma\mathbf{e}_2 = {}^i\mathbf{j}_\varsigma$ for y axis, and ${}^\varsigma\hat{\mathbf{e}}_3 = {}^i\mathbf{k}_\varsigma$ for z axis, where the subscript and leading superscript ς in each unit vector and basis vector indicate to which frame the basis belongs.

Along the present work, the following frames were used:

- Inertial Frame
- Earth-Centered-Earth-Fixed Frame
- North-East-Down Frame
- Body Frame
- Vehicle-Carried Frame
- Blade Frame
- Tail Frame
- Geodetic or Latitude-Longitude-Altitude Coordinates

Inertial Frame

The origin of the inertial frame \mathbf{C}_i is at the center of mass of the Earth, and it is oriented as follows. The ${}^i\mathbf{i}_i$ axis is pointing towards the equinox meridian and lies in the equatorial plane, the ${}^i\mathbf{k}_i$ axis is collinear with the spin axis of the Earth and is positive towards North. The ${}^i\mathbf{j}_i$ axis is orthogonal to them and obeys the right-hand Cartesian rule, taking ${}^i\mathbf{k}_i$ axis to ${}^i\mathbf{i}_i$, see figure 2.1.

Earth-Centered-Earth-Fixed Frame

The next frame is called **E**arth **C**entered **E**arth **F**ixed frame and is denoted by \mathbf{C}_E . This frame is attached at the center of mass of the Earth and rotates with it. The ${}^i\mathbf{i}_E$ axis is oriented along the Greenwich meridian, while the ${}^i\mathbf{k}_E$ axis is collinear with the spin axis of the earth and is

positive towards North. The \mathbf{j}_E axis is orthogonal to them and obeys the right-hand Cartesian rule, taking \mathbf{k}_E axis to \mathbf{i}_E .

North-East-Down Frame

With the goal to set some local reference, the frame \mathcal{C}_N known as North-East-Down frame is defined. The frame \mathcal{C}_N is attached at the surface the Earth at any arbitrary point and moves with it but does not rotate with respect to the Earth. The axes \mathbf{i}_N and \mathbf{j}_N define the local horizontal or tangential plane to the local surface. Axis \mathbf{i}_N is oriented towards the North, axis \mathbf{j}_N points towards East, and axis \mathbf{k}_N is orthogonal to them and obeys the right-hand rule. Thus it is normal to the reference ellipsoid and points downwards to the Earth, as shown in figure 2.1

Body Frame

Sometimes it is convenient to express the Newton-Euler equations in a frame attached to the body's center of gravity (*cg*). The body frame \mathcal{C}_b has its origin at the aircraft's center of mass. It is fixed with the aircraft body and moves and rotates with it. The \mathbf{i}_b axis points to the aircraft's nose. Looking the aircraft from top and having the nose ahead, axis \mathbf{j}_b points to the right and is orthogonal to \mathbf{i}_b axis and obeys the right-hand cartesian rule. Finally axis \mathbf{k}_b is orthogonal to the other two axis and follows the right-hand cartesian rule, as it is shown in figure 2.2.

Vehicle-Carried Frame

A second frame attached to the aircraft's *cg* is the vehicle carried frame \mathcal{C}_ν . The origin of this frame is at the *cg* of the helicopter and its orientation is as follows. \mathbf{i}_ν axis is pointing towards the Earth's North. Axis \mathbf{j}_ν points towards East and \mathbf{k}_ν axis points downwards to the Earth and all the time is normal to the reference ellipsoid as shown in figure 2.2.

Blade Frame

A third frame attached to the helicopter is the blade frame denoted by \mathcal{C}_{bl} . This frame has its origin in the center of the hub of the main rotor and is in the plane of the blades or disk plane. \mathbf{i}_{bl} axis is aligned with one of the blades, \mathbf{j}_{bl} axis is orthogonal with respect to axis \mathbf{i}_{bl} and obeys the right-hand rule, and \mathbf{k}_{bl} is collinear with the main rotor hub axis and is positive downwards as shown in 2.2.

Tail Frame

The last frame is the tail frame denoted by \mathbf{C}_t . Its origin is in the joint of the tail hub and the tail blades. The \mathbf{i}_t and \mathbf{j}_t axes lie in the disc plane and rotate with it. The orientation of this frame is as follows. \mathbf{i}_t axis is aligned with one blade, \mathbf{j}_t is orthogonal to \mathbf{i}_t axis and is normal to the leading edge of the tail blade in which the \mathbf{i}_t is aligned. Finally, \mathbf{k}_t is orthogonal to the other two and obeys the hand-right rule, taking \mathbf{i}_t to \mathbf{j}_t . In this way, seen the helicopter from behind, the \mathbf{k}_t axis is positive to the right hand side.

Geodetic Coordinates

An important set of coordinates used in navigation and also used by the Global Position System or GPS, is the Geodetic coordinates, denoted by \mathbf{C}_{lla} . These three coordinates, are Longitude λ , Latitude ζ , and Height h . The first two coordinates are used to locate a point on the Earth's surface, meanwhile the last one is to refer its altitude or elevation above the reference ellipsoid. Longitude λ measures the angle in the equatorial plane. Seeing the Earth from the North Pole, it is positive in the counterclockwise direction (East), starting at the Greenwich meridian and finalizing at 180° in its antipodal point. Is negative in the clockwise direction (West), starting at the same Greenwich meridian and finalizing at -180° in its antipode. Latitude ζ measures the angle from the equator to the poles. It is negative in the direction from the Equator to the South Pole with a range of 0° to -90° , and is positive from the Equator to the North Pole, with a range from 0° to 90° . The third coordinate is the altitude above the reference ellipsoid, it is based on the parameters defined by the Department of Defense World Geodetic System 84, or WGS84 for short, proposed in 1984 by the National Imagery and Mapping Agency, see [25].

Figure 2.1 shows all the frames definitions, whereas figure 2.2 shows the helicopter's frames.

With all these frames, will be necessary to represent vectors expressed in one frame to another. To achieve this task, and because the goal of this research is not to do acrobatics maneuvers, a transformation called Direction Cosine Matrix or DCM is used [26]. The columns of the DCM exclusively

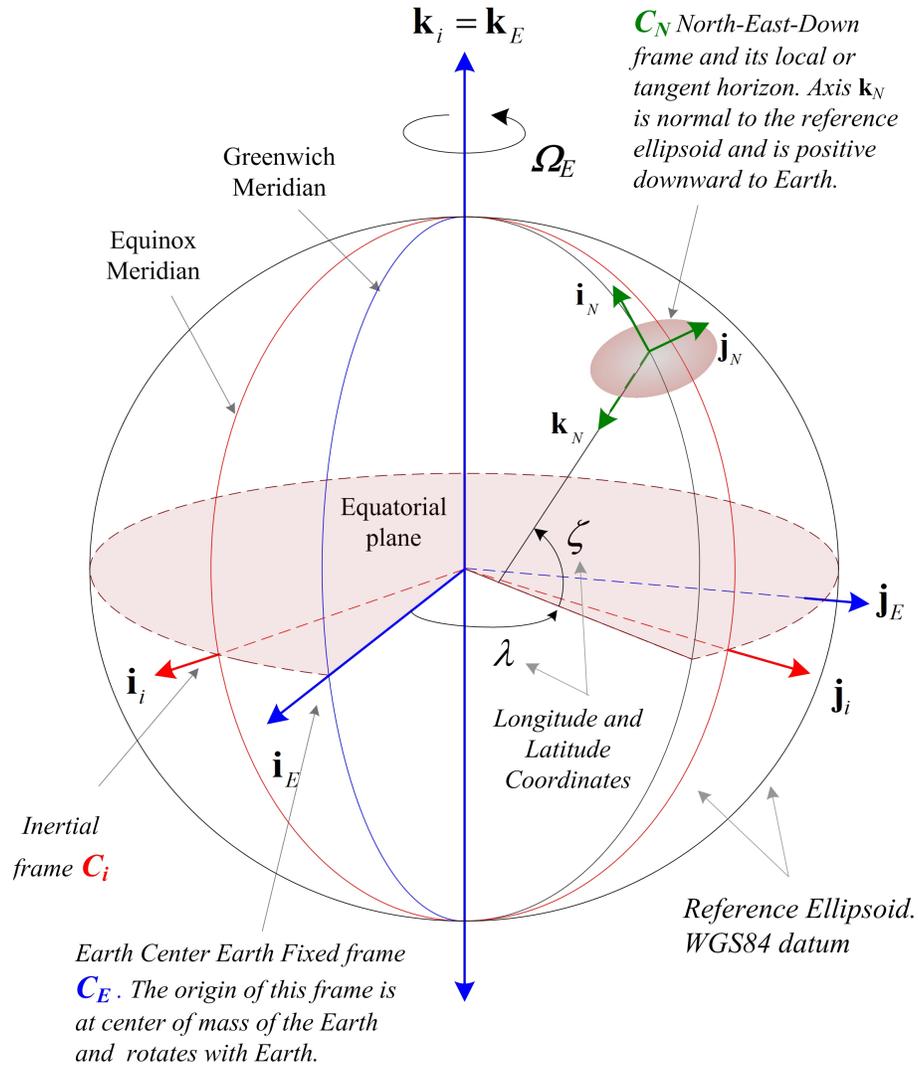


Figure 2.1: Inertial C_i , Earth-Centered-Earth-Fixed C_E , North-East-Down "NED" C_N , and Longitude, Latitude, and Altitude above the reference ellipsoid coordinates C_{lla} .

represent coordinates transformation of the vector basis expressed in one frame to another. The most common notation [26] used for DCM is C_ν^b which states a change of basis of any vector expressed in frame ν to frame b , according to the following expression:

$${}^b \mathbf{x} = C_\nu^b \mathbf{x} \quad (2.1)$$

To get any given vector in some desired frame, matrix C_ν^b in equation (2.1), might be the result of one or more successive rotations or transformations. Each DCM can be computed by doing the inner products between the unit

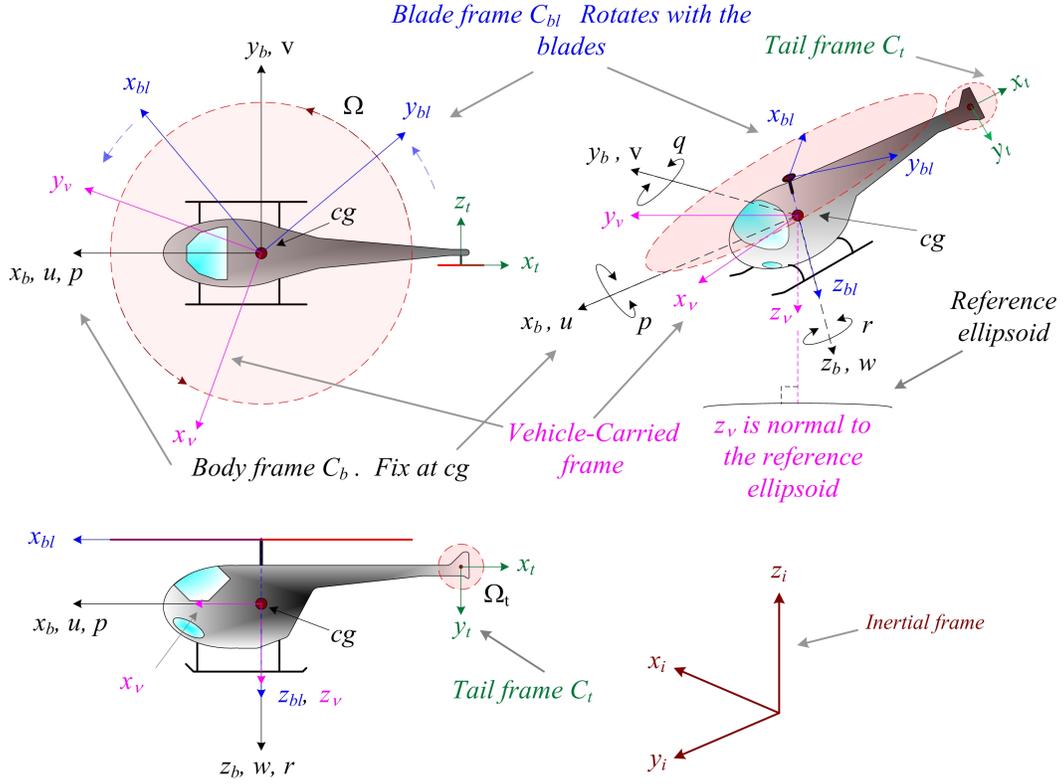


Figure 2.2: Aircraft's frames

vectors of each frame, as is described by the following equation

$$\mathbf{C}_\nu^b = \begin{bmatrix} \mathbf{i}_b \cdot \mathbf{i}_\nu & \mathbf{i}_b \cdot \mathbf{j}_\nu & \mathbf{i}_b \cdot \mathbf{k}_\nu \\ \mathbf{j}_b \cdot \mathbf{i}_\nu & \mathbf{j}_b \cdot \mathbf{j}_\nu & \mathbf{j}_b \cdot \mathbf{k}_\nu \\ \mathbf{k}_b \cdot \mathbf{i}_\nu & \mathbf{k}_b \cdot \mathbf{j}_\nu & \mathbf{k}_b \cdot \mathbf{k}_\nu \end{bmatrix} \quad (2.2)$$

recalling that the inner product $\mathbf{i}_b \cdot \mathbf{i}_\nu$ is given by

$$\mathbf{i}_b \cdot \mathbf{i}_\nu = \mathbf{i}_\nu \cdot \mathbf{i}_b = \|\mathbf{i}_b\| \|\mathbf{i}_\nu\| \cos \varsigma = \cos \varsigma \quad (2.3)$$

where ς is the angle between \mathbf{i}_b and \mathbf{i}_ν , and $\|\cdot\|$ is the L_2 norm computed as:

$$\|\mathbf{i}_b\| = \sqrt{\mathbf{i}_b \cdot \mathbf{i}_b} = 1 \quad (2.4)$$

For more detail in the DCM derivation see section A.1 in appendix A.

The transformations used in this work are described next.

Body to Vehicle-Carried \mathbf{C}_b^ν

The orientation of the body frame respect to a reference frame is known as the aircraft attitude. Greek letters ψ , θ , and ϕ will be used to denote the aircraft's yaw, pitch, and roll angles respectively. These three angles are also known as Euler angles.

To compute these angles between the body frame and the reference frame \mathbf{C}_ν , three successive rotations need to be done on the \mathbf{C}_ν frame. The first rotation, shown in figure 2.3(a), is about the z_ν axis by an angle of ψ degrees, and is given by

$$\mathbf{C}_{z_\nu} = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The second rotation of θ degrees, shown in figure 2.3(b), must be done on the new frame $\mathbf{C}_{\nu'}$ about the $y_{\nu'}$, that is

$$\mathbf{C}_{y_{\nu'}} = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}$$

The last rotation of ϕ degrees on the second new frame $\mathbf{C}_{\nu''}$, see figure 2.3(c), is about axis $x_{\nu''}$, yielding

$$\mathbf{C}_{x_{\nu''}} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}$$

Therefore, the Body to Vehicle-Carried matrix transformation is given by the multiplication of these three rotations as follows [27] $\mathbf{C}_\nu^b = \mathbf{C}_{x_{\nu''}} \mathbf{C}_{y_{\nu'}} \mathbf{C}_{z_\nu}$ yielding

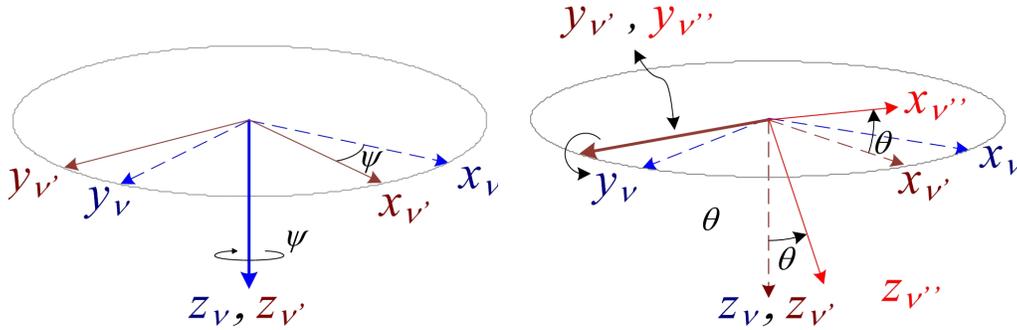
$$\mathbf{C}_\nu^b = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ -c\phi s\psi + s\phi s\theta c\psi & c\phi c\psi + s\phi s\theta s\psi & s\phi c\theta \\ s\phi s\psi + c\phi s\theta c\psi & -s\phi c\psi + c\phi s\theta s\psi & c\phi c\theta \end{bmatrix} \quad (2.5)$$

Due to the fact that matrix \mathbf{C}_ν^b is orthonormal, transformation from Body to Vehicle-Carried frame is just the transpose of the matrix given

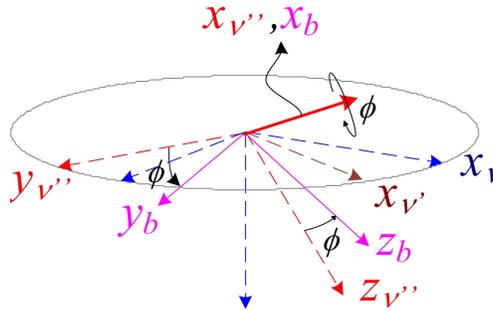
by equation (2.5), and is

$$\mathbf{C}_b^\nu = (\mathbf{C}_\nu^b)^T = \begin{bmatrix} c\theta c\psi & -c\phi s\psi + s\phi s\theta c\psi & s\phi s\psi + c\phi s\theta c\psi \\ c\theta s\psi & c\phi c\psi + s\phi s\theta s\psi & -s\phi c\psi + c\phi s\theta s\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (2.6)$$

Where s and c denote $\sin(\cdot)$ and $\cos(\cdot)$ functions respectively.



(a) Rotation of ψ degrees about z_ν axis. (b) Rotation of θ degrees about $y_{\nu'}$ axis.



(c) Rotation of ϕ degrees about $x_{\nu''}$ axis.

Figure 2.3: Euler angles. Axis Rotations on frame \mathbf{C}_ν to compute the Euler angles between reference frame (Vehicle-Carried frame \mathbf{C}_ν) and Body frame \mathbf{C}_b .

Body to Local North-East-Down \mathbf{C}_b^N

In general the orientation between the Vehicle-Carried frame and the local North-East-Down frame is different and changes as the helicopter moves away from the local NED. However, for the distance at which the helicopter will fly, no more than 150 m away from the local NED, it can be considered that both frames, \mathbf{C}_ν and \mathbf{C}_N have the same orientation.

Under this assumption, the transformation matrix from Body to NED

frame can be written as:

$$\mathbf{C}_b^N \approx \mathbf{C}_b^\nu \quad (2.7)$$

Similarly the inverse transformation, NED to Body frame can be written as

$$\mathbf{C}_N^b \approx \mathbf{C}_\nu^b \quad (2.8)$$

ECEF to NED \mathbf{C}_E^N

To represent a vector expressed in \mathbf{C}_E frame to \mathbf{C}_N frame, two successive rotations are required. The first rotation of λ degrees is about z_E axis on frame \mathbf{C}_E . This define a new frame $\mathbf{C}_{E'}$ in which a left-hand rotation of $\frac{\pi}{2} + \zeta$ degrees is done about $y_{E'}$ axis. Expressing this mathematically yields

$$\mathbf{C}_E^N = \mathbf{C}_{y_{E'}} \mathbf{C}_{z_E} = \begin{bmatrix} -\sin \zeta & 0 & \cos \zeta \\ 0 & 1 & 0 \\ -\cos \zeta & 0 & -\sin \zeta \end{bmatrix} \begin{bmatrix} \cos \lambda & \sin \lambda & 0 \\ -\sin \lambda & \cos \lambda & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and doing the corresponding algebra the \mathbf{C}_E^N transformation matrix becomes

$$\mathbf{C}_E^N = \begin{bmatrix} -\sin \zeta \cos \lambda & -\sin \zeta \sin \lambda & \cos \zeta \\ -\sin \lambda & \cos \lambda & 0 \\ -\cos \zeta \cos \lambda & -\cos \zeta \sin \lambda & -\sin \zeta \end{bmatrix} \quad (2.9)$$

The inverse transformation can be computed by taking the transpose of the orthonormal matrix \mathbf{C}_E^N hence

$$\mathbf{C}_N^E = (\mathbf{C}_E^N)^T = \begin{bmatrix} -\sin \zeta \cos \lambda & -\sin \zeta \sin \lambda & \cos \zeta \\ -\sin \lambda & \cos \lambda & 0 \\ -\cos \zeta \cos \lambda & -\cos \zeta \sin \lambda & -\sin \zeta \end{bmatrix} \quad (2.10)$$

Body to Blade \mathbf{C}_b^{bl}

Since the axis z_{bl} is collinear with z_b axis, the transformation from body to blade frame requires only one rotation of Ψ degrees about the z_b axis, and is given by

$$\mathbf{C}_b^{bl} = \begin{bmatrix} \cos \Psi & \sin \Psi & 0 \\ -\sin \Psi & \cos \Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.11)$$

and the inverse transformation given by its transpose, as

$$\mathbf{C}_{bl}^b = (\mathbf{C}_b^{bl})^T \quad (2.12)$$

Body to Tail \mathbf{C}_b^t

This transformation is similar to the body to blade rotation. However, it requires first a rotation of $-\frac{\pi}{2}$ degrees about the x_b axis, and then a second rotation of Ψ degrees about z_b axis, that is:

$$\mathbf{C}_b^t = \mathbf{C}_{z_b} \mathbf{C}_{x_b}$$

hence, the total rotation is given by the following expression

$$\mathbf{C}_b^t = \begin{bmatrix} \cos \Psi & 0 & -\sin \Psi \\ -\sin \Psi & 0 & -\cos \Psi \\ 0 & 1 & 0 \end{bmatrix} \quad (2.13)$$

And the inverse transformation is just its transpose

$$\mathbf{C}_t^b = (\mathbf{C}_b^t)^T \quad (2.14)$$

Geodetic to Earth-Centered-Earth-Fixed \mathbf{P}_E

All GPS receivers return their position as a set of three Geodetic coordinates. Unfortunately, these coordinate are not suitable to do computation when other frames are involved. For this reason, a transformation from longitude, latitude and altitude frame \mathbf{C}_{lla} to Earth-Centered-Earth-Fixed frame \mathbf{C}_E is required. It is based on the WGS84 datum and is given by the following expression

$$\begin{bmatrix} x_E \\ y_E \\ z_E \end{bmatrix} = \begin{bmatrix} (N(\zeta) + h) \cos \zeta \cos \lambda \\ (N(\zeta) + h) \cos \zeta \sin \lambda \\ [(1 - e^2) N(\zeta) + h] \sin \lambda \end{bmatrix} \quad (2.15)$$

and $N(\zeta)$ is given by

$$N(\zeta) = \frac{a}{\sqrt{1 - e^2(\sin \zeta)^2}}$$

where $N(\zeta)$ is the prime vertical radius of curvature, e is the first nu-

merical eccentricity and a is the semi-major axis. The values for e and a are given by the WGS84 datum, see [25] and are reproduced in section A.2, tables A.1 and A.2 in appendix A for a quick reference.

The inverse transformation from ECEF to latitude-longitude-altitude (lla) requires an iterative process and can be done using the Bowring's method, [28]. In this work it is not necessary to calculate this inverse transformation; however, the details can be found in [29], [28], and [30].

Euler Rates to Body Angular rates $\mathbf{C}_{\dot{\phi}}^{\omega}$

Since the transformation between Body and Vehicle-Carried frame was done by computing three successive rotations in three different frames, \mathbf{C}_{ν} , $\mathbf{C}_{\nu'}$, and $\mathbf{C}_{\nu''}$, the relation between Euler rates and body angular velocities, need to follow exactly the same sequence, that is

$${}^b\boldsymbol{\omega} = (\mathbf{C}_{x_{\nu''}})^T \dot{\phi} \mathbf{i}_{\nu''} + (\mathbf{C}_{x_{\nu''}})^T (\mathbf{C}_{y_{\nu'}})^T \dot{\theta} \mathbf{j}_{\nu'} + (\mathbf{C}_{x_{\nu''}})^T (\mathbf{C}_{y_{\nu'}})^T (\mathbf{C}_{z_{\nu}})^T \dot{\psi} \mathbf{k}_{\nu}$$

which can be reduced to

$$\boldsymbol{\omega} = \mathbf{C}_{\dot{\phi}}^{\omega} \dot{\phi}$$

where matrix $\mathbf{C}_{\dot{\phi}}^{\omega}$ is given by

$$\mathbf{C}_{\dot{\phi}}^{\omega} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \quad (2.16)$$

Equation (2.16) represents the transformation matrix from Euler rates to body angular velocity. The inverse transformation can be computed by taking the inverse of equation (2.16), and is

$$\mathbf{C}_{\omega}^{\dot{\phi}} = (\mathbf{C}_{\dot{\phi}}^{\omega})^{-1} = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \quad (2.17)$$

Equations (2.16) and (2.17) represent the relation between Euler rates and body angular velocities. For more detail see [31]

2.2 Rigid-body General motion equations

In this section, a brief presentation of the general equations of motion are given as an introduction to the helicopter's dynamic model.

For non relativistic mechanics, the fundamental equations that describe the motion of a rigid body are the Newton-Euler equations [32]. These equations state that the summation of all external forces applied to a rigid body must be equal to the rate of change of the linear momentum. In addition, the summation of all external moments acting on a rigid body must be equal to the rate of change of the angular momentum. These can be expressed as follows

$$\sum \mathbf{F} = \frac{\partial(m\mathbf{v}_{cg})}{\partial t} \quad (2.18)$$

$$\sum \mathbf{M}_{cg} = \frac{\partial(\mathbf{h}_{cg})}{\partial t} \quad (2.19)$$

where \mathbf{F} is the vector of forces applied to the body, m is the total body mass, vector \mathbf{v}_{cg} is the resultant velocity of the body's cg , vector \mathbf{M}_{cg} represents the moments acting on the body, and vector \mathbf{h}_{cg} is the resultant angular momentum.

The small scale helicopter used in this work is an Air Star Evolution and is electric powered. Thus, the mass of the helicopter remains constant during the flight. For a small unmanned aerial vehicle (UAV) using fuel, in which the fuel mass m_f does not represent considerable percentage of the total mass the assumption of constant mass is often used. However, for a high endurance UAV where the change of mass is significant this must be taken into account by the controller gains for a proper operation.

To simplify the derivation of the helicopter's model, it is convenient to set the working frame attached to the body's cg and allow it to rotate with the body. By picking the body frame \mathbf{C}_b as a working frame, the total helicopter's mass inertia tensor \mathbf{I}_T associated to the angular momentum \mathbf{h}_{cg} will remain constant for all the time.

Therefore, equations (2.18) and (2.19) for the dynamics of helicopter are

$$\sum \mathbf{F} = m\dot{\mathbf{v}}_{cg} + \boldsymbol{\omega} \times m\mathbf{v}_{cg} \quad (2.20)$$

$$\sum \mathbf{M}_{cg} = \dot{\mathbf{h}}_{cg} + \boldsymbol{\omega} \times \mathbf{h}_{cg} \quad (2.21)$$

Equations (2.20) and (2.21) represent the motion equations of the helicopter expressed in the body frame \mathbf{C}_b .

Solving for the time derivatives on the left hand side of equation (2.20), yield the first three set of state equations that describe the helicopter's dynamics,

$$\dot{\mathbf{v}}_{cg} = \frac{1}{m} \left[\sum \mathbf{F} - \boldsymbol{\omega} \times \mathbf{v}_{cg} \right] \quad (2.22)$$

The other set of three state equations are related to the angular momentum and are implicit in equation (2.21) and their final form depend on the nature of the angular momentum \mathbf{h}_{cg} .

The Euler angles rate of change $\dot{\boldsymbol{\varphi}} = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ and the velocity vector $\dot{\boldsymbol{\chi}} = [\dot{x}_i, \dot{y}_i, \dot{z}_i]^T$ with respect to the inertial frame, complete the twelve state differential equations according to:

$$\dot{\boldsymbol{\varphi}} = \mathbf{C}_{\boldsymbol{\omega}}^{\dot{\boldsymbol{\varphi}}} \boldsymbol{\omega} \quad (2.23)$$

$$\dot{\boldsymbol{\chi}} = \mathbf{C}_b^{i b} \mathbf{v}_{cg} \quad (2.24)$$

Equations (2.21) to (2.24) completely describe the Helicopter's non-linear dynamics.

Helicopters are complex machines that involve many forces and moments acting on them. However, for the Air Star Evolution helicopter the most significant ones considered by [22], are shown in figure 2.4. The resultant total force \mathbf{F}_T and moment \mathbf{M}_{cg} , shown in figure 2.4, are expressed as the summation of the followings terms:

$$\mathbf{F}_T = \mathbf{T}_r + \mathbf{T}_t + \mathbf{H}_r + \mathbf{H}_t + \mathbf{W} + \mathbf{F}_D \quad (2.25)$$

and

$$\mathbf{M}_{cg} = \mathbf{M}_{\phi} + \mathbf{M}_{\theta} + \mathbf{M}_t + \mathbf{M}_{H_r} + \mathbf{M}_{H_t} + \mathbf{Q}_r + \mathbf{Q}_t + \mathbf{M}_f \quad (2.26)$$

where:

- \mathbf{T}_r is the main rotor thrust vector
- \mathbf{T}_t is the tail rotor thrust vector
- \mathbf{H}_r is the main rotor in-plane force vector
- \mathbf{H}_t is the tail rotor in-plane force vector
- \mathbf{W} is the helicopter weight vector
- \mathbf{F}_D is the aerodynamic drag vector due to the fuselage
- \mathbf{M}_ϕ is the roll moment vector due to the cyclic comand
- \mathbf{M}_θ is the pitch moment vector due to the cyclic command
- \mathbf{M}_t is the moment vector due to the tail rotor thrust
- \mathbf{M}_{H_r} is the moment vector due to the main rotor in-plane force
- \mathbf{M}_{H_t} is the moment vector due to the tail rotor in-plane force
- \mathbf{Q}_r is the drag torque vector due to the main rotor
- \mathbf{Q}_t is the drag torque vector due to the tail rotor
- \mathbf{M}_f is the moment vector due to the fly bar

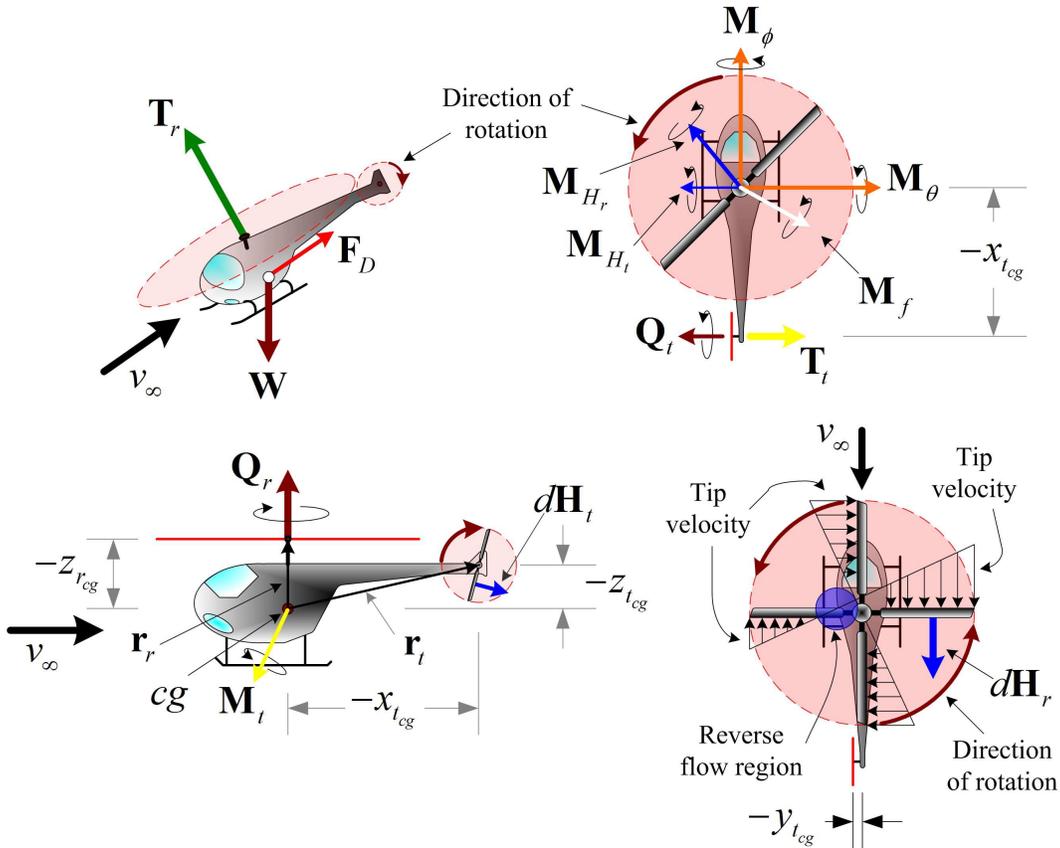


Figure 2.4: Helicopter forces and moments

and their corresponding components are:

$$\begin{aligned}
\mathbf{T}_r &= -T_r \mathbf{k}_b \\
\mathbf{T}_t &= T_t \mathbf{j}_b \\
\mathbf{H}_r &= H_{r_x} \mathbf{i}_b + H_{r_y} \mathbf{j}_b \\
\mathbf{H}_t &= H_{t_x} \mathbf{i}_b + H_{t_z} \mathbf{k}_b \\
\mathbf{F}_D &= F_{D_x} \mathbf{i}_b + F_{D_y} \mathbf{j}_b + F_{D_z} \mathbf{k}_b \\
\mathbf{M}_\phi &= M_\phi \mathbf{i}_b \\
\mathbf{M}_\theta &= M_\theta \mathbf{j}_b \\
\mathbf{M}_t &= {}^b \mathbf{r}_t \times {}^b \mathbf{T}_t = [-x_{t_{cg}}, -y_{t_{cg}}, -z_{t_{cg}}]_b^T \times [0, T_t, 0]_b^T = z_{t_{cg}} T_t \mathbf{i}_b - x_{t_{cg}} T_t \mathbf{k}_b \\
\mathbf{M}_{H_r} &= {}^b \mathbf{r}_r \times {}^b \mathbf{H}_r = [0, 0, -z_{r_{cg}}]_b^T \times [H_{r_x}, H_{r_y}, 0]_b^T = z_{r_{cg}} H_{r_y} \mathbf{i}_b - z_{r_{cg}} H_{r_x} \mathbf{j}_b \\
\mathbf{M}_{H_t} &= {}^b \mathbf{r}_t \times {}^b \mathbf{H}_t = [-x_{t_{cg}}, -y_{t_{cg}}, -z_{t_{cg}}]_b^T \times [H_{t_x}, 0, H_{t_z}]_b^T = -y_{t_{cg}} H_{t_z} \mathbf{i}_b + \\
&(-z_{t_{cg}} H_{t_x} + x_{t_{cg}} H_{t_z}) \mathbf{j}_b + y_{t_{cg}} H_{t_x} \mathbf{k}_b \\
\mathbf{Q}_r &= Q_r \mathbf{k}_b \\
\mathbf{Q}_t &= -Q_t \mathbf{j}_b \\
\mathbf{M}_f &= \left(-\alpha_1 \delta_{\theta_{lon}} + \alpha_2 \dot{\theta} \right) \mathbf{i}_b + \left(\alpha_3 \delta_{\theta_{lat}} + \alpha_4 \dot{\phi} \right) \mathbf{j}_b, \text{ see [8]}
\end{aligned}$$

where ${}^b \mathbf{r}_r$ and ${}^b \mathbf{r}_t$ are the position of the main rotor and the tail rotor with respect to the helicopter's *cg*. δ_{lon} and δ_{lat} are the corresponding longitudinal and lateral cyclic commands.

The total forces and moment acting on the helicopter are written in matrix form as follows:

$$\mathbf{F}_T = \begin{bmatrix} H_{r_x} + H_{t_x} \\ T_t + H_{r_y} \\ -T_r + H_{t_z} \end{bmatrix}_b + {}^b \mathbf{W} + {}^b \mathbf{F}_D \quad (2.27)$$

and

$$\mathbf{M}_{cg} = \begin{bmatrix} M_\phi + z_{t_{cg}} T_t + z_{r_{cg}} H_{r_y} - y_{t_{cg}} H_{t_z} \\ M_\theta - z_{r_{cg}} H_{r_x} + (x_{t_{cg}} H_{t_z} - z_{t_{cg}} H_{t_x}) - Q_t \\ Q_r - x_{t_{cg}} T_t + y_{t_{cg}} H_{t_x} \end{bmatrix}_b + {}^b \mathbf{M}_f \quad (2.28)$$

In the next section, computation of the total angular momentum \mathbf{h}_{cg} , including the contribution of the main rotor will be described, and in section 2.4 the aerodynamic forces will be derived.

2.3 Helicopter Dynamic Model

The total angular momentum of the helicopter about its center of gravity is [22]:

$$\mathbf{h}_{cg} = \mathbf{I}\boldsymbol{\omega} + \mathbf{C}_{bl}^b {}^{bl}\mathbf{h}_r + \mathbf{r}_r \times m_r ({}^b\mathbf{v}_{cg} + \boldsymbol{\omega} \times \mathbf{r}_r) \quad (2.29)$$

where m_r is the rotor mass, \mathbf{I} is the helicopter's mass moment of inertia tensor in the body frame including the tail rotor disk but excluding the main rotor disk, and \mathbf{r}_r is the position of the rotor hub from the helicopter's cg in the body frame.

The angular momentum of the blade rotor is given by ${}^{bl}\mathbf{h}_r = \mathbf{I}_r {}^{bl}\boldsymbol{\omega}_r$ where ${}^{bl}\boldsymbol{\omega}_r = \mathbf{C}_b^{bl} {}^b\boldsymbol{\omega}_r$, and ${}^b\boldsymbol{\omega}_r$ is the instantaneous total angular velocity of the rotor. In body coordinates this is given by $\boldsymbol{\omega}_r = \boldsymbol{\omega} + \boldsymbol{\Omega}$, and $\boldsymbol{\Omega} = [0, 0, -\Omega]^T$. The rotation matrix from body to blades coordinates is:

$$\mathbf{C}_b^{bl} = \begin{bmatrix} \cos \Psi & \sin \Psi & 0 \\ -\sin \Psi & \cos \Psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

and $\mathbf{C}_{bl}^b = (\mathbf{C}_b^{bl})^T$. The rotor inertia tensor in the blade frame \mathbf{C}_{bl} is given by:

$$\mathbf{I}_{bl} = \begin{bmatrix} I_{x_{bl}} & 0 & 0 \\ 0 & I_{y_{bl}} & 0 \\ 0 & 0 & I_{z_{bl}} \end{bmatrix}$$

Now using Newton-Euler method, the total moment acting on the helicopter's cg can be written as:

$$\mathbf{M}_{cg}(\Psi) = \frac{{}^i d ({}^b\mathbf{h}_{cg})}{dt} = \frac{{}^b d ({}^b\mathbf{h}_{cg})}{dt} + {}^b\boldsymbol{\omega} \times {}^b\mathbf{h}_{cg} \quad (2.30)$$

It is important to note that the third term on the right hand side of equation (2.29) is the angular momentum of the helicopter introduced by the mass of the main blades at rest. At low linear velocities and due to the relative small mass of the main rotor blades, about 0.318 kg each (compared of the overall helicopter mass of 10.8 Kg), this term can be neglected without a large error. Substituting (2.29) into (2.30) and neglecting this term results in

$$\mathbf{M}_{cg}(\Psi) = \frac{{}^b d}{dt} (\mathbf{I} {}^b\boldsymbol{\omega} + \mathbf{C}_{bl}^b {}^{bl}\mathbf{h}_r) + {}^b\boldsymbol{\omega} \times (\mathbf{I} {}^b\boldsymbol{\omega} + \mathbf{C}_{bl}^b {}^{bl}\mathbf{h}_r)$$

and expanding terms yields

$$\mathbf{M}_{cg}(\Psi) = \mathbf{I}^b \dot{\boldsymbol{\omega}} + \frac{{}^b d(\mathbf{C}_{bl}^b)}{dt} {}^{bl} \mathbf{h}_r + \mathbf{C}_{bl}^b \frac{{}^{bl} d({}^{bl} \mathbf{h}_r)}{dt} + {}^b \boldsymbol{\omega} \times \mathbf{I}^b \boldsymbol{\omega} + {}^b \boldsymbol{\omega} \times \mathbf{C}_{bl}^b {}^{bl} \mathbf{h}_r$$

The time derivative of the rotation matrix can be computed as

$$\frac{{}^b d(\mathbf{C}_{bl}^b)}{dt} = \begin{bmatrix} -\dot{\Psi} \sin \Psi & -\dot{\Psi} \cos \Psi & 0 \\ \dot{\Psi} \cos \Psi & -\dot{\Psi} \sin \Psi & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

but the rate of change of the azimuth angle $\dot{\Psi}$ is the rotor angular velocity Ω resulting in

$$\frac{{}^b d(\mathbf{C}_{bl}^b)}{dt} {}^{bl} \mathbf{h}_r = \begin{bmatrix} -\Omega \sin \Psi & -\Omega \cos \Psi & 0 \\ \Omega \cos \Psi & -\Omega \sin \Psi & 0 \\ 0 & 0 & 0 \end{bmatrix} {}^{bl} \mathbf{h}_r = -\boldsymbol{\Omega} \times (\mathbf{C}_{bl}^b {}^{bl} \mathbf{h}_r)$$

hence, the moment equation is

$$\mathbf{M}_{cg}(\Psi) = \mathbf{I} \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega} - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b {}^{bl} \mathbf{h}_r + \mathbf{C}_{bl}^b \frac{{}^{bl} d({}^{bl} \mathbf{h}_r)}{dt} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b {}^{bl} \mathbf{h}_r$$

but from [22] using ${}^{bl} \mathbf{h}_r = \mathbf{I}_r \mathbf{C}_b^{bl} {}^b \boldsymbol{\omega}_r$ for the angular momentum of the rotor and for the total angular velocity [22] results in ${}^b \boldsymbol{\omega}_r = \boldsymbol{\omega} + \boldsymbol{\Omega}$ then

$$\begin{aligned} \mathbf{M}_{cg}(\Psi) &= \mathbf{I} \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} (\boldsymbol{\omega} + \boldsymbol{\Omega})] \\ &\quad - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} (\boldsymbol{\omega} + \boldsymbol{\Omega})] + \mathbf{C}_{bl}^b \frac{{}^{bl} d}{dt} [\mathbf{I}_r \mathbf{C}_b^{bl} (\boldsymbol{\omega} + \boldsymbol{\Omega})] \end{aligned}$$

This equation can be expanded as

$$\begin{aligned} \mathbf{M}_{cg}(\Psi) &= \mathbf{I} \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] \\ &\quad - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] + \mathbf{C}_{bl}^b \frac{{}^{bl} d}{dt} [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] + \\ &\quad \mathbf{C}_{bl}^b \frac{{}^{bl} d}{dt} [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] = \\ &\quad \mathbf{I} \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] - \\ &\quad \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] + \\ &\quad \mathbf{C}_{bl}^b \left[\mathbf{I}_r \frac{{}^{bl} d(\mathbf{C}_b^{bl})}{dt} \boldsymbol{\omega} + \mathbf{I}_r \mathbf{C}_b^{bl} \frac{{}^b d(\boldsymbol{\omega})}{dt} \right] + \mathbf{C}_{bl}^b \left[\mathbf{I}_r \frac{{}^{bl} d(\mathbf{C}_b^{bl})}{dt} \boldsymbol{\Omega} + \mathbf{I}_r \mathbf{C}_b^{bl} \frac{{}^b d(\boldsymbol{\Omega})}{dt} \right] \end{aligned}$$

To simplify this equation the fact that the time derivative of \mathbf{C}_b^{bl} is just the transpose of the time derivative of \mathbf{C}_{bl}^b and in terms of the cross product it can be expressed as

$$\frac{d(\mathbf{C}_b^{bl})}{dt} \boldsymbol{\omega} = \boldsymbol{\Omega} \times (\mathbf{C}_b^{bl} \boldsymbol{\omega})$$

and

$$\frac{d(\mathbf{C}_{bl}^b)}{dt} \boldsymbol{\Omega} = \boldsymbol{\Omega} \times (\mathbf{C}_{bl}^b \boldsymbol{\Omega})$$

and, the term

$$\boldsymbol{\Omega} \times (\mathbf{C}_b^{bl} \boldsymbol{\Omega}) = 0$$

Substituting into the above equation the moment equation results in

$$\begin{aligned} \mathbf{M}_{cg}(\Psi) = & \mathbf{I} \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] \\ & - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega}] - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b [\mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega}] + \mathbf{C}_{bl}^b [\mathbf{I}_r (\boldsymbol{\Omega} \times \mathbf{C}_b^{bl} \boldsymbol{\omega}) + \mathbf{I}_r \mathbf{C}_b^{bl} \dot{\boldsymbol{\omega}}] + \\ & \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} \dot{\boldsymbol{\Omega}} \end{aligned} \quad (2.31)$$

Rearranging equation (2.31) it becomes

$$\begin{aligned} \mathbf{M}_{cg}(\Psi) = & \mathbf{I} \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I} \boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega} \\ & - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega} - \boldsymbol{\Omega} \times \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\Omega} - \\ & \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} \boldsymbol{\omega} \times \boldsymbol{\Omega} + \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} (\dot{\boldsymbol{\omega}} + \dot{\boldsymbol{\Omega}}) \end{aligned} \quad (2.32)$$

Equation (2.32) represents the moment of the helicopter and the contribution of the main rotor for a given instantaneous azimuth angle Ψ . To simplify the model it is necessary to do an average about one complete revolution [33], [34]. The terms involving the azimuth angle Ψ in equation (2.32) are those related to the basis transformation matrices and the rotor inertia tensor. Therefore, defining the inertia tensor \mathbf{I}_s as [22] $\mathbf{I}_s = \frac{1}{2\pi} \int_0^{2\pi} \mathbf{C}_{bl}^b \mathbf{I}_r \mathbf{C}_b^{bl} d\Psi$ results in

$$\mathbf{I}_s = \frac{1}{2\pi} \int_0^{2\pi} \begin{bmatrix} I_{x_{bl}} \cos^2 \Psi + I_{y_{bl}} (1 - \cos^2 \Psi) & (I_{x_{bl}} - I_{y_{bl}}) \cos \Psi \sin \Psi & 0 \\ (I_{x_{bl}} - I_{y_{bl}}) \cos \Psi \sin \Psi & I_{y_{bl}} \cos^2 \Psi + I_{x_{bl}} (1 - \cos^2 \Psi) & 0 \\ 0 & 0 & I_{z_{bl}} \end{bmatrix} d\Psi$$

Solving the integral yields

$$\mathbf{I}_s = \begin{bmatrix} \frac{I_{x_{bl}} + I_{y_{bl}}}{2} & 0 & 0 \\ 0 & \frac{I_{y_{bl}} + I_{x_{bl}}}{2} & 0 \\ 0 & 0 & I_{z_{bl}} \end{bmatrix}$$

Then substituting this expression into equation (2.32) yields

$$\begin{aligned} \overline{\mathbf{M}}_{cg}(\Psi) = & \mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{I}_s\boldsymbol{\omega} + \boldsymbol{\omega} \times \mathbf{I}_s\boldsymbol{\Omega} - \boldsymbol{\Omega} \times \mathbf{I}_s\boldsymbol{\omega} - \boldsymbol{\Omega} \times \mathbf{I}_s\boldsymbol{\Omega} - \\ & \mathbf{I}_s\boldsymbol{\omega} \times \boldsymbol{\Omega} + \mathbf{I}_s \left(\dot{\boldsymbol{\omega}} + \dot{\boldsymbol{\Omega}} \right) \end{aligned}$$

and using $\boldsymbol{\Omega} \times \mathbf{I}_s\boldsymbol{\Omega} = 0$, the final expression for the moment is given by

$$\overline{\mathbf{M}}_{cg}(\Psi) = (\mathbf{I} + \mathbf{I}_s)\dot{\boldsymbol{\omega}} + \mathbf{I}_s\dot{\boldsymbol{\Omega}} + \boldsymbol{\omega} \times [(\mathbf{I} + \mathbf{I}_s)\boldsymbol{\omega} + \mathbf{I}_s\boldsymbol{\Omega}] \quad (2.33)$$

Equation (2.33) represents the total average moment over one revolution of main rotor of the Air Star Evolution helicopter including the contribution of the main rotor moment, [22]. Furthermore, it is important to notice that for the majority of the flight regimens, of these type of helicopters, the rotor speed ${}^b\boldsymbol{\Omega}$ is constant or quasi-constant, therefore; the rotor acceleration ${}^b\dot{\boldsymbol{\Omega}} = 0$ or ${}^b\dot{\boldsymbol{\Omega}} \approx 0$ and the term $\mathbf{I}_s {}^b\dot{\boldsymbol{\Omega}}$, on equation (2.33), is zero or can be neglected. In fact, a main rotor speed control is implemented to hold the motor speed constant for the test helicopter - more details later in this thesis.

2.4 Helicopter Aerodynamic model

The aerodynamic moments and forces are derived from the blade element theory and from the momentum theory [34], [33]. Figure 2.5, shows the position of an element p on the main rotor blade which is referred to a frame \mathbf{C}_{bl} attached to the center of the hub and its axes are defined as $[x_{bl}, y_{bl}, z_{bl}]$. The blade frame \mathbf{C}_{bl} rotates with the blades at an angular speed Ω . The z_{bl} axis is aligned with axis z_b from the body frame and is defined positive downward. Axis x_{bl} is aligned with one of the blade, see figure 2.5, and axis y_{bl} is orthogonal to the other two axes and obeys the right-hand rule. The total velocity of p ${}^{bl}\mathbf{v}_p$ expressed in blade coordinates is given by [22], [34]:

$${}^{bl}\mathbf{v}_p = \mathbf{C}_b^{bl} \left({}^b\mathbf{v}_{cg} - {}^b\mathbf{v}_w + {}^b\boldsymbol{\omega} \times {}^b\mathbf{r}_r + {}^b\boldsymbol{\omega}_r \times \mathbf{C}_{bl}^b \tau R \mathbf{i}_{bl} \right) + v_i \mathbf{k}_{bl} = \begin{bmatrix} U_r \\ U_t \\ U_n \end{bmatrix} \quad (2.34)$$

as shown in figure 2.5, and figure 2.6, where $\tau = \frac{y}{R}$ and y is the distance of the point p to the origin of frame \mathbf{C}_{bl} . The vector $[U_r, U_t, U_n]^T$ represents the velocity at point p of the blade, in which U_r , U_t , and U_n are the radial,

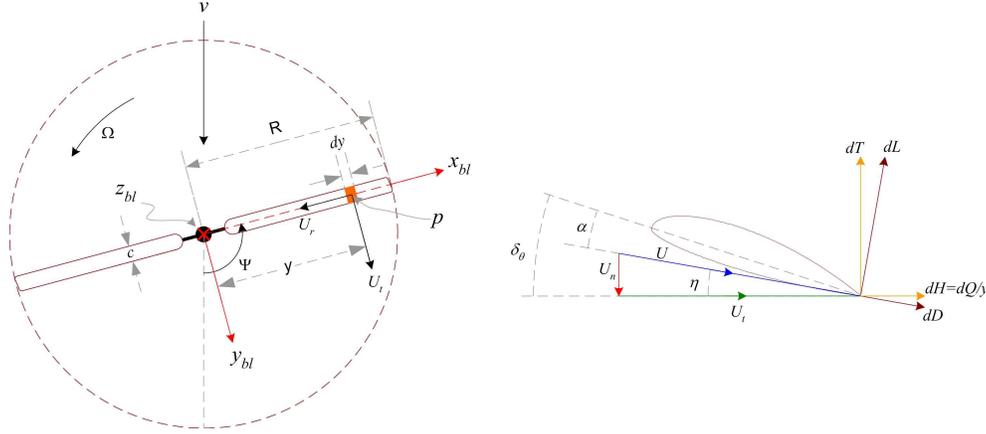


Figure 2.5: Element of the blade on point p and longitudinal cut on point p showing the blade airfoil and the incoming U stream and its components.

tangential, and normal components of the air relative velocity respectively, expressed in the blade frame \mathbf{C}_{bl} . \mathbf{r}_r is the position from the cg to the hub of the helicopter and is given by $\mathbf{r}_r = [0, 0, -z_{r_{cg}}]$. Based on the momentum and blade element theory [33] and [34], see figure 2.5, the lift and drag in forward flight on a blade element is

$$dL = \frac{1}{2} \rho U^2 c dy C_L \quad (2.35)$$

and

$$dD = \frac{1}{2} \rho U^2 c dy C_D \quad (2.36)$$

where ρ is the air density, C_L and C_D are the lift and drag coefficients respectively, and U is the resultant oncoming air stream. Based on figure 2.5, the thrust T and drag Q torque on the blade element is:

$$dT = dL \cos \eta - dD \sin \eta \quad (2.37)$$

and

$$dQ = (dL \sin \eta + dD \cos \eta) y \quad (2.38)$$

Here, it is assumed that the blade angle η is small. Therefore, the small angle approximations holds and the thrust and drag torque can be reduced to [33]

$$dT \approx dL \quad (2.39)$$

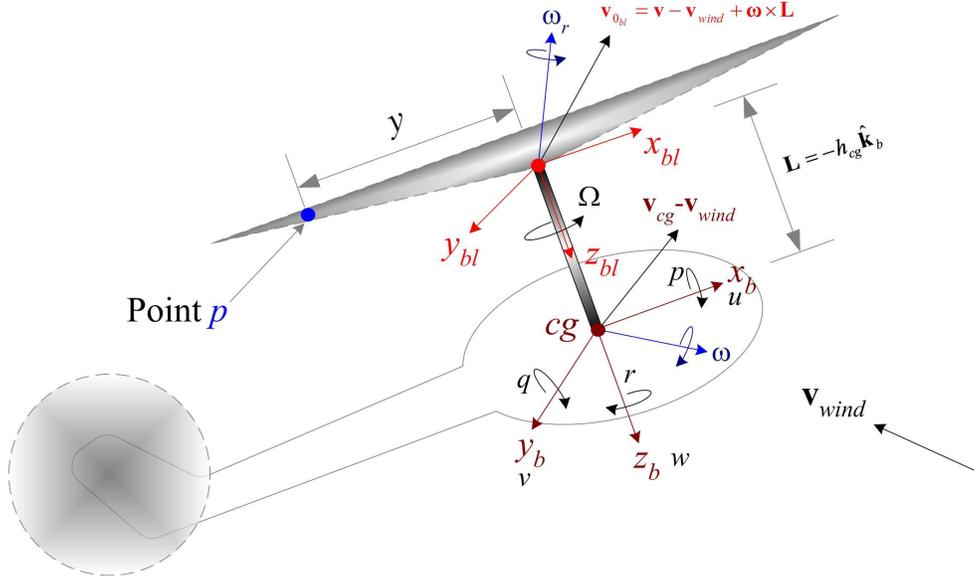


Figure 2.6: Velocity of the element p and angular velocities ω , Ω and ω_r

and

$$dQ = (dL\eta + dD)y \quad (2.40)$$

Equations (2.39) and (2.40) are only valid for small angles η , since this assumption is used to calculate the thrust and aerodynamic drag torque generated by the main and tail rotor blades and other forces acting on the helicopter.

Next, the main rotor thrust, drag torque, pitch and roll moments due to the cyclic pitch angle, in-plane forces, tail rotor thrust, tail rotor drag torque, and the tail rotor in-plane forces will be derived based on [22], [33] and [34].

2.4.1 Main Rotor Thrust

To calculate the thrust generated by the main blades, the dimensionless thrust coefficient is first defined as

$$dC_{T_r} = \frac{dT_r}{A\rho(\Omega R)^2} \approx \frac{dL}{\pi R^2\rho(\Omega R)^2} = \frac{\frac{1}{2}\rho U^2 c dy C_L}{\pi R^2\rho(\Omega R)^2} \quad (2.41)$$

Where $A = \pi R^2$ is the area of the rotor disc. The lift coefficient is defined as $C_L = C_{L_\alpha} \alpha$, where C_{L_α} is the lift slope and α is the angle of attack. From figure 2.5 it is apparent that $\alpha = (\delta_\theta - \eta)$, where δ_θ is the **collective** and **cyclic** pitch angle commands for the main rotor blades. Therefore, substituting in

results in.

$$dC_{T_r} = \frac{U^2 c}{2\pi R (\Omega R)^2} C_{L_\alpha} (\delta_\theta - \eta) \frac{dy}{R}$$

Assuming for a great range of operation of the helicopter that the angle η between the free stream velocity U and the horizontal line will not change more than $\pm 15^\circ \approx \pm 0.26$ rad. This is small enough to approximate U as U_t and $\tan \eta = \frac{U_n}{U_t} \approx \eta$, resulting in

$$dC_{T_r} = \frac{C_{L_\alpha} c}{2 \pi R (\Omega R)^2} \left(\delta_\theta - \frac{U_n}{U_t} \right) \frac{dy}{R} = \frac{C_{L_\alpha} c}{2 \pi R} \left(\delta_\theta \frac{U_t^2}{(\Omega R)^2} - \frac{U_t U_n}{\Omega R \Omega R} \right) \frac{dy}{R}$$

Defining the dimensionless velocities $u_t = \frac{U_t}{\Omega R}$ and $u_n = \frac{U_n}{\Omega R}$, and introducing the variable change $\tau = \frac{y}{R}$ and $d\tau = \frac{dy}{R}$, and substituting into above results in

$$dC_{T_r} = \frac{C_{L_\alpha} c}{2 \pi R} (\delta_\theta u_t^2 - u_t u_n) d\tau$$

Which is only valid just for one blade. When adding the second blade the total thrust coefficient becomes.

$$dC_{T_r} = \frac{C_{L_\alpha} 2c}{2 \pi R} u_t^2 \left(\delta_\theta - \frac{u_n}{u_t} \right) d\tau$$

Introducing the solidity factor \mathcal{S} defined as $\mathcal{S} = \frac{\text{Total blade area}}{\text{Disc area}} = \frac{2cR}{\pi R^2} = \frac{2c}{\pi R}$, the thrust expression becomes

$$dC_{T_r} = \frac{\mathcal{S} C_{L_\alpha}}{2} u_t^2 \left(\delta_\theta - \frac{u_n}{u_t} \right) d\tau \quad (2.42)$$

The blade type used in the Air Start Evolution helicopter are non-twisted blades. This means that the cord c is constant all along the blade span. The cyclic pitch angle is assumed to change only as a function of the azimuth angle Ψ depending on the cyclic commands. Furthermore, blades are assumed light enough compared to the full scale ones, such that lagging motion also can be neglected. Nevertheless, angle δ_θ and the air stream velocity are functions of the azimuth angle Ψ . So it is necessary to do an average over one complete revolution and along the length of the blades, from the hinge position to the

tip. Performing this average, the thrust coefficient becomes

$$C_{T_r} = \frac{\mathcal{S}}{2} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \left[\frac{1}{2\pi} \int_0^{2\pi} C_{L_\alpha} \left(\delta_\theta - \frac{u_n}{u_t} \right) u_t^2 d\Psi \right] d\tau \quad (2.43)$$

The cyclic pitch angle δ_θ on the main rotor are often decomposed as a Fourier series expansion of the form $\delta_\theta = A_0 + A_1 \cos \Psi + B_1 \sin \Psi + \dots$ [33] and [34]. The constant and next to terms in this Fourier expansion are enough to represent the behavior of the collective and cyclic pitch angles. Coefficient A_0 in the expression represents the collective pitch angle. Coefficient A_1 is known as lateral cyclic pitch angle and coefficient B_1 is the longitudinal cyclic pitch angle [33]. Using this angle δ_θ can be written as $\delta_\theta = \delta_{\theta_{col}} + \delta_{\theta_{lat}} \cos \Psi + \delta_{\theta_{lon}} \sin \Psi$ and substituting it in equation (2.43), the thrust coefficient becomes

$$C_{T_r} = \frac{\mathcal{S}}{4\pi} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \int_0^{2\pi} C_{L_\alpha} \left(\delta_{\theta_{col}} + \delta_{\theta_{lat}} \cos \Psi + \delta_{\theta_{lon}} \sin \Psi - \frac{u_n}{u_t} \right) u_t^2 d\Psi d\tau \quad (2.44)$$

After rearranging equation (2.41) the thrust is obtained by multiplying the thrust coefficient by the term $\pi \rho R^2 (\Omega R)^2$ resulting in

$$T_r = \frac{\mathcal{S} \rho R^2 (\Omega R)^2}{4} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \int_0^{2\pi} C_{L_\alpha} \left(\delta_{\theta_{col}} + \delta_{\theta_{lat}} \cos \Psi + \delta_{\theta_{lon}} \sin \Psi - \frac{u_n}{u_t} \right) u_t^2 d\Psi d\tau \quad (2.45)$$

In equation (2.45), the main rotor thrust can be broken down in two terms. The first term is independent of the cyclic pitch angles whereas the second one depends on it. In this way, the thrust becomes

$$T_r = T_{r_0} + T_{r_\theta} \delta_\theta \quad (2.46)$$

and is denoted as affine-form [22].

To compute the integrals of the thrust expression, it is necessary to find the expression for the tangential and normal velocity given by equation 2.34. Additional components of normal velocity U_n are v_c and v_i where v_c is the climbing velocity and v_i is the induced velocity as shown in figure 2.7. However, for hover and forward flight, we can consider $v_c \approx 0$.

Dimensionless quantities are introduced by dividing both side of equation (2.34) by ΩR .

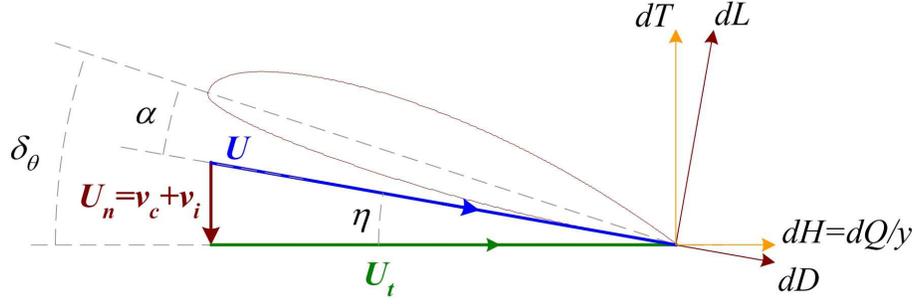


Figure 2.7: Vertical and induced velocity and forces components on the blade airfoil.

$${}^{bl}\mathbf{v}_p = \frac{\mathbf{C}_b^{bl} ({}^b\mathbf{v}_{cg} - {}^b\mathbf{v}_w + {}^b\boldsymbol{\omega} \times {}^b\mathbf{r}_r + {}^b\boldsymbol{\omega}_r \times \mathbf{C}_{bl}^b \tau R \mathbf{i}_{bl}) + v_i \mathbf{k}_{bl}}{\Omega R} = \begin{bmatrix} u_r \\ u_t \\ u_n \end{bmatrix} \quad (2.47)$$

Expanding equation (2.47) in terms of the three components results in:

$$u_r = \frac{1}{\Omega R} [\cos \Psi (u - u_w - qz_{r_{cg}} - (r - \Omega) \tau R \sin \Psi) + \sin \Psi (v - v_w + pz_{r_{cg}} + (r - \Omega) \tau R \cos \Psi)] \quad (2.48)$$

$$u_t = \frac{1}{\Omega R} [-\sin \Psi (u - u_w - qz_{r_{cg}} - (r - \Omega) \tau R \sin \Psi) + \cos \Psi (v - v_w + pz_{r_{cg}} + (r - \Omega) \tau R \cos \Psi)] \quad (2.49)$$

$$u_n = \frac{1}{\Omega R} [w - w_w + p\tau R \sin \Psi - q\tau R \cos \Psi - v_i] \quad (2.50)$$

where u , v , and w are the forward, lateral, and heave velocities of the helicopter, u_w , v_w , and w_w are the components of the wind velocity and p , q , and r are the roll, pitch and yaw angular velocities of the helicopter.

Substituting u_t and u_n into equation (2.45) and doing the corresponding

integrals results in

$$\begin{aligned}
T_r = & \pi\rho R^2(\Omega R)^2 \left\{ -\frac{SC_{L\alpha}}{4} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \left[\frac{(v-v_w)+pz_{rcg}}{\Omega R} \frac{(r-\Omega)}{\Omega} \right] \delta_{\theta_{lat}} + \right. \\
& \frac{SC_{L\alpha}}{4} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \left[\frac{(u-u_w)-qz_{rcg}}{\Omega R} \frac{(r-\Omega)}{\Omega} \right] \delta_{\theta_{lon}} + \\
& \left\{ \frac{SC_{L\alpha}}{6} \left(\frac{R_2^3}{R^3} - \frac{R_h^3}{R^3} \right) \left[\frac{(\Omega-r)}{\Omega} \right]^2 + \frac{SC_{L\alpha}}{4} \left(\frac{R_2}{R} - \frac{R_h}{R} \right) \right. \\
& \left. \left(\left[\frac{(v-v_w)+pz_{rcg}}{\Omega R} \right]^2 + \left[\frac{(u-u_w)-qz_{rcg}}{\Omega R} \right]^2 \right) \right\} \delta_{\theta_{col}} + \\
& \left. \frac{SC_{L\alpha}}{8} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \left(\frac{2(\Omega-r)}{\Omega} \left[\frac{(w-w_w)}{\Omega R} - \frac{v_i}{\Omega R} \right] + \frac{(u-u_w)}{\Omega R} \frac{p}{\Omega} + \frac{(v-v_w)}{\Omega R} \frac{q}{\Omega} \right) \right\}
\end{aligned}$$

For simplification of the notation, the following terms are defined

$$\begin{aligned}
\kappa_1 &= \frac{R_2^4}{R^4} - \frac{R_h^4}{R^4}, & \kappa_2 &= \frac{R_2^3}{R^3} - \frac{R_h^3}{R^3}, & \kappa_3 &= \frac{R_2^2}{R^2} - \frac{R_h^2}{R^2}, & \kappa_4 &= \frac{R_2}{R} - \frac{R_h}{R} \\
\mu_p &= \frac{p}{\Omega}, & \mu_q &= \frac{q}{\Omega}, & \mu_r &= \frac{(\Omega-r)}{\Omega}, & \mu_x &= \frac{(u-u_w)}{\Omega R}, \\
\mu_y &= \frac{(v-v_w)}{\Omega R}, & \mu_z &= \frac{(w-w_w)}{\Omega R}, & \lambda_i &= \frac{v_i}{\Omega R}, \\
v_{hub} &= \frac{(v-v_w)+pz_{rcg}}{\Omega R}, & u_{hub} &= \frac{(u-u_w)-qz_{rcg}}{\Omega R}.
\end{aligned} \tag{2.51}$$

and substituted into the previous expression, which results in

$$\begin{aligned}
T_r = & \frac{\rho R^4 \Omega^2 SC_{L\alpha} \kappa_3}{8} [2\mu_r (\mu_z - \lambda_i) + \mu_p \mu_x + \mu_q \mu_y] + \\
& \frac{\rho R^4 \Omega^2 SC_{L\alpha}}{4} \left\{ \left(\frac{2}{3} \kappa_2 \mu_r^2 + \kappa_4 [v_{hub}^2 + u_{hub}^2] \right) \delta_{\theta_{col}} + \right. \\
& \left. \kappa_3 \mu_r v_{hub} \delta_{\theta_{lat}} - \kappa_3 \mu_r u_{hub} \delta_{\theta_{lon}} \right\}
\end{aligned} \tag{2.52}$$

Splitting equation (2.52) in two terms, one which is independent of the angle rotor blade δ_θ command, denoted T_{r_o} , and the other which is dependent on δ_θ and denoted $T_{r_{\delta_\theta}}$, results in:

$$T_{r_o} = \frac{\rho R^4 \Omega^2 \mathcal{S}}{8} C_{L\alpha} \kappa_3 [2\mu_r (\mu_z - \lambda_i) + \mu_p \mu_x + \mu_q \mu_y] \tag{2.53}$$

and

$$T_{r\theta} = \frac{\rho R^4 \Omega^2 \mathcal{S}}{4} C_{L\alpha} \begin{bmatrix} \frac{2}{3} \kappa_2 \mu_r^2 + \kappa_4 (u_{hub}^2 + v_{hub}^2) \\ \kappa_3 \mu_r v_{hub} \\ -\kappa_3 \mu_r u_{hub} \end{bmatrix}^T \quad (2.54)$$

where the angle input vector command is just $\delta_\theta = [\delta_{\theta_{col}}, \delta_{\theta_{lat}}, \delta_{\theta_{lon}}]$

Equations (2.53) and (2.54) represents the main rotor thrust components of the affine-mode equation (2.46) [22].

Although the main rotor thrust depends exclusively on the collective command $\delta_{\theta_{col}}$, and the lateral and longitudinal commands $\delta_{\theta_{lat}}$ and $\delta_{\theta_{lon}}$, command vector δ_θ will be augmented to include nonlinear terms that appear on the drag torque and the in-plane forces as will be shown in the following sections.

The induced velocity in equation (2.47) is required to compute the thrust; However, this computation is an iterative process. For example, the method of [5] is often used to compute the thrust. In this work a closed form proposed by [35] based on a modification of the Glauert's equation is used –see [36] and [22]. This method computes the induced velocity v_i as

$$v_i = \frac{v_a V_h}{\sqrt{1 + \mu^2}} \quad (2.55)$$

where V_h is the normal induced velocity in hover given by

$$V_h = \sqrt{\frac{T}{2\rho A}}$$

where T is the thrust, and A the rotor disc area. μ is what is called the advanced ratio, $\mu = V \cos \alpha / V_h \approx V / V_h$. The term v_a is given by

$$v_a = \begin{cases} -\frac{1}{2} V_a - \sqrt{\frac{V_a^2}{4} - 1} & \forall V_a \leq -2 \\ 1 - \frac{1}{2} V_a + \frac{25 V_a^2}{12} + \frac{7}{6} V_a^3 & \forall -2 < V_a < 0 \\ -\frac{1}{2} V_a + \sqrt{\frac{V_a^2}{4} + 1} & \forall V_a \geq 0 \end{cases} \quad (2.56)$$

where V_a is the normal component of the free stream air velocity oncoming to the rotor disk. Using equations (2.55) and (2.56) the induced velocity can be approximately computed in a closed form without iteration.

2.4.2 Main Rotor Drag Torque

The aerodynamic drag torque generated by the main rotor is computed in similar way to that of the rotor thrust. Referring to figure 2.5 the drag torque is given by

$$dQ_r = y dH$$

but the in-plane force is

$$dH = dD \cos \eta + dL \sin \eta$$

substituting the drag and lift, equations (2.35) and (2.36), results in

$$dQ_r = y \left(\frac{1}{2} \rho U^2 c C_D \cos \eta dy + \frac{1}{2} \rho U^2 c C_L \sin \eta dy \right) \quad (2.57)$$

The drag coefficient C_D has two components. The first one is known as the zero lift-drag coefficient denoted as C_{D_0} . This coefficient is independent of the lift generation and needs to be estimated. The second one is the induced drag denoted by C_{D_i} which is related to any surface that generates lift.

The induced drag coefficient for subsonic flight is computed as follow:

$$C_{D_i} = \frac{C_L^2}{\pi e \mathcal{R}}$$

Where \mathcal{R} is known as the wing aspect ratio. For constant wing cord the aspect ratio is given by [37]

$$\mathcal{R} = \frac{b}{c}$$

where b is the wing span and c is the cord. The term e is known as the Oswald number, and for a zero angle wing respect to the fuselage is given by [37]

$$e = 1.78 (1 - 0.045 \mathcal{R}^{0.68}) - 0.64$$

Substituting these expressions into equation (2.57), and approximating $\cos \eta \approx 1$, and $\sin \eta \approx \eta$ the torque equation reduces to

$$dQ_r = \frac{1}{2} \rho U^2 c \left[\left(C_{D_0} + \frac{C_L^2}{\pi e \mathcal{R}} \right) + C_L \eta \right] y dy$$

From figure 2.5 can be seen that $C_L = C_{L_\alpha} \alpha = C_{L_\alpha} \left(\delta_\theta - \frac{U_n}{U_t} \right)$ and $\frac{U_n}{U_t} = \tan \eta$

and approximating $\tan \eta \approx \eta$ results in

$$dQ_r = \frac{1}{2} \rho U^2 c \left[C_{D_0} + \frac{C_{L_\alpha}^2 \left(\delta_\theta - \frac{U_n}{U_t} \right)^2}{\pi e \mathcal{R}} + C_{L_\alpha} \left(\delta_\theta - \frac{U_n}{U_t} \right) \frac{U_n}{U_t} \right] y dy$$

recalling than $U \approx U_t$ due to small η assumption yields

$$dQ_r = \frac{1}{2} \rho c \left[C_{D_0} U_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta U_t - U_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta U_t U_n - U_n^2) \right] y dy$$

Referring to figure 2.5, $y = \tau R$ and $d\tau = \frac{1}{R} dy$ so substituting into the torque equation results in

$$dQ_r = \frac{1}{2} \rho c R^2 \left[C_{D_0} U_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta U_t - U_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta U_t U_n - U_n^2) \right] \tau d\tau \quad (2.58)$$

The definition of torque coefficient given by [34] is used,

$$dC_{Q_r} = \frac{dQ_r}{\rho A R (\Omega R)^2} = \frac{dQ_r}{\rho \pi R^2 R (\Omega R)^2} \quad (2.59)$$

and dQ_r is substituted from equation (2.58) into equation (2.59) and the contribution of both blades are added. Then the torque coefficient results in

$$dC_{Q_r} = \frac{2c}{2\pi R (\Omega R)^2} \left[C_{D_0} U_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta U_t - U_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta U_t U_n - U_n^2) \right] \tau d\tau$$

As in the previous section, using the solidity factor \mathcal{S} and the dimensionless velocities $u_t = \frac{U_t}{\Omega R}$ and $u_n = \frac{U_n}{\Omega R}$ yields

$$dC_{Q_r} = \frac{\mathcal{S}}{2} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_t u_n - u_n^2) \right] \tau d\tau \quad (2.60)$$

Substituting equation (2.60) into equation (2.59) results in

$$dQ_r = \frac{\rho \pi R^5 \Omega^2 \mathcal{S}}{2} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_t u_n - u_n^2) \right] \tau d\tau \quad (2.61)$$

Since δ_θ is function of the azimuth angle Ψ , it is necessary to average equation

(2.61) over one complete revolution resulting in

$$Q_r = \frac{\rho R^5 \Omega^2 \mathcal{S}}{4} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \int_0^{2\pi} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_t u_n - u_n^2) \right] \tau d\Psi d\tau \quad (2.62)$$

Substituting the cyclic pitch angle given by [33] into equation (2.62), and solving the corresponding integrals the drag torque reduces to

$$Q_r = Q_{r_0} + Q_{r_\theta} \delta_\theta \quad (2.63)$$

where Q_{r_0} and Q_{r_θ} are

$$Q_{r_0} = \frac{\rho R^5 \Omega^2 \mathcal{S} C_{L_\alpha}}{8e\mathcal{R}} \left\{ \frac{1}{2} (C_{L_\alpha} - \pi e \mathcal{R}) [4\kappa_3 (\mu_z - \lambda_i)^2 + \kappa_1 (\mu_p^2 + \mu_q^2)] - \pi e \mathcal{R} \frac{C_{D_0}}{C_{L_\alpha}} [\kappa_1 \mu_r^2 + \kappa_3 (u_{hub}^2 + v_{hub}^2)] \right\} \quad (2.64)$$

and

$$Q_{r_\theta} = \frac{\rho R^5 \Omega^2 \mathcal{S} C_{L_\alpha}}{e\mathcal{R}} \begin{bmatrix} \frac{1}{8} C_{L_\alpha} [\kappa_3 (u_{hub}^2 + v_{hub}^2) + \kappa_1 \mu_r^2] \\ \frac{1}{16} C_{L_\alpha} \left[\frac{\kappa_3}{2} (u_{hub}^2 + 3v_{hub}^2) + \kappa_1 \mu_r^2 \right] \\ \frac{1}{16} C_{L_\alpha} \left[\frac{\kappa_3}{2} (3u_{hub}^2 + v_{hub}^2) + \kappa_1 \mu_r^2 \right] \\ \frac{1}{3} C_{L_\alpha} \kappa_2 \mu_r v_{hub} \\ -\frac{1}{3} C_{L_\alpha} \kappa_2 \mu_r u_{hub} \\ -\frac{1}{8} C_{L_\alpha} \kappa_3 u_{hub} v_{hub} \\ -\frac{\kappa_2 (\pi e \mathcal{R} - 2C_{L_\alpha})}{6} \left[(\lambda_i - \mu_z) \mu_r + \frac{\mu_p \mu_x}{2} + \frac{\mu_q \mu_y}{2} \right] \\ -\frac{(\pi e \mathcal{R} - 2C_{L_\alpha})}{8} [\kappa_3 v_{hub} (\mu_z - \lambda_i) - \kappa_1 \mu_q \mu_r] \\ -\frac{(\pi e \mathcal{R} - 2C_{L_\alpha})}{8} [\kappa_3 u_{hub} (\mu_z - \lambda_i) - \kappa_1 \mu_p \mu_r] \end{bmatrix}^T \quad (2.65)$$

and the augmented command vector δ_θ is

$$\delta_\theta = [\delta_{\theta_{col}}^2, \delta_{\theta_{lat}}^2, \delta_{\theta_{lon}}^2, \delta_{\theta_{col}} \delta_{\theta_{lat}}, \delta_{\theta_{col}} \delta_{\theta_{lon}}, \delta_{\theta_{lat}} \delta_{\theta_{lon}}, \delta_{\theta_{col}}, \delta_{\theta_{lat}}, \delta_{\theta_{lon}}]^T$$

Equation (2.63) represent the affine-mode [22] for the drag torque component of the Air Star Evolution helicopter.

2.4.3 Roll and Pitch Moments

Roll and pitch moments are generated by the cyclic lateral and longitudinal commands, $\delta_{\theta_{lat}}$ and $\delta_{\theta_{lon}}$ respectively. When $\delta_{\theta_{lat}}$ is commanded, one side of the lower Swash-Plate mechanism decreases its position with respect to the collective angle, while the other side increases. This difference in position happens in the direction of the y_b axis as can be seen from figure 2.8. However, because of the upper Swash-Plate mechanism, this increment and decrement in position is translated to x_b axis. The pitch angle in the blades as a result of the exerted lateral commands is a function of the azimuth angle Ψ . As a consequence of this, it produces a differential thrust in the rotor disc, see figure 2.8. For the lateral command, the pitch angle is maximum when the blades are aligned with the x_b axis, that is when the blades points to the helicopter's nose and rear, making the pitch angle in the lateral command function of $\cos \Psi$ as can be seen in figure 2.8. Therefore, a pitch moment M_ϕ is generated, which

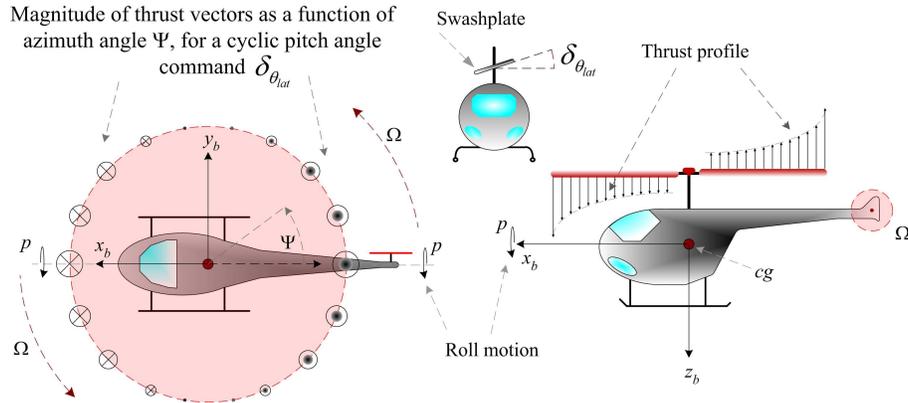


Figure 2.8: Induced roll moment due to the differential thrust produced by the lateral cyclic command.

causes a roll motion, due to the gyroscopic effect generated by the spinning blades, as can be appreciated in the term $\omega \times I_s \Omega$ on equation (2.33).

Similar situation happens when a longitudinal command $\delta_{\theta_{lon}}$ is exerted. In this case, the lower Swash-Plate mechanism increases and decreases its position with respect the collective angle in the direction of the x_b axis. However, because the upper Swash-Plate mechanism, it produces a differential increment in the direction of the y_b axis. In this case, the pitch blade angle reaches its maximum and minimum when the blades are aligned with the y_b axis, that is, $\delta_{\theta_{lon}} \sin \Psi$, as can be seen in figure 2.9.

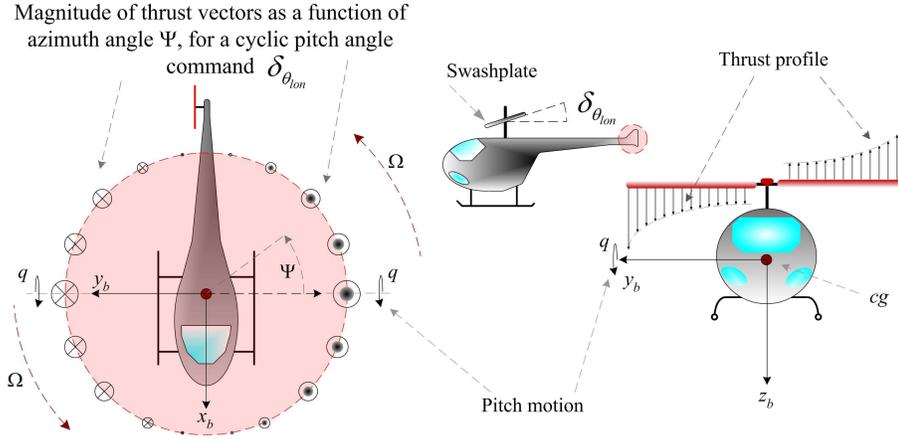


Figure 2.9: Induced pitch moment due to the differential thrust produced by the longitudinal cyclic command.

This action induces a pitch motion about the the helicopter's cg due to the gyroscopic effect.

Referring to figure 2.5 and figure 2.7, the roll moment can be written as:

$$dM_\phi = dL \sin(-\Psi)y = -dL \sin \Psi y$$

Since $dL = \frac{1}{2}\rho U^2 c dy C_L$ and $C_L = C_{L_\alpha} \alpha$ and from figure 2.7 $\alpha = (\delta_\theta - \eta)$ then

$$dL = \frac{1}{2}\rho U^2 c dy C_{L_\alpha} (\delta_\theta - \eta)$$

substituting this expression into dM_ϕ equation yields

$$dM_\phi = -\frac{1}{2}\rho U^2 c dy C_{L_\alpha} (\delta_\theta - \eta) \sin \Psi y$$

From figure 2.7, $\frac{U_n}{U_t} = \tan \eta$, and using the small η approximation, $\frac{U_n}{U_t} \approx \eta$ and $U \approx U_t$, results in

$$dM_\phi = -\frac{1}{2}\rho c C_{L_\alpha} \left(\delta_\theta - \frac{U_n}{U_t} \right) U_t^2 y \sin \Psi dy$$

Defining the dimensionless differential roll coefficient dC_{M_ϕ} as [22]:

$$dC_{M_\phi} = \frac{dM_\phi}{A \rho R (\Omega R)^2}$$

where $A = \pi R^2$ is the main rotor disc area, results in

$$dC_{M_\phi} = \frac{dM_\phi}{\pi R^2 \rho R (\Omega R)^2} \quad (2.66)$$

Substituting dM_ϕ into equation (2.66) and multiplying by 2 due to account for two blades yields

$$dC_{M_\phi} = -\frac{C_{L_\alpha}}{2R^2} \frac{2c}{\pi R} \left[\delta_\theta \left(\frac{U_t}{\Omega R} \right)^2 - \frac{U_n}{\Omega R} \frac{U_t}{\Omega R} \right] y \sin \Psi dy$$

Substituting the solidity factor $\mathcal{S} = \frac{2c}{\pi R}$ and the dimensionless velocities $u_t = \frac{U_t}{\Omega R}$ and $u_n = \frac{U_n}{\Omega R}$ yields

$$dC_{M_\phi} = -\frac{C_{L_\alpha} \mathcal{S}}{2R^2} \left(\delta_\theta - \frac{u_n}{u_t} \right) u_t^2 y \sin \Psi dy$$

Introducing a variable change of $\tau = \frac{y}{R}$ and $d\tau = \frac{dy}{R}$ to this equation results in

$$dC_{M_\phi} = -\frac{C_{L_\alpha} \mathcal{S}}{2} \left(\delta_\theta - \frac{u_n}{u_t} \right) u_t^2 \tau \sin \Psi d\tau \quad (2.67)$$

As in the previous cases, equation (2.67) needs to be integrated over one complete revolution and all along the blade's length as follows:

$$C_{M_\phi} = -\frac{\mathcal{S}}{2} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{L_\alpha} \left(\delta_\theta - \frac{u_n}{u_t} \right) u_t^2 \sin \Psi d\Psi \right] \tau d\tau$$

Substituting the Fourier expansion of δ_θ [33], the value of the dimensionless velocities u_n and u_t given by equations (2.50) and (2.49), and solving the integrals yields

$$\begin{aligned} C_{M_\phi} = & -\frac{1}{8} \mathcal{S} C_{L_\alpha} \left[\frac{1}{2} \left(\frac{R_2^4}{R^4} - \frac{R_h^4}{R^4} \right) \frac{p(\Omega-r)}{\Omega^2} + \frac{1}{8} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \frac{(u-u_w-qz_{rcg})(w-w_w-v_i)}{\Omega^2 R^2} \right] \\ & -\frac{1}{6} \left(\frac{R_2^3}{R^3} - \frac{R_h^3}{R^3} \right) \mathcal{S} C_{L_\alpha} \frac{(\Omega-R)(u-u_w-qz_{rcg})}{\Omega^2 R} \delta_{\theta_{col}} + \\ & \frac{1}{16} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \mathcal{S} C_{L_\alpha} \frac{(v-v_w+pz_{rcg})(u-u_w-qz_{rcg})}{\Omega^2 R^2} \delta_{\theta_{lat}} \\ & -\frac{\mathcal{S} C_{L_\alpha}}{16} \left[\left(\frac{R_2^4}{R^4} - \frac{R_h^4}{R^4} \right) \frac{(\Omega-r)^2}{\Omega^2} + \frac{1}{2} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \frac{(v-v_w+pz_{rcg})^2 + 3(u-u_w-qz_{rcg})^2}{\Omega^2 R^2} \right] \delta_{\theta_{lon}} \end{aligned}$$

After substitute the definition of equation (2.51) results in

$$C_{M_\phi} = -\frac{SC_{L\alpha}}{8} [\kappa_3 (\mu_z - \lambda_i) u_{hub} + \frac{1}{2}\kappa_1\mu_p\mu_r] - \frac{SC_{L\alpha}}{6}\kappa_2u_{hub}\mu_r\delta_{\theta_{col}} + \frac{SC_{L\alpha}}{16}\kappa_3u_{hub}v_{hub}\delta_{\theta_{lat}} - \frac{SC_{L\alpha}}{16} [\kappa_1\mu_r^2 + \frac{1}{2}\kappa_3(3u_{hub}^2 + v_{hub}^2)] \delta_{\theta_{lon}}$$

From equation (2.66) is apparent that the roll moment M_ϕ can be obtained by multiplying the roll coefficient C_{M_ϕ} by the term $\pi\rho\Omega^2R^5$. For convenience the equation is divided into two terms which results in

$$M_\phi = M_{\phi_0} + M_{\phi_\theta}\delta_\theta \quad (2.68)$$

where

$$M_{\phi_0} = -\pi\rho\Omega^2R^5\frac{SC_{L\alpha}}{8} \left[\kappa_3 (\mu_z - \lambda_i) u_{hub} + \frac{1}{2}\kappa_1\mu_p\mu_r \right] \quad (2.69)$$

and

$$M_{\phi_\theta} = \pi\rho\Omega^2R^5\frac{SC_{L\alpha}}{6} \begin{bmatrix} -\kappa_2\mu_r u_{hub} \\ \frac{3}{8}\kappa_3u_{hub}v_{hub} \\ -\frac{3}{8} [\kappa_1\mu_r^2 + \frac{1}{2}\kappa_3(3u_{hub}^2 + v_{hub}^2)] \end{bmatrix}^T \quad (2.70)$$

The derivation of the pitch moment M_θ proceeds exactly as the previous derivation of the roll moment.

From figures 2.5 and 2.7 the differential pitch moment dM_θ is written as

$$dM_\theta = dL \cos \Psi y$$

Substituting the differential lift and using the small η approximation the differential pitch moment becomes

$$dM_\theta = \frac{1}{2}\rho c C_{L\alpha} \left(\delta_\theta - \frac{U_n}{U_t} \right) U_t^2 y \cos \Psi dy$$

The dimensionless differential pitch coefficient dC_{M_θ} is defined as:

$$dC_{M_\theta} = \frac{dM_\theta}{\pi R^2 \rho R (\Omega R)^2} \quad (2.71)$$

Substituting dM_θ into equation (2.71), adding the effect of the second blade,

using the solidity factor, doing the variable change $\tau = \frac{y}{R}$ and using the dimensionless velocities u_t and u_n results in

$$dC_{M_\theta} = \frac{C_{L_\alpha} \mathcal{S}}{2} \left(\delta_\theta - \frac{u_n}{u_t} \right) u_t^2 \tau \cos \Psi d\tau \quad (2.72)$$

Equation (2.72) must be integrated over one complete revolution and along the blades length as follows:

$$C_{M_\theta} = \frac{\mathcal{S}}{2} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{L_\alpha} \left(\delta_\theta - \frac{u_n}{u_t} \right) u_t^2 \cos \Psi d\Psi \right] \tau d\tau$$

Substituting the Fourier expansion of δ_θ [34] and [33], and the dimensionless velocities u_n and u_t given by equations (2.50) and (2.49), and then doing the integrals yields

$$\begin{aligned} C_{M_\theta} = & -\frac{1}{8} \mathcal{S} C_{L_\alpha} \left[\frac{1}{2} \left(\frac{R_2^4}{R^4} - \frac{R_h^4}{R^4} \right) \frac{q(\Omega-r)}{\Omega^2} + \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \frac{(w-w_w-v_i)(v-v_w+pz_{rcg})}{\Omega^2 R^2} \right] - \\ & \frac{1}{6} \mathcal{S} C_{L_\alpha} \left(\frac{R_2^3}{R^3} - \frac{R_h^3}{R^3} \right) \frac{(\Omega-r)(v-v_w+pz_{rcg})}{\Omega^2 R} \delta_{\theta_{col}} + \\ & \frac{1}{16} \mathcal{S} C_{L_\alpha} \left[\left(\frac{R_2^4}{R^4} - \frac{R_h^4}{R^4} \right) \frac{(\Omega-r)^2}{\Omega^2} + \frac{1}{2} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \frac{(u-u_w-qz_{rcg})^2 + 3(v-v_w+pz_{rcg})^2}{\Omega^2 R^2} \right] \delta_{\theta_{lat}} \\ & - \frac{1}{16} \left(\frac{R_2^2}{R^2} - \frac{R_h^2}{R^2} \right) \mathcal{S} C_{L_\alpha} \frac{(v-v_w+pz_{rcg})(u-u_w-qz_{rcg})}{\Omega^2 R^2} \delta_{\theta_{lon}} \end{aligned}$$

Using the definition given by equation (2.51), and rearranging the roll coefficient becomes

$$\begin{aligned} C_{M_\theta} = & -\frac{1}{8} \mathcal{S} C_{L_\alpha} \left[\frac{1}{2} \kappa_1 \mu_q \mu_r + \kappa_3 (\mu_z - \lambda_i) v_{hub} \right] - \\ & \frac{1}{6} \mathcal{S} C_{L_\alpha} \kappa_2 \mu_r v_{hub} \delta_{\theta_{col}} + \\ & \frac{1}{16} \mathcal{S} C_{L_\alpha} \left[\kappa_1 \mu_r^2 + \frac{1}{2} \kappa_3 (u_{hub}^2 + 3v_{hub}^2) \right] \delta_{\theta_{lat}} \\ & - \frac{1}{16} \kappa_3 \mathcal{S} C_{L_\alpha} u_{hub} v_{hub} \delta_{\theta_{lon}} \end{aligned}$$

From equation (2.71) it is apparent that the pitch moment M_θ can be obtained by multiplying the pitch coefficient C_{M_θ} by the term $\pi \rho \Omega^2 R^5$, separating the pitch moment in two terms, one independent of the δ_θ commands, and the dependant term yields

$$M_\theta = M_{\theta_0} + M_{\theta_\delta} \delta_\theta \quad (2.73)$$

where

$$M_{\theta_0} = -\frac{\pi\rho\Omega^2 R^5 \mathcal{S}C_{L\alpha}}{8} \left[\frac{1}{2}\kappa_1\mu_q\mu_r + \kappa_3(\mu_z - \lambda_i)v_{hub} \right] \quad (2.74)$$

and

$$M_{\theta_\theta} = \pi\rho R^5 \Omega^2 \frac{\mathcal{S}C_{l\alpha}}{6} \left[\begin{array}{c} -\kappa_2\mu_r v_{hub} \\ \frac{3}{8} [\kappa_1\mu_r^2 + \frac{1}{2}\kappa_3 (u_{hub}^2 + 3v_{hub}^2)] \\ -\frac{3}{8}\kappa_3 u_{hub} v_{hub} \end{array} \right]^T \quad (2.75)$$

In summary equations (2.68) and (2.73) represent the roll and pitch moment of the helicopter in affine-mode.

2.4.4 Main Rotor in-plane Force

In forward flight, the in-plane force in the main rotor disc is present. This force is produced due to the oncoming free air stream v_∞ when the helicopter is moving forward. This results in the profile distribution of the air on each leading edge of the two blades to be different. The air velocity is greater for the blade moving forward in the direction of the flight path, and is lower in the opposite direction yielding an unbalanced aerodynamic drag. This produces a net force \mathbf{H} in the rotor disc or Tip-Path-Plane, see figure 2.4. As expected, the in-plane force is a function of the azimuth angle Ψ and might have any orientation on the Tip-Path-Plane, depending on the flight path direction.

Referring to figure 2.7 we can see that the components of in-plane force \mathbf{H}_r are:

$$H_{r_x} = (dD \cos \eta + dL \sin \eta) \sin(-\Psi) = -(dD \cos \eta + dL \sin \eta) \sin \Psi$$

and

$$H_{r_y} = (dD \cos \eta + dL \sin \eta) \cos(-\Psi) = (dD \cos \eta + dL \sin \eta) \cos \Psi$$

substituting the drag and lift forces given by equations (2.35) and (2.36) yield

$$H_{r_x} = -\left(\frac{1}{2}\rho U^2 c dy C_D \cos \eta + \frac{1}{2}\rho U^2 c dy C_L \sin \eta \right) \sin \Psi$$

$$H_{r_y} = \left(\frac{1}{2} \rho U^2 c dy C_D \cos \eta + \frac{1}{2} \rho U^2 c dy C_L \sin \eta \right) \cos \Psi$$

Substituting the drag coefficient C_D with the zero lift coefficient C_{D_0} plus the induced drag coefficient $C_{D_i} = \frac{C_L^2}{\pi e \mathcal{R}}$ and assuming the small angle approximation of η , the components of \mathbf{H}_r results in

$$H_{r_x} = -\frac{1}{2} \rho U^2 c dy \left(C_{D_0} + \frac{C_L^2}{\pi e \mathcal{R}} + C_L \eta \right) \sin \Psi$$

$$H_{r_y} = \frac{1}{2} \rho U^2 c dy \left(C_{D_0} + \frac{C_L^2}{\pi e \mathcal{R}} + C_L \eta \right) \cos \Psi$$

The lift coefficient is given by $C_L = C_{L_\alpha} \alpha = C_{L_\alpha} (\delta_\theta - \eta)$ and η can be approximated by $\frac{U_n}{U_t}$, and $U \approx U_t$, see figure 2.7, H_{r_x} and H_{r_y} become

$$H_{r_x} = -\frac{1}{2} \rho U_t^2 c dy \left[C_{D_0} + \frac{C_{L_\alpha}^2 \left(\delta_\theta - \frac{U_n}{U_t} \right)^2}{\pi e \mathcal{R}} + C_{L_\alpha} \left(\delta_\theta - \frac{U_n}{U_t} \right) \frac{U_n}{U_t} \right] \sin \Psi$$

$$H_{r_y} = \frac{1}{2} \rho U_t^2 c dy \left[C_{D_0} + \frac{C_{L_\alpha}^2 \left(\delta_\theta - \frac{U_n}{U_t} \right)^2}{\pi e \mathcal{R}} + C_{L_\alpha} \left(\delta_\theta - \frac{U_n}{U_t} \right) \frac{U_n}{U_t} \right] \cos \Psi$$

Defining the dimensionless differential in-plane force coefficient as

$$dC_{H_{r_x}} = \frac{dH_{r_x}}{\pi R^2 \rho (\Omega R)^2}$$

$$dC_{H_{r_y}} = \frac{dH_{r_y}}{\pi R^2 \rho (\Omega R)^2}$$

and, substituting the corresponding differential forces yield

$$dC_{H_{r_x}} = -\frac{1}{2} \frac{c}{\pi R^2} dy \left[C_{D_0} \frac{U_t^2}{(\Omega R)^2} + \frac{C_{L_\alpha}^2 \left(\delta_\theta \frac{U_t}{\Omega R} - \frac{U_n}{\Omega R} \right)^2}{\pi e \mathcal{R}} + C_{L_\alpha} \left(\delta_\theta \frac{U_n}{\Omega R} \frac{U_t}{\Omega R} - \frac{U_n^2}{(\Omega R)^2} \right) \right] \sin \Psi$$

$$dC_{H_{r_y}} = \frac{1}{2} \frac{c}{\pi R^2} dy \left[C_{D_0} \frac{U_t^2}{(\Omega R)^2} + \frac{C_{L_\alpha}^2 \left(\delta_\theta \frac{U_t}{\Omega R} - \frac{U_n}{\Omega R} \right)^2}{\pi e \mathcal{R}} + C_{L_\alpha} \left(\delta_\theta \frac{U_n}{\Omega R} \frac{U_t}{\Omega R} - \frac{U_n^2}{(\Omega R)^2} \right) \right] \cos \Psi$$

Using the definition of the dimensionless velocities, adding the effect of the second blade and using the solidity factor definition, results in

$$dC_{H_{rx}} = -\frac{\mathcal{S}}{2} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_n u_t - u_n^2) \right] \sin \Psi \frac{dy}{R}$$

$$dC_{H_{ry}} = \frac{\mathcal{S}}{2} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_n u_t - u_n^2) \right] \cos \Psi \frac{dy}{R}$$

In order to get the coefficients, an integration of both components of \mathbf{H}_r all along the span of the blades and all along one revolution about the azimuth angle Ψ is needed. Using the variable change $d\tau = \frac{dy}{R}$, and with the integrals the coefficients are

$$C_{H_{rx}} = -\frac{\mathcal{S}}{2} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_n u_t - u_n^2) \right] \sin \Psi d\Psi d\tau$$

$$C_{H_{ry}} = \frac{\mathcal{S}}{2} \int_{\frac{R_h}{R}}^{\frac{R_2}{R}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{D_0} u_t^2 + \frac{C_{L_\alpha}^2 (\delta_\theta u_t - u_n)^2}{\pi e \mathcal{R}} + C_{L_\alpha} (\delta_\theta u_n u_t - u_n^2) \right] \cos \Psi d\Psi d\tau$$

Substituting the Fourier expansion of the cyclic pitch angle [34] and [33], substituting the values of u_t and u_n from equations (2.49) and (2.50), and using the terms definitions given by equation (2.51), the coefficients of the in-plane force become

$$C_{H_{rx}} = -\frac{SC_{L_\alpha} \kappa_3}{4\pi e \mathcal{R}} \left[\mu_p (\mu_z - \lambda_i) (C_{L_\alpha} - \pi e \mathcal{R}) - \pi e \mathcal{R} \frac{C_{D_0}}{C_{L_\alpha}} \mu_r u_{hub} \right] +$$

$$\frac{SC_{L_\alpha}^2 \kappa_3}{4\pi e \mathcal{R}} \mu_r u_{hub} \delta_{\theta_{col}}^2 + \frac{SC_{L_\alpha}^2 \kappa_3}{16\pi e \mathcal{R}} \mu_r u_{hub} \delta_{\theta_{lat}}^2 + \frac{3SC_{L_\alpha}^2 \kappa_3}{16\pi e \mathcal{R}} \mu_r u_{hub} \delta_{\theta_{lon}}^2 +$$

$$\frac{SC_{L_\alpha}^2 \kappa_4}{4\pi e \mathcal{R}} v_{hub} u_{hub} \delta_{\theta_{col}} \delta_{\theta_{lat}} - \frac{SC_{L_\alpha}^2}{6\pi e \mathcal{R}} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (3u_{hub}^2 + v_{hub}^2) \right] \delta_{\theta_{col}} \delta_{\theta_{lon}} -$$

$$\frac{SC_{L_\alpha}^2 \kappa_3}{8\pi e \mathcal{R}} \mu_r v_{hub} \delta_{\theta_{lat}} \delta_{\theta_{lon}} + \frac{SC_{L_\alpha} (\pi e \mathcal{R} - 2C_{L_\alpha})}{4\pi e \mathcal{R}} \left[-\frac{1}{3} \kappa_2 \mu_q \mu_r + \kappa_4 u_{hub} (\mu_z - \lambda_i) \right] \delta_{\theta_{col}} -$$

$$\frac{SC_{L_\alpha} \kappa_3 (\pi e \mathcal{R} - 2C_{L_\alpha})}{32\pi e \mathcal{R}} [\mu_q u_{hub} + \mu_p v_{hub}] \delta_{\theta_{lat}} -$$

$$\frac{SC_{L_\alpha} \kappa_3 (\pi e \mathcal{R} - 2C_{L_\alpha})}{8\pi e \mathcal{R}} \left[\mu_r (\mu_z - \lambda_i) - \frac{3}{4} \mu_p \mu_x - \frac{1}{4} \mu_q v_{hub} + \frac{3}{4} \mu_q \mu_p Z'_{rcg} \right] \delta_{\theta_{lon}}$$

and

$$\begin{aligned}
C_{H_{r_y}} = & \frac{SC_{L_\alpha} \kappa_3}{4\pi e \mathcal{R}} \left[\mu_q (\mu_z - \lambda_i) (C_{L_\alpha} - \pi e \mathcal{R}) - \pi e \mathcal{R} \frac{C_{D_0}}{C_{L_\alpha}} \mu_r v_{hub} \right] - \\
& \frac{SC_{L_\alpha}^2 \kappa_3}{4\pi e \mathcal{R}} \mu_r v_{hub} \delta_{\theta_{col}}^2 - \frac{3SC_{L_\alpha}^2 \kappa_3}{16\pi e \mathcal{R}} \mu_r v_{hub} \delta_{\theta_{lat}}^2 - \frac{SC_{L_\alpha}^2 \kappa_3}{16\pi e \mathcal{R}} \mu_r v_{hub} \delta_{\theta_{lon}}^2 - \\
& \frac{SC_{L_\alpha}^2}{6\pi e \mathcal{R}} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (u_{hub}^2 + 3v_{hub}^2) \right] \delta_{\theta_{col}} \delta_{\theta_{lat}} + \frac{SC_{L_\alpha}^2 \kappa_4}{4\pi e \mathcal{R}} v_{hub} u_{hub} \delta_{\theta_{col}} \delta_{\theta_{lon}} + \\
& \frac{SC_{L_\alpha}^2 \kappa_3}{8\pi e \mathcal{R}} \mu_r u_{hub} \delta_{\theta_{lat}} \delta_{\theta_{lon}} + \frac{SC_{L_\alpha} (\pi e \mathcal{R} - 2C_{L_\alpha})}{4\pi e \mathcal{R}} \left[\frac{1}{3} \kappa_2 \mu_q \mu_r - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right] \delta_{\theta_{col}} + \\
& \frac{SC_{L_\alpha} \kappa_3 (\pi e \mathcal{R} - 2C_{L_\alpha})}{8\pi e \mathcal{R}} \left[-\mu_r (\mu_z - \lambda_i) + \frac{3}{4} \mu_p \mu_y + \frac{1}{4} \mu_p u_{hub} + \frac{3}{4} \mu_q \mu_p Z'_{rcg} \right] \delta_{\theta_{lat}} - \\
& \frac{SC_{L_\alpha} \kappa_3 (\pi e \mathcal{R} - 2C_{L_\alpha})}{32\pi e \mathcal{R}} [\mu_q u_{hub} + \mu_p v_{hub}] \delta_{\theta_{lon}}
\end{aligned}$$

where

$$Z'_{rcg} = \frac{z_{rcg}}{R}$$

The components of the in-plane force \mathbf{H}_r are obtained by multiplying the corresponding coefficients by the term $\pi \rho R^4 \Omega^2$. Therefore, the in-plane components in affine-mode form becomes

$$H_{r_x} = H_{r_{x_o}} + H_{r_{x_\theta}} \delta_\theta \quad (2.76)$$

and

$$H_{r_y} = H_{r_{y_o}} + H_{r_{y_\theta}} \delta_\theta \quad (2.77)$$

where

$$H_{r_{x_o}} = -\frac{\rho R^4 \Omega^2 SC_{L_\alpha} \kappa_3}{4e \mathcal{R}} \left[\mu_p (\mu_z - \lambda_i) (C_{L_\alpha} - \pi e \mathcal{R}) - \pi e \mathcal{R} \frac{C_{D_0}}{C_{L_\alpha}} \mu_r u_{hub} \right] \quad (2.78)$$

$$H_{r_{x\theta}} = \frac{\rho R^4 \Omega^2 S C_{L\alpha}}{e \mathcal{R}} \begin{bmatrix} \frac{1}{4} C_{L\alpha} \kappa_3 \mu_r u_{hub} \\ \frac{1}{16} C_{L\alpha} \kappa_3 \mu_r u_{hub} \\ \frac{3}{16} C_{L\alpha} \kappa_3 \mu_r u_{hub} \\ \frac{1}{4} C_{L\alpha} \kappa_4 v_{hub} u_{hub} \\ -\frac{C_{L\alpha}}{6} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (3u_{hub}^2 + v_{hub}^2) \right] \\ -\frac{1}{8} C_{L\alpha} \kappa_3 \mu_r v_{hub} \\ \frac{1}{4} (\pi e \mathcal{R} - 2C_{L\alpha}) \left[-\frac{1}{3} \kappa_2 \mu_q \mu_r + \kappa_4 u_{hub} (\mu_z - \lambda_i) \right] \\ -\frac{1}{32} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) [\mu_q u_{hub} + \mu_p v_{hub}] \\ -\frac{1}{8} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left[\mu_r (\mu_z - \lambda_i) \right. \\ \left. - \frac{3}{4} \mu_p \mu_x - \frac{1}{4} \mu_q v_{hub} + \frac{3}{4} \mu_q \mu_p z'_{rcg} \right] \end{bmatrix}^T \quad (2.79)$$

$$H_{r_{y\theta}} = \frac{\rho R^4 \Omega^2 S C_{L\alpha} \kappa_3}{4e \mathcal{R}} \left[\mu_q (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) - \pi e \mathcal{R} \frac{C_{D_0}}{C_{L\alpha}} \mu_r v_{hub} \right] \quad (2.80)$$

$$H_{r_{y\theta}} = \frac{\rho R^4 \Omega^2 S C_{L\alpha}}{e \mathcal{R}} \begin{bmatrix} -\frac{1}{4} C_{L\alpha} \kappa_3 \mu_r v_{hub} \\ -\frac{3}{16} C_{L\alpha} \kappa_3 \mu_r v_{hub} \\ -\frac{1}{16} C_{L\alpha} \kappa_3 \mu_r v_{hub} \\ -\frac{C_{L\alpha}}{6} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (u_{hub}^2 + 3v_{hub}^2) \right] \\ \frac{1}{4} C_{L\alpha} \kappa_4 v_{hub} u_{hub} \\ \frac{1}{8} C_{L\alpha} \kappa_3 \mu_r u_{hub} \\ \frac{1}{4} (\pi e \mathcal{R} - 2C_{L\alpha}) \left[\frac{1}{3} \kappa_2 \mu_q \mu_r - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right] \\ \frac{1}{8} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left[-\mu_r (\mu_z - \lambda_i) \right. \\ \left. + \frac{3}{4} \mu_q \mu_y + \frac{1}{4} \mu_p u_{hub} + \frac{3}{4} \mu_q \mu_p Z'_{r_{cg}} \right] \\ \left. -\frac{1}{32} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) [\mu_q u_{hub} + \mu_p v_{hub}] \right] \end{bmatrix}^T \quad (2.81)$$

and the augmented command vector δ_θ is

$$\delta_\theta = [\delta_{\theta_{col}}^2, \delta_{\theta_{lat}}^2, \delta_{\theta_{lon}}^2, \delta_{\theta_{col}} \delta_{\theta_{lat}}, \delta_{\theta_{col}} \delta_{\theta_{lon}}, \delta_{\theta_{lat}} \delta_{\theta_{lon}}, \delta_{\theta_{col}}, \delta_{\theta_{lat}}, \delta_{\theta_{lon}}]^T$$

Equations (2.76) to (2.81) are the in-plane force components expressed in affine-mode[22].

Next the forces of the tail rotor will be outlined. The procedure to derive them is follows this section with the only main difference being the orientation of the tail rotor compared to that of the main rotor.

2.4.5 Tail Rotor Thrust

By using the dimensionless tail thrust coefficient, similar to the main rotor analysis, and recalling that the angle η_t is small, see figure 2.7, the differential thrust coefficient can be written as

$$dC_{T_t} = \frac{d\mathbf{T}_t}{A_t \rho (\Omega_t R_t)^2} \approx \frac{d\mathbf{L}_t}{\pi \rho R_t^2 (\Omega_t R_t)^2} = \frac{\frac{1}{2} \rho U_t^2 c_t dy C_{L_t}}{\pi \rho R_t^2 (\Omega_t R_t)^2}$$

where A_t is the tail rotor disc area, R_t is the radius of the tail rotor disc, and Ω_t is the tail rotor angular speed. Since the definition of lift coefficient $C_{L_t} = C_{L_{\alpha_t}} \alpha_t = C_{L_{\alpha_t}} (\delta_{\theta_t} - \eta_t) = C_{L_{\alpha_t}} \left(\delta_{\theta_t} - \frac{U_{n_t}}{U_{t_t}} \right)$, and the oncoming free air stream U_t is approximated as $U_t \approx U_{t_t}$, the thrust coefficient becomes

$$dC_{T_t} = \frac{1}{2} \frac{c_t}{\pi R_t} C_{L_{\alpha_t}} \left(\delta_{\theta_t} \left[\frac{U_t}{\Omega_t R_t} \right]^2 - \frac{U_{t_t}}{\Omega_t R_t} \frac{U_{n_t}}{\Omega_t R_t} \right) \frac{dy}{R_t}$$

Using the definitions of the dimensionless velocities $u_{t_t} = \frac{U_{t_t}}{\Omega_t R_t}$, $u_{n_t} = \frac{U_{n_t}}{\Omega_t R_t}$, adding the effect of the second blade, recalling the of solidity factor $\mathcal{S}_t = \frac{2c_t}{\pi R_t}$, and introducing the variable change $d\tau = \frac{dy}{R_t}$, the tail thrust coefficients is

$$dC_{T_t} = \frac{\mathcal{S}_t C_{L_{\alpha_t}}}{2} \left(\delta_{\theta_t} - \frac{u_{n_t}}{u_{t_t}} \right) u_{t_t}^2 d\tau \quad (2.82)$$

In equation (2.82) the angle δ_{θ_t} is the pitch angle of the tail blades, and coefficient $C_{L_{\alpha_t}}$ is the lift slope given by the tail airfoil profile. The tail blade normal and tangential air velocities are computed in similar manner to that of the main rotor blades using the blade element theory and the momentum theory. Using reference frames showed in figure 2.10, the velocity of a point p_t on the tail blade expressed in body coordinates is given by:

$$\mathbf{v}_{p_t} = {}^b\mathbf{v}_{cg} + {}^b\mathbf{v}_{r_t} + {}^b\mathbf{v}_{r_p} \quad (2.83)$$

Using figure 2.10 the velocities are calculated as

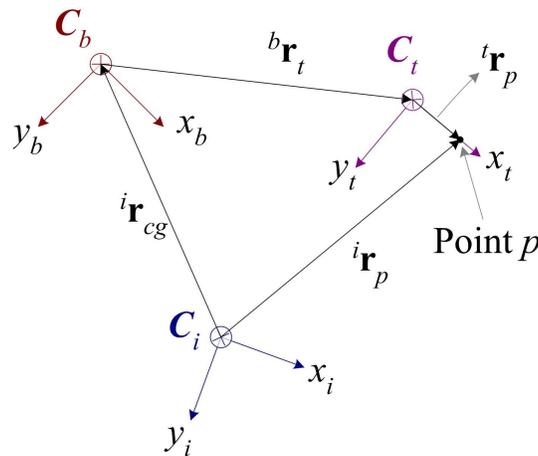


Figure 2.10: Tail rotor and body frames used to compute the velocity of a point p_t on the tail blades.

$$\begin{aligned}
{}^b\mathbf{v}_{cg} &= \mathbf{C}_i^b \frac{{}^i d({}^i \mathbf{r}_{cg})}{dt} \\
{}^b\mathbf{v}_{r_t} &= \frac{{}^i d({}^b \mathbf{r}_t)}{dt} = \frac{{}^b d({}^b \mathbf{r}_t)}{dt} + \boldsymbol{\omega} \times {}^b \mathbf{r}_t \\
{}^b\mathbf{v}_{r_p} &= \mathbf{C}_t^b \frac{{}^i d({}^b \mathbf{r}_{p_t})}{dt} = \mathbf{C}_t^b \left(\frac{{}^t d({}^t \mathbf{r}_{p_t})}{dt} + \boldsymbol{\omega}_t \times {}^t \mathbf{r}_{p_t} \right)
\end{aligned}$$

where $\boldsymbol{\omega}_t = \boldsymbol{\omega} + \boldsymbol{\Omega}_t$ is the total angular velocity of the tail frame, and $\boldsymbol{\Omega}_t = [0, \Omega_t, 0]^T$ is the tail frame angular velocity. The position of the tail rotor \mathbf{r}_t with respect to the body frame \mathbf{C}_b , and the position vector \mathbf{r}_{p_t} of the point p_t in the tail frame \mathbf{C}_t does not change in time. Thus the terms $\frac{{}^b d({}^b \mathbf{r}_t)}{dt}$ and $\frac{{}^t d({}^t \mathbf{r}_{p_t})}{dt}$ are zero. Substituting the corresponding velocities into equation (2.83) and expressing the velocity \mathbf{v}_{P_t} in tail frame results in

$$\mathbf{v}_{P_t} = \mathbf{C}_b^t \left({}^b \mathbf{v}_{cg} + \boldsymbol{\omega} \times {}^b \mathbf{r}_t + \boldsymbol{\omega}_t \times \mathbf{C}_t^{bt} \mathbf{r}_{p_t} \right)$$

Separating the velocity of point p_t into the radial, tangential and normal velocities U_{r_t} , U_{t_t} and U_{n_t} and then dividing both sides by the term $\Omega_t R_t$ to get dimensionless velocities u_{r_t} , u_{t_t} , and u_{n_t} results in

$${}^t \mathbf{v}_{p_t} = \frac{\mathbf{C}_b^t \left({}^b \mathbf{v}_{cg} + \boldsymbol{\omega} \times {}^b \mathbf{r}_t + \boldsymbol{\omega}_t \times \mathbf{C}_t^{bt} \mathbf{r}_{p_t} \right)}{\Omega_t R_t} = \begin{bmatrix} u_{r_t} \\ u_{t_t} \\ u_{n_t} \end{bmatrix} \quad (2.84)$$

Solving for the dimensionless velocities results in

$$u_{r_t} = \frac{1}{\Omega_t R_t} \left[\cos \Psi \left(u - u_w - qz_{t_{cg}} + ry_{t_{cg}} - (q + \Omega_t) \tau R_t \sin \Psi \right) - \sin \Psi \left(w - w_w - py_{t_{cg}} + qx_{t_{cg}} - (q + \Omega_t) \tau R_t \cos \Psi \right) \right] \quad (2.85)$$

$$u_{t_t} = \frac{1}{\Omega_t R_t} \left[-\sin \Psi \left(u - u_w - qz_{t_{cg}} + ry_{t_{cg}} - (q + \Omega_t) \tau R_t \sin \Psi \right) - \cos \Psi \left(w - w_w - py_{t_{cg}} + qx_{t_{cg}} - (q + \Omega_t) \tau R_t \cos \Psi \right) \right] \quad (2.86)$$

$$u_{n_t} = \frac{1}{\Omega_t R_t} \left[v - v_w - rx_{t_{cg}} + pz_{t_{cg}} + \tau R_t \left(r \cos \Psi + p \sin \Psi \right) - v_{it} \right] \quad (2.87)$$

Equation (2.82) requires integration over one complete revolution and all along the tail blade span as:

$$C_{T_t} = \frac{S_t C_{L\alpha_t}}{2} \int_{\frac{R_{h_t}}{R_t}}^{\frac{R_{2_t}}{R_t}} \frac{1}{2\pi} \int_0^{2\pi} \left(\delta_{\theta_t} - \frac{u_{n_t}}{u_{t_t}} \right) u_{t_t}^2 d\Psi d\tau$$

Substituting equations (2.86) and (2.87) into this equation and doing the corresponding integrals yields

$$C_{T_t} = \frac{S_t C_{L\alpha_t}}{8} \left(\frac{R_{t_2}^2}{R_t^2} - \frac{R_{t_h}^2}{R_t^2} \right) \left[\frac{p}{\Omega_t} \left(\frac{(u-u_w)-qz_{tcg}}{\Omega_t R_t} \right) - \frac{2(q+\Omega_t)}{\Omega_t} \left(\frac{(v-v_w)+pz_{tcg}-rx_{tcg}-v_{i_t}}{\Omega_t R_t} \right) \right] + \frac{r}{\Omega_t} \left(\frac{(w-w_w)+qx_{tcg}-\varkappa v_i}{\Omega_t R_t} \right) + \frac{S_t C_{L\alpha_t}}{4} \left[\frac{2}{3} \left(\frac{R_{t_2}^3}{R_t^3} - \frac{R_{t_h}^3}{R_t^3} \right) \left(\frac{(q+\Omega_t)}{\Omega_t} \right)^2 + \left(\frac{R_{t_2}^4}{R_t^4} - \frac{R_{t_h}^4}{R_t^4} \right) \left\{ \left(\frac{(u-u_w)-qz_{tcg}}{\Omega_t R_t} + \frac{r}{\Omega_t} \frac{y_{tcg}}{R_t} \right)^2 + \left(\frac{(w-w_w)+qx_{tcg}-\varkappa v_i}{\Omega_t R_t} - \frac{p}{\Omega_t} \frac{y_{tcg}}{R_t} \right)^2 \right\} \right] \delta_{\theta_t}$$

Where $\varkappa v_i$ is the contribution of the induced velocity of the main rotor in the z_b axis added to the wind velocity w_w .

To simplify the notation the following terms are defined

$$\begin{aligned} \mathbf{u}_t &= \frac{(u-u_w)-qz_{tcg}}{\Omega_t R_t}, & \mathbf{v}_t &= \frac{(v-v_w)+pz_{tcg}-rx_{tcg}-v_{i_t}}{\Omega_t R_t}, & \mathbf{w}_t &= \frac{(w-w_w)+qx_{tcg}-\varkappa v_i}{\Omega_t R_t} \\ \mu_{p_t} &= \frac{p}{\Omega_t}, & \mu_{q_t} &= \frac{(q+\Omega_t)}{\Omega_t}, & \mu_{r_t} &= \frac{r}{\Omega_t} \\ X'_{t_{cg}} &= \frac{x_{tcg}}{R_t}, & Y'_{t_{cg}} &= \frac{y_{tcg}}{R_t}, & Z'_{t_{cg}} &= \frac{z_{tcg}}{R_t} \\ \kappa_{t_1} &= \left(\frac{R_{t_2}^4}{R_t^4} - \frac{R_{t_h}^4}{R_t^4} \right), & \kappa_{t_2} &= \left(\frac{R_{t_2}^3}{R_t^3} - \frac{R_{t_h}^3}{R_t^3} \right), & \kappa_{t_3} &= \left(\frac{R_{t_2}^2}{R_t^2} - \frac{R_{t_h}^2}{R_t^2} \right) \\ \kappa_{t_4} &= \left(\frac{R_{t_2}}{R_t} - \frac{R_{t_h}}{R_t} \right) \end{aligned} \quad (2.88)$$

The tail thrust is obtained by multiplying the tail thrust coefficient by the term $\pi \rho R_t^4 \Omega_t^2$ and substituting the terms defined in equation (2.88) to yield

$$T_t = T_{t_o} + T_{t_\theta} \delta'_\theta \quad (2.89)$$

where

$$T_{t_o} = \frac{\pi \rho R_t^4 \Omega_t^2 S_t C_{L\alpha_t} \kappa_{t_3}}{8} [\mu_{p_t} \mathbf{u}_t - 2\mu_{q_t} \mathbf{v}_t + \mu_{r_t} \mathbf{w}_t] \quad (2.90)$$

and

$$T_{t_\theta} = \frac{\pi \rho R_t^4 \Omega_t^2 S_t C_{L\alpha_t}}{4} \left\{ \frac{2}{3} \kappa_{t_2} \mu_{q_t}^2 + \kappa_{t_4} \left[\left(\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}} \right)^2 + \left(\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}} \right)^2 \right] \right\} \quad (2.91)$$

and $\delta'_\theta = [\delta_{\theta_t}]^T$.

Equation (2.89) represents the affine-mode form for the tail thrust.

2.4.6 Tail Drag Torque

Similar to the main rotor drag torque, the tail drag torque, see figure 2.5, is given by

$$dQ_t = y dH_t = y (dD \cos \eta + dl \sin \eta) \quad (2.92)$$

After substitute equations (2.35) and (2.36) into equation (2.92), and recalling the assumption of the small angle η , the drag force results in

$$dQ_t = \frac{1}{2} \rho c_t \left[C_{D_{ot}} U_{t_t}^2 + \frac{C_{L_{\alpha t}}^2 (\delta_{\theta t} U_{t_t} - U_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha t}} (\delta_{\theta t} U_{n_t} U_{t_t} - U_{n_t}^2) \right] y dy \quad (2.93)$$

Defining the dimensionless drag torque coefficient as

$$dC_{Q_t} = \frac{dQ_t}{\rho A_t R_t (\Omega_t R_t)^2} = \frac{1}{2} \rho c_t \left[C_{D_{ot}} \left(\frac{U_{t_t}}{\Omega_t R_t} \right)^2 + \frac{C_{L_{\alpha t}}^2 \left(\delta_{\theta t} \frac{U_{t_t}}{\Omega_t R_t} - \frac{U_{n_t}}{\Omega_t R_t} \right)^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha t}} \left(\delta_{\theta t} \frac{U_{n_t}}{\Omega_t R_t} \frac{U_{t_t}}{\Omega_t R_t} - \left(\frac{U_{n_t}}{\Omega_t R_t} \right)^2 \right) \right] y dy$$

Then the effect of the second blade, using the definition of solidity factor \mathcal{S} and dimensionless velocities $u_{n_t} = \frac{U_{n_t}}{\Omega_t R_t}$ and $u_{t_t} = \frac{U_{t_t}}{\Omega_t R_t}$, and introducing the variable change $\tau = \frac{y}{R_t}$ and $d\tau = \frac{dy}{R_t}$ results in

$$dC_{Q_t} = \frac{\mathcal{S}_t}{2} \left[C_{D_{ot}} u_{t_t}^2 + \frac{C_{L_{\alpha t}}^2 (\delta_{\theta t} u_{t_t} - u_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha t}} (\delta_{\theta t} u_{n_t} u_{t_t} - u_{n_t}^2) \right] \tau d\tau \quad (2.94)$$

Torque coefficient is computed by integrating equation (2.94) over one complete revolution and all along the blade span as follows:

$$C_{Q_t} = \frac{\mathcal{S}_t}{2} \int_{\frac{R_{t_h}}{R_t}}^{\frac{R_{t_2}}{R_t}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{D_{ot}} u_{t_t}^2 + \frac{C_{L_{\alpha t}}^2 (\delta_{\theta t} u_{t_t} - u_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha t}} (\delta_{\theta t} u_{n_t} u_{t_t} - u_{n_t}^2) \right] d\Psi \tau d\tau$$

Drag torque can be obtained by multiplying the drag torque coefficient by the term $\rho \pi R_t^5 \Omega_t^2$. Therefore, substituting the tangential and normal air velocities given by equations (2.86) and (2.87), using the terms defined by equation (2.88), and after solving the integrals the tail rotor drag torque becomes:

$$Q_t = Q_{t_o} + Q_{t_\theta} \delta'_\theta \quad (2.95)$$

where

$$Q_{t_o} = \frac{\rho R_t^5 \Omega_t^2 \mathcal{S}_t C_{L\alpha_t}}{4e_t \mathcal{R}_t} \left\{ (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \left[\frac{1}{4} \kappa_{t1} (\mu_{rt}^2 + \mu_{pt}^2) - \kappa_{t3} \mathbf{v}_t^2 \right] + \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t}}{2C_{L\alpha_t}} \left[\kappa_{t1} \mu_{qt}^2 + \kappa_{t3} \left\{ (\mathbf{u}_t + \mu_{rt} Y'_{t_{cg}})^2 + (\mathbf{w}_t - \mu_{pt} Y'_{t_{cg}})^2 \right\} \right] \right\} \quad (2.96)$$

and

$$Q_{t_\theta} = \frac{\rho R_t^5 \Omega_t^2 \mathcal{S}_t C_{L\alpha_t}}{e_t \mathcal{R}_t} \left[\frac{C_{L\alpha_t}}{8} \left\{ \kappa_{t1} \mu_{qt}^2 + \kappa_{t3} \left[(\mathbf{u}_t + \mu_{rt} Y'_{t_{cg}})^2 + (\mu_{pt} Y'_{t_{cg}} - \mathbf{w}_t)^2 \right] \right\} \right. \\ \left. \frac{\kappa_{t2}}{6} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \left(\mu_{qt} \mathbf{v}_t - \frac{\mu_{rt} \mathbf{w}_t + \mu_{pt} \mathbf{u}_t}{2} \right) \right]^T \quad (2.97)$$

and the augmented tail command vector δ'_θ is

$$\delta'_\theta = [\delta_{\theta_t}^2, \delta_{\theta_t}]^T$$

Equations (2.95) to (2.97) represents the affine-mode form of the tail thrust.

2.4.7 Tail In-plane Force

The in-plane force of the tail rotor can be computed in similar manner as the main rotor one. As shown in figure 2.7 the components of the tail in-plane force can be written as:

$$H_{t_x} = (dD_t \cos \eta_t + dL_t \sin \eta_t) \sin(-\Psi_t) = - (dD_t \cos \eta_t + dL_t \sin \eta_t) \sin \Psi_t \quad (2.98)$$

and

$$H_{t_z} = (dD_t \cos \eta_t + dL_t \sin \eta_t) \cos(-\Psi_t) = (dD_t \cos \eta_t + dL_t \sin \eta_t) \cos \Psi_t \quad (2.99)$$

After substitute equations (2.35) and (2.36) into equations (2.98) and (2.99), and recalling the small angle assumption of angle η_t , results in

$$H_{t_x} = -\frac{1}{2} \rho U_{t_t}^2 c_t dy \left[C_{D\alpha_t} + \frac{C_{L\alpha_t}^2 \left(\delta_{\theta_t} - \frac{U_{n_t}}{U_{t_t}} \right)^2}{\pi e_t \mathcal{R}_t} + C_{L\alpha_t} \left(\delta_{\theta_t} - \frac{U_{n_t}}{U_{t_t}} \right) \frac{U_{n_t}}{U_{t_t}} \right] \sin \Psi_t$$

$$H_{tz} = \frac{1}{2} \rho U_{t_t}^2 c_t dy \left[C_{D_{o_t}} + \frac{C_{L_{\alpha_t}}^2 \left(\delta_{\theta_t} - \frac{U_{n_t}}{U_{t_t}} \right)^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha_t}} \left(\delta_{\theta_t} - \frac{U_{n_t}}{U_{t_t}} \right) \frac{U_{n_t}}{U_{t_t}} \right] \cos \Psi_t$$

Defining the dimensionless differential in-plane force coefficients as

$$dC_{H_{tx}} = \frac{dH_{tx}}{\pi R_t^2 \rho (\Omega_t R_t)^2}$$

$$dC_{H_{tz}} = \frac{dH_{tz}}{\pi R_t^2 \rho (\Omega_t R_t)^2}$$

After substituting the in-plane components, and adding the contribution of the second blade, the coefficients become

$$dC_{H_{tx}} = -\frac{S_t}{2} \left[C_{D_{o_t}} u_{t_t}^2 + \frac{C_{L_{\alpha_t}}^2 (\delta_{\theta_t} u_{t_t} - u_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha_t}} (\delta_{\theta_t} u_{n_t} u_{t_t} - u_{n_t}^2) \right] \sin \Psi_t \frac{dy}{R_t}$$

$$dC_{H_{tz}} = \frac{S_t}{2} \left[C_{D_{o_t}} u_{t_t}^2 + \frac{C_{L_{\alpha_t}}^2 (\delta_{\theta_t} u_{t_t} - u_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha_t}} (\delta_{\theta_t} u_{n_t} u_{t_t} - u_{n_t}^2) \right] \cos \Psi_t \frac{dy}{R_t}$$

Coefficients $dC_{H_{tx}}$ and $dC_{H_{tz}}$ need to be integrated over one complete revolution and along the tail blade span. Introducing the variable change $\tau = \frac{dy}{R_t}$ the integrals become

$$C_{H_{tx}} = -\frac{S_t}{2} \int_{\frac{R_{t_h}}{R_t}}^{\frac{R_{t_2}}{R_t}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{D_{o_t}} u_{t_t}^2 + \frac{C_{L_{\alpha_t}}^2 (\delta_{\theta_t} u_{t_t} - u_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha_t}} (\delta_{\theta_t} u_{n_t} u_{t_t} - u_{n_t}^2) \right] \sin \Psi_t d\Psi_t d\tau$$

and

$$C_{H_{tz}} = \frac{S_t}{2} \int_{\frac{R_{t_h}}{R_t}}^{\frac{R_{t_2}}{R_t}} \frac{1}{2\pi} \int_0^{2\pi} \left[C_{D_{o_t}} u_{t_t}^2 + \frac{C_{L_{\alpha_t}}^2 (\delta_{\theta_t} u_{t_t} - u_{n_t})^2}{\pi e_t \mathcal{R}_t} + C_{L_{\alpha_t}} (\delta_{\theta_t} u_{n_t} u_{t_t} - u_{n_t}^2) \right] \cos \Psi_t d\Psi_t d\tau$$

The in-plane force components are obtained by multiplying the corresponding in-plane coefficients by the term $\pi \rho R_t^4 \Omega_t^2$. Substituting the values of the tangential and normal velocities for the tail rotor given by equations (2.86) and (2.87), and using the terms definitions expressed by equation (2.88) and doing the corresponding integrals, the components of the tail in-plane force become:

$$H_{tx} = H_{tx_o} + H_{tx_\theta} \delta'_\theta \quad (2.100)$$

and

$$H_{tz} = H_{tz_o} + H_{tz_\theta} \delta'_\theta \quad (2.101)$$

where

$$H_{t_{x_o}} = \frac{\rho R_t^4 \Omega_t^2 \mathcal{S}_t C_{L_{\alpha_t}} \kappa_{t3}}{4e_t \mathcal{R}_t} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left[\mu_{p_t} \mathbf{v}_t + \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}}}{C_{L_{\alpha_t}}} \mu_{q_t} (\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}}) \right] \quad (2.102)$$

$$H_{t_{x_\theta}} = \frac{\rho R_t^4 \Omega_t^2 \mathcal{S}_t C_{L_{\alpha_t}}}{e_t \mathcal{R}_t} \left[\begin{array}{c} \frac{C_{L_{\alpha_t}} \kappa_{t3}}{4} (\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}}) \mu_{q_t} \\ \frac{1}{4} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left[\kappa_{t4} (\mathbf{u}_t - \mu_{r_t} Y'_{t_{cg}}) \mathbf{v}_t - \frac{\kappa_{t2} \mu_{p_t} \mu_{q_t}}{3} \right] \end{array} \right]^T \quad (2.103)$$

and

$$H_{t_{z_o}} = -\frac{\rho R_t^4 \Omega_t^2 \mathcal{S}_t C_{L_{\alpha_t}} \kappa_{t3}}{4e_t \mathcal{R}_t} \left[(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \mu_{r_t} \mathbf{v}_t - \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}}}{C_{L_{\alpha_t}}} \mu_{q_t} (\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}}) \right] \quad (2.104)$$

$$H_{t_{z_\theta}} = \frac{\rho R_t^4 \Omega_t^2 \mathcal{S}_t C_{L_{\alpha_t}}}{e_t \mathcal{R}_t} \left[\begin{array}{c} -\frac{C_{L_{\alpha_t}} \kappa_{t3}}{4} (\mathbf{w}_t + \mu_{p_t} Y'_{t_{cg}}) \mu_{q_t} \\ \frac{1}{4} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left[\frac{\kappa_{t2} \mu_{r_t} \mu_{q_t}}{3} - \kappa_{t4} (\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}}) \mathbf{v}_t \right] \end{array} \right]^T \quad (2.105)$$

and the augmented tail command vector δ'_{θ_t} is

$$\delta_\theta = [\delta_{\theta_t}^2, \delta_{\theta_t}]^T$$

Equations (2.100) to (2.105) are the tail in-plane forces in affine-form [22].

2.4.8 Aerodynamic Drag Force

The performance of any aircraft is strongly related to the drag forces. Parameters like maximum speed, and range are determined by the shape and size of the surfaces and structure of the aircraft. It is important to estimate accurately all drag sources and classify them as shown in figure 2.11. In the case of the Air Star Evolution helicopter the main sources of both induced drag D_i and Zero-Lift drag D_0 are produced by the main and tail rotor as are described in sections 2.4.2 and 2.4.6. There are other sources of Zero-Lift drag force such as the fuselage, landing gear, tail boom, and vertical fin.

The canopy of the helicopter together with the avionic box are considered

as the fuselage of the Air Star Evolution helicopter. Although the tail boom is part of the fuselage in a full scale helicopter, in the case of the Air Star Evolution is not considered as a part of the fuselage due to its small projected area (about 0.03 m²) and it will be considered separately.

The Zero-Lift drag force of the fuselage at hover condition or at forward velocity lower than the hover induced velocity v_{hi} can be computed as [5]

$$F_{D_{fu}} = \frac{1}{2} \rho v_{hi}^2 \begin{bmatrix} S_{x_{fu}} \frac{1}{v_{hi}} & 0 \\ 0 & S_{y_{fu}} \frac{1}{v_{hi}} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \quad (2.106)$$

and for forward velocities greater than v_{hi} [5]

$$F_{D_{fu}} = \frac{1}{2} \rho V_{\infty} \begin{bmatrix} S_{x_{fu}} & 0 & 0 \\ 0 & S_{y_{fu}} & 0 \\ 0 & 0 & S_{z_{fu}} \end{bmatrix} \begin{bmatrix} u - u_w \\ v - v_w \\ w - w_w + v_{hi} \end{bmatrix} \quad (2.107)$$

where $S_{x_{fu}}$, $S_{y_{fu}}$, and $S_{z_{fu}}$ are the projected or effective drag areas of the fuselage, and $V_{\infty} = \sqrt{(u - u_w)^2 + (v - v_w)^2 + (w - w_w + v_{hi})^2}$

The tail boom, landing gear and vertical fin are considered as circular cylinders under a stream flow.

For example, the vertical fin of the Air Star Evolution helicopter is made of aluminium rod of 9.5 mm of diameter and a length of 48 cm as shown in figure 2.12. It has a total projected area S_{vf} of 4.56×10^{-3} m². The area fraction of the vertical fin under direct effect of the tail rotor induced velocity v_{it} is approximately 60.2%. Mainly, the drag force exerted on the vertical fin is in the direction of the body axis y_b and is given by

$$F_{D_{vf}} = \frac{1}{2} \rho S_{vf} V_{vf}^2 C_{D_{vf}} \quad (2.108)$$

where S_{vf} is the projected area of the vertical fin, and V_{vf} is the relative air velocity in the direction of the body axis y_b . Basically, the drag coefficient $C_{D_{vf}}$ is the summation of the skin drag force coefficient and the form or pressure drag coefficient

$$C_{D_{vf}} = C_{D_{0,vf}} = C_{D_{f,vf}} + C_{D_{p,vf}}$$

where $C_{D_{f,vf}}$ is the skin friction drag coefficient of the vertical fin and $C_{D_{p,vf}}$ is

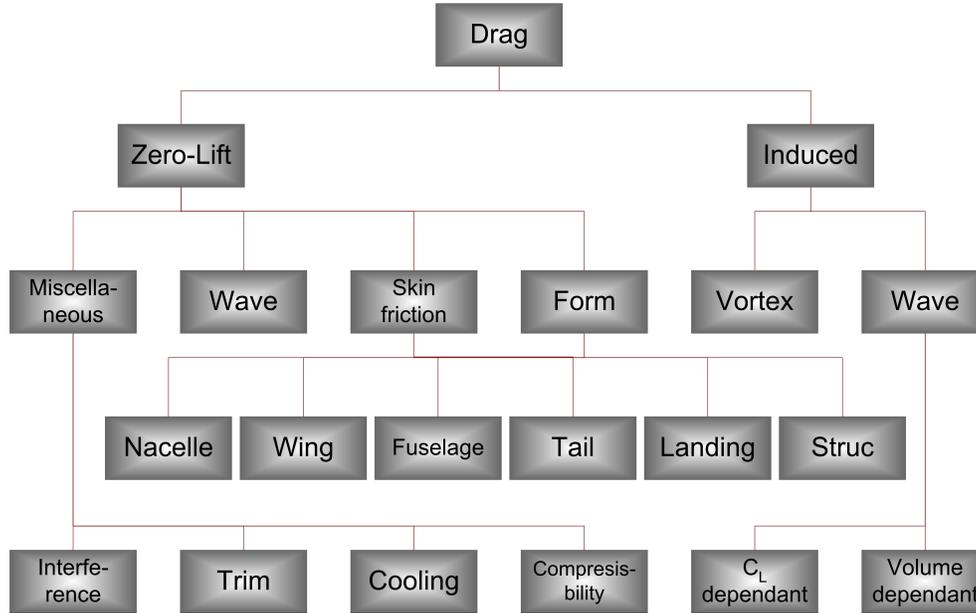


Figure 2.11: Drag classification [37].

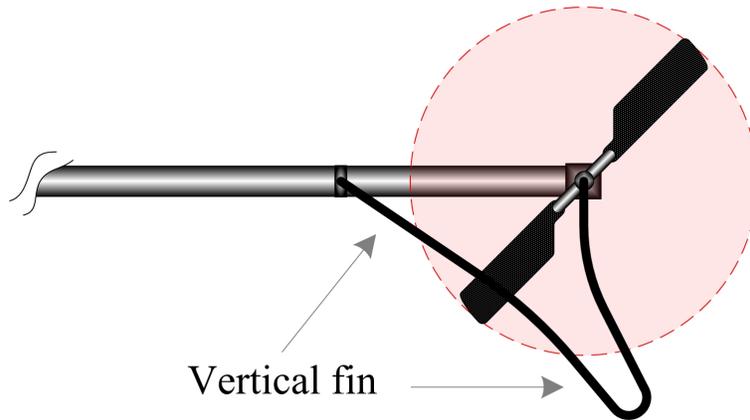


Figure 2.12: Vertical Fin Shape.

the corresponding form drag coefficient. By simulating a cylinder of diameter d under a steady air stream for different velocities, is possible to estimate the vertical fin drag coefficient value [38] – [47]. The Reynolds number $\mathcal{R}_e = \frac{\rho V_{vf} d}{\mu}$ for side velocities from 0.1 to 4 m/s, an air dynamic viscosity μ of 1.6×10^{-5} kg/(m · s) and an air density of 1.1073 kg/m³ are 65.75 and 2629.83 respectively. Then the resulting values of $C_{D_{f,vf}}$ as a function of the Reynolds number are shown in figure 2.13 [38] – [47]: For the case of the vertical fin, the side velocity V_{vf} in equation (2.108) is the summation of the free air stream

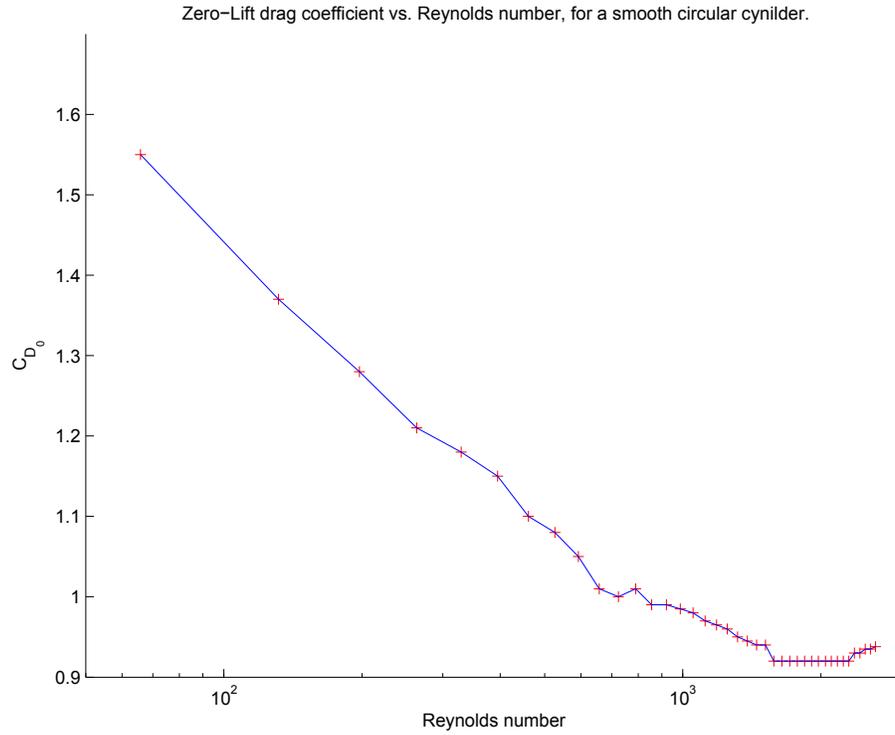


Figure 2.13: Zero-Lift Drag Coefficient for a circular cylinder of diameter d .

velocity plus the total side velocity [5]

$$V_{vf} = (V_{\infty}^2 + v_{vf_1}^2) + (V_{\infty}^2 + v_{vf_2}^2)$$

where

$$V_{\infty} = \sqrt{(u - u_w)^2 + [(w - w_w) + x_{tcg}q - \mathcal{X}v_i]^2}$$

and

$$v_{vf_1} = (v - v_w) - v_{it} - x_{tcg}r$$

$$v_{vf_2} = (v - v_w) - x_{tcg}r$$

Hence the side drag force for the vertical fin is

$$F_{D_{vf}} = \frac{1}{2} \rho S_{vf} \left[0.602 (V_{\infty}^2 + v_{vf_1}^2) C_{D_{vf_1}} + (1 - 0.602) (V_{\infty}^2 + v_{vf_2}^2) C_{D_{vf_2}} \right] \quad (2.109)$$

By using the values of figure 2.13 in equation (2.109) the side drag force for the vertical fin is computed.

Following the same procedure, the side drag forces for the tail boom and landing gear are computed.

2.5 Complete Helicopter Model

Based on the equations, described in sections 2.2 to 2.4, the total external force and moment can be written in affine-mode as follows:

$$\mathbf{F}_T = \mathbf{F}_{T_o} + \mathbf{W} + \mathbf{F}_D + \mathbf{F}_{T_\theta} \delta_\theta \quad (2.110)$$

and

$$\mathbf{M}_{cg} = \mathbf{M}_{cg_o} + \mathbf{M}_{f_o} + (\mathbf{M}_{f_\theta} + \mathbf{M}_{cg_\theta}) \delta_\theta \quad (2.111)$$

where \mathbf{M}_{f_o} is the fly bar component independent of the cyclic commands and \mathbf{M}_{f_θ} is the fly bar component dependant of the cyclic commands. Thus the total augmented vector δ_θ is:

$$\delta_\theta = [\delta_{\theta_{col}}^2, \delta_{\theta_{lat}}^2, \delta_{\theta_{lon}}^2, \delta_{\theta_{col}} \delta_{\theta_{lat}}, \delta_{\theta_{col}} \delta_{\theta_{lon}}, \delta_{\theta_{lat}} \delta_{\theta_{lon}}, \delta_{\theta_{col}}, \delta_{\theta_{lat}}, \delta_{\theta_{lon}}, \delta_{\theta_t}^2, \delta_{\theta_t}]^T$$

The components of \mathbf{F}_{T_o} and \mathbf{F}_{T_θ} are defined as:

$$\mathbf{F}_{T_o} = \begin{bmatrix} F_{x_o} \\ F_{y_o} \\ F_{z_o} \end{bmatrix}_b = \begin{bmatrix} H_{r_{x_o}} + H_{t_{x_o}} \\ T_{t_o} + H_{r_{y_o}} \\ -T_{r_o} + H_{t_{z_o}} \end{bmatrix}_b \quad (2.112)$$

and

$$\mathbf{F}_{T_\theta} \delta_\theta = \begin{bmatrix} F_{x_\theta} \\ F_{y_\theta} \\ F_{z_\theta} \end{bmatrix}_b \delta_\theta = \begin{bmatrix} \mathbf{a}_{1,1} & \mathbf{a}_{1,2} & \mathbf{a}_{1,3} & \mathbf{a}_{1,4} & \mathbf{a}_{1,5} & \mathbf{a}_{1,6} & \mathbf{a}_{1,7} & \mathbf{a}_{1,8} & \mathbf{a}_{1,9} & \mathbf{a}_{1,10} & \mathbf{a}_{1,11} \\ \mathbf{a}_{2,1} & \mathbf{a}_{2,2} & \mathbf{a}_{2,3} & \mathbf{a}_{2,4} & \mathbf{a}_{2,5} & \mathbf{a}_{2,6} & \mathbf{a}_{2,7} & \mathbf{a}_{2,8} & \mathbf{a}_{2,9} & \mathbf{a}_{2,10} & \mathbf{a}_{2,11} \\ \mathbf{a}_{3,1} & \mathbf{a}_{3,2} & \mathbf{a}_{3,3} & \mathbf{a}_{3,4} & \mathbf{a}_{3,5} & \mathbf{a}_{3,6} & \mathbf{a}_{3,7} & \mathbf{a}_{3,8} & \mathbf{a}_{3,9} & \mathbf{a}_{3,10} & \mathbf{a}_{3,11} \end{bmatrix}_b \begin{bmatrix} \delta_{\theta_{col}}^2 \\ \delta_{\theta_{lat}}^2 \\ \delta_{\theta_{lon}}^2 \\ \delta_{\theta_{col}} \delta_{\theta_{lat}} \\ \delta_{\theta_{col}} \delta_{\theta_{lon}} \\ \delta_{\theta_{lat}} \delta_{\theta_{lon}} \\ \delta_{\theta_{col}} \\ \delta_{\theta_{lat}} \\ \delta_{\theta_{lon}} \\ \delta_{\theta_t}^2 \\ \delta_{\theta_t} \end{bmatrix} \quad (2.113)$$

Define the components of \mathbf{M}_{cg_o} and \mathbf{M}_{cg_θ} as:

$$\mathbf{M}_{cg_o} = \begin{bmatrix} \mathcal{M}_{\phi_o} \\ \mathcal{N}_{\theta_o} \\ \mathcal{L}_{\psi_o} \end{bmatrix}_b = \begin{bmatrix} M_{\phi_o} + z_{tcg} T_{t_o} + z_{rcg} H_{r_{y_o}} - y_{tcg} H_{t_{z_o}} \\ M_{\theta_o} - z_{rcg} H_{r_{x_o}} + x_{tcg} H_{t_{z_o}} - z_{tcg} H_{t_{x_o}} - Q_{t_o} \\ Q_{r_o} - x_{tcg} T_{t_o} + y_{tcg} H_{t_{x_o}} \end{bmatrix}_b \quad (2.114)$$

and

$$\mathbf{M}_{cg\theta} \delta_\theta = \begin{bmatrix} \mathcal{M}_{\phi_\theta} \\ \mathcal{N}_{\theta_\theta} \\ \mathcal{L}_{\psi_\theta} \end{bmatrix}_b \delta_\theta = \begin{bmatrix} \delta_{\theta_{col}}^2 \\ \delta_{\theta_{lat}}^2 \\ \delta_{\theta_{lon}}^2 \\ \delta_{\theta_{col}} \delta_{\theta_{lat}} \\ \delta_{\theta_{col}} \delta_{\theta_{lon}} \\ \delta_{\theta_{lat}} \delta_{\theta_{lon}} \\ \delta_{\theta_{col}} \\ \delta_{\theta_{lat}} \\ \delta_{\theta_{lon}} \\ \delta_{\theta_t}^2 \\ \delta_{\theta_t} \end{bmatrix} \quad (2.115)$$

$$\begin{bmatrix} \mathbf{b}_{1,1} & \mathbf{b}_{1,2} & \mathbf{b}_{1,3} & \mathbf{b}_{1,4} & \mathbf{b}_{1,5} & \mathbf{b}_{1,6} & \mathbf{b}_{1,7} & \mathbf{b}_{1,8} & \mathbf{b}_{1,9} & \mathbf{b}_{1,10} & \mathbf{b}_{1,11} \\ \mathbf{b}_{2,1} & \mathbf{b}_{2,2} & \mathbf{b}_{2,3} & \mathbf{b}_{2,4} & \mathbf{b}_{2,5} & \mathbf{b}_{2,6} & \mathbf{b}_{2,7} & \mathbf{b}_{2,8} & \mathbf{b}_{2,9} & \mathbf{b}_{2,10} & \mathbf{b}_{2,11} \\ \mathbf{b}_{3,1} & \mathbf{b}_{3,2} & \mathbf{b}_{3,3} & \mathbf{b}_{3,4} & \mathbf{b}_{3,5} & \mathbf{b}_{3,6} & \mathbf{b}_{3,7} & \mathbf{b}_{3,8} & \mathbf{b}_{3,9} & \mathbf{b}_{3,10} & \mathbf{b}_{3,11} \end{bmatrix}_b$$

where the coefficients $\mathbf{a}_{m,n}$ and $\mathbf{b}_{m,n}$ in equations (2.113) and (2.115) are the summation of the corresponding terms of forces and moments dependant of the cyclic command given in sections 2.2 to 2.4 and are defined in appendix A section A.3.

Using the definitions of equations (2.112) to (2.115), equations (2.110) and (2.111) are rewritten in component in form:

$$\mathbf{F}_T = \begin{bmatrix} F_{x_o} \\ F_{y_o} \\ F_{z_o} \end{bmatrix}_b + \mathbf{C}_\nu^b \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}_\nu + \begin{bmatrix} F_{x_D} \\ F_{y_D} \\ F_{z_D} \end{bmatrix}_b + \begin{bmatrix} F_{x_\theta} \\ F_{y_\theta} \\ F_{z_\theta} \end{bmatrix}_b \quad (2.116)$$

and

$$\mathbf{M}_{cg} = \begin{bmatrix} \mathcal{M}_{\phi_o} \\ \mathcal{N}_{\theta_o} \\ \mathcal{L}_{\psi_o} \end{bmatrix}_b + \begin{bmatrix} \alpha_2 \dot{\theta} \\ -\alpha_4 \dot{\phi} \\ 0 \end{bmatrix}_b + \begin{bmatrix} -\alpha_1 \delta_{\theta_{lon}} \\ \alpha_3 \delta_{\theta_{lat}} \\ 0 \end{bmatrix}_b + \begin{bmatrix} \mathcal{M}_{\phi_\theta} \\ \mathcal{N}_{\theta_\theta} \\ \mathcal{L}_{\psi_\theta} \end{bmatrix}_b \quad (2.117)$$

Now substituting equation (2.116) into equation (2.22) and writing in terms of components results in

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m_T} \left(\begin{bmatrix} F_{x_o} \\ F_{x_o} \\ F_{x_o} \end{bmatrix}_b + \mathbf{C}_\nu^b \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}_\nu + \begin{bmatrix} F_{x_D} \\ F_{y_D} \\ F_{z_D} \end{bmatrix}_b + \begin{bmatrix} F_{x_\theta} \\ F_{y_\theta} \\ F_{z_\theta} \end{bmatrix}_b \right) - \begin{bmatrix} wq - vr \\ ur - wp \\ vp - uq \end{bmatrix}_b \quad (2.118)$$

Similarly, substituting equation (2.117) into equation (2.33) assuming $\dot{\Omega} \approx 0$, the moment state equations are written as:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{\mathcal{M} - 2q \left\{ -I_{z_{bl}} \Omega + r \left[(I_z + I_{z_{bl}}) - \left(I_y + \frac{I_{x_{bl}} + I_{y_{bl}}}{2} \right) \right] \right\}}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \\ \frac{\mathcal{N} - 2p \left\{ I_{z_{bl}} \Omega + r \left[\left(I_x + \frac{I_{x_{bl}} + I_{y_{bl}}}{2} \right) - (I_z + I_{z_{bl}}) \right] \right\}}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \\ \frac{\mathcal{L} + qp(I_x - I_y)}{I_z + I_{z_{bl}}} \end{bmatrix}_b \quad (2.119)$$

where \mathcal{M} , \mathcal{N} and \mathcal{L} are defined as

$$\begin{aligned} \mathcal{M} &= \mathcal{M}_{\phi_o} + \alpha_2 \dot{\theta} - \alpha_1 \delta_{\theta_{lon}} + \mathcal{M}_{\phi_\theta} \\ \mathcal{N} &= \mathcal{N}_{\theta_o} - \alpha_4 \dot{\phi} + \alpha_3 \delta_{\theta_{lat}} + \mathcal{N}_{\theta_\theta} \\ \mathcal{L} &= \mathcal{L}_{\psi_o} + \mathcal{L}_{\psi_\theta} \end{aligned}$$

and $\dot{\phi}$ and $\dot{\theta}$ from (2.23) are:

$$\dot{\phi} = p + (q \sin \phi + r \cos \phi) \tan \theta$$

and

$$\dot{\theta} = q \cos \phi - r \sin \phi$$

Equations (2.118) and (2.119) together with equations (2.23) and (2.24) represent the non linear dynamic model [22], and are rewritten in section 2.6

for a quick reference.

Table 2.1 summarizes the helicopter's parameters needed for the model. Most of the values given in table 2.1 are measured but some, like main blades rotor mass moment of inertia, are estimated by computation by approximation of the geometry to a rectangular shape.

Table 2.1: Air Star Evolution Helicopter's Parameters

Parameter	Value	Units	Description
m	10.8	kg	Helicopter's mass, excluding main rotor blades
m_r	0.318	kg	Main rotor blade mass (each blade)
$z_{r_{cg}}$	-0.31	m	Distance from cg to main rotor hub in z_b axis
I_x	0.5140	$\text{kg} \cdot \text{m}^2$	Helicopter's mass moment of inertia about x_b axis
I_y	1.5944	$\text{kg} \cdot \text{m}^2$	Helicopter's mass moment of inertia about y_b axis
I_z	1.2741	$\text{kg} \cdot \text{m}^2$	Helicopter's mass moment of inertia about z_b axis
$I_{x_{bl}}$	0.0002	$\text{kg} \cdot \text{m}^2$	Main blades' mass moment of inertia about x_{bl} axis
$I_{y_{bl}}$	0.0957	$\text{kg} \cdot \text{m}^2$	Main blades' mass moment of inertia about y_{bl} axis
$I_{z_{bl}}$	0.0964	$\text{kg} \cdot \text{m}^2$	Main blades' mass moment of inertia about z_{bl} axis
$C_{L\alpha}$	6.4	$\frac{1}{\text{rad}}$	Main rotor lift slope
C_{D_o}	0.012		Main rotor zero lift drag coefficient
Q_{r_h}	5.9	$\text{N} \cdot \text{m}$	Main rotor drag torque in hover
\mathcal{R}	11.31		Main blade Aspect Ratio
v_{h_i}	3.86	$\frac{\text{m}}{\text{s}}$	Main rotor induced velocity in hover
c	0.084	m	Main cord of the main rotor blades
b	0.95	m	Main rotor blade span
e	0.723		Main blade Oswald number
\mathcal{S}	0.0514		Main rotor Solidity factor
Ω	115.2	$\frac{\text{rad}}{\text{s}}$	Main rotor angular velocity
R	1.04	m	Main rotor disc
R_h	0.09	m	Main rotor hinge offset
$S_{x_{fu}}$	0.1019	m^2	Frontal fuselage area
$S_{y_{fu}}$	0.8256	m^2	Lateral fuselage area
$S_{z_{fu}}$	0.5057	m^2	Vertical fuselage area
S_{vf}	0.0456	m^2	Vertical fin area
$C_{D_{vf1}}$	0.95		Zero-lift Vertical fin drag coefficient in hover
$C_{D_{vf2}}$	0.0		Zero-lift Vertical fin drag coefficient in hover
$x_{t_{cg}}$	-1.21	m	Distance from cg to tail rotor hub in x_b axis
$y_{t_{cg}}$	-0.06	m	Distance from cg to tail rotor hub in y_b axis
$z_{t_{cg}}$	-0.09	m	Distance from cg to tail rotor hub in z_b axis
$C_{L\alpha_t}$	5.0	$\frac{1}{\text{rad}}$	Tail rotor lift slope
$C_{D_{o_t}}$	0.01		Tail rotor zero lift drag coefficient
\mathcal{R}_t	3.7		Tail blade Aspect Ratio
$v_{i_{t_h}}$	5.30	$\frac{\text{m}}{\text{s}}$	Tail rotor induced velocity in hover
c_t	0.033	m	Main cord of the tail rotor blades
b_t	0.122	m	Tail rotor blade span
e_t	0.945		Tail blade Oswald number
\mathcal{S}_t	0.133		Tail rotor Solidity factor
Ω_t	691.2	$\frac{\text{rad}}{\text{s}}$	Tail rotor angular velocity
R_t	0.158	m	Tail rotor disc
R_{t_h}	0.036	m	Tail rotor hinge offset
α_1	1	$\text{N} \cdot \text{m}$	
α_2	0.45	$\text{N} \cdot \text{m} \cdot \text{s}$	
α_3	1	$\text{N} \cdot \text{m}$	
α_4	0.45	$\text{N} \cdot \text{m} \cdot \text{s}$	
\varkappa	0.35		z_b axis wind contribution factor due to the main rotor induced velocity
ρ	1.107	$\frac{\text{kg}}{\text{m}^3}$	Air density
g	9.81	$\frac{\text{m}}{\text{s}^2}$	Gravity acceleration

2.6 Helicopter Model Summary

In this section a summary of the motion equations of the Air Star Evolution helicopter is presented.

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \frac{1}{m_T} \left(\begin{bmatrix} F_{x_o} \\ F_{x_o} \\ F_{x_o} \end{bmatrix}_b + \mathbf{C}_\nu^b \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix}_\nu + \begin{bmatrix} F_{x_D} \\ F_{y_D} \\ F_{z_D} \end{bmatrix}_b + \begin{bmatrix} F_{x_\theta} \\ F_{y_\theta} \\ F_{z_\theta} \end{bmatrix}_b \right) - \begin{bmatrix} wq - vr \\ ur - wp \\ vp - uq \end{bmatrix}_b$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{\mathcal{M} - 2q \left\{ -I_{z_{bl}} \Omega + r \left[(I_z + I_{z_{bl}}) - \left(I_y + \frac{I_{x_{bl}} + I_{y_{bl}}}{2} \right) \right] \right\}}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \\ \frac{\mathcal{N} - 2p \left\{ I_{z_{bl}} \Omega + r \left[\left(I_x + \frac{I_{x_{bl}} + I_{y_{bl}}}{2} \right) - (I_z + I_{z_{bl}}) \right] \right\}}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \\ \frac{\mathcal{L} + qp(I_x - I_y)}{I_z + I_{z_{bl}}} \end{bmatrix}_b$$

$$\dot{\boldsymbol{\varphi}} = \mathbf{C}_\omega^{\dot{\boldsymbol{\varphi}}} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

$$\dot{\boldsymbol{\chi}} = \mathbf{C}_b^i \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

Chapter 3

MOI Testbed and HIL Force Testbed

In this chapter the mass moment of inertia tensor of the Air Star Evolution helicopter is experimentally estimated. Although this testbed has been specifically designed to estimate the mass moment of inertia of this helicopter the procedure is a general one that could be adapted to other helicopters. The mass moments of inertia are found using the **Moment Of Inertia (MOI)** testbed designed for this purpose. To get the helicopter properties a mathematical model of the testbed is needed. Then the combined system of testbed and helicopter is used to get the helicopter properties. In section 3.1.1, the mathematical model of the MOI testbed is derived, and in section 3.1.2, the roll, pitch and yaw mass moment about x_b , y_b and z_b axes in the body frame \mathbf{C}_b are found.

In section 3.2.1, the equations of the Hardware-In-the-loop (HIL) force testbed used to estimate the lift slope and zero-lift drag coefficient C_{L_α} and C_{D_o} are derived. In section 3.2.3 the kinematics of the lower swashplate mechanism is derived and finally, in section 3.2.4, the lift slope C_{L_α} and the zero-lift drag coefficient C_{D_o} of the main rotor are estimated using the HIL force testbed designed for that purpose.

3.1 MOI Testbed

The general layout of the MOI testbed is shown in figure 3.1. It consist of a flat square aluminium bed, two pulleys, one horizontal and one vertical, a 2000 division incremental angular position encoder and a [class 6](#) control gauge hanging mass. The helicopter is attached over the aluminium bed aligning the

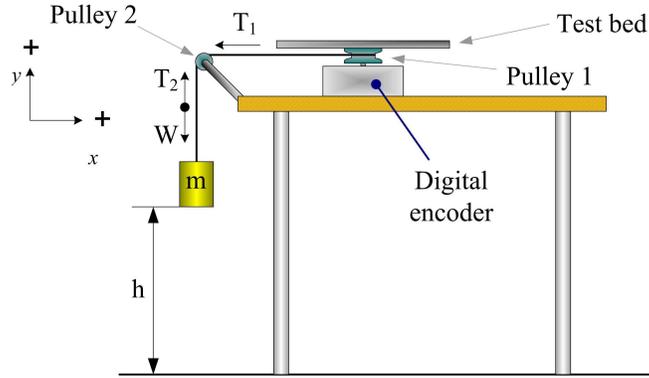


Figure 3.1: *MOI testbed general layout.*

helicopter *cg* to the center of the MOI testbed. When the control hanging mass is released, the helicopter starts to rotate about its *cg* and the angular position from the incremental encoder is collected by a PC equipped with a quadrature encoder interface. A very light aluminium frame is used to keep the helicopter in the correct position for the pitch and yaw moments experiment, see figures 3.2 to 3.4.

3.1.1 Moment of Inertia Testbed Mathematical Model

The equation of the hanging mass in the MOI testbed as shown in figure 3.1 is

$$T_2 - mg = -ma \quad (3.1)$$

where g is the gravity acceleration, T_2 is the tension on the string between the pulley 2 and the hanging gauge, m is the mass of the hanging gauge and a is the mass acceleration in the direction of y axis. Considering the dynamics of pulley 2 in the MOI testbed, the motion equation can be written as follows

$$\sum \tau = I_{p_2} \alpha_2 + b_2 \omega_2 + \tau_{s_2}$$

According to figure 3.1, the external torques are given by $\sum \tau = (T_2 - T_1)r_2$, hence

$$(T_2 - T_1)r_2 = I_{p_2} \alpha_2 + b_2 \omega_2 + \tau_{s_2} \quad (3.2)$$

In equation (3.2), the coefficient b_2 is the damping factor associated with the viscous friction, τ_{s_2} represents the moment of the dry kinetic friction, I_{p_2} is the mass moment of inertia of pulley 2, α_2 and ω_2 are the angular acceleration and

angular velocity respectively, r_2 is the radius of pulley 2 and T_1 is the tension of the string between pulley 1 and pulley 2. Similarly, the equation for pulley 1 is given by

$$\sum \tau = (I_{p_1} + I_{T_B} + I_H)\alpha_1 + b_1\omega_1 + \tau_{s_1}$$

where I_{p_1} is the mass moment of inertia of pulley 1, I_{T_B} is the flat aluminium bed moment of inertia, I_H is the moment of inertia of the object to be tested, b_1 is the damping coefficient associated with the viscous friction of pulley 1, τ_{s_1} is its moment of the dry kinetic friction, and α_1 and ω_1 are the angular acceleration and angular velocity respectively of pulley 1. From figure 3.1, the external torque of pulley 1 is r_1T_1 so

$$T_1r_1 = (I_{p_1} + I_{T_B} + I_H)\alpha_1 + b_1\omega_1 + \tau_{s_1} \quad (3.3)$$

Combining equations (3.2) and (3.3) and assuming that string is inextensible, the equation that describes the dynamics of the MOI testbed results in

$$\left(\frac{I_{pB} + I_H}{r_1} + \frac{I_{p_2}r_1}{r_2^2} + mr_1 \right) \ddot{\theta}_1 + \left(\frac{b_1}{r_1} + \frac{b_2r_1}{r_2^2} \right) \dot{\theta}_1 + \frac{\tau_{s_1}}{r_1} + \frac{\tau_{s_2}}{r_2} - mg = 0 \quad (3.4)$$

Here, I_{pB} is the summation of the moments of inertia of pulley 1 and the aluminium bed, that is, $I_{pB} = I_{p_1} + I_{T_B}$. Equation (3.4) represents the angular displacement of pulley 1 as a function of time. The hanging mass m is the control parameter and the moment of inertia of the object I_H is the parameter to be estimated.

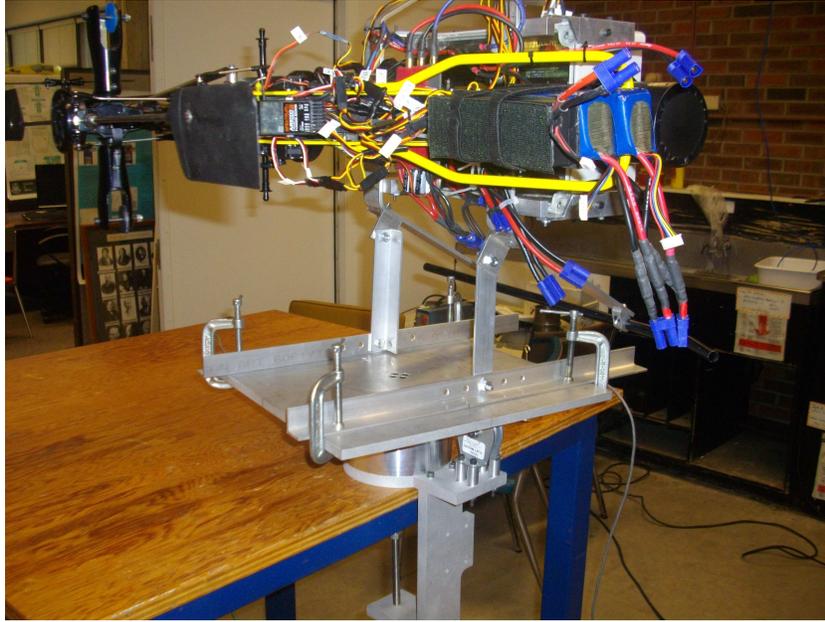
Solving equation (3.4) and using the initial conditions, the angular displacement for pulley 1 as a function of time is given by:

$$\theta(t) = \frac{(mg - k_3)(k_1 + mr_1)}{k_2^2} \left(e^{-\frac{t}{\tau}} - 1 \right) + \frac{mg - k_3}{k_2} t \quad (3.5)$$

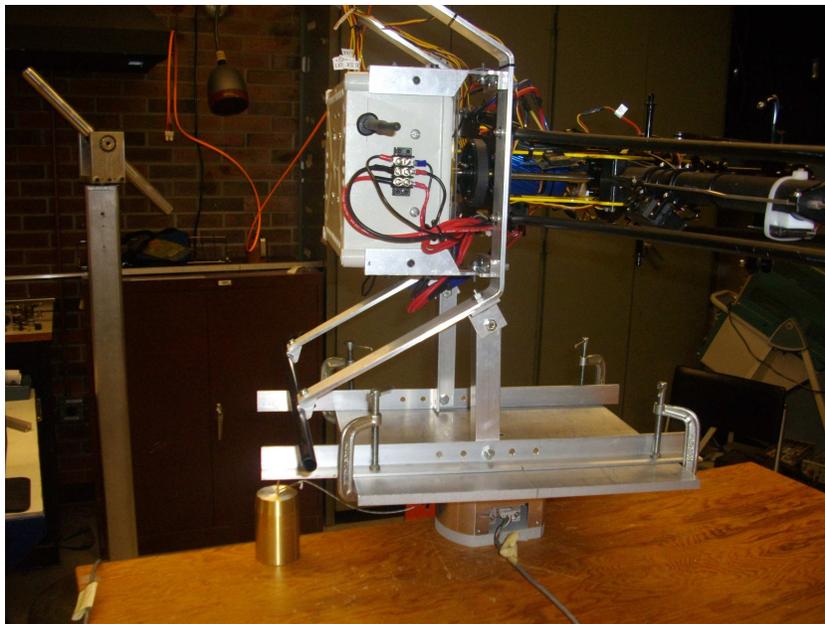
where m is the hanging mass, $k_1 = \frac{I_{pB} + I_H}{r_1} + \frac{I_{p_2}r_1}{r_2^2}$, $k_2 = \frac{b_1}{r_1} + \frac{b_2r_1}{r_2^2}$, $k_3 = \frac{\tau_{s_1}}{r_1} + \frac{\tau_{s_2}}{r_2}$ and $\tau = \frac{k_1 + mr_1}{k_2}$.

Since $\theta(t)$ is measured the moment I_H of the tested device can be experimentally determined.

The mass moment of inertia for pulley 1 and 2 are computed by assuming the pulleys as equivalent solid thin disks. The test plate MOI is assuming a



(a)



(b)

Figure 3.2: (a) Helicopter mounted for Pitch moment estimation, front view. (b) Helicopter mounted for pitch moment estimation, rear view.

thin solid rectangle with even mass distribution. Details of computations and the corresponding uncertainty analysis are in section B.2 of appendix B. In appendix B, table B.2, the physical characteristics of the components of the MOI testbed are summarized. In table 3.1 the results of the mass moments

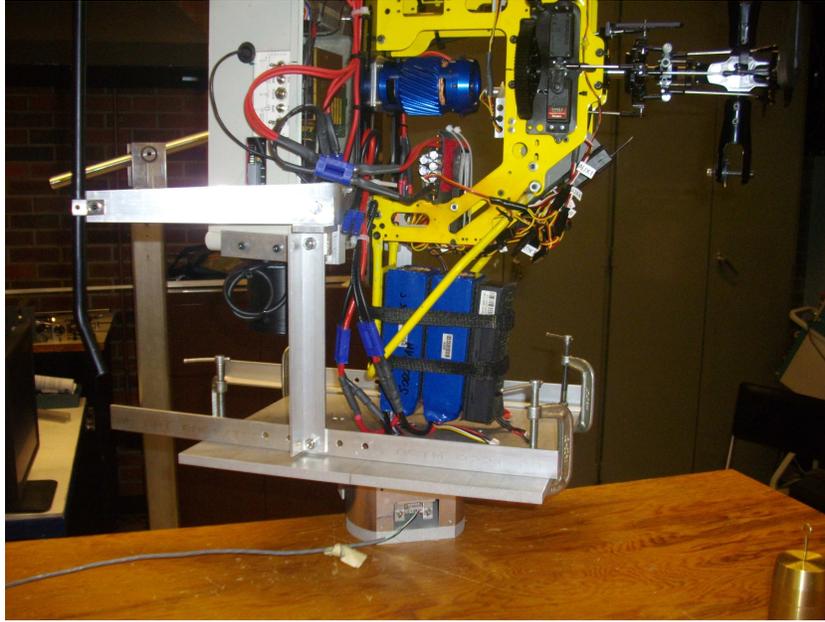


Figure 3.3: *Helicopter mounted for roll moment estimation. Partial view.*



Figure 3.4: *Helicopter mounted for roll moment estimation. Full view.*

computations are summarized. Listed in table 3.1, the dynamic and static friction coefficients for both pulley are summed together because $r_1 \approx r_2$, that

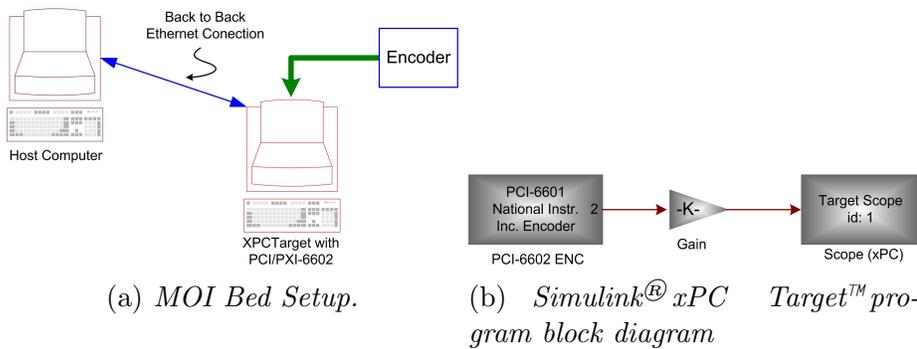
Table 3.1: *MOI testbed parameters.*

Parameter	Value	Units	Description
I_{p_1}	$6.21 \times 10^{-5} \pm 5.93 \times 10^{-8}$	kg·m ²	Pulley 1 MOI about it spinning axis
I_{p_2}	$3.69 \times 10^{-5} \pm 3.94 \times 10^{-8}$	kg·m ²	Pulley 1 MOI about it spinning axis
I_{T_B}	$0.0594 \pm 1.42 \times 10^{-4}$	kg·m ²	Testbed MOI about y axis

is, $b = \frac{1}{r_1}(b_1 + b_2)$ and $\tau = \frac{1}{r_1}(\tau_{s_1} + \tau_{s_2})$. Although, the values for these coefficients are small, they are necessary to get accurate inertia properties of the helicopter.

3.1.2 Helicopter Mass Moment of Inertia Estimation

To estimate the main helicopter's mass moments of inertia tensor three sets of thirty experiments are performed. Inertia products are considered small enough to be neglected without loss of accuracy. A 2000 divisions per revolution HEDS-9040 quadrature incremental position encoder are used to determine the angular position of the MOI testbed shaft. A MatLab™ xPC Target™ program on a PC interfaced to a National Instrument™, PCI/PXI-6602 I/O digital card is used to convert the quadrature output into angular position. Figure 3.5(a) shows an schematic diagram of the setup, and figure 3.5(b) shows the block diagram of the xPC Target™ program.

**Figure 3.5:** *Mass Moment Of Inertia Setup and Simulink[®] diagram.*

The first set of experiments are done about z_b axis to determine the yaw moment. The cg of the helicopter is carefully aligned to the MOI's center with the aid of a laser guide system. Helicopter is held to the MOI's flat aluminium testbed with four small clamps adding less than 75 grams to the system. One kilogram gauge mass of class 6, is used in the experiment. The experiment is

repeated 30 times and data is recorded on the host computer for analysis.

The same procedure is used for the pitch moment about the y_z axis, and for the roll moment about the x_b axis. In both cases, a light aluminium frame is used to hold the helicopter over the MOI's aluminium testbed as can be seen from figure 3.2. The mass moment of inertia introduced by the holding aluminium frame is estimated using the MOI testbed with the holding aluminium frame alone and then subtracted to get the values of the Helicopter mass moment of inertia about its different axes.

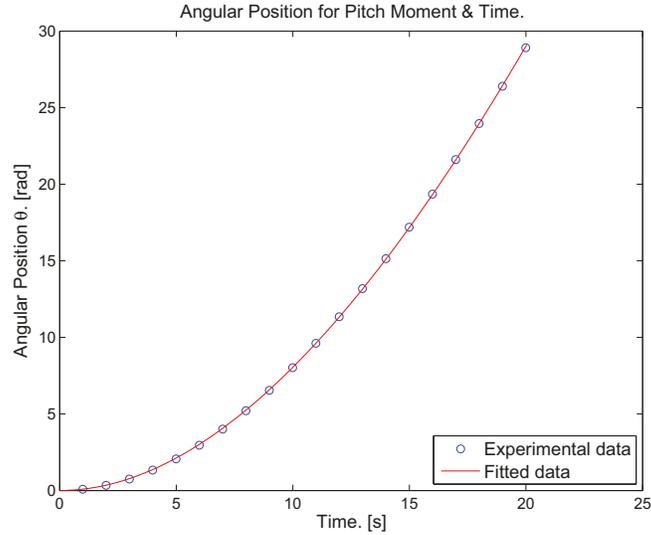
The MatLab™ fit curve tool box is use to fit data to the model given. Coefficients k_1 , k_2 , and k_3 in equation (3.5) are estimated using a non-linear regression algorithm.

Figures 3.6 to 3.8 compare the model fit to the data for all three experiment sets. In figure 3.6(a), an angular rotation of about 25 rad occurs in approximately 18 second. This is equivalent to approximately 3.98 revolutions in 18 second. For the roll moment, in figure 3.7(a) the same angular rotation of 25 rad occurs in about 10 seconds. This is because the pitch mass moment of inertia is large than the roll moment of inertia. The same is also observed for the angular velocities in figures 3.6(b) to 3.8(b), where the fastest angular velocity is in figure 3.7(b) for the roll mass moment, $4.5 \frac{\text{rad}}{\text{s}}$ in just 9 second. Although the angular velocities are computed by taking the approximate derivative from the angular position (as opposed to use of a state observer), the low noise level allows the fitting algorithm to properly fit the data within 95% of confidence interval.

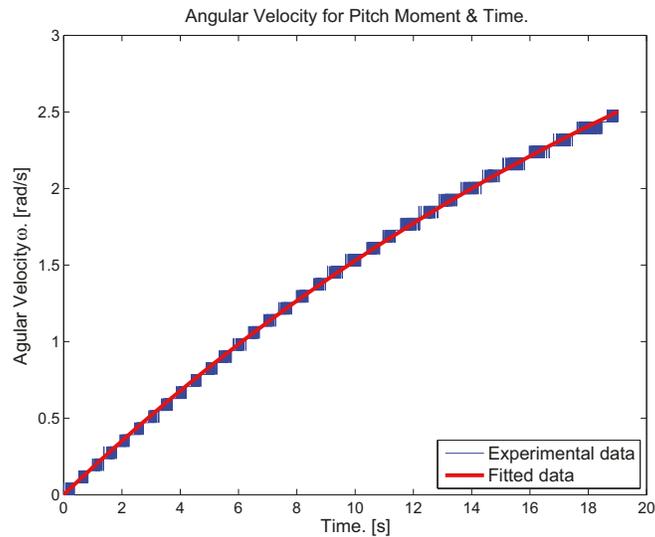
Table 3.2 summarize the results of the fitting process. As expected, the pitch moment is the largest, followed by the yaw moment and then the roll.

Table 3.2: *Helicopter's MOI.*

	MOI [Kg · m ²]	b [N · m · s]	τ_s [N · m]
Pitch	$1.5944 \pm 3.6456 \times 10^{-2}$	$0.0500 \pm 3.2588 \times 10^{-3}$	0.0
Yaw	$1.2741 \pm 9.298 \times 10^{-3}$	$0.0790 \pm 6.2519 \times 10^{-4}$	0.0
Roll	$0.5140 \pm 3.1739 \times 10^{-3}$	$0.0130 \pm 4.8792 \times 10^{-4}$	0.0



(a)

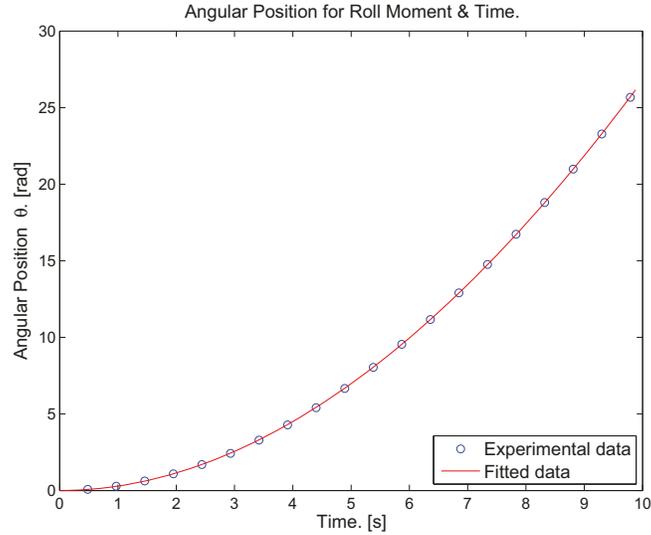


(b)

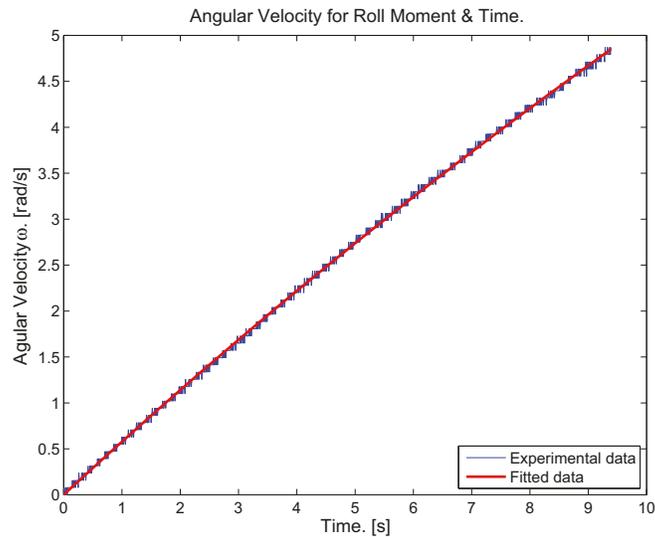
Figure 3.6: a) encoder position data, and b) velocity data for the pitch moment experiment.

3.2 HIL Force Testbed

A second testbed is described. The force testbed is used to calibrate and estimate parameters such as main rotor lift slope and drag coefficient, $C_{L\alpha}$, and $C_{D\alpha}$ respectively. The setup is a 1.8 m steel pole anchored to the ground as shown in figures 3.9 and 3.10. A $\frac{1}{4}$ inch (6 mm) thick rectangular steel plate is welded at the upper end of the pole and is used as a base to attach eight load



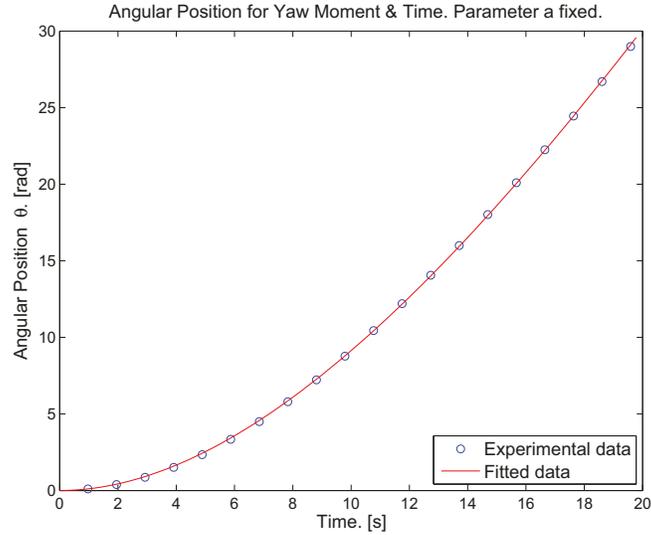
(a)



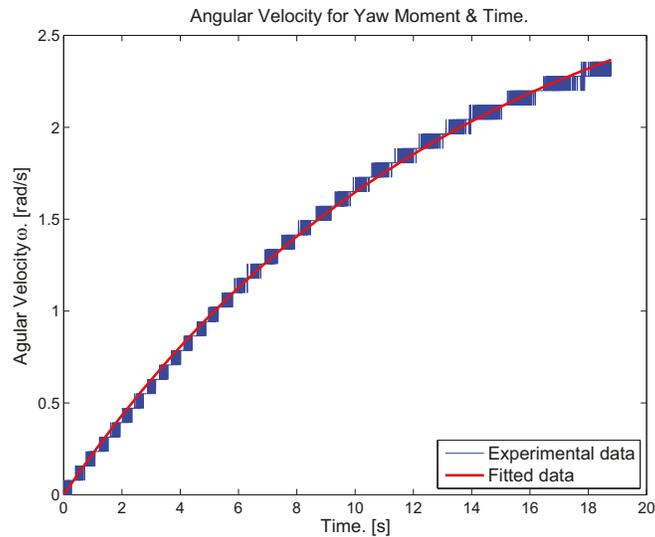
(b)

Figure 3.7: a) encoder position data, and b) velocity data for the roll moment experiment.

cells to read lateral, longitudinal and horizontal forces. A second aluminium plate is mounted on top the load cells is used to secure the helicopter. The helicopter is attached to the top plate so forces exerted by the helicopter can be measured. All the load cells are measured using a National Instrument™ PCI-6052E A/D-D/A converter card running xPC Target™ program on a PC. The same PC system is used to generate six pulse width modulation (PWM) signals



(a)



(b)

Figure 3.8: a) encoder position data, and b) velocity data for the yaw moment experiment.

to control the four swash-plate servomotors, the tail servo, and the throttle. A second computer acting as the host is connected to the xPC Target™ computer through an ethernet cable. The host computer is used to upload xPC Target™ programs for the different experiments and retrieve the corresponding data for analysis.

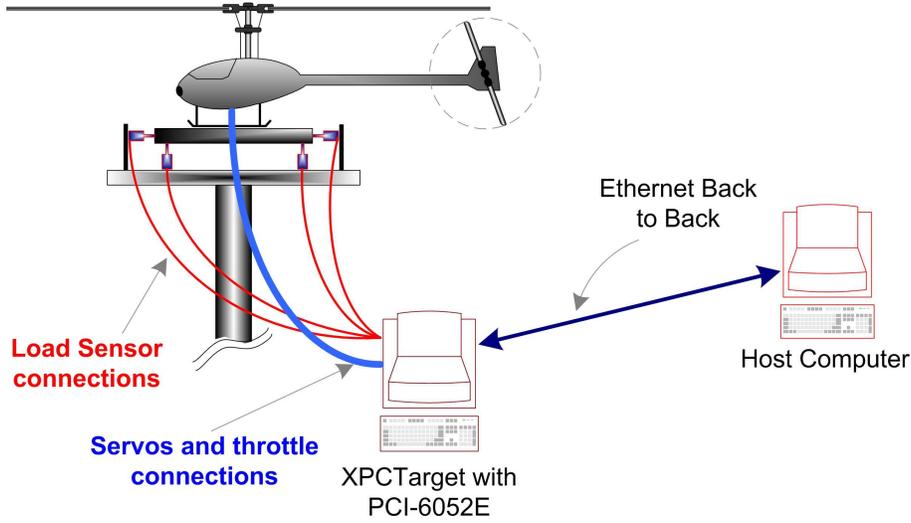


Figure 3.9: Block diagram of the Force Bed Setup. *xPC Target™* computer controls the four servo motors from the swash plate, the tail rotor servo and the throttle. At the same time the computer receives the analog information of the eight load sensors attached to the plate that supports the helicopter.

3.2.1 Force Testbed Equations

The free body diagram for the load cell sensors are shown in figure 3.11. Referring to figure 3.11(a), the moment about the z axis can be written as:

$$\mathbf{M}_{\psi} = \sum_{i=5}^8 \mathbf{M}_i \quad (3.6)$$

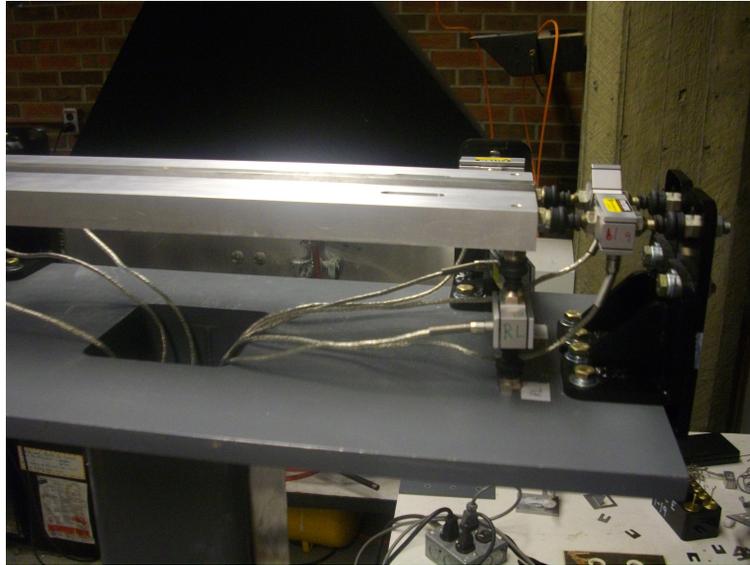
where the i -th moment is given by

$$\mathbf{M}_i = \mathbf{r}_i \times \mathbf{F}_i$$

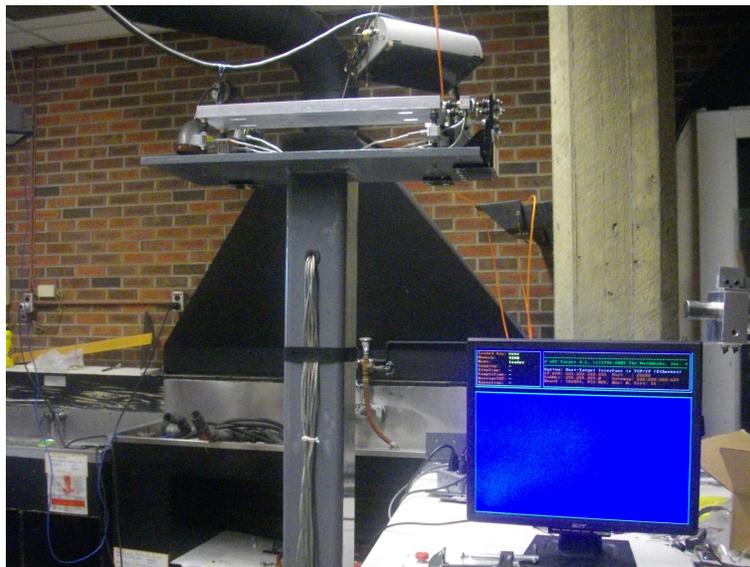
The components of the position vector from the testbed cg to each of load cells L_5 to L_8 are shown in figure 3.11(a) and are:

$$\begin{aligned} \mathbf{r}_5 &= -d_1 \mathbf{i} + d_5 \mathbf{j} \\ \mathbf{r}_6 &= d_2 \mathbf{i} + d_5 \mathbf{j} \\ \mathbf{r}_7 &= d_6 \mathbf{i} + d_3 \mathbf{j} \\ \mathbf{r}_8 &= d_6 \mathbf{i} - d_4 \mathbf{j} \end{aligned}$$

The component of the forces \mathbf{F}_5 and \mathbf{F}_6 are collinear with the y axis and the components of the forces \mathbf{F}_7 and \mathbf{F}_8 are collinear with the x axis.



(a)

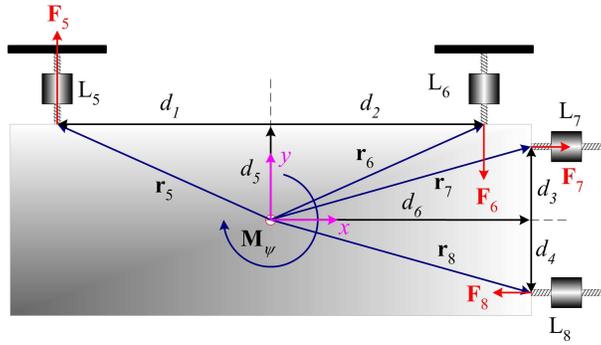


(b)

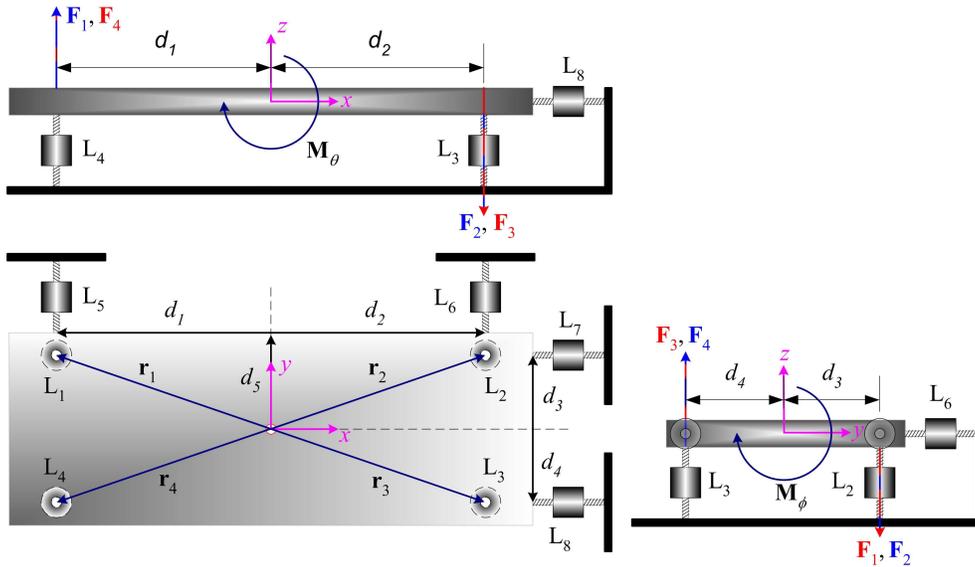
Figure 3.10: (a) Force testbed HIL aluminium platform. Here is possible to see some of the load cells (b) Partial view of the Force testbed HIL in which can be observed the pole, platform and the xPC Target™ program.

Referring to figure 3.11(b) is also possible to see that the total moment \mathbf{M}_T produced by the forces \mathbf{F}_1 to \mathbf{F}_4 is given by

$$\mathbf{M}_T = \sum_{i=1}^4 \mathbf{r}_i \times \mathbf{F}_i \quad (3.7)$$



(a) Free body diagram for the lateral load cells.



(b) Free body diagram for the longitudinal load cells.

Figure 3.11: Force testbed free body diagram for the lateral and longitudinal load cell sensors .

Using the geometry of the testbed in figure 3.11(b), the position vectors \mathbf{r}_1 to \mathbf{r}_4 in terms of its components are:

$$\begin{aligned} \mathbf{r}_1 &= -d_1\mathbf{i} + d_3\mathbf{j} \\ \mathbf{r}_2 &= d_2\mathbf{i} + d_3\mathbf{j} \\ \mathbf{r}_3 &= d_2\mathbf{i} - d_4\mathbf{j} \\ \mathbf{r}_4 &= -d_1\mathbf{i} - d_4\mathbf{j} \end{aligned}$$

The component of the forces \mathbf{F}_1 to \mathbf{F}_4 are collinear with the z axis. hence, the total moment is $\mathbf{M}_T = \mathbf{M}_\phi + \mathbf{M}_\theta$.

The main thrust force \mathbf{T} produced by the rotor, is just the summation of the four longitudinal forces

$$T = F_1 + F_2 + F_3 + F_4 \quad (3.8)$$

The other two forces exerted on the bed are the longitudinal force

$$\mathbf{F}_h = (F_7 + F_8)\mathbf{i} \quad (3.9)$$

and the lateral force.

$$\mathbf{F}_\ell = (F_5 + F_6)\mathbf{j} \quad (3.10)$$

Equations 3.6 to 3.10 are the forces and moment exerted by the helicopter on the Force Bed HIL.

3.2.2 Servo Motors and Load Cell Calibration

To obtain accurate values it is necessary to calibrate the servo control motors angles, and load cell forces.

For the servo calibration procedure, the 2000 division rotary wheel and the incremental encoder described in section 3.1.2 is used to calibrate all four servomotors of the swash-plate mechanism and the tail servo motor. A PWM signal with a pulse width from 0 ms to 3 ms is generated by xPC Target™ program. This signal is used as the input to the servos and then the output from the angular incremental encoder is measured. All the servos have linear response to the PWM input signal in the operational region as shown in figure B.1 of appendix B. Swash-plate servos have a linear operation range of $-73.26^\circ < \theta_{SP} < 68^\circ$ and $-55.55^\circ < \theta_T < 59.53^\circ$ for the tail servo. The functional relation between the applied pulse width and the angular displacement is given by equation (B.20) in appendix B.

A summary of the servo calibration outcomes is in table B.3 of appendix B.

Load cells model LRF325 with capacity of 455 N from FUTEK™ are calibrated using a class 6 gauge mass set, from 0.100 kg to 3.500 kg. All cells are tested both in tension and compression. All load cells exhibit a linear response within the tested range of ± 34 N, as can be seen in figures B.2 and B.3 of appendix B. According to these figures, the relation between the applied force

and the output voltage is given by a linear equation (B.21). The inverse of this equation is used to calculate the mechanical load exerted by the helicopter on the HIL force testbed load cells.

Table B.4 in appendix B, summarizes the load cell calibration results.

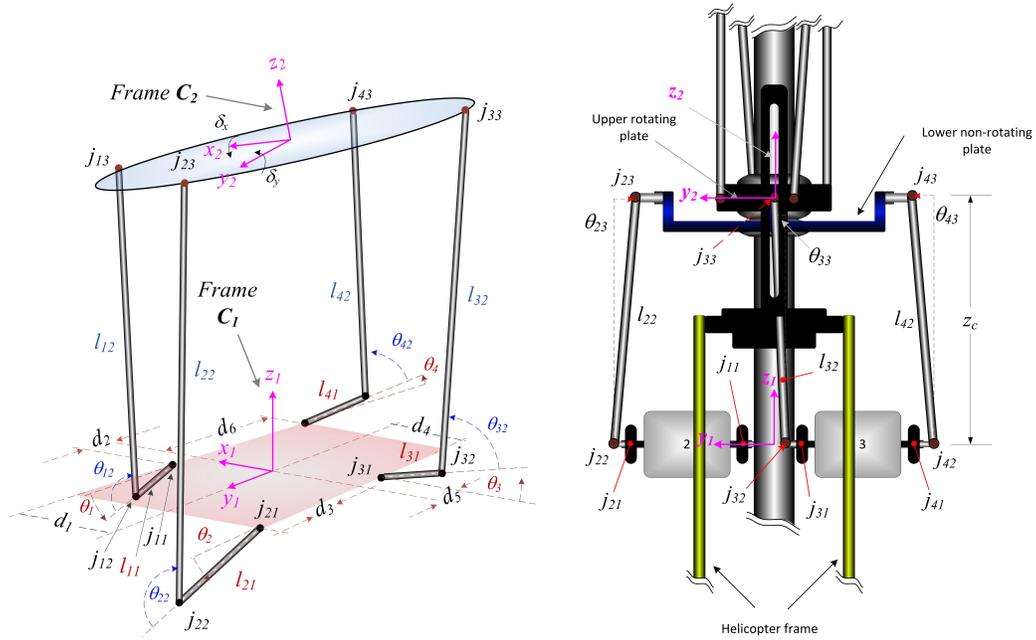
3.2.3 Lower Swash-plate Kinematics

To compute the lift force exerted by the helicopter, it is necessary to know the correct amount of collective pitch angle applied by the xPC Target™ program. To do this easily, the functional relation between the lower swash-plate angles and the servos' angular displacements is needed. Once these relations are known, the corresponding blade angles are measured using a digital inclinometer and an incremental encoder on the blade gripper mechanism.

In figure 3.12 is possible to see some aspects of the four servo swash-plate mechanism. Two coordinate systems are shown in figure 3.12(a). The first one, frame \mathbf{C}_1 is attached in the same plane where the servo motors are fixed, that is the frame that intersects the joints j_{11} , j_{21} , j_{31} , and j_{41} . This frame is fixed and does not move. The second frame, \mathbf{C}_2 has its origin in the plane that intersects joints j_{13} , j_{23} , j_{33} , and j_{43} . This frame does not rotate about the z_2 axis but about x_2 and y_2 following the motion of these joints – see figure 3.12(a), 3.12(b) and 3.12(c). To compute the relation between angles θ_1 to θ_4 and angles δ_x , δ_y and the linear distance z_c , the methodology presented in [24] is followed.

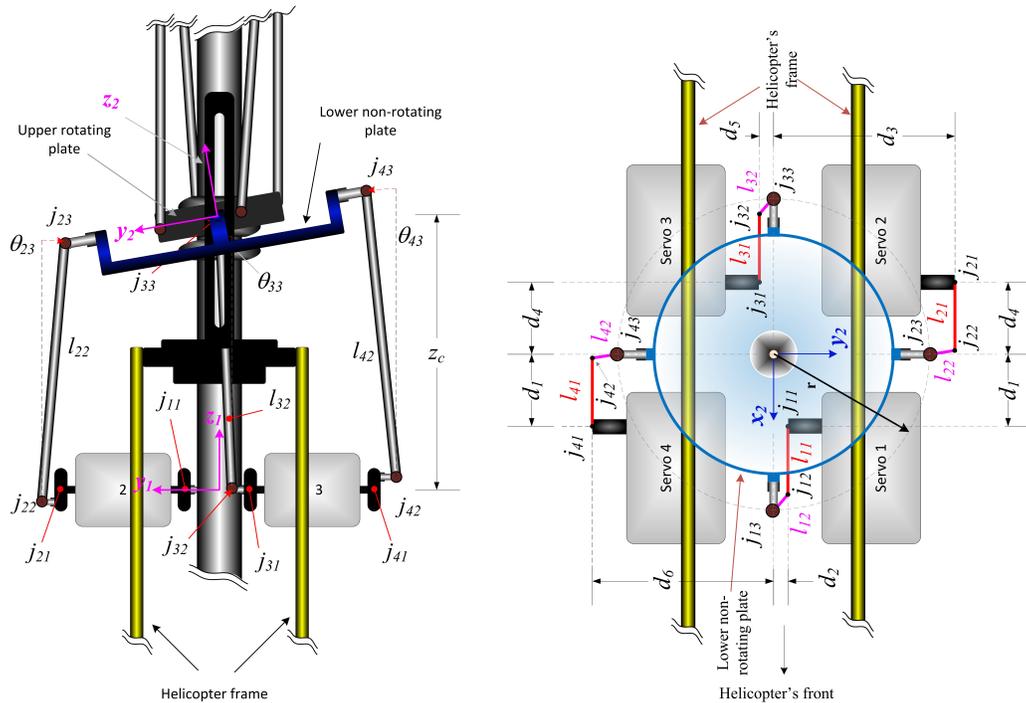
According to figure 3.12(a), the homogenous transformation \mathbf{C}_2^1 from frame \mathbf{C}_2 to frame \mathbf{C}_1 is accomplished by: a linear translation \mathbf{T}_{z_c} of a distance $-z_c$ along z_1 axis; followed by a rotation \mathbf{C}_{δ_y} of δ_y degrees about y_1 axis, finally with a rotation \mathbf{C}_{δ_x} of δ_x degrees about x_1 axis. This is mathematically represented by:

$$\mathbf{C}_2^1 = \mathbf{T}_{z_c} \mathbf{C}_{\delta_y} \mathbf{C}_{\delta_x} = \begin{bmatrix} \cos \delta_y & \sin \delta_y \sin \delta_x & \sin \delta_y \cos \delta_x & 0 \\ 0 & \cos \delta_x & -\sin \delta_x & 0 \\ -\sin \delta_y & \cos \delta_y \sin \delta_x & \cos \delta_y \cos \delta_x & -z_c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



(a) Frame definition for the lower swash-plate mechanism..

(b) Swash-plate mechanism lateral view with zero lateral and longitudinal angle.



(c) Swash-plate mechanism lateral view

(d) Swash-plate mechanism top view.

Figure 3.12: Schematic diagram of the swash-plate mechanism.

where the homogenous transformations \mathbf{T}_{z_c} , \mathbf{C}_{δ_y} , and \mathbf{C}_{δ_x} are detailed in section B.4 of appendix B.

Defining the augmented position vectors from the origin of frame \mathbf{C}_2 to joints j_{13} , j_{23} , j_{33} , and j_{43} , as follows

$$\begin{aligned} {}^2\mathbf{r}_{j_{13}} &= [r, 0, 0, 1]^T \\ {}^2\mathbf{r}_{j_{23}} &= [0, r, 0, 1]^T \\ {}^2\mathbf{r}_{j_{33}} &= [-r, 0, 0, 1]^T \\ {}^2\mathbf{r}_{j_{43}} &= [0, -r, 0, 1]^T \end{aligned}$$

Then the position vectors expressed in frame \mathbf{C}_1 are

$$\begin{aligned} {}^1\mathbf{r}_{j_{13}} &= \mathbf{C}_2^1 {}^2\mathbf{r}_{j_{13}} \\ {}^1\mathbf{r}_{j_{23}} &= \mathbf{C}_2^1 {}^2\mathbf{r}_{j_{23}} \\ {}^1\mathbf{r}_{j_{33}} &= \mathbf{C}_2^1 {}^2\mathbf{r}_{j_{33}} \\ {}^1\mathbf{r}_{j_{43}} &= \mathbf{C}_2^1 {}^2\mathbf{r}_{j_{43}} \end{aligned} \quad (3.11)$$

To solve the set of equations (3.11) for θ_1 , to θ_4 in terms of z_c , δ_x , and δ_y , is necessary to express ${}^1\mathbf{r}_{j_{13}}$ to ${}^1\mathbf{r}_{j_{43}}$ in an alternative form by following the linkages from the origin of frame \mathbf{C}_1 to each of the joints in the plane $x_2 - y_2$ as follows:

$$\begin{aligned} {}^1\mathbf{r}_{j_{13}} &= \mathbf{T}_{d_2} \mathbf{T}_{d_1} \mathbf{C}_{\theta_1} \mathbf{T}_{\ell_{11}} \mathbf{C}_{\theta_{12}} \mathbf{C}_{\theta_{13}} \boldsymbol{\ell}_{12} \\ {}^1\mathbf{r}_{j_{23}} &= \mathbf{T}_{d_3} \mathbf{T}_{d_4} \mathbf{C}_{\theta_2} \mathbf{T}_{\ell_{21}} \mathbf{C}_{\theta_{22}} \mathbf{C}_{\theta_{23}} \boldsymbol{\ell}_{22} \\ {}^1\mathbf{r}_{j_{33}} &= \mathbf{T}_{d_5} \mathbf{T}_{d_4} \mathbf{C}_{\theta_3} \mathbf{T}_{\ell_{31}} \mathbf{C}_{\theta_{32}} \mathbf{C}_{\theta_{33}} \boldsymbol{\ell}_{32} \\ {}^1\mathbf{r}_{j_{43}} &= \mathbf{T}_{d_6} \mathbf{T}_{d_1} \mathbf{C}_{\theta_4} \mathbf{T}_{\ell_{41}} \mathbf{C}_{\theta_{42}} \mathbf{C}_{\theta_{43}} \boldsymbol{\ell}_{42} \end{aligned} \quad (3.12)$$

where $\boldsymbol{\ell}_{12} = [\ell_{12}, 0, 0, 1]^T$, $\boldsymbol{\ell}_{22} = [\ell_{22}, 0, 0, 1]^T$, $\boldsymbol{\ell}_{32} = [-\ell_{32}, 0, 0, 1]^T$, $\boldsymbol{\ell}_{42} = [-\ell_{42}, 0, 0, 1]^T$. The definition of all homogenous translations and rotation matrices found in equation (3.12) are in section B.4 of appendix B.

Equating each of equations given in (3.11) and (3.12) results in:

$$r \cos \delta_y = \cos(\theta_1 + \theta_{12}) l_{12} \cos \theta_{13} + l_{11} \cos \theta_1 + d_1 \quad (3.13a)$$

$$0 = l_{12} \sin \theta_{13} + d_2 \quad (3.13b)$$

$$-r \sin \delta_y - z_c = -\sin(\theta_1 + \theta_{12}) l_{12} \cos \theta_{13} - l_{11} \sin \theta_1 \quad (3.13c)$$

$$r \sin \delta_y \sin \delta_x = \cos(\theta_2 + \theta_{22}) l_{22} \cos \theta_{23} + l_{21} \cos \theta_2 - d_4 \quad (3.14a)$$

$$r \cos \delta_x = l_{22} \sin \theta_{23} + d_3 \quad (3.14b)$$

$$r \cos \delta_y \sin \delta_x - z_c = -\sin(\theta_2 + \theta_{22}) l_{22} \cos \theta_{23} - l_{21} \sin \theta_2 \quad (3.14c)$$

$$-r \cos \delta_y = -\cos(\theta_3 + \theta_{32}) l_{32} \cos \theta_{33} - l_{31} \cos \theta_3 - d_4 \quad (3.15a)$$

$$0 = -l_{32} \sin \theta_{33} - d_5 \quad (3.15b)$$

$$r \sin \delta_y - z_c = \sin(\theta_3 + \theta_{32}) l_{32} \cos \theta_{33} + l_{31} \sin \theta_3 \quad (3.15c)$$

$$-r \sin \delta_y \sin \delta_x = -\cos(\theta_4 + \theta_{42}) l_{42} \cos \theta_{43} - l_{41} \cos \theta_4 + d_1 \quad (3.16a)$$

$$-r \cos \delta_x = -l_{42} \sin \theta_{43} - d_6 \quad (3.16b)$$

$$-r \cos \delta_y \sin \delta_x - z_c = \sin(\theta_4 + \theta_{42}) l_{42} \cos \theta_{43} + l_{41} \sin \theta_4 \quad (3.16c)$$

To get an expression for θ_1 from equation (3.13) these steps are needed [24]: squaring both sides of equations (3.13a) and (3.13c), adding them together, and solving for θ_1 . This results in:

$$\theta_1 = \arccos \left[\frac{C_{a1}^2 + C_{b1}^2 + \ell_{11}^2 - C_{c1}^2}{2C_{a1}\ell_{11}} \cos \vartheta_1 \right] - \vartheta_1 \quad (3.17)$$

Following the same procedure for sets (3.14) to (3.16) results in

$$\theta_2 = \arccos \left[\frac{C_{a2}^2 + C_{b2}^2 + \ell_{21}^2 - C_{c2}^2}{2C_{a2}\ell_{21}} \cos \vartheta_2 \right] - \vartheta_2 \quad (3.18)$$

$$\theta_3 = \arccos \left[\frac{C_{c3}^2 - C_{a3}^2 - C_{b3}^2 - \ell_{31}^2}{2C_{a3}\ell_{31}} \cos \vartheta_3 \right] - \vartheta_3 \quad (3.19)$$

$$\theta_4 = \arccos \left[\frac{C_{c4}^2 - C_{a4}^2 - C_{b4}^2 - \ell_{41}^2}{2C_{a4}\ell_{41}} \cos \vartheta_4 \right] - \vartheta_4 \quad (3.20)$$

where

$$\begin{aligned}
C_{a1} &= r \cos \delta_y - d_1 & C_{a2} &= r \sin \delta_y \sin \delta_x + d_4 \\
C_{b1} &= -r \sin \delta_x - z_c & C_{b2} &= r \cos \delta_y \sin \delta_x - z_c \\
C_{c1} &= \ell_{12} \cos \theta_{13} & C_{c2} &= \ell_{22} \cos \theta_{23} \\
\vartheta_1 &= \arctan 2(C_{b1}, C_{a1}) & \vartheta_2 &= \arctan 2(C_{b2}, C_{a2}) \\
\theta_{13} &= \arcsin \left[\frac{-d_2}{\ell_{12}} \right] & \theta_{23} &= \arcsin \left[\frac{r \cos \delta_x - d_3}{\ell_{22}} \right] \\
\\
C_{a3} &= -r \cos \delta_y + d_4 & C_{a4} &= r \sin \delta_y \sin \delta_x - d_1 \\
C_{b3} &= r \sin \delta_x - z_c & C_{b4} &= -r \cos \delta_y \sin \delta_x - z_c \\
C_{c3} &= \ell_{32} \cos \theta_{33} & C_{c4} &= \ell_{42} \cos \theta_{43} \\
\vartheta_3 &= \arctan 2(C_{b3}, C_{a3}) & \vartheta_4 &= \arctan 2(C_{b4}, C_{a4}) \\
\theta_{33} &= \arcsin \left[\frac{-d_5}{\ell_{32}} \right] & \theta_{43} &= \arcsin \left[\frac{r \cos \delta_x - d_6}{\ell_{42}} \right]
\end{aligned}$$

To solve for the above terms, the dimensions of the lower swash-plate mechanism linkages and frame distances are in table B.5 of appendix B.

Expression (3.17) to (3.20) are the output angles of the four swash-plate mechanism servos in response to the required angles δ_x , δ_y , and offset z_c . These equations constitute the inverse kinematics of the lower swash-plate mechanism.

To get the functional relation between the servo angles and the blades collective pitch angle $\delta_{\theta_{col}}$, six measurements with a digital inclinometer attached on the blade gripper are taken. As shown in figure 3.13 there is a linear relation of the form

$$\delta_{\theta_{col}} = mz_c + b$$

between the blade collective pitch angle $\delta_{\theta_{col}}$ and the lower swash-plate linear displacement z_c , with $m = 1214 \frac{\circ}{\text{m}}$ and $b = -80.79^\circ$. This result is consistent with results from the incremental encoder attached to the blade gripper, which resulted in a linear relation and similar parameters of $m = 1229 \frac{\circ}{\text{m}}$ and $b = -84.00^\circ$, as shown in figure 3.14. Although it is observed in figure 3.14 that the whole swash-plate exhibits a small (0.3°) amount of mechanical hysteresis, it is not significant, thus this linear relation is used.

Figures B.4 to B.6 in section B.5 of appendix B, detail the encoder output vs. lower swash-plate mechanism linear displacement z_c , the servos responses vs. z_c , and the functional relation between the servo angles and the blade collective pitch angle $\delta_{\theta_{col}}$. In addition, table B.6 summarizes the coefficients

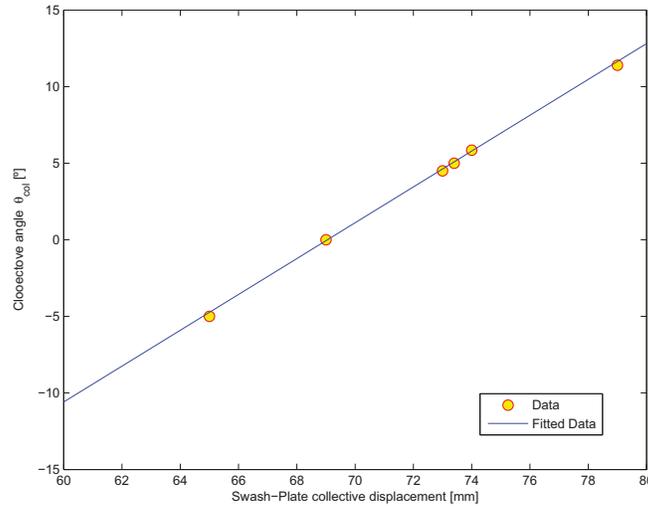


Figure 3.13: Blade collective pitch angle $\delta_{\theta_{col}}$ vs. swash-plate linear displacement z_c and its fitted curve. Data taken with digital inclinometer.

of the functional relations.

With this linear relation, the inverse kinematic equations given in (3.17) to (3.20), and the servo and the load cell equations obtained in the calibration process, is possible to produce adequate excitation like controlled throttle input and smooth collective angle input (see next section) for the HIL Force Bed experiments.

3.2.4 HIL experiments and Parameters ID

To test the helicopter the experiment setup described in section 3.2 which consist of a plate attached to eight load cells to sense the forces exerted is used. All the load cell are connected to a data acquisition system, see figure 3.10, and processed by an xPC Target™ program that controls the motions of all servo motors and records data for further analysis. The Simulink® xPC Target™ block diagram program is shown on figure 3.15.

Analog signals coming from the load cells are filtered and amplified in the *Load Cell Reading* block. The filtered signals are then input into the *HIL Force Bed Equation* block which transforms the eight load cell raw forces into thrust, lateral and longitudinal forces and moments M_ψ , M_θ and M_ϕ according to equations (3.6) to (3.10). Each of these forces are stored in the *recording*

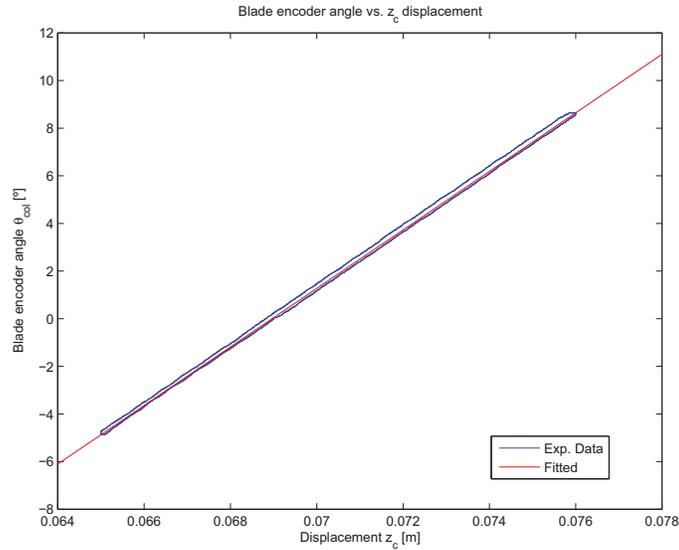


Figure 3.14: Blade collective pitch angle $\delta_{\theta_{col}}$ vs. swash-plate linear displacement z_c and its fitted curve. Data taken with incremental encoder.

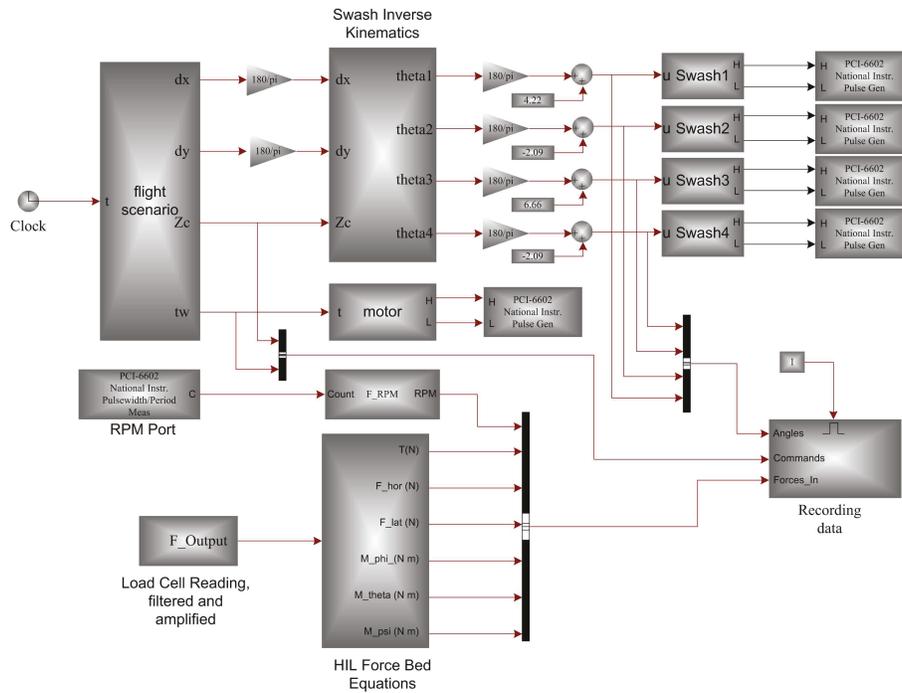


Figure 3.15: HIL force testbed xPC Target™ main program block diagram. Raw signals coming from the sensors are filtered and amplified in the Load cell reading block and transferred to the HIL block to compute helicopter thrust, longitudinal and lateral forces and magnitude moments.

block, as shown in figure 3.15. Simultaneously, the excitation signals are generated in the *Flight scenario* block. Here, the lower swash-plate angles δ_x , δ_y

and its linear displacement z_c are designed to test different aspects of the helicopter. For instance, to test helicopter thrust, angles δ_x and δ_y are both set to zero for all the time while z_c is set to a sinusoidal, random or ramp signal. These three signals are transformed in the *Swash Inverse Kinematics* block into the corresponding four servo angles that control the lower swash-plate. The *flight scenario* block also generates throttle signal t_w that controls the main rotor speed. This signal is transferred to the *motor* block which transforms the pulse width into equivalent high and low pulses. Finally, the output from blocks *motor* and *Swash Inverse Kinematics*, go into four identical blocks *PCI-6602* to generate the PWM signal that control all servos and main rotor driver.

The throttle signal is designed to smoothly start the main rotor. Main rotor angular speed Ω was set to 850 RPM, approximately $90 \frac{\text{rad}}{\text{s}}$. This speed was maintained for 235 second which is the duration of the thrust test, as shown in figure 3.16. A stair shape blade collective pitch angle signal $\delta_{\theta_{col}}$ starting at 50

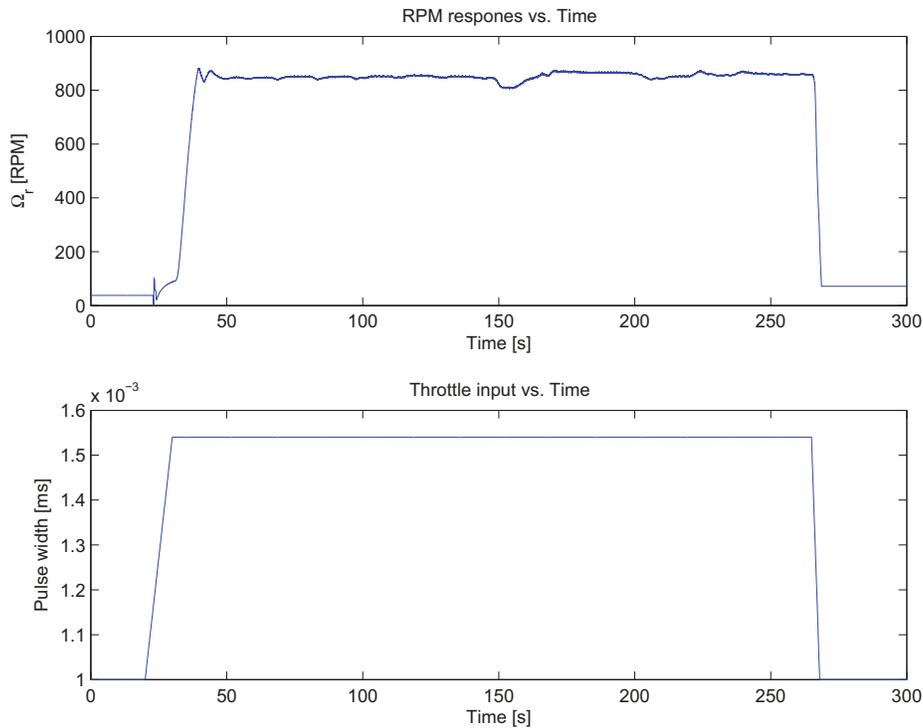


Figure 3.16: *Throttle input and RPM signals used in several test.*

seconds and ending at 260 second, shown in figure 3.17(d), is implemented in

the *flight scenario* block. Angles δ_x and δ_y were set to zero all along the test. The resulting thrust force \mathbf{T}_r and drag torque \mathbf{Q}_r exerted by the helicopter are plotted versus time in figure 3.17 along with the collective pitch input signal and the main rotor angular velocity. Thrust reaches its maximum value of 136.5 N at an input angle $\delta_{\theta_{col}}$ of 8.7° while the drag torque reaches a value of 12.5 N · m simultaneously.

Equations (2.53) and (2.54) and equations (2.64) and (2.65) are used to estimate the lift slope coefficient C_{L_α} and the zero lift drag coefficient C_{D_0} respectively. These tests are done under lab conditions, so the wind velocity vector components u_w , v_w and w_w are small and assumed zero. In addition, body velocities u , v , and w as well as the moments p , q , r are zero because helicopter is constrained not to move. Hence, equations (2.53) and (2.54) reduce to

$$T_r = \frac{\rho\pi R^4 \Omega^2 \mathcal{S}}{4} C_{L_\alpha} \left[\frac{2}{3} \delta_{\theta_{col}} - \lambda_i \right] \quad (3.21)$$

Although helicopter never detaches from the HIL force testbed, and because the ground effect is minimized by running the test on the pole 1.8 m above the ground, helicopter may be considered operating in a virtual hover condition. Therefore, the induced velocity v_i is just the induced flow in hover and can be computed as

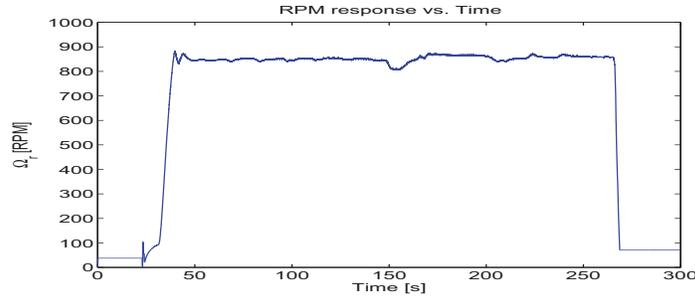
$$v_i = \sqrt{\frac{|T_r|}{2\rho\pi R^2}} \quad (3.22)$$

Equation (3.21) and (3.22) are used to reproduce the thrust force corresponding to a number of measurements and the lift slope coefficient C_{L_α} is estimated using iterations. After some trial and error, a value of $C_{L_\alpha} = 6.4$ is found to have a small error between the model and the experimental data as shown in figure 3.18.

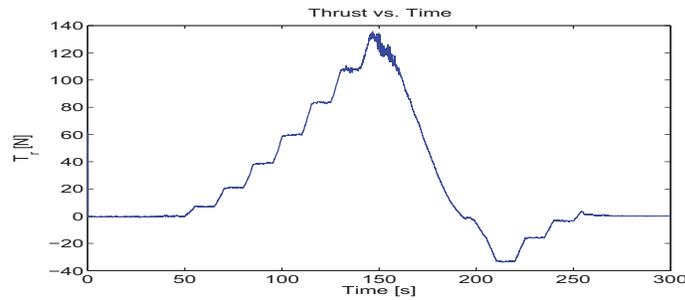
Using equations (2.64) and (2.65) in similar way it is found that the zero lift drag coefficient is about 0.012.

The mass moment of inertia tensor, the main rotor zero lift drag coefficient and the main rotor lift slope coefficient of the Air Star Evolution helicopter have been determine through these two simple but effective experiments.

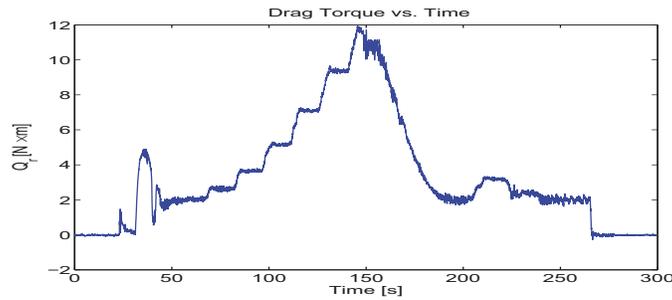
The next chapter describes in detail the instrumentation of the helicopter,



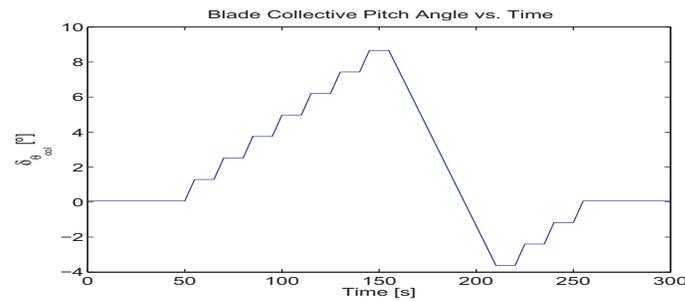
(a)



(b)



(c)



(d)

Figure 3.17: (a) Main rotor angular speed Ω_r . (b) exerted thrust force T_r with a maximum of 136.6 N at a collective angle $\delta_{\theta_{col}}$ of approximately 8.7° . (c) drag torque Q_R with a maximum of 11.69 N·m at the same angle. (d) Collective pitch angle input $\delta_{\theta_{col}}$.

the ground station software, and the communication protocol used to communicate to the onboard computer and the ground station.

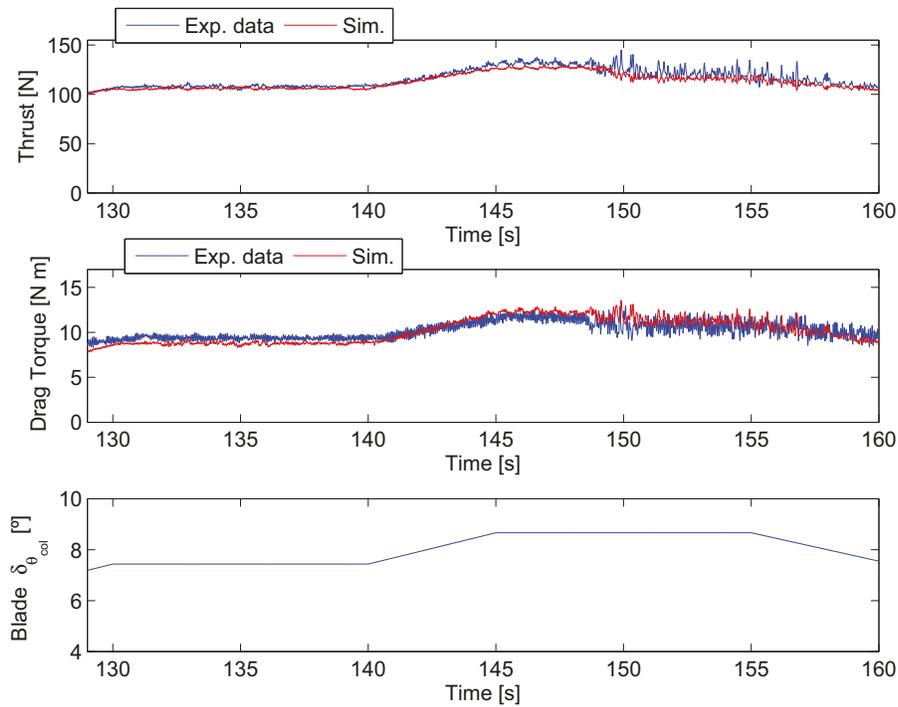


Figure 3.18: Main rotor thrust and drag force from the HIL force testbed. Experimental data and simulation results to identify the main rotor lift slope $C_{L\alpha}$ and the zero-lift drag coefficient C_{D_0} .

Chapter 4

Helicopter's Instrumentation

Electronics' Control, Instrumentation and Communication Hardware, or ECICH for short, are essential to UAV and UAS operation. Despite all the State-Of-The-Art mechanical wonders, clearly without any electronics, any UAV as we know today would not be possible. The implementation of an ECICH or Avionic Box is particularly challenging for small and miniature UAV due to the size limitations, payload constraints, power consumption restrictions, and the Air Star Evolution helicopter is not the exception. In addition, because the RC market is relatively small, it seems to lack standard parts. There is even variation in physical dimensions and characteristics even between the same RC model and brand, this adds even more complexity to the ECICH design and implementation.

Subject to all these general constraints, two possible design philosophies could be followed. The first one is a custom hardware design in which all the electronics are specifically designed to meet all the constraints and requirements of each specific helicopter. A second design philosophy is to do the best integration of all available hardware in the market to broadly cover most UAVs that are widely available.

Both philosophies have pros and cons. For example, in the custom design it is possible to achieve full integration in size, payload constraints, hardware and software performance, and the ability to modify the firmware and software at all levels to meet any additional requirement or fix any unexpected behavior. However, with this flexibility comes a considerable amount of time in planing, designing, implementing, debugging, and cost. In other words, the

complete design from scratch of an ECICH is an intensive time consuming and expansive project.

The second philosophy of design, integration of the ECICH has two main advantages. The first one is the quick turnaround time process. Second, it has relatively low cost compared to the custom design. The quick turnaround and low cost come with several disadvantages when all the modules are integrated into one system. For example, it is necessary to do some mechanical modifications on the helicopter structure (landing gear) to allocate and physically attach all the modules to the helicopter. Another important problem is the wiring process itself. Wiring all the modules with the power supply, the servos, the RC reception module, and others wiring requirements is challenging since these connections are likely sources of failure due to false contact or short circuits during helicopter operation. In addition, many modules have extra features that in most of the cases are never used. Finally, programming the modules and setting the communication between them is potentially difficult because many modules have proprietary firmware and/or software design that cannot easily be changed. In addition, some of these modules do not provide time stamp information with date provided introducing time uncertainty in the analysis.

In order to overcome some of these disadvantages but attempt to take the best of each design philosophy, a compromised philosophy is followed. The ECICH in this work is implemented by the integration of various standard and high quality hardware available in the market, combined with custom hardware. This is done to reduce the possible source of failure while maintaining performance and reliability.

In the following section a complete and detailed description of the Avionics Box or ECICH is given. In section 4.2, the software for the ground station is described. In section 4.3 the ECICH is tested to validate its performance and reliability, including a car ride test.

4.1 ECICH, Avionic Box

The Avionic Box or ECICH used in the present work is a combination of different integrated modules and custom hardware design as shown in figure 4.1, and consists of several modules. The main computer is an industrial embedded PC/104™ computer model ADL855PC™. This computer executes all the necessary tasks to coordinate the communication between modules, acquires all the flight parameters and stores them on to a compact flash memory card. In addition, the PC/104™ controls all the servo motors through a servo switch card, executes all the incoming commands from the ground station and transmits many parameters back to the ground station. The Inertial Measurement Unit (IMU) is a Crossbow® NAV440® Series Inertial System and is a complete navigation solution. The IMU module communicates to the PC/104™ using a full duplex RS-232 interface at a rate of 57,600 bps. The servo Switch Controller (SSC) module SSA20024™ from Omni Instruments is a combination of FPGAs and a micro-controller. The SSC receives the servo control command from the RC receiver and transfers them to the corresponding servomotors under the control and supervision of the embedded computer. The SSC also receives servo commands from the embedded computer for autonomous control of the helicopter. SSC communicates to the PC/104™ using a full duplex RS-232 interface at a rate of 115,200 bps. The ranging device model URG-04LX™ from Hokuyo Automatic, is a laser range finder used to measure the altitude in the take-off and landing phase of the Air Star Evolution helicopter operation. The Laser Range Finder or LRF, communicates to the PC/104™ through a full duplex RS-232 interface at a rate of 115,200 bps. The Battery Monitor System or BMS, consists of a μ Controller, an analog to digital converter, hall-effect sensors, barometric pressure and temperature sensor. This module collects values of voltages and currents from all the batteries and transfer them to the PC/104™ through an RS-232 interface at a rate of 38,400 bps. The BMS module also communicates to the pressure and temperature sensor via the Synchronous Serial Port, or SPP, using the Serial Peripheral Interface, or SPI protocol to get the current temperature and barometric pressure measurements and transfer them to the PC/104™. The next module is the XStream® Radio modem from Digi®. This RF modem has an RF throughput of 20,000 bps, and provides the radio link data between the Avionic box and the Ground Station. The RF modem communicates to the PC/104™ through a full duplex

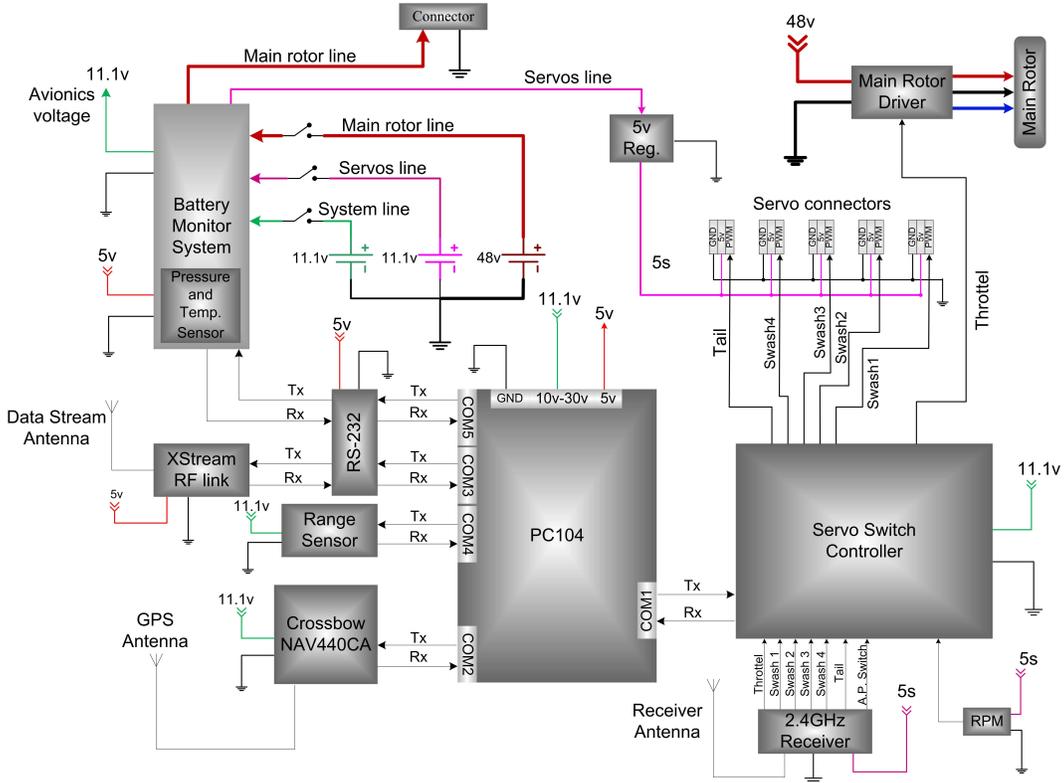


Figure 4.1: ECICH Block Diagram

RS-232 interface at a rate of 19,200 bps. The last module of the ECICH is the 5 V power supply. It consists of two separate 5 A, 5 V switched IC modules that deliver power to all servo motors, RC receiver and the main rotor driver.

The entire system is powered by three different sets of batteries. One set for the motor of the main rotor which requires, from the Electronic Speed Controller to operate properly, a range of voltages between 32 V to 42 V depending on the rotor load and RPM settings. The second battery is for the servo motors. All the servos on the Air Star Evolutions works in a range of 4.8 V to 6.0 V except for the tail servo that works only at 4.8 V. Therefore, a battery of 11.1 V is used and connected to the step down converter power supply on the avionic box to set the output voltage to 5 V. The last 3-cell LiPo battery is used to power the whole Avionic Box. The battery configuration is to provide a smooth flow of DC current while minimizing possible charge unbalance or current peaks induced from the servo load or the main rotor load. Appendix C.1 shows all the schematics diagrams and the PCB layout configuration of the ECICH, while figure 4.2 highlights some of the important aspects

of the ECICH.

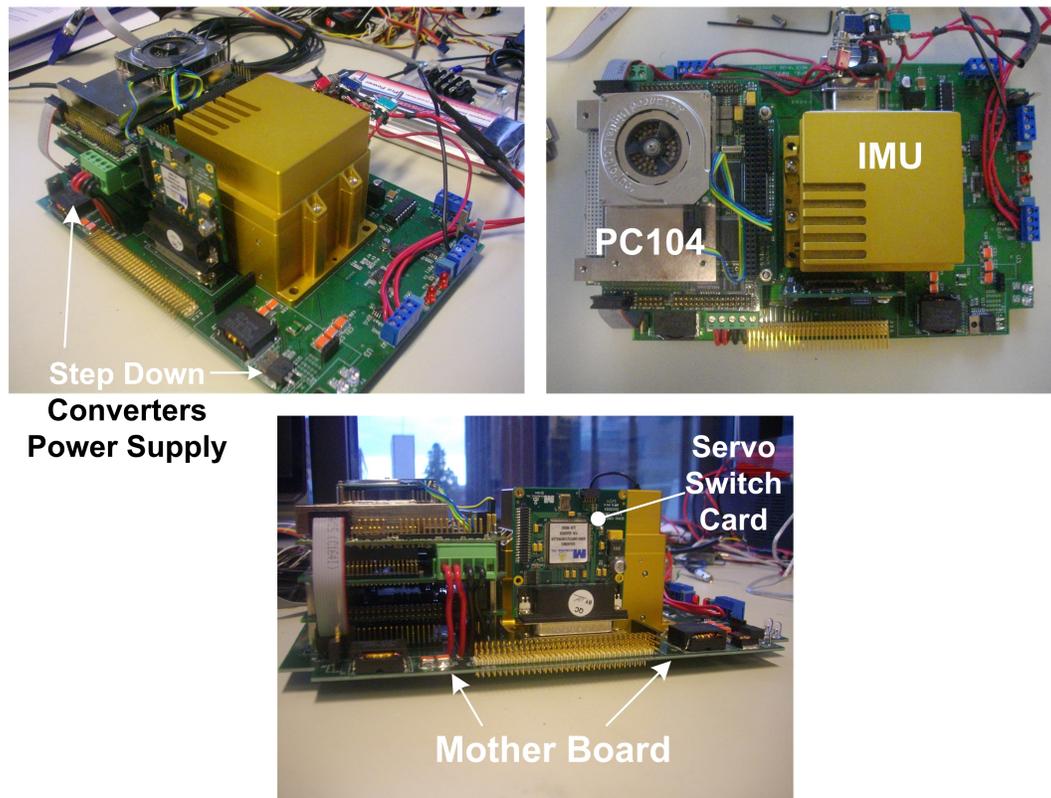


Figure 4.2: *Different aspects of the ECICH box.*

4.1.1 Inertial Measurement Unit

The Inertial Measurement Unit or IMU for short, is a Crossbow[®] NAV440[®] Series Inertia Systems. It is a 6 Degree-Of-Freedom (DOF) fully integrated IMU. Figure 4.3 is a block diagram of the hardware while figure 4.4 is a block diagram of the software implementation.

The core of the IMU is the 6 DOF sensor cluster constituted by 3 independent Micro-Electro-Mechanical-Systems, or MEMS for short, gyros and 3 independent MEMS accelerometers as shown in figure 4.3. In addition, the cluster includes a solid state temperature sensor employed to compensate the temperature bias errors of the MEMS devices over a range of $-40\text{ }^{\circ}\text{C}$ to $71\text{ }^{\circ}\text{C}$. The outputs of the MEMS are input of separate digital controlled programmable analog Low-Pass filters, which are used as anti-aliasing, and to adjust the aircraft dynamics according to different maneuvers that it might perform. A three axis analog magnetometer is included to make corrections in

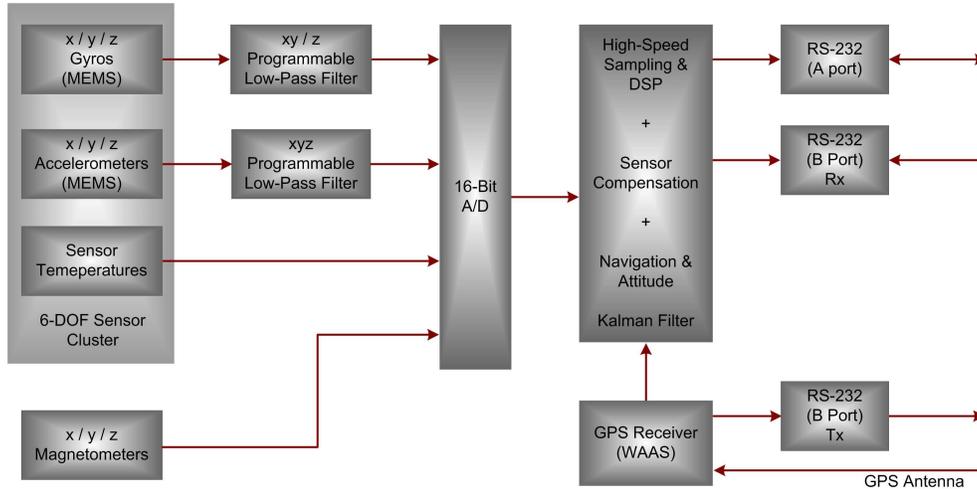


Figure 4.3: IMU Hardware block diagram.

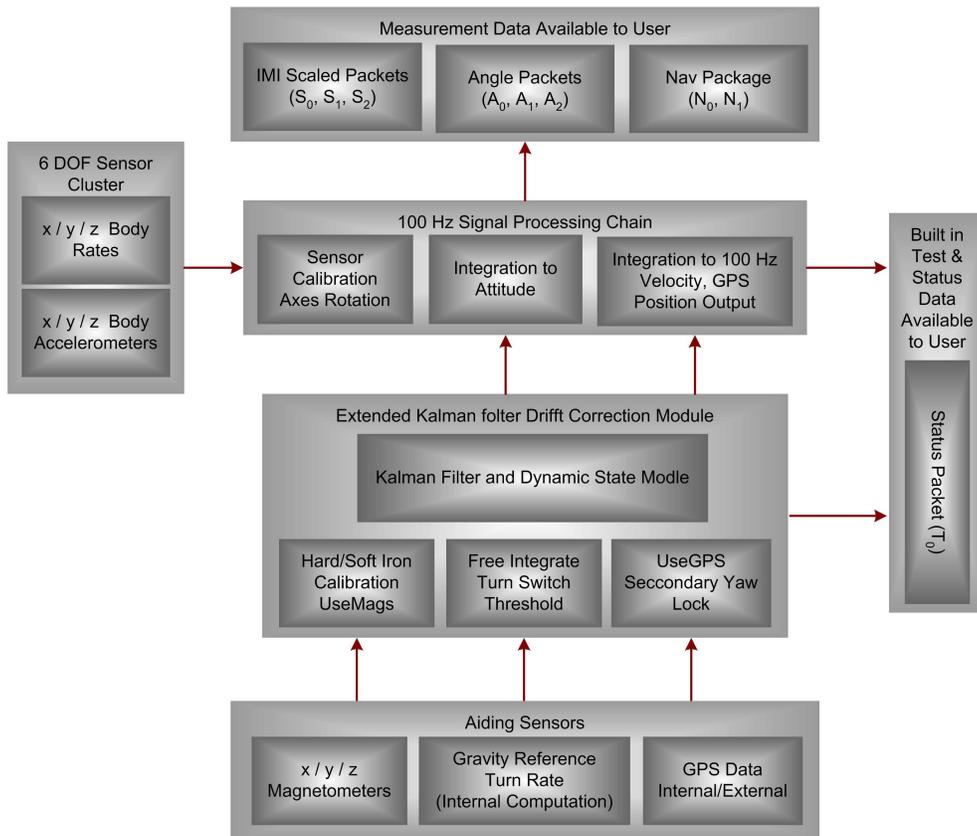


Figure 4.4: IMU Software block diagram.

the heading measurement, resulting in a complete navigation solution. Signals coming from the filters and the magnetometer are measured using a 16 bit A/D converter sampled at a rate of 1 KHz. After sampling, the Digital Signal Processor (DSP) uses a digital Finite Impulse Response (FIR) filter to deci-

mate the signal to 100 Hz. The NAV440™ also includes a GPS with embedded Wide Area Augmentation System hardware known as WAAS, which is used for accuracy improvement. An external GPS can also be connected on port B to work as a DGPS. In some cases, synchronous satellites that transmit the WAAS signal are difficult to receive at northern latitudes and so for navigation purposes, an external GPS might be needed. However, in the case of the flight test GPS seems to work properly, as is described in section 4.3.

The key feature of the NAV440™ firmware is the implementation of the Extended Kalman Filter (EKF), as shown in figure 4.4. Other measurements such as temperature and magnetometer sensors and the GPS block, are used by the EKF to correct drift errors in position. In addition, proprietary algorithms employ calibration data stored into the DSP EEPROM to correct scale factors, non-linearities and misalignments that affect the performance and accuracy of the IMU. Finally, the embedded GPS uses the standard WGS84 Ellipsoid to provide Latitude, Longitude and Altitude and the velocities are in C_N frame. Table 4.1 summarizes the IMU characteristics and settings used in this work. For more details in the NAV400® operation, see [48].

4.1.2 Servo Switch Control Module

The Servo Switch Controller card, model SSA20024™ from Omni Instruments, is a digital programmable PWM signal selector switch and a PWM signal generator. It is capable to handle up to 24 signals from which 8 of them are defined as outputs and the rest may be programmed as input/output. In addition, the SSC has 6 auxiliary ports which can be programmed as a digital input/output, timer input, pulse accumulator input, timer output and pulse output. The RPM detector IC from Allegro Semiconductors is connected to the first auxiliary port and is used to monitor the main rotor RPM – refer to the electrical diagram in appendix C.1. The main computer, PC/104™, communicates to the SSC through an asynchronous serial port at a rate of 115,200 bps as shown in the block diagram in figure 4.5. This serial link is used by PC/104™ to monitor the signal status like the RPM signal from the main rotor and PWM signals coming from the pilot from the RC receiver. A record of all the commands from the pilot is kept for future analysis in the identification of the helicopter parameters. The main computer also transmits

Table 4.1: IMU Characteristics and Settings Summary.

Description	Digital Range	Conversion Factor	Units	Reference Frame
ϕ	-2^{15} to 2^{15-1}	$360^\circ \div 2^{16}$	$^\circ$	C_i
θ	-2^{15} to 2^{15-1}	$360^\circ \div 2^{16}$	$^\circ$	C_i
ψ (true)	-2^{15} to 2^{15-1}	$360^\circ \div 2^{16}$	$^\circ$	C_i
p	-2^{15} to 2^{15-1}	$1260^\circ \div 2^{16}$	$^\circ$	C_b
q	-2^{15} to 2^{15-1}	$1260^\circ \div 2^{16}$	$^\circ$	C_b
r	-2^{15} to 2^{15-1}	$1260^\circ \div 2^{16}$	$^\circ$	C_b
\ddot{x}_b	-2^{15} to 2^{15-1}	$20 \div 2^{16}$	g	C_b
\ddot{y}_b	-2^{15} to 2^{15-1}	$20 \div 2^{16}$	g	C_b
\ddot{z}_b	-2^{15} to 2^{15-1}	$20 \div 2^{16}$	g	C_b
v_N	-2^{15} to 2^{15-1}	$512 \div 2^{16}$	$\frac{m}{s}$	C_N
v_E	-2^{15} to 2^{15-1}	$512 \div 2^{16}$	$\frac{m}{s}$	C_N
v_D	-2^{15} to 2^{15-1}	$512 \div 2^{16}$	$\frac{m}{s}$	C_N
Longitude	-2^{31} to 2^{31-1}	$360^\circ \div 2^{32}$	$^\circ$	GWS84
Latitude	-2^{31} to 2^{31-1}	$360^\circ \div 2^{32}$	$^\circ$	GWS84
Altitude	-2^{15} to 2^{15-1}	$2^{14} \div 2^{16}$	m	GWS84
Position/Velocity				
Position Accuracy (CEP)	2.5	m	—	
x, y Velocity Accuracy	< 0.4	$\frac{m}{s}$ (rms)	—	
z Velocity Accuracy	< 0.5	$\frac{m}{s}$ (rms)	—	
Heading				
Range	± 180	$^\circ$	—	
Accuracy	< 2.0	$^\circ$	—	
Resolution	< 0.1	$^\circ$	—	
Attitude				
Range: Roll, Pitch	$\pm 180, \pm 90$	$^\circ$	—	
Accuracy	< 0.5	$^\circ$ (rms)	—	
Resolution	< 0.1	$^\circ$ (rms)	—	
Angular Rate				
Range: Roll, Pitch, Yaw	± 200	$\frac{^\circ}{s}$	—	
Scaled factor Accuracy	< 1	%	—	
Resolution	< 0.06	$\frac{^\circ}{s}$	—	
Acceleration				
Range: x / y / z	± 4	mg	—	
Scaled factor Accuracy	< 1	%	—	
Resolution	< 0.6	mg	—	

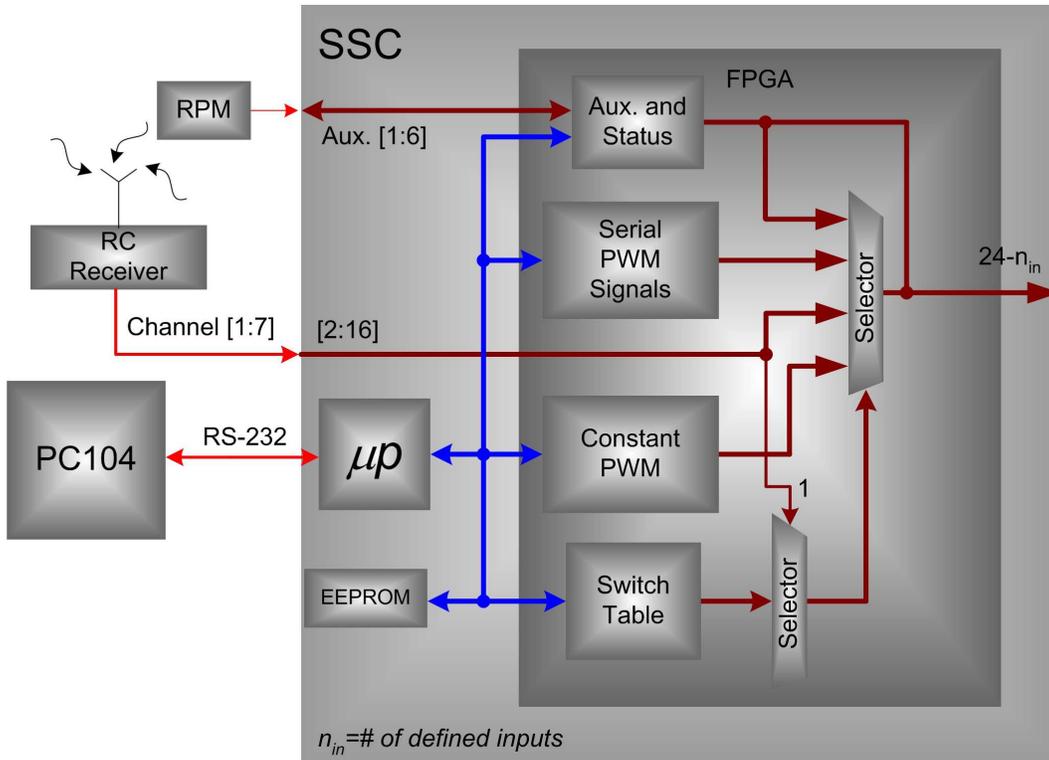


Figure 4.5: *SSC block diagram.*

PWM position command to those channels that have been previously defined as outputs. Port number one on the SSC has been defined as the control switch for autopilot command. Channel 7 from the RC pilot's transmitter, is connected through the RC receiver to port number one on the SSC. In this way, when the pilot toggles the switch of channel 7 on the RC transmitter, the PWM signal coming out from channel 7 on the receiver, changes from 2 ms to 1 ms pulse width. Then the SSC selects the channels that have been previously defined to pick the incoming commands from the serial port and transfers them out to the selected servos. The SSC channel definition table is stored in the SSC EEPROM and needs to be programmed before the normal operation of the SSC module. Table 4.2 shows the switch table settings used for all flight tests.

More details on programming the SSC module are contained in [49].

Table 4.2: *SSC Switch Table Settings used during the test flights.*

Switch Table						
Output Ch	0ms	1ms	1.5ms	2ms	Const	Period
1	Const	Const	Const	Const	1500	20000
2	Const	Const	Const	Const	1500	20000
3	Pin 6	Serial	Pin 6	Pin 6	1500	20000
4	Const	Const	Const	Const	1500	20000
5	Pin 8	Serial	Pin 8	Pin 8	1500	20000
6	Const	Const	Const	Const	1500	20000
7	Pin 9	Serial	Pin 9	Pin 9	1500	20000
8	Const	Const	Const	Const	1500	20000
9	Const	Serial	Pin 5	Pin 5	1500	20000
A	Const	Const	Const	Const	1500	20000
B	Pin 4	Serial	Pin 4	Pin 4	1500	20000
C	Const	Const	Const	Const	1500	20000
D	Pin 3	Serial	Pin 3	Pin 3	1500	20000
E	Const	Const	Const	Const	1500	20000
F	Const	Serial	Last	Pin 2	1500	20000

4.1.3 Communication Module

To get information about the helicopter performance and its operational status during flight, it is important to get important flight variables back to the ground station. Variables such as helicopter attitude, main rotor speed or RPM, battery voltages and currents, and others are monitored on the ground station in near real time. To achieve this goal, two radio modems XStream[®] from Digi[®], are used to get these variables from the avionic box to ground station. This modem provides a RF data link transfer rate up to 20,000 bps, with 100 mW of output power. Using a dipole antenna of 2.1 dB_i of gain, the modems can reach a distance of up to 11 km Line Of Sight (LOS) in an outdoor environment. The RF modems use a BFSK modulation scheme and operates in a frequency band from 902 MHz to 928 MHz. This is particularly convenient when other RF devices, like the RC servo receiver that operates in the band of 2.4 GHz, are present. In addition, the RF modems use a technique called Frequency-Hopping Spread Spectrum (FHSS) and Code Division Multiple Access (CDMA). There are two ideas in the FHSS technique [50], [51]. The first one is to intentionally add several high frequencies in order to increase the bandwidth of the baseband signal, but keeping the same amount

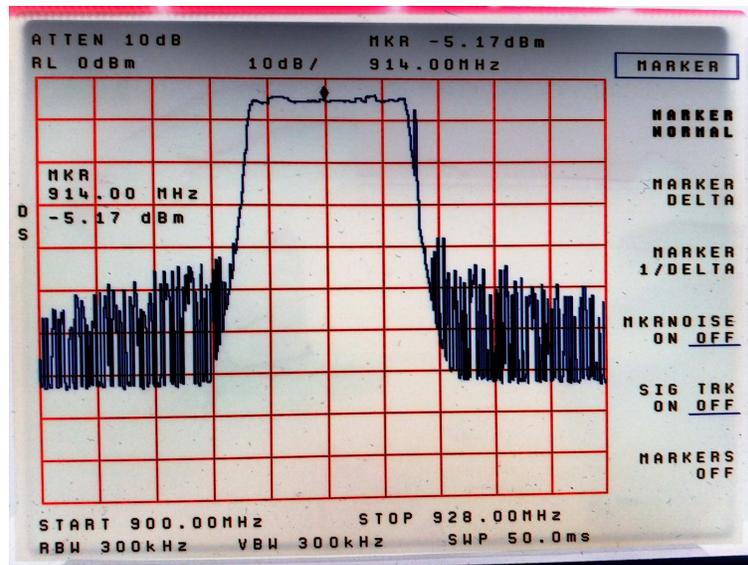


Figure 4.6: XStream[®] RF modems accumulative spectrum.

of energy at the transmitter. As a consequence, the energy is spread over a wider frequency band reducing the S/N ratio level. This action may cause the transmitted signal to appear as noise at the receiver. However, because of the CDMA, only those receivers who have the proper key are capable to despread the signal to recover it. This has the advantages that the transmitted signal is resistant to interference, intentional or unintentional jamming. In addition, the transmitted signal is resistant to fading or multipath effects. For instance, reflections on the RF signal produces interference due to the multiple trajectories. However, because of the despreading process, the first signal arriving at the receiver is synchronized and the others ones are rejected. Frequency-Hopping causes the signal carrier hops from frequency to frequency over a particular bandwidth in a Pseudo Random Number sequence previously established in both radios. The time interval at which frequencies are hopping to different channels is determined by the spread width itself. Figure 4.6 shows the spectrum of the accumulative Frequency-Hopping of the XStream[®] radios. As can be seen, the frequency-hopping goes approximately from 910 MHz to 918 MHz and the central band in this case was about 914 MHz.

The ground station and the ECICH modem, both communicate with the PC/104[™] via RS-232 interface at a rate of 19,200 bps. The ground station modem is a stand alone device allocated in a rugged industrial enclosure, and operates with a 9 V battery. The modem in the avionic box is an OEM version

with a size of 4.06 cm x 7.17 cm and weight of 24 g, which make it suitable for the ECICH box.

An important feature in the RF data link between the helicopter and GS is the communication protocol. Despite all the improvements and high performance of the RF modems, some momentary disruptions in the data link are possible. For example, due to the radiation pattern of dipole antennas, see figure 4.7, is highly probable to get momentary disruptions or breakdowns in the RF data link. This is because the arbitrary helicopter motion results in times when the antennas are oriented in their blind or low energy spots. This can cause breakdowns in communication. The communication protocol must be able to handle those breakdowns and then recover synchronization between the transmitter and receiver. Although, it is possible to reduce the blind spots by setting an antenna array in order to improve the radiation pattern, there still exists probability that synchronization between modems will be lost or there will be communication glitches or breakdowns.

A protocol has been designed so that future upgrades are simple. It has fixed length and consists of two different frames. The frame sent by the ECICH to GS and the frame sent by the GS to ECICH as shown in figure 4.8.

The ECICH-GS frame is 69 bytes long and is organized in the following fields:

- Synchronization
- Payload
- Command status
- CRC
- EOF

Synchronization field consist of two consecutive bytes of hexadecimal characters 0x55. They are used to tell GS that a valid frame has begun.

The payload field is divided in different blocks according to ECICH's instrument's data. The first block in figure 4.9 corresponds to the IMU. It consists of 34 bytes and is the largest block in the payload field. This block starts with

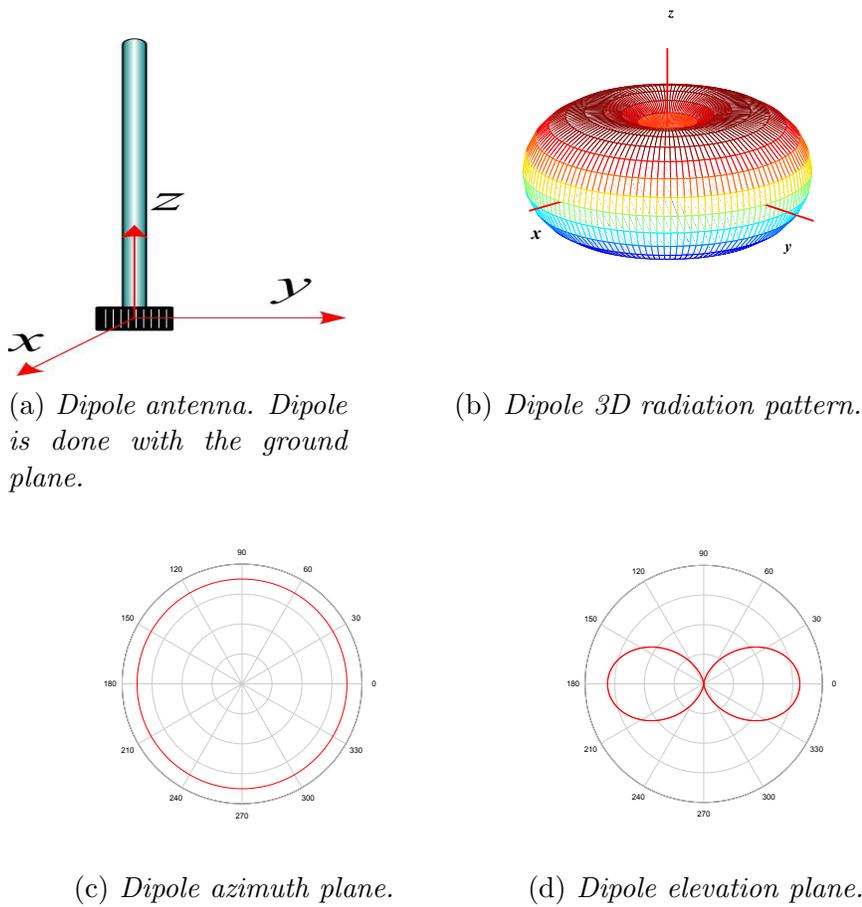


Figure 4.7: Dipole antenna radiation pattern. Due to the dipole nature, the radiation pattern has blind spots in the direction of the z axis.

GS to ECICH frame



ECICH to GS frame



Figure 4.8: Communication protocol general layout.

the roll, pitch and yaw angles followed by their corresponding rates. Then, the body accelerations and velocities in frame C_N are next. Finally, longitude, latitude and altitude close the IMU data Block. The next block is the main rotor RPM data and has a length of two bytes. The Laser Range Finder (LRF) data is the next block with two bytes of length. The Battery Monitor System data is the next block in the payload field and consist of 12 bytes that show the instantaneous voltages and currents of the three different batteries. The first two bytes correspond to the ECICH's battery voltage followed by two bytes for the main rotor's battery voltage, two bytes for the servo's battery voltage, two bytes for the ECICH's battery current, two bytes for the main rotor's battery current and two bytes for the servo's battery current. The next block in the payload field corresponds to the estimated battery charge delivered to the different helicopter's systems. This data block is used by the GS to estimate the remaining charge on the batteries. The first four bytes are the estimated charge of the ECICH's battery followed by four bytes of the main rotor battery charge and four bytes for the servo's battery charge.

The command Status field has a length of one byte and allows the ECICH to

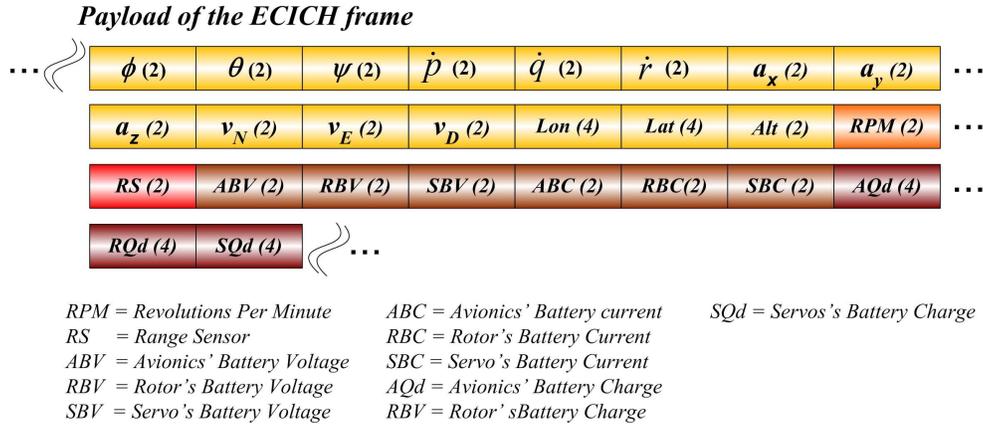


Figure 4.9: Payload blocks of the ECICH frame.

indicate the execution status of the avionic box. This field could be modified to include future commands status or different command status structure.

The Checksum field consist of two bytes and is computed using a 16 bits version of the Adler-32 checksum algorithm, given by the following expression [52], [53].

$$CKS = \left[\left(n + \sum_{k=1}^n (n+1-k) D_k \right) \bmod a \right] 2^8 + \left(1 + \sum_{k=1}^n D_k \right) \bmod a \quad (4.1)$$

where D_k is the frame data vector, $n = 67$ is the number of byte used to compute the checksum in the frame excluding the two bytes of the checksum itself, and $a = \max\{\text{prime}(2^8)\}$ is the maximum prime number between 1 and 2^8 . In this way, the two bytes on the CRC field represented in ASCII format are computed as:

$$CKSH = \left\lfloor \frac{CKS}{2^8} \right\rfloor \quad (4.2)$$

$$CKSL = CKS \bmod 2^8$$

where the function $\lfloor \cdot \rfloor$ in the equation (4.2) denotes the floor function.

Finally, the last field is just the end of the frame, and is composed by two bytes, the line-feed, 0x0A in hexadecimal, and the carriage-return, 0x0D also in hexadecimal.

The GS frame contains the following fields:

- Synchronization
- Command ID
- Payload
- CRC
- EOF

where the synchronization field is three bytes long with the hexadecimal (Hex) character 0x55. As in the ECICH frame, this field serves to tell the ECICH that a valid frame has begun. The next field is the command ID and is one byte long. Here is where the GS sends commands to the ECICH to be executed. With one byte, the GS is capable to send up to 255 different commands like, set initial states conditions \mathbf{x}_o , changes controller's gains or changes between controllers, go to the next way-point, etc. Five commands have been implemented:

1. Request data. Hex character 0x06
2. Start recording data. Hex character 0x52
3. Stop recording data. Hex character 0x53
4. Upload controller's gains. Hex character 0x47

5. Set Initial states. Hex character 0x49

Every single time that the GS needs data from helicopter, the Request data command is executed. In this way the GS is in total control of the communication. Start recording, and Stop recording data, are commands to instruct the ECICH when to record data into the onboard CF memory. This is a useful command because some parts of the flight do not need to be recorded, as it will be described in section 4.1.7, saving valuable space in the CF memory. The Upload controller's gains is an important command during the real flight for tuning gains. This allows a quick way to setup the controller and to tune gains during flight. Finally the Set initial state conditions allows the ECICH to copy the initial states before the takeoff phase.

The Payload is the next field and is variable length. The length of this field changes according to the sent command on the command ID field. This field is used in conjunction with the command ID field to send appropriate data. For example, the initial conditions of state vector \mathbf{x}_o are sent to change between controllers. The penultimate field is the CRC checksum field. It is two bytes long and uses the same reduced version of the Adler-32 checksum algorithm described in equations (4.1) and (4.2).

Finally the last field is the EOF and is exactly the same as the ECICH's EOF field.

Table 4.3 summarize the communication protocol from ECICH to GS, and table 4.4 shows the frame protocol from GS to ECICH.

During data acquisition flight tests, this communication protocol has performed well. In all the five different flight tests, communication has been reliable and data is properly transferred between the helicopter and the GS in both directions. Although in some parts of flight three and four there are momentary breakdowns in the data link, the protocol was able to recover synchronization and continued requesting data from the ECICH and transferred them properly to the GS.

4.1.4 Voltage and Current Monitor Module

From the electrical point of view, and because the Air Star Evolution is an electrical powered helicopter, it is convenient to divide it in three different

Table 4.3: Communication protocol frame from ECICH to GS.

Desc.	B #	1	2	3	4	5	6	7	8	9	10
Field name		Sync	Sync	ϕh	ϕl	θh	θl	ψh	ψl	ρh	ρl
ASCII Value		U	U	nn	nn	nn	nn	nn	nn	nn	nn
Hex Value		0x55	0x55	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY

Desc.	B #	11	12	13	14	15	16	17	18	19	20
Field name		$\dot{q}h$	$\dot{q}l$	$\dot{r}h$	$\dot{r}l$	$a_x h$	$a_x l$	$a_y h$	$a_y l$	$a_z h$	$a_z l$
ASCII Value		nn	nn	nn	nn	nn	nn	nn	nn	nn	nn
Hex Value		0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY

Desc.	B #	21	22	23	24	25	26	27	28	29	30
Field name		$v_N h$	$v_N l$	$v_E h$	$v_E l$	$v_D h$	$v_D l$	Louh	Loul	Loh	Lol
ASCII Value		nn	nn	nn	nn	nn	nn	nn	nn	nn	nn
Hex Value		0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY

Desc.	B #	31	32	33	34	35	36	37	38	39	40
Field name		Lauh	Laul	Lah	Lal	Alh	All	RPMh	RPMl	RS h	RS l
ASCII Value		nn									
Hex Value		0xYY									

Desc.	B #	41	42	43	44	45	46	47	48	49	50
Field name		CBVh	CBVl	RBVh	RBVl	SBVh	SBVl	ABCh	ABCl	RBCh	RBCl
ASCII Value		nn									
Hex Value		0xYY									

Desc.	B #	51	52	53	54	55	56	57	58	59	60
Field name		SBVh	SBVl	AQduh	AQdul	AQdh	AQdl	RQduh	RQdul	RQdh	RQdl
ASCII Value		nn	nn	nn	nn	nn	nn	nn	nn	nn	nn
Hex Value		0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY

Desc.	B #	61	62	63	64	65	66	67	68	69
Field name		SQduh	SQdul	SQdh	SQdl	RecS	CRCh	CRCl	LF	CR
ASCII Value		nn	nn	nn	nn	nn	nn	nn	10	13
Hex Value		0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0xYY	0x0A	0x0D

Table 4.4: Communication protocol frame from GS to ECICH.

Desc.	Byte No.	1	2	3	4	5	6	7
Field name		Sync	ID	PL	CRCh	CRCl	LF	CR
ASCII Value		U	Comm	nn	nn	nn	10	13
Hex Value		0xYY	0xYY	0xYY	0xYY	0xYY	0x0A	0x0D

Nomenclature	Hex	Decimal	ASCII
Sync = Synchronization Character	0x55	85	U
ID = Command Identifier	0xYY	ZZ	nn
CRCh = Checksum	0xYY	ZZ	nn
PL = Pay load	0xYY	ZZ	nn
LF = Line feed	0x0A	10	LF
CR = Carriage return	0x0D	13	CR

ID Command Description	Hex	Decimal	ASCII
Acknowledge	0x06	06	ACK
Data Start Recording on Helicopter	0x52	82	R
Data Stop Recording	0x53	83	S

electrical load systems, each with their own independent batteries as follows:

- Main rotor motor
- Servo motors
- ECICH or Avionic box

The main rotor system is powered by a Plettemberg[®] three phase brushless inrunner DC motor model HP370/40/A2[™]. This motor can operate, according to the manufacturer's data sheet [54], over a voltage range of 20.0 V to 36.3 V, with a maximum rotational speed of 15,000 RPM, and capable to generate over 14.5 N·cm to 125.9 N·cm of torque. The sink current depends on the main blade aerodynamic drag force, applied voltage and RPM speed, and may go from 10.0 A up to 64.4 A or more. From electrical point of view, the main rotor motor is the most noisy source in the system due to its high inductive load and its high peak current demands.

In order to translate the incoming PWM signal from the RC receiver into rotational speed, an Electronic Speed Controller (ESC) is used. The Future 40.160H[™] from Schulze[®] is an ESC capable to sink up to 160 A of continuous current for shorts period of time [55]. To operate properly, the ESC requires that the battery voltage never drop below 32 V, otherwise the motor will be automatically shutdown by the ESC. Due to the high current and voltage constraints, the battery must be carefully selected and the BMS must be carefully designed.

The servo system is a set of 5 servos. One of them, the Futaba S9256[®] is used to control the pitch angle of the tail rotor. It works over a voltage of 4.8 V and the instantaneous sink current depends on the load demands, given fundamentally by the aerodynamic drag exerted by the tail rotor blades. The last four, Futaba S9551[®] servos are used to control the cyclic and collective pitch angle of the main blades through a 3 DOF swashplate mechanism. They operate over a voltage range of 4 V to 6 V, and similar to the tail rotor case, the instantaneous sink current depends on the load conditions. The swashplate servos should operate at the maximum allowable voltage to reduce their operational currents. In other words, for a given load torque, increasing the voltage keeps the current low at the same amount of power demanded by the

load, rather than reducing the voltage and increasing the current. However, because the tail servo operates at 4.8 V, it was decided to operate all servos at a voltage of 5 V in order to use only one DC to DC step down converter, as will be explained on.

The third and last electrical load is the ECICH or avionic box itself. As shown in figure 4.1 it contains all the instruments, sensors, μ controller and computer to process data and perform all the control actions. The various instruments operate at different voltages ranging from 5 V to 30 V. After some experimental testing, it is determined that all instruments and sensors operate in normal and safe conditions with voltage range of 9–12 V, consuming about 3 A of average current. Clearly the power supply that feeds the ECICH must be reliable while minimizing the electric noise.

In order to design and integrate the proper power supply systems and the BMS, the following factors were considered:

1. Flight time, battery life and payload capacity.

A tradeoff which is particularly important for UAVs and UAS is between the battery charge capacity, UAV's payload capacity, available space and mission's time duration. For example, from the payload point of view, it would be ideal to use one large capacity, low weight and low size battery to feed all the electrical systems. However, such a battery is difficult or impossible to get due to budget and/or technological limitations. In the case of the Air Star Evolution helicopter, and because the flight test are done and will be done without the helicopter canopy, the space limitation is not the most restrictive parameter. The main constraints comes from the payload capacity. According to the manufacturer, this type of helicopter is capable to handle a payload about 7 kg without compromising its frame structure. Taking this into account, the search for optimal battery is focused on its charge capacity and weight. Therefore, in this case, the flight duration is determine by the tradeoff between payload, ESC voltage requirements and charge capacity.

2. Electrical induced noise to the ECICH.

From the ECICH point of view, it is important to have a smooth and clean voltage source, to assume continuous and safe operation of the avionic box. For this reason, it is essential to have an independent power

supply just for the avionic box. For example, if noise is generated by the inductive loads, like the main rotor or the servo motors, will be eliminated from the ground path trajectories and the effects of high negative currents peaks over the non-zero ground impedance will not change the avionic box voltage. This is due to the fact that the other loads are not sharing the same main ground. If each load has its own independent battery, each load will have its own main ground path trajectory, although the grounds between batteries must be tied to equalize the potentials or bring the same ground reference among them.

3. Power supply failures.

Clearly all the electrical systems in the helicopter are critical. The ECICH and the servo system are particular critical. As was mentioned before. If the ECICH fails, the helicopter eventually will crash. Furthermore, if one servo motor is not operational, the helicopter cannot be controlled properly and it may crash. However, if the main rotor battery, the ESC or other mechanical component in the main rotor system like the rubber belt, that transfers motion from the motor to the hub, fails, it is possible to land the helicopter safely or with a minimum damage. This is only possible if each electric system has its own battery. In this way any malfunction on the main rotor electrical system will not be reflected on the other electric loads.

Based on this, it was decided to design the system so that each of the three electrical systems has an independent battery.

Currently, the most reliable batteries in the RC helicopter market are Lithium-Polymer batteries (known as LiPo batteries). They can be manufactured in different shapes, sizes and charge capacities. The most common and popular LiPo batteries available on the market are listed in table 4.5.

Two LiPo batteries of 22.2 V and 5000 mAh of charge capacity connected in series are used for the main rotor electrical system. One 11.1 V and 3200 mAh capacity LiPo battery for the Servo Motors system and the RC receiver, and one 11.1 V and 5000 mAh capacity LiPo battery for the ECICH. The battery weights are $(2 \times 0.857 \text{ kg})$ 1.715 kg, 0.283 kg and 0.409 kg for the main rotor, servos and ECICH respectively, making a total weight of 2.407 kg .

Table 4.5: *Most common commercial LiPo batteries available in the market. Source:[56].*

Voltage [V]	No. Cells	Charge capacity [mAh]	Weight [g]
3.7	1S	5400 to 6000	133 to 142
7.4	2S	800 to 5000	134 to 295
11.1	3S	800 to 5000	198 to 432
14.8	4S	2200 to 5000	245 to 566
18.5	5S	2200 to 5000	320 to 703
22.2	6S	2200 to 5000	373 to 838
29.6	8S	4000 to 8000	600 to 1000
37.0	10S	4000 to 8000	1116 to 1425
44.4	12S	4000 to 8000	1326 to 1698

The Battery Monitor System must perform four different tasks:

1. Continuously monitor three different voltages:
 - (a) Avionic box voltage. Up to 11.1 V.
 - (b) Main Rotor voltage. Up to 44.4 V.
 - (c) Servo motor voltage. Up to 11.1 V.
2. Continuously monitor their corresponding currents.
3. Estimate the delivered charge to each system base on their measured current.
4. Transfer all data upon main computer requests.

To achieve these goals, an 8 bit μ controller with embedded 10 bit A/D from is used. A block diagram of the BMS is shown in figure 4.10.

Because all the signals coming out from the attenuator are unipolar, the voltage monitor ranges are set at 12 V, 50 V, and 12 V for the avionic box, main rotor and servos respectively. Three voltage divisors are used to step down all the voltages at a range from 0 V to 5 V. The corresponding attenuation factors are $k_a = 0.41\bar{6}$, $k_r = 0.1$ and $k_s = 0.41\bar{6}$ respectively, and the corresponding voltage resolutions were given by:

$$\Delta v_a = \Delta v_s = \frac{5}{k_a 2^{10}} = 0.0117\text{V}$$

$$\Delta v_r = \frac{5}{k_r 2^{10}} = 0.0488\text{V}$$

During preliminary testing prior to helicopter's flight as well as during the flight tests, these voltage resolutions are found to be sufficient. This is shown on the

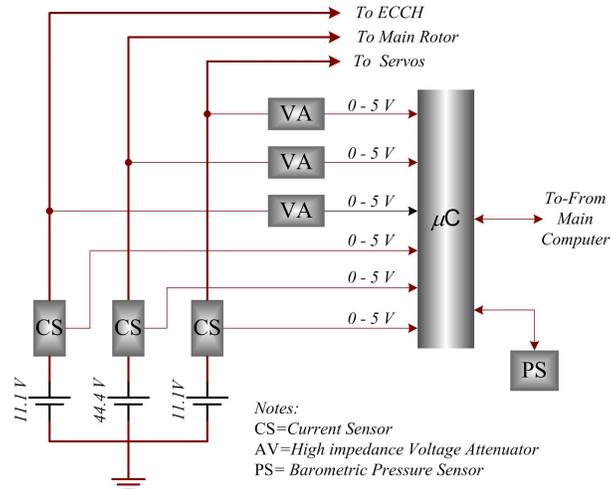


Figure 4.10: Battery Monitor System Block Diagram.

voltage plots of figure 4.11. However, from the Avionics and Rotor Battery Voltage plots on figure 4.11, as well as in those on figure C.4 to figure C.8 in appendix C, some outlier voltages are clearly visible. These outliers might be due to data transmission glitches from ECICH to GS or data corruption in the data link transmission that the protocol is unable to detect. It is assumed that those outliers are measurement errors and do not reflect the real current or voltage values. Future tests are needed to exactly diagnose the cause of these outliers.

For the current monitor system three hall-effect current sensor from Allegro MicroSystem, Inc. are used. Previous current measurements on the ECICH show that the average current is about 3.0 A with peaks of 3.5 A during the boot phase. For this reason, a 5 A current sensor is selected for the ECICH. For the servo motors system, previous measurement shows that the current never exceeded 3 A when testing without any mechanical load. As it is difficult a priori to estimate the maximum current with complete mechanical load in real flight, a 20 A sensor is selected. Finally for the main rotor, and looking at Petenberg[®] tables, the maximum reported current is about 65 A. Therefore, a 200 A current sensor is selected to have a enough range to cover all possible values.

Figure 4.12 shows the voltage-current characteristics for each sensor. As shown in these plots, it is necessary to amplify the output signals from the sensor to adjust them to the full voltage range from 0 to 5 V required by A/D

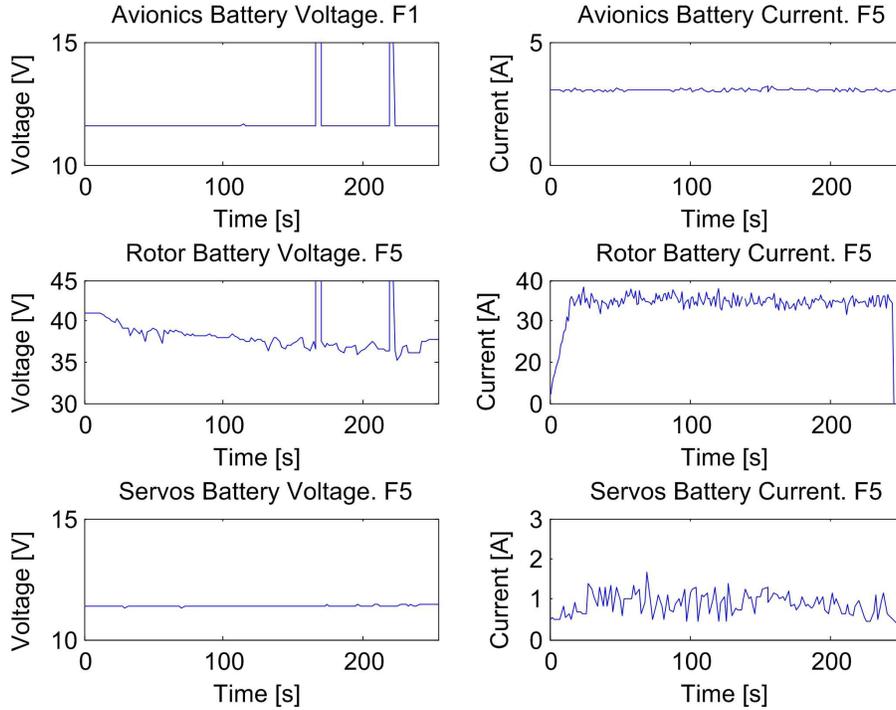


Figure 4.11: Voltages and current plot from flight No. 5. From left to right and top to bottom, ECICH's battery voltage and current, main rotor's battery voltage and current and Servos' battery voltage and current.

converter. A detail electric diagram is shown in section C.1 of appendix C –see figures C.1 to C.3

The current readings during the real flight are within the predicted range as shown in figure 4.11 and figures C.4 to figure C.8 in appendix C. The currents for the avionic box and servos never exceeded 5 A. In the case of the main rotor, the current reading shows a current average of about 33 A. These results indicate that the 200 A sensor can be replaced with a 100 A sensor. Figures C.4 to C.8 on appendix C, also have some outliers in the current plots.

To estimate the amount of charge delivered by the batteries to the loads, the instantaneous current is integrated over time. The instantaneous current flowing through a conductor can be computed as:

$$i(t) = \frac{dQ}{dt} \quad (4.3)$$

where dQ is the charge differential. Integrating the charge in equation (4.3) as

$$Q = Q_{B_0} + \int_0^t i(\tau) d\tau \quad (4.4)$$

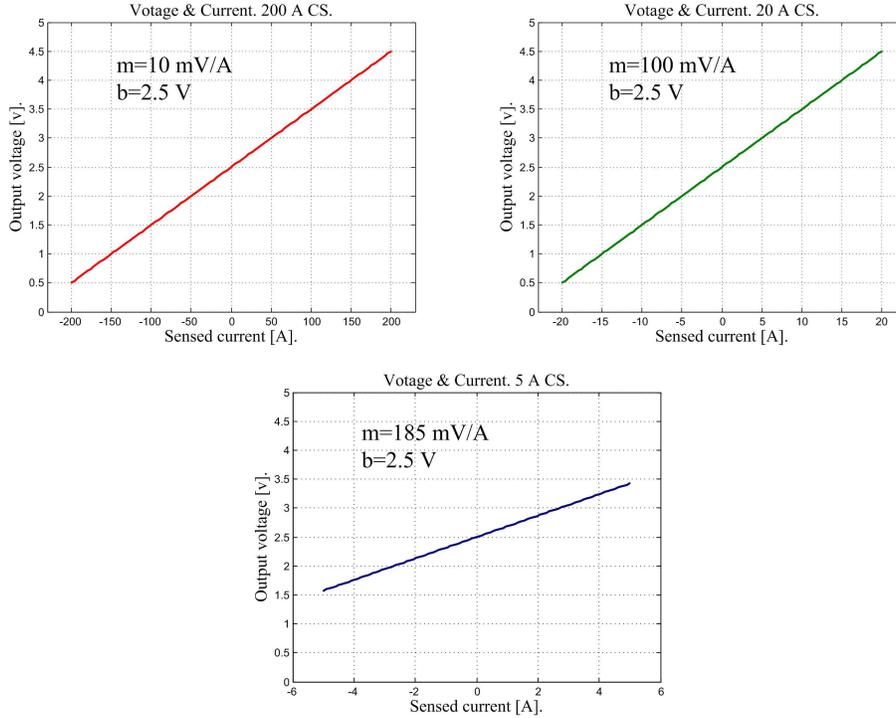


Figure 4.12: Voltage Vs. Current Characteristics of the three current sensors. From top left to right to bottom, 200 A, 20 A, and 5 A current sensors characteristics.

were Q_{B_o} is the initial estimated battery charge. To obtain an accurate estimate of the charge, the current from the three sensors needs to be integrated in real time. To achieve this goal the μC sampled the currents at a sampling rate of 4 ms, and numerically integrates the current using the trapezoidal method. All the sensor currents are integrated as soon as A/D converter has a valid data. The discrete integration is done according to the following expression:

$$Q_{x_m} = T_s \frac{i_{k-1} + i_k}{2} \quad (4.5)$$

where i_k and i_{k-1} are the actual and the previous current values, T_s is the sampling rate, and Q_{x_m} is the corresponding charge in Coulombs. Therefore, the computed charge Q_{x_m} is added to the previous one, in order to get the total or accumulated charge at a given time, that is:

$$\hat{Q}_x = \hat{Q}_x + Q_{x_m} \quad (4.6)$$

where \hat{Q}_x is the estimated accumulated charge at given time for each of the three batteries. These three estimated charges \hat{Q}_a , \hat{Q}_r , and \hat{Q}_s are stored in a

32 bit long variables on the μC and transmitted to the main computer upon request. Once the accumulated estimated charges are transmitted down by the PC/104™ to GS, the program on the GS subtract them from the initial estimated charge on each battery, that is:

$$Q_{B_{rem}} = \hat{Q}_{B_o} - \hat{Q}_x \quad (4.7)$$

This process is described in detail on section 4.2. At the beginning of the program execution, all the three variables \hat{Q}_x are initialized to zero. Detailed operation of the μC program is in section C.3 of appendix C.

Finally, the last task of the μC is to transfer data to the onboard main computer. As in the case of transmission between ECICH and GS, there is a protocol between the PC/104™ and the μC . Figure 4.13 shows the fields of both frames, the PC/104™ frame and the μC frame. As can be seen in figure

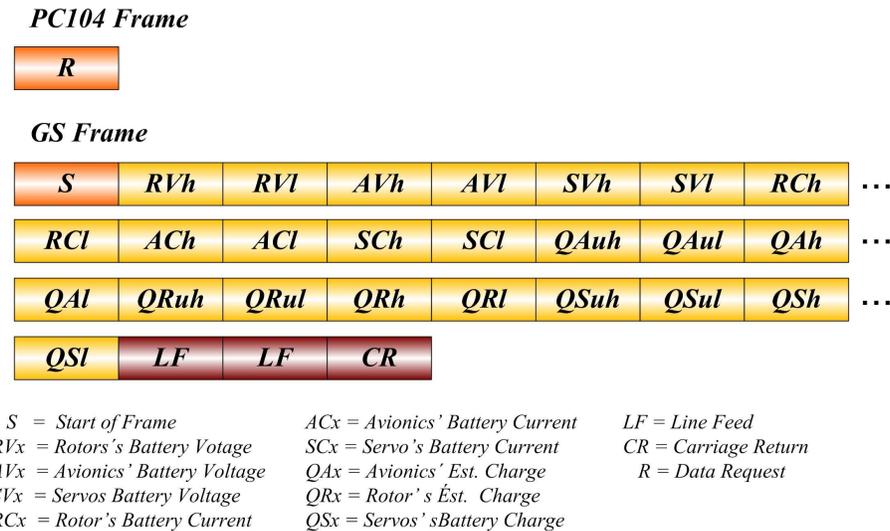


Figure 4.13: Protocol frames between GS and μC .

4.13 the frame transmitted from the PC/104™ to the μC is composed by the data request field, Hex character 0x52. This tells the μC that data need to be transferred to PC/104™. Because the communication between PC/104™ and μC takes place directly on the mother board, a CRC or Check Sum is not implemented.

The frame from μC to PC/104™ is 28 bytes long and is constituted by the following fields:

- Start field
- Payload
- EOF

The start field on the μC frame is one byte long and character 0x53 Hex is used. This tells to PC/104™ that valid frame has begun and the requested data have started to be transmitted.

The next field corresponds to the payload and is formed by 24 bytes. The main rotor's battery voltage followed by the ECICH's battery voltage and the servo's battery voltage are each respectively two bytes. Next two bytes, bytes No. 8 and 9, are the rotor's battery current, followed by the avionics' battery current and the servo's battery current. Finally the estimated charge delivered by the three batteries are the next fields. The first four bytes of this section are the charge delivered by the ECICH's battery, next are the estimated charge of the battery of the main rotor and the last four ones correspond to the estimated charge delivered by the servos' battery.

The last fields on the μC frame is the EOF. It is formed by two consecutive LF characters, Hex 0x0A, and one carriage return character, Hex 0x0D. Table 4.6 details the protocol frame between PC/104™ and GS.

Table 4.6: *Communication protocol frames between PC/104™ and GS.*

PC/104™ Data Request										
Desc.	B #									
Field name	DRQ									
ASCII Value	R									
Hex Value	0x52									

BMS Response. μC to PC/104™											
Desc.	B #	1	2	3	4	5	6	7	8	9	10
Field name	Start	RVh	RVl	AVh	AVl	SVh	SVl	RCh	RCl	ACH	
ASCII Value	S	nn									
Hex Value	0x53	0xYY									

Desc.	B #	11	12	13	14	15	16	17	18	19	20
Field name	ACl	SCh	SCl	QAuh	QAul	QAh	QAl	QRuh	QRul	QRh	
ASCII Value	S	nn									
Hex Value	0x53	0xYY									

Desc.	B #	21	22	23	24	25	26	27	28
Field name	QRl	QSuh	QSul	QSh	QSl	LF	LF	CR	
ASCII Value	nn	nn	nn	nn	nn	10	10	13	
Hex Value	0xYY	0xYY	0xYY	0xYY	0xYY	0x0A	0x0A	0x0D	

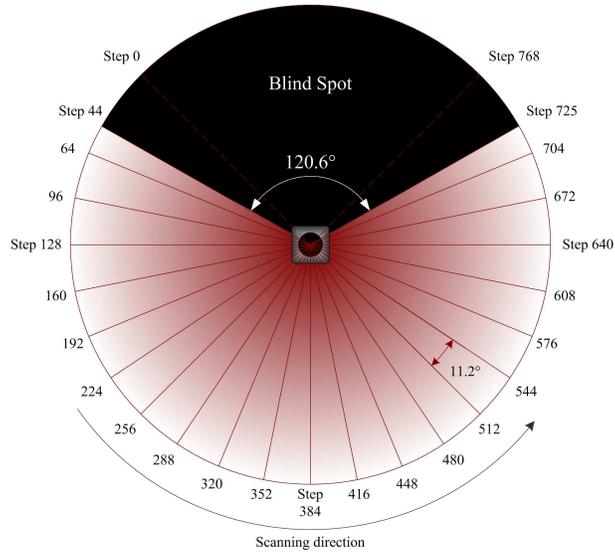


Figure 4.14: *Range Sensor Scanning Area*

4.1.5 Laser Range Finder Module

The ECICH incorporates a module to measure the distance of the helicopter from the ground. This is during the takeoff and landing phase. The scanning Laser Range Finder (LRF), model URG-04LX™ from Hokuyo Automatic Co. LTD is used. This device uses an infrared Laser diode with a 785 nm wavelength with a range of 4 m in the lab. The laser scans the surrounding area over angles of about 240°. The step scanning resolution is about $\frac{360^\circ}{1024} \approx 0.36^\circ$. Figure 4.14 shows the scanning patterns and the blind spots. The scanning steps are grouped in sectors or clusters. The size of clusters and the scanning area can be programmed dynamically from the serial port using a specific command. The LRF communicates to PC/104™ through a RS-232 interface at a rate of 115,200 bps. The measured distance is given in millimeters and the maximum measured distance is $2^{12} - 1 = 4095$ mm with a resolution of ± 1 mm according to its manufacturer [57]. The twelve bits are broken down into two sextets of bits. A value of 30 Hex is added to each sextet and sent out in the LRF data frame upon PC/104™ request.

In most cases the helicopter will land in a relative flat and free area, that is free of obstacles the angular scanning range of the LRF can be reduced. After some lab tests, it is determined that an angle of 11.25° is sufficient. This angle correspond to 32 laser beams scanning steps, (steps 368 to 400) organized in group of 64. Since the distance measured for every 64 laser beams steps is

grouped, the 32 measurements in our case are reported as one single measure, which is the minimum one of all of them. In addition, a compensation for pitch angle, according to the following expression:

$$R_g = R_L \cos(\theta)$$

is performed, where R_g is the distance from helicopter to ground, R_L is the LRF measured distance and θ is the helicopter's pitch angle. Figure 4.15 shows the scanning area used. It is possible that in the future application it might

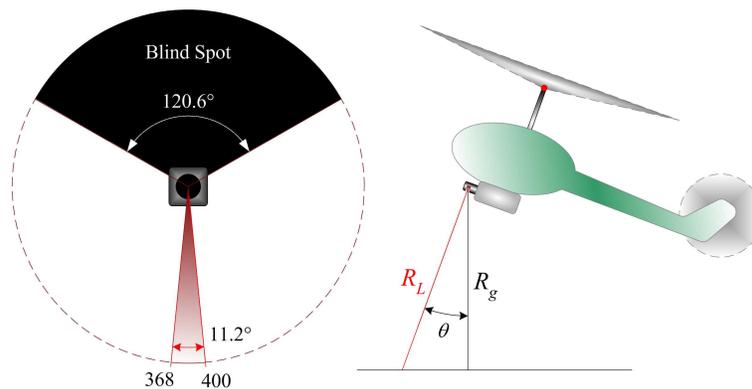


Figure 4.15: *LRF Area Set and Pitch Compensation.*

be necessary to increase this angle.

The Hokuyo Automatic Co, LTD proprietary communication protocol is shown in figure 4.16. This protocol does not have a start of frame field or a CRC or a checksum field. The transmitted frame from PC/104™ to LRF has three fields:

- Command
- Parameters
- EOF

LRF command field is one byte long and accepts several commands like information and settings commands. Nevertheless, the only command used by the PC/014™ is the measure distance command which is command character 0x47 Hex.

The parameter field length depends on the type of transmitted command. For the measure command, it is 8 bytes long. The first three bytes tells the LRF

at what step the scanning area will begin. The next three bytes contain the end step and the last two bytes indicate how many steps will be grouped and reported as a single measurement. These last two bytes indicated the length of the cluster.

The last field, as shown in figure 4.16, is just **EOF** one byte long and can be either a **LF** or **CR** character.

The frame transmission from LRF to PC/104™ is organized in:

- Echo Back field
- Status field
- Separator field
- Payload field
- **EOF**

Computer Frame



RS Frame

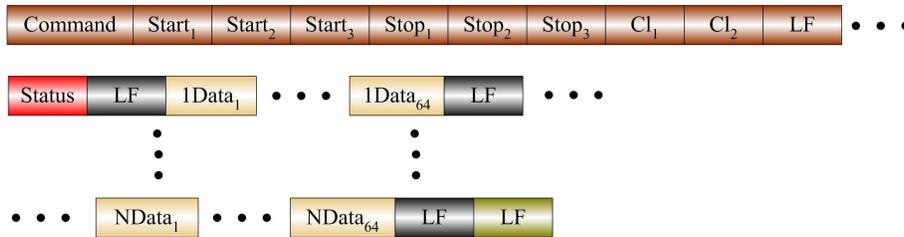


Figure 4.16: *LRF Communication Protocol Frame.*

The echo back field is just that, an echo of the received frame from the PC/104™. This means that the LRF will send back to computer the complete received frame.

The next section is the status field. It contains information regarding errors during the measurement. If status byte is 0x30 **Hex** the measurement is error free. Any other number is an error. For more details in the status field, see [57]. The next field is a separation between the status field and the payload field.

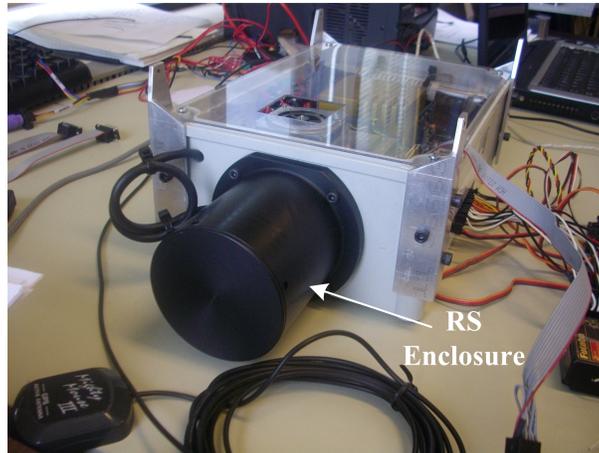


Figure 4.17: *Details of the LRF enclosure attached to the Avionic Box.*

The payload field is variable length and the length depends on the number of clusters assigned by the computer in the cluster field. After the payload field a **LF** character is added. If the payload exceed 64 bytes long, a **LF** character is added at the end of each group of 64 bytes. Finally, the **EOF** is one byte long, with a **LF** character. For more detail on the LRF operation see [57].

The LRF is designed for indoor use, so it is necessary to build a special case to isolate it from as much ambient light as possible. Although this case introduces an extra weight in the payload, this weight is small enough to not compromise the payload limit of the Air Star Evolution helicopter. Details of enclosure are shown in figure 4.17.

During flight test the LRF performs well when the ground surface is asphalt or concrete. However, on grass or dark soil it has some problems measuring the distance. It is desirable that the landing areas be flat and reflective surface for the LRF to work properly.

To increase the performance and reliability to measure the distances from helicopter to ground, it is recommended to use a microwave range finder, like the Miniature Radar Altimeter, or MRA for short, from Roke Manor Research LTD. They manufacture two types of MRA. MRA type 1 can measure distances up to 700 m with ± 50 cm of resolution and the MRA type 2 has a distance range of 100 m and a resolution of ± 2 cm. The weight of these units is about 400 g for MRA type 1 and 375 g for MRA type 2. However, the power consumption is about 3 W with peaks of 7 W for both units. This could be a

factor to consider for sizing the batteries charge capacities. To overcome this problem and save battery charge, the MAR can be turned on by the ECICH during takeoff and landing phase, and turn off during the rest of the flight.

4.1.6 Step Down Voltage Converter

To reduce and simplify connections between battery and servos, as well as minimize poor contacts and possible short circuits, a DC to DC step down voltage power supply is embedded on the ECICH mother board.

Two SDC LM2976 ICs with 5 A of continuous current capacity each with current peaks of 7 A are used. The output voltage is fixed at 5 V while the input voltage varies from 8 V to 40 V. This system has an efficiency greater than 90% due to its switching frequency operation at 256 KHz and its MOSFET low power output resistance which is suitable for RC batteries. In addition, SDC can operate over a wide temperature range from -40°C to $+125^{\circ}\text{C}$ which is ideal for extreme conditions. As was mentioned previously, a 11.1 V with 3200 mAh battery is connected to the SDC inputs. Another important feature is the soft start operation. When the power supply is turned ON, the current increases linearly making the transition from OFF to ON smoother. This reduces the amount of noise and peaks to the systems in the ON procedure. For more detail in the SDC operation see [58].

Located at one flank of the Avionic Box connections to the servos, RPM and ESC have been simplified through the block terminals JPWM, JVcc, JGND, see figure C.1 in appendix C. The total of 10 A current has been divided into two sets of 5 A current load, as shown in figure C.1. Voltage from one SDC is sharing terminal pins on the middle row from 1 to 16 whereas the second SDC shares pins 17 to 33 also on the middle row –see figure 4.18.

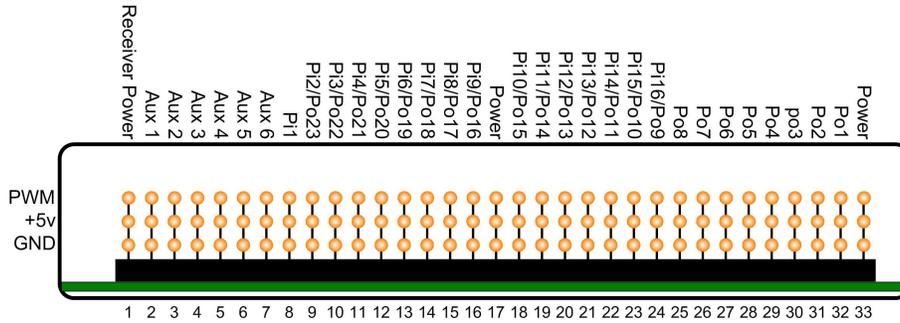


Figure 4.18: Block Terminal Connections for Servos, RPM and ESC.

All terminal pins in the lower row of the block connector are ground where the upper row pins are the PWM signals coming in or out from the ECICH.

The recommended power ON sequence is as follows:

1. Turn On the ECICH and wait until program is running
2. Turn ON the servos and ESC controller
3. Connect the Main rotor battery to the ESC high voltage terminal

Turning ON the ECICH first allows smooth transition to the servos when they are powered ON, avoiding stress in the linkages mechanism and the servos due to wrong PWM information coming from the RC receiver or the SSC.

As an extra feature, there are two additional switches on one side of the ECICH to select the power supply from internal batteries or external power, as well as to recharge batteries without removing them from the enclosure, as can be seen in figure 4.19

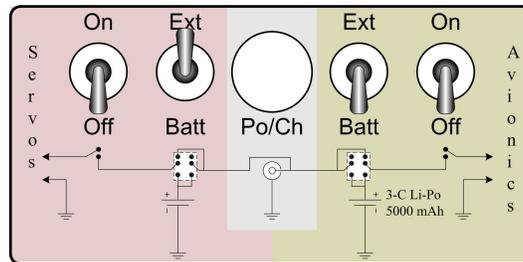


Figure 4.19: ECICH interruptors sets.

Finally, the PCB was manufactured using 2 Oz of copper deposit instead of the standard 1 Oz copper. This was done to get an extra security margin in case of unexpected large current peaks on the servo's power paths that might

damage narrow tracks on the PCB, which could cause power supply system failure, with its catastrophic consequences.

During the flying test the two SDC perform well, validating that they are enough to supply the total amount of current needed by the servos, RC receiver and ESC loads.

4.1.7 Computer Module

The computer module is the core of the ECICH. An industrial rugged embedded pentium computer known as PC/104™ from Advanced Digital Logic™ model ASL855PC™, is used. This system has three stacked circuit boards. It is equipped with a single core Intel® Pentium® processor running at 1.6 GHz, with 1 GB of RAM, one compact flash memory card of 1 GB, two communication serial ports, one PS2 keyboard interface, one VGA interface and other peripherals that are not used. The second circuit board is the power supply model ADFLPS104CF™ from Advanced Digital Logic™. It provides an output voltage of 5 V with 10 A of current capacity with an input voltage from 8 V to 28 V. In addition, this module includes the CF socket to allocate the CF memory. A third layer with eight additional serial ports from Diamond Systems Corporation, model EMERALD-MM-8 is used. This extra serial port allows the main computer to communicate with all the instruments on the ECICH. For more detail on PC/104™ see [59], [60] and [61].

The main computer executes the following functions repetitively in sequence:

1. Request status from the SSC
2. Send commands to servos through the SSC
3. Request attitude and navigation parameters from the IMU
4. Request range to ground from LRF
5. Decoded data from LRF
6. Request data from BMS
7. Assemble communication protocol and transmit data to GS
8. Execute requested commands from GS
9. Store data onto CF memory

10. Execute control actions

To implement all the tasks on the main computer, MATLAB[®] program with Simulink[®] and xPC Target[™] are used. Figure 4.20 shows a block diagram of the Simulink[®] program used for the flight tests, including variable acquisition for identification purpose.

As can be seen from figure 4.20, it is possible to identify twelve blocks that execute all the required functions as follows:

1. *PWM Computation from Servo Switch Card* block
2. *PC104 Controller* block
3. *IMU Data* block
4. *Storage of Range Sensor Altitude Data* block
5. *Servo Switch card Data Request* block
6. *Crossbow Data Request* block
7. *Range sensor Request* block
8. *A/D Data Request* block
9. *Ground Station Data Transmission* block
10. *Voltage Current Monitor* block
11. *Range Computation* block
12. *GE Command Execution* block

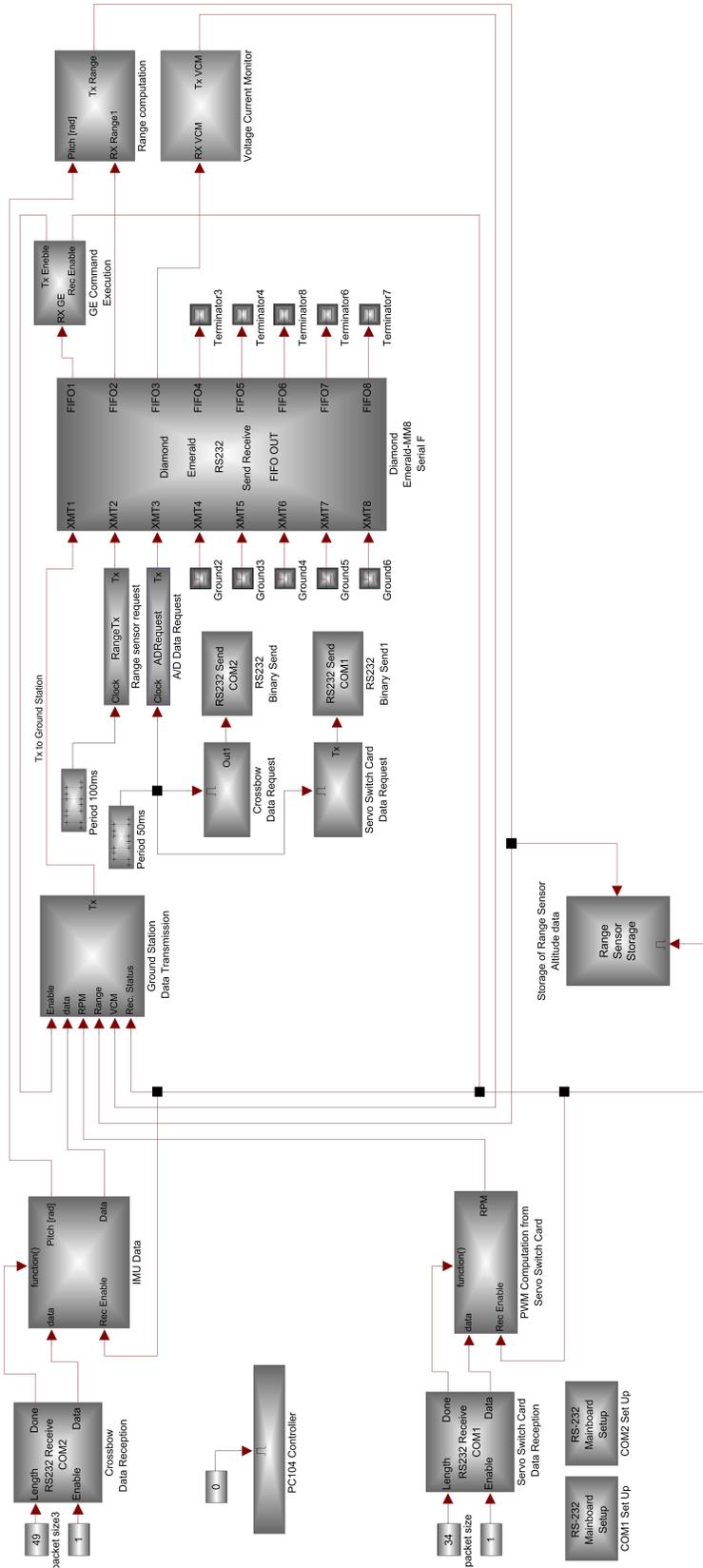


Figure 4.20: PC104 Program Block Diagram

To request status from SSC and send commands to servos through SSC, main computer uses serial port COM1. Through this port, computer gets information of the **PWM** signals coming from the pilot through the RC receiver and the main rotor speed coming from the RPM sensor located underneath of the hub gear. These signals are recorded on the CF for further analysis. Through the same port the main computer sends commands that are translated into **PWM** signals by the SSC. In this way, computer controls the helicopter by actuating the servos that are selected by the assignation table stored on the SSC **EEPROM** memory, (see table 4.2) as performed blocks 1, 2 and 5 are used to accomplish these tasks.

Block number 5 is the block that requests status data to the SSC. There are two frames on this block as shown in figure C.13 of appendix C. One frame, the one that has the command ID 13, is to request the current value of the pulse width of the **PWM** signal. These are the commands to servos coming from the pilot. The second frame, ID 14, is to request information about the auxiliary ports regarding the main rotor speed provided by the RPM sensor. That information is received and processed by block number 1 –see figure 4.20 and figure C.9. Once the RPM and the **PWM** width are computed, they are recorded in the CF memory by the *Pilot Command Storage* block –see figure C.10. Details on the RPM computation can be found on the program list C.1 in appendix C.

In block number 2 several control routines will be implemented. As a preliminary test, some random and sinusoidal signals are fed to the servos, –see figure C.11. Before signals can be transmitted, it is necessary to compute the Fletcher's Check Sum and assemble the corresponding frame in the program block *Check Sum Computation and serial Packet Transmission to Servo Controller* –see figure C.12. Details of the Fletcher Check Sum is in the program list C.2.

Blocks number 6 and 3, see figure 4.20, are the blocks related to the IMU data request and data processing respectively. The frame on the *Crossbow Data Request* block, see figure C.16, is formed by a preamble field, character 0x555 **Hex**, the packet request command, character 0x4750 **Hex**, packet length field, the payload type, which in our case is the payload *N1*, and the **CRC** field. The **CRC** field is computed using the 16 bit **CRC-CCITT** algorithm width 16 bits long format, a polynomial 0x1012 and initial value of 0xFFFF. For more details on the implementation see [48].

Data from IMU are received by block 3, *IMU Data* block. The whole frame is partitioned and each data field is processed according to the code listed in C.3. After each data field is converted into proper units and readable values, they are recorded in the CF memory using the block program *IMU Data Storage*. See figures C.14 and C.15 in appendix C for details.

The LRF is implemented in blocks 7 and 11 of figure 4.20. The range sensor is the only instrument in which data are updated every 100 ms since the scanning spinning motor does not allow faster data acquisition rates. This slower sampling rate did not compromise the helicopter operational safety during the flight tests.

As was described in subsection 4.1.5, block *Range Sensor Request* implements the frame to request the proper scanning pattern to read the distance from helicopter to ground. The scanned area is set to an angle of 11.2° as is described in subsection 4.1.5. This area can be modified dynamically as required, by changing the initial and final step and the number of steps in a cluster on the request frame –see figure 4.16, and program code C.4 for details.

Once data have been requested to LRF, block *Range Computation* in figure 4.20, the inverse process to decode and transform them into two sets of 8 bits is executed. The decoding process is done by splitting the received data into two sets of eight bits and subtracting 0x30 Hex and then merging them back together. Simulink[®] block diagram in figure C.18 and program code C.5 contain the details of the decoding process.

The battery Monitor System (BMS) is contained in block 8 and 10 of figure 4.20. Data from the BMS are requested every 50 ms by block 8, details of the code is in the code listing C.6. Data coming from BMS are received and processed by Simulink[®] block *Voltage Current Monitor* –see figure 4.20. Received data are converted in packets of 16 bits and sent to *Ground Station Data Transmission* block for retransmission to GS. Details of Simulink[®] *Voltage Current Monitor* block are shown in Figure C.19.

Blocks 9 and 12 execute all the tasks to transmit and receive data from GS. Data coming from all instruments are sent to block 9, the *Ground Station Data Transmission*. The data is decoded from their different formats and reassembled according to frame shown in tables 4.3 and 4.4. Block *TXRF* of figure C.20 contains the program that executes the frame assembly and transmission

to GS, as can be seen on code listing C.4.4.

Commands sent by GS are interpreted and executed by block *GE Command Execution* in figure 4.20. Currently, four commands are implemented:

- TX Enabled/Disabled.
- Start/Stop Data Recording.
- Controller gain update.
- Initial conditions setup.

The TX Enable/Disable command, allows GS to control the communication going to and coming from helicopter. When data were not necessary, for example, when the helicopter is flying between two waypoints, the transmission might be disabled or update transmission rate might be reduced to reduce battery usage from the Avionic box.

The Start/Stop Data Recording command is used for the GS to manage the total amount of data saved on the onboard CF memory. During flight tests only certain parts of the flight or particular flight regimes are recorded so that the CF memory capacity is not exceeded.

The controller gain update command is particularly useful to set and update the controller gains at any time during real flight. This saves considerable amount of time during the gain tuning process and also set the platform to change controller gains between different controllers for future applications.

The Initial condition setup command is used to set the current value of the state variables as initial conditions for the controller. This is useful not only for the present controller but also in a scheduling controller scheme.

In the future, further commands like waypoint data coordinate transfer, switch between controllers, etc. could be implemented in this protocol structure.

The electrical connections between the ECICH and the electrical subsystems are shown in figure 4.21.

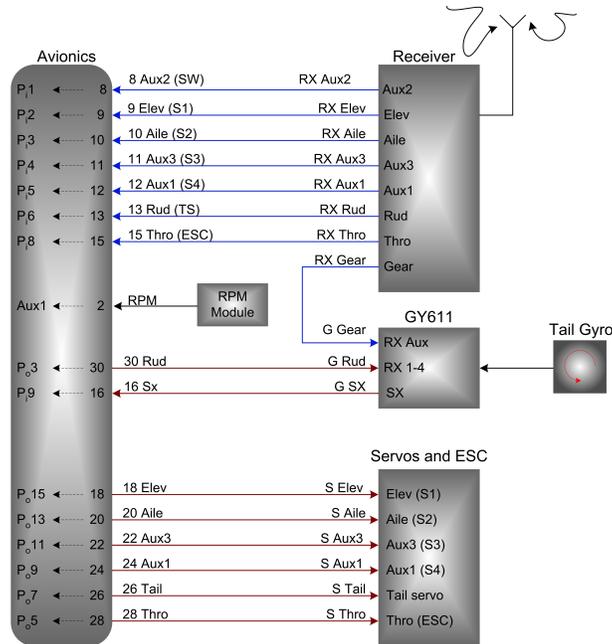


Figure 4.21: Detailed connections between ECICH and Helicopter's Electric systems.

4.2 Ground Station

UAVs-VTOL like helicopters are complex dynamical systems and demand considerable resources. Current State-Of-The-Art systems have high performance and are reliable systems, but they often require constant monitoring. This monitoring is often required even for full autonomous operation.

To meet the requirements for this work, Ground Station (GS) software is developed. The design requirements are:

1. Software must have the necessary instruments to monitor the helicopter and main flight parameters.
2. There must be a set of commands in order to interact with the GS program and helicopter's ECICH system.
3. The GS program must allow the user to fix the initial setting for a proper operation easily.
4. The program must have a data base to store some secondary parameters that are not critical for the ECICH but still important enough for the whole flight performance.

Based on these requirements the GS program is divided into these four sections and subsections:

1. Instruments

- (a) Attitude Director Indicator
- (b) Heading Compass
- (c) Altimeter
- (d) Variometer (RCDI) or Vertical Speed Indicator (VSI)
- (e) LRF range
- (f) RPM indicator
- (g) Voltage monitors
- (h) Current monitors
- (i) Charge estimation monitor

2. Commands

- (a) Start/Stop Requesting data from ECICH
- (b) Start/Stop recording
- (c) Set Ground Station position (Longitude, Latitude and Altitude)
- (d) Set Helicopter local Ground Reference Level
- (e) Display GPS data
- (f) Display IMU data
- (g) Set initial state conditions
- (h) Upload controller's gains

3. Program Settings

- (a) Communication settings
- (b) Battery capacity settings

4. Data Base

- (a) Show data base
- (b) Start/Stop recording on data base



Figure 4.22: GS Main Screen General Layout.

- (c) Export to file data base
- (d) Erase data base

The GS Main Screen general layout is shown in figure 4.22.

The GS Main Screen is divided in seven sections, one status bar and tab bar. Each section has a group of one or more instruments. In the left upper corner there are the instruments related with the altitude and vertical speed, as shown in figure 4.22. The first one at the left of this group is the variometer also known as *Rate of Climb Descent Indicator*, RCDI or *Vertical Speed Indicator*, VSI. This instrument displays the vertical speed given by the IMU, which is the speed on the z_N axis. Its maximum display range is 8 m/s which corresponds to 28.8 km/h. In addition to the graphic scale, there is a numeric display at the bottom of it to facilitate the reading during flight.

The altimeter is to the left. This instrument displays the altitude given by the GPS embedded on the IMU. On the menu bar at the top of the screen, there is a command named *Set Ground Level* which sets the local reference ground. This command should be executed when the helicopter is on the ground in order to subtract the local altitude given by the GPS.

The helicopter to ground distance indicator is next. The maximum range

is just 4 m, as was described on subsection 4.1.5. Attention should be paid to this measurement during takeoff and landing.

The next section is the Attitude Director Indicator (ADI). This virtual instrument displays the pitch, roll and yaw angles given by the IMU. It has a pitch range of 50° above and below the horizon and $\pm 180^\circ$ range of roll or banking angle. In addition, the ADI also displays the yaw angle using a cross lines that move horizontally across the instrument. The position of the helicopter is symbolized by a \wedge open triangle shape that moves at the center of the cross lines. This shapes moves in the opposite direction of the pitch and roll angle depicting the real helicopter attitude with respect to the C_N frame. There are two digital displays with the Pitch and Roll angle. The ADI has an audible alert when the roll angle exceed 60° of banking.

The right upper section instrument the Heading compass. This instrument displays the true yaw angle given by the IMU. In addition, it also displays the magnetic north deviation, which according to [62] is about 15.18° East at latitude of 53.527° North and longitude of 113.52° West (Edmonton Alberta).

The first instrument bottom left is the RPM indicator which displays the main rotor speed coming from the RPM IC located underneath the main gear attached to the rotor shaft. The computation of the RPM is done on the helicopter and sent back to GS. Because the helicopter operates at 1000 RPM, a 60% more from the nominal value of 1000 RPM is set as maximum of the instrument –see figure 4.20.

The next three sections of instruments on the lower half of the screen, are virtually identical and are related to the BMS. The first one on the left displays information about the avionic box voltages, currents and battery charge estimation. The second and third one do exactly the same but for the main rotor system and the servos system respectively. They consist of two virtual gauge instruments, one for the battery voltage and the second for the drained current from the batteries. The third item in each group is a bar shape, that displays the estimated charge given from the ECICH system. At the beginning of the operation, the battery capacity for each system must be initialized. This is done by clicking the tab command *Set Battery Capacity* on the menu bar

located at the top of the program screen, and then enter the three batteries' capacities. When the BMS on the ECICH sends back the estimated charge, the GS program just subtracts the estimate charge to the capacity entered by the user, resulting in an estimate of the remaining charge. The charge displayed on these instrument is just an estimation. Nevertheless, using this instrument in combination with the voltage and current instruments, it is possible to get a reasonable estimate of the remaining charge during the flight tests. All these groups also have three numeric displays, one for the voltage, one for the current and one for the battery capacity.

There are 9 tabs in the menu bar located on top of the GS screen program. These 9 tabs allow the user to configure different parameters and execute specific commands to operate the GS. The tabs are as follows:

1. ComPort
 - (a) Open
 - (b) Close
 - (c) Settings
 - (d) Show Tx, Rx windows
2. Commands
 - (a) Start data request
 - (b) Stop data request
 - (c) Start recording
 - (d) Stop recording
 - (e) Upload controller's gains
 - (f) Set initial conditions
3. Set Batteries Capacities
4. Set Ground Level
5. Set GS Position
6. Display inertial Data
7. Display GPS Data
8. V and C Data Base
 - (a) Start recording
 - (b) Stop Recording
 - (c) Show data base
 - (d) Export to text file (.dat)
 - (e) Erase data base

9. Exit

The ComPort tab allows the user to open and close the communication port. The user can select several com ports from COM1 to COM15. In addition, it is possible to select several transmission standard speeds from 300 bps to 115,200 bps, and set the data transmission parity. It is also possible to display a window showing the transmitted and received data frames for visual inspections and debugging.

The Command tab group has all necessary commands to modify some aspects of the ECICH. The first command allows the GS to control the communication between GS and ECICH. When this command is selected by clicking on this tab, or by pressing F1 key, GS starts to request data packet to ECICH, according to the protocol described in subsection 4.1.3. Immediately after data are received, packed decoded and data properly displayed onto the GS screen, another data packet request is initiated until the command is disabled by pressing the F2 key on the GS.

Star/Stop recording data allows the user to control the total amount of data recorded on the onboard CF memory card. For instance, during the flight test, data are recorded just from the takeoff phase to the landing phase, avoiding the phase priori to takeoff.

When the F5 key is pressed, a dialog window pops up asking for the controller's gains. Once the gains are properly set, by pressing the enter or arrows keys the gains are sent to the ECICH. Once the gains are received by the avionics box, a positive or negative confirmation is sent back to GS indicating whether or not the gains are properly received and loaded into the program.

The last command is used to set the initial values of the states of the controller program.

As was mentioned before Set Batteries Capacities tab is used to set the initial battery capacity for each of the three batteries. When this tab is selected, a dialog window ask the user to enter the batteries capacities in mAh.

The Set Ground Level tab is used to set the local altitude given by the IMU. This value must be initialized when helicopter is landed. The ground reference value is used by the GS in conjunction with the GPS position to compute the distance magnitude between the GS and the helicopter.

Set GPS Station Position tab uses the Helicopter GPS to set the GS local coordinate in latitude, longitude and altitude. This tab must be activated with helicopter just beside the GS to enter a valid GS location.

The Inertial Data tab is used to display data given by the IMU. Here a secondary window is displayed containing numeric format information about Euler angles, rate angles, body accelerations, and North-East-Down velocities.

The GPS Data tab is similar to the Inertial Data Tab but it displays the GPS data embedded in the IMU. Here a secondary window displays in numeric format the latitude, longitude, and altitude data from the IMU.

The next tab is the V and C DataBase tab. When selected a submenu window containing five different command or actions is displayed. The first two commands are the Star/Stop data recording on the database. The Voltages and Currents from the three helicopters' batteries during flight test are recorded on this database. Data in figures 4.12 and C.4 to C.8 are from this database. The third item in the submenu is the View DataBase command. When selected, a secondary window appears showing all the stored data. This allows the total amount of data stored and if data is stored correctly to be visually checked. The fourth command is the Export DataBase to File. When selected, data is exported to a text file with tab separation between columns. This command is used to export data from the database for analysis. The last command in this tab is the Erase DataBase command. Although the database can store a considerable amount of data from flights it is often convenient to just store the data from one single flight.

Finally the last tab corresponds to the Exit tab. Selecting this will close the database and terminate the GS program execution.

The last part of the GS program screen is the status bar located at the bottom of the screen. It is divided in three sections. The first one displays information of the used communication COM port, the transmission speed in bits per second, port parity, status port, open or close, and the number of data frames received.

The next displays information regarding the GS latitude, longitude, and altitude, and the number of registers recorded in the database after the Start Recording command.

Last section of the status bar, contains information about the helicopter's ground reference level and the distance between the helicopter and GS in meters. This distance is computed by GS taking a vector difference between the GS local position on frame C_N and the helicopter current position sent by the ECICH system on the same frame.

The overall time delay between data requests and the delays due to the transmission is found to be 260 ms. The GS program has performed well during preliminary test and during the flight tests. However, if reducing the overall time delay is required, then two things need to be change. First, the current XStream[®] RF modem should be replaced by model XCite[®] to increase the RF throughput. Since the current RF modem has an RF throughput of 20,000 bps (which allows a maximum transfer data rate of 19,200 bps). The whole frame of 69 bytes, is transmitted in 39.93 ms. By increasing the transfer data rate to the same rate as the IMU (57,600 bps), the same frame could be transmitted in about 11.98 ms. However, it is essential not to overload the PC/104[™] by requesting data too often, otherwise the data might often be corrupted or frame synchronization might often be lost.

4.3 ECICH Data Validation

Before ECICH is used for flight tests, data transmitted back to the GS is validated. Validation of the ECICH was done for the following tests:

1. GPS stationary position accuracy
2. Moving position and velocity data on frame C_N
3. RPM data
4. LRF data
5. Voltage and current data

GPS stationary position accuracy is reported as *Circular Error Probable* or CEP –see [63], [64], and [65]. This means that the likelihood of a computed position lies within this circle is 50%. If a given GPS specifies a position accuracy of 2 cm at rest, this means that 50% of computed position data will be inside of this 2 cm diameter circle.

In order to estimate the IMU’s CEP, three sets of position data are acquired at 500 milliseconds time sample. The first set contains data for about 20.26 hours, the second set is 11.15 hours long and the last one is 25.95 hours long. Data are taken on different days with a time gap of several days to minimize possible correlations in the GPS satellite configuration. The IMU is stationary at a fixed location with a clear LOS to the GPS satellite constellation. Figures 4.23(a) to 4.25(a) show the 3D position histogram, in which the joint events occurrence frequency for the three data sets is plotted. Figures 4.23(b) to 4.25(b) exhibit the 2D scatter pattern with its corresponding histograms for each orientation, North-South and East-West, for each of the three data sets. The histograms and the scatter pattern are observed to differ slightly between data sets.

Before analysis, data are processed as follows: position vector ${}^{lla}\mathbf{P}$ containing the Latitude, Longitude and Altitude, are transformed first to frame \mathbf{C}_E and then to frame \mathbf{C}_N according to:

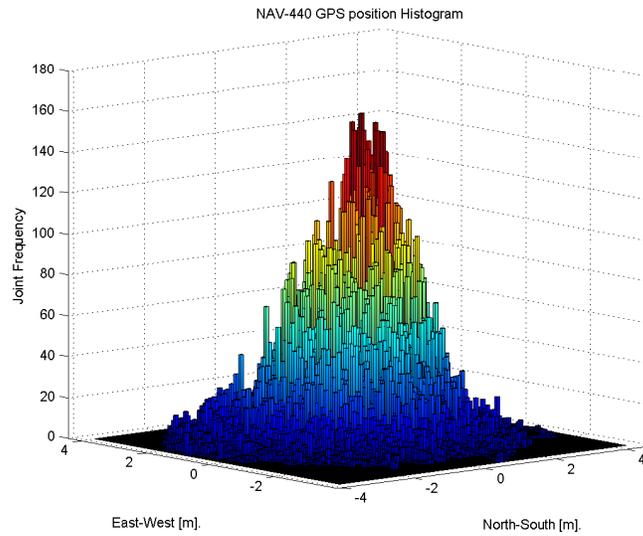
$${}^N\mathbf{P} = \mathbf{R}_E^N \mathbf{R}_{lla}^E {}^{lla}\mathbf{P} \quad (4.8)$$

Then the mean value of each variable on vector ${}^N\mathbf{P}$ is subtracted to get a normalized zero mean random variable –see figures 4.23 to 4.25.

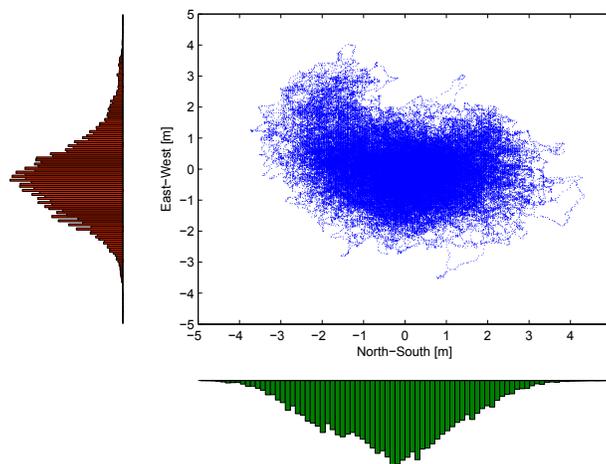
Figures 4.23(b) to 4.25(b) indicate that the corresponding histograms for North-South and East-West position might be approximated by a two variables Normal joint probability density function, known as a Bivariate normal distribution, which is given by:

$$f(x, y) = \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)} \left[\left(\frac{x-\mu_x}{\sigma_x} \right)^2 - 2\rho \left(\frac{x-\mu_x}{\sigma_x} \right) \left(\frac{y-\mu_y}{\sigma_y} \right) + \left(\frac{y-\mu_y}{\sigma_y} \right)^2 \right]} \quad (4.9)$$

where σ_x^2 and σ_y^2 represent the variance of each random variable, ρ is the correlation between them and μ_x and μ_y are the corresponding mean values.

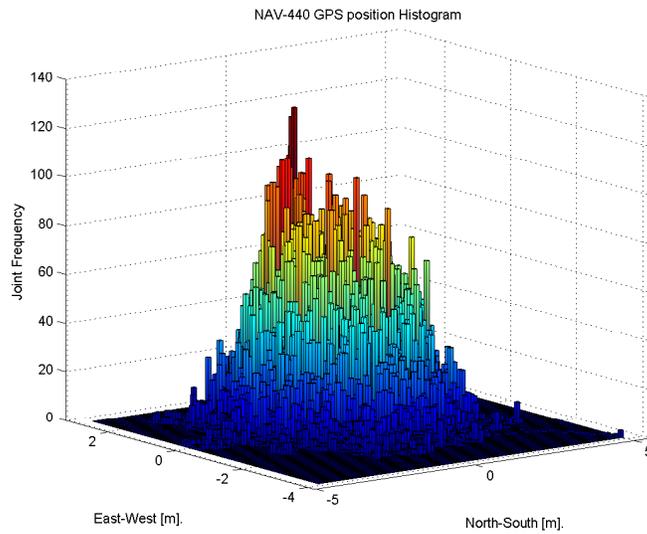


(a) 3D Position Data Histogram

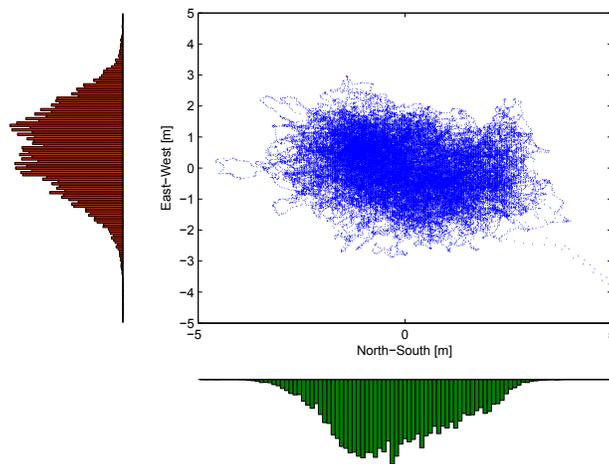


(b) 2D Position Scatter Pattern and N-S, E-W Histograms.

Figure 4.23: First Data Set Position 3D Histogram and 2D Scatter Pattern with its corresponding 2D North-South and East-West Histograms for the NAV440[®] GPS.

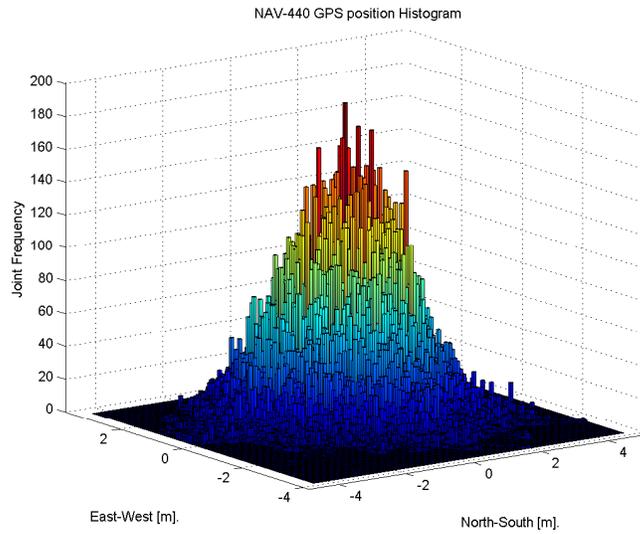


(a) 3D Position Data Histogram

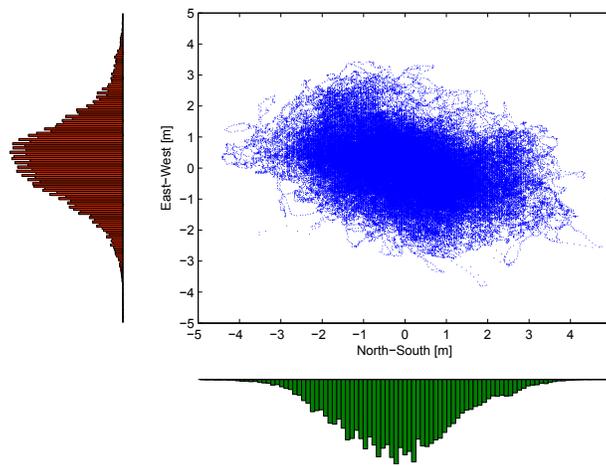


(b) 2D Position Scatter Pattern and N-S, E-W Histograms.

Figure 4.24: Second Data Set Position 3D Histogram and 2D Scatter Pattern with its corresponding 2D Histograms for for the NAV440[®] GPS.



(a) 3D Position Data Histogram



(b) 2D Position Scatter Pattern and N-S, E-W Histograms.

Figure 4.25: Third Data Set Position 3D Histogram and 2D Scatter Pattern with its corresponding 2D Histograms for for the NAV440[®] GPS.

The argument of the exponential function is also known as the Mahalanobis distance and represents a statistical norm, see [66]. This norm might be interpreted, in statistical sense as how far a joint event is from the cluster. In two dimensions case, the Mahalanobis distance describes an ellipse shape in the plane. If data are uncorrelated with equal variance σ^2 and zero mean value, bivariate pdf function reduces to

$$f(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2} \left[\frac{x^2+y^2}{\sigma^2} \right]} \quad (4.10)$$

The argument of the exponent on equation (4.10) represents a circle. The probability that a joint event lies within this circle can be found by integrating equation (4.9) as:

$$P(R \leq D) = \iint_{x^2+y^2 \leq D^2} \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2} \left(\frac{x^2+y^2}{\sigma^2} \right)} dx dy \quad (4.11)$$

Therefore, since the CEP is defined as the probability of 50% that an event lies inside a circle of radius R. Equation (4.11) can be rewritten by changing to polar coordinates with $r^2 = x^2 + y^2$ and $dx dy = r dr d\theta$ and defining a variable u as $u = \frac{-r^2}{2\sigma^2}$, and $du = \frac{-r}{\sigma^2} dr$, hence

$$P(R \leq D) = -\frac{1}{2\pi} \int_0^{2\pi} \int_0^{\frac{-D^2}{(2\sigma^2)}} e^u du d\theta$$

solving this equation yield

$$P(R \leq D) = 1 - e^{\frac{-D^2}{2\sigma^2}} \quad (4.12)$$

Equation (4.12) represents the cumulative density function and provides the probability that a particular computed position lies within a circle of radius R .

The corresponding statistical values for the three collected horizontal position data sets are given in table 4.7. The tabular values of Standard Deviation for the N-S and E-W position are different, even between the same data set. Furthermore, the N-S and E-W position for each data set are correlated by a magnitudes of 0.23, 0.24, and 0.362 respectively. This can be appreciated by carefully looking the scatter pattern on figures 4.23(b) to 4.25(b), since

the data are not spread in a circular pattern. This means at least, for the

Table 4.7: *Horizontal Position Statistical Summary for NAV440-GPS Data*

Direction	σ	σ^2	ρ
North-South 1	1.25	1.56	-0.23
East-West 1	0.91	0.82	
North-South 2	1.31	1.73	-0.24
East-West 2	0.89	0.80	
North-South 3	1.30	1.69	-0.36
East-West 3	0.94	0.88	

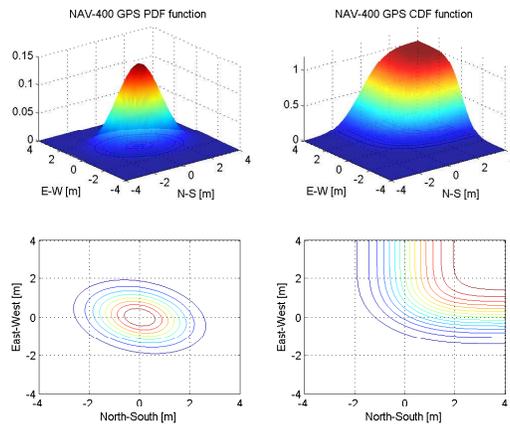
NAV440[®]-GPS IMU, that equation (4.12) is not valid and does not provide the required accuracy in the position error estimation, despite being widely used. As a consequence, Mahalanobis distance with all its terms are used in this work to compute the Ellipse-Error-Probable, or EEP, rather than the CEP.

One way to solve for EEP is to numerically integrate equation (4.9). The left top and bottom plots in figures 4.26(a) to 4.26(c) show the bivariate pdf function with its equiprobable density contours whereas right top and bottom plots of the same figures show the corresponding Cumulative Density function or cdf, with the corresponding equal cumulative density contour. Clearly, the shapes of the contour in figure 4.26 are rotated ellipses that show different areas of the same constant statistical distance. Furthermore, these ellipses are in fact the EEP for different probabilities. The orientation of these EEP depends on the correlation between the random variables, and its elongation is dependant on the error differences of each variable from the mean value.

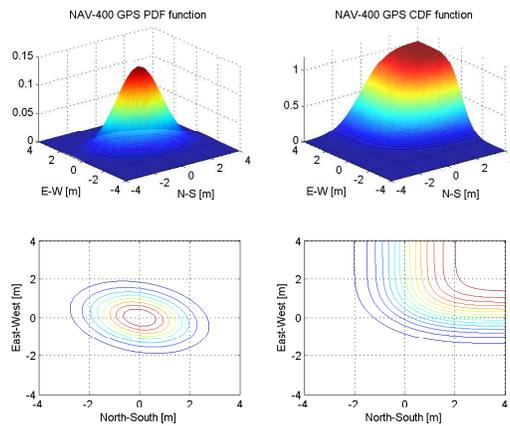
Rewritten the Mahalanobis distance D^2 in vector form in equation (4.9) results in:

$$D^2 = (\mathbf{x} - \boldsymbol{\mu})^T \Xi^{-1} (\mathbf{x} - \boldsymbol{\mu}) \quad (4.13)$$

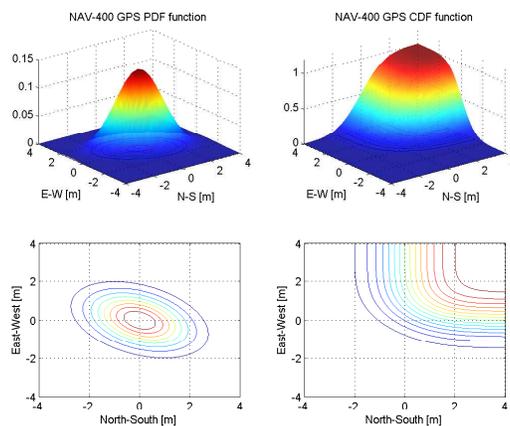
where \mathbf{x} is a n dimension vector of random variables, $\boldsymbol{\mu}$ is a mean vector for the n random variables and Ξ is the n by n covariance matrix. Equation (4.13) is a generalization of the Mahalanobis distance for n random variables space [66]. Substituting equation (4.13) into equation (4.9), results in a multi-variable normal PDF.



(a) Data Set 1.



(b) Data set 2.



(c) Data Set 3.

Figure 4.26: Stationary Position Probability Density Function and Cumulative Density Function with its respective contours for the three data set of the Nav440[®]-GPS.

The relation between an ellipse of constant statistical distance D^2 and the cumulative probability is the χ^2 *Chi square* distribution [66]. Thus the probability that a joint event lies within the EEP is given by

$$P(D^2 \leq \chi_n^2(\xi)) = \xi$$

where $\chi_n^2(\xi)$ represents the quantile of the Chi square distribution with a n degree of freedom related with the cumulative distribution ξ . In addition, the orientation of the ellipse will be given by the orthonormal rotation matrix \mathbf{R}_{EEP} computed from the eigenvectors of the covariance matrix, that is

$$\mathbf{R}_{EEP} = \text{eigv}(\Xi) \quad (4.14)$$

and the ellipse axis will be

$$a_n = \sqrt{\chi_n^2(\xi)\lambda_n} \quad (4.15)$$

where a_n are the axis of the ellipsoid, or ellipse in the case of two random variables, $n=2$, and λ_n represents the corresponding eigenvalue for each eigenvector. In this way, is easy to compute different ellipses for different joint probabilities.

The outcome of applying equations (4.14) and (4.15) and computing correlation and the covariance to the three sets of collected data is tabulated in table 4.8.

Table 4.8: *Statistical Summary based on the Mahalanobis Distance*

	Correlation Matrix		Covariance Matrix		Eigenvectors	λ_n	50% EEP Axis [m]	95 % EEP Axis [m]
	N-S	E-W	N-S	E-W				
N-S (x1)	1	-0.2326	0.8189	-0.263	0.3033i + 0.9529j	0.7352	1.0096	2.0988
E-W (y1)	-0.2326	1	-0.2630	1.5615	0.9529i - 0.3033j	1.6452	1.5102	3.1397
N-S (x2)	1	-0.2400	0.7999	-0.2822	0.2697i + 0.9629j	0.7209	0.9997	2.07883
E-W (y2)	-0.2400	1	-0.2822	1.7283	0.9629i - 0.2697j	1.8074	1.5829	3.2907
N-S (x3)	1	-0.3617	0.8772	-0.4402	0.4014i + 0.9159j	0.6843	0.9740	2.0248
E-W (y3)	-0.3617	1	-0.4402	1.6886	0.9159i - 0.4014j	1.8816	1.6151	3.3576

The size, orientation and enclosed areas of the ellipses in the three data sets are similar. For example, in the first data set, the 50% EEP enclosed area is 4.79 m² while for data set two, it is 4.97 m² for the same probability. This is only a 3.62% difference.

To interpret what these areas mean using an example, assume that helicopter is in hover phase and the computed position will lie within a 95% EEP therefore; helicopter might hover on an equivalent plane area of about 21 m² which correspond to an ellipse of radii 3.35 m by 2.04 m ($A = \pi(3.36)(2.04) \approx 21 \text{ m}^2$)

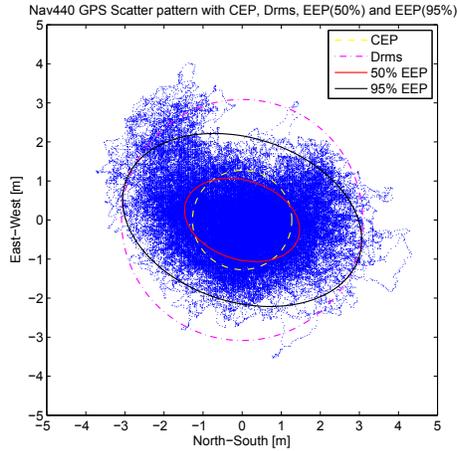
Table 4.9: *Enclosed areas by the ellipses and its rotation angle for the 50% and 95% probability.*

Data set	50% EEP	95% EEP	Rot. Angle
1	4.79 m ²	20.72 m ²	72.34°
2	4.97 m ²	21.48 m ²	74.35°
3	4.94 m ²	21.35 m ²	66.33°

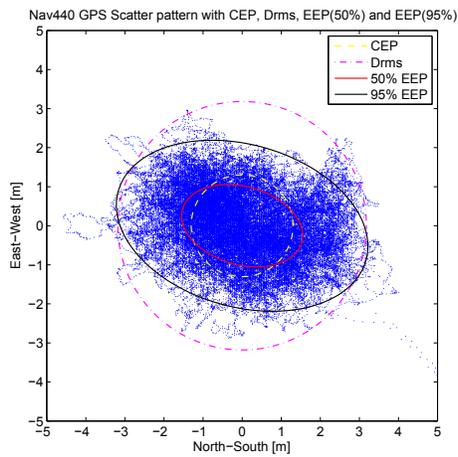
In order to compare between CEP and the EEP, CEP was approximated by the expression $CEP = 0.56\sigma_x + 0.62\sigma_y$ and the 95% Circle Error Probable was computed as two times the Distance Root Mean Square, that is $2DRMS = 2\sqrt{\sigma_x^2 + \sigma_y^2}$ [64]. Figure 4.27 shows the 50% and 95% EEP as well as the CEP and 2DRMS using these approximations for the three collected data sets.

For all the three cases the plots show that, overall the EEP has a better error estimations than the corresponding CEP and 2Drms. When the difference in the variance between the variables decreases, for example, data set three in which variance difference is about 0.29, the scatter pattern tends to fit better on the 2DRMS circle error, with less empty spaces is expected. Nevertheless, the rotation of the cluster data due to the correlation between N-S and E-W variables is only captured by the superposition of the EEP.

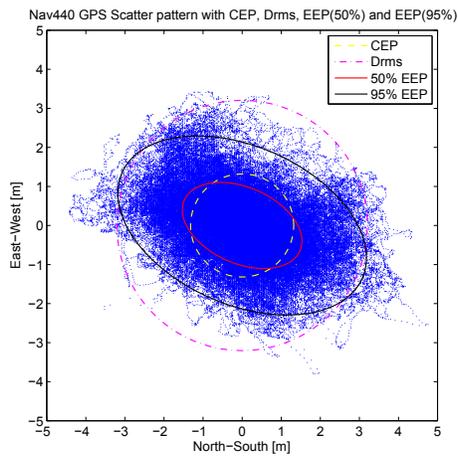
The scattered pattern of figure 4.27 are useful when the helicopter motion is constrained to the North-South-East-West plain. However, this analysis can be easily extended into three dimension space by including the altitude data. This is done by computing the statistical Mahalanobis distance to estimate the corresponding equiprobable ellipsoids. The statistical summary of the outcomes given by equations (4.14) and (4.15) and the computed covariance matrix are given in table 4.10. Figures 4.28 to 4.30 plot the three dimension scatter pattern with the corresponding 50% and 95% EEP for the same three data sets including the altitude.



(a) *Data set 1.*



(b) *Data set 2.*

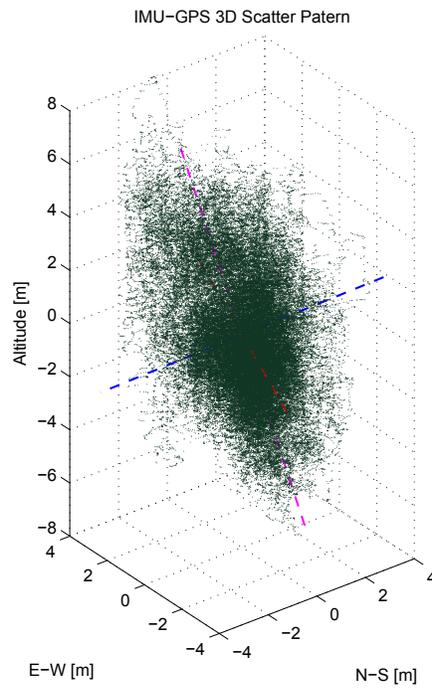


(c) *Data set 3.*

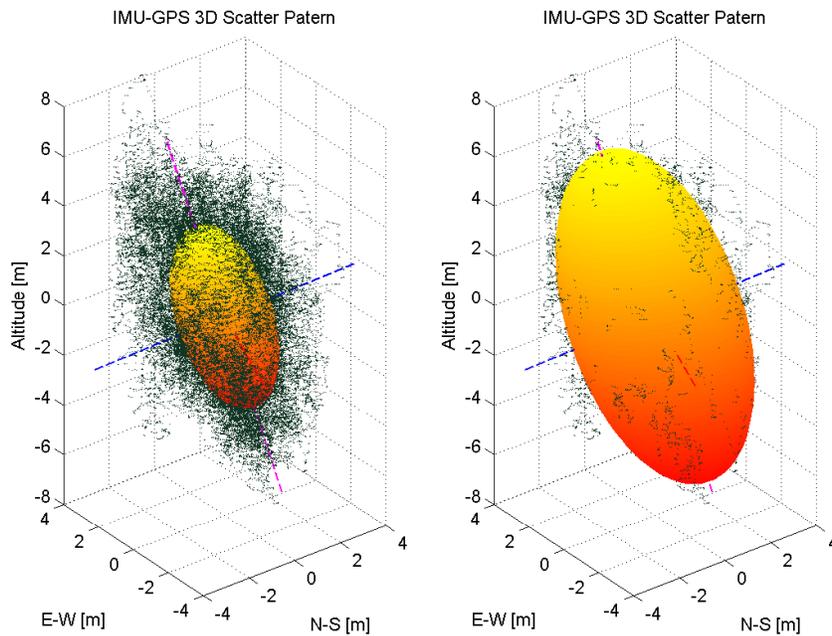
Figure 4.27: Position Scatter Pattern for the three data set showing the 50% and 95% EEP and the approximate CEP and 2DRMS circles.

Table 4.10: Three Dimension Space Statistical Summary based on the Mahalanobis Distance for the three data sets.

Description	Data set 1			Data set 2			Data set 3			
	N-S	E-W	Altitude	N-S	E-W	Altitude	N-S	E-W	Altitude	
ρ Matrix	N-S	1	-0.2326	-0.4521	1	-0.2400	-0.2621	1	-0.3617	-0.2629
	E-W	-0.2326	1	0.1524	-0.2400	1	0.1271	-0.3617	1	0.1351
	Alt	-0.4521	0.1524	1	-0.2621	0.1271	1	-0.2629	0.1351	1
Ξ Matrix	N-S	1.5615	-0.2630	-1.4120	1.7283	-0.2822	-0.8136	1.6886	0.4402	-0.8943
	E-W	-0.2630	0.8189	0.3447	-0.2822	0.7999	0.2684	-0.4402	0.8772	0.3312
	Alt	-1.4120	0.3447	6.2482	-0.8136	0.2684	5.5734	-0.8943	0.3312	6.8503
50 % EEP axis	1.32 m	1.70 m	3.97 m	1.31 m	1.96 m	3.69 m	1.27 m	2.01 m	4.08 m	
95 % EEP axis	2.39 m	3.10 m	7.22 m	2.37 m	3.56 m	6.71 m	2.31 m	3.65 m	7.41 m	
50 % EEP volume	37.23 m^3				39.53 m^3			43.62 m^3		
95 % EEP volume	223.50 m^3				237.29 m^3			261.81 m^3		
Eig. Vec.	\hat{e}_i	0.3497î	0.9363ĵ	0.0310k̂	0.2626î	0.9649ĵ	-0.0093k̂	0.4089î	0.9125ĵ	0.0103k̂
	\hat{e}_j	-0.8974î	0.3443ĵ	-0.2760k̂	-0.9436î	0.2548ĵ	-0.2116k̂	-0.8966î	0.4038ĵ	-0.1818k̂
	\hat{e}_k	-0.2691î	0.0687ĵ	0.9606k̂	-0.2018î	0.0644ĵ	0.9773k̂	-0.1701î	0.0651ĵ	0.9833k̂
Eig. Val.	λ_1	λ_2	λ_3	λ_1	λ_2	λ_3	λ_1	λ_2	λ_3	
	0.7321	1.2281	6.6684	0.7205	1.6221	5.7591	0.6837	1.7055	7.0700	
Rot. Angles	ϕ	θ	ψ	ϕ	θ	ψ	ϕ	θ	ψ	
	105.61°	69.86°	88.22°	101.64°	75.24°	90.53°	99.79°	66.18°	89.41°	

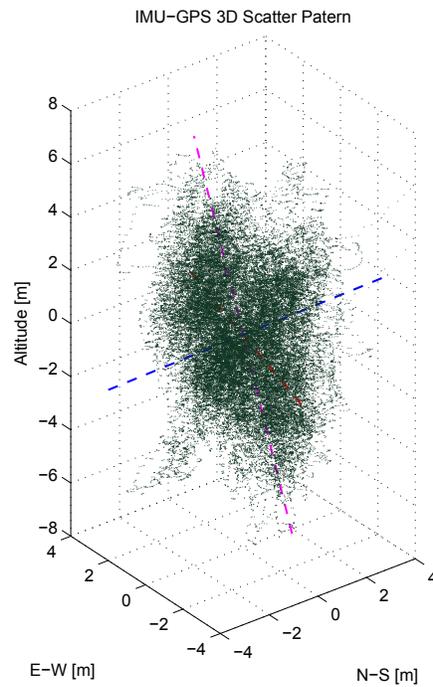


(a) Scatter Pattern.

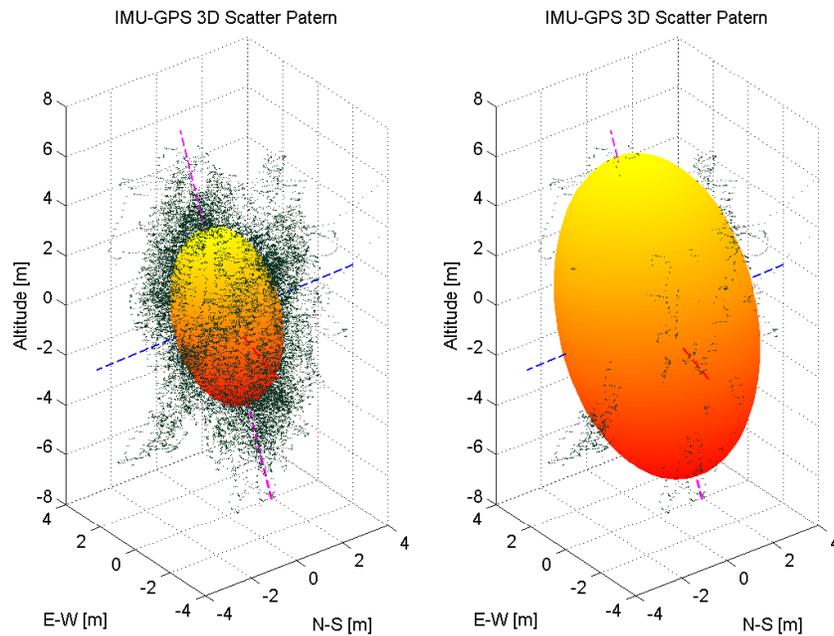


(b) 50% left and 95% right EEP.

Figure 4.28: 3D Position Scatter Pattern for the first data set showing the scatter pattern figure a), the 50% EEP, figure b) left and 95% EEP figure b) right.

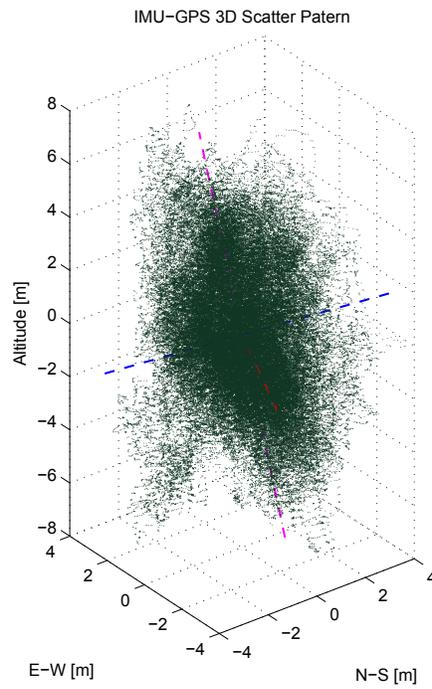


(a) Scatter Pattern.

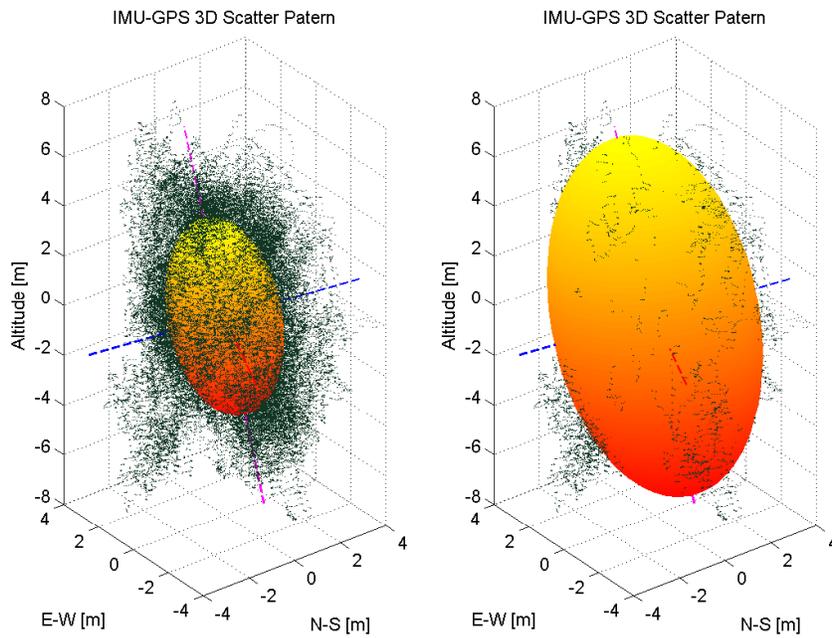


(b) 50% left and 95% right EEP.

Figure 4.29: 3D Position Scatter Pattern for the second data set showing the scatter pattern figure a), the 50% EEP, figure b) left and 95% EEP figure b) right.



(a) Scatter Pattern.



(b) 50% left and 95% right EEP.

Figure 4.30: 3D Position Scatter Pattern for the third data set showing the scatter pattern figure a), the 50% EEP, figure b) left and 95% EEP figure b) right.

Table 4.10 has the correlation between North-South and East-West position which are exactly the same to the previous 2D ones as expected. For the altitude, data set one shows a considerable correlation, about -0.4521, between North-South direction and altitude, meanwhile data sets two and three show similar magnitude correlation factors among them. What is clear from table 4.10 is that the North-South directions seems to have a higher correlation with the altitude than the East-West, for the Nav440[®]-GPS. This is deduced from the orientation of the ellipsoids shown in figures 4.28 to 4.30. In addition, dispersion of the scatter pattern in the altitude direction is also observed. The altitude position range is from -8 m to 8 m for 95%. Although the 50% ellipsoid minimum z axis is 3.69 m, for data set two, this does not guarantee that helicopter will hover over a range of ± 3.96 m from the mean altitude value. Furthermore, in the best scenario, helicopter might hover on a 95% ellipsoid of 223.50 m³ of volume, which correspond to the ellipsoid dimension of data set one.

From this analysis is clear that the altitude given by the Nav400[®]-GPS is not accurate enough for the takeoff and landing phase. Nevertheless, for the cruise phase between two way points, when the altitude is greater than 16 m from the reference ground level the altitude given by GPS could be used. Additional instruments like the barometric altimeter and/or Radar Altimeter, provide a better estimation of the altitude and can be implemented in a state observer in combination with the GPS.

An important question for the hover phase is how fast does the position change with respect to the mean value or initial reference. The position drift error over the entire time interval for data set one is shown in figure 4.31, while figures 4.32 and 4.33 show 5 and 3 minute sections at 3rd and 17th hours of continuous operation. The rate of change of the error and its magnitude varies in time in these figures. For example, in figure 4.32 between minutes 184.186 and 184.353, there is a change in the North-South position from 1.676 m to -0.4786 m which correspond to a magnitude position drift of 2.155 m in just 10.02 seconds, that is a drift speed of 21.5 cm/s. In figure 4.33, between minutes 1031.174 and 1031.457 there is a drift in altitude of 2.75 m in just 16.98 second, corresponding to a drift speed of 16.2 cm/s. These position changes are the fastest for the data shown in figures. Table 4.11 summarizes the position drift magnitudes and its corresponding rate of change for the data

displayed in figures 4.32 and 4.33.

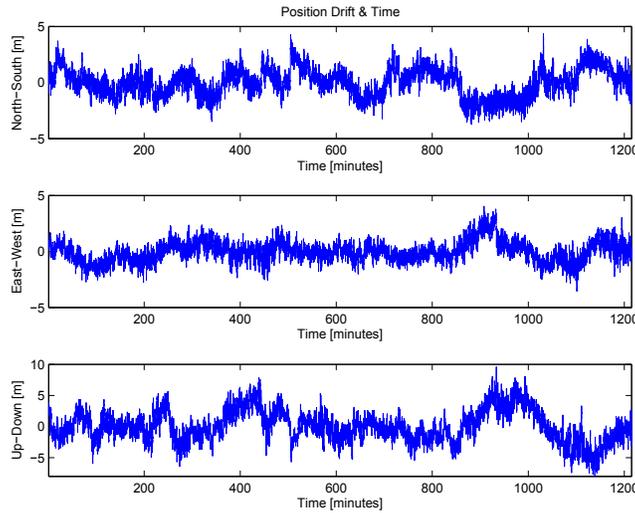


Figure 4.31: *Error Position Drift Vs. Time for data set one.*

Based on the analyzed data sections, it is unlikely that position error changes dramatically from one sample to another. Smooth drift velocity changes are also expected. These results in periods of slow displacement and periods of rapid position drift during the hover phase, are more apparent for long hover periods. This implies that the helicopter should land quickly to avoid landing error or landing accidents. If this is not achievable, pilot assisted landing must be considered.

To test the moving position and velocity accuracy, a car ride with the ECICH box is performed. Position, and velocity data are recorded every 500 ms. A predefined path is followed and the position trajectory is plotted on GoogleMaps[®] for qualitative analysis. In general the moving position data given by the ECICH correspond to the trajectory followed by the vehicle as shown in figure 4.34. However, a close examination show some deviations in the followed path, as shown in figure 4.35. Here, the motion is from East to West, which corresponds to a motion from right to left on figure 4.35. The street is two way street with two lanes in each direction. The upper two lanes run from East to West meanwhile the lower two are for the oncoming direction. The car is driven in the center lane. As shown in the figure, the measured trajectory is close to the center lane, the right side of figure 4.35. As the car moves West, the measured trajectory deviates from the central lane even going on the sidewalk,

Table 4.11: *Magnitude Drift Error and Drift Error Speed.*

Figure 4.32				
North-South Position				
t [min]	Δt [s]	Drift [m]	$ \Delta D $ [m]	Drift Speed [m/s]
181.794		-1.534		
182.294	30	0.706	2.24	0.075
184.186		1.676		
184.353	10.02	-0.4786	2.155	0.215
East-West Position				
181.444		-0.157		
181.678	14.04	-1.211	1.368	0.097
Up-Down Position				
182.569		-1.425		
183.319	45	1.325	2.750	0.061
184.003	41.04	-1.925	3.250	0.080
Figure 4.33				
North-South Position				
t [min]	Δt [s]	Drift [m]	$ \Delta D $ [m]	Drift Speed [m/s]
1031.557		4.336		
1032.432	52.5	0.286	4.050	0.077
East-West Position				
1031.432		-0.351		
1031.807	22.5	-1.833	1.482	0.066
Up-Down Position				
1031.174		-0.175		
1031.457	16.98	2.575	2.750	0.162

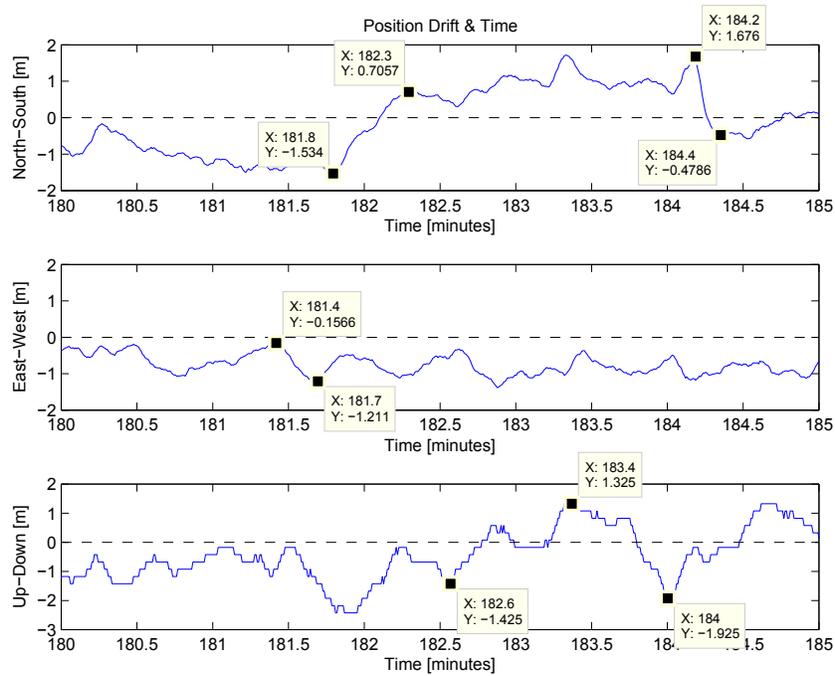


Figure 4.32: Data chunk of three minutes error position drift after 3 hours of continuous operation.

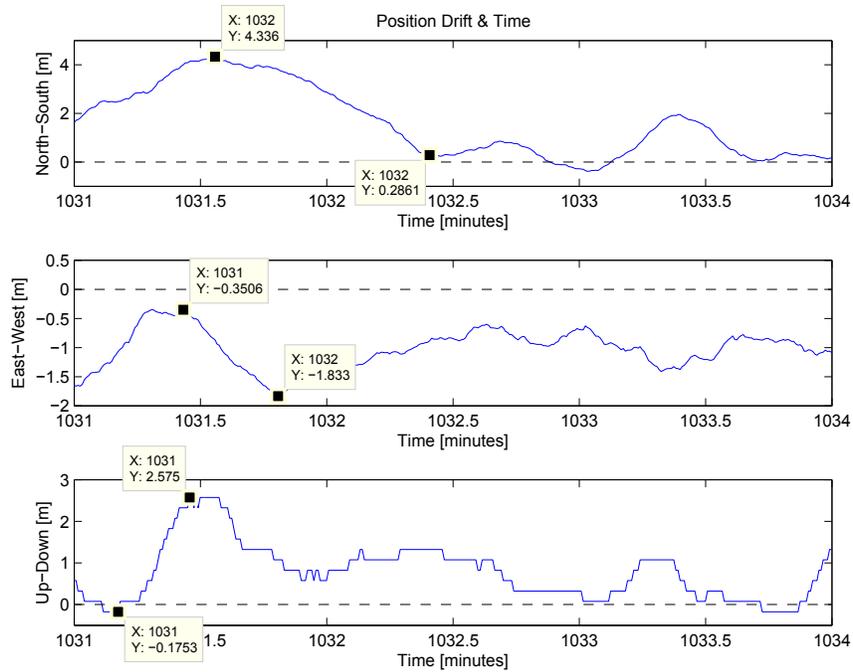


Figure 4.33: Data chunk of three minutes error position drift after 17 hours of continuous operation.

at the bottom part of the figure. As the car continued West, the measured trajectory returned to the central lane. These errors might be introduced by changes in the satellite constellation, multipath signal trajectory, and/or errors induced by the ionosphere. Therefore, deviations from the nominal trajectory is expected to occur in flight formation and must be taken into account by introducing an extra space margin between helicopters during this type of flight regime. Drift magnitude from a nominal path was not experimentally measured. One possibility, is to install a long track in which a controlled speed platform on wheels moves along the tracks. With the ECICH mounted on the moving platform. Then the same single trajectory for long periods of time could be tested to obtain enough data for statistical analysis. However, this is beyond the scope of the present work.

For velocity validation, the automobile is driven at a constant velocity of $112 \text{ km/h} \pm 1 \text{ km/h}$, controlled by the cruise control. The reference instrument is the automobile speedometer. Figure 4.36 shows the North/South, East-West and Up-Down velocities and its magnitude, and figure 4.37 shows the histogram for velocity magnitude. The computed velocity mean μ is $111.06 \text{ km/h} \pm 0.08 \text{ km/h}$ with 95% of confidence interval with a standard deviation σ of 0.70 km/h . The difference between the reference value and the one given by the ECICH is 0.84% which is acceptable.

To validate the RPM speed sensor, a set of 3,264 data points are collected at a sampling frequency of 20 Hz. The main rotor speed is set to 1,000 RPM. The mean value and the standard deviation are computed. A stroboscope model STROBOTAC 1531 from General Radio Co.[®] is used as a reference instrument. Figure 4.38(a) shows the RPM sensor output and figure 4.38(b) shows its histogram plot. The computed mean value μ was $1013.59 \text{ RPM} \pm 0.51 \text{ RPM}$ with 95% of confidence interval and a standard deviation σ of 12.78 RPM . The reading of the stroboscope was $1020 \text{ RPM} \pm 10 \text{ RPM}$, yielding a difference of 0.62% between them. Clearly, readings given by the RPM sensor are accurate enough.

The LRF is validated in similar way to the RPM sensor. A set of 6,336 data are collected at a sampling frequency of 20 Hz. The distance from the LRF to the target was $1234 \text{ mm} \pm 0.5 \text{ mm}$ measured with a standard scale,

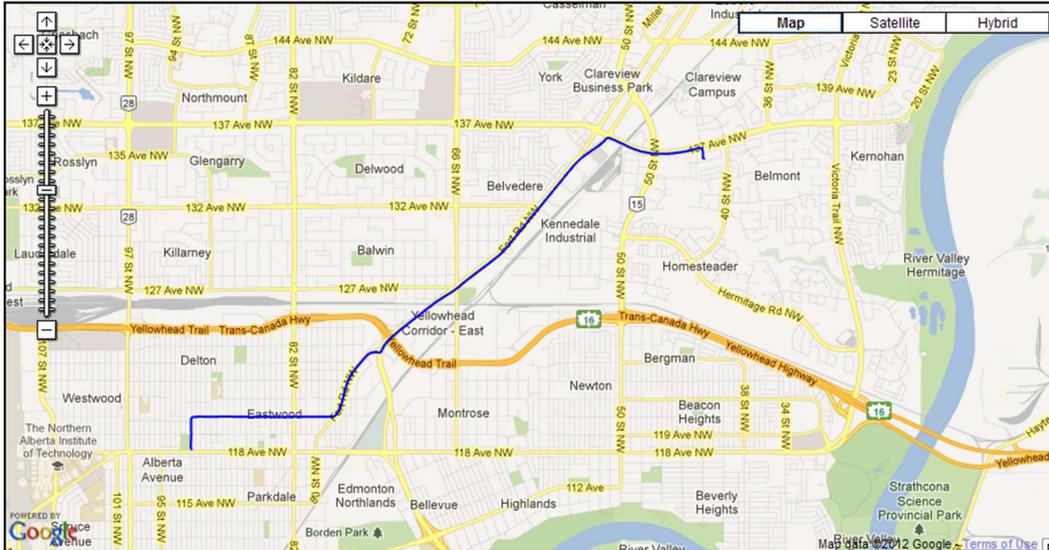


Figure 4.34: Car ride track trajectory.

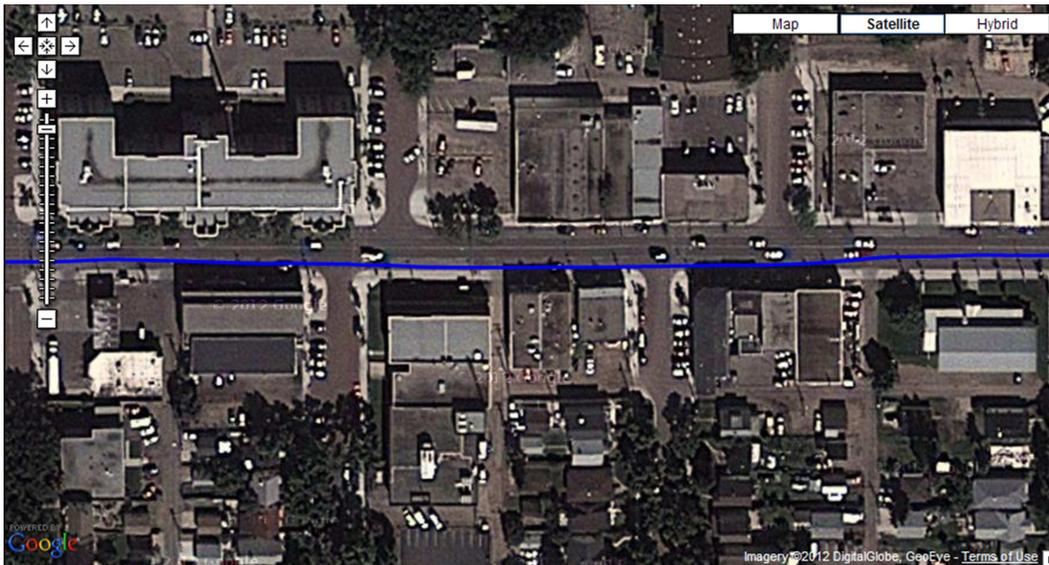


Figure 4.35: Car ride track trajectory zoom in which is possible to observe some deviations from the nominal trajectory.

and assuming it as a reference measure. The LRF cluster angle is 11.2° as was described in section 4.1.5. The computed mean value μ is $1217.7 \text{ mm} \pm 0.094 \text{ mm}$ with 95% of confidence interval and standard deviation σ of 3.82 mm. Figures 4.39(a) shows the LRF output, meanwhile figure 4.39(b) shows its corresponding histogram. The difference between the reference measure and the LRF output is about 1.35%, making it suitable for the takeoff and landing phases.

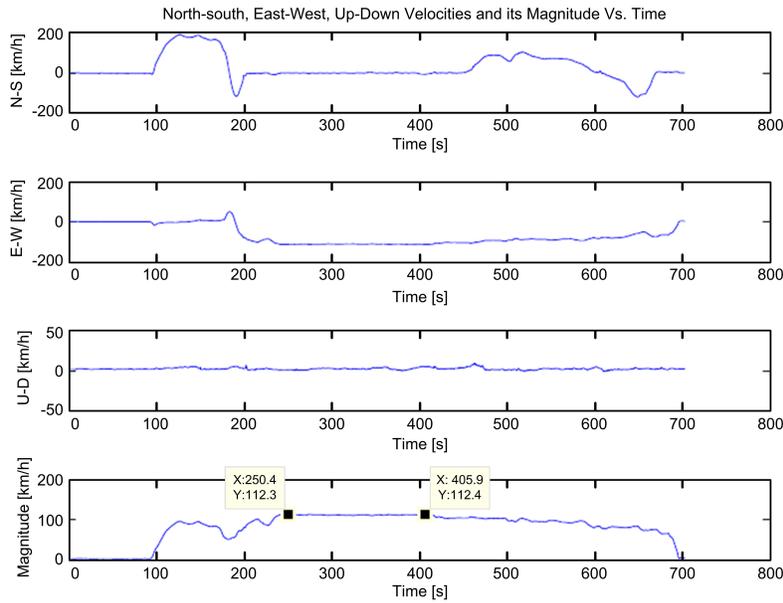


Figure 4.36: *ECICH velocities validation.*

For the voltage and current data validation, 114 seconds of data are recorded at a sampling frequency of 0.5 Hz. The reference multimeter model UT55 from STEREN[®] showed a constant voltage of $11.43 \text{ V} \pm 10 \text{ mV}$. For the voltage data, the computed mean value μ is $11.1558 \text{ V} \pm 3.65 \text{ mV}$ with a standard deviation σ of 13.8 mV . Figure 4.40(a) shows the voltage output of the BMS and figure 4.40(b) exhibits its corresponding histogram. The difference between the reference instrument value and the BMS output is 2.37%, which is sufficient for voltage monitoring. For the current data, the same amount of data and sampling frequency is used. The computed mean value μ was $3.095 \text{ A} \pm 4.1 \text{ mA}$ with 95% of confidence interval and a standard deviation σ of 15.4 mA . The reference instrument is exactly the same as in the voltage case, with a current value of 3.12 A , yielding a percentage difference of 0.81% which is sufficient to estimate of the delivered batteries charge. Figure 4.41(a) shows the BMS output current while figure 4.41(b) shows the corresponding histogram.

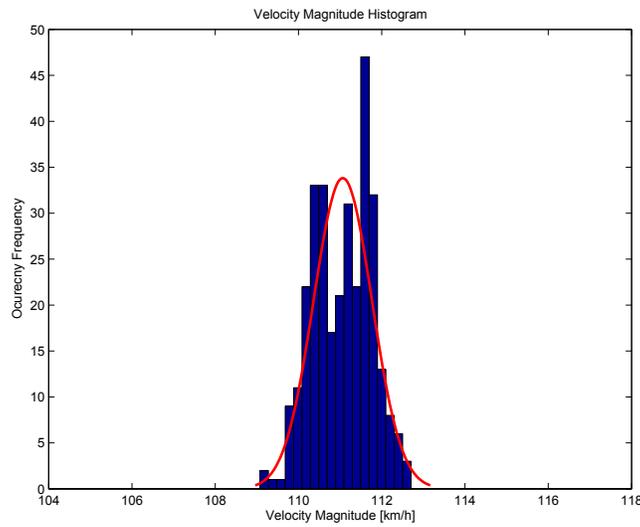
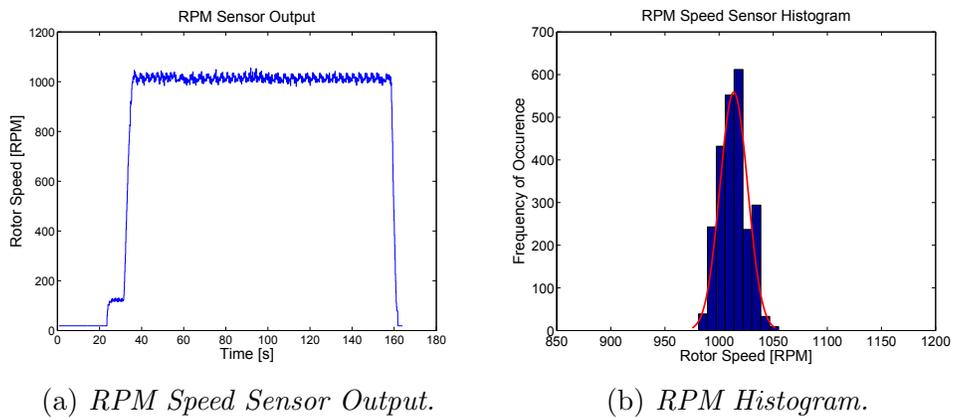


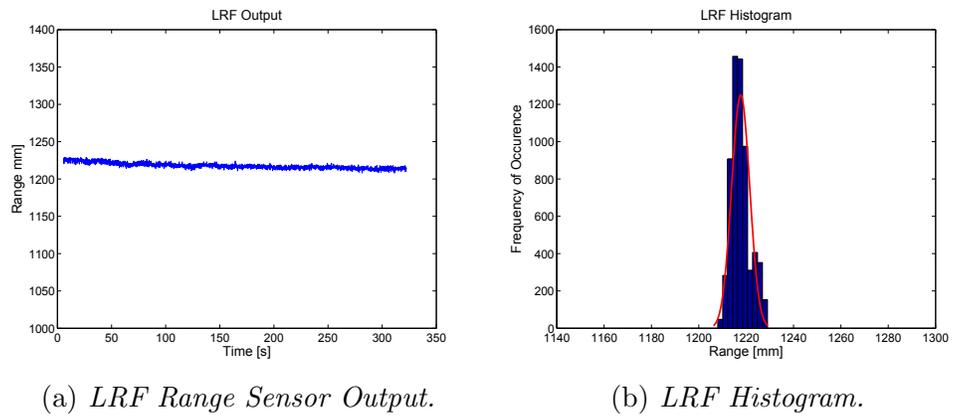
Figure 4.37: Magnitude Velocity Histogram.



(a) RPM Speed Sensor Output.

(b) RPM Histogram.

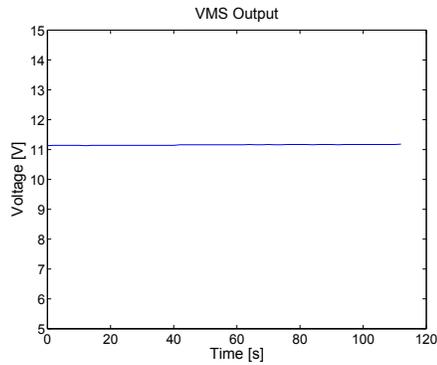
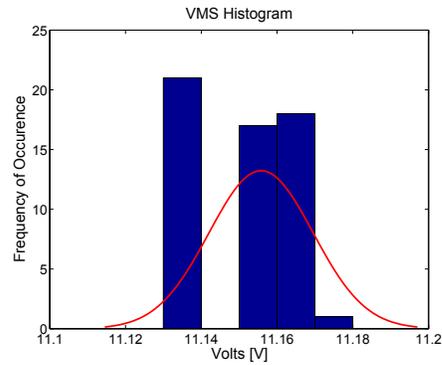
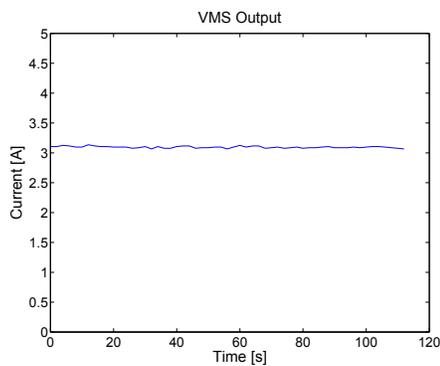
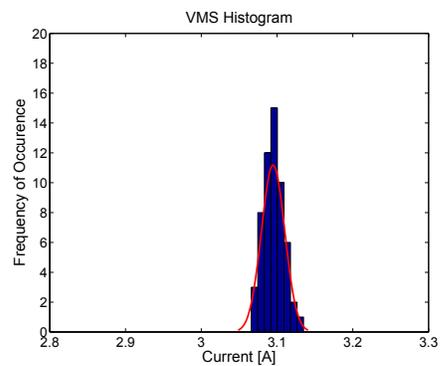
Figure 4.38: RPM Sensor Speed Output a), and RPM histogram b).



(a) LRF Range Sensor Output.

(b) LRF Histogram.

Figure 4.39: LRF Range output a), and LRF Histogram b).

(a) *BMS Voltage Output.*(b) *BMS Histogram.***Figure 4.40:** *BMS Voltage Output a), and BMS Histogram b).*(a) *BMS Current Output.*(b) *BMS Histogram.***Figure 4.41:** *BMS Current Output a), and BMS Histogram b).*

4.4 Data Flight

After having completed ground testing of the ECICH, actual flight data are recorded. Two preliminary flights are performed to verify that all systems and subsystems are properly working. The telemetry and communication link, are tested and recorded data are verified and analyzed.

After analysis, five flight with different maneuvers and flight scenarios are performed. Twenty four parameters are recorded on the ECICH in each flight as shown in figures 4.42 to 4.54.

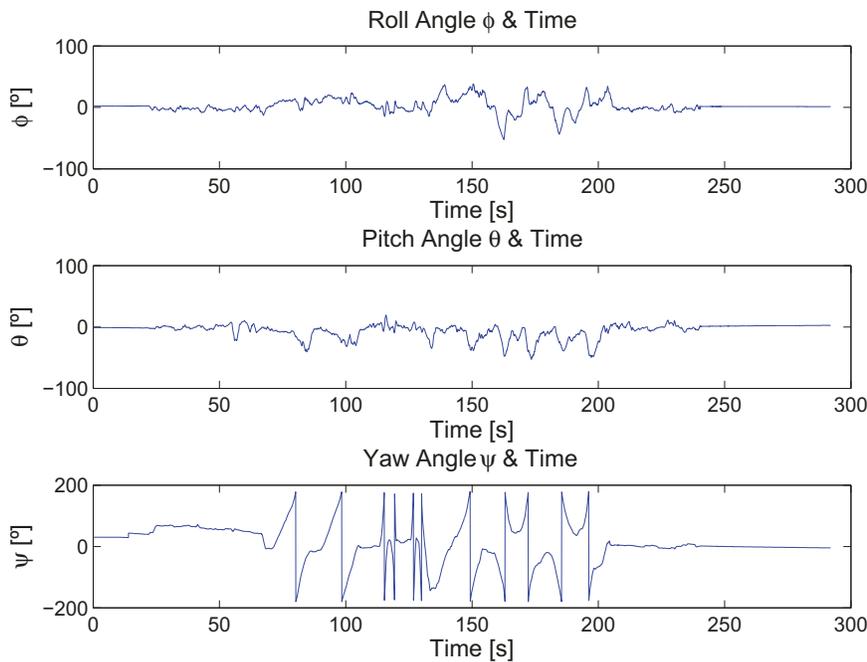


Figure 4.42: *Fifth Flight Euler angles.*

Figure 4.42 shows three plots corresponding to each of the Euler angles ϕ , θ and ψ respectively. It can be seen from the bottom plot of figure 4.42 that the heading angle ψ jumps from -180° to 180° every time the helicopter rotates one half of a turn. The top and middle plots show that the banking and pitch angle ϕ and θ never exceed $\pm 50^\circ$ in the case of roll and $\pm 60^\circ$ for pitch.

Figures 4.43 and 4.44 shows the corresponding angular rates and body accelerations. Here is possible to see linear acceleration peaks on the three axes at about 240 seconds, as well as a peak in the roll rate, (top plot of figure 4.43) at the same time. These peaks were caused by the landing impact

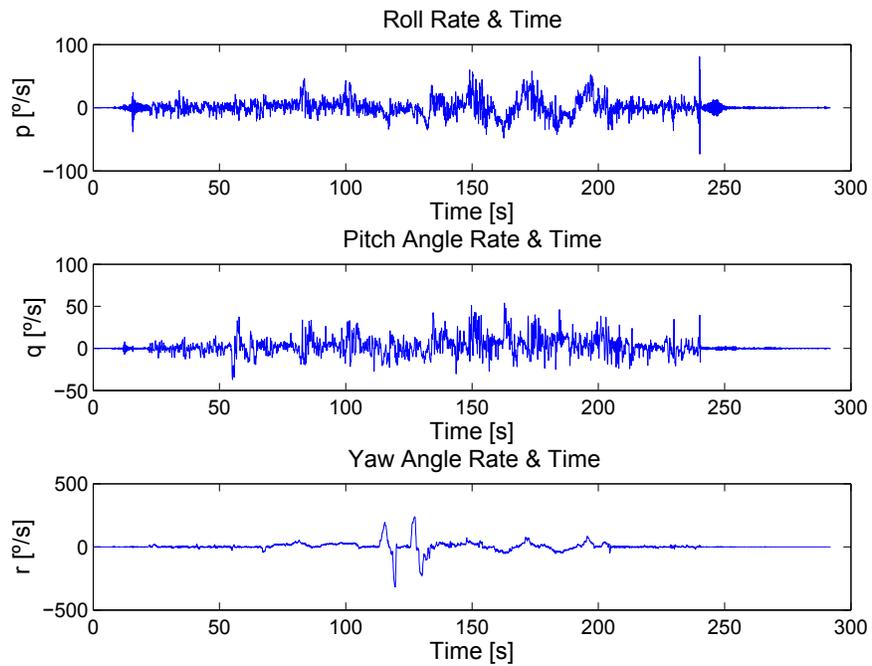


Figure 4.43: *Fifth Flight Body angular rates.*

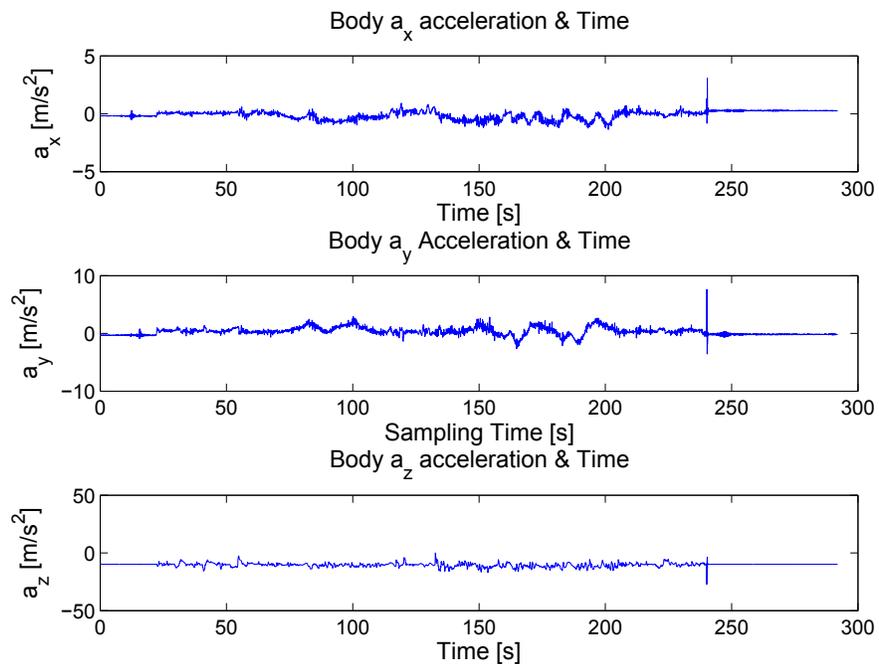


Figure 4.44: *Fifth Flight Accelerations coming out from the IMU.*

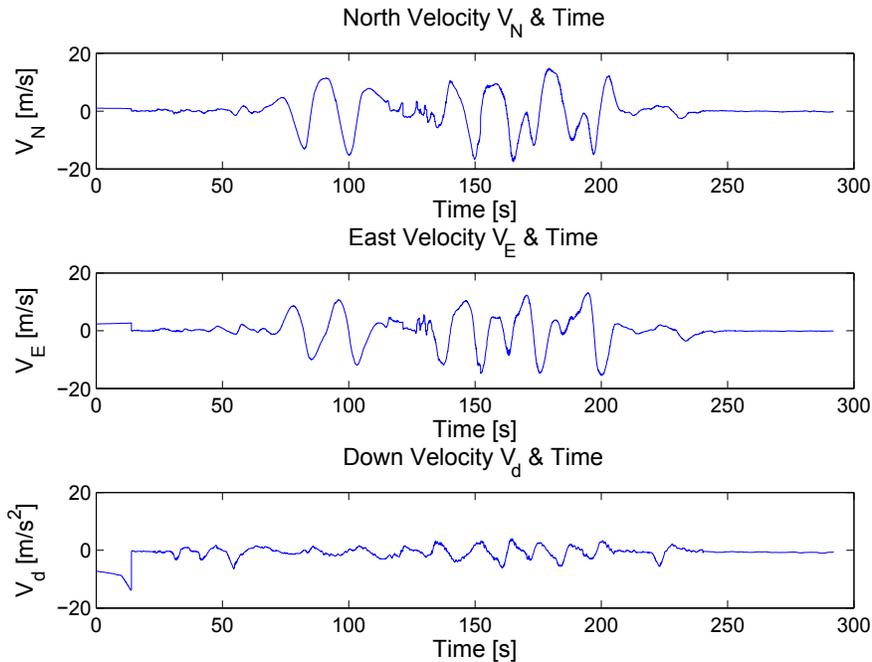


Figure 4.45: *Fifth Flight Frame C_N velocities.*

which is confirmed by the bottom plot of figure 4.44. Here the acceleration peak at 240 seconds of magnitude 25 m/s^2 , which correspond to about 2.5 g of acceleration which seems to be reasonable for a landing impact.

Figure 4.45 shows the linear velocities in the North-East-Down frame with some velocity peaks of 16 m/s (equivalent to 60 km/h).

Figure 4.46 shows helicopter's position given by GPS in geodetic coordinates. It is important to notice the drift in altitude readings. The initial reading, see bottom plot of figure 4.46, shows an altitude of 717 m above the mean sea level. This altitude is read before the helicopter's takeoff, and is used as local ground reference level. After few second the altitude reading shows a decrement in altitude. This is also apparent in figure 4.52 where altitude relative to the local ground level is plotted. According to this graph, at 14 second helicopter would have descended 1 m , which is not possible because at 14 seconds helicopter is still on ground. As discussed previously this altitude drift is inherent to the GPS system as described in section 4.3, see figures 4.28 to 4.30.

Figure 4.47 shows the throttle pulse width command in ms vs. time, and the main rotor angular speed in RPM vs. time. All the flights are conducted

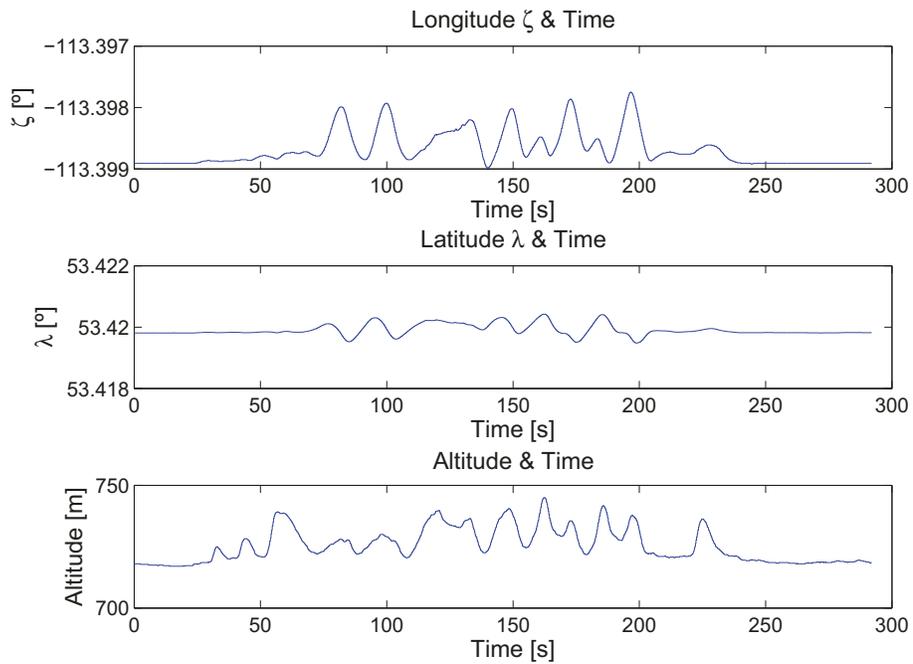


Figure 4.46: *Fifth Flight GPS position.*

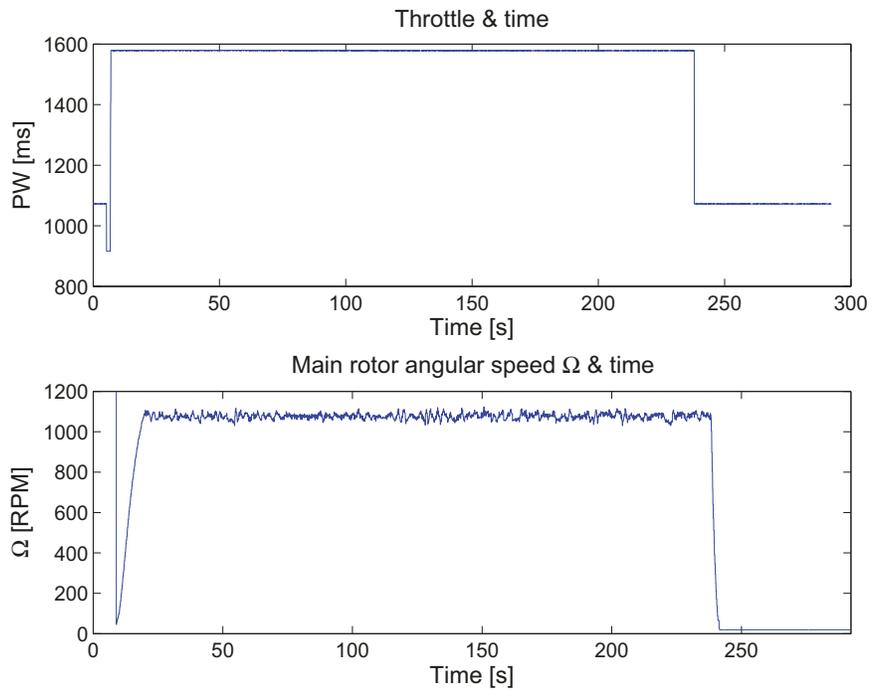


Figure 4.47: *Fifth Flight Throttle and main rotor RPM.*

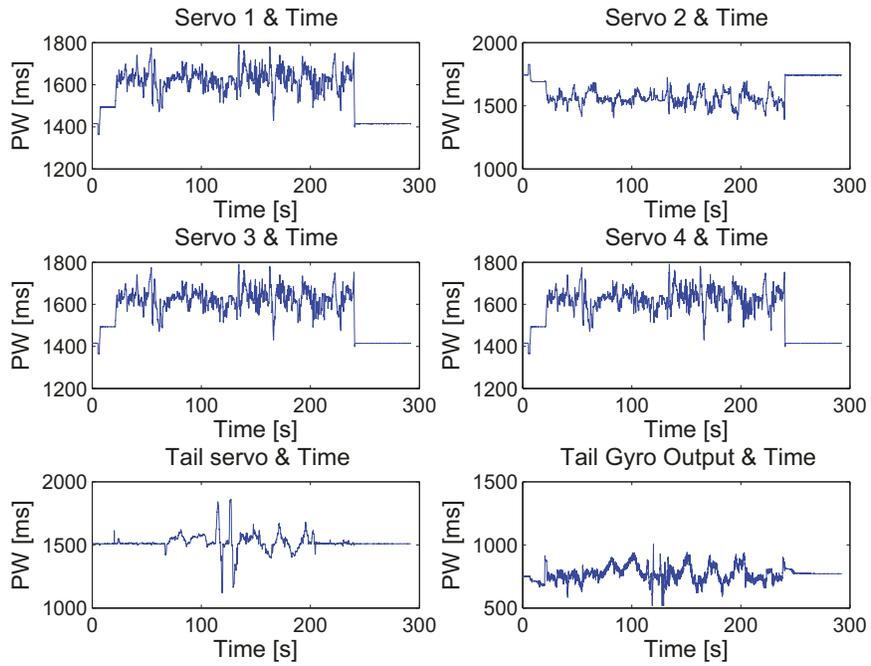


Figure 4.48: Fifth Flight Pilot servo commands.

with a main rotor constant angular speed of 1100 RPM.

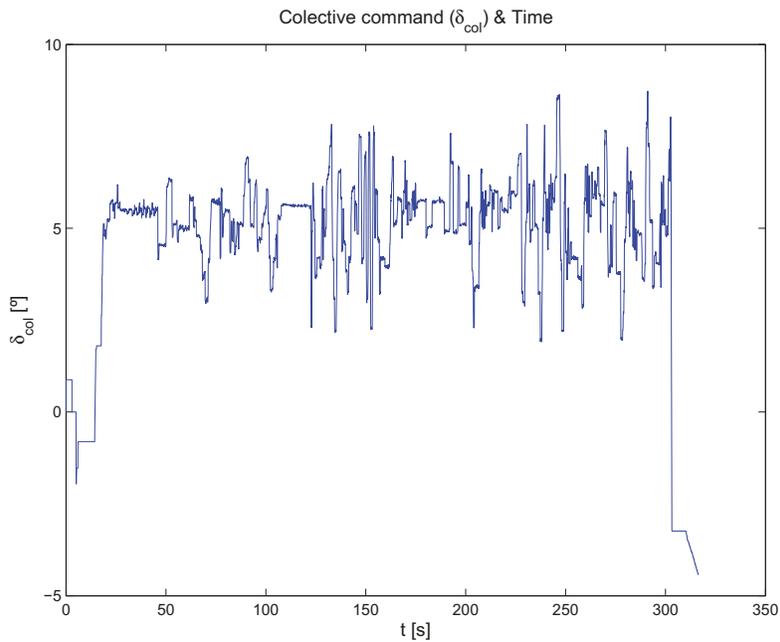


Figure 4.49: Collective angle for the main rotor derived from the expression
$$\delta_{col} = \frac{+\theta - -\theta}{2}$$

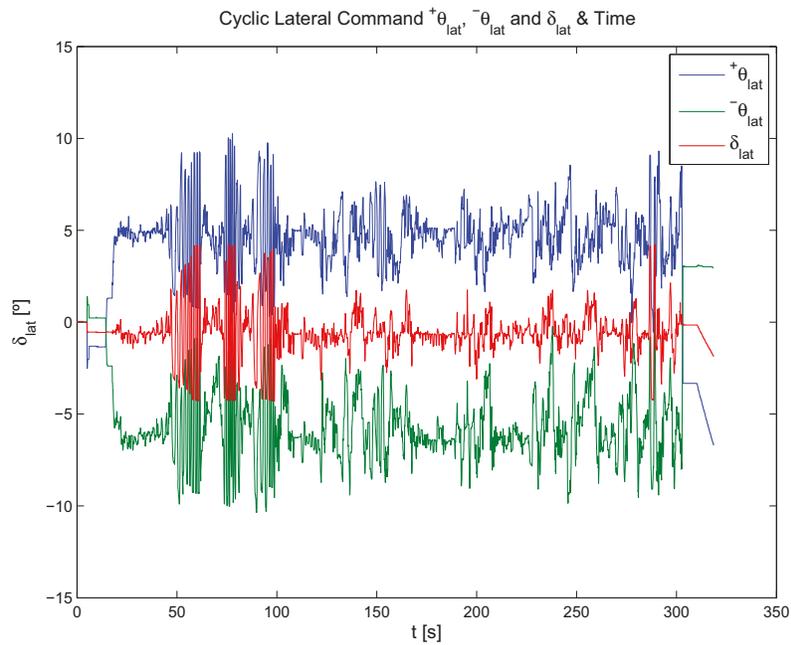


Figure 4.50: Cyclic lateral command derived from expression $\delta_{lat} = \frac{+\theta_{lat} + -\theta_{lat}}{2}$.

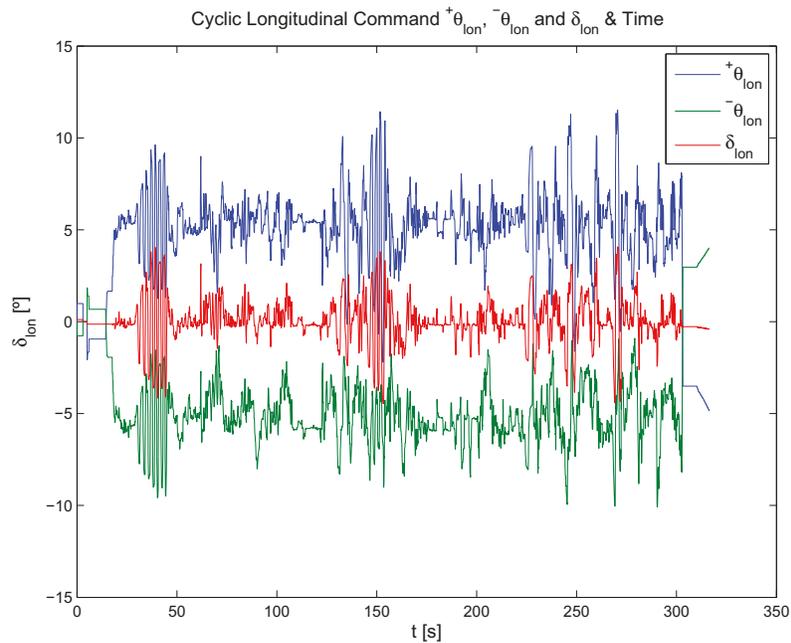


Figure 4.51: Cyclic longitudinal command derived from expression $\delta_{lat} = \frac{+\theta_{lon} + -\theta_{lon}}{2}$.

The recorded pilot's command for all servos and the output produced by the tail gyro controller are shown in figure 4.48. These command are recorded

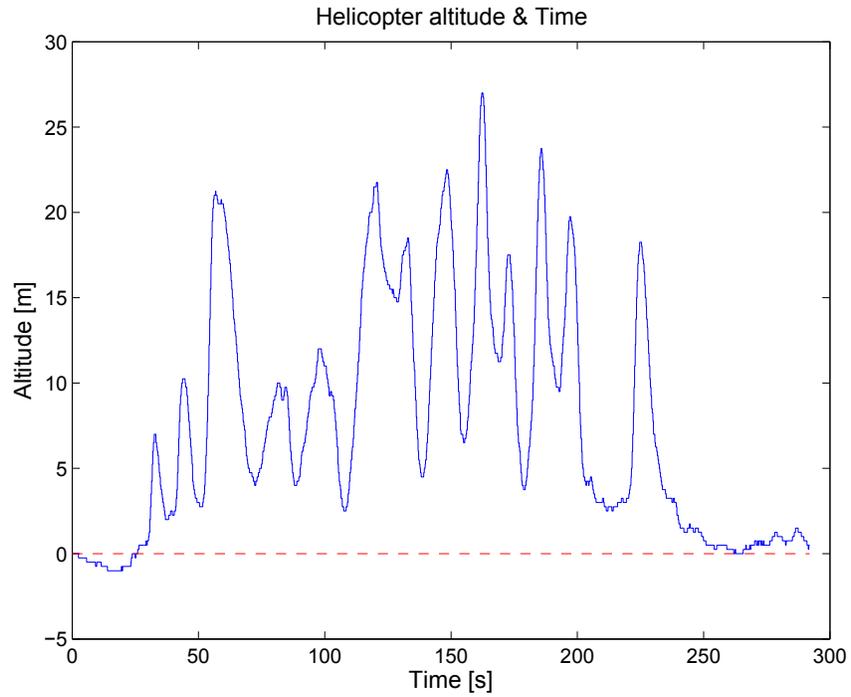


Figure 4.52: *Fifth Flight Helicopter GPS altitude.*

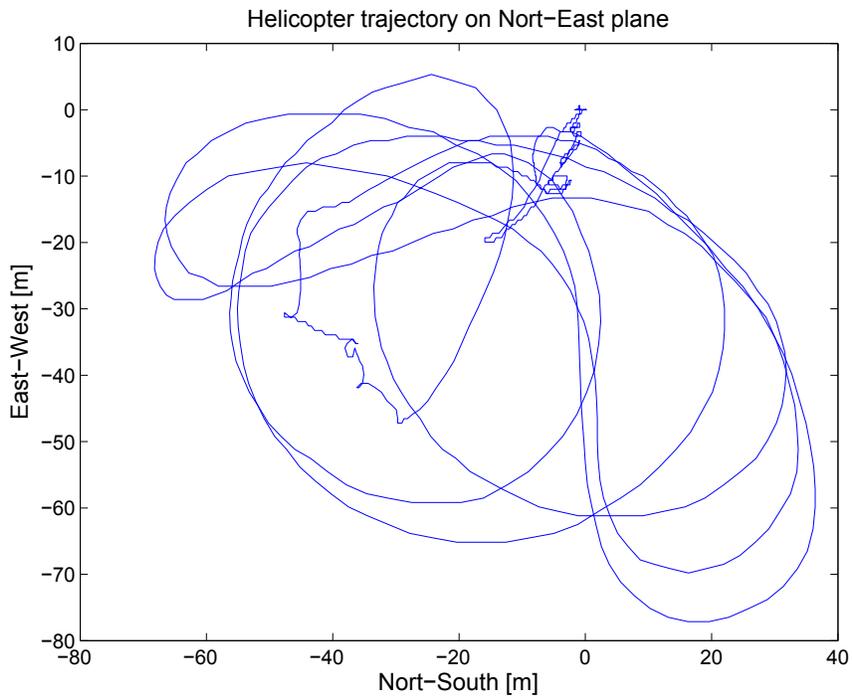


Figure 4.53: *Fifth Flight Helicopter trajectory in the North-East plane.*

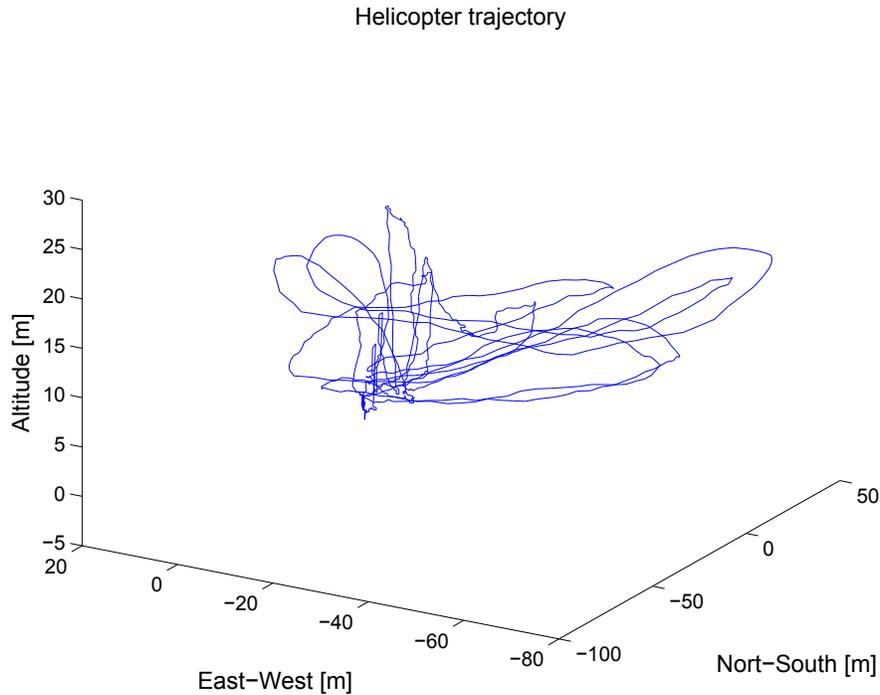


Figure 4.54: *Fifth Flight Helicopter trajectory on space.*

to emulate the pilot actions in the lab for the parameter identification phase.

Figure 4.49 shows the collective angle from the pilot derived by the expression $\delta_{col} = \frac{+\theta - \theta}{2}$. Here it is possible to see that the collective pitch angle for hover is about 5.6° . Figures 4.50 and 4.51 show the cyclic lateral and longitudinal commands respectively derived from the expression $\delta_{lat,lon} = \frac{+\theta_{lat,lon} + \theta_{lat,lon}}{2}$.

Finally different aspects of the helicopter trajectory in fifth flight are shown in figures 4.52 to 4.54. Figures C.31 to C.36 in appendix C, show similar plots for the second and fourth flight data and have similar outcomes.

Chapter 5

System Linearization and Optimal LQR/LQG Controller

In this chapter the dynamical equations of motion of an Air Star Evolution helicopter (derived in chapter 2) are linearized around hover using a small perturbation linearization method.

A combination of a state feedback deterministic Linear Quadratic Regulator (LQR) and a Linear Quadratic Gaussian (LQG) compensator with reference set points $u, v, w, p, q, r, \phi, \theta$ and ψ , according to the dynamic equations, in an output feedback configuration is proposed. Although the helicopter is linearized about hover, the output feedback with set points allows other flight regime like forward, heave, lateral and heading flight that are not “too” far from the hover equilibrium point.

5.1 Model linearization

From the small perturbation linearization method [27] – [70] a system of the form

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, t) \quad (5.1)$$

working in equilibrium at some operating point

$$\dot{\mathbf{x}}_{eq} = f(\mathbf{x}_{eq}, \mathbf{u}_{eq}, t)$$

where \mathbf{x}_{eq} and \mathbf{u}_{eq} are the selected operating states and operating inputs respectively, can be linearized by expanding the system given by (5.1) in a Taylor's series. If the non-linear terms of the series expansion can be neglected by operating the system close to its equilibrium point, that is

$$\Delta \mathbf{x} = \mathbf{x} - \mathbf{x}_{eq}, \quad \Delta \mathbf{u} = \mathbf{u} - \mathbf{u}_{eq}$$

and

$$\Delta \dot{\mathbf{x}} = \dot{\mathbf{x}} - \dot{\mathbf{x}}_{eq} = \dot{\mathbf{x}} - f(\mathbf{x}_{eq}, \mathbf{u}_{eq}, t)$$

then the system can be represented as

$$\Delta \dot{\mathbf{x}} = \mathbf{A} \Delta \mathbf{x} + \mathbf{B} \Delta \mathbf{u} \quad (5.2)$$

where $\Delta \mathbf{x}$ and $\Delta \mathbf{u}$ are small deviations from the operating point. Matrices \mathbf{A} and \mathbf{B} are computed using the Jacobian evaluated at the equilibrium points $\mathbf{x}_{eq} = [u_o, v_o, w_o, p_o, q_o, r_o, \phi_o, \theta_o, \psi_o]$ and $\mathbf{u}_{eq} = [\delta_{col_o}, \delta_{lat_o}, \delta_{lon_o}, \delta_{tail_o}]$. The resulting Jacobian matrices are:

$$\mathbf{A} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial u} & \frac{\partial \dot{u}}{\partial v} & \frac{\partial \dot{u}}{\partial w} & \frac{\partial \dot{u}}{\partial p} & \frac{\partial \dot{u}}{\partial q} & \frac{\partial \dot{u}}{\partial r} & \frac{\partial \dot{u}}{\partial \phi} & \frac{\partial \dot{u}}{\partial \theta} & \frac{\partial \dot{u}}{\partial \psi} \\ \frac{\partial \dot{v}}{\partial u} & \frac{\partial \dot{v}}{\partial v} & \frac{\partial \dot{v}}{\partial w} & \frac{\partial \dot{v}}{\partial p} & \frac{\partial \dot{v}}{\partial q} & \frac{\partial \dot{v}}{\partial r} & \frac{\partial \dot{v}}{\partial \phi} & \frac{\partial \dot{v}}{\partial \theta} & \frac{\partial \dot{v}}{\partial \psi} \\ \frac{\partial \dot{w}}{\partial u} & \frac{\partial \dot{w}}{\partial v} & \frac{\partial \dot{w}}{\partial w} & \frac{\partial \dot{w}}{\partial p} & \frac{\partial \dot{w}}{\partial q} & \frac{\partial \dot{w}}{\partial r} & \frac{\partial \dot{w}}{\partial \phi} & \frac{\partial \dot{w}}{\partial \theta} & \frac{\partial \dot{w}}{\partial \psi} \\ \frac{\partial \dot{p}}{\partial u} & \frac{\partial \dot{p}}{\partial v} & \frac{\partial \dot{p}}{\partial w} & \frac{\partial \dot{p}}{\partial p} & \frac{\partial \dot{p}}{\partial q} & \frac{\partial \dot{p}}{\partial r} & \frac{\partial \dot{p}}{\partial \phi} & \frac{\partial \dot{p}}{\partial \theta} & \frac{\partial \dot{p}}{\partial \psi} \\ \frac{\partial \dot{q}}{\partial u} & \frac{\partial \dot{q}}{\partial v} & \frac{\partial \dot{q}}{\partial w} & \frac{\partial \dot{q}}{\partial p} & \frac{\partial \dot{q}}{\partial q} & \frac{\partial \dot{q}}{\partial r} & \frac{\partial \dot{q}}{\partial \phi} & \frac{\partial \dot{q}}{\partial \theta} & \frac{\partial \dot{q}}{\partial \psi} \\ \frac{\partial \dot{r}}{\partial u} & \frac{\partial \dot{r}}{\partial v} & \frac{\partial \dot{r}}{\partial w} & \frac{\partial \dot{r}}{\partial p} & \frac{\partial \dot{r}}{\partial q} & \frac{\partial \dot{r}}{\partial r} & \frac{\partial \dot{r}}{\partial \phi} & \frac{\partial \dot{r}}{\partial \theta} & \frac{\partial \dot{r}}{\partial \psi} \\ \frac{\partial \dot{\phi}}{\partial u} & \frac{\partial \dot{\phi}}{\partial v} & \frac{\partial \dot{\phi}}{\partial w} & \frac{\partial \dot{\phi}}{\partial p} & \frac{\partial \dot{\phi}}{\partial q} & \frac{\partial \dot{\phi}}{\partial r} & \frac{\partial \dot{\phi}}{\partial \phi} & \frac{\partial \dot{\phi}}{\partial \theta} & \frac{\partial \dot{\phi}}{\partial \psi} \\ \frac{\partial \dot{\theta}}{\partial u} & \frac{\partial \dot{\theta}}{\partial v} & \frac{\partial \dot{\theta}}{\partial w} & \frac{\partial \dot{\theta}}{\partial p} & \frac{\partial \dot{\theta}}{\partial q} & \frac{\partial \dot{\theta}}{\partial r} & \frac{\partial \dot{\theta}}{\partial \phi} & \frac{\partial \dot{\theta}}{\partial \theta} & \frac{\partial \dot{\theta}}{\partial \psi} \\ \frac{\partial \dot{\psi}}{\partial u} & \frac{\partial \dot{\psi}}{\partial v} & \frac{\partial \dot{\psi}}{\partial w} & \frac{\partial \dot{\psi}}{\partial p} & \frac{\partial \dot{\psi}}{\partial q} & \frac{\partial \dot{\psi}}{\partial r} & \frac{\partial \dot{\psi}}{\partial \phi} & \frac{\partial \dot{\psi}}{\partial \theta} & \frac{\partial \dot{\psi}}{\partial \psi} \end{bmatrix} \quad \begin{array}{l} \mathbf{x} = \mathbf{x}_{eq} \\ \mathbf{u} = \mathbf{u}_{eq} \end{array}$$

and

$$\mathbf{B} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{u}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{u}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{u}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{v}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{v}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{v}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{v}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{w}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{w}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{w}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{w}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{p}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{p}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{p}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{p}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{q}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{q}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{q}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{q}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{r}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{r}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{r}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{r}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{1at}}} & \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{1on}}} & \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{tail}}} \end{bmatrix} \quad \begin{array}{l} \mathbf{x} = \mathbf{x}_{eq} \\ \mathbf{u} = \mathbf{u}_{eq} \end{array}$$

The 117 derivatives for the matrix elements of \mathbf{A} and \mathbf{B} evaluated at the equilibrium point \mathbf{x}_{eq} and \mathbf{u}_{eq} are detailed in appendix D.2.

Before numerically evaluating all the derivatives, it is necessary to find the equilibrium values \mathbf{x}_{eq} and \mathbf{u}_{eq} for hover flight.

$$0 = f(\mathbf{x}_{eq}, \mathbf{u}_{eq})$$

5.1.1 Linearization About Hover

In hover, the body velocities u , v , and w and the angular rates p , q and r are zero. For simplicity let us assume that all the components of the wind velocity are zero. The heading angle ψ can be arbitrarily set without compromising the equilibrium point for hover. So, a zero value for ψ is chosen. These quantities represent the equilibrium values at hover for the body velocities, angular velocities and heading angle. To find θ_o , ϕ_o and the corresponding collective and cyclic commands, some additional values are needed to solve equations (2.118) and (2.119).

The main rotor induced velocity in hover v_{h_i} and the tail rotor induced velocity v_{t_i} are given by

$$v_{h_i} = \sqrt{\frac{T_r}{2\rho\pi R^2}} = \sqrt{\frac{m_T g}{2\rho\pi R^2}}$$

and

$$v_{it} = \sqrt{\frac{\frac{|Q_{r_h}|}{|x_{tcg}|}}{2\rho\pi R_t^2}}$$

where Q_{r_h} is the main rotor drag torque in hover.

From the flight data shown in figure 4.49 the collective pitch angle in hover is about 5.6° and from figure 3.17(c) the corresponding drag torque $Q_{r_h} = 5.9 \text{ N} \cdot \text{m}$. Substituting these values, the resulting induced velocities are:

$$v_{h_i} = 3.86 \text{ m/s}$$

and

$$v_{it} = 5.3 \text{ m/s}$$

For the vertical fin under the effect of the tail rotor induced velocity, the corresponding Reynolds number is

$$\mathcal{R}_e = \frac{\rho V_{vf_1} d}{\mu} = 3483.55$$

Therefore from figure 2.13 the value of the drag coefficient is

$$C_{D_{vf_1}} \approx 0.95$$

and for the percentage of the vertical fin out of the effect of the induced velocity, the drag coefficient $C_{D_{vf_2}}$ is zero.

Setting the rates of change \dot{u} , \dot{v} , \dot{w} , \dot{p} , \dot{q} and \dot{r} to zero in equations (2.118) and (2.119), and substituting the parameters given in table 2.1, the induced velocities of the main and tail rotor for hover, the body and wind velocities, the angular rates, and the coefficients $C_{D_{vf_1}}$, $C_{D_{vf_2}}$ into equations (2.118) and (2.119) results in

$$\begin{aligned}
0 &= -9.81 \sin(\theta_o) - 49.1249\delta_{\theta_{col_o}} \delta_{\theta_{lon_o}} - 2.596588\delta_{\theta_{lon_o}} \\
0 &= 0.3976 + 9.81 \sin(\phi_o) \cos(\theta_o) + 49.1249\delta_{\theta_{col_o}} \delta_{\theta_{lat_o}} + 2.5966\delta_{\theta_{lat_o}} + \\
&\quad 4.5037\delta_{\theta_{tail_o}} \\
0 &= 10.3506 + 9.81 \cos(\phi_o) \cos(\theta_o) - 197.1844\delta_{\theta_{col_o}} + 0.0462\delta_{\theta_{tail_o}}^2 - \\
&\quad 0.0004\delta_{\theta_{tail_o}} \\
0 &= 0.7956 + 338.6925\delta_{\theta_{col_o}} \delta_{\theta_{lat_o}} + 17.9022\delta_{\theta_{lat_o}} - 1565.4666\delta_{\theta_{lon_o}} - \\
&\quad 0.0616\delta_{\theta_{tail_o}}^2 + 9.0154\delta_{\theta_{tail_o}} \\
0 &= 0.0072 + 103.0446\delta_{\theta_{col_o}} \delta_{\theta_{lon_o}} + 476.2812\delta_{\theta_{lat_o}} + 5.4466\delta_{\theta_{lon_o}} - \\
&\quad 0.9148\delta_{\theta_{tail_o}}^2 + 0.0173\delta_{\theta_{tail_o}} \\
0 &= -3.9885 + 292.2915\delta_{\theta_{col_o}}^2 + 146.1457\delta_{\theta_{lat_o}}^2 + 146.1457\delta_{\theta_{lon_o}}^2 + \\
&\quad 27.6698\delta_{\theta_{col_o}} - 45.4729\delta_{\theta_{tail_o}}
\end{aligned}$$

Solving the above equations for $\delta_{\theta_{col_o}}$, $\delta_{\theta_{lat_o}}$, $\delta_{\theta_{lon_o}}$, $\delta_{\theta_{tail_o}}$ and θ_o and ϕ_o in a range of $-28.6^\circ \leq \theta \leq 28.6^\circ$ and $-5.7^\circ \leq \phi \leq 5.7^\circ$, the resultant values are in table 5.1.

Table 5.1: *Hover equilibrium values of collective and cyclic commands.*

Variable	Value
$\delta_{\theta_{col_o}}$	5.85°
$\delta_{\theta_{lat_o}}$	0°
$\delta_{\theta_{lon_o}}$	0.04°
$\delta_{\theta_{tail_o}}$	2.38°
θ_o	-0.03°
ϕ_o	-3.42°

Substituting the equilibrium values into the derivatives (see appendix D.2),

the resulting matrices \mathbf{A} and \mathbf{B} for hover flight are:

$$\mathbf{A} = \begin{bmatrix} -0.016228 & 0 & 0.000502 & -0.023924 & 0.005074 & 0.000094 & 0 & -9.81 & 0 \\ 0 & -0.091468 & -0.000051 & -0.011841 & -0.022575 & 0.090735 & 9.792578 & -0.000338 & 0 \\ -0.002 & -0.000099 & -2.680855 & -0.00002 & 0.000383 & 0.260183 & 0.584364 & 0.005671 & 0 \\ -2.889713 & -0.263449 & 0.000284 & -13.620866 & 23.206268 & 0.253591 & 0 & 0 & 0 \\ 0.034331 & -0.883550 & -0.00265 & -6.527671 & -4.141861 & 0.007946 & 0 & 0 & 0 \\ 0.000781 & 0.757739 & 0.394601 & 0.066088 & -0.010647 & -1.031759 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0.000034 & -0.000578 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.998224 & 0.059568 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.059568 & 0.998224 & 0 & 0 & 0 \end{bmatrix} \quad (5.3)$$

and

$$\mathbf{B} = \begin{bmatrix} -0.036649 & 0 & -7.614912 & 0 \\ -0.001885 & 7.614912 & 0 & 4.503726 \\ -197.184402 & 0 & 0 & 0.003393 \\ -0.012996 & 52.501128 & -1565.466581 & 9.010286 \\ 0.076875 & 476.281235 & 15.973067 & -0.05866 \\ 87.387531 & -0.011215 & 0.218061 & -45.472878 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (5.4)$$

The state space realization is

$$\begin{aligned} \Delta \dot{\mathbf{x}} &= \mathbf{A}\Delta \mathbf{x} + \mathbf{B}\Delta \mathbf{u} \\ \mathbf{y} &= \mathbf{C}\Delta \mathbf{x} \end{aligned} \quad (5.5)$$

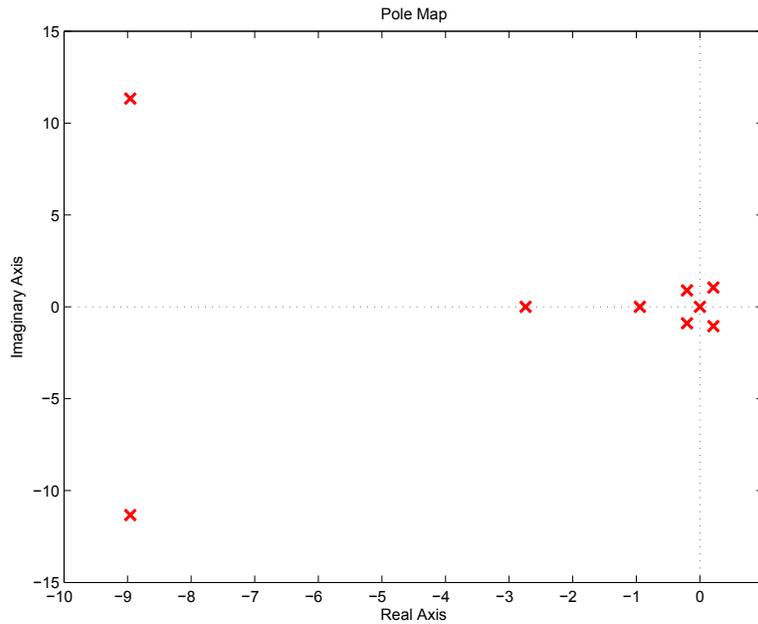
with matrix $\mathbf{C} = \mathbf{I}_{9 \times 9}$ because all the states are available for the controller.

5.2 LQR/LQG Controller

The system described by equation (5.5) is unstable as can be seen from the corresponding eigenvalues of matrix \mathbf{A} in table 5.2, and the pole plot of figure 5.1.

Table 5.2: *Eigenvalues of the open-loop system (5.5)*

Eigenvalues of matrix A
0
-8.9587 + 11.3303i
-8.9587 - 11.3303i
-2.7411
0.2125 + 1.0525i
0.2125 - 1.0525i
-0.2022 + 0.8916i
-0.2022 - 0.8916i
-0.9449

**Figure 5.1:** *Pole map of the Open-Loop linearized dynamics of the Air Star Evolution helicopter.*

For example, in figures 5.2, 5.3, and 5.4 can be seen that the responses of the body velocities u , v , and w to an impulsive cyclic command inputs δ_{col} , δ_{lat} , δ_{lon} , and δ_{tail} rapidly increase unbounded. The same responses are observed for the rest of the states, showing that the system is highly unstable.

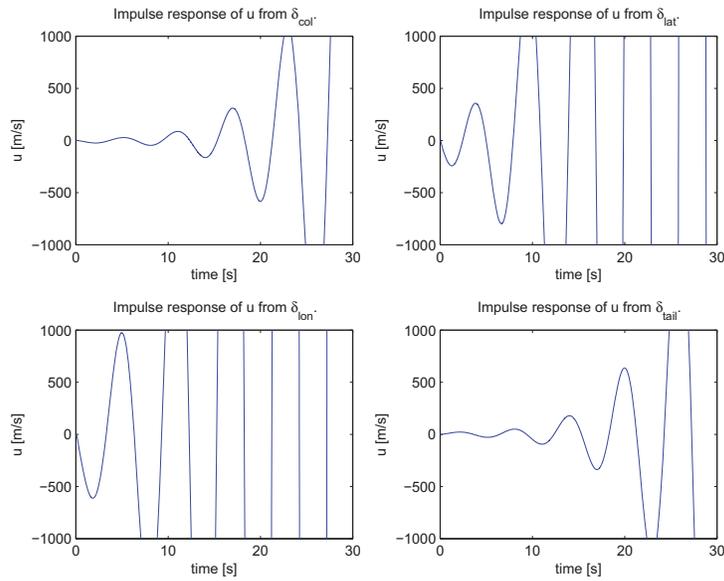


Figure 5.2: *Open-Loop impulse response of the body velocity u vs. collective, cyclic and tail input commands.*

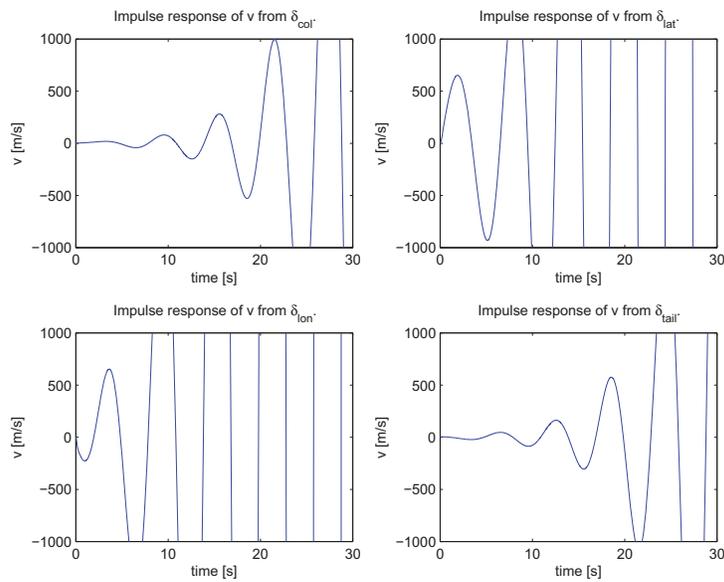


Figure 5.3: *Open-Loop impulse response of the body velocity v vs. collective, cyclic and tail input commands.*

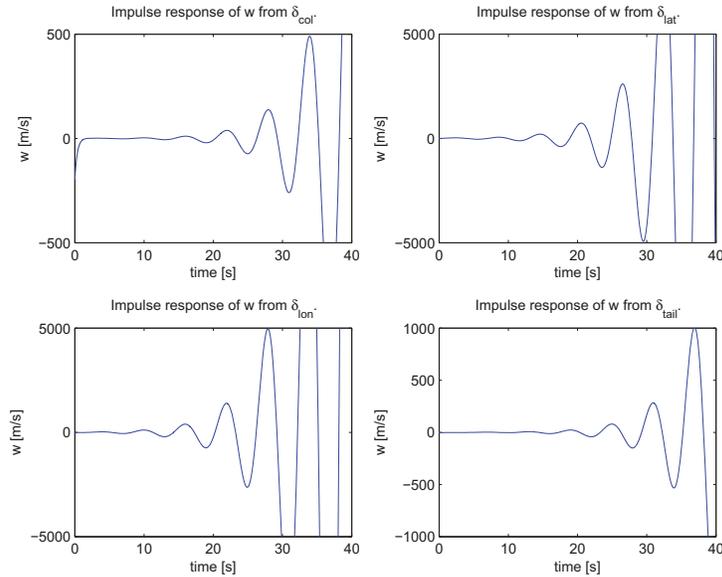


Figure 5.4: *Open-Loop impulse response of the body velocity w vs. collective, cyclic and tail input commands.*

Despite the fact that the system is highly unstable, the rank of the controllability matrix, $\text{rank}(\mathcal{C}_{9 \times 36}) = 9$ and the rank of the observability matrix, $\text{rank}(\mathcal{O}_{81 \times 9}) = 9$ are full row rank and full column rank matrices, so the system is controllable and observable.

In addition, all the states are available and measured using the instrumentation on the helicopter, (see chapter 4). Therefore, a full state feedback is implemented in combination with an Optimal Linear Gaussian compensator.

5.2.1 Optimal Linear Quadratic Regulator

The optimal LQR controller is shown in figure 5.5 [67] – [82] where the state

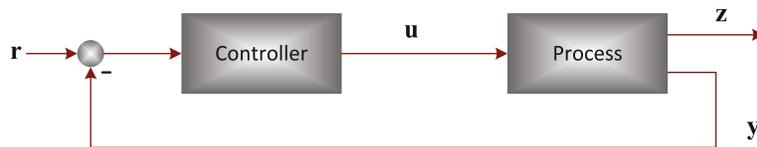


Figure 5.5: *Block diagram of the LQR implementation.*

feedback control law \mathbf{u}_c is

$$\mathbf{u}_c = -\mathbf{K}\Delta\mathbf{x} + \mathbf{r} \quad (5.6)$$

where \mathbf{r} are set points close to the equilibrium point. The output \mathbf{z} represents the states to be controlled, and are given by

$$\mathbf{z} = \mathbf{G}\Delta\mathbf{x} + \mathbf{H}\mathbf{u}_c \quad (5.7)$$

where $\mathbf{G} = \mathbf{C}$ and $\mathbf{H} = 0$ reducing equation (5.7) to

$$\mathbf{z} = \mathbf{C}\Delta\mathbf{x} \quad (5.8)$$

To compute the optimal gain \mathbf{K} , a cost function of the form [68]

$$J_{LQR} = \int_0^{\infty} [\mathbf{z}^T \mathbf{Q} \mathbf{z} + \rho \mathbf{u}_c^T \mathbf{R} \mathbf{u}_c] dt \quad (5.9)$$

is introduced. The parameter ρ is a positive constant to mediate between the energy of the controlled states \mathbf{z} and the energy of the controller outputs \mathbf{u}_c . The element of matrices \mathbf{Q} and \mathbf{R} are estimated using the Bryson's rule

$$Q_{ii} = \frac{1}{\max\{z_i^2\}}$$

and

$$R_{jj} = \frac{1}{\max\{u_{c_j}^2\}}$$

After minimized the cost function (5.9), the optimal controller gain is

$$\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} \quad (5.10)$$

where matrix \mathbf{P} is the solution to the following associated Algebraic Riccati Equation (ARE)

$$0 = \mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} + \mathbf{Q} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P}$$

Then the closed-loop system with optimum gain \mathbf{K} is then defined as

$$\begin{aligned} \Delta \dot{\mathbf{x}} &= (\mathbf{A} - \mathbf{B} \mathbf{K}) \Delta \mathbf{x} + \mathbf{B} \mathbf{r} \\ \mathbf{y} &= \mathbf{C} \Delta \mathbf{x} \end{aligned} \quad (5.11)$$

Table 5.3 show the selected maximum allowed values for the nine controlled states \mathbf{z} and the four input commands \mathbf{u} according to the helicopter's dynamics.

Three different values of the parameter ρ are selected: 0.01, 0.1 and 1.

MatLab[®] *lqr* command is used to compute the gain matrix \mathbf{K} that minimize the cost function (5.9). Three different optimal gain matrix \mathbf{K} are computed. Figures 5.6 to 5.14 show the impulse response of the 36 transfer func-

Table 5.3: Selected maximum value of the states and the command inputs

State	Max Value
u	20 m/s
v	20 m/s
w	20 m/s
p	60 deg/s
q	60 deg/s
r	60 deg/s
ϕ	60 deg
θ	60 deg
ψ	60 deg
Input	Max Value
δ_{col}	13.18 deg
δ_{lat}	13.18 deg
δ_{lon}	13.18 deg
δ_{tail}	13.18 deg

tions for the three different values of the parameter ρ . From these graphs, the best performance of all the individual responses is with the value of $\rho = 0.01$. The corresponding optimal gain matrix is

$$\mathbf{K} = \begin{bmatrix} -0.0019 & 0.0075 & -0.0079 & 0.0002 & 0.0000 & 0.5961 & 0.0186 & 0.0508 & 0.5955 \\ -0.0361 & 0.0001 & 0.0001 & 0.0234 & 0.6866 & 0.0005 & 0.0539 & 0.9794 & -0.0408 \\ -0.0002 & -0.0368 & -0.0013 & -0.6845 & 0.0228 & -0.0054 & -0.9850 & 0.0541 & -0.0048 \\ 0.0010 & -0.0036 & -0.0189 & 0.0038 & -0.0001 & -0.3524 & 0.0323 & -0.0292 & -0.3515 \end{bmatrix} \quad (5.12)$$

5.2.2 State Observer and LQG Compensator

A full state observer is often used [68] to minimize the effect of the measurement noise and process perturbations and to introduce reference inputs to the controller. The plant can be modeled as

$$\Delta \dot{\mathbf{x}} = \mathbf{A} \Delta \mathbf{x} + \mathbf{B} \Delta \mathbf{u} + \mathbf{G}_d \boldsymbol{\vartheta}_d \quad (5.13)$$

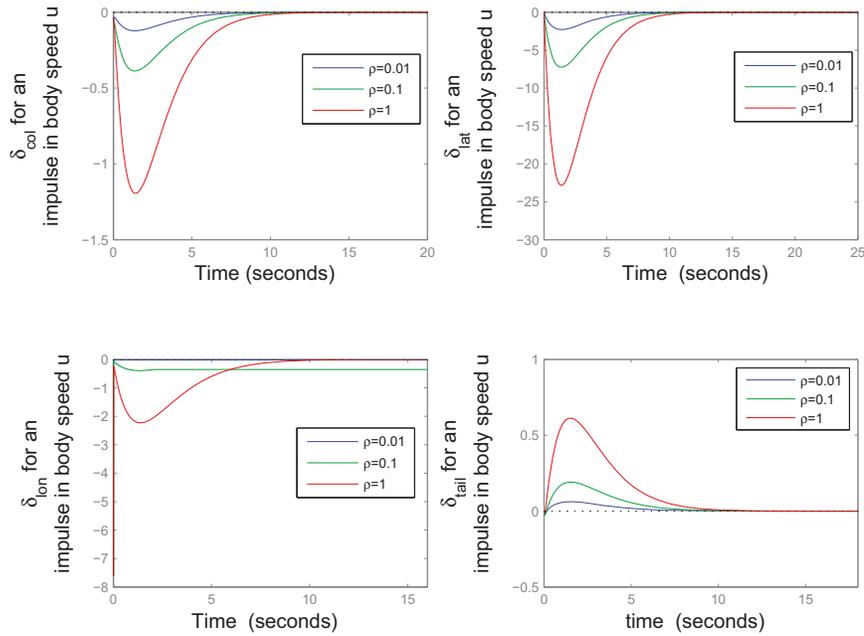


Figure 5.6: Closed-loop control inputs due to an impulse in body speed u

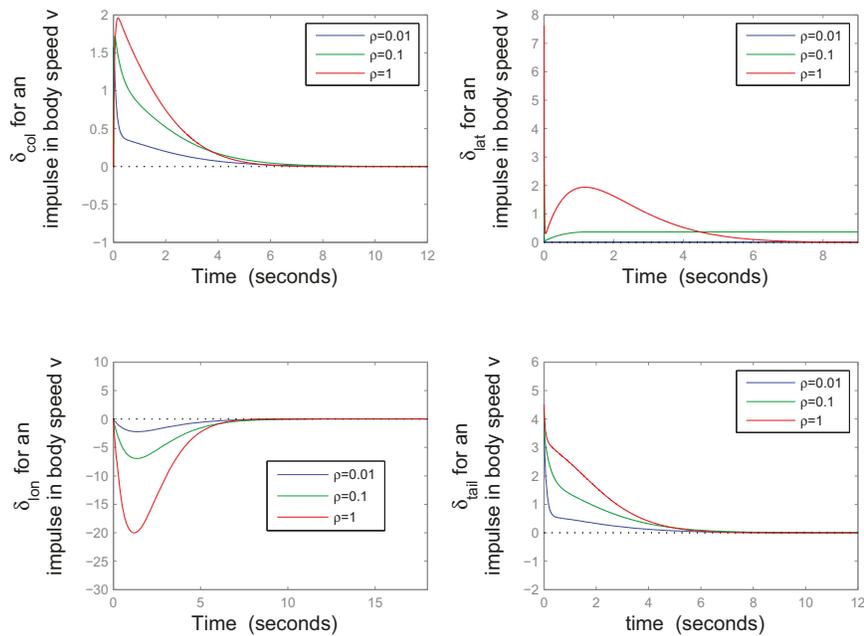


Figure 5.7: Closed-loop control inputs due to an impulse in body speed v

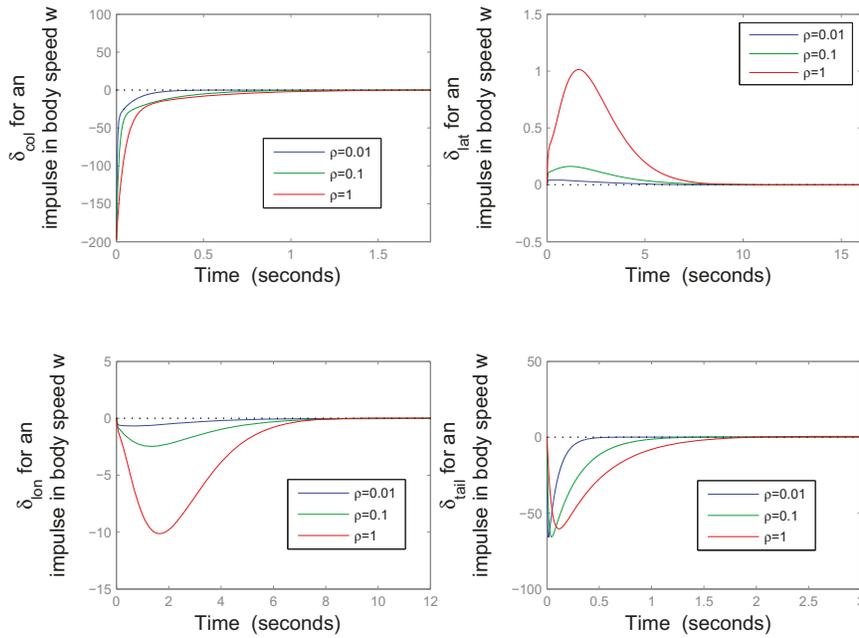


Figure 5.8: Closed-loop control inputs due to an impulse in body speed w

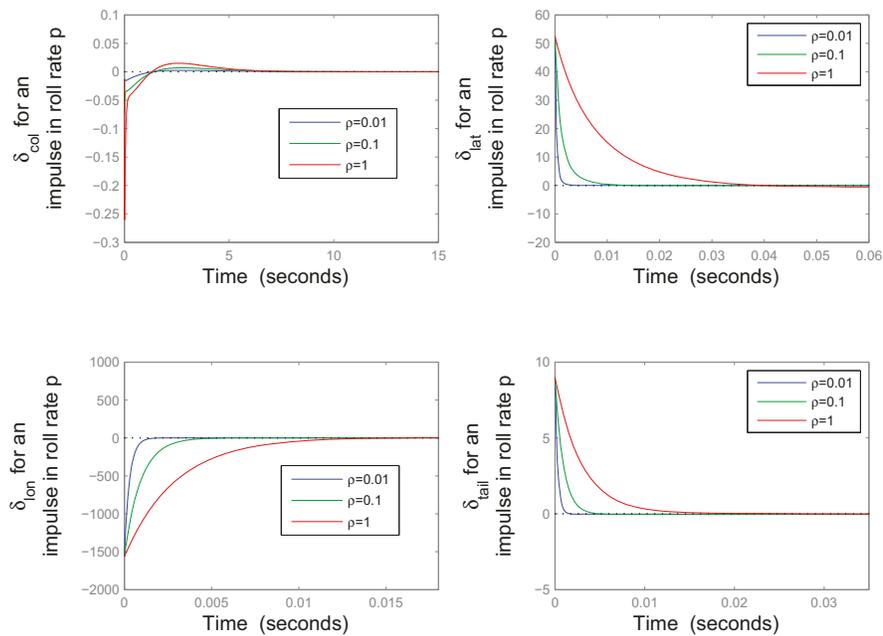


Figure 5.9: Closed-loop control inputs due to an impulse in roll rate p

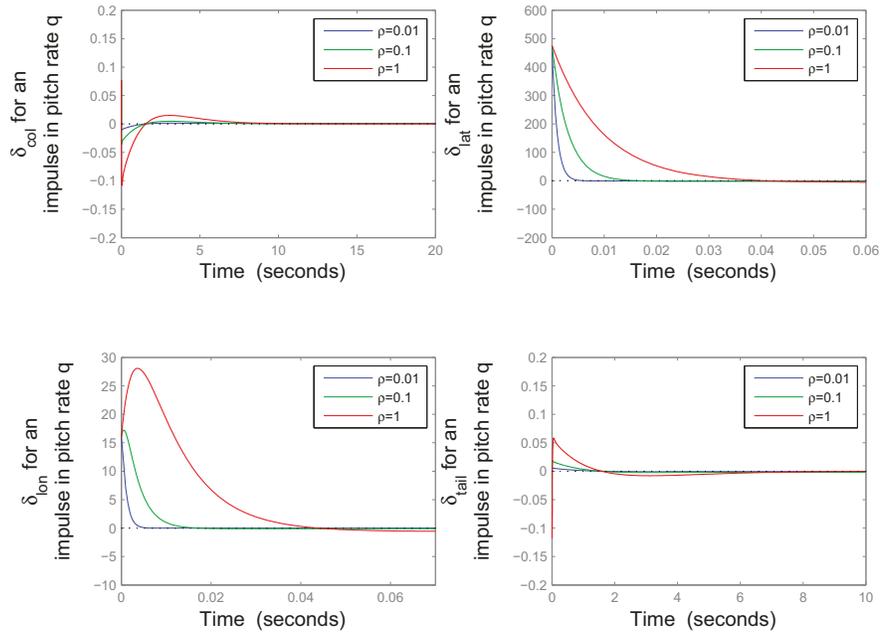


Figure 5.10: Closed-loop control inputs due to an impulse in pitch rate q

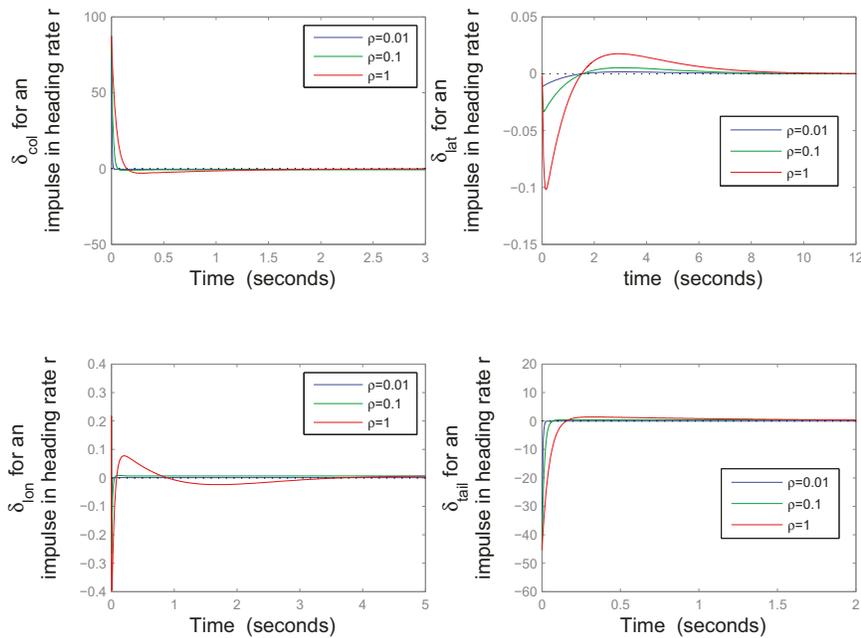


Figure 5.11: Closed-loop control inputs due to an impulse in heading rate r

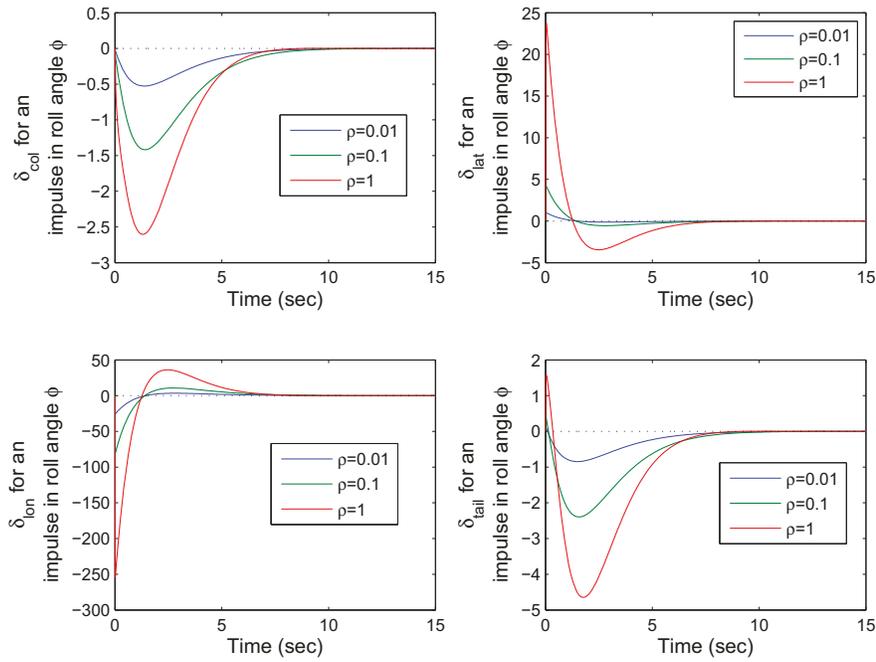


Figure 5.12: Closed-loop control inputs due to an impulse in roll angle ϕ

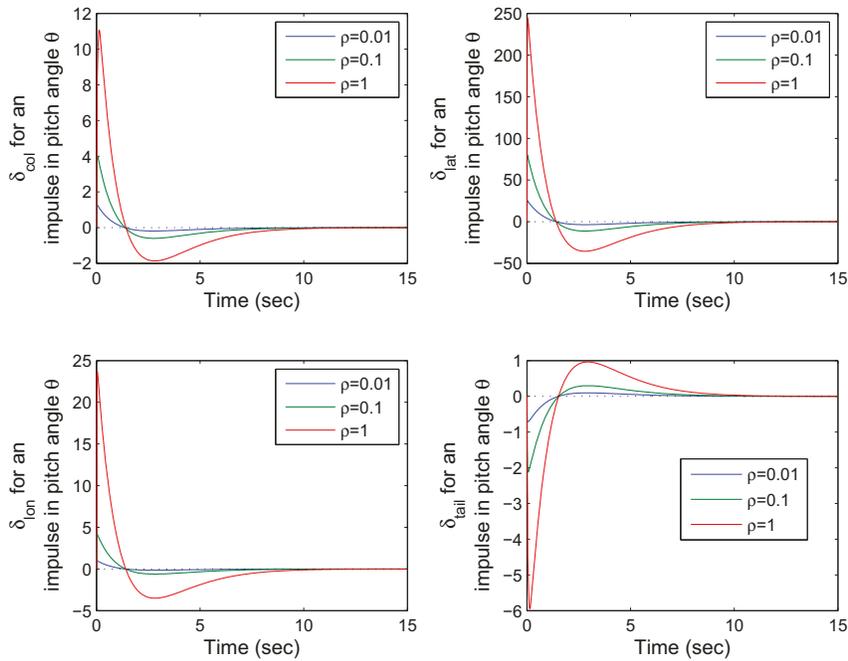


Figure 5.13: Closed-loop control inputs due to an impulse in pitch angle θ

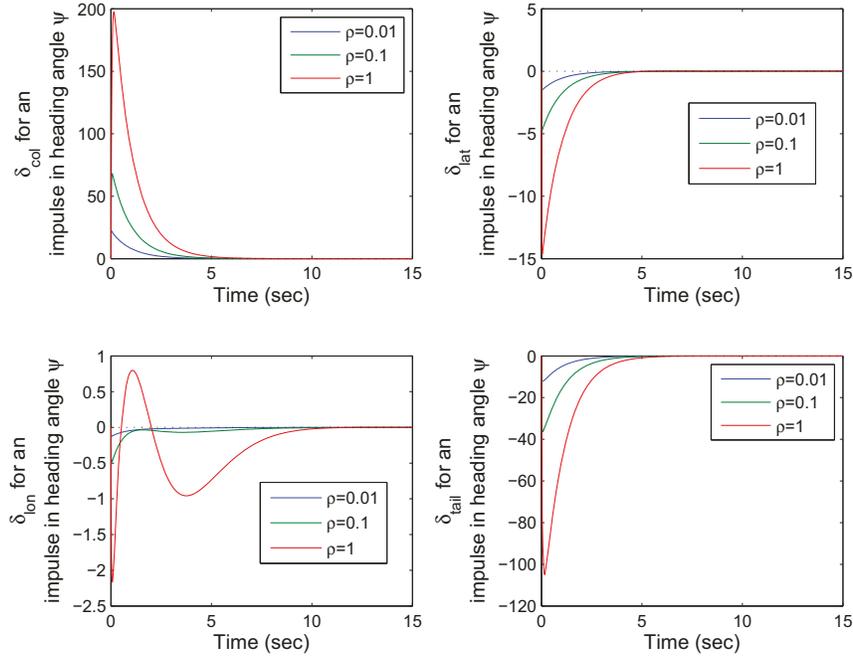


Figure 5.14: Closed-loop control inputs due to an impulse in heading angle ψ

and

$$\mathbf{y} = \mathbf{C}\Delta\mathbf{x} + \mathbf{D}\Delta\mathbf{u} + \mathbf{H}_d\boldsymbol{\vartheta}_d + \boldsymbol{\vartheta}_n \quad (5.14)$$

where $\boldsymbol{\vartheta}_d$ and $\boldsymbol{\vartheta}_n$ are the process perturbation and measurement noise and are assumed zero mean uncorrelated Gaussian white noise with covariance value matrices $\mathbf{Q}_d = \text{cov}[\boldsymbol{\vartheta}_d\boldsymbol{\vartheta}_d^T]$ and $\mathbf{R}_n = \text{cov}[\boldsymbol{\vartheta}_n\boldsymbol{\vartheta}_n^T]$.

The general form of a state observe is [68, 74]

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}\hat{\mathbf{x}} + \mathbf{B}\tilde{\mathbf{u}} + \mathbf{L}\tilde{\mathbf{y}} + \mathbf{M}\mathbf{r} \quad (5.15)$$

where

$$\mathbf{u} = \tilde{\mathbf{u}} + \mathbf{S}\mathbf{r}$$

$$\tilde{\mathbf{u}} = -\mathbf{K}\hat{\mathbf{x}}$$

$$\tilde{\mathbf{y}} = \mathbf{y} - \hat{\mathbf{y}}$$

$$\hat{\mathbf{y}} = \mathbf{C}\hat{\mathbf{x}}$$

$\hat{\mathbf{x}}$ is the the estimated state vector, and $\mathbf{L} \in \mathbb{R}^{m \times m}$ is the optimal observer

with the augmented system output of:

$$\mathbf{y} = \begin{bmatrix} \mathbf{C} & -\mathbf{DK} \\ \mathbf{0}_{9 \times 9} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \hat{\mathbf{x}} \end{bmatrix} + \begin{bmatrix} \mathbf{DS} & \mathbf{H}_d & \mathbf{I}_{9 \times 9} \\ \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.17)$$

Or by substituting $\dot{\hat{\mathbf{x}}} = \Delta \dot{\mathbf{x}} - \dot{\mathbf{e}}$ into equations (5.20) and (5.21) the augmented closed-loop plant with the error states (plant-error) is:

$$\begin{bmatrix} \Delta \dot{\mathbf{x}} \\ \dot{\mathbf{e}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{BK} & \mathbf{BK} \\ -\mathbf{LDK} & (\mathbf{A} - \mathbf{LC} - \mathbf{LDK}) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \mathbf{e} \end{bmatrix} + \begin{bmatrix} \mathbf{BS} & \mathbf{G}_d & \mathbf{0}_{9 \times 9} \\ (\mathbf{BS} - \mathbf{LDS} - \mathbf{M}) & (\mathbf{G}_d - \mathbf{LH}_d) & -\mathbf{L} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.18)$$

and

$$\mathbf{y} = \begin{bmatrix} (\mathbf{C} - \mathbf{DK}) & \mathbf{DK} \\ \mathbf{0}_{9 \times 9} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \mathbf{e} \end{bmatrix} + \begin{bmatrix} \mathbf{DS} & \mathbf{H}_d & \mathbf{I}_{9 \times 9} \\ \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.19)$$

In the linearized model of the Air Star Evolution helicopter the matrix $\mathbf{D} = 0$. In addition, the measured outputs, \mathbf{y} are assumed to have zero mean Gaussian white noise $\boldsymbol{\vartheta}_n$ only, then matrix \mathbf{H}_d is set to zero. By choosing matrices $\mathbf{M} = -\mathbf{L}$ and $\mathbf{S} = 0$, the reference points \mathbf{r} are set only through the observer. Under these assumptions, the observer and the plant equations reduces to

$$\dot{\hat{\mathbf{x}}} = (\mathbf{A} - \mathbf{BK} - \mathbf{LC}) \Delta \hat{\mathbf{x}} + \mathbf{LCx} - \mathbf{Lr} + \mathbf{L}\boldsymbol{\vartheta}_n \quad (5.20)$$

$$\Delta \dot{\mathbf{x}} = \mathbf{A}\Delta \mathbf{x} + \mathbf{B}\mathbf{u}_c + \mathbf{G}_d\boldsymbol{\vartheta}_d \quad (5.21)$$

$$\mathbf{y} = \mathbf{C}\Delta \mathbf{x} + \boldsymbol{\vartheta}_n \quad (5.22)$$

Therefore, equations (5.16) to (5.19) reduce to:

$$\begin{bmatrix} \Delta \dot{\mathbf{x}} \\ \dot{\hat{\mathbf{x}}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{BK} \\ \mathbf{LC} & (\mathbf{A} - \mathbf{BK} - \mathbf{LC}) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \hat{\mathbf{x}} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{9 \times 9} & \mathbf{G}_d & \mathbf{0}_{9 \times 9} \\ -\mathbf{L} & \mathbf{0}_{9 \times 9} & \mathbf{L} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.23)$$

$$\mathbf{y} = \begin{bmatrix} \mathbf{C} & -\mathbf{0}_{9 \times 9} \\ \mathbf{0}_{9 \times 9} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \hat{\mathbf{x}} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} & \mathbf{I}_{9 \times 9} \\ \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.24)$$

$$\begin{bmatrix} \Delta \dot{\mathbf{x}} \\ \dot{\mathbf{e}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} - \mathbf{BK} & \mathbf{BK} \\ \mathbf{0}_{9 \times 9} & (\mathbf{A} - \mathbf{LC}) \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \mathbf{e} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{9 \times 9} & \mathbf{G}_d & \mathbf{0}_{9 \times 9} \\ \mathbf{L} & \mathbf{G}_d & -\mathbf{L} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.25)$$

and

$$\mathbf{y} = \begin{bmatrix} \mathbf{C} & \mathbf{0}_{9 \times 9} \\ \mathbf{0}_{9 \times 9} & \mathbf{C} \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x} \\ \mathbf{e} \end{bmatrix} + \begin{bmatrix} \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} & \mathbf{I}_{9 \times 9} \\ \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} & \mathbf{0}_{9 \times 9} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \boldsymbol{\vartheta}_d \\ \boldsymbol{\vartheta}_n \end{bmatrix} \quad (5.26)$$

This configuration is known as LQR/LQG output feedback with control set points.

MATLAB[®] *kalman* command is used to numerically compute the optimal filter gain \mathbf{L} , which is given by

$$\mathbf{L} = \mathbf{P}\mathbf{C}^T\mathbf{R}_n^{-1} \quad (5.27)$$

where the matrix \mathbf{P} is a unique positive definite matrix that solves the following associated ARE

$$\mathbf{A}\mathbf{P} + \mathbf{P}\mathbf{A}^T + \mathbf{G}_d\mathbf{Q}_d\mathbf{G}_d^T - \mathbf{P}\mathbf{C}^T\mathbf{R}_n^{-1}\mathbf{C}\mathbf{P} = 0$$

Zero mean Gaussian white noise is simulated by generating a couple of random normal distributed vectors of 1000 data length. Matrices \mathbf{Q}_d and \mathbf{R}_n are generated as described in subsection 5.2.2 and are used as starting point to get the final matrices. Two different gain matrices \mathbf{L} are computed. The first one is base on the the following plant definition $\Delta\dot{\mathbf{x}} = \mathbf{A}\Delta\mathbf{x} + \mathbf{B}\mathbf{u}_c$ and $\mathbf{y} = \mathbf{C}\Delta\mathbf{x}$ without perturbation and measurement noise. The second gain is computed using the plant defined by equations (5.21) and (5.22) including the simulated process perturbation and measurement noise. The corresponding gain matrices \mathbf{L} are as follows:

$$\mathbf{L} = \begin{bmatrix} 8.9268 & -0.9887 & 6.3292 & 982.45 & -33.041 & 3.3930 & 0.0093 & -0.9116 & 0.0006 \\ -0.9887 & 82.164 & -195.08 & 36.223 & 981.32 & -543.37 & 0.9054 & 0.0095 & -0.0096 \\ 6.3292 & -195.08 & 23943 & -2.0771 & 0.2314 & -8636.4 & 0.0397 & 0.0010 & -0.0005 \\ 982.45 & 36.223 & -2.0771 & 202200 & 0.3640 & -59.245 & 0.9997 & 0.0046 & 0 \\ -33.041 & 981.32 & 0.2314 & 0.364 & 61510 & 9.3437 & -0.0145 & 0.9975 & -0.0594 \\ 3.393 & -543.37 & -8636.4 & -59.245 & 9.3437 & 9318.1 & 0.0891 & 0.0614 & 0.9970 \\ 0.0093 & 0.9054 & 0.0397 & 0.9997 & -0.0145 & 0.0891 & 0.4126 & -0.0005 & 0.0059 \\ -0.9116 & 0.0095 & 0.0010 & 0.0046 & 0.9975 & 0.0614 & -0.0005 & 0.4108 & 0.0005 \\ 0.0006 & -0.0096 & -0.0005 & 0 & -0.0594 & 0.997 & 0.0059 & 0.0005 & 0.9999 \end{bmatrix} \quad (5.28)$$

and

$$\mathbf{L}_n = \begin{bmatrix} 129.36 & 0 & -0.0009 & -0.1118 & 0.0114 & -0.0001 & -0.0001 & -4.898 & 0 \\ 0 & 129.28 & 0.0165 & -0.0214 & -0.4527 & 0.4241 & 4.8892 & -0.0011 & 0.0011 \\ -0.0009 & 0.0165 & 126.45 & 0.0001 & -0.0012 & 0.3223 & 0.2917 & 0.0029 & 0.0011 \\ -0.1118 & -0.0214 & 0.0001 & 4068.8 & -5.6163 & 0.0715 & 0.9656 & 0.0052 & -0.0001 \\ 0.0114 & -0.4527 & -0.0012 & -5.6163 & 125.19 & -0.0049 & -0.0333 & 0.4833 & -0.0289 \\ -0.0001 & 0.4241 & 0.3223 & 0.0715 & -0.0049 & 128.07 & 0.0064 & 0.0295 & 0.4951 \\ -0.0001 & 4.8892 & 0.2917 & 0.9656 & -0.0333 & 0.0064 & 129.01 & -0.0001 & 0 \\ -4.898 & -0.0011 & 0.0029 & 0.0052 & 0.4833 & 0.0295 & -0.0001 & 129.01 & 0 \\ 0 & 0.0011 & 0.0011 & -0.0001 & -0.0289 & 0.4951 & 0 & 0 & 129.1 \end{bmatrix} \quad (5.29)$$

respectively.

Next section shows the corresponding simulation results for the two computed filter gains \mathbf{L} , \mathbf{L}_n , and the LQR computed gain \mathbf{K} in an output feedback configuration with set points.

5.2.3 LQR/LQG Simulation Results

Based on the optimal computed gain matrices \mathbf{K} and \mathbf{L} , three sets of simulation are done. The first simulation set does not include process perturbations and measurement noise. It is done by changing one reference at a time with a value close to the hover condition but allowing the helicopter model to perform forward, lateral, heave and heading flight. On the second set of simulation the same set points are used with the addition of process disturbances and measurement noise. The last set of simulation is done by using the recorded states

of the flight test as the set point and having the controller to follow them.

For the first simulation set, with helicopter initially in hover, a set point of 0.5 m/s in the body velocity u is introduced with the rest of the state at equilibrium point value -see figures 5.16 and 5.17.

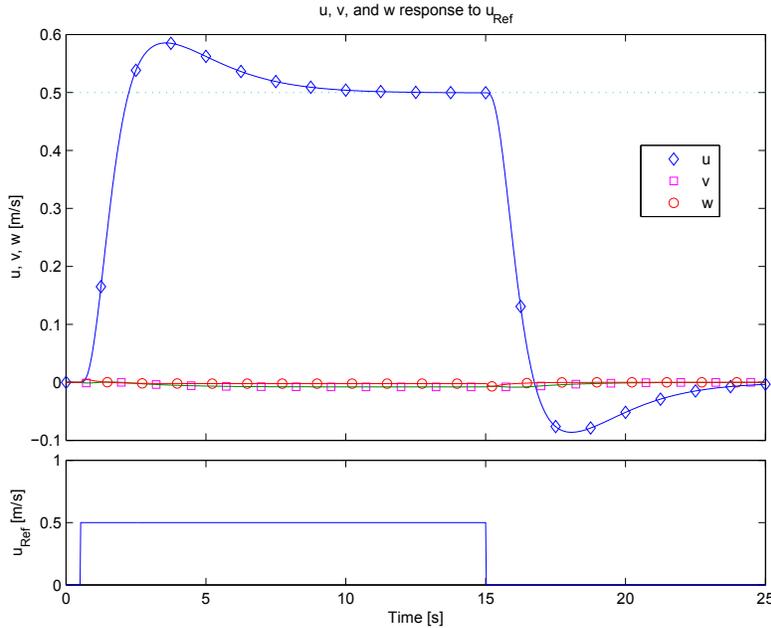


Figure 5.16: Response of body velocities u , v and w due to set point in body velocity u . Symbols are used only to denote line.

As can be seen from figure 5.16 the forward velocity of 0.5 m/s imposed by the set point is reached after 10 second. The other two body velocities v and w remain unchanged as expected. Figure 5.17 shows the corresponding change in the pitch rate q . As is expected after the body velocity u has reached the set point, the pitch rate decreases towards zero. Also from the same figure it is possible to see the overshoot on the pitch rate. The other angular velocities do not exhibit any change.

From figure 5.18 the required set point is reached in about 7.5 seconds. The roll rate exhibit similar behavior as the pitch rate. It reaches a maximum and minimum value about 0.08 and -0.08 rad/s in the transitions of the lateral velocity set point, -see figure 5.19. Figure 5.20 shows the corresponding roll angle associated to a lateral motion. Figure 5.21 shows the required set point of 0.5 m/s for the heave velocity. Here it is possible to see that the response of

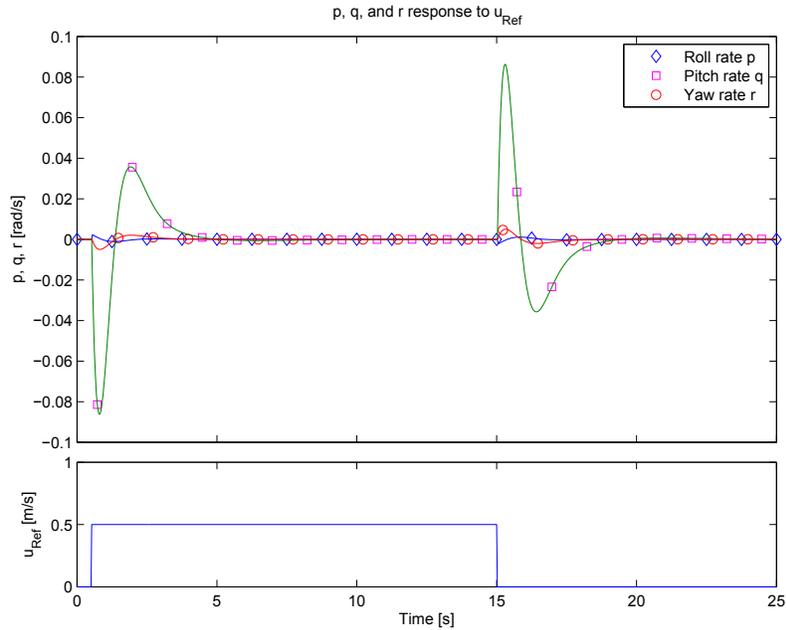


Figure 5.17: Response of angular velocities p , q and r due to set point in body velocity u . Symbols are used only to denote line.

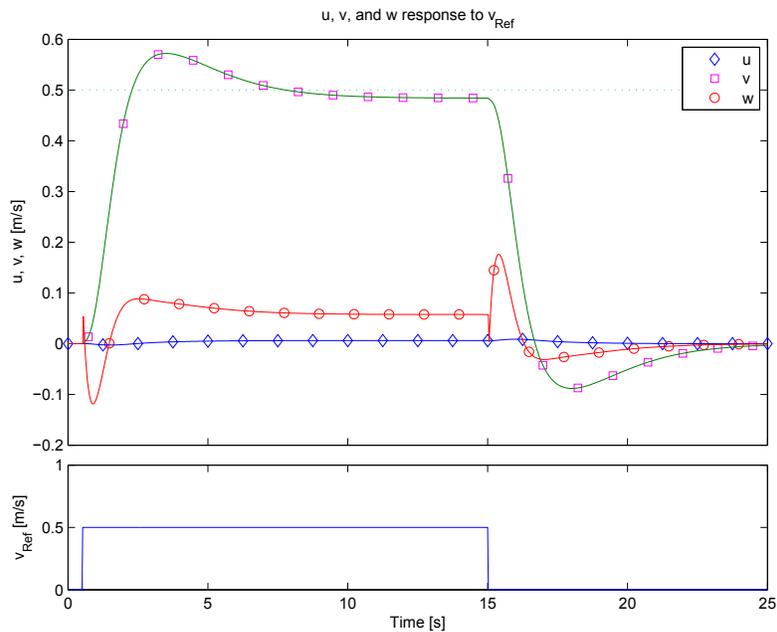


Figure 5.18: Response of body velocities u , v and w due to set point in body velocity v . Symbols are used only to denote line.

the heave velocity is faster than the other two cases. Nevertheless, it does not reach the required 0.5 m/s. Instead, it just reach 0.4 m/s. Also it is possible

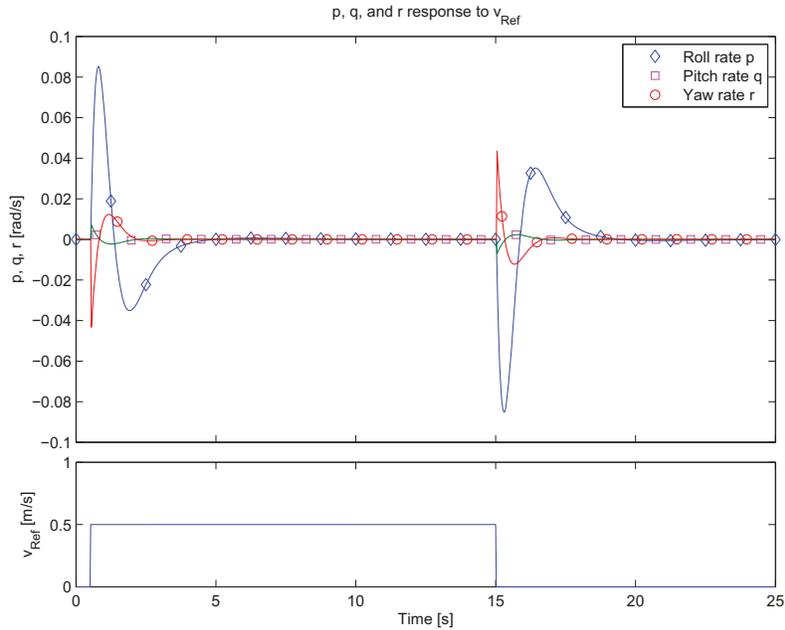


Figure 5.19: Response of angular velocities p , q and r due to set point in body velocity v . Symbols are used only to denote line.

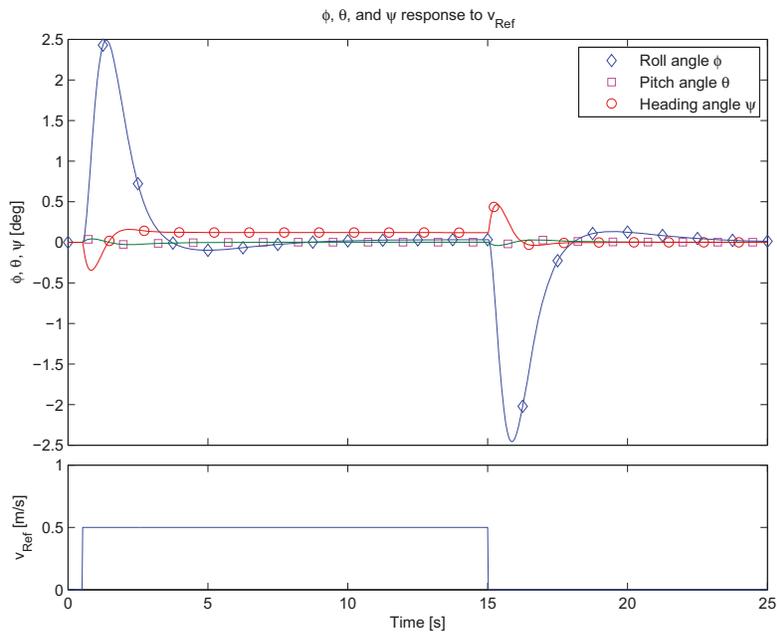


Figure 5.20: Response of the Euler angles ϕ , θ and ψ due to set point in body velocity v . Symbols are used only to denote line.

to see from figure 5.22 that the angular rates are almost unchanged. Finally, the response to a heading set point of 10 degrees can be seen from figure 5.23.

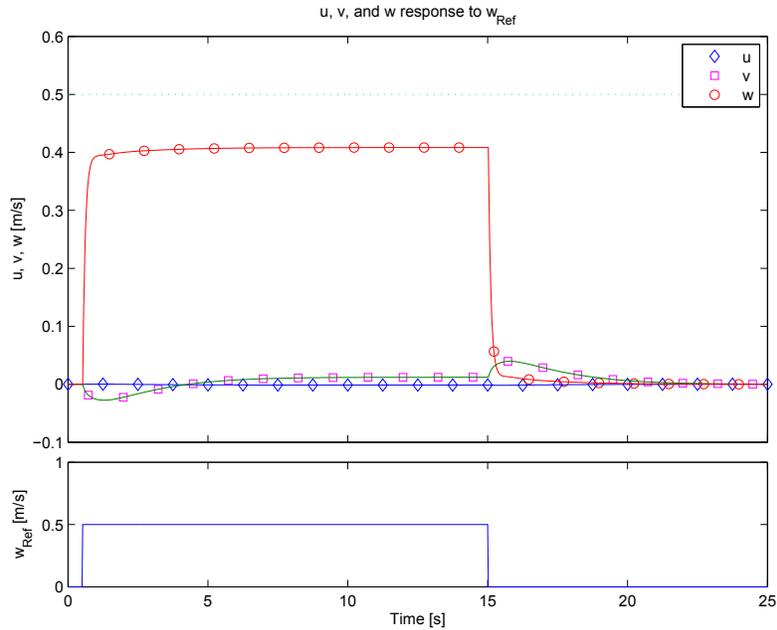


Figure 5.21: Response of body velocities u , v and w due to set point in body velocity w . Symbols are used only to denote line.

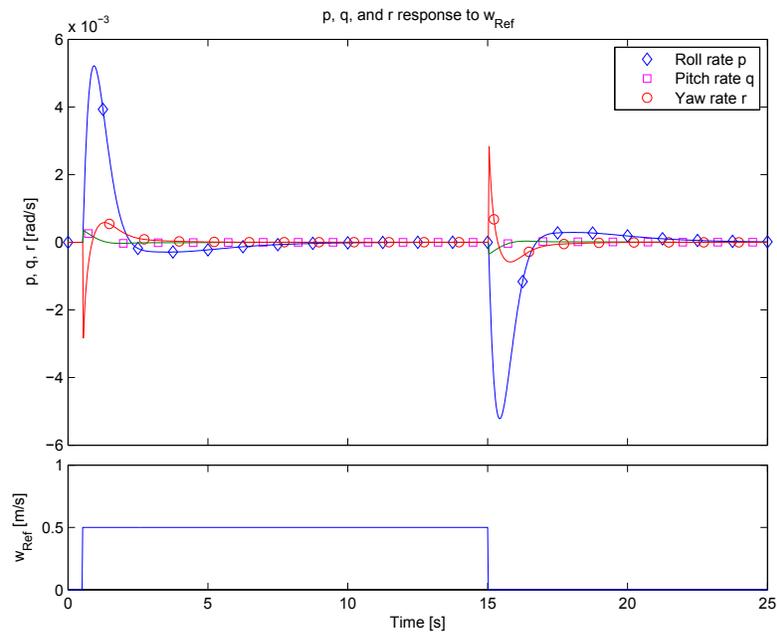


Figure 5.22: Response of angular velocities p , q and r due to set point in body velocity w . Symbols are used only to denote line.

The desired set point is reached in about 8 seconds. The corresponding yaw rate r , has a peak of 0.06 rad/s before reaches a value of zero –see figure 5.24.

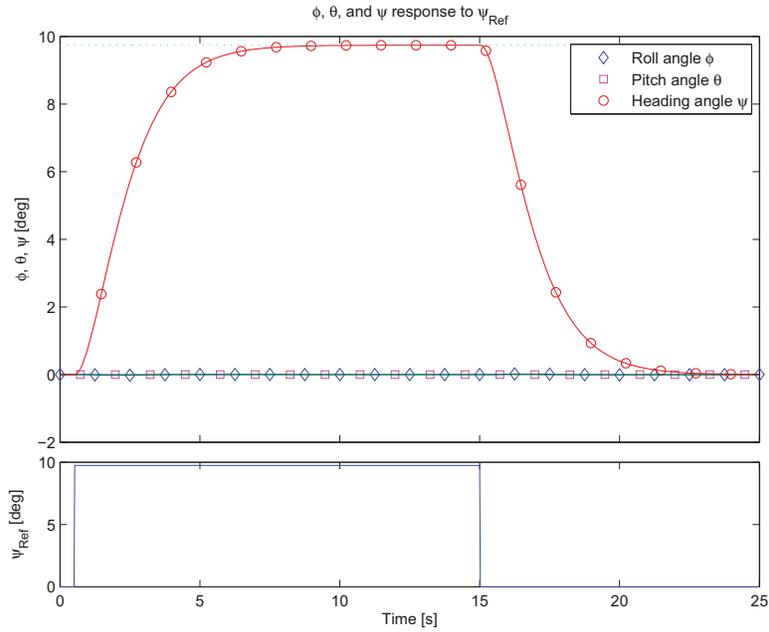


Figure 5.23: Response of the Euler angles ϕ , θ and ψ due to set point in heading angle ψ . Symbols used are only to denote line.

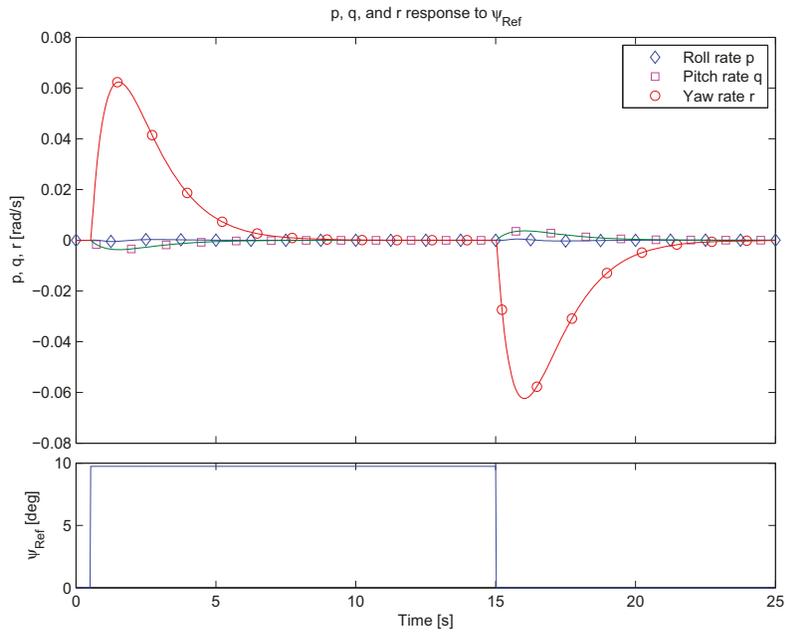


Figure 5.24: Response of angular velocities p , q and r due to set point in heading angle ψ . Symbols are used only to denote line.

The body velocities u , v and w practically remain at zero as can be seen from figure 5.25

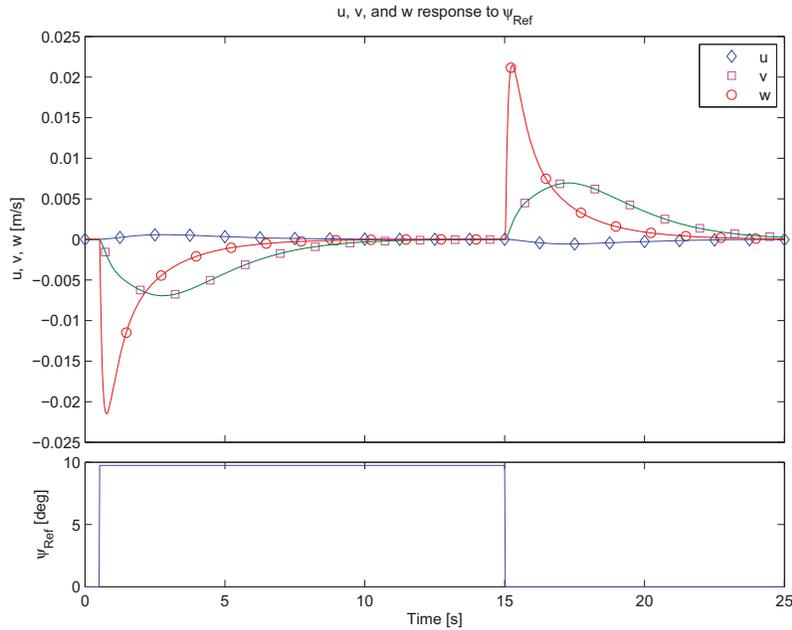


Figure 5.25: Response of body velocities u , v and w due to set point in heading angle ψ . Symbols are used only to denote line.

On the second set of simulations the same set points are applied. Zero-mean Gaussian white noise is added to both, the process and the measurements with a variance of $\sigma_p = 0.003$ and $\sigma_m = 0.002$.

Figures 5.26 to 5.28 show the response of all states to a set point in the body velocity u of 0.5 m/s, with the initial condition of the remaining states at equilibrium point value. As can be seen from figure 5.26 the set point is reached in about 8 seconds, and the overshoot has been eliminated. Figures 5.27 and 5.28 show the response of the angular velocities and the Euler angles respectively. As expected, the angular rates remain unchanged with the exception of the pitch rate. Once the set point is set, the pitch rate increases to a maximum of -0.03 rad/s. Then decreases to a value of zero in about 1.5 seconds. It remains at zero during the whole set point until time is equal to 15 seconds when the set point is changed to zero. Then the pitch rate increases up to 0.03 rad/s and decreases to a value of zero after 1.5 seconds. The pitch angle presents a value different from zero as expected, however this value is not clear determine due to the signal noise pollution –see figure 5.28.

The lateral velocity presents a similar response as in the case of the simu-

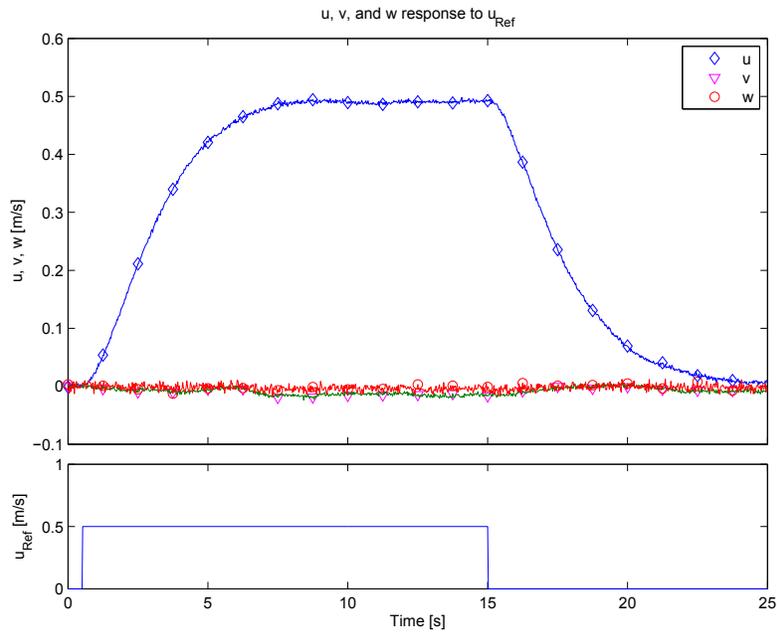


Figure 5.26: Response of body velocities u , v and w due to set point in the body velocity u added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

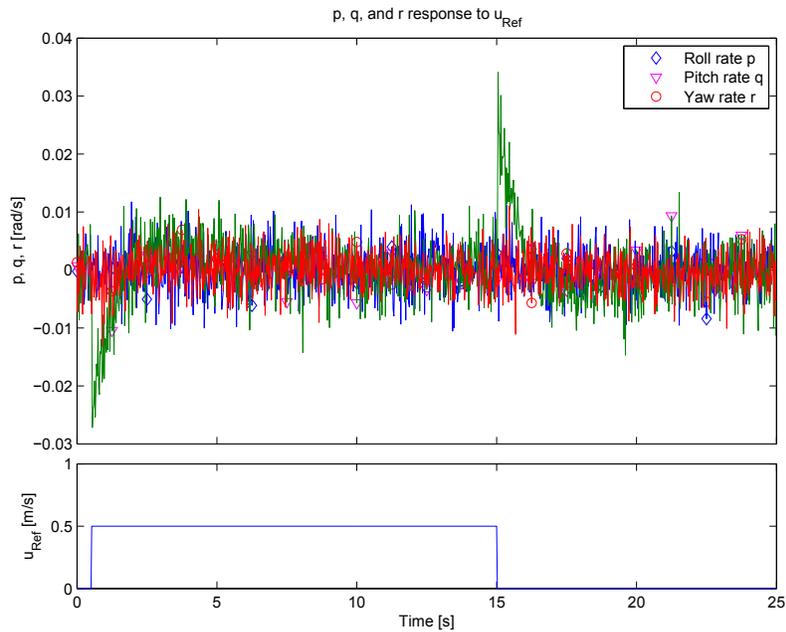


Figure 5.27: Response of body angular velocities p , q and r due to set point in the body velocity u added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

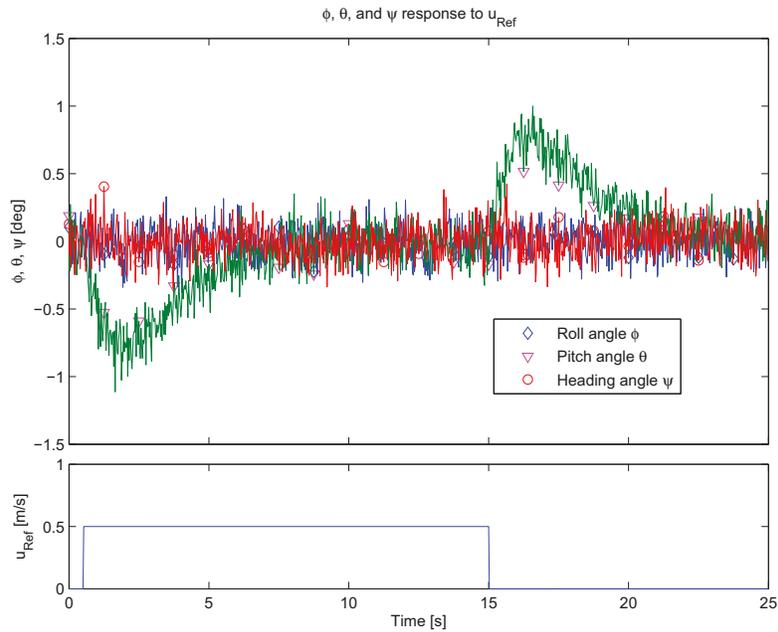


Figure 5.28: Response of Euler angles ϕ , θ and ψ due to set point in the body velocity u added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

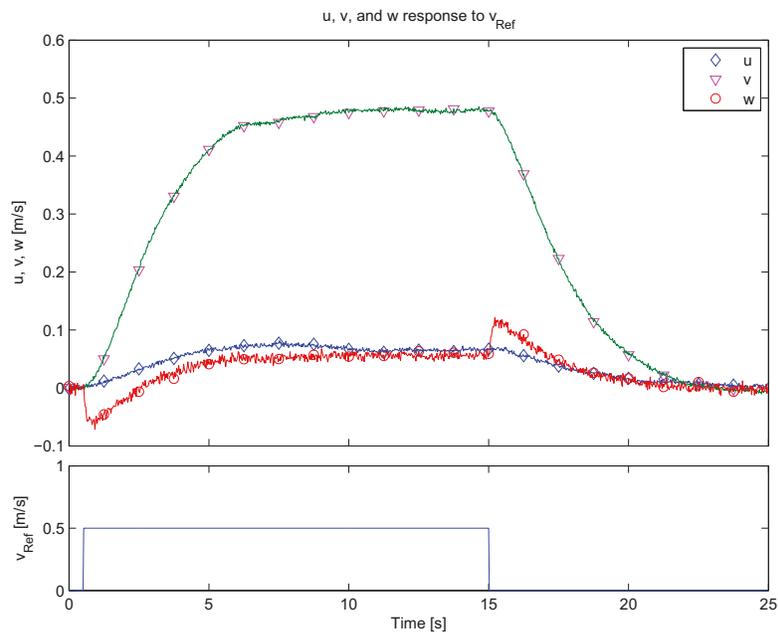


Figure 5.29: Response of body velocities u , v and w due to set point in the body velocity v added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

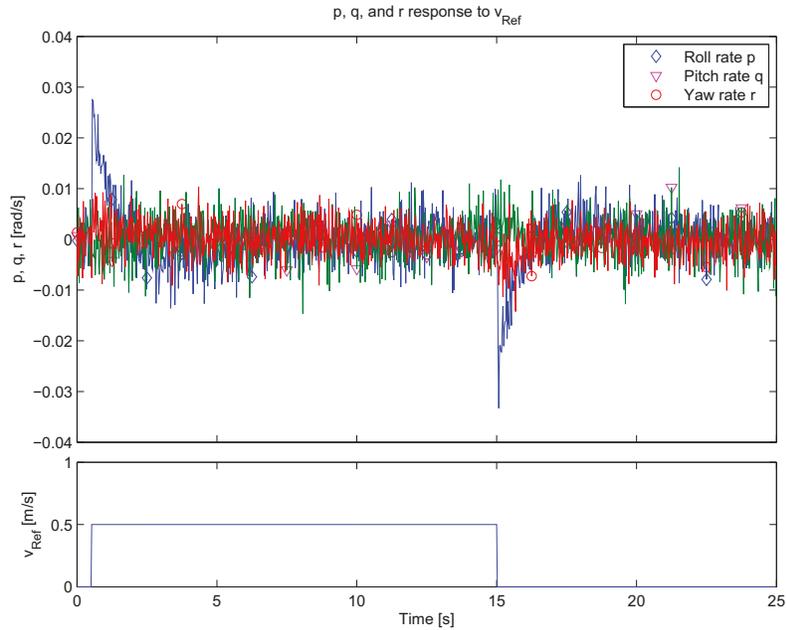


Figure 5.30: Response of body angular velocities p , q and r due to set point in the body velocity v added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

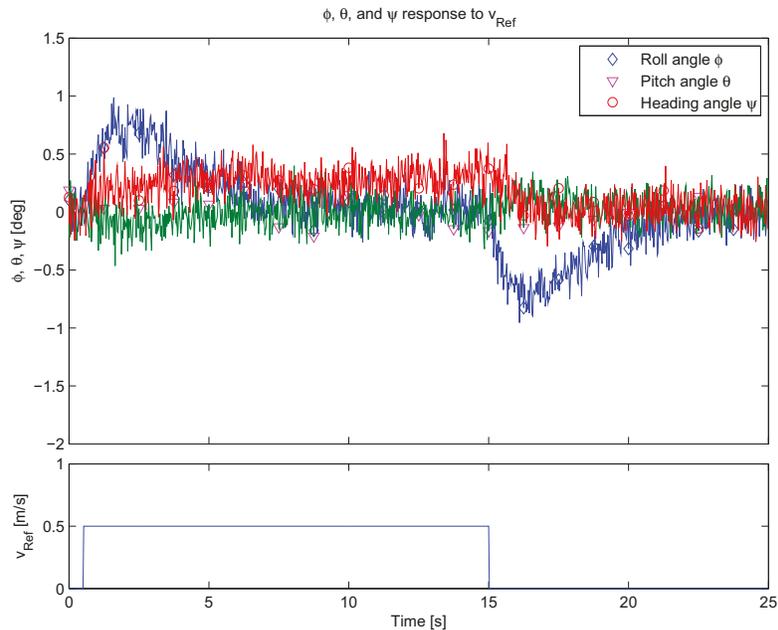


Figure 5.31: Response of Euler angles ϕ , θ and ψ due to set point in the heading angle v added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

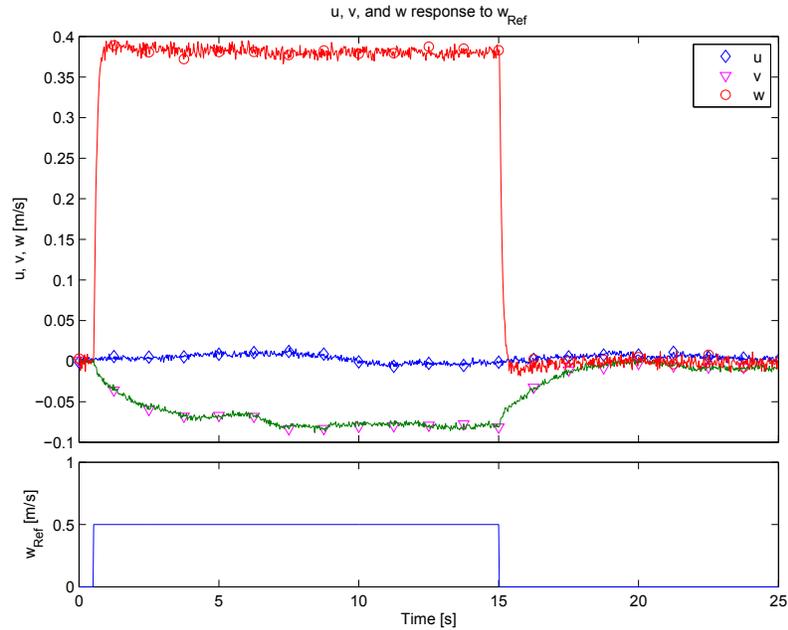


Figure 5.32: Response of body velocities u , v and w due to set point in the body velocity w added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

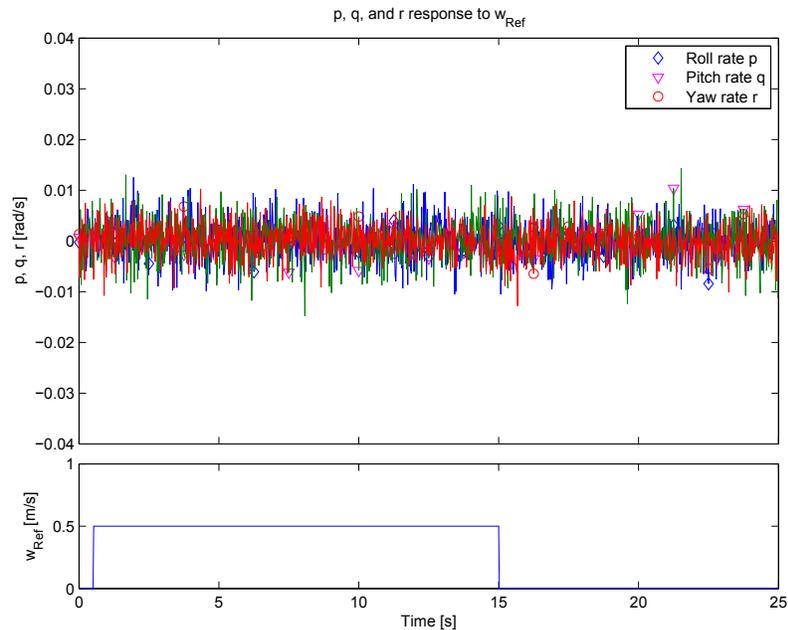


Figure 5.33: Response of body angular velocities p , q and r due to set point in the body velocity w added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

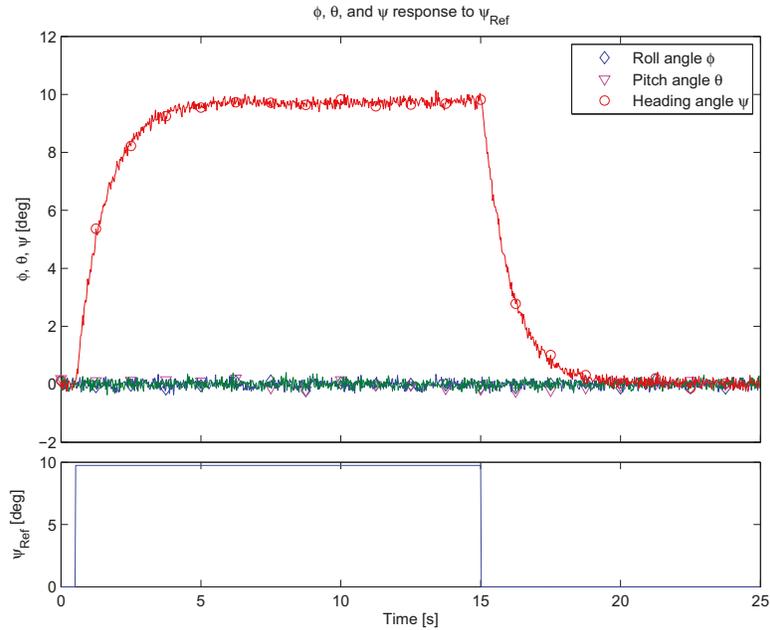


Figure 5.34: Response of Euler angles ϕ , θ and ψ due to set point in the heading angle ψ added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

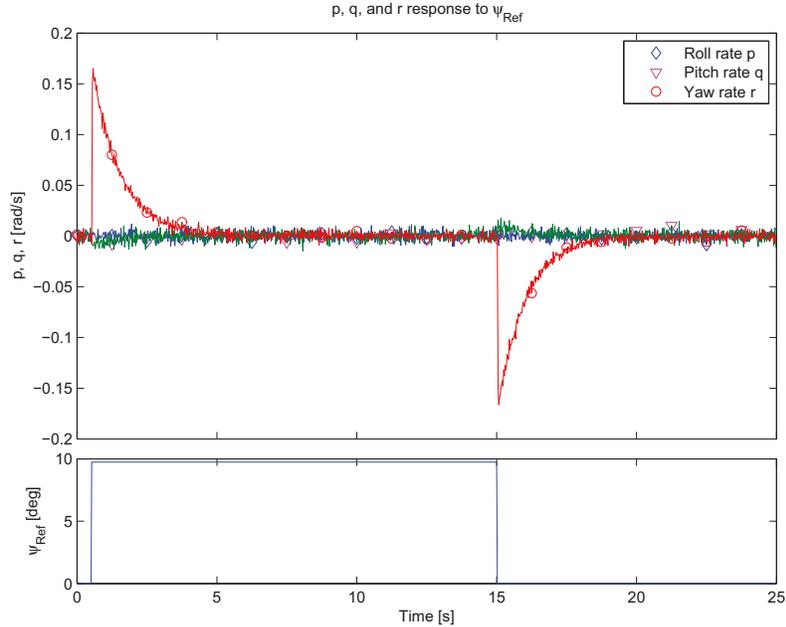


Figure 5.35: Response of body angular velocities p , q and r due to set point in the heading angle ψ added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

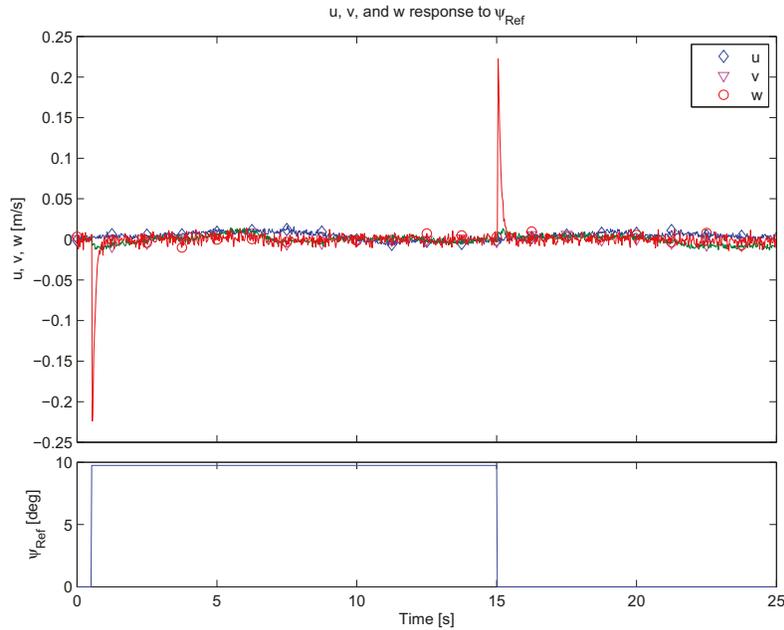


Figure 5.36: Response of body velocities u , v and w due to set point in the heading angle ψ added with process and measurement noise $\sigma_p = 0.003$ and $\sigma_m = 0.002$. Symbols are used only to denote line.

lation without noise. It reaches the desired set point in about 9 seconds –see figure 5.29. In addition from the same figure is possible to see that overshoot has been eliminated. These two effects are attributed to the change in the filter gain matrix \mathbf{L} . As can be seen from figures 5.30 and 5.31 the angular rates and the roll angle present a similar response as before. The heave velocity response can be seen in figure 5.32. This response is almost identical to the response without noise, reaching a steady state value of 0.4 m/s. Finally for the second set of simulations, the heading response can be seen on figures 5.34 to 5.36. The desired heading set point is reached in about 5 second, approximately 3 seconds faster than the simulation without noise –see figure 5.34. The corresponding heading rate exhibits a peaks of 0.15 rad/s and -0.15 rad/s before decreasing to zero in the transitions of the set point, as can be seen from figure 5.35. The body velocities u , and v remain unchanged –see figure 5.36. From the same figure it is apparent that the heave velocity w presents a peak of -0.25 m/s for a short duration at about 500 ms in the transition of the set point. These can be interpreted as a change in the counter torque exerted by the tail rotor produces an instantaneous change in the heave velocity.

In the third set of simulations the states u , v , w , p , q , r , ϕ , θ and ψ are taken from the first flight test and are used as set points for the simulated plant and the controller. The response of the plant plus the controller and the flight set points are plotted in figures 5.37 to 5.45.

For the body forward velocity u , the controller and the simulated plant properly follows the real flight velocity trajectory although there is an overshoot at about 12 seconds –see figure 5.37. Although the plant follows the

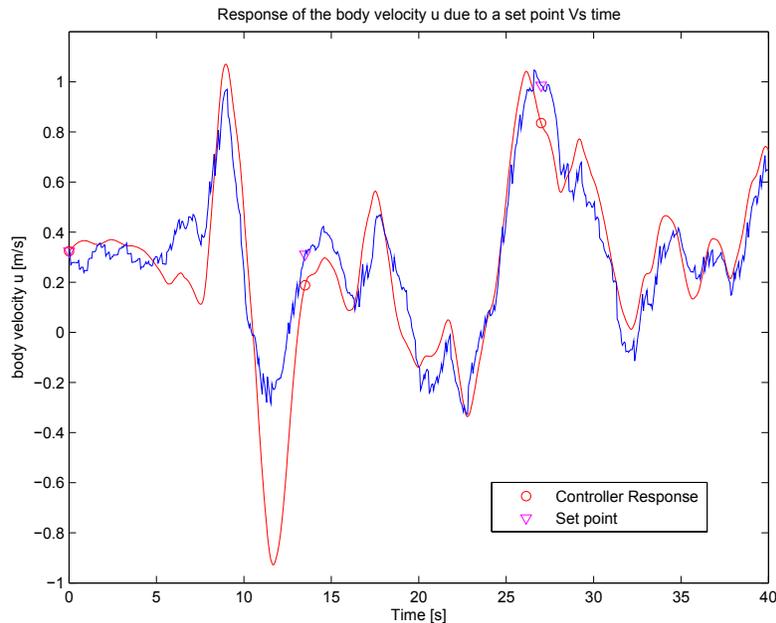


Figure 5.37: Comparison between controller response of the body velocity u and the body velocity u of the flight test. Symbols are used only to denote line.

trend of the real lateral velocity v , there is an average shift in the response of about -0.2 m/s from the real trajectory as shown in figure 5.38. This may be caused due to an improper selection of matrices \mathbf{Q}_d and \mathbf{R}_n that does not capture the whole dynamics of the system. The heave velocity w , in figure 5.39, follows the trend of real velocity w , but as in the case of forward velocity u , there is some overshoot.

For the angular velocities, the controller properly follows the imposed set points as can be seen from figures 5.40, 5.41 and 5.42. Despite the quick transitions in the angular velocities, the observer properly computes the current value of the states without overshoots. Finally, the roll, pitch and heading angle responses are shown in figures 5.43, 5.44 and 5.45. In the case of the roll angle ϕ , the controlled follows the trend of the flight roll angle. However,

the response seems to have a positive offset over the whole response as can be seen in figure 5.43. For the pitch angle θ , the controller properly follows the flight pitch angle as can be seen from figure 5.44. Finally the heading angle ψ generated by the controller virtually mimics the flight heading angle as it is shown in figure 5.45.

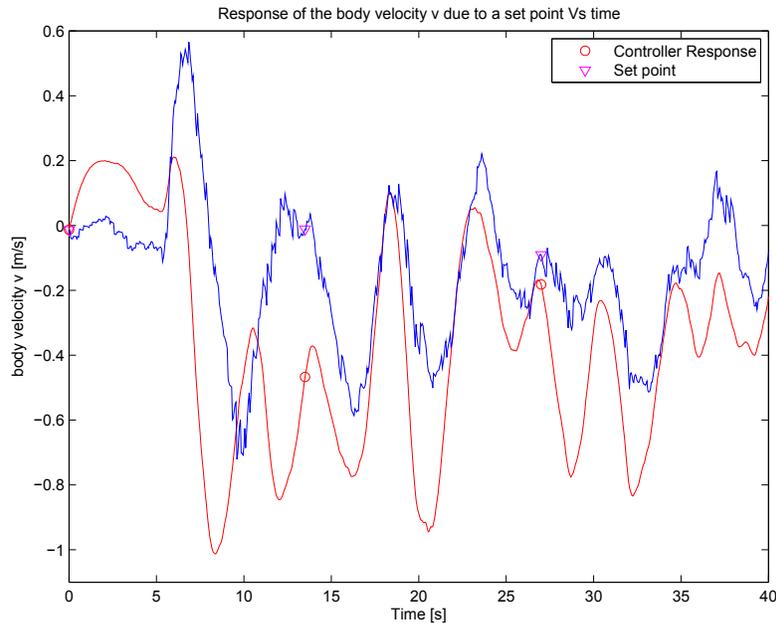


Figure 5.38: Comparison between controller response of the body velocity v and the body velocity v of the flight test. Symbols are used only to denote line.

From the simulation results it is shown that the helicopter can be controlled about hover. The impulse response of the stabilized model shows that the best hover response is reached for a ρ value of 0.01. The optimal control law $\mathbf{u} = -\mathbf{K}\mathbf{x}$ in combination with the Kalman observer connected in an output feedback configuration with reference points allows the helicopter to adequately follow the states trajectories of the flight test. Additional analysis can be performed to modify matrices \mathbf{Q}_d and \mathbf{R}_n to yield faster responses and reduce overshoots.

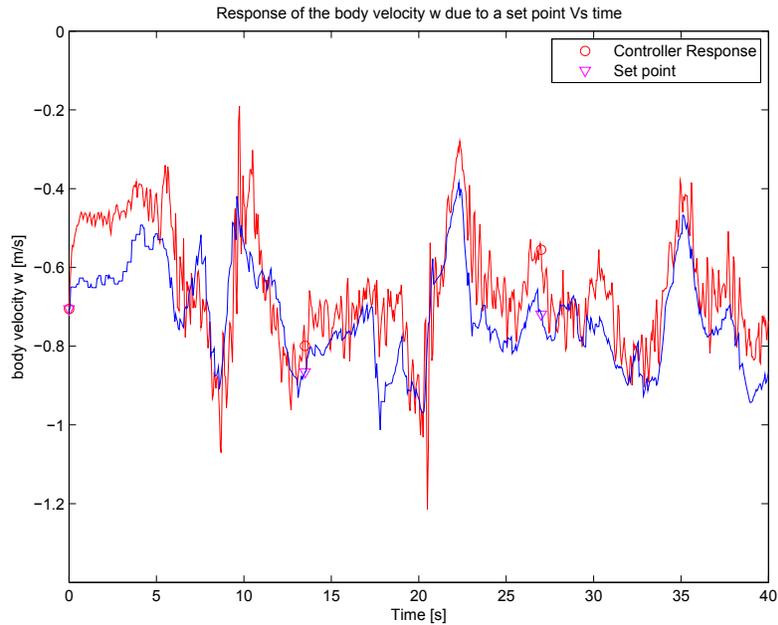


Figure 5.39: Comparison between controller response of the body velocity w and the body velocity w of the flight test. Symbols are used only to denote line.

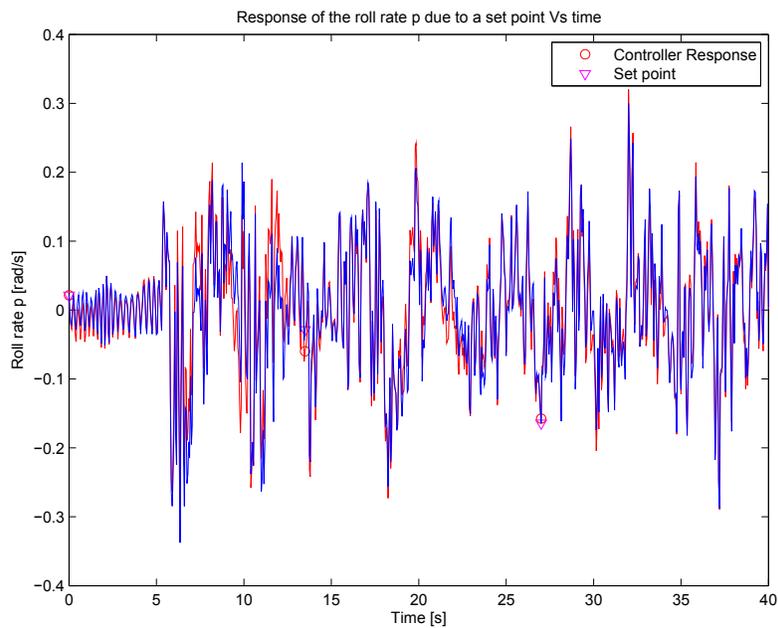


Figure 5.40: Comparison between controller response of the roll rate p and the angular velocity p of the flight test. Symbols are used only to denote line.

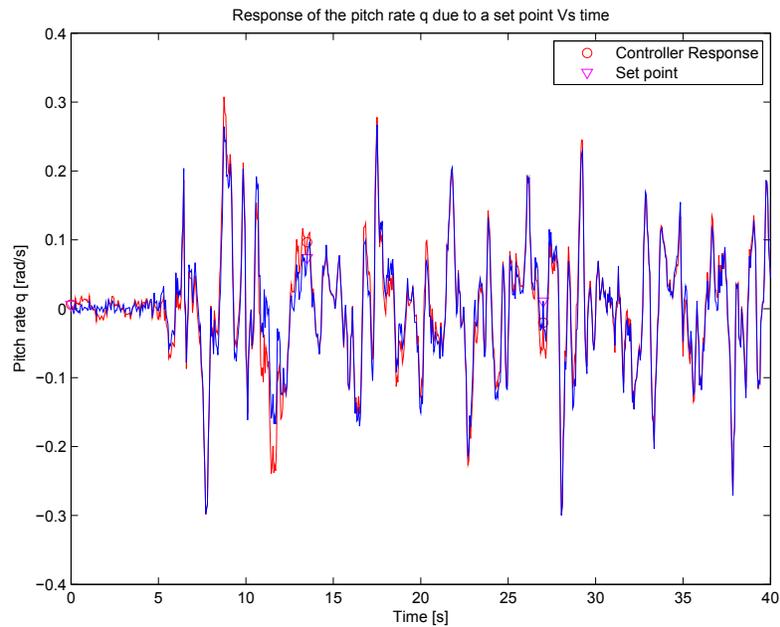


Figure 5.41: Comparison between controller response of the pitch rate q and the angular velocity q of the flight test. Symbols are used only to denote line.

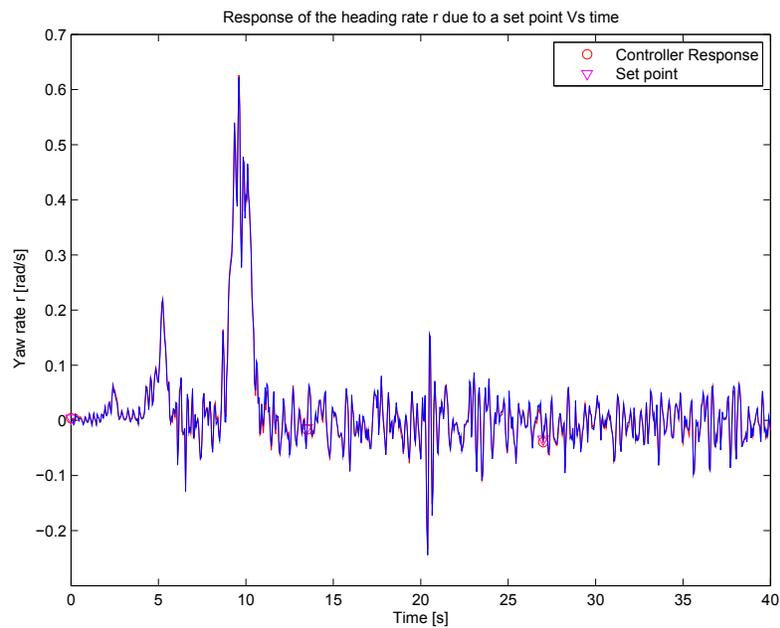


Figure 5.42: Comparison between controller response of the yaw rate r and the angular velocity r of the flight test. Symbols are used only to denote line.

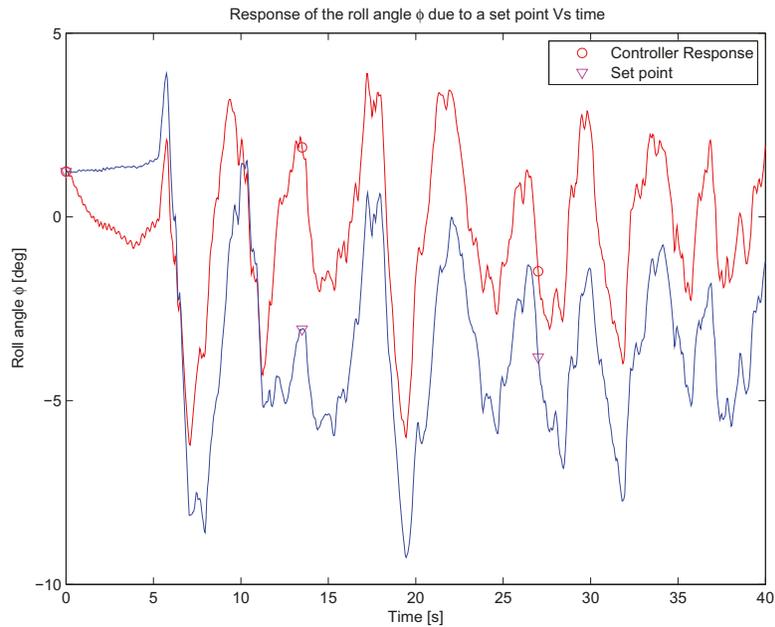


Figure 5.43: Comparison between controller response of the roll angle ϕ and the roll angle ϕ of the flight test. Symbols are used only to denote line.

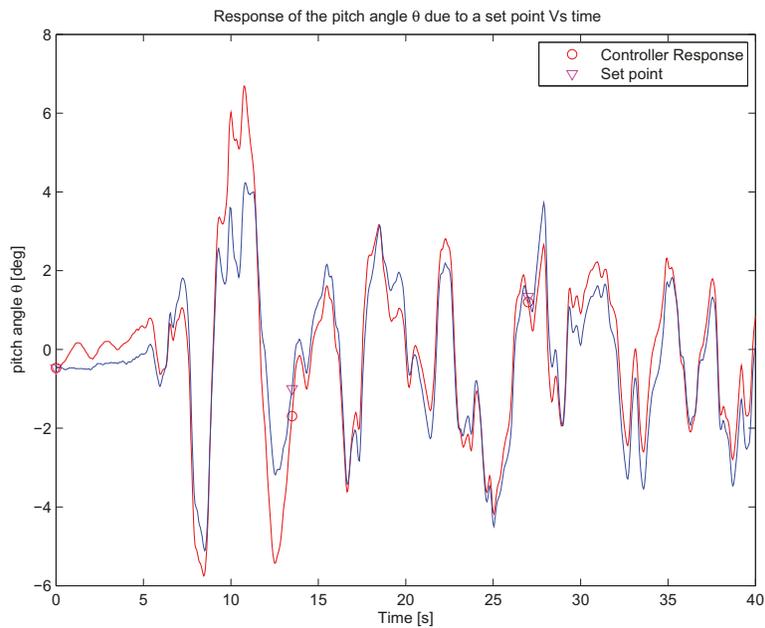


Figure 5.44: Comparison between controller response of the pitch angle θ and the pitch angle θ of the flight test. Symbols are used only to denote line.

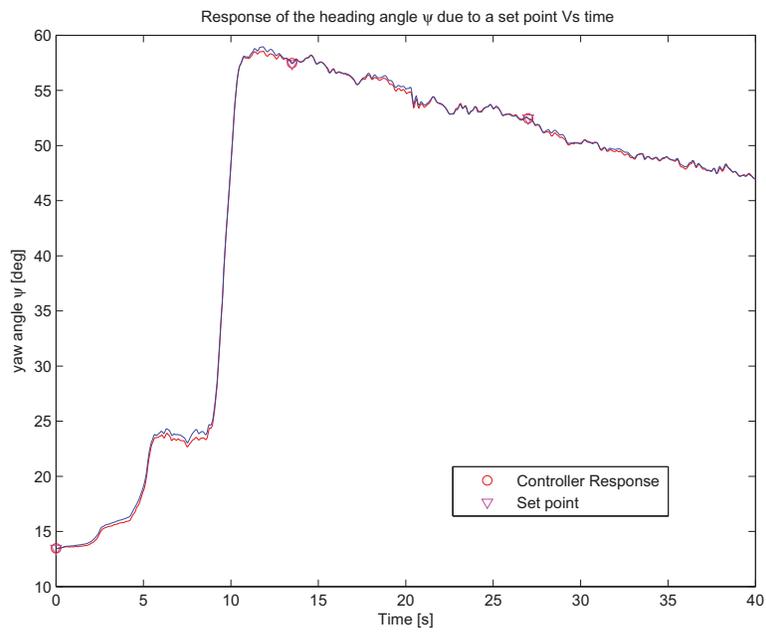


Figure 5.45: Comparison between controller response of the yaw angle ψ and the heading angle ψ of the flight test. Symbols are used only to denote line.

Chapter 6

Conclusions

The dynamic model of the Air Star Evolution helicopter proposed in [22] is presented in detail. The rotational dynamic model includes the moment of inertia of the rotor. Rotor aerodynamic forces are derived by using the blade element theory and the momentum theory assuming small angle of the free air stream velocity U_∞ respect to the horizontal line.

The mass moment of inertia tensor of the helicopter, the main rotor slope lift coefficient C_{L_α} and the main rotor zero-lift drag torque C_{D_o} have been determined through two simple but effective experiments. These are performed using the Moment Of Inertia testbed (MOI) and the Hardware in the Loop (HIL) force testbed both of which were developed for this work. The angular position as a function of time in the MOI experiment are determined. A 2000 division quadrature encoder provides the necessary resolution to record the angular position as a function of time and this is used to estimate the corresponding moments of inertia of the principal body axis of the helicopter. The estimate value of the inertia tensor, listed in table 2.1 is in the range of other helicopters of the same class. Special metal frames are designed to attach the helicopter to measure the roll moment I_x and the pitch moment I_y as can be seen in figures 3.2 and 3.3. Although the MOI testbed works well, a possible improvement is an air suspension system or magnetic suspension system be implemented to eliminate the uncertainties of the static and dynamic friction coefficients in the pulley bearings.

The force testbed in combination with two desktop computers constitute the Hardware In the Loop that is used to determine the main rotor aerodynamic slope coefficient C_{L_α} and the zero-lift drag coefficient C_{D_o} . The equa-

tions of the force testbed are determined and the data of the eight load cells are recorded. Based on the measured thrust force and the helicopter mathematical model, parameters are fit resulting in a lift slope coefficient $C_{L\alpha}$ of 6.4 rad^{-1} , as can be seen from figure 3.18. This value is in the range of similar blade profile and wing span. Similarly, using the helicopter equation for the drag torque and the recorded data of the drag torque, the zero-lift drag torque coefficient C_{D_0} is found with a value of 0.012, -see figure 3.18. This value is also in the range of similar helicopters with similar blade span and blade aerodynamic profile.

The Avionics instrumentation is reliable as an integrated system. Given the design constraints, good integrations of all the subsystems is achieved as can be seen in figures 4.2. The battery monitor system properly estimates the voltages, currents, and remaining battery charge of the three electrical systems, the rotor system, the servo system and the avionics system. The outliers on the Avionics battery and the rotor battery seen on figure 4.11 are caused by unavoidable momentary glitches in communication. However, the system is reliable and accurately monitoring the electrical status of the helicopter.

The radio communication link and its associated protocol are also found to be reliable. The protocol described in chapter 4, correctly manages to recover synchronization in the communication whenever glitches or gaps are presented.

Special attention is paid to the position reported by the GPS, particularly the altitude data. Statistical analysis is done on the GPS position. It is found that the latitude and longitude data are correlated. This produces that the Circle Error Probable (CEP) elongates proportional to the magnitude of the correlation between longitude and latitude, resulting in an Ellipse Error Probable (EEP) as can be seen in the 3D histograms of figures 4.23 to 4.25 as well as in figures 4.26 and 4.27. In addition, the altitude presents an important drift of about 16 m as can be seen in figures 4.28 to 4.30. Therefore the resultant static position of the GPS in a 24 hours period is an ellipsoid of a volume of 43.62 m^3 for 50% probability and a volume of 261.81 m^3 for 95% probability –see figures 4.28 to 4.30 and table 4.10. It is found that the drift rate in altitude is about 0.162 m/s for some cases, as shown in figures 4.31 to 4.33 and table 4.11. The direction of drift is random as can be seen in these figures. Although the altitude drift is not instantaneous and its direction is random, in time its accumulative effect or trend needs to be considered for

long flights. A safe corridor of at least 35 m in diameter must be set for flight formation in long periods of time.

Data from the helicopter instrumentation is validated and avionics box successfully record all helicopters states and parameters as the main rotor RPM, body velocities u , v , and w , Euler angles ϕ , θ , and ψ angular rate of change p , q and r , body accelerations a_x , a_y and a_z , and pilot's commands – see figures 4.43 to 4.54. All recorded parameters are used for system identification purposes.

The Ground Station is reliable and accurately displays all the helicopter parameters and electric system status of the helicopter. The virtual instruments displays the data from the onboard instrumentation and received them correctly. Controller gains in the testbed or in real flight, are properly transmitted to the onboard computer in real time. This feature reduces the tuning time significantly compared to restarting each time, as is demonstrated during the testbed controller characterization. In addition, the modular design in the software and communication protocol facilitate the additions of new commands or features for future applications.

The linear dynamic of the helicopter is achieved using the small perturbation method. Hover flight regime is selected as operating point. The optimal LQR properly stabilized the plant about the equilibrium point as can be seen from the simulation result of the 36 impulse responses shown in figures 5.6 to 5.14. The body velocity u – see figure 5.6 has no overshoot response for a ρ factor of 0.01. Similar results are obtained for the lateral body velocity v and the heave velocity w as can be seen from figures 5.7 and 5.8. The same good results are found for the angular velocities p , q , and r . As can be seen from figures 5.9 to 5.11, the best response for all the angular velocities is found with a ρ factor of 0.01. The same best impulse response of the Euler angles is obtained with the same ρ factor of 0.01 – see figures 5.12 to 5.14

In addition to the Linear Quadratic Regulator, an LQG compensator in Output Feedback configuration with external references is implemented based on the separation principle. A full state Kalman observer is implemented. Three sets of simulation are done. The first one does not include process and measurement noise. One reference is changed at a time for all states to examine the response. In figure 5.16 it is possible to see the response of the

body velocities u , v and w to a reference forward body velocity u of 0.5 m/s. The body forward velocity reaches its steady state in about 10 seconds after the reference is set, meanwhile the lateral and heave velocities remains unchanged as expected. In figure 5.17 the pitch rate increases quickly reaching a peak value of about -0.09 rad/s in approximately 1 second. Then, it decreases towards zero with a positive overshoot of 0.04 rad/s to finally reach a value of zero as expected in about 6 seconds.

Similar results are found for the body velocities input reference v and w as can be see from figures 5.18 to 5.22. In figure 5.18, the lateral velocity reaches its steady state in about 7.5 second. These simulations suggest that the controller can follow the trajectories imposed by the input references close to the equilibrium point.

In the second set of simulations, white Gaussian noise with variances of $\sigma_p = 0.003$ and $\sigma_m = 0.02$ is added as process and measurement noise respectively. As can be seen from figures 5.26 to 5.36. Despite the presence of noise in the states, the controller is robust enough to properly follow the imposed reference and stabilized the helicopter about them demonstrating that is feasible to control the helicopter about hover despite the parameter uncertainties, and noise.

Finally, in the third set of simulations, the vector states \mathbf{x} from the first flight is set as a reference, figures 5.37 to 5.45. Body velocity u in figure 5.37 presents an overshoot of -0.9 m/s meanwhile the real response is about -0.3 m/s. This difference of 0.5 m/s may be attributed to some nonlinear dynamics that the linearized model is unable to follow. Nevertheless, the overall response of the controller properly follows the imposed reference. In the case of the lateral velocity the controller follows the trend. However, the trajectory is shifted down –see figure 5.38, and similar behavior can be seen in the trajectory response of the roll angle ϕ , figure 5.43 in which the state trajectory is shifted up from the reference. In the cases of the heave velocity w , roll rate p , pitch rate q , heading rate r , pitch angle θ and heading angle ψ , figures 5.39 to 5.42 and figures 5.44 and 5.45, the controller properly follows the reference state trajectories for each case.

Appendix A

Transformations, WGS84 Datum and Model Coefficients

A.1 Direction Cosine Matrix

A widely mathematical tool use to represent vectors in different frames is the Direction Cosine Matrix transformation or DCM. The columns of the DCM exclusively represent coordinates transformation of the vector basis expressed in one frame to another. The most common notation used for DCM is \mathbf{C}_ν^b which states a transformation of any vector expressed in frame ν to frame b , according to the following expression:

$${}^b\mathbf{x} = \mathbf{C}_\nu^b \mathbf{x} \quad (\text{A.1})$$

To get any given vector in some desired frame, matrix \mathbf{C}_ν^b on equation (A.1), might be the result of one or more successive rotations or transformations. Each DCM can be computed by doing the inner products between the unit vectors of each frame, as is described by the following equation

$$\mathbf{C}_\nu^b = \begin{bmatrix} \mathbf{i}_b \cdot \mathbf{i}_\nu & \mathbf{i}_b \cdot \mathbf{j}_\nu & \mathbf{i}_b \cdot \mathbf{k}_\nu \\ \mathbf{j}_b \cdot \mathbf{i}_\nu & \mathbf{j}_b \cdot \mathbf{j}_\nu & \mathbf{j}_b \cdot \mathbf{k}_\nu \\ \mathbf{k}_b \cdot \mathbf{i}_\nu & \mathbf{k}_b \cdot \mathbf{j}_\nu & \mathbf{k}_b \cdot \mathbf{k}_\nu \end{bmatrix} \quad (\text{A.2})$$

recalling that the inner product $\mathbf{i}_b \cdot \mathbf{i}_\nu$ is given by

$$\mathbf{i}_b \cdot \mathbf{i}_\nu = \mathbf{i}_\nu \cdot \mathbf{i}_b = \|\mathbf{i}_b\| \|\mathbf{i}_\nu\| \cos \varsigma \quad (\text{A.3})$$

where ς is the angle between the basis of \mathbf{i}_b and \mathbf{i}_ν , and $\|\cdot\|$ is the L_2 norm computed as:

$$\|\mathbf{i}_b\| = \sqrt{\mathbf{i}_b \cdot \mathbf{i}_b} = 1 \quad (\text{A.4})$$

For example, let say that the velocity vector ${}^\nu\mathbf{v}$ measured in frame \mathbf{C}_ν needs to be expressed in frame \mathbf{C}_b . Then we can proceed as follow:

$${}^b\mathbf{v} = \mathbf{C}_\nu^{b\nu} \mathbf{v} = \begin{bmatrix} \mathbf{i}_b \cdot \mathbf{i}_\nu & \mathbf{i}_b \cdot \mathbf{j}_\nu & \mathbf{i}_b \cdot \mathbf{k}_\nu \\ \mathbf{j}_b \cdot \mathbf{i}_\nu & \mathbf{j}_b \cdot \mathbf{j}_\nu & \mathbf{j}_b \cdot \mathbf{k}_\nu \\ \mathbf{k}_b \cdot \mathbf{i}_\nu & \mathbf{k}_b \cdot \mathbf{j}_\nu & \mathbf{k}_b \cdot \mathbf{k}_\nu \end{bmatrix} \begin{bmatrix} v_{x_\nu} \\ v_{y_\nu} \\ v_{z_\nu} \end{bmatrix} \quad (\text{A.5})$$

In other words, equation (A.5) says that the velocity vector ${}^b\mathbf{v}$ on frame \mathbf{C}_b can be computed by multiplying the DCM by velocity vector ${}^\nu\mathbf{v}$ expressed in the basis of frame \mathbf{C}_ν .

Once we have the DCM transformation for a particular couple of frames, it is possible to calculate the inverse transformation in the following way

$${}^\nu\mathbf{v} = \mathbf{C}_b^{\nu b} \mathbf{v} = (\mathbf{C}_\nu^b)^{-1} {}^b\mathbf{v} \quad (\text{A.6})$$

Because the matrix \mathbf{C}_ν^b is orthonormal, the transpose of the DCM is its inverse, thai is

$$(\mathbf{C}_\nu^b)^{-1} = (\mathbf{C}_\nu^b)^T = \mathbf{C}_\nu^\nu = \begin{bmatrix} \mathbf{i}_\nu \cdot \mathbf{i}_b & \mathbf{i}_\nu \cdot \mathbf{j}_b & \mathbf{i}_\nu \cdot \mathbf{k}_b \\ \mathbf{j}_\nu \cdot \mathbf{i}_b & \mathbf{j}_\nu \cdot \mathbf{j}_b & \mathbf{j}_\nu \cdot \mathbf{k}_b \\ \mathbf{k}_\nu \cdot \mathbf{i}_b & \mathbf{k}_\nu \cdot \mathbf{j}_b & \mathbf{k}_\nu \cdot \mathbf{k}_b \end{bmatrix} \quad (\text{A.7})$$

Matrices on equations (A.5) and (A.7) represent the total transformations between these two fames. To compute the DCM, for example, matrix \mathbf{C}_ν^b , three successive rotations need to be done on frame \mathbf{C}_ν , where the angles of these three rotations are known as the Euler angles, ψ , θ and ϕ . First, let do a righthand rotation of ψ degrees about \mathbf{k}_ν axis as shows figure 2.3(a). Let call this new rotated frame as $\mathbf{C}_{\nu'}$, then the transformation between \mathbf{C}_ν and $\mathbf{C}_{\nu'}$ is

$$\mathbf{C}_\nu^{\nu'} = \begin{bmatrix} \mathbf{i}_{\nu'} \cdot \mathbf{i}_\nu & \mathbf{i}_{\nu'} \cdot \mathbf{j}_\nu & \mathbf{i}_{\nu'} \cdot \mathbf{k}_\nu \\ \mathbf{j}_{\nu'} \cdot \mathbf{i}_\nu & \mathbf{j}_{\nu'} \cdot \mathbf{j}_\nu & \mathbf{j}_{\nu'} \cdot \mathbf{k}_\nu \\ \mathbf{k}_{\nu'} \cdot \mathbf{i}_\nu & \mathbf{k}_{\nu'} \cdot \mathbf{j}_\nu & \mathbf{k}_{\nu'} \cdot \mathbf{k}_\nu \end{bmatrix}$$

where from figure A.1(a) it is possible to compute the following values

$$\begin{aligned}\mathbf{i}_{\nu'} \cdot \mathbf{i}_{\nu} &= \cos \psi, \quad \mathbf{j}_{\nu'} \cdot \mathbf{i}_{\nu} = \cos \left(\frac{\pi}{2} + \psi \right) = -\sin \psi, \quad \mathbf{k}_{\nu'} \cdot \mathbf{i}_{\nu} = \cos \frac{\pi}{2} = 0, \\ \mathbf{i}_{\nu'} \cdot \mathbf{j}_{\nu} &= \cos \left(\frac{\pi}{2} - \psi \right) = \sin \psi, \quad \mathbf{j}_{\nu'} \cdot \mathbf{j}_{\nu} = \cos \psi, \quad \mathbf{k}_{\nu'} \cdot \mathbf{j}_{\nu} = \cos \frac{\pi}{2} = 0, \\ \mathbf{i}_{\nu'} \cdot \mathbf{k}_{\nu} &= \cos \frac{\pi}{2} = 0, \quad \mathbf{j}_{\nu'} \cdot \mathbf{k}_{\nu} = \cos \frac{\pi}{2} = 0, \quad \mathbf{k}_{\nu'} \cdot \mathbf{k}_{\nu} = \cos 0 = 1\end{aligned}$$

so, rewriting $\mathbf{C}_{\nu'}^{\nu'}$ yields:

$$\mathbf{C}_{\nu'}^{\nu'} = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{A.8})$$

The second righthand rotation of θ degrees about $\mathbf{j}_{\nu'}$ axis produces a new second frame $\mathbf{C}_{\nu''}$ as can be seen on figure 2.3(b). The corresponding rotation matrix is given by:

$$\mathbf{C}_{\nu''}^{\nu''} = \begin{bmatrix} \mathbf{i}_{\nu''} \cdot \mathbf{i}_{\nu'} & \mathbf{i}_{\nu''} \cdot \mathbf{j}_{\nu'} & \mathbf{i}_{\nu''} \cdot \mathbf{k}_{\nu'} \\ \mathbf{j}_{\nu''} \cdot \mathbf{i}_{\nu'} & \mathbf{j}_{\nu''} \cdot \mathbf{j}_{\nu'} & \mathbf{j}_{\nu''} \cdot \mathbf{k}_{\nu'} \\ \mathbf{k}_{\nu''} \cdot \mathbf{i}_{\nu'} & \mathbf{k}_{\nu''} \cdot \mathbf{j}_{\nu'} & \mathbf{k}_{\nu''} \cdot \mathbf{k}_{\nu'} \end{bmatrix}$$

From figure 2.3(b) we get the following expressions for the inner products

$$\begin{aligned}\mathbf{i}_{\nu''} \cdot \mathbf{i}_{\nu'} &= \cos \theta, \quad \mathbf{j}_{\nu''} \cdot \mathbf{i}_{\nu'} = \cos \frac{\pi}{2} = 0, \quad \mathbf{k}_{\nu''} \cdot \mathbf{i}_{\nu'} = \cos \left(\frac{\pi}{2} - \theta \right) = \sin \theta, \\ \mathbf{i}_{\nu''} \cdot \mathbf{j}_{\nu'} &= \cos \frac{\pi}{2} = 0, \quad \mathbf{j}_{\nu''} \cdot \mathbf{j}_{\nu'} = \cos 0 = 1, \quad \mathbf{k}_{\nu''} \cdot \mathbf{j}_{\nu'} = \cos \frac{\pi}{2} = 0, \\ \mathbf{i}_{\nu''} \cdot \mathbf{k}_{\nu'} &= \cos \left(\frac{\pi}{2} + \theta \right) = -\sin \theta, \quad \mathbf{j}_{\nu''} \cdot \mathbf{k}_{\nu'} = \cos \frac{\pi}{2} = 0, \quad \mathbf{k}_{\nu''} \cdot \mathbf{k}_{\nu'} = \cos \theta\end{aligned}$$

Then, rewriting $\mathbf{C}_{\nu''}^{\nu''}$ we get:

$$\mathbf{C}_{\nu''}^{\nu''} = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \quad (\text{A.9})$$

The third and last rotation of ϕ degrees is about axis $\mathbf{i}_{\nu''}$ yielding the final frame \mathbf{C}_b , as can be seen on figure 2.3(c). Thus the corresponding transformation is given by the following expression

$$\mathbf{C}_{\nu''}^b = \begin{bmatrix} \mathbf{i}_b \cdot \mathbf{i}_{\nu''} & \mathbf{i}_b \cdot \mathbf{j}_{\nu''} & \mathbf{i}_b \cdot \mathbf{k}_{\nu''} \\ \mathbf{j}_b \cdot \mathbf{i}_{\nu''} & \mathbf{j}_b \cdot \mathbf{j}_{\nu''} & \mathbf{j}_b \cdot \mathbf{k}_{\nu''} \\ \mathbf{k}_b \cdot \mathbf{i}_{\nu''} & \mathbf{k}_b \cdot \mathbf{j}_{\nu''} & \mathbf{k}_b \cdot \mathbf{k}_{\nu''} \end{bmatrix}$$

And the inner products values according to figure 2.3(c) are:

$$\begin{aligned}\mathbf{i}_b \cdot \mathbf{i}_{\nu'''} &= \cos 0 = 1, \quad \mathbf{j}_b \cdot \mathbf{i}_{\nu'''} = \cos \frac{\pi}{2} = 0, \quad \mathbf{k}_b \cdot \mathbf{i}_{\nu'''} = -\cos \frac{\pi}{2} = 0, \\ \mathbf{i}_b \cdot \mathbf{j}_{\nu'''} &= \cos \frac{\pi}{2} = 0, \quad \mathbf{j}_b \cdot \mathbf{j}_{\nu'''} = \cos \phi, \quad \mathbf{k}_b \cdot \mathbf{j}_{\nu'''} = \cos\left(\frac{\pi}{2} + \phi\right) = -\sin \phi, \\ \mathbf{i}_b \cdot \mathbf{k}_{\nu'''} &= \cos \frac{\pi}{2} = 0, \quad \mathbf{j}_b \cdot \mathbf{k}_{\nu'''} = \cos\left(\frac{\pi}{2} - \phi\right) = \sin \phi, \quad \mathbf{k}_b \cdot \mathbf{k}_{\nu'''} = \cos \phi\end{aligned}$$

Thus, rewriting $\mathbf{C}_{\nu'''}^b$ yields

$$\mathbf{C}_{\nu'''}^b = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \quad (\text{A.10})$$

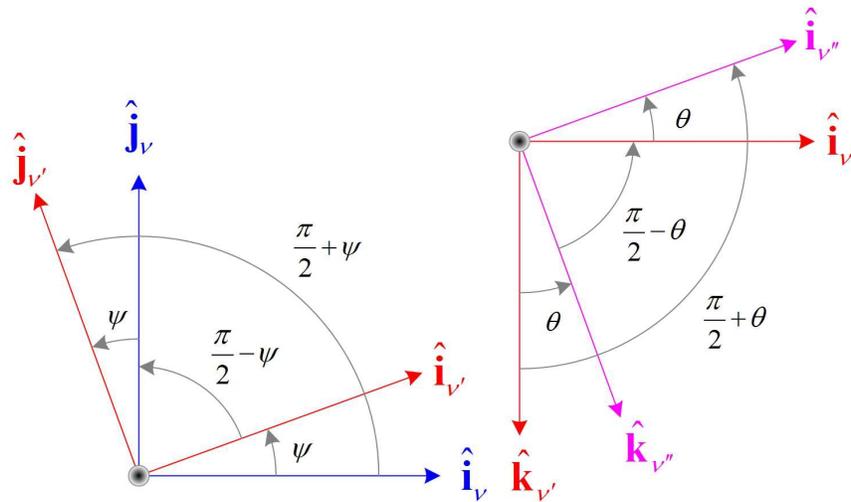
Therefore, the final DCM is the product of equations (A.8), (A.9), and (A.10) in the following order:

$$\mathbf{C}_{\nu}^b = \mathbf{C}_{\nu'''}^b \mathbf{C}_{\nu''}^{\nu'''} \mathbf{C}_{\nu}^{\nu''}$$

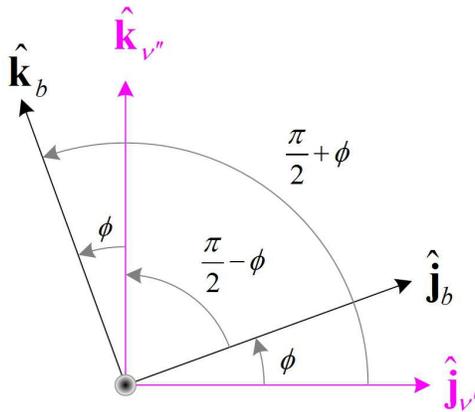
hence

$$\mathbf{C}_{\nu}^b = \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ -c\phi s\psi + s\phi s\theta c\psi & c\phi c\psi + s\phi s\theta s\psi & s\phi c\theta \\ s\phi s\psi + c\phi s\theta c\psi & -s\phi c\psi + c\phi s\theta s\psi & c\phi c\theta \end{bmatrix} \quad (\text{A.11})$$

Where s and c are the $\sin(\cdot)$ and $\cos(\cdot)$ functions respectively. Recalling that DCM is orthonormal, the inverse transformation can be computed by taking the transpose of equation (A.11).



(a) Rotation of ψ degrees about z_v axis. (b) Rotation of θ degrees
 Axes z_v and $z_{v'}$ are normal to the page and positive towards the reader. Axes $y_{v'}$
 and $y_{v''}$ are normal to the page and positive towards the reader.



(c) Rotation of ϕ degrees about $x_{v''}$ axis. Axes $x_{v''}$ and x_b are normal to the
 page and positive towards the reader.

Figure A.1: DCM inner products angle computation.

A.2 WGS84 Geodetic data

Table A.1: *WGS84 Four Defining Parameters. Reproduced from [25]*

Parameter	Description	Value
a	Semi-major axis	6378137.0 m
$\frac{1}{f}$	Reciprocal of Flattering	298.2572236
ϖ	Earth's Angular Velocity	$7292115.0 \times 10^{-11} \frac{\text{rad}}{\text{s}}$
GM	Earth's Gravitational Constant	$3986004.418 \times 10^8 \frac{\text{rad}^3}{\text{s}^2}$

Table A.2: *WGS84 Ellipsoid Geometric Constants. Reproduced form [25]*

Parameter	Description	Value
$\overline{C}_{2,0}$	Second degree Zonal Harmonic	$-0.484166774985 \times 10^{-3}$
b	Semi-minor Axis	6356752.3142 m
e	First Eccentricity	$8.1819190842622 \times 10^{-2}$
e ²	First Eccentricity Squared	$6.69437999014 \times 10^{-3}$
e'	Second Eccentricity	$8.2094437949696 \times 10^{-2}$
e' ²	Second Eccentricity Squared	$6.73949674228 \times 10^{-3}$
E	Linear Eccentricity	$5.2185400842339 \times 10^5$
c	Polar Radius of Curvature	6399593 m
$\frac{b}{a}$	Axis Ratio	0.996644719
R ₁	Mean Radius of Semi-axes	6371008.7714 m
R ₂	Radius of Sphere of Equal Area	6371007.1809 m
R ₃	Radius of Sphere of Equal Volume	6371000.7900 m

A.3 Affine-Form Coefficients

In section 2.5, the components of the total force and moments dependant of the cyclic commands are defined by equations (2.113) and (2.115), in which the corresponding coefficients are the summation of the terms associated with collective, lateral, longitudinal cyclic angls and the rotor tail pitch angle, of the equations given in section 2.2 to section 2.4. Here are presented for a quick reference.

Defining the following terms:

$$k_1 = \frac{1}{8}\pi\rho R^4\Omega^2\mathcal{S}C_{L\alpha}\kappa_3$$

$$k_2 = \frac{1}{4}\pi\rho R^4\Omega^2\mathcal{S}C_{L\alpha}$$

$$k_3 = \frac{1}{8}\frac{\rho R^5\Omega^2\mathcal{S}C_{L\alpha}}{e\mathcal{R}}$$

$$k_4 = \frac{\rho R^5\Omega^2\mathcal{S}C_{L\alpha}}{e\mathcal{R}}$$

$$k_5 = \frac{1}{4}\frac{\rho R^4\Omega^2\mathcal{S}C_{L\alpha}\kappa_3}{e\mathcal{R}}$$

$$k_6 = \frac{\rho R^4\Omega^2\mathcal{S}C_{L\alpha}}{e\mathcal{R}}$$

$$k_7 = \frac{1}{8}\pi\rho R^5\Omega^2\mathcal{S}C_{L\alpha}$$

$$k_8 = \frac{1}{6}\pi\rho R^5\Omega^2\mathcal{S}C_{L\alpha}$$

$$k_9 = \frac{1}{8}\pi\rho R_t^4\Omega_t^2\mathcal{S}_tC_{L\alpha_t}\kappa_{t3}$$

$$k_{10} = \frac{1}{4}\pi\rho R_t^4\Omega_t^2\mathcal{S}_tC_{L\alpha_t}$$

$$k_{11} = \frac{1}{4}\frac{\rho R_t^5\Omega_t^2\mathcal{S}_tC_{L\alpha_t}}{e_t\mathcal{R}_t}$$

$$k_{12} = \frac{\rho R_t^5\Omega_t^2\mathcal{S}_tC_{L\alpha_t}}{e_t\mathcal{R}_t}$$

$$k_{13} = \frac{1}{4}\frac{\rho R_t^4\Omega_t^2\mathcal{S}_tC_{L\alpha_t}\kappa_{t3}}{e_t\mathcal{R}_t}$$

$$k_{14} = \frac{\rho R_t^4 \Omega_t^2 \mathcal{S}_t C_{L\alpha t}}{e_t \mathcal{R}_t}$$

Coefficients of equation (2.113).

$$\mathbf{a}_{1,1} = \frac{k_6}{4} C_{L\alpha} \kappa_3 \mu_r u_{hub}$$

$$\mathbf{a}_{1,2} = \frac{k_6}{16} C_{L\alpha} \kappa_3 \mu_r u_{hub}$$

$$\mathbf{a}_{1,3} = \frac{3k_6}{16} C_{L\alpha} \kappa_3 \mu_r u_{hub}$$

$$\mathbf{a}_{1,4} = \frac{k_6}{4} C_{L\alpha} u_{hub} v_{hub}$$

$$\mathbf{a}_{1,5} = -\frac{k_6}{6} C_{L\alpha} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (3u_{hub}^2 + v_{hub}^2) \right]$$

$$\mathbf{a}_{1,6} = -\frac{k_6}{8} C_{L\alpha} \kappa_3 \mu_r v_{hub}$$

$$\mathbf{a}_{1,7} = \frac{k_6}{4} (\pi e \mathcal{R} - 2C_{L\alpha}) \left[-\frac{1}{3} \kappa_2 \mu_q \mu_r + \kappa_4 u_{hub} (\mu_z - \lambda_i) \right]$$

$$\mathbf{a}_{1,8} = -\frac{k_6}{32} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) [\mu_q u_{hub} + \mu_p v_{hub}]$$

$$\mathbf{a}_{1,9} = -\frac{k_6}{8} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left[\mu_r (\mu_z - \lambda_i) - \frac{3}{4} \mu_p \mu_x - \frac{1}{4} \mu_q v_{hub} + \frac{3}{4} \mu_q \mu_p Z'_{rcg} \right]$$

$$\mathbf{a}_{1,10} = \frac{k_{14}}{4} \kappa_{t3} C_{L\alpha t} \mu_{qt} \left[\mathbf{u}_t + \mu_{rt} Y'_{tcg} \right]$$

$$\mathbf{a}_{1,11} = \frac{k_{14}}{4} (\pi e_t \mathcal{R}_t - 2C_{L\alpha t}) \left[-\frac{1}{3} \kappa_{t2} \mu_{pt} \mu_{qt} + \kappa_{t4} (u_{hub}^2 + 3v_{hub}^2) \mathbf{v}_t \right]$$

$$\mathbf{a}_{2,1} = -\frac{k_6}{4} C_{L\alpha} \kappa_3 \mu_r v_{hub}$$

$$\mathbf{a}_{2,2} = -\frac{3k_6}{16} C_{L\alpha} \kappa_3 \mu_r v_{hub}$$

$$\mathbf{a}_{2,3} = -\frac{k_6}{16} C_{L_\alpha} \kappa_3 \mu_r v_{hub}$$

$$\mathbf{a}_{2,4} = -\frac{k_6}{6} C_{L_\alpha} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (u_{hub}^2 + 3v_{hub}^2) \right]$$

$$\mathbf{a}_{2,5} = \frac{k_6}{4} C_{L_\alpha} \kappa_4 u_{hub} v_{hub}$$

$$\mathbf{a}_{2,6} = \frac{k_6}{8} C_{L_\alpha} \kappa_3 \mu_r u_{hub}$$

$$\mathbf{a}_{2,7} = \frac{k_6}{4} (\pi e \mathcal{R} - 2C_{L_\alpha}) \left[\frac{1}{3} \kappa_2 \mu_q \mu_r - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right]$$

$$\mathbf{a}_{2,8} = \frac{k_6}{8} \kappa_3 (\pi e \mathcal{R} - 2C_{L_\alpha}) \left[-\mu_r (\mu_z - \lambda_i) + \frac{3}{4} \mu_p \mu_y + \frac{1}{4} \mu_p u_{hub} + \frac{3}{4} \mu_q \mu_p Z'_{rcg} \right]$$

$$\mathbf{a}_{2,9} = -\frac{k_6}{32} \kappa_3 (\pi e \mathcal{R} - 2C_{L_\alpha}) [\mu_q u_{hub} + \mu_p v_{hub}]$$

$$\mathbf{a}_{2,10} = 0$$

$$\mathbf{a}_{2,11} = k_{10} \left\{ \frac{2}{3} \kappa_{t_2} \mu_{qt}^2 + \kappa_{t_4} \left[\left(\mathbf{u}_t + \mu_{r_t} Y'_{tcg} \right)^2 + \left(\mathbf{w}_t - \mu_{p_t} Y'_{tcg} \right)^2 \right] \right\}$$

$$\mathbf{a}_{3,1} = 0$$

$$\mathbf{a}_{3,2} = 0$$

$$\mathbf{a}_{3,3} = 0$$

$$\mathbf{a}_{3,4} = 0$$

$$\mathbf{a}_{3,5} = 0$$

$$\mathbf{a}_{3,6} = 0$$

$$\mathbf{a}_{3,7} = k_2 \left[\frac{2}{3} \kappa_2 \mu_r^2 + \kappa_4 (u_{hub}^2 + v_{hub}^2) \right]$$

$$\mathbf{a}_{3,8} = k_2 \kappa_3 \mu_r v_{hub}$$

$$\mathbf{a}_{3,9} = -k_2 \kappa_3 \mu_r u_{hub}$$

$$\mathbf{a}_{3,10} = -\frac{k_{14}}{4} C_{L\alpha_t} \kappa_{t_3} \mu_{qt} \left(\mathbf{w}_t - \mu_{pt} Y'_{tcg} \right)$$

$$\mathbf{a}_{3,11} = \frac{k_{14}}{4} \left(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t} \right) \left[\frac{1}{3} \kappa_{t_2} \mu_{rt} \mu_{qt} - \kappa_{t_4} \left(\mathbf{w}_t - \mu_{pt} Y'_{tcg} \right) \mathbf{v}_t \right]$$

Coefficients of equation (2.115).

$$\mathbf{b}_{1,1} = -\frac{k_6}{4} C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg}$$

$$\mathbf{b}_{1,2} = -\frac{3k_6}{16} C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg}$$

$$\mathbf{b}_{1,3} = -\frac{k_6}{3} C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg}$$

$$\mathbf{b}_{1,4} = -\frac{k_6}{6} C_{L\alpha} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (u_{hub}^2 + 3v_{hub}^2) \right] z_{rcg}$$

$$\mathbf{b}_{1,5} = \frac{k_6}{4} C_{L\alpha} \kappa_4 v_{hub} u_{hub} z_{rcg}$$

$$\mathbf{b}_{1,6} = \frac{k_6}{8} C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg}$$

$$\mathbf{b}_{1,7} = -k_8 \kappa_2 \mu_r u_{hub} + \frac{k_6}{4} (\pi e \mathcal{R} - 2C_{L\alpha}) \left[\frac{1}{3} \kappa_2 \mu_q \mu_r - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right] z_{rcg}$$

$$\mathbf{b}_{1,8} = \frac{3k_8}{8} \kappa_3 v_{hub} u_{hub} + \frac{k_6}{8} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left[-\mu_r (\mu_z - \lambda_i) \right. \\ \left. + \frac{3}{4} \mu_q \mu_y + \frac{1}{4} \mu_p u_{hub} + \frac{3}{4} \mu_q \mu_p Z'_{rcg} \right] z_{rcg}$$

$$\mathbf{b}_{1,9} = -\frac{3k_8}{8} \left[\kappa_1 \mu_r^2 + \frac{1}{2} \kappa_3 (3u_{hub}^2 + v_{hub}^2) \right] - \frac{k_6}{32} \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) [\mu_q u_{hub} + \mu_p v_{hub}] z_{rcg}$$

$$\mathbf{b}_{1,10} = \frac{k_{14}}{4} C_{L\alpha_t} \kappa_{t_3} \left(\mathbf{w}_t + \mu_{pt} Y'_{tcg} \right) \mu_{qt} y_{tcg}$$

$$\mathbf{b}_{1,11} = k_{10} \left\{ \frac{2}{3} \kappa_{t_2} \mu_{qt}^2 + \kappa_{t_4} \left[\left(\mathbf{u}_t + \mu_{rt} Y'_{tcg} \right)^2 + \left(\mathbf{w}_t - \mu_{pt} Y'_{tcg} \right)^2 \right] \right\} z_{tcg} - \\ \frac{k_{14}}{4} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \left[\frac{\kappa_{t_2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t_4} \left(\mathbf{w}_t - \mu_{pt} Y'_{tcg} \right) \mathbf{v}_t \right] y_{tcg}$$

$$\begin{aligned}
\mathbf{b}_{2,1} &= -\frac{k_6}{4} C_{L_\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \\
\mathbf{b}_{2,2} &= -\frac{k_6}{16} C_{L_\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \\
\mathbf{b}_{2,3} &= -\frac{3k_6}{16} C_{L_\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \\
\mathbf{b}_{2,4} &= -\frac{k_6}{4} C_{L_\alpha} \kappa_4 v_{hub} u_{hub} z_{rcg} \\
\mathbf{b}_{2,5} &= \frac{k_6}{6} C_{L_\alpha} \left[\kappa_2 \mu_r^2 + \frac{3}{4} \kappa_4 (3u_{hub}^2 + v_{hub}^2) \right] z_{rcg} \\
\mathbf{b}_{2,6} &= \frac{k_6}{8} C_{L_\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \\
\mathbf{b}_{2,7} &= -k_8 \kappa_2 \mu_r u_{hub} - \frac{k_6}{4} (\pi e_t \mathcal{R} - 2C_{L_\alpha}) \left[-\frac{1}{3} \kappa_2 \mu_q \mu_r + \kappa_4 u_{hub} (\mu_z - \lambda_i) \right] z_{rcg} \\
\mathbf{b}_{2,8} &= \frac{3k_8}{8} \left[\kappa_1 \mu_r^2 + \frac{1}{2} \kappa_3 (u_{hub}^2 + 3v_{hub}^2) \right] + \frac{k_6}{32} \kappa_3 (\pi e_t \mathcal{R} - 2C_{L_\alpha}) [\mu_q u_{hub} + \mu_p v_{hub}] z_{rcg} \\
\mathbf{b}_{2,9} &= -\frac{3k_8}{8} \kappa_3 u_{hub} v_{hub} + \frac{1}{8} \kappa_3 (\pi e_t \mathcal{R} - 2C_{L_\alpha}) \left[\mu_r (\mu_z - \lambda_i) - \frac{3}{4} \mu_p \mu_x \right. \\
&\quad \left. - \frac{1}{4} \mu_q v_{hub} + \frac{3}{4} \mu_q \mu_p z'_{rcg} \right] z_{rcg} \\
\mathbf{b}_{2,10} &= -\frac{k_{14}}{4} C_{L_{\alpha_t}} \kappa_{t3} \mu_{qt} \left(\mathbf{w}_t + \mu_{pt} Y'_{tcg} \right) x_{tcg} - \frac{k_{14}}{4} C_{L_{\alpha_t}} \kappa_{t3} \mu_{qt} \left(\mathbf{u}_t + \mu_{rt} Y'_{tcg} \right) z_{tcg} - \\
&\quad \frac{k_{12}}{8} C_{L_{\alpha_t}} \left\{ \kappa_{t1} \mu_{qt}^2 + \kappa_{t3} \left[\left(\mathbf{u}_t + \mu_{rt} Y'_{tcg} \right)^2 + \left(\mu_{pt} Y'_{tcg} - \mathbf{w}_t \right)^2 \right] \right\} \\
\mathbf{b}_{2,11} &= \frac{k_{14}}{4} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left[\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} \left(\mathbf{w}_t - \mu_{pt} Y'_{tcg} \right) \mathbf{v}_t \right] x_{tcg} - \\
&\quad \frac{k_{14}}{4} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left[\kappa_{t4} \left(\mathbf{u}_t - \mu_{rt} Y'_{tcg} \right) \mathbf{v}_t - \frac{\kappa_{t2} \mu_{pt} \mu_{qt}}{3} \right] z_{tcg} - \\
&\quad \frac{k_{12}}{6} \kappa_{t2} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left(\mu_{qt} \mathbf{v}_t - \frac{\mu_{rt} \mathbf{w}_t + \mu_{pt} \mathbf{u}_t}{2} \right) \\
\mathbf{b}_{3,1} &= \frac{k_4}{8} C_{L_\alpha} \left[\kappa_3 (u_{hub}^2 + v_{hub}^2) + \kappa_1 \mu_r^2 \right] \\
\mathbf{b}_{3,2} &= \frac{k_4}{16} C_{L_\alpha} \left[\frac{\kappa_3}{2} (u_{hub}^2 + 3v_{hub}^2) + \kappa_1 \mu_r^2 \right]
\end{aligned}$$

$$\begin{aligned}
\mathfrak{b}_{3,3} &= \frac{k_4}{16} C_{L_\alpha} \left[\frac{\kappa_3}{2} (3u_{hub}^2 + v_{hub}^2) + \kappa_1 \mu_r^2 \right] \\
\mathfrak{b}_{3,4} &= \frac{k_4}{3} C_{L_\alpha} \kappa_2 \mu_r v_{hub} \\
\mathfrak{b}_{3,5} &= -\frac{k_4}{3} C_{L_\alpha} \kappa_2 \mu_r u_{hub} \\
\mathfrak{b}_{3,6} &= -\frac{k_4}{8} C_{L_\alpha} \kappa_3 u_{hub} v_{hub} \\
\mathfrak{b}_{3,7} &= -\frac{k_4}{6} \kappa_2 (\pi e \mathcal{R} - 2C_{L_\alpha}) \left[(\lambda_i - \mu_z) \mu_r + \frac{\mu_p \mu_x}{2} + \frac{\mu_q \mu_y}{2} \right] \\
\mathfrak{b}_{3,8} &= -\frac{k_4}{8} (\pi e \mathcal{R} - 2C_{L_\alpha}) [\kappa_3 v_{hub} (\mu_z - \lambda_i) - \kappa_1 \mu_q \mu_r] \\
\mathfrak{b}_{3,9} &= -\frac{k_4}{8} (\pi e \mathcal{R} - 2C_{L_\alpha}) [\kappa_3 u_{hub} (\mu_z - \lambda_i) - \kappa_1 \mu_p \mu_r] \\
\mathfrak{b}_{3,10} &= \frac{k_{14}}{4} C_{L_{\alpha_t}} \kappa_{t3} \left(\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}} \right) \mu_{q_t} y_{t_{cg}} \\
\mathfrak{b}_{3,11} &= -k_{10} \left\{ \frac{2}{3} \kappa_{t2} \mu_{q_t}^2 + \kappa_{t4} \left[\left(\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}} \right)^2 + \left(\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}} \right)^2 \right] \right\} x_{t_{cg}} + \\
&\quad \frac{k_{14}}{4} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left[\kappa_{t4} \left(\mathbf{u}_t - \mu_{r_t} Y'_{t_{cg}} \right) \mathbf{v}_t - \frac{\kappa_{t2} \mu_{p_t} \mu_{q_t}}{3} \right] y_{t_{cg}}
\end{aligned}$$

Appendix B

MOI Test Bed and HIL Force Bed

B.1 Test Bed Model

As was described on the chapter 3, the motion equation for pulley 1 and 2 were:

$$(T_2 - T_1)r_2 = I_{p_2}\alpha_2 + b_2\omega_2 + \tau_{s_2} \quad (\text{B.1})$$

and

$$T_1r_1 = (I_{p_1} + I_{T_B} + I_H)\alpha_1 + b_1\omega_1 + \tau_{s_1} \quad (\text{B.2})$$

Let be $I_{pB} = I_{p_1} + I_{T_B}$ and isolating T_1 on equation (B.2) yields

$$T_1 = \frac{I_{pB} + I_H}{r_1}\alpha_1 + \frac{b_1}{r_1}\omega_1 + \frac{\tau_{s_1}}{r_1} \quad (\text{B.3})$$

Substituting (B.3) into (B.1) and solving for T_2 yields

$$T_2 = \frac{I_{pB} + I_H}{r_1}\alpha_1 + \frac{I_{p_2}}{r_2}\alpha_2 + \frac{b_1}{r_1}\omega_1 + \frac{b_2}{r_2}\omega_2 + \frac{\tau_{s_1}}{r_1} + \frac{\tau_{s_2}}{r_2} \quad (\text{B.4})$$

Because the linear velocity of the string is the same in every single point of it, the tangential velocity at pulley 1 is equal to the tangential velocity at pulley 2, therefore; $\omega_1r_1 = \omega_2r_2$. Solving for ω_2 and substituting into (B.4) yields

$$T_2 = \frac{I_{pB} + I_H}{r_1}\alpha_1 + \frac{I_{p_2}}{r_2}\alpha_2 + \left(\frac{b_1}{r_1} + \frac{b_2r_1}{r_2^2}\right)\omega_1 + \frac{\tau_{s_1}}{r_1} + \frac{\tau_{s_2}}{r_2} \quad (\text{B.5})$$

Substituting this expression into (3.1) and expressing the angular acceleration in terms of angular velocities we get

$$\frac{I_{pB} + I_H}{r_1} \dot{\omega}_1 + \frac{I_{p2}}{r_2} \dot{\omega}_2 + \left(\frac{b_1}{r_1} + \frac{b_2 r_1}{r_2^2} \right) \omega_1 + \frac{\tau_{s1}}{r_1} + \frac{\tau_{s2}}{r_2} - mg = -ma \quad (\text{B.6})$$

On the other hand, the linear acceleration a is equal to $\dot{\omega}_2 r_2$ yielding

$$\frac{I_{pB} + I_H}{r_1} \dot{\omega}_1 + \frac{I_{p2}}{r_2} \dot{\omega}_2 + \left(\frac{b_1}{r_1} + \frac{b_2 r_1}{r_2^2} \right) \omega_1 + \frac{\tau_{s1}}{r_1} + \frac{\tau_{s2}}{r_2} - mg = -m\dot{\omega}_2 r_2 \quad (\text{B.7})$$

Finally, expressing ω_2 in terms of ω_1 and regrouping terms yield

$$\left(\frac{I_{pB} + I_H}{r_1} + \frac{I_{p2} r_1}{r_2^2} + mr_1 \right) \dot{\omega}_1 + \left(\frac{b_1}{r_1} + \frac{b_2 r_1}{r_2^2} \right) \omega_1 + \frac{\tau_{s1}}{r_1} + \frac{\tau_{s2}}{r_2} - mg = 0 \quad (\text{B.8})$$

Or in terms of the angular position

$$\left(\frac{I_{pB} + I_H}{r_1} + \frac{I_{p2} r_1}{r_2^2} + mr_1 \right) \ddot{\theta}_1 + \left(\frac{b_1}{r_1} + \frac{b_2 r_1}{r_2^2} \right) \dot{\theta}_1 + \frac{\tau_{s1}}{r_1} + \frac{\tau_{s2}}{r_2} - mg = 0 \quad (\text{B.9})$$

In order to solve equation (B.8) and (B.9), let be $k_1 = \frac{I_{pB} + I_H}{r_1} + \frac{I_{p2} r_1}{r_2^2}$, $k_2 = \frac{b_1}{r_1} + \frac{b_2 r_1}{r_2^2}$, and $k_3 = \frac{\tau_{s1}}{r_1} + \frac{\tau_{s2}}{r_2}$ therefore, equation (B.8) and (B.9) become

$$(k_1 + mr_1) \dot{\omega}_1 + k_2 \omega_1 + k_3 - mg = 0 \quad (\text{B.10})$$

and

$$(k_1 + mr_1) \ddot{\theta}_1 + k_2 \dot{\theta}_1 + k_3 - mg = 0 \quad (\text{B.11})$$

Solving equation (B.10) yield

$$\omega(t) = C_1 e^{\frac{-k_2}{k_1 + mr_1} t} + \frac{mg - k_3}{k_2} \quad (\text{B.12})$$

However, the angular velocity at time $t = 0$ is zero, thus $\omega(0) = 0 = C_1 + \frac{mg - k_3}{k_2}$ or equivalent $C_1 = -\frac{mg - k_3}{k_2}$. Let be $\tau = \frac{k_1 + mr_1}{k_2}$ therefore we can rewrite equation (B.12) as:

$$\omega(t) = \frac{mg - k_3}{k_2} \left(1 - e^{\frac{-t}{\tau}} \right) \quad (\text{B.13})$$

Equation (B.13) represents the angular velocity of the test bed as a func-

tion of time, the hanging mass m , and the object's moment of inertia to be estimated.

On the other hand, solving equation (B.11) yields

$$\theta(t) = -\frac{C_1(k_1 + mr_1)}{k_2} e^{-\frac{k_2}{k_1 + mr_1}t} + C_2 + \frac{mg - k_3}{k_2}t \quad (\text{B.14})$$

But we can set the position at time $t = 0$ equal to zero, hence $\theta(0) = 0 = -\frac{C_1(k_1 + mr_1)}{k_2} + C_2$ or equivalent, $C_2 = \frac{C_1(k_1 + mr_1)}{k_2}$. Once again, let be $\tau = \frac{k_1 + mr_1}{k_2}$ then equation (B.14) transforms into

$$\theta(t) = \frac{C_1(k_1 + mr_1)}{k_2} \left(1 - e^{-\frac{t}{\tau}}\right) + \frac{mg - k_3}{k_2}t \quad (\text{B.15})$$

But $C_1 = C = \frac{mg - k_3}{k_2}$ hence

$$\theta(t) = -\frac{mg - k_3}{k_2} \frac{k_1 + mr_1}{k_2} \left(1 - e^{-\frac{t}{\tau}}\right) + \frac{mg - k_3}{k_2}t$$

Or equivalent

$$\theta(t) = \frac{(mg - k_3)(k_1 + mr_1)}{k_2^2} \left(e^{-\frac{t}{\tau}} - 1\right) + \frac{mg - k_3}{k_2}t \quad (\text{B.16})$$

Equation (B.16) represents the angular position as a function of time, the hanging mass, and the object's moment of inertia to be estimated. The angular position is read directly from the position encoder attached to the MOI bed.

B.2 Pulley's and Rectangular shapes MOI computation and uncertainty analysis

B.2.1 Moment of inertia

The pulleys' moments of inertia were computed assuming them as thin circular even mass distributed shape of radius r and mass m , using the following analytical expression:

$$I = \frac{1}{2}mr^2 \quad (\text{B.17})$$

Substituting the corresponding values given in table B.2 on equation (B.17) yields:

$$I_{p1} = 6.21 \times 10^{-5} \text{ kg} \cdot \text{m}^2$$

$$I_{p_2} = 3.69 \times 10^{-5} \text{ kg} \cdot \text{m}^2$$

In similar way, the mass moment of inertia of the rectangular aluminium bed was calculated using the expression:

$$I = \frac{1}{12} m (w^2 + \ell^2) \quad (\text{B.18})$$

where w is the wide and ℓ is the length. Substituting the corresponding values given in table B.2 yields:

$$I_B = 0.0594 \text{ kg} \cdot \text{m}^2$$

B.2.2 Error propagation

The measurement tools and its characteristics used to determine the dimensions and mass of the MOI test bed components are listed in table B.1

Table B.1: *Measurement Instruments Characteristics.*

Instrument	Brand	Rnge	Span	Resolution	Accuracy
Measure Tape	Stanley	0 to 8 m	8 m	1 mm	± 0.794 mm
Digital caliper	unknown	0 to 153.52 mm	153.52 mm	0.01 mm	± 0.02 mm
Digital Scale	Mettler PE16	0 to 16 kg	16 kg	1 gr	± 0.2 gr

Based on statistical analysis, was determined that ten measurements with each instrument were enough to reduced the total uncertainty in the range of the instruments accuracy within 95% of confidence interval. Therefore, the error propagation was computed with the following expression:

$$\varepsilon_x(\alpha_i) = \sqrt{\left[\sum_{i=1}^n \frac{\partial x}{\partial \alpha_i} \varepsilon_{\alpha_i} \right]^2}$$

where $\varepsilon_x(\alpha_i)$ is the uncertainty in the computed x measurement, α_i are measurement parameter, and ε_{α_i} are the specific uncertainties in the measurement parameters.

Computing the uncertainties of the mass moments of inertia given on equations (B.17) and (B.18) for pulleys and rectangular shapes yields

$$\begin{aligned} \varepsilon_{I_{P_{1,2}}} &= \frac{r}{2} \sqrt{r^2 \varepsilon_m^2 + 4m^2 \varepsilon_r^2} \\ \varepsilon_{I_B} &= \frac{1}{12} \sqrt{(w^2 + \ell^2) \varepsilon_m^2 + 4m^2 (w^2 \varepsilon_w^2 + \ell^2 \varepsilon_\ell^2)} \end{aligned} \quad (\text{B.19})$$

Using the accuracy given in table B.1 as a bias uncertainty and substituting the corresponding values given in table B.2, into equation (B.19) the uncertainties on the mass moments of inertia for pulleys and rectangular bed ares:

$$\begin{aligned}\varepsilon_{I_{p_1}} &= \pm 5.93 \times 10^{-8} \text{ N} \cdot \text{m}^2 \\ \varepsilon_{I_{p_2}} &= \pm 3.94 \times 10^{-8} \text{ N} \cdot \text{m}^2 \\ \varepsilon_{I_B} &= \pm 1.42 \times 10^{-4} \text{ N} \cdot \text{m}^2\end{aligned}$$

Table B.2 summarize the computed mass moments of inertia and MOI bed characteristics. In addition, it shows the used equation to calculate the local gravity.

B.3 Servo and Load Cell Calibration Outcomes

According to the graphics of figure B.1 the relation between the PWM signal and the angular displacement in the servo motors' operational region is given by the expression

$$\theta_{s_i} = m_i t_w + b_i \quad (\text{B.20})$$

where m_i is the corresponding slope for each servo, b_i is their corresponding offset and t_w is the pulse width in ms. The operational range of the swash-plate servo, Futaba S9551 is in the range of $-73.26^\circ < \theta_{SP} < 68^\circ$ and for the tail servo, Futaba S9256 is in the range of $-55.55^\circ < \theta_T < 59.53^\circ$.

Table B.3 summarize the calibration outcomes.

Load cells calibration response can be seen on the plots of figures B.2 and B.3. Here is shown that the relation between the applied force and the voltage response is linear within the tested range, and can be represented by the following equation:

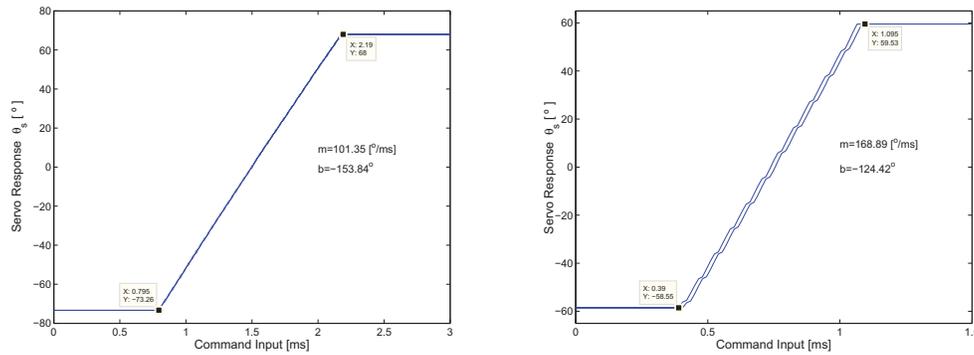
$$v_{\ell_i} = m_{\ell_i} \mathbf{F} + b_{\ell_i} \quad (\text{B.21})$$

where m_{ℓ_i} are the slopes of load cells in $\frac{\text{mV}}{\text{N}}$, b_{ℓ_i} are their corresponding offset and \mathbf{F} is the applied force in N.

Table B.4 summarize the outcomes of the calibration process for all the cells.

Table B.2: Physical Characteristics of the MOI Bed Elements.

Parameter	Units	Pulley 1	Pulley 2	Bed
Wide	m	—	—	0.3417 ± 0.8 mm
Length	m	—	—	0.3412 ± 0.8 mm
High/Thick	m	0.0309 ± 0.02 mm	0.2039 ± 0.02 mm	0.0100 ± 0.02 mm
Radius	m	0.0275 ± 0.02 mm	0.0190 ± 0.02 mm	—
Mass	kg	0.3596 ± 0.2 gr	0.2039 ± 0.2 gr	3.0549 ± 0.2 gr
MOI	$\text{kg} \cdot \text{m}^2$	$6.21 \times 10^{-5} \pm 5.93 \times 10^{-8}$	$3.69 \times 10^{-5} \pm 3.94 \times 10^{-8}$	$0.0594 \pm 1.42 \times 10^{-4}$
Constants				
Local Gravity g_L	9.8119	$g_L = 9.780327 [1 + 0.0053024(\sin \zeta)^2 - 5.8 \times 10^{-6}(\sin \zeta)^2] - 3.086 \times 10^{-6}h$		
Latitude ζ	53.6692	[°]		
Altitude h	668	[m]		

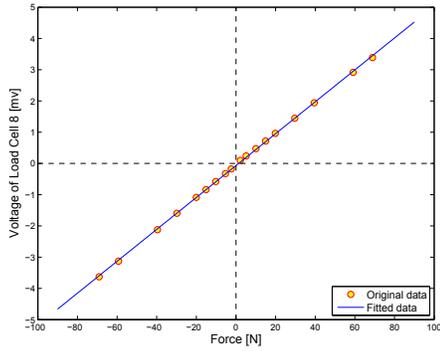


(a) *Futaba S9551 swash-plate calibration* (b) *Futaba S9256 tail calibration servo response.*

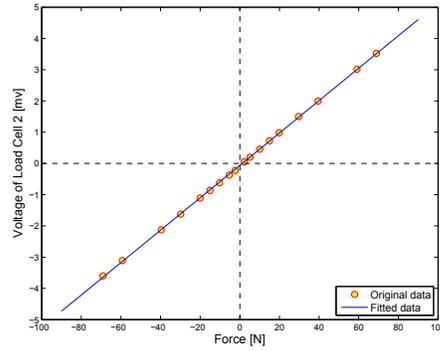
Figure B.1: Calibration servo response for swash-plate mechanism and tail rotor. Observe the fastest response on the tail servo, figure b) compared to the swash-plate servos, figure a)

Table B.3: Servo Calibration Outcome Summary.

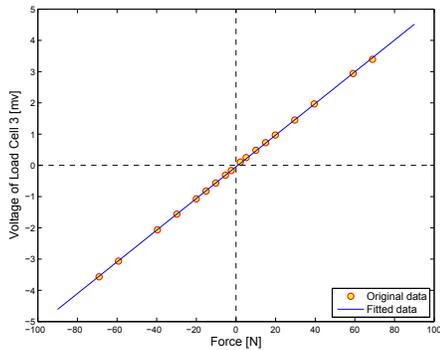
Servo	m [$\frac{^\circ}{\text{ms}}$]	b [°]
S_{w1}	101.32	-160
S_{w2}	101.32	-159
S_{w3}	101.32	-158
S_{w4}	101.32	-152
S_t	168.89	-124.42



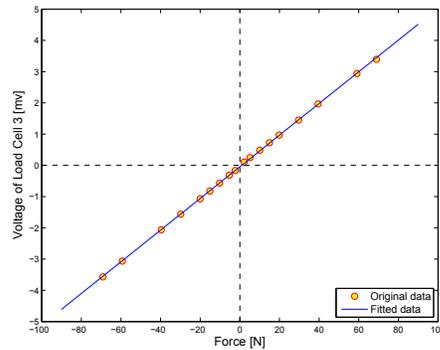
(a) Cell No. 1.



(b) Cell No. 2.



(c) Cell No. 3.

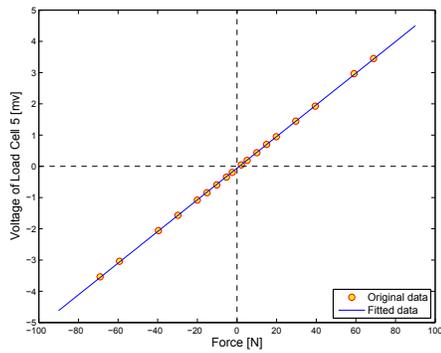


(d) Cell No. 4.

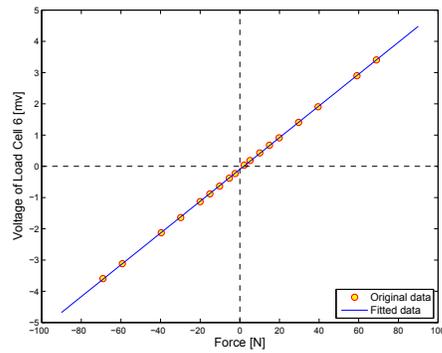
Figure B.2: Longitudinal load cell responses.

Table B.4: Load Cells Calibration Outcome Summary.

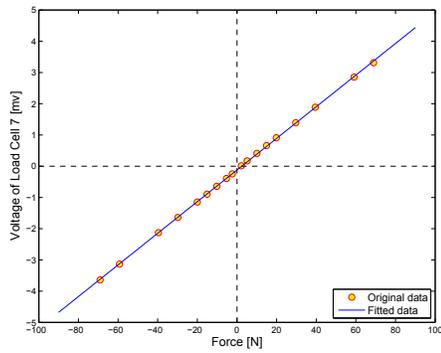
Cell	$m \left[\frac{mV}{N} \right]$	$b [mV]$
1	0.0511 ± 0.0004	-0.0685 ± 0.0153
2	0.0519 ± 0.0003	-0.0627 ± 0.0113
3	0.0507 ± 0.0003	-0.0472 ± 0.0109
4	0.0502 ± 0.0003	-0.1132 ± 0.0119
5	0.0507 ± 0.0002	-0.0606 ± 0.0081
6	0.0509 ± 0.0002	-0.1009 ± 0.0075
7	0.0506 ± 0.0003	-0.1213 ± 0.0109
8	0.0498 ± 0.0003	-0.0446 ± 0.0091



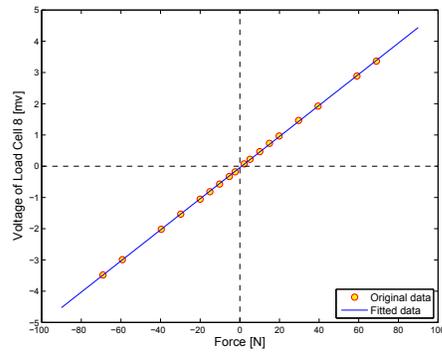
(a) Cell No. 5. Lateral



(b) Cell No. 6. Lateral



(c) Cell No. 7. Horizontal



(d) Cell No. 8. Horizontal

Figure B.3: Lateral and horizontal load cell responses.

B.4 Homogenous Transformations and Solution

In this section, the homogeneous translation and rotation matrices used in section 3.2.3 on chapter 3 are defined. In addition, a brief explanation in the solution of equation (3.13) to (3.16) is given.

$$\mathbf{T}_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -z_c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\delta_y} = \begin{bmatrix} \cos \delta_y & 0 & \sin \delta_y & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \delta_y & 0 & \cos \delta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{C}_{\delta_x} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \delta_x & -\sin \delta_x & 0 \\ 0 & \sin \delta_x & \cos \delta_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_{d_2} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{d_1} = \begin{bmatrix} 1 & 0 & 0 & d_1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{\ell_{11}} = \begin{bmatrix} 1 & 0 & 0 & l_{11} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_{d_3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{d_4} = \begin{bmatrix} 1 & 0 & 0 & -d_4 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{\ell_{21}} = \begin{bmatrix} 1 & 0 & 0 & l_{21} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_{d_5} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -d_5 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{T}_{\ell_{31}} = \begin{bmatrix} 1 & 0 & 0 & -l_{31} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{T}_{d_6} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -d_6 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{T}_{l_{41}} = \begin{bmatrix} 1 & 0 & 0 & -l_{41} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\theta_1} = \begin{bmatrix} \cos \theta_1 & 0 & \sin \theta_1 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_1 & 0 & \cos \theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{C}_{\theta_{12}} = \begin{bmatrix} \cos \theta_{12} & 0 & \sin \theta_{12} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_{12} & 0 & \cos \theta_{12} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\theta_{13}} = \begin{bmatrix} \cos \theta_{13} & -\sin \theta_{13} & 0 & 0 \\ \sin \theta_{13} & \cos \theta_{13} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{C}_{\theta_2} = \begin{bmatrix} \cos \theta_2 & 0 & \sin \theta_2 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_2 & 0 & \cos \theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\theta_{22}} = \begin{bmatrix} \cos \theta_{22} & 0 & \sin \theta_{22} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_{22} & 0 & \cos \theta_{22} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{C}_{\theta_{23}} = \begin{bmatrix} \cos \theta_{23} & -\sin \theta_{23} & 0 & 0 \\ \sin \theta_{23} & \cos \theta_{23} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\theta_3} = \begin{bmatrix} \cos \theta_3 & 0 & \sin \theta_3 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_3 & 0 & \cos \theta_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{C}_{\theta_{32}} = \begin{bmatrix} \cos \theta_{32} & 0 & \sin \theta_{32} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_{32} & 0 & \cos \theta_{32} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\theta_{33}} = \begin{bmatrix} \cos \theta_{33} & -\sin \theta_{33} & 0 & 0 \\ \sin \theta_{33} & \cos \theta_{33} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{C}_{\theta_4} = \begin{bmatrix} \cos \theta_4 & 0 & \sin \theta_4 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_4 & 0 & \cos \theta_4 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{C}_{\theta_{42}} = \begin{bmatrix} \cos \theta_{42} & 0 & \sin \theta_{42} & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta_{42} & 0 & \cos \theta_{42} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{C}_{\theta_{43}} = \begin{bmatrix} \cos \theta_{43} & -\sin \theta_{43} & 0 & 0 \\ \sin \theta_{43} & \cos \theta_{43} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The solution for equations (3.13) to (3.16) given by [24] is explained in detail for the first set (3.13). The rest were solved following exactly the same procedure.

Rewriting equations (3.13a) and (3.13c) as

$$r \cos \delta_y - d_1 - \ell_{11} \cos \theta_1 = \cos(\theta_1 + \theta_{12}) - \ell_{12} \cos \theta_{13}$$

and

$$-r \sin \delta_x - z_c + \ell_{11} \sin \theta_1 = \sin(\theta_1 + \theta_{12}) - \ell_{12} \cos \theta_{13}$$

Let be $C_{a1} = r \cos \delta_y - d_1$, $C_{b1} = -r \sin \delta_x - z_c$, and $C_{c1} = \ell_{12} \cos \theta_{13}$, squaring both sides of both equations and adding them up yields:

$$(C_{a1} - \ell_{11} \cos \theta_1)^2 + (C_{b1} + \ell_{11} \sin \theta_1)^2 = C_{c1}^2 [\cos(\theta_1 + \theta_{12})^2 + \sin(\theta_1 + \theta_{12})^2]$$

Expanding and doing some simplifications we write

$$C_{a1} \cos \theta_1 - C_{b1} \sin \theta_1 = \frac{C_{a1}^2 + C_{b1}^2 + \ell_{11}^2 - C_{c1}^2}{2\ell_{11}}$$

Using the following trigonometry identity

$$a \cos x \pm b \sin x = \sqrt{a^2 + b^2} \cos(x \mp \vartheta)$$

with $a = C_{a1}$ and $b = C_{b1}$, $\vartheta = \arctan 2(C_{b1}, C_{a1})$ and recalling that $\sqrt{C_{a1}^2 + C_{b1}^2} = \frac{C_{a1}}{\cos \vartheta}$ yields the solution for θ_1

$$\theta_1 = \arccos \left[\frac{C_{a1}^2 + C_{b1}^2 + \ell_{11}^2 - C_{c1}^2}{2C_{a1}\ell_{11}} \cos \vartheta_1 \right] - \vartheta_1$$

B.5 Linkage, Frame Dimensions and Angle Relations

Table B.5: Linkages and frame dimension for the lower swash-plate mechanism of the Air Star Evolution Helicopter.

Linkage	Value [mm]		Description
	± 0.02	mm	
l_{11}	18.18		Linkage form j_{11} to j_{12}
l_{12}	73.40		Linkage form j_{12} to j_{13}
l_{21}	18.33		Linkage form j_{21} to j_{22}
l_{22}	73.20		Linkage form j_{22} to j_{23}
l_{31}	18.18		Linkage form j_{31} to j_{32}
l_{32}	74.16		Linkage form j_{32} to j_{33}
l_{41}	18.03		Linkage form j_{41} to j_{42}
l_{42}	73.35		Linkage form j_{42} to j_{43}
r	69.44		Lower swash-plate radius
d_1	18.66		x distance of j_{11} and j_{41} to the origin of frame C_1
d_2	1.78		y distance of j_{11} to the origin of frame C_1
d_3	47.99		y distance of j_{21} to the origin of frame C_1
d_4	19.41		x distance of j_{21} and j_{31} to the origin of frame C_1
d_5	1.93		y distance of j_{31} to the origin of frame C_1
d_6	48.52		y distance of j_{41} to the origin of frame C_1

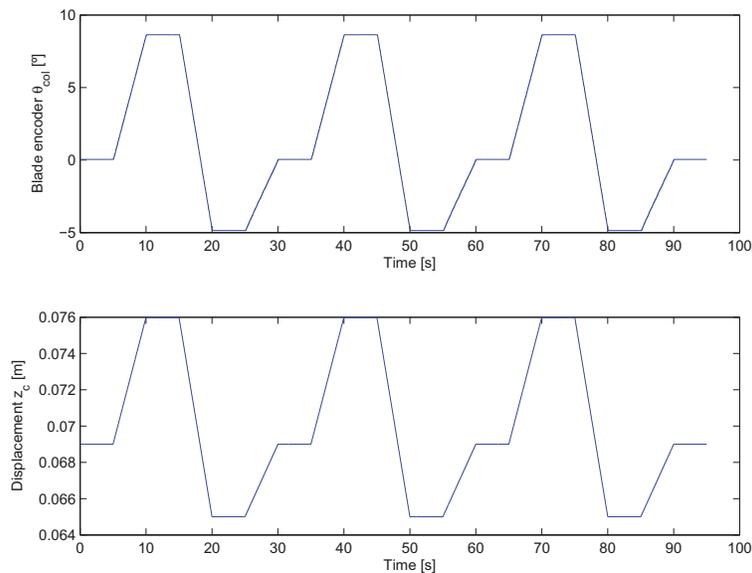


Figure B.4: Top, blade encoder collective pitch angle $\delta_{\theta_{col}}$ response vs. time. Bottom, lower swash-plate linear displacement z_c vs. time.

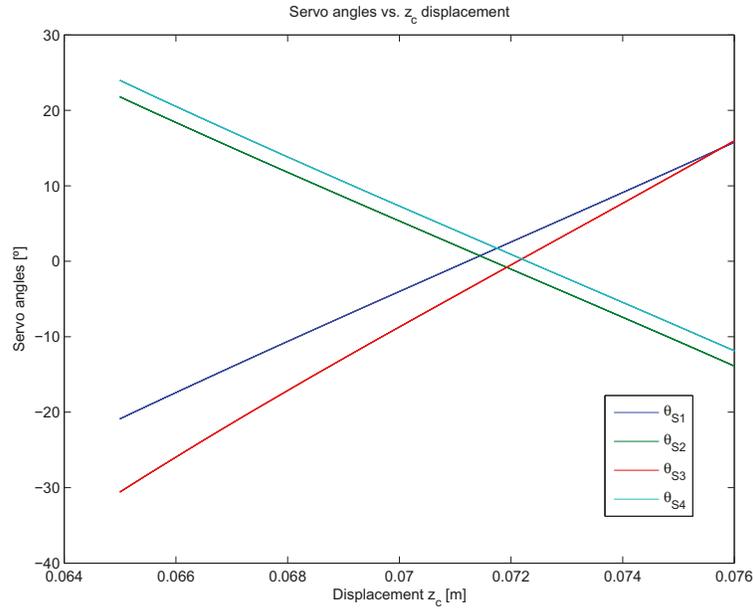
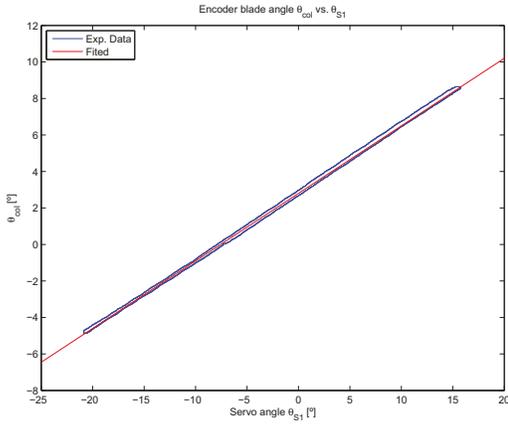


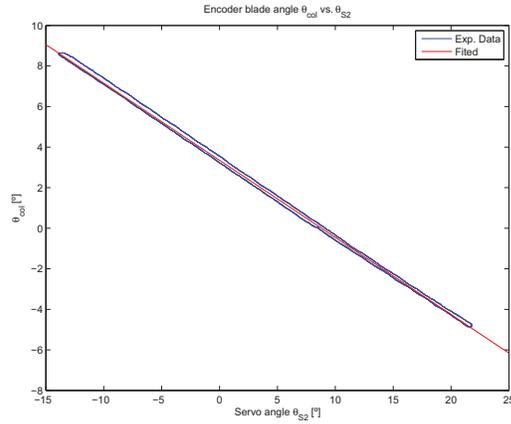
Figure B.5: Functional relation between servo angle computed using equations (3.17) to (3.20) and lower wash-plate linear displacement z_c .

Table B.6: Summary of coefficients outcomes for the linear functional relations between blade collective pitch angle $\delta\theta_{col}$, z_c , and servo angles.

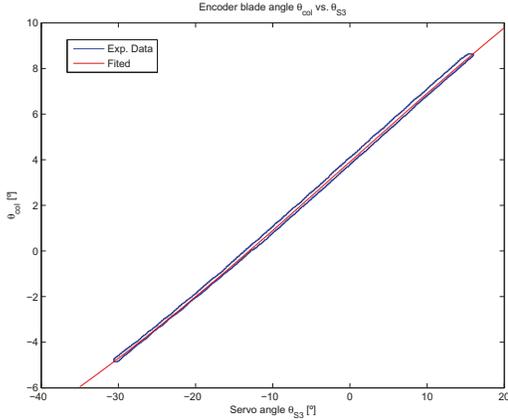
Angle relation	a_1x^3	a_2x^2	a_3x	a_4
$\delta\theta_{col}$ vs. z_c (Inclinometer)	0.0	0.0	$1170 \frac{\circ}{m}$	-80.79°
$\delta\theta_{col}$ vs. z_c (Encoder)	0.0	0.0	$1229 \frac{\circ}{m}$	-84.76°
$\delta\theta_{col}$ vs. $\theta - s_1$	0.0	0.0	0.370	2.80°
$\delta\theta_{col}$ vs. $\theta - s_2$	0.0	0.0	-0.380	3.35°
$\delta\theta_{col}$ vs. $\theta - s_3$	-1.797×10^{-5}	-4.898×10^{-5}	0.302	3.91°
$\delta\theta_{col}$ vs. θ_{s4}	0.0	0.0	-0.378	4.11°



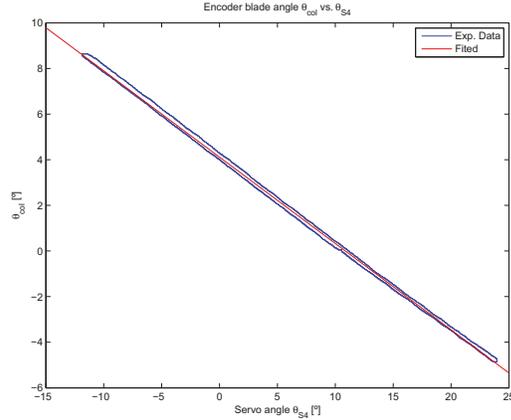
(a) Blade collective angle $\delta_{\theta_{col}}$ vs. servo angle θ_{S_1} experimental data and fitted curve.



(b) Blade collective angle $\delta_{\theta_{col}}$ vs. servo angle θ_{S_2} experimental data and fitted curve.



(c) Blade collective angle $\delta_{\theta_{col}}$ vs. servo angle θ_{S_3} experimental data and fitted curve.



(d) Blade collective angle $\delta_{\theta_{col}}$ vs. servo angle θ_{S_4} experimental data and fitted curve.

Figure B.6: Functional relation between blade collective pitch angle $\delta_{\theta_{col}}$ and servo angles. Figures (a), (b) and (d) present linear response between blade collective pitch angle and the swash-plate linear displacement z_c . Figure (c), clearly exhibit a non-linear response between the blade collective pitch angle and servo angle number 3. For this case the fitted curve is a 3rd order polynomial.

Appendix C

ECICH and GS Technical Information

C.1 Electrical Diagram

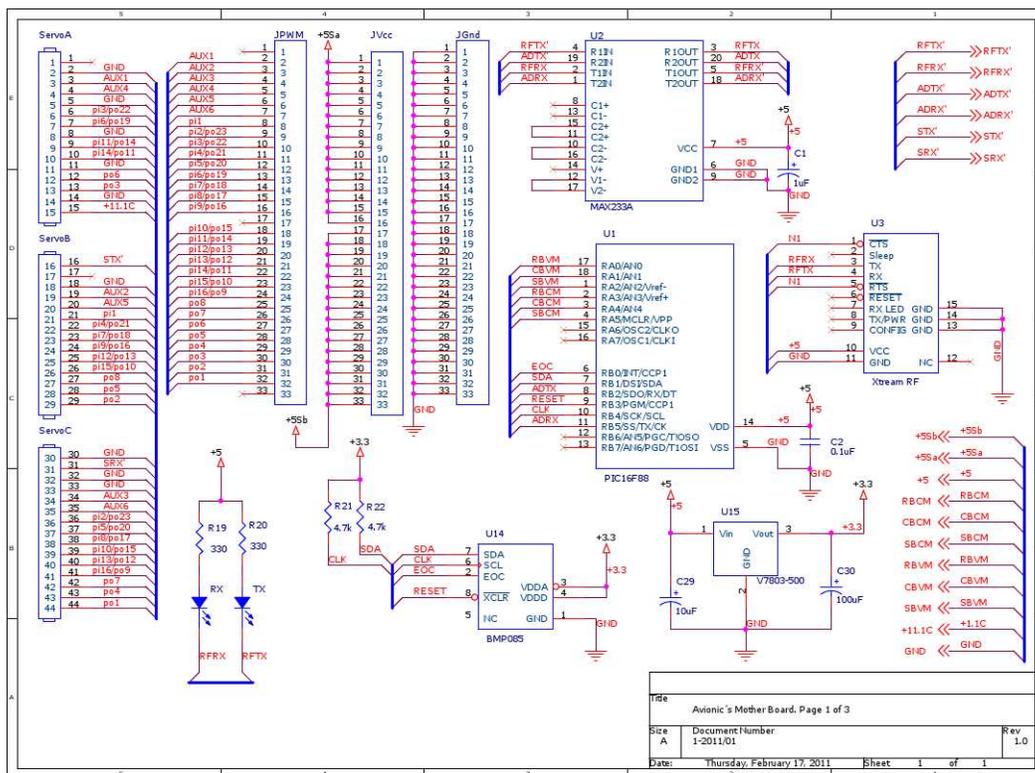


Figure C.1: ECICH Electrical Diagram page 1 of 3

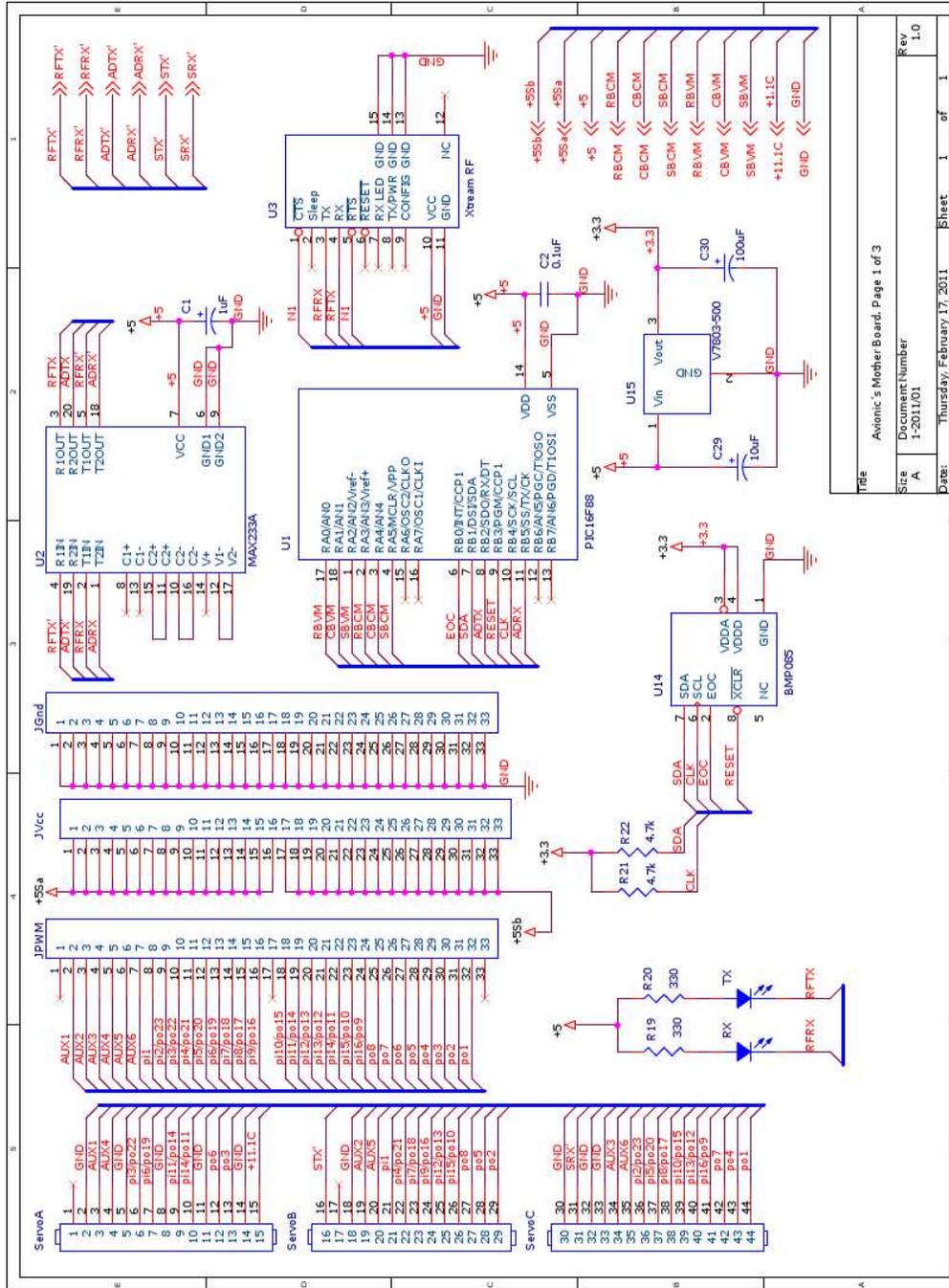
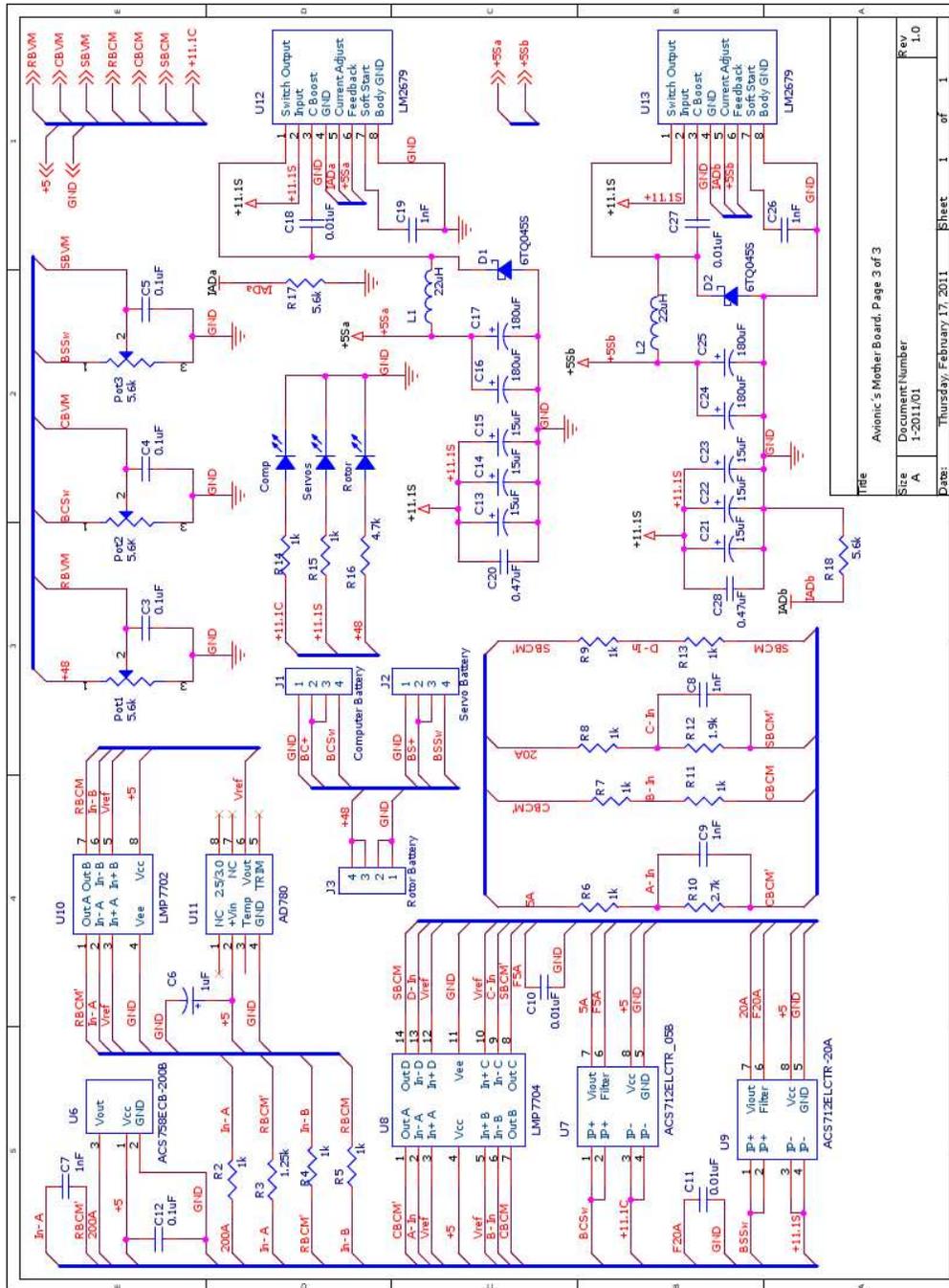


Figure C.2: ECICH Electrical Diagram page 2 of 3



Title		Avionic's Mother Board, Page 3 of 3	
Size	Document Number		
A	1-2011/01		
Date:	Thursday, February 17, 2011	Sheet	1 of 1
Rev			
1.0			

Figure C.3: ECICH Electrical Diagram page 3 of 3

C.2 BMS Plots

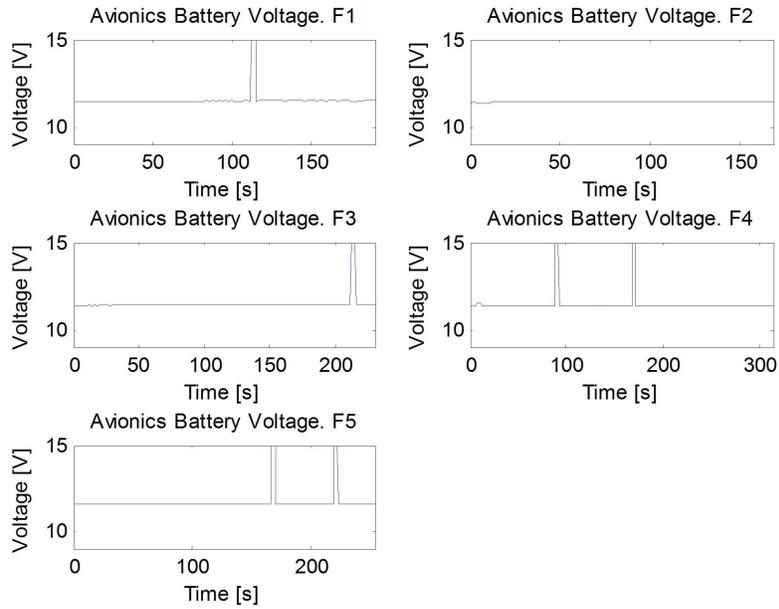


Figure C.4: *Five flights ECICH's Battery Voltages Plots*

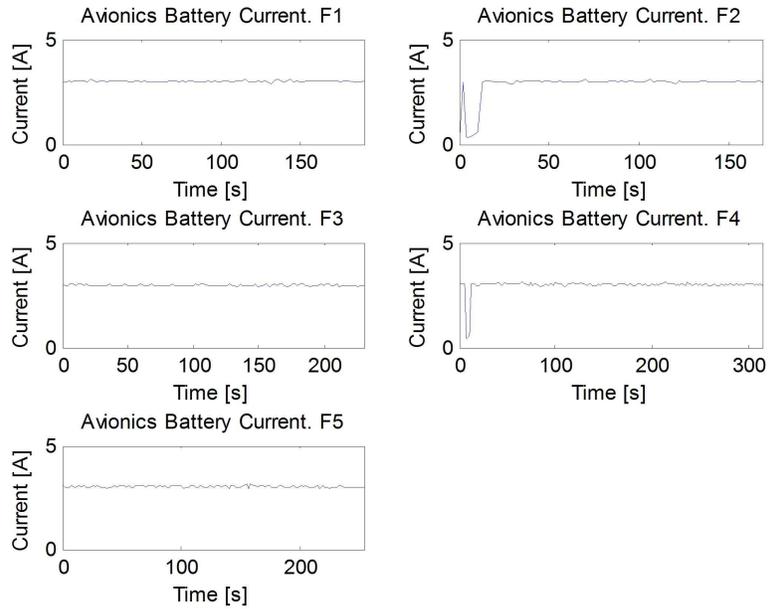


Figure C.5: *Five flights ECICH's Battery Current Plots*

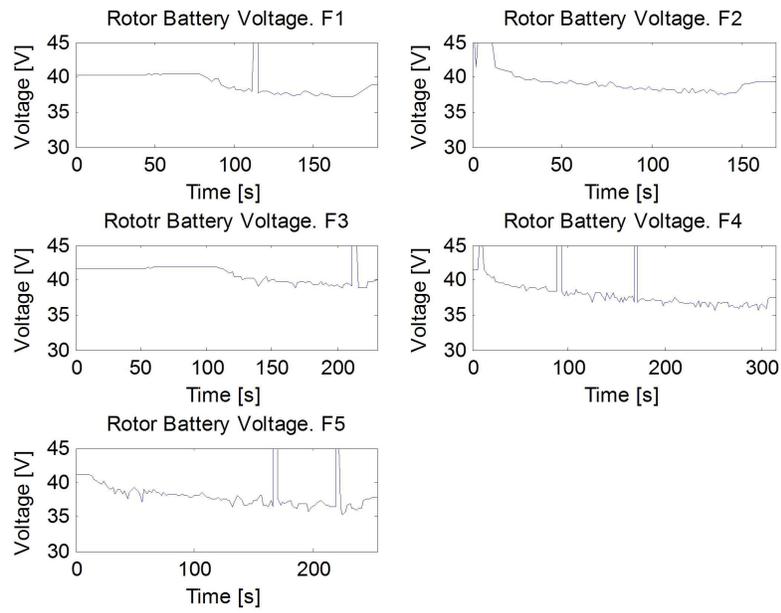


Figure C.6: *Five flights Rotor's Battery Voltages Plots*

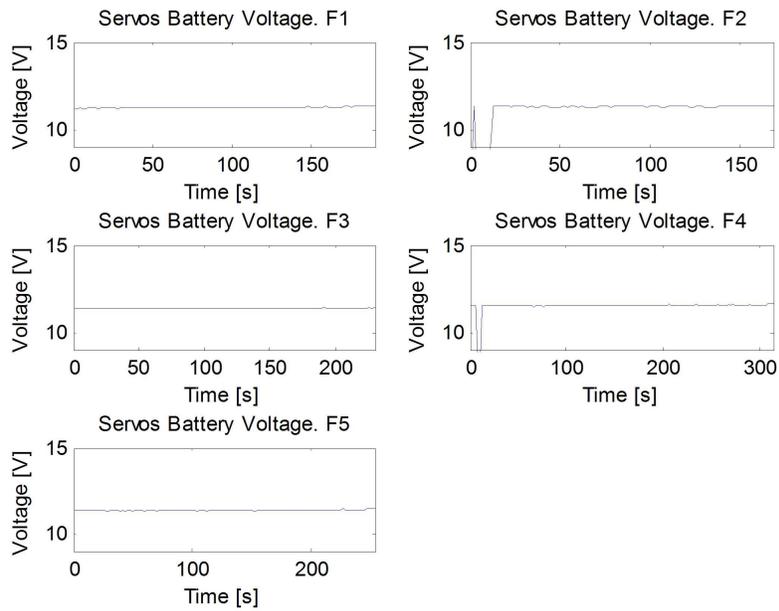


Figure C.7: *Five flights Servos' Battery Voltages Plots*

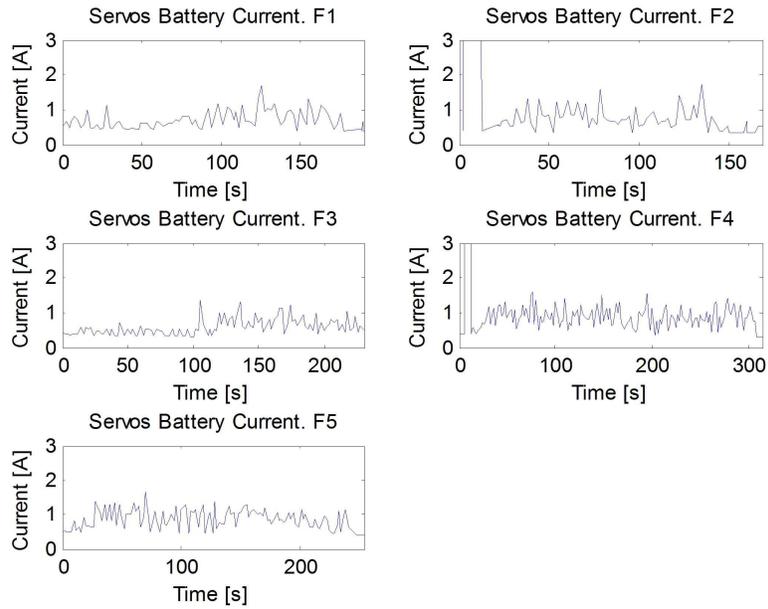


Figure C.8: *Five flights Servos' Battery Current Plots*

C.3 μ Controller program

```

;*****
;           Program Helicopter Battery Monitor.asm           *
;                                                                 *
; This program reads the voltages and currents of the avionic box *
; battery, main rotor battery and servos battery.             *
;                                                                 *
; When data are requested from the PC104 is issued, the m-controller *
; start the conversions and makes the average of the readings over *
; n times. Once the average is done, average for each channel *
; is transmitted to host through the serial port at rate *
; of 38,400 bits/s *
;                                                                 *
;           Firmware version: 1.0 *
;           Hardware: version 1.0 *
;                                                                 *
; Developed by: AMG           Date: April 2011 *
;*****

title "Attendance Register"
list p=16f88,f=inhx32
#include <p16f88.inc>
errorlevel -226
errorlevel -202

;Bit Definition

#define RVM PORTA,0 ;Rotor battery voltage monitor
#define CVM PORTA,1 ;Computer battery voltage monitor
#define SVM PORTA,2 ;Servos battery voltage monitor
#define RCM PORTA,3 ;Rotor battery current monitor
#define CCM PORTA,4 ;Computer battery current monitor
#define SCM PORTB,6 ;Servo battery current monitor

#define Tx PORTB,5 ;Data TX
#define Rx PORTB,2 ;Data RX

;***** Constants definitions *****
C_STX set 0x53 ;Start of frame 'S' character
C_ETX set 0x03 ;End of frame
LF set 0x0A ;Line fit
CR set 0x0D ;Carriage return
Req set 0x52 ;Request remote data. 'R' character
AvSize set 0x05 ;A/D Reading average over five
AD_Delay set 0x10 ;Constant for time delay for A/D acquisition time
;which is a delay of 23.5us working at 8MHz

;***** RAM Variable definition *****
;*** Variables for the transmission table

```

```

AD0H equ 0x31 ;Byte high of AD channel 0
AD0L equ 0x32 ;Byte low  of AD channel 0
AD1H equ 0x33 ;Byte high of AD channel 1
AD1L equ 0x34 ;Byte low  of AD channel 1
AD2H equ 0x35 ;Byte high of AD channel 2
AD2L equ 0x36 ;Byte low  of AD channel 2
AD3H equ 0x37 ;Byte high of AD channel 3
AD3L equ 0x38 ;Byte low  of AD channel 3
AD4H equ 0x39 ;Byte high of AD channel 4
AD4L equ 0x3A ;Byte low  of AD channel 4
AD5H equ 0x3B ;Byte high of AD channel 5
AD5L equ 0x3C ;Byte low  of AD channel 5

;*** Variables for the current values of the A/D
Ac0H equ 0x50 ;Byte high del AD channel 0 Current value
Ac0L equ 0x51 ;Byte low  del AD channel 0 Current value
Ac1H equ 0x52 ;Byte high del AD channel 1 Current value
Ac1L equ 0x53 ;Byte low  del AD channel 1 Current value
Ac2H equ 0x54 ;Byte high del AD channel 2 Current value
Ac2L equ 0x55 ;Byte low  del AD channel 2 Current value
Ac3H equ 0x56 ;Byte high del AD channel 3 Current value
Ac3L equ 0x57 ;Byte low  del AD channel 3 Current value
Ac4H equ 0x58 ;Byte high del AD channel 4 Current value
Ac4L equ 0x59 ;Byte low  del AD channel 4 Current value
Ac5H equ 0x5A ;Byte high del AD channel 5 Current value
Ac5L equ 0x5B ;Byte low  del AD channel 5 Current value

;*** Variables for the Average values and adder
Av0H equ 0x60 ;Byte high del AD channel 0 Average Value
Av0L equ 0x61 ;Byte low  del AD channel 0 Average Value
Av1H equ 0x62 ;Byte high del AD channel 1 Average Value
Av1L equ 0x63 ;Byte low  del AD channel 1 Average Value
Av2H equ 0x64 ;Byte high del AD channel 2 Average Value
Av2L equ 0x65 ;Byte low  del AD channel 2 Average Value
Av3H equ 0x66 ;Byte high del AD channel 3 Average Value
Av3L equ 0x67 ;Byte low  del AD channel 3 Average Value
Av4H equ 0x68 ;Byte high del AD channel 4 Average Value
Av4L equ 0x69 ;Byte low  del AD channel 4 Average Value
Av5H equ 0x6A ;Byte high del AD channel 5 Average Value
Av5L equ 0x6B ;Byte low  del AD channel 5 Average Value
AvH  equ 0x6C ;Average high byte of the adder
AvL  equ 0x6D ;Average low byte of the adder
AdrH equ 0x6E ;High byte of the adder
AdrL equ 0x6F ;Low byte of the adder

V_ATTx equ 0x20 ;TX pointer variable
Del equ 0x21 ;Delay variable
TxReq equ 0x22 ;Request transmission flag
AvNum equ 0x23 ;Average variable
NumeH equ 0x24 ;Numerator High byte
NumeL equ 0x25 ;Numerator Low byte
DenoH equ 0x26 ;Denominator High byte
DenoL equ 0x27 ;Denominator Low byte
RemaH equ 0x28 ;Remainder High byte

```



```

L_ISR_2
BANKSEL INTCON ;Bank 0
bsf INTCON,6 ;enable PEIE
bsf INTCON,7 ;GIE is enable

;Restore context before returning from interrupt
    movf PCLATH_TEMP,W ; retrieve copy of PCLATH register
    movwf PCLATH ; restore pre-isr PCLATH register contents
    movf STATUS_TEMP,W ; retrieve copy of STATUS register
    movwf STATUS ; restore pre-isr STATUS register contents
    swapf W_TEMP,F
    swapf W_TEMP,W ; restore pre-isr W register contents
    retfie ; return from interrupt

;***** Main program *****
Start
    call Peri_Ini ;Initialize peripheral
    call Ini_Vars ;Initialize variables
    BANKSEL INTCON ;Bank 0
    bsf INTCON,6 ;enable PEIE
    bsf INTCON,7 ;GIE is enable
    BANKSEL PORTB ;Bank 0
    bsf PORTB,3 ;Pressure sensor reset release
    goto Main_Loop

Main_Loop
    BANKSEL STATUS ;Bank 0
    movfw TxReq ;W gets Tx flg info
    xorlw 0x01 ;Check iif Tx request is pending
    btfss STATUS,Z ;Check if Zero flag is clear
    goto Main_Loop ;Continues into the Main Loop

;**** Starts Conversions and averaging ****
    call Clear_Av ;Clears the average variables
    movlw AvSize ;W takes the average number
    movwf AvNum ;File gets the number
Main_Loop_L1
    call Read_Rotor_Voltage ;Reads main battery voltage
    movfw AcOH ;Transfer high byte value to de adder
    movwf AdrH ;
    movfw AcOL ;Transfer low bye value to de adder
    movwf AdrL ;
    movfw AvOH ;Transfer the high byte of the accumulated average
    movwf AvH ;
    movfw AvOL ;Transfer the low byte of the accumulated average
    movwf AvL ;
    call D_add ;Double precision adder
    movfw AvH ;Transfer high byte result to the corresponding
    movwf AvOH ;channel
    movfw AvL ;Transfer low byte result to the corresponding
    movwf AvOL ;channel

    call Read_Rotor_Current ;Reads main battery current

```

```
movfw Ac3H ;Transfers high bye value to de adder
movwf AdrH ;
movfw Ac3L ;Transfers low bye value to de adder
movwf AdrL ;
movfw Av3H ;Transfer the high byte of the accumulated average
movwf AvH ;
movfw Av3L ;Transfer the low byte of the accumulated average
movwf AvL ;
call D_add ;Double precision adder
movfw AvH ;Transfer high byte result to the corresponding
movwf Av3H ;channel
movfw AvL ;Transfer low byte result to the corresponding
movwf Av3L ;channel

call Read_PC104_Voltage ;Reads avionics' battery voltage
movfw Ac1H ;Transfers high bye value to de adder
movwf AdrH ;
movfw Ac1L ;Transfers low bye value to de adder
movwf AdrL ;
movfw Av1H ;Transfer the high byte of the accumulated average
movwf AvH ;
movfw Av1L ;Transfer the low byte of the accumulated average
movwf AvL ;
call D_add ;Double precision adder
movfw AvH ;Transfer high byte result to the corresponding
movwf Av1H ;channel
movfw AvL ;Transfer low byte result to the corresponding
movwf Av1L ;channel

call Read_PC104_Current ;Reads avionics battery current
movfw Ac4H ;Transfers high bye value to de adder
movwf AdrH ;
movfw Ac4L ;Transfers low bye value to de adder
movwf AdrL ;
movfw Av4H ;Transfer the high byte of the accumulated average
movwf AvH ;
movfw Av4L ;Transfer the low byte of the accumulated average
movwf AvL ;
call D_add ;Double precision adder
movfw AvH ;Transfer high byte result to the corresponding
movwf Av4H ;channel
movfw AvL ;Transfer low byte result to the corresponding
movwf Av4L ;channel

call Read_Servos_Voltage ;Reads servos' battery voltage
movfw Ac2H ;Transfers high bye value to de adder
movwf AdrH ;
movfw Ac2L ;Transfers low bye value to de adder
movwf AdrL ;
movfw Av2H ;Transfer the high byte of the accumulated average
movwf AvH ;
movfw Av2L ;Transfer the low byte of the accumulated average
movwf AvL ;
call D_add ;Double precision adder
movfw AvH ;Transfer high byte result to the corresponding
```

```

movwf Av2H ;channel
movfw AvL ;Transfer low byte result to the corresponding
movwf Av2L ;channel

call Read_Servos_Current ;Reads servos battery current
movfw Ac5H ;Transfers high byte value to de adder
movwf ADrH ;
movfw Ac5L ;Transfers low byte value to de adder
movwf ADrL ;
movfw Av5H ;Transfer the high byte of the accumulated average
movwf AvH ;
movfw Av5L ;Transfer the low byte of the accumulated average
movwf AvL ;
call D_add ;Double precision adder
movfw AvH ;Transfer high byte result to the corresponding
movwf Av5H ;channel
movfw AvL ;Transfer low byte result to the corresponding
movwf Av5L ;channel

; BANKSEL STATUS ;Bank 0
;   movfw TxReq ;W gets Tx flg info
; xorlw 0x01 ;Check iif Tx request is pending
; btfsc STATUS,2 ;Check if Zero flag is clear
; call Transmit ;Transmit data

decfsz AvNum,f ;Decrements file and if zero skips
goto Main_Loop_L1 ;Continues with the average computation
call Compute_Av ;Routine that computes average
call Transmit ;Transmit data
    goto Main_Loop ;Jumps to main Loop

;*****
;*
;*   Peripheral initialization routines
;*
;*****

Peri_Ini

;Set Internal Oscillator at 8 Mhz (4MHz)
BANKSEL OSCCON ;
movlw 0x70 ;Internal oscillator 8 Mhz (4 Mhz old value 0x60)
movwf OSCCON ;
nop
nop

;Set port A and A/D port
BANKSEL PORTA ;Port A bank select
    clrf PORTA ;
BANKSEL ANSEL ;Analog selector register
movlw 0x3F ;Bit 0 to 5 as analog inputs, bit 6 digital I/O
movwf ANSEL ;
movlw 0xFF ;All bits on trisa are inputs
movwf TRISA ;

```

```

BANKSEL TRISB ;Bank 1
    movlw 0xD7 ;RB3 and RB5 are out, rest are inputs
movwf TRISB ;
BANKSEL PORTB ;Bank 0
bcf PORTB,3 ;Reset for the Pressure sensor
BANKSEL ADCON1 ;
movlw 0X0C ;RIGHT JUSTIFIED, ADCS2=1,VREF+=Vdd,VREF-=Vss
movwf ADCON1 ;
nop

BANKSEL ADCON0
bcf ADCON0,0 ;A/D off
nop
nop
    bsf ADCON0,6 ;Focs/16, Look ADCON1

;***** UART 1 Initialization *****
BANKSEL TXSTA ;Select register
movlw 0x24 ;Enable TX and select BRGH=1
movwf TXSTA ;
    BANKSEL SPBRG ;Select bank 3
movlw 0x0C ;Constant for baud rate of 38400
movwf SPBRG ;
BANKSEL RCSTA ;Select bank 0
movlw 0x90 ;Enable Rx and continuous Rx
movwf RCSTA ;
BANKSEL PIE1 ;Bank 1
bsf PIE1,5 ;Enable Rx interrupt
return ;Routine ends

;***** I2C port initialization *****

;*****
;
;           Routine variable initialization
;
;*****
Ini_Vars

;Initializing Rx and Tx data table pointers
BANKSEL STATUS ;Bank 0
movlw TTx ;Beginning of the rx data table
movwf V_ATTx ;
movlw TTx ;W gets table first position
movwf FSR ;Reg gets indirect pointer
movlw C_STX ;W gets first position of frame
movwf INDF ;
movlw TTx+0x0D ;W gets pointer +13
movwf FSR ;Reg gets the pointer
movlw LF ;W gets Line feed
movwf INDF ;Variable gets beginning frame

```

```

movlw TTx+0x0E ;W gets beginning table
movwf FSR ;Reg gets pointer
movlw LF ;W gets LINE feed
movwf INDF ;Variable gets frame beginning
movlw TTx+0x0F ;W gets initial table position
movwf FSR ;Dir Reg gets pointer
movlw CR ;W gets CR
movwf INDF ;
clrf TxReq ;Clear Tx request flag
return ;Routine ends

;*****
;*
;*          Program Routines
;*
;*****

;*****
;*
;* Routine gets_A/D data. This routine gets data from
;* A/D converter and store them in its corresponding
;* variables. Latter on an average of n readings will
;* be made
;*
;*****

;Main Rotor battery voltage reading
Read_Rotor_Voltage
BANKSEL ADCON0 ;
clrf ADCON0 ;Clear ADCON0 and select AN0
bsf ADCON0,6 ;Fosc/(8 or 16) see ADCS2 on ADCON1
nop
    nop
    bsf ADCON0,0 ;AD on
    movlw AD_Delay ;Constant delay 23.5 us approx at 8 MHz
movwf Del ;
L_RRV_2
decfsz Del,1
goto L_RRV_2
bsf ADCON0,2 ;Start Conversion
L_RRV_1
btfsc ADCON0,2 ;Check if finish
goto L_RRV_1 ;If not continue waiting

; movlw 0x10 ;
; movwf Del ;
;L_RRV_4
; decfsz Del,1 ;
; goto L_RRV_4 ;

BANKSEL ADRESL ;
movfw ADRESL ;
BANKSEL ADCON0 ;

```

```

movwf AcOL ;Save Data
movfw ADRESH ;
movwf AcOH ;
clrf ADCON0 ;AD off
return ;

;Avionic box battery voltage reading
Read_PC104_Voltage
BANKSEL ADCON0 ;
    clrf ADCON0 ;Clears ADCON0
bsf ADCON0,3 ;Select AN1
    bsf ADCON0,6 ;Fosc/(8 or 16) see ADCS2 on ADCON1
nop
nop
    bsf ADCON0,0 ;AD on
    movlw AD_Delay ;Constant delay 23.5 us approx at 8 MHz
movwf Del ;Variable takes the value
L_RPV_2
decfsz Del,1
goto L_RPV_2
bsf ADCON0,2 ;Start Conversion
L_RPV_1
btfsc ADCON0,2 ;Check if finish
goto L_RPV_1 ;If not continue waiting

; movlw 0x10 ;
; movwf Del ;Variable takes the value
;L_RPV_4
; decfsz Del,1
; goto L_RPV_4

BANKSEL ADRESL ;Select the bank 1
movfw ADRESL ;
BANKSEL ADCON0 ;Bank 0
movwf Ac1L ;save Data
    movfw ADRESH ;W gets high byte result
movwf Ac1H ;Save data
clrf ADCON0 ;AD off
return ;Routine ends

;Servoss battery voltage reading
Read_Servos_Voltage
BANKSEL ADCON0 ;
    clrf ADCON0 ;Clears ADCON0
bsf ADCON0,4 ;Select AN2
    bsf ADCON0,6 ;Fosc/(8 or 16) see ADCS2 on ADCON1
nop
nop
    bsf ADCON0,0 ;AD on
    movlw AD_Delay ;Constant delay 23.5 us approx at 8 MHz
movwf Del ;Variable takes the value
L_RSV_2
decfsz Del,1

```

```

goto L_RSV_2
bsf ADCON0,2 ;Start Conversion
L_RSV_1
btfsc ADCON0,2 ;Check if finish
goto L_RSV_1 ;If not continue waiting

;   movlw 0x10 ;
; movwf Del   ;Variable takes the value
;L_RSV_4
; decfsz Del,1
; goto   L_RSV_4

BANKSEL ADRESL ;Select the bank 1
movfw ADRESL ;
BANKSEL ADCON0 ;Bank 0
movwf Ac2L ;save Data
    movfw ADRESH ;W gets high byte result
movwf Ac2H ;Save data
clrf ADCON0 ;AD off
return ;Routine ends

;Main Rotor battery current reading
Read_Rotor_Current
BANKSEL ADCON0 ;
    clrf ADCON0 ;Clears ADCON0
bsf ADCON0,3 ;Select AN3
bsf   ADCON0,4 ;
    bsf ADCON0,6 ;Fosc/(8 or 16) see ADCS2 on ADCON1
nop
nop
    bsf ADCON0,0 ;AD on
    movlw AD_Delay ;Constant delay 23.5 us approx at 8 Mhz
movwf Del ;Variable takes the value
L_RRC_2
decfsz Del,1
goto L_RRC_2
bsf ADCON0,2 ;Start Conversion
L_RRC_1
btfsc ADCON0,2 ;Check if finish
goto L_RRC_1 ;If not continue waiting

;   movlw 0x10 ;
; movwf Del   ;Variable takes the value
;L_RRC_4
; decfsz Del,1
; goto   L_RRC_4

BANKSEL ADRESL ;Select the bank 1
movfw ADRESL ;
BANKSEL ADCON0 ;Bank 0
movwf Ac3L ;save Data
    movfw ADRESH ;W gets high byte result
movwf Ac3H ;Save data
clrf ADCON0 ;AD off

```

```

return ;Routine ends

;Avionic box battery current reading
Read_PC104_Current
BANKSEL ADCON0 ;
    clrf ADCON0 ;Clears ADCON0
bsf ADCON0,5 ;Select AN4
    bsf ADCON0,6 ;Fosc/(8 or 16) see ADCS2 on ADCON1
nop
nop
    bsf ADCON0,0 ;AD on
    movlw AD_Delay ;Constant delay 23.5 us approx at 8 MHz
movwf Del ;Variable takes the value
L_RPC_2
decfsz Del,1
goto L_RPC_2
bsf ADCON0,2 ;Start Conversion
L_RPC_1
btfsc ADCON0,2 ;Check if finish
goto L_RPC_1 ;If not continue waiting

;    movlw 0x10 ;
; movwf Del ;Variable takes the value
;L_RPC_4
; decfsz Del,1
; goto L_RPC_4

BANKSEL ADRESL ;Select the bank 1
movfw ADRESL ;
BANKSEL ADCON0 ;Bank 0
movwf Ac4L ;save Data
    movfw ADRESH ;W gets high byte result
movwf Ac4H ;Save data
clrf ADCON0 ;AD off
return ;Routine ends

;Servoss battery current reading
Read_Servos_Current
BANKSEL ADCON0 ;
    clrf ADCON0 ;Clears ADCON0
bsf ADCON0,3 ;Select AN5
bsf ADCON0,5 ;
    bsf ADCON0,6 ;Fosc/(8 or 16) see ADCS2 on ADCON1
nop
nop
    bsf ADCON0,0 ;AD on
    movlw AD_Delay ;Constant delay 23.5 us approx at 8 MHz
movwf Del ;Variable takes the value
L_RSC_2
decfsz Del,1
goto L_RSC_2
bsf ADCON0,2 ;Start Conversion
L_RSC_1
btfsc ADCON0,2 ;Check if finish

```

```

goto L_RSC_1 ;If not continue waiting

; movlw 0x10 ;
; movwf Del ;Variable takes the value
;L_RSC_4
; decfsz Del,1
; goto L_RSC_4

BANKSEL ADRESL ;Select the bank 1
movfw ADRESL ;
BANKSEL ADCONO ;Bank 0
movwf Ac5L ;save Data
    movfw ADRESH ;W gets high byte result
movwf Ac5H ;Save data
clrf ADCONO ;AD off
return ;Routine ends

;Transmit data routine
Transmit
BANKSEL 0 ;Bank 0
movlw 0x10 ;Table length
movwf V_ATTx ;
movlw TTx ;W gets beginning table
movwf FSR ;Reg gets pointer
L_Trans_2
movfw INDF ;W gets data
L_Trans_1
    BANKSEL TXSTA ;Select TXSTAT
    btfss TXSTA,1 ;Check if TX buffer is empty
    goto L_Trans_1 ;If not wait
    BANKSEL TXREG ;
    movwf TXREG ;
    incf FSR,1 ;
    decfsz V_ATTx,1 ;Decrement counter
    goto L_Trans_2
    BANKSEL STATUS ;Bank 0
    clrf TxReq ;Clear Tx Request flag
    return

;***** Double precision addition *****
; This routine receives the adder on variables AdrH and AdrL
;And return de result on variables AvH, AvL
D_add
BANKSEL AdrL ;
movf AdrL,w ;W takes first term
addwf AvL,f ;Makes low byte addition
btfss STATUS,C ;
goto D_add_L1 ;if no carry jumps
movlw 0x01 ;
addwf AvH,f ;Adds carry to second high byte
D_add_L1
movfw AdrH ;W takes higher byte
addwf AvH,f ;Adds High bytes
return ;Routine ends

```

```
;Routine that clears all average and adder variables
Clear_Av
clrf AvH ;Clears adder variables
clrf AvL
clrf AdrH
clrf AdrL
clrf Av0H ;Clears channel average variables
clrf Av0L
clrf Av1H
clrf Av1L
clrf Av2H
clrf Av2L
clrf Av3H
clrf Av3L
clrf Av4H
clrf Av4L
clrf Av5H
clrf Av5L
return

;Routine that computes the average and transfer results to the transmission table
Compute_Av
clr w ;Clear accumulator
movwf DenoH ;Denominator high byte takes zero
movlw AvSize ;w takes low byte of denominator
movwf DenoL ;Variable takes data

;****Channel 0 average
movf Av0H,w ;
movwf NumeH ;
movf Av0L,w ;
movwf NumeL ;
call D_div ;
movf QuotH,w ;Transfers channel 0 values
movwf ADOH ;
movf QuotL,w ;
movwf ADOL ;

;****Channel 1 average
movf Av1H,w ;
movwf NumeH ;
movf Av1L,w ;
movwf NumeL ;
call D_div ;
movf QuotH,w ;Transfers channel 1 values
movwf AD1H ;
movf QuotL,w ;
movwf AD1L

;****Channel 2 average
movf Av2H,w ;
movwf NumeH ;
movf Av2L,w ;
movwf NumeL ;
```

```

call D_div ;
movf QuotH,w ;Transfers channel 2 values
movwf AD2H ;
movf QuotL,w ;
movwf AD2L

;****Channel 3 average
movf Av3H,w ;
movwf NumeH ;
movf Av3L,w ;
movwf NumeL ;
call D_div ;
movf QuotH,w ;Transfers channel 3 values
movwf AD3H ;
movf QuotL,w ;
movwf AD3L

;****Channel 4 average
movf Av4H,w ;
movwf NumeH ;
movf Av4L,w ;
movwf NumeL ;
call D_div ;
movf QuotH,w ;Transfers channel 4 values
movwf AD4H ;
movf QuotL,w ;
movwf AD4L

;****Channel 5 average
movf Av5H,w ;
movwf NumeH ;
movf Av5L,w ;
movwf NumeL ;
call D_div ;
movf QuotH,w ;Transfers channel 0 values
movwf AD5H ;
movf QuotL,w ;
movwf AD5L

return ;routine ends

;*****
; Double Precision Division *
; *
; ( Optimized for Code : Looped Code ) *
; *
;*****
; Division : Nume(16 bits) / Deno(16 bits) -> Nume(16 bits) with
; Remainder in Rema (16 bits)
; (a) Load the Denominator in location DenoH & DenoL ( 16 bits )
; (b) Load the Numerator in location NumeH & NumeL ( 16 bits )
; (c) CALL D_div
; (d) The 16 bit result is in location NumeH & NumeL
; (e) The 16 bit Remainder is in locations RemaH & RemaL

```

```

;
; Performance :
; Program Memory : 31 (UNSIGNED)
; Clock Cycles : 300 (UNSIGNED : excluding CALL & RETURN)
;
;*****
; Double Precision Divide ( 16/16 -> 16 )
;
; ( Nume/Deno -> Nume with remainder in Rema ) : 16 bit output
; with Quotient in Nume (NumeH,NumeL) and Remainder in Rema (RemaH,RemaL).
;
; B/A = (Q) + (R)/A
; or B = A*Q + R
;
; where B : Numerator
; A : Denominator
; Q : Quotient (Integer Result)
; R : Remainder
;
; Note : Check for ZERO Denominator or Numerator is not performed
; A ZERO Denominator will produce incorrect results
;
;*****
D_div
clrf count
bsf count,4 ; set count = 16
clrf RemaH
clrf RemaL
clrf QuotL
clrf QuotH
;
; Looped code
;
divLoop
bcf STATUS,C
rlf NumeL, F
rlf NumeH, F
rlf RemaL, F
rlf RemaH, F
movf DenoH,w
subwf RemaH,w ; check if a>c
btfss STATUS,Z
goto notz
movf DenoL,w
subwf RemaL,W ; if msb equal then check lsb
notz
btfss STATUS,C ; carry set if c>a
goto nosub ; if c < a
subca
movf DenoL,w ; c-a into c
subwf RemaL,F
movf DenoH,w
;**** Implementation of instruction subwfb

```

```
subwf RemaH,F ;Substract f-w
btfss STATUS,C ;Check if borrow
goto D_div_L1
clrw ;If borrow then complement carry
subwf RemaH,F ;Substract f-w'
goto D_div_L2
D_div_L1
movlw 0x01 ;If not borrow complement carry
subwf RemaH,F ;Substract f-W'
D_div_L2
;**** end of implementation
bsf STATUS,C ; shift a 1 into d (result)
nosub
rlf QuotL,F
rlf QuotH,F
decfsz count,F
goto divLoop
return

end
```


C.4 MATLAB[®] Programs and Simulink Block Diagrams

C.4.1 SSC and RPM Programs

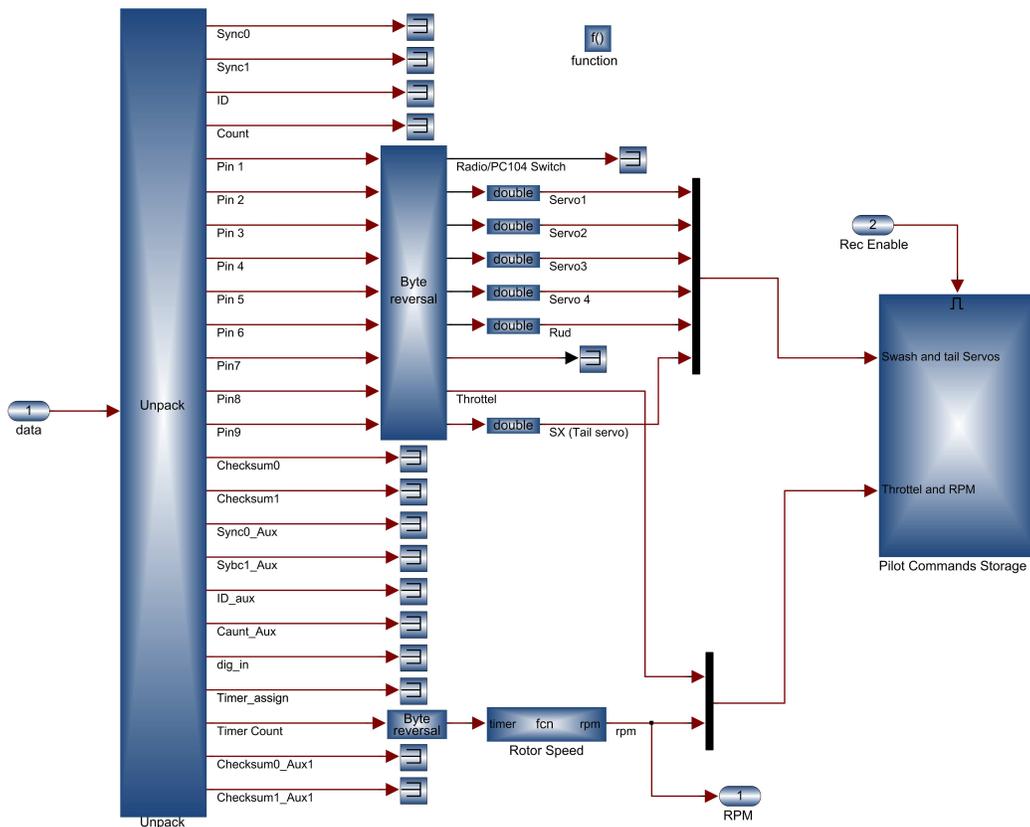


Figure C.9: Simulink[®] PWM Computation from Servo Switch Card Block.

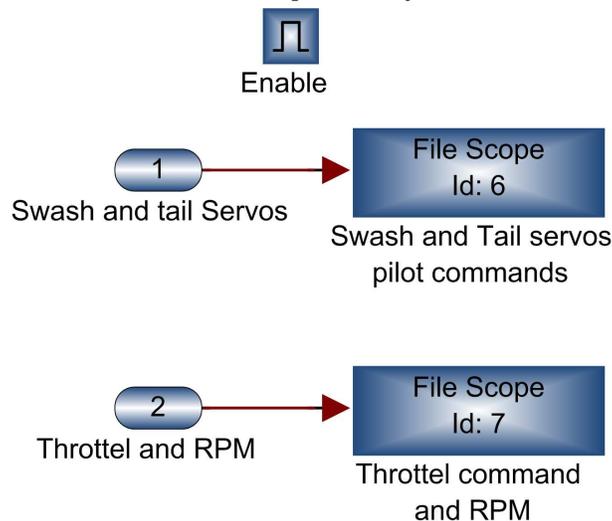


Figure C.10: Simulink[®] Pilot's Commands Storage Block.

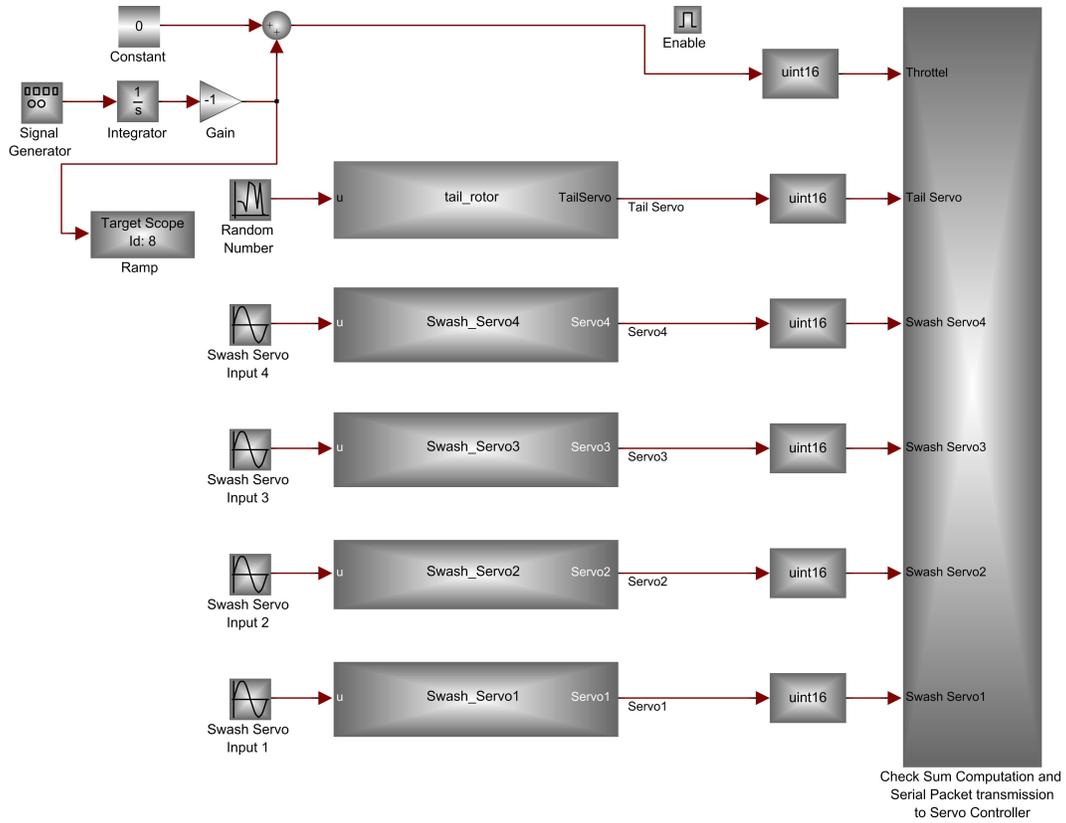


Figure C.11: *Simulink® PC104 Controller Block.*

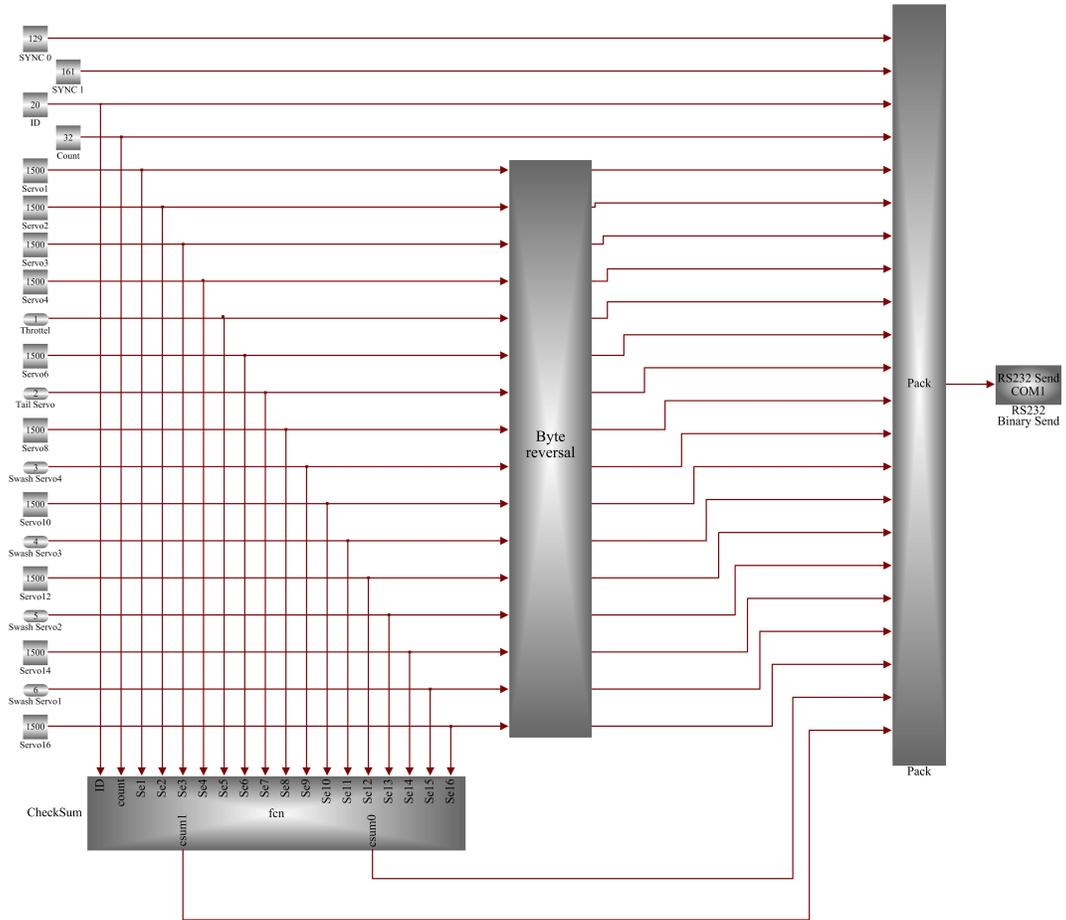


Figure C.12: *Simulink[®] Check Sum Computation and Serial Packet Transmission to Servo Controller Block.*

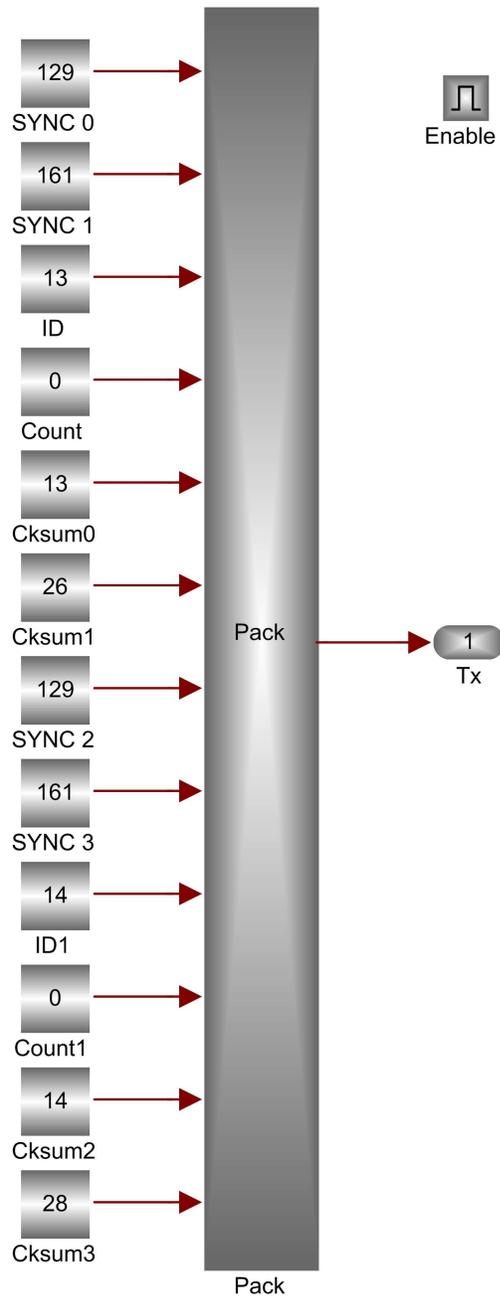


Figure C.13: *Simulink® Servo Switch Card Data Request Block.*

Listing C.1: *Program to compute Rotor Speed based on the RPM sensor readings.*

```
function rpm = fcn(timer)
% The goal of this function is to compute the RPM base on the
%Auxiliary 1 input pin on the Servo switch card
%This function does an average of n consecutive readings
```

```

%Local persistent variables
persistent Arpm Crpm Index

if isempty(Arpm)
    Arpm=uint16(0); Index=0; Crpm=uint16(0);
end

%Constant definitions
n=5;      % number of readings or average

% The preset time base in the SSC configuration is as follows,
time_base = 32; % microseconds

% break into bytes
timer_a=uint8(bitshift(timer,-8));
timer_b=uint8(timer-uint16(timer_a)*256);

timer_a=uint8(bitshift(timer_a,2));
timer_a=uint8(bitshift(timer_a,-2));

pack = double(uint16(timer_a)*256+uint16(timer_b));

if Index == n
    rpm=Arpm/n;
    Crpm=Crpm;
    Index=0;Arpm=uint16(0);
else
    Arpm =Arpm + uint16(6e7/(6*pack*time_base));
    rpm=Crpm;
    Index=Index+1;
end

```

Listing C.2: *Program to compute the Fletcher Checksum.*

```

function [csum1,csum0]= fcn(ID,count,Se1,Se2,Se3,Se4,Se5,Se6,Se7,Se8,...
    Se9,Se10,Se11,Se12,Se13,Se14,Se15,Se16)

% The purpose of this function is to calculate the fletcher checksum values
% as defined in internet RFC 1145
% See page 11 of the Microbotics Servo Switch/Controller Users Manual for
% more information

% break into bytes
%Se1H=uint8(bitshift(Se1,-8)); Se1L=uint8(Se1-uint16(Se1H)*256);
Se1H=bitshift(Se1,-8); Se1L=mod(Se1,256);
Se2H=bitshift(Se2,-8); Se2L=mod(Se2,256);
Se3H=bitshift(Se3,-8); Se3L=mod(Se3,256);
Se4H=bitshift(Se4,-8); Se4L=mod(Se4,256);
Se5H=bitshift(Se5,-8); Se5L=mod(Se5,256);
Se6H=bitshift(Se6,-8); Se6L=mod(Se6,256);
Se7H=bitshift(Se7,-8); Se7L=mod(Se7,256);
Se8H=bitshift(Se8,-8); Se8L=mod(Se8,256);

```

```
Se9H=bitshift (Se9, -8); Se9L=mod (Se9, 256);
Se10H=bitshift (Se10, -8); Se10L=mod (Se10, 256);
Se11H=bitshift (Se11, -8); Se11L=mod (Se11, 256);
Se12H=bitshift (Se12, -8); Se12L=mod (Se12, 256);
Se13H=bitshift (Se13, -8); Se13L=mod (Se13, 256);
Se14H=bitshift (Se14, -8); Se14L=mod (Se14, 256);
Se15H=bitshift (Se15, -8); Se15L=mod (Se15, 256);
Se16H=bitshift (Se16, -8); Se16L=mod (Se16, 256);
byte=uint16 ([ID, count, Se1H, Se1L, Se2H, Se2L, Se3H, Se3L, Se4H, Se4L, Se5H, Se5L, ...
             Se6H, Se6L, Se7H, Se7L, Se8H, Se8L, Se9H, Se9L, Se10H, Se10L, ...
             Se11H, Se11L, Se12H, Se12L, Se13H, Se13L, Se14H, Se14L, Se15H, Se15L, ...
             Se16H, Se16L]);

cksum0=uint16 (0); cksum1=uint16 (0); % initialize to zero
for i=1:(count+2)
    cksum0=mod ((cksum0+byte (i)), 256);
    cksum1=mod ((cksum1+cksum0), 256);
end
csum0=uint8 (cksum0); csum1=uint8 (cksum1);
```

C.4.2 IMU Programs

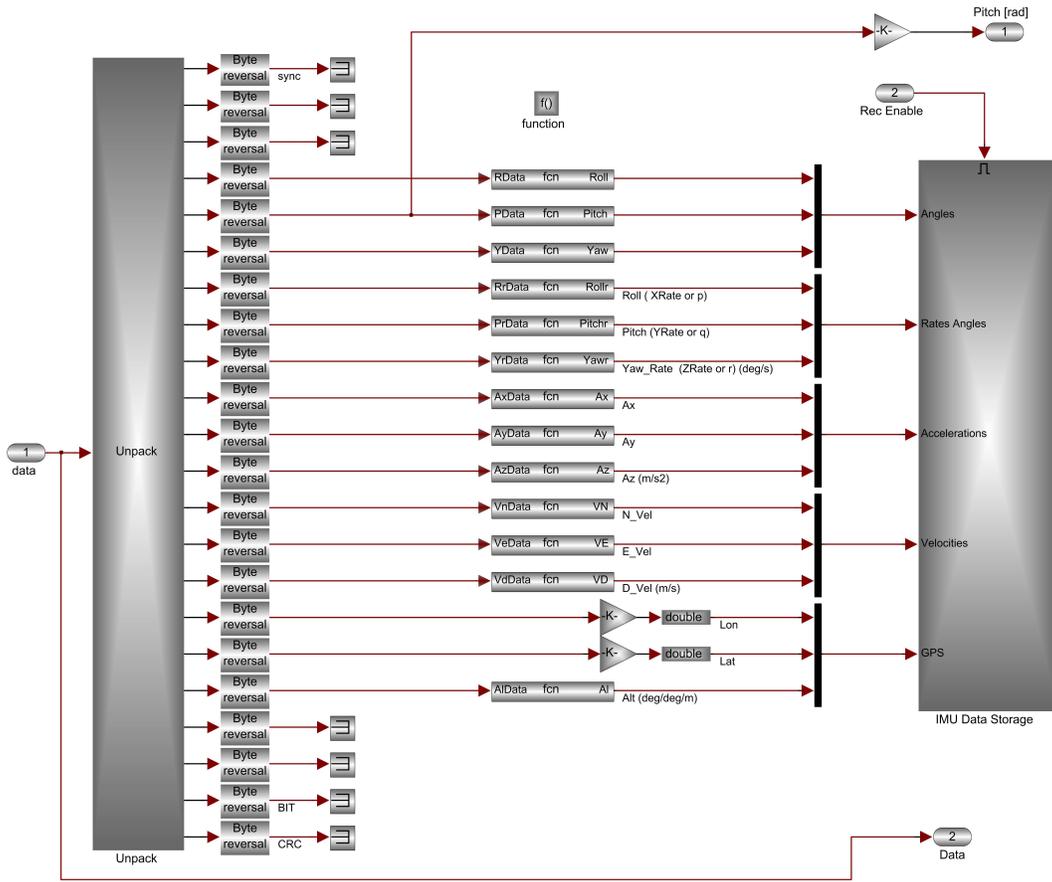


Figure C.14: Simulink[®] IMU Data Block.

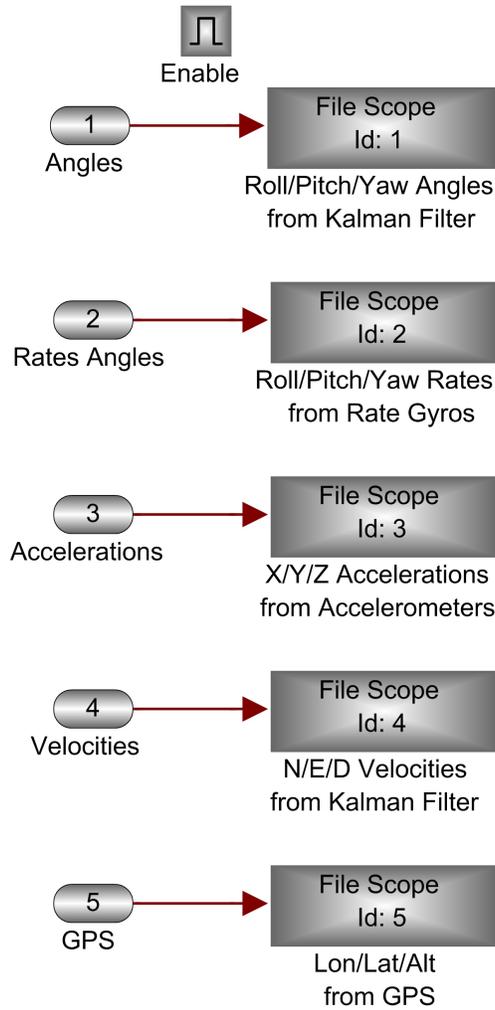


Figure C.15: Simulink® IMU Data Storage Block.

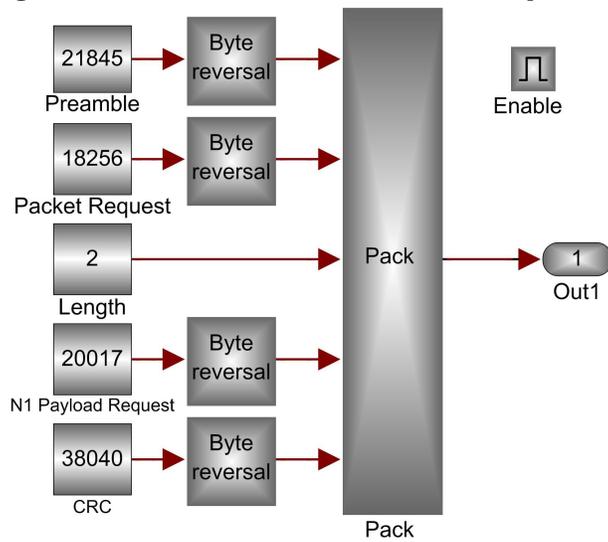


Figure C.16: Simulink® Crossbow Data Request Block.

Listing C.3: *Functions to compute and convert to proper units, the data coming from the IMU.*

```
function Roll = fcn(RData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

Dat=double(RData);
if Dat > 32767
    Dat=Dat-65536;
end
Roll=Dat*360/2^16;
end

function Pitch = fcn(PData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

Dat=double(PData);
if Dat > 32767
    Dat=Dat-65536;
end
Pitch=Dat*360/2^16;
end

function Yaw = fcn(YData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

Dat=double(YData);
if Dat > 32767
    Dat=Dat-65536;
end
Yaw=Dat*360/2^16;
end

function Rollr = fcn(RrData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.

Dat=double(RrData);
if Dat > 32767
    Dat=Dat-65536;
end
Rollr=Dat*1260/2^16;
end

function Pitchr = fcn(PrData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(PrData);
if Dat > 32767
    Dat=Dat-65536;
end
Pitchr=Dat*1260/2^16;
end
```

```
function Yawr = fcn(YrData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(YrData);
if Dat > 32767
    Dat=Dat-65536;
end
Yawr=Dat*1260/2^16;
end
```

```
function Ax = fcn(AxData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(AxData);
if Dat > 32767
    Dat=Dat-65536;
end
Ax=Dat*20/2^16;
end
```

```
function Ay = fcn(AyData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(AyData);
if Dat > 32767
    Dat=Dat-65536;
end
Ay=Dat*20/2^16;
end
```

```
function Az = fcn(AzData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(AzData);
if Dat > 32767
    Dat=Dat-65536;
end
Az=Dat*20/2^16;
end
```

```
function VN = fcn(VnData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(VnData);
if Dat > 32767
    Dat=Dat-65536;
end
VN=Dat*512/2^16;
end
```

```
function VE = fcn(VeData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(VeData);
if Dat > 32767
    Dat=Dat-65536;
end
VE=Dat*512/2^16;
end
```

```
function VD = fcn(VdData)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```
Dat=double(VdData);
if Dat > 32767
    Dat=Dat-65536;
end
VD=Dat*512/2^16;
end
```

C.4.3 Range Sensor Programs

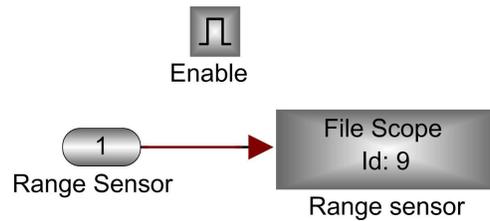


Figure C.17: *Simulink[®] Storage Range Sensor Altitude Data Block.*

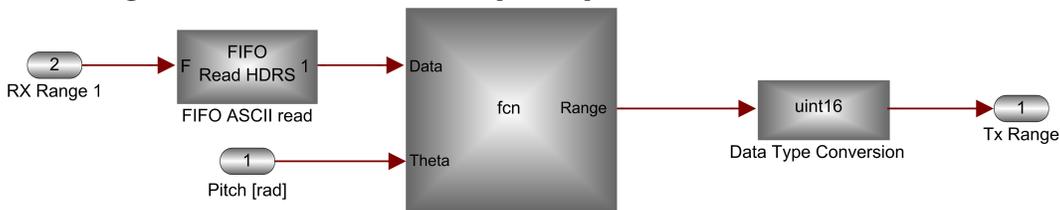


Figure C.18: *Simulink[®] Range Computation Block.*

Listing C.4: *Frame construction to request data to RS.*

```
function Tx = RangeTx(Clock)

%Reange request steps from 213 to 555 in clusters of 64
Dat0 = uint16(71);   %0x47  G
Dat1 = uint16(51);   %0x33  3
Dat2 = uint16(54);   %0x36  6
Dat3 = uint16(56);   %0x38  8
Dat4 = uint16(52);   %0x34  4
Dat5 = uint16(48);   %0x30  0
Dat6 = uint16(48);   %0x30  0
Dat7 = uint16(54);   %0x33  6
Dat8 = uint16(52);   %0x32  4
Dat9 = uint16(10);   %0x0A  Ctr + J

if Clock == 1
    Size= uint16(10);
else
    Size= uint16(0);
end
byte = [Size ,Dat0 ,Dat1 ,Dat2 ,Dat3 ,Dat4 ,Dat5 ,Dat6 ,Dat7 ,Dat8 ,Dat9 ];
Tx=uint16(byte);
```

Listing C.5: *Program to decode data coming from the RS.*

```
function Range = fcn(Data, Theta)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
```

```

% outputs the command channel state (CCS) given status output

%constant values
k1=uint16(63);      %0x3F
k2=uint16(48);      %0x30

% First check if no errors and if the end of frame is correct
% Error = 0x300A and End= 0x0a0A
%MinRange=uint16(5000);
%value=uint16(zeros(1,11));
if (uint8(Data(1))==71) %&& (uint8(Data(11))==48) %Header==0x47, Error==0x30
    k=13;
%   for k=13:2:(length(Data)-3)
        bh=uint16(Data(k));
        bl=uint16(Data(k+1));
        bh=bh-k2;
        bl=bl-k2;
        bh=bitand(bh,k1);
        bl=bitand(bl,k1);
        bh=bitshift(bh,6);
%       value((k-11)/2)=uint16(bitxor(bh,bl));
        value=uint16(bitxor(bh,bl));
%   end           %End for
%   for m = 1:11
%       if value(m) ~= 0;
%           if MinRange > value(m)
%               MinRange=value(m);
%           end %End second if
%       end %End first if
%   end %end for
%   Range=MinRange*abs(cos(Theta));
    Range=value*abs(cos(Theta));
else
    Range=uint16(4100);
end

```

C.4.4 BMS Programs

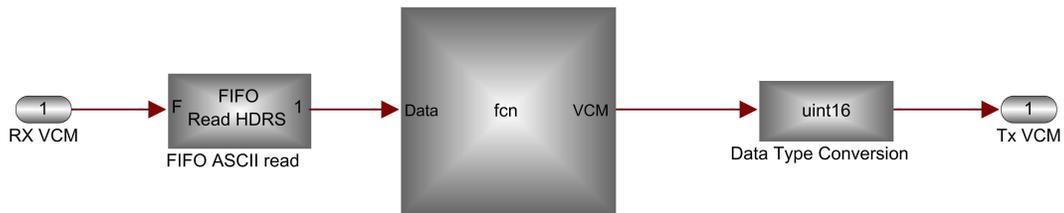


Figure C.19: Simulink® Voltage Current Monitor Block.

Listing C.6: BMS data request code.

```
function Tx = ADRequest(Clock)

%A/D Data Request for Voltages and current monitor
Dat0 = uint16(82); %0x52 "R"

if Clock == 1
    Size= uint16(1);
else
    Size= uint16(0);
end
byte = [Size ,Dat0];
Tx=uint16(byte);
```

C.4.5 GS Programs

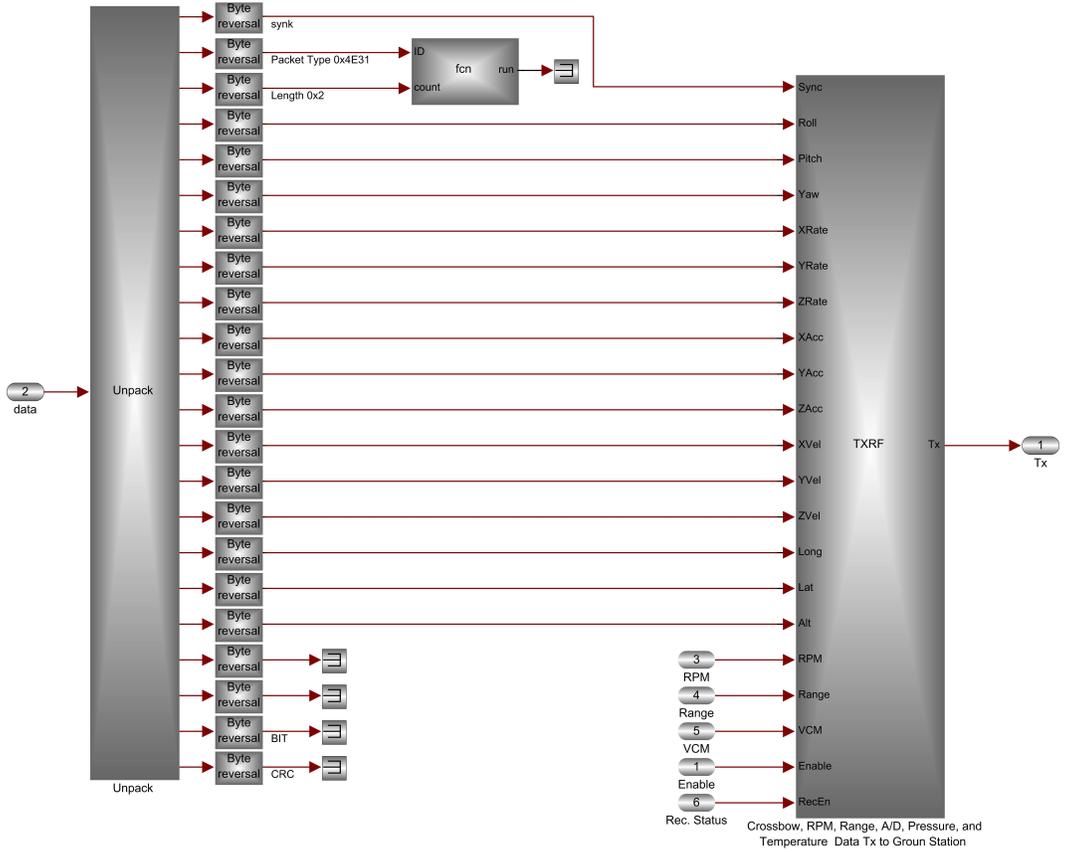


Figure C.20: Simulink[®] Ground Station Data Transmission Block.

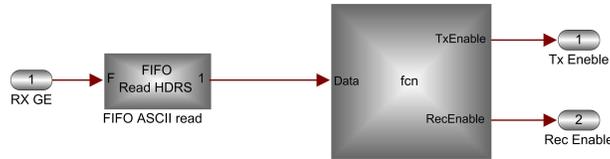


Figure C.21: *Simulink[®] Ground Station Command Execution Block.*

Listing C.7: *Data Frame Construction to transmit parameters to GS*

```
function Tx = TXRF(Sync, Roll, Pitch, Yaw, XRate, YRate, ZRate, ...
                  XAcc, YAcc, ZAcc, XVel, YVel, ZVel, Long, Lat, ...
                  Alt, RPM, Range, VCM, Enable, RecEn)

%Frame size
if Enable == 0
    Size = uint16(67);
else
    Size = uint16(0);
end

%Angles
Dat0 = uint16(bitshift(Sync, -8));
Dat1 = Sync;
Dat2 = uint16(bitshift(Roll, -8));
Dat3 = Roll;
Dat4 = uint16(bitshift(Pitch, -8));
Dat5 = Pitch;
Dat6 = uint16(bitshift(Yaw, -8));
Dat7 = Yaw;

%Rate Angles
Dat8 = uint16(bitshift(XRate, -8));
Dat9 = XRate;
Dat10 = uint16(bitshift(YRate, -8));
Dat11 = YRate;
Dat12 = uint16(bitshift(ZRate, -8));
Dat13 = ZRate;

%Accelerations
Dat14 = uint16(bitshift(XAcc, -8));
Dat15 = XAcc;
Dat16 = uint16(bitshift(YAcc, -8));
Dat17 = YAcc;
Dat18 = uint16(bitshift(ZAcc, -8));
Dat19 = ZAcc;

%Velocity
Dat20 = uint16(bitshift(XVel, -8));
Dat21 = XVel;
Dat22 = uint16(bitshift(YVel, -8));
Dat23 = YVel;
```

```

Dat24 = uint16 ( bitshift ( ZVel, -8));
Dat25 = ZVel;

%Longitude GPS data
Dat26=uint16 ( bitshift ( Long, -24));
Dat27=uint16 ( bitshift ( Long, -16));
Dat28=uint16 ( bitshift ( bitshift ( Long,8), -16));
Dat29=uint16 ( bitshift ( bitshift ( Long,16), -16));

%Latitude GPS data
Dat30=uint16 ( bitshift ( Lat, -24));
Dat31=uint16 ( bitshift ( Lat, -16));
Dat32=uint16 ( bitshift ( bitshift ( Lat,8), -16));
Dat33=uint16 ( bitshift ( bitshift ( Lat,16), -16));

%Altitude GPS Data
Dat34 = uint16 ( bitshift ( Alt, -8));
Dat35 = Alt;

%RPM Data
RPMH=uint16 ( bitshift ( RPM, -8));
RPML=RPM;

%Range Data
RangeH=uint16 ( bitshift ( Range, -8));
RangeL=Range;

%Voltage and Current Monitor "VCM"
ABVH=VCM(1);ABVL=VCM(2);
RBVH=VCM(3);RBVL=VCM(4);
SBVH=VCM(5);SBVL=VCM(6);
ABCH=VCM(7);ABCL=VCM(8);
RBCH=VCM(9);RBCL=VCM(10);
SBCH=VCM(11);SBCL=VCM(12);

%Battery delivered charge
AQd3=VCM(13);AQd2=VCM(14);AQd1=VCM(15);AQd0=VCM(16);
RQd3=VCM(17);RQd2=VCM(18);RQd1=VCM(19);RQd0=VCM(20);
SQd3=VCM(21);SQd2=VCM(22);SQd1=VCM(23);SQd0=VCM(24);

%Recording status byte
RecS=uint16 (RecEn);

%End of frame characters
LF=uint16 (10);
CR=uint16 (13);

byte = [ Size , Dat0 , Dat1 , Dat2 , Dat3 , Dat4 , Dat5 , Dat6 , Dat7 , Dat8 , Dat9 , Dat10 , ...
        Dat11 , Dat12 , Dat13 , Dat14 , Dat15 , Dat16 , Dat17 , Dat18 , Dat19 , Dat20 , ...
        Dat21 , Dat22 , Dat23 , Dat24 , Dat25 , Dat26 , Dat27 , Dat28 , Dat29 , Dat30 , ...
        Dat31 , Dat32 , Dat33 , Dat34 , Dat35 , RPMH, RPML, RangeH, RangeL , ...
        ABVH, ABVL, RBVH, RBVL, SBVH, SBVL, ABCH, ABCL, RBCH, RBCL, SBCH, SBCL, ...
        AQd3, AQd2, AQd1, AQd0, RQd3, RQd2, RQd1, RQd0, SQd3, SQd2, SQd1, SQd0 , ...
        RecS, LF, CR ] ;
Tx=uint16 (byte );

```

Listing C.8: *Routine that interprets the command sent by the GS to ECICH.*

```
function [TxEnable, RecEnable] = fcn(Data)
% This block supports an embeddable subset of the MATLAB language.
% See the help menu for details.
% outputs the command channel state (CCS) given status output

%Flag for Recording data. Local persistent variable. FRec=0 in not recording
persistent FRec

if isempty(FRec)
    FRec=0;
end

%Command analysis for transmission
if (Data(1) == 'U') && (Data(2) == uint8(06))
    TxEnable=0;
else
    TxEnable=1;
end

%Command Analysis for storage data
if (Data(1) == 'U') && (Data(2) == uint8(82)) %82 dec= 'R'
    FRec=1;
end
if (Data(1) == 'U') && (Data(2) == uint8(83)) %83 dec= 'S'
    FRec=0;
end
RecEnable=FRec;
```

C.5 Ground Station Program

```

*****
*
*                               Main unit
*
*****

unit GPS;

interface

uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs, DB,
  StdCtrls, ExtCtrls, ComSerie, SDL_Meter, SDL_NumLab, hhAvHSI, hhAvADI, Math,
  hhAvComp, hhAvGS, hhAvALT, Buttons, StrUtils, MPlayer, SDL_ProgBar,
  SDL_gradfl, ComCtrls, XYZ, Menus, IBCtrls, UPortSettings, UIPD, UGPSData,
  hhAvVS, UBatCap, DBTables, UVCMonitorDB, DateUtils;

type
  TGPSMain = class(TForm)
    RxTimer: TTimer;
    RPM: TMeter;
    NLRPM: TNumLab;
    NLAltitude: TNumLab;
    P_Vel: TPanel;
    StaticText1: TStaticText;
    P_Attitude: TPanel;
    P_Compas: TPanel;
    NLPitch: TNumLab;
    NLRoll: TNumLab;
    StaticText19: TStaticText;
    StaticText20: TStaticText;
    NLCompass: TNumLab;
    Altimeter: ThhAvALT;
    StaticText22: TStaticText;
    Audio: TMediaPlayer;
    PBRange: TProgBar;
    LRange: TLabel;
    StaticText17: TStaticText;
    Horizont: TXYZ;
    PVollMonitor: TPanel;
    AvVoltage: TMeter;
    AVCurr: TMeter;
    DAvVoltage: TNumLab;
    DAvCurr: TNumLab;
    StaticText18: TStaticText;
    MainMenu1: TMainMenu;
    Com1: TMenuItem;
    Open1: TMenuItem;
    Close1: TMenuItem;
    Settings1: TMenuItem;
    InertialData1: TMenuItem;
  end;

```

```

GPSData1: TMenuItem;
Exit1: TMenuItem;
StatBar: TStatusBar;
SetGroundLevel1: TMenuItem;
SetGPSEstationPosition1: TMenuItem;
P_RPM: TPanel;
StaticText2: TStaticText;
VSpeed: ThhAvVS;
LVSpeed: TLabel;
StaticText3: TStaticText;
Panel1: TPanel;
RotVoltage: TMeter;
RotCurr: TMeter;
DRVoltage: TNumLab;
DRCurr: TNumLab;
Panel2: TPanel;
SerVoltage: TMeter;
SerCurr: TMeter;
DSVoltage: TNumLab;
DSCurr: TNumLab;
AVCap: TProgBar;
RotCap: TProgBar;
SerCap: TProgBar;
AvBc: TLabel;
SetBatteriesCapacities1: TMenuItem;
RotBc: TLabel;
SerBc: TLabel;
DSVCMonitor: TDataSource;
TVCMonitor: TTable;
VCDDataBase1: TMenuItem;
StartRecording1: TMenuItem;
StopRecording1: TMenuItem;
ShowDataBase1: TMenuItem;
ErraseDataBase1: TMenuItem;
Edit2: TEdit;
Edit1: TEdit;
Commands1: TMenuItem;
StartHelicopterTx1: TMenuItem;
StopHelicopterTx1: TMenuItem;
ReqTx: TTimer;
HelicopterStartRecording1: TMenuItem;
HelicopterStopRecording1: TMenuItem;
TDelay: TTimer;
Compass: ThhAvHSI;
ExporttoTextFiledat1: TMenuItem;
SaveD: TSaveDialog;
procedure FormCreate(Sender: TObject);
procedure RxTimerTimer(Sender: TObject);
procedure Exit1Click(Sender: TObject);
procedure Close1Click(Sender: TObject);
procedure Open1Click(Sender: TObject);
procedure Settings1Click(Sender: TObject);
procedure InertialData1Click(Sender: TObject);
procedure GPSData1Click(Sender: TObject);
procedure SetGroundLevel1Click(Sender: TObject);

```

```

    procedure SetGPSEstationPosition1Click(Sender: TObject);
    procedure SetBatteriesCapacities1Click(Sender: TObject);
    procedure ShowDataBase1Click(Sender: TObject);
    procedure StopRecording1Click(Sender: TObject);
    procedure StartRecording1Click(Sender: TObject);
    procedure ErraseDataBase1Click(Sender: TObject);
    procedure StartHelicopterTx1Click(Sender: TObject);
    procedure StopHelicopterTx1Click(Sender: TObject);
    procedure ReqTxTimer(Sender: TObject);
    procedure HelicopterStartRecording1Click(Sender: TObject);
    procedure HelicopterStopRecording1Click(Sender: TObject);
    procedure StatBarDrawPanel(StatusBar: TStatusBar; Panel: TStatusPanel;
        const Rect: TRect);
    procedure TDelayTimer(Sender: TObject);
    procedure ExporttoTextFiledat1Click(Sender: TObject);
private
    { Private declarations }
    procedure modificarEstadoControles( estado: boolean);
    Procedure Recepcion_Datos;
    Procedure Recibe;
    Procedure Angles;
    Procedure Rates;
    Procedure ClearAll;
    function HexToDec(Dat: String): Integer;
    function Hex32ToDec(Dat: String): longword;
    Procedure RPMComp;
    Procedure Accelerations;
    Procedure Velocity;
    Procedure Longitude;
    Procedure Latitude;
    Procedure GPSAltitude;
    Procedure Range;
    Procedure AVoltage;
    Procedure RVoltage;
    Procedure SVoltage;
    Procedure ACurrent;
    Procedure RCurrent;
    Procedure SCurrent;
    procedure Tx_Ack_Comm;
    Procedure Distance;
    Procedure Lla2ECEF(Long, Lat, Alt: Extended; var x, y, z: Extended);
    Procedure HRecStatus;
    Procedure Dummy;
    Procedure StartRecordingData;
    Procedure StopRecordingData;
    procedure Save2Text;
public
    { Public declarations }
end;

{Constant definitions}
const
    SP=67;           // Size Package
    ag=3;           // Angle start pointer
    rt=9;           // Rates start pointer

```

```

ac=15;      // Acceleration start pointer
ve=21;      // Velocities start pointer
lo=27;      // Longitude start pointer
la=31;      // Latitude Start Pointer
al=35;      // GPS Altitude start pointer
Prpm=37;    // RPM Pointer
PRange=39;  // Range pointer
PAvV=41;    // Avionic voltage pointer
PRV=43;     // Rotor voltage pointer
PSV=45;     // Servos voltage pointer
PAC=47;     // Avionic current pointer
PRC=49;     // Rotor current pointer
PSC=51;     // Servos current pointer
PAQd3=53;   // Avionic Battery Delivered charge pointer
PRQd3=57;   // Rotor Battery Delivered charge pointer
PSQd3=61;   // Servos Battery Delivered charge pointer
PRecS=65;   // Pointer to the byte of recording status 0= no recording
Ts=0.05;    // Sampling time
Average=1;  // No. of Average

// Conversion factors
Delta_Ang=360/65536;      // Angle conversion
Delta_Rat=1260/65536;    // Angle rates conversion
Delta_Acc=20/65536;      // Acceleration conversion factor
Delta_Vel=512/65536;    // Velocity conversion factor
Delta_Loa=360/4294967296; // Long and Lat conversion factor
Delta_Alt=16384/65536;  // GPS Altitude conversion factor
DeltaAvV=12/1024;       // Avionic voltage conversion factor
DeltaRV=50/1024;        // Rotor voltage conversion factor
DeltaSV=12/1024;        // Servos voltage conversion factor
DeltaAC=5/1024;         // Avionic current conversion factor
DeltaRC=5/1024;         // Rotor current conversion factor
DeltaSC=5/1024;         // Servos current conversion factor

var
GPSMain: TGPSMain;
Puerto: TPuertoSerie;
Buffer1 , Buffer : string;
Cont: Integer;
GRLevel: Real;
GRLFlag, GPSMonFlag: Boolean;
NoV, NoR, DatV, DatR: integer;
AvBCap, RotBCap, SerBCap: Real;
AQd, RQd, SQd: Real;      // Variables for the delivered charge
AQt, RQt, SQt: Real;     // Variables for total charge
AQR, RQR, SQR: Real;     // Variables for the battery remainder charge
iA1, iAo, iR1, iRo, iS1, iSo: Real;
ABCcurrent, SBCcurrent: Real;
AvCount, SCount: Integer;
Rec, Tx: String;        // Variable for remote transmission and recording data
VCFlag: Boolean;        // Flag to start recording voltages and currents on DB
CTime: TDateTime;      // Current time variable
TFuture: TDateTime;
Sync: Boolean;          // Variables for check synchro of Received frame

```

```

Tam, Point : Integer ;
TxFlag : Boolean ;           // Variable to start transmission from Helicopter
GELong, GELat, GEAlt : Extended ; // Variables for distance computation
HLong, HLat, HAlt : Extended ;
Fcom : Boolean ;           // Flag for open or close Com port
FHRec : Boolean ;         // Flag for Recording or not Data on Helicopter 's memory
DatAnt : Integer ;
CommandF : Integer ;       //Command Flag Pending
                             // CommandF=0: No command is pending
                             // CommandF=1: Start Recording Data on CPU memory
                             // CommandF=2: Stop recording Data on CPU memory
FileName : String ;        //Variable to store data in the filed name
FName : TextFile ;         //Variable of file name

```

implementation

```
{$R *.DFM}
```

```
Procedure TGPSMain. ClearAll;
```

```
begin
```

```

  NLPitch. Value:=0.0; NLRoll. Value:=0.0;
  NLAltitude. Value:=0; NLRPM. Value:=0;
  RPM. Value:=0;
  Horizont. Pitch:=0; Horizont. Roll:=0; Horizont. CmdPitch:=0; Horizont. CmdRoll:=0;
  Compass. Bearing:=17; Compass. Course1:=0; //Compass. Constraints:=0;
  Altimeter. Altitude:=0;
  PBRange. Value:=0.0; LRange. Caption:='0.0';
  AvVoltage. Value:=0; DAvVoltage. Value:=0.0;
  RotVoltage. Value:=0; DRVoltage. Value:=0.0;
  VSpeed. Speed:=0; LVSpeed. Caption:='0.0';

```

```
end;
```

```
// Rutina de conversion de hexadecimal a decimal de dos bytes representados
// en una cadena de dos caracteres. La conversin es de 16 bits a decimal
```

```
function TGPSMain.HexToDec(Dat : String) : Integer;
```

```
var
```

```
  i : Integer;
```

```
  H : Real;
```

```
begin
```

```
  Dat:=IntToHex(Ord(Dat[1]),2)+IntToHex(Ord(Dat[2]),2);
```

```
  H:=0;
```

```
  for i:=1 to 4 do
```

```
    if Dat[i] in ['0'..'9'] then
```

```
      H:=H+StrToInt(Dat[i])*Power(16,4-i)
```

```
    else
```

```
      H:=H+(Ord(Dat[i])-55)*Power(16,4-i);
```

```
  Result:=Trunc(H);
```

```
end;
```

```
// Rutina de conversion de hexadecimal a decimal de dos bytes representados
// en una cadena de dos caracteres. La conversin es de 32 bits a decimal
```

```

function TGPSMain.Hex32ToDec(Dat: String): LongWord;
var
  i: Integer;
  H: Real;
begin
  Dat:=IntToHex(Ord(Dat[1]),2)+IntToHex(Ord(Dat[2]),2)+IntToHex(Ord(Dat[3]),2)+
    IntToHex(Ord(Dat[4]),2);
  H:=0;
  for i:=1 to 8 do
    if Dat[i] in ['0'..'9'] then
      H:=H+StrToInt(Dat[i])*Power(16,8-i)
    else
      H:=H+(Ord(Dat[i])-55)*Power(16,8-i);
  Result:=Trunc(H);
end;

procedure TGPSMain.FormCreate(Sender: TObject);
begin
  GRLFlag:=False;
  ClearAll;
  RPM.Value:=0;
  Horizont.Pitch:=0;Horizont.Roll:=0;Horizont.CmdPitch:=0;Horizont.CmdRoll:=0;
  Compass.Bearing:=0;Compass.Course1:=0;Altimeter.Altitude:=0;
  cont:=0;
  PBRange.Value:=0.0;LRange.Caption:='0.0';

  AvVoltage.Value:=0;DAvVoltage.Value:=0.0;
  AvCurr.Value:=0.0;DAvCurr.Value:=0.0;

  RotVoltage.Value:=0;DRVoltage.Value:=0.0;
  RotCurr.Value:=0.0;DRCurr.Value:=0;

  SerVoltage.Value:=0;DsVoltage.Value:=0.0;
  SerCurr.Value:=0.0;DSCurr.Value:=0;

  NoV:=0;NoR:=0;DatV:=0;DatR:=0;
  VSpeed.Speed:=0;LVSpeed.Caption:='0.0';

  AvCount:=0;ABCCurrent:=0;
  Rec:='S';Tx:='P';
  Sync:=False;

  // Open Current and Voltage Data base
  TVCMonitor.Open;
  VCFlag:=False;
  TxFlag:=False;
  StatBar.Panels[8].Text:='VCDB:No Rec.';

  CTime:=Now;
  Commands1.Enabled:=False;

  // Format for status bar
  StatBar.Panels[0].Style:=PsOwnerDraw;

```

```

StatBar.Panels [1]. Style:=PsOwnerDraw;
StatBar.Panels [2]. Style:=PsOwnerDraw;
StatBar.Panels [3]. Style:=PsOwnerDraw;
StatBar.Panels [4]. Style:=PsOwnerDraw;
StatBar.Panels [5]. Style:=PsOwnerDraw;
StatBar.Panels [6]. Style:=PsOwnerDraw;
StatBar.Panels [7]. Style:=PsOwnerDraw;
StatBar.Panels [8]. Style:=PsOwnerDraw;
StatBar.Panels [9]. Style:=PsOwnerDraw;
StatBar.Panels [10]. Style:=PsOwnerDraw;
StatBar.Panels [11]. Style:=PsOwnerDraw;

// Initializing variables for distance from GE to Helicopter computaion
GELong:=0.0;
GELat:=0.0;
GEAlt:=0.0;
HAlt:=0.0;

// Initializing com port flag
FCom:=False;

// Initializing Helicopter's Recordin flag
FHRec:=False;
DatAnt:=10;

// Initializing commadF
CommandF:=0;
end;

procedure TGPSMain.modificarEstadoControles( estado: boolean);
begin
  If not estado then
  begin
    FCom:=True;
    StatBar.Panels [3]. Text:='Status: Open';
    StatBar.Panels [3]. Bevel:=pbRaised;
  end
  else
  begin
    FCom:=False;
    StatBar.Panels [3]. Text:='Status: Close';
    StatBar.Panels [3]. Bevel:=pbLowered;
  end;
  Close1.Checked:=estado; Close1.Default:=estado;
  Open1.Checked:=not(estado); Open1.Default:=not(estado);
  Settings1.Enabled:=estado;
  InertialData1.Enabled:=not(estado);
  GPSData1.Enabled:=not(estado);
  SetGroundLevel1.Enabled:=not(estado);
  SetGPSEstationPosition1.Enabled:=not(estado);
  SetBatteriesCapacities1.Enabled:=estado;
  RxTimer.Enabled:=not(estado);
end;

```

```

Procedure TGPSMain.Angles;
var
  Dat: Integer;
  Heading: Integer;
begin
  // Roll angle
  Dat:=HexToDec(Buffer1[ag]+Buffer1[ag+1]);
  if Dat >32767 then
    Dat:=Dat-65536;
  if FIPD.Showing then
    begin
      FIPD.EdRoll.Clear;FIPD.EdRoll.SelText:=FloatToStrF(Dat*Delta_Ang,ffFixed,5,3);
    end;
  Horizont.Roll:=Trunc(Dat*Delta_Ang);
  Horizont.CmdRoll:=Trunc(Dat*Delta_Ang);
  NLRoll.Value:=Dat*Delta_Ang;
  If (Dat*Delta_Ang <-60) or (Dat*Delta_Ang >60) then
    begin
      Audio.FileName:='C:\Alar2.wav';
      Audio.Play;
    end;
  // Pitch angle
  Dat:=HexToDec(Buffer1[ag+2]+Buffer1[ag+3]);
  if Dat >32767 then
    Dat:=Dat-65536;
  if FIPD.Showing then
    begin
      FIPD.EdPitch.Clear;FIPD.EdPitch.SelText:=FloatToStrF(Dat*Delta_Ang,ffFixed,5,3);
    end;
  Horizont.Pitch:=Trunc(Dat*Delta_Ang);
  Horizont.CmdPitch:=Trunc(Dat*Delta_Ang);
  NLPitch.Value:=Dat*Delta_Ang;
  // Yaw Angle
  Dat:=HexToDec(Buffer1[ag+4]+Buffer1[ag+5]);
  if Dat >32767 then
    Dat:=Dat-65536;
  if FIPD.Showing then
    begin
      FIPD.EdYaw.Clear;FIPD.EdYaw.SelText:=FloatToStrF(Dat*Delta_Ang,ffFixed,5,3);
    end;
  Heading:=Trunc(Dat*Delta_Ang);
  Horizont.CmdYaw:=Heading;
  Compass.Course1:=Heading; // Compass.Bearing:=Dat*Delta_Ang;
  Compass.Compass:=Heading;
  NLCompass.Value:=Heading;
end;

Procedure TGPSMain.Rates;
var
  Dat: Integer;
begin
  // XRate
  Dat:=HexToDec(Buffer1[rt]+Buffer1[rt+1]);

```

```

if Dat >32767 then
  Dat:=Dat-65536;
if FIPD.Showing then
begin
  FIPD.EdXRate.Clear;FIPD.EdXRate.SelText:=FloatToStrF(Dat*Delta_Rat,ffFixed,5,3);
end;
// YRate
Dat:=HexToDec(Buffer1[rt+2]+Buffer1[rt+3]);
if Dat >32767 then
  Dat:=Dat-65536;
if FIPD.Showing then
begin
  FIPD.EdYRate.Clear;FIPD.EdYRate.SelText:=FloatToStrF(Dat*Delta_Rat,ffFixed,5,3);
end;
// Zrate
Dat:=HexToDec(Buffer1[rt+4]+Buffer1[rt+5]);
if Dat >32767 then
  Dat:=Dat-65536;
if FIPD.Showing then
begin
  FIPD.EdZRate.Clear;FIPD.EdZRate.SelText:=FloatToStrF(Dat*Delta_Rat,ffFixed,5,3);
end;
end;

```

```

Procedure TGPSMain.Accelerations;

```

```

var

```

```

  Dat: Integer;

```

```

begin

```

```

  // XRate

```

```

  Dat:=HexToDec(Buffer1[ac]+Buffer1[ac+1]);

```

```

  if Dat >32767 then

```

```

    Dat:=Dat-65536;

```

```

  if FIPD.Showing then

```

```

  begin

```

```

    FIPD.EdXAcc.Clear;FIPD.EdXAcc.SelText:=FloatToStrF(Dat*Delta_Acc,ffFixed,5,3);

```

```

  end;

```

```

  // YRate

```

```

  Dat:=HexToDec(Buffer1[ac+2]+Buffer1[ac+3]);

```

```

  if Dat >32767 then

```

```

    Dat:=Dat-65536;

```

```

  if FIPD.Showing then

```

```

  begin

```

```

    FIPD.EdYAcc.Clear;FIPD.EdYAcc.SelText:=FloatToStrF(Dat*Delta_Acc,ffFixed,5,3);

```

```

  end;

```

```

  // Zrate

```

```

  Dat:=HexToDec(Buffer1[ac+4]+Buffer1[ac+5]);

```

```

  if Dat >32767 then

```

```

    Dat:=Dat-65536;

```

```

  if FIPD.Showing then

```

```

  begin

```

```

    FIPD.EdZAcc.Clear;FIPD.EdZAcc.SelText:=FloatToStrF(Dat*Delta_Acc,ffFixed,5,3);

```

```

  end;

```

```

end;

```

```

Procedure TGPSMain.Velocity;

var
  Dat: Integer;

Begin
  // XRate
  Dat:=HexToDec( Buffer1 [ ve]+Buffer1 [ ve+1]);
  if Dat >32767 then
    Dat:=Dat-65536;
  if FGPSData.Showing then
  begin
    FGPSData.EdNorth.Clear;
    FGPSData.EdNorth.SelText:=FloatToStrF(Dat*Delta_Vel, ffFixed, 5, 3);
  end;
  // YRate
  Dat:=HexToDec( Buffer1 [ ve+2]+Buffer1 [ ve+3]);
  if Dat >32767 then
    Dat:=Dat-65536;
  if FGPSData.Showing then
  begin
    FGPSData.EdEast.Clear;
    FGPSData.EdEast.SelText:=FloatToStrF(Dat*Delta_Vel, ffFixed, 5, 3);
  end;
  // Zrate
  Dat:=HexToDec( Buffer1 [ ve+4]+Buffer1 [ ve+5]);
  if Dat >32767 then
    Dat:=Dat-65536;
  if FGPSData.Showing then
  begin
    FGPSData.EdDown.Clear;
    FGPSData.EdDown.SelText:=FloatToStrF(Dat*Delta_Vel, ffFixed, 5, 3);
  end;
  VSpeed.Speed:=Round((Dat*Delta_Vel)*1000);
  LVSpeed.Caption:= FloatToStrF((Dat*Delta_Vel), ffFixed, 5, 1);
end;

Procedure TGPSMain.Longitude;

var
  Long: Extended;
  Dat: Int64;
begin
  // XRate
  Dat:=Hex32ToDec( Buffer1 [ lo]+Buffer1 [ lo+1]+Buffer1 [ lo+2]+Buffer1 [ lo+3]);
  Long:=Dat*Delta_Loa;
  HLong:=Long;
  if Long > 180.0 then
    Long:=Long-360;
  if FGPSData.Showing then
  begin
    FGPSData.EdLongitude.Clear;
    FGPSData.EdLongitude.SelText:=FloatToStrF(Long, ffFixed, 13, 9);
  end;
end;

```

```

end;
If GPSMonFlag then
begin
  GELong:=Long;
  StatBar.Panels[6].Text:='Est. Long: '+FloatToStrF(Long, ffFixed,13,9);
end;
end;

```

Procedure TGPSMain.Latitude;

```

var
  Dat: Int64;
  Lat: Extended;
Begin
  Dat:=Hex32ToDec(Buffer1[1a]+Buffer1[1a+1]+Buffer1[1a+2]+Buffer1[1a+3]);
  Lat:=Dat*Delta_Loa;
  HLat:=Lat;
  if FGPSData.Showing then
  begin
    FGPSData.EdLatitude.Clear;FGPSData.
    EdLatitude.SelText:=FloatToStrF(Lat, ffFixed,13,9);
  end;
  If GPSMonFlag then
  begin
    GELat:=LAt;
    StatBar.Panels[5].Text:='Est. Lat: '+FloatToStrF(Lat, ffFixed,13,9);
  end;
end;

```

Procedure TGPSMain.GPSAltitude;

```

var
  Dat: Integer;
  Altitude: Real;
Begin
  Dat:=HexToDec(Buffer1[al]+Buffer1[al+1]);
  Altitude:=Dat*Delta_Alt;
  HAlt:=Altitude;
  if Dat >32767 then
    Dat:=Dat-65536;
  if FGPSData.Showing then
  begin
    FGPSData.EdAltitude.Clear;
    FGPSData.EdAltitude.SelText:=FloatToStrF(Altitude, ffFixed,5,1);
  end;
  NLAltitude.Value:=Dat*Delta_Alt;
  Altimeter.Altitude:=Round(Dat*Delta_Alt-GRLevel);
  If GPSMonFlag then
  begin
    GEAlt:=Altitude;
    StatBar.Panels[7].Text:='Est. Alt: '+FloatToStrF(Altitude, ffFixed,5,1);
    GPSMonFlag:=False;
  end;
  If GRLFlag then

```

```

begin
  GRLevel:=Altitude;
  StatBar.Panels[9].Text:='Helicopter GRL: '+FloatToStrF(GRLevel,ffFixed,5,1);
  GRLFlag:=False;
end
end;

Procedure TGPSMain.RPMComp;

Var
  Dat: Integer;

Begin
  Dat:=HexToDec(Buffer1[Prpm]+Buffer1[Prpm+1]);
  RPM.Value:=Dat/1000;
  NLRPM.Value:=Dat;
end;

Procedure TGPSMain.Range;

var
  Dat: Integer;

Begin
  Dat:=HexToDec(Buffer1[PRange]+Buffer1[PRange+1]);
  LRange.Caption:=FloatToStrF(Dat/10,ffFixed,5,1);
  PBRange.Value:=Dat/10;
end;

// Avionics battery voltage monitor routine
Procedure TGPSMain.AVoltage;
var
  Dat: Integer;
  Volt: Real;
begin
  Dat:=HexToDec(Buffer1[PAvV]+Buffer1[PAvV+1]);
  Volt:=Dat*DeltaAvV;
  AvVoltage.Value:=Volt;
  DAvVoltage.Value:=Volt;
end;

// Main rotor battery voltage monitor routine
Procedure TGPSMain.RVVoltage;
var
  Dat: Integer;
  Volt: Real;
begin
  Dat:=HexToDec(Buffer1[PRV]+Buffer1[PRV+1]);
  Volt:=Dat*DeltaRV;
  RotVoltage.Value:=Volt;
  DRVoltage.Value:=Volt;
end;

// Servos battery voltage monitor routine

```

```

Procedure TGPSMain.SVoltage;
var
    Dat: Integer;
    Volt: Real;
begin
    Dat:=HexToDec( Buffer1 [PSV]+Buffer1 [PSV+1]);
    Volt:=Dat*DeltaSV;
    SerVoltage.Value:=Volt;
    DSVoltage.Value:=Volt;
end;

// Avionics computer battery current monitor routine
Procedure TGPSMain.ACurrent;
var
    Dat: Integer;
    Curr: Real;
begin
    Dat:=HexToDec( Buffer1 [PAC]+Buffer1 [PAC+1]);
    Curr:=2*(Dat*DeltaAC-2.5);
    AvCurr.Value:=Curr;
    DAvCurr.Value:=Curr;

    // Computing the integration

    AQt:=Hex32ToDec( Buffer1 [PAQd3]+Buffer1 [PAQd3+1]+Buffer1 [PAQd3+2]+Buffer1 [PAQd3+3])/10000;
    AQR:=(AQt-AQd)*100/AQt;           // Battery remainder charge in percent

    if AQR >0 then
        AvCap.Value:=Aqr;

    if (AvCap.Value < 40) AND (AvCap.Value > 25) then
        AvCap.ColorMin:=clYellow;
    if AvCap.Value < 25 then
        AvCap.ColorMin:=clRed;
end;

// Main rotor battery current monitor routine
Procedure TGPSMain.RCurrent;
var
    Dat: Integer;
    Curr: Real;
begin
    Dat:=HexToDec( Buffer1 [PRC]+Buffer1 [PRC+1]);
    Curr:=80*(Dat*DeltaRC-2.5);
    RotCurr.Value:=abs(Curr);
    DRCurr.Value:=abs(Curr);

    // Computing the integration

    RQd:=Hex32ToDec( Buffer1 [PRQd3]+Buffer1 [PRQd3+1]+Buffer1 [PRQd3+2]+Buffer1 [PRQd3+3])/10000;
    RQR:=(RQt-RQd)*100/RQt;           // Battery remainder charge in percent

    if RQR > 0 then
        RotCap.Value:=RQR;
    if (RotCap.Value < 40) AND (RotCap.Value > 25) then

```

```

    RotCap.ColorMin:=clYellow;
  if RotCap.Value < 25 then
    RotCap.ColorMin:=clRed;
end;

// Servos battery current monitor routine
Procedure TGPSMain.SCurrent;
var
  Dat: Integer;
  Curr: Real;
begin
  Dat:=HexToDec( Buffer1 [PSC]+Buffer1 [PSC+1]);
  Curr:=2*(Dat*DeltaSC - 2.5);
  SerCurr.Value:=Curr;
  DSCurr.Value:=Curr;

  // Computing the integration

  SQd:=Hex32ToDec( Buffer1 [PSQd3]+Buffer1 [PSQd3+1]+Buffer1 [PSQd3+2]+Buffer1 [PSQd3+3])/10000;
  SQR:=(SQt-SQd)*100/SQT;

  if SQR >0 then
    SerCap.Value:=SQR;
  if (SerCap.Value < 40) AND (SerCap.Value > 25) then
    SerCap.ColorMin:=clYellow;
  if SerCap.Value < 25 then
    SerCap.ColorMin:=clRed;
end;

// This routine transmit an ACK and commands immediately after all data
// have been received
procedure TGPSMain.Tx_Ack_Comm;
var
  TxData:string;
  ID,PL: Char;
begin
  Puerto.FlushTX;
  ID:=#06;
  PL:=#00;
  TxData:='U'+ID+PL+#10+#13;
  puerto.Envia(Pchar(TxData),length(TxData));
end;

//This routine transforms the latitud, longitude and Altitud coordinates
//into cartesian coordinates based on the ellipsoid WGS84
Procedure TGPSMain.Lla2ECEF(Long, Lat, Alt: Extended; var x,y,z: Extended);
const
  // WGS84 ellipsoid constants:
  a = 6378137;
  e = 8.1819190842622e-2;
var
  N: Extended;
begin
  N:=a/sqrt(1 - Power(e,2) * Power(sin(Lat),2));

```

```

x:=(N+Alt)*cos(Lat)*cos(Long);
y:=(N+Alt)*cos(Lat)*sin(Long);
z:=((1-Power(e,2))*N + Alt)*sin(Lat);

end;

// Computes the distance from Helicopter to Ground Station
Procedure TGPSMain.Distance;
var
  R,x1,y1,z1,x2,y2,z2:Extended;
  GELatr,GELongr:Extended;
begin
  GELatr:=GELat*pi/180;
  GELongr:=GELong*pi/180;
  HLat:=HLat*pi/180;
  HLong:=HLong*pi/180;
  // Transforming from Long, lat to ECEF coordinates
  Lla2ECEF(GELongr,GELatr,GEAlt,x1,y1,z1);
  Lla2ECEF(HLong,HLat,HAlt,x2,y2,z2);
  // Computing the distance between them
  R:=sqrt(Power((x1-x2),2)+Power((y1-y2),2)+Power((z1-z2),2));
  StatBar.Panels[11].Text:='R: '+FloatToStrF(R,ffNumber,18,2)+' m';
end;

// This routine checks the Helicopter recording status byte
Procedure TGPSMain.HRecStatus;
begin
  Edit1.Clear;Edit1.SelText:=IntToStr(Ord(Buffer1[PRECS]));
  if DatAnt <> Ord(Buffer1[PRECS]) then
  begin
    DatAnt:=Ord(Buffer1[PRECS]);
    if DatAnt = 1 then
    begin
      FHRec:=True;
      StatBar.Panels[10].Text:='Rec.';
    end
    else
    begin
      FHRec:=False;
      StatBar.Panels[10].Text:='Not Rec.';
    end;
  end;
end;

Procedure TGPSMain.Recepcion_Datos;
begin
  Angles;
  Rates;
  Accelerations;
  Velocity;
  Longitude;
  Latitude;

```

```

GPSAltitude;
RPMComp;
Range;
AVoltage;
RVoltage;
SVoltage;
ACurrent;
RCurrent;
SCurrent;
Distance;
HRecStatus;
// Battery data base
if VCFlag then
  if Now > CTime then
    begin
      TVCMonitor.First;
      TVCMonitor.Insert;
      TVCMonitor.FieldName('Time').AsDateTime:=Now;
      TVCMonitor.FieldName('Avionic V').AsFloat:=RoundTo(DAvVoltage.Value,-2);
      TVCMonitor.FieldName('Avionic I').AsFloat:=RoundTo(DAvCurr.Value,-3);
      TVCMonitor.FieldName('Rotor V').AsFloat:=RoundTo(DRVoltage.Value,-2);
      TVCMonitor.FieldName('Rotor I').AsFloat:=RoundTo(DRCurr.Value,-3);
      TVCMonitor.FieldName('Servo V').AsFloat:=RoundTo(DSVoltage.Value,-2);
      TVCMonitor.FieldName('Servo I').AsFloat:=RoundTo(DSCurr.Value,-3);
      TVCMonitor.Post;
      CTime:=IncSecond(CTime,2);
      StatBar.Panels[8].Text:='VCDB Rec: '+IntToStr(TVCMonitor.RecordCount);
      if FVCMonitorDB.Showing then
        begin
          FVCMonitorDB.EdRecords.Clear;
          FVCMonitorDB.EdRecords.SelText:=IntToStr(TVCMonitor.RecordCount);
        end;
      end;
    end;
  // End battery data base
end;

Procedure TGPSMain.Recibe;
begin
  Buffer:=puerto.Recibir;
  Buffer1:=Buffer1+Buffer;
  Tam:=length(Buffer1);
  if (Tam > 2) and (Not Sync) then
    begin
      Point:=1;
      repeat
        if (Buffer1[Point]='U') and (Buffer1[Point+1]='U') then
          begin
            Buffer1:=copy(Buffer1,Point,Tam);
            Sync:=True;
          end
        else
          Inc(Point);
      until (Sync) or (Point + 1 > Tam)
    end;
  // End if for get synchronization

```

```

if Sync then
begin
  if (length(Buffer1)>=SP) and (Buffer1[SP-1]=#10) and (Buffer1[SP]=#13) then
  begin
    ReqTx.Enabled:=False;
    Cont:=Cont+1;
    StatBar.Panels[4].Text:='Received Frames: '+IntToStr(Cont);
    Recepcion_Datos;
    SetLength(Buffer,0);
    SetLength(Buffer1,0);
    Puerto.FlushRX;
    Sync:=False;
    case CommandF of // Check for pending commands
      1:StartRecordingData;
      2:StopRecordingData;
    else // Else of case
    begin
      if TxFlag then
      begin
        Tx_Ack_Comm;
        ReqTx.Enabled:=True;
      end
      else
        ReqTx.Enabled:=False;
    end; // End of else of the case
    end; // End of Case
  end; // End of if Sync
end;
end;

procedure TGPSMain.RxTimerTimer(Sender: TObject);
begin
  if puerto.estaAbierto then
    Recibe;
end;

procedure TGPSMain.Exit1Click(Sender: TObject);
begin
  TVCMonitor.Close;
  Close;
end;

procedure TGPSMain.Close1Click(Sender: TObject);
begin
  puerto.CerrarPuerto;
  puerto.Destroy;
  ModificarEstadoControles(true);
  ClearAll;
  Sync:=False;
  RxTimer.Enabled:=False;
  Commands1.Enabled:=False;
  ReqTx.Enabled:=False;
end;

```

```

procedure TGPSMain.Open1Click(Sender: TObject);
var
  ultimoError: DWord;
  mensajeError: string;
begin
  Puerto:= TPuertoSerie.Create;
  puerto.puerto:=Tpuestos( FPortSettings.ComPuertos.ItemIndex);
  puerto.bauds := strToInt( FPortSettings.ComVelocidad.Text);
  puerto.paridad := FPortSettings.ComParidad.ItemIndex;
  puerto.stopBits:=0;
  puerto.bitsDatos:=8;
  if not puerto.estaAbierto then
  begin
    RxTimer.Enabled:=True;
    ClearAll;
    RPM.Value:=0;
    Horizont.Pitch:=0;Horizont.Roll:=0;
    Compass.Bearing:=17;Compass.Course1:=0;Compass.Compass:=0;
    Altimeter.Altitude:=0;
    Sync:=False;
    SetLength(Buffer1,0);
    SetLength(Buffer1,0);
    Puerto.FlushRX;
    Puerto.FlushTX;
    Commands1.Enabled:=True;
    StartHelicopterTx1.Enabled:=True;
    StopHelicopterTx1.Enabled:=False;

    if not puerto.AbrirPuerto(puerto.puerto) then
    begin
      ultimoError := puerto.ultimoError;
      mensajeError := 'Se produjo el error no. ' + IntToStr( ultimoError) +
        ' al intentar abrir el puerto.';
      Application.MessageBox( PChar(mensajeError), 'Error, no se puede abrir el puerto',
        MB_ICONWARNING or MB_OK);
    end
    else
    begin
      modificarEstadoControles (false);
    end;
  end;
end;

procedure TGPSMain.Settings1Click(Sender: TObject);
begin
  FPortSettings.Show;
end;

procedure TGPSMain.InertialData1Click(Sender: TObject);
begin
  FIPD.Show;
end;

procedure TGPSMain.GPSData1Click(Sender: TObject);

```

```

begin
    FGPSData.Show;
end;

procedure TGPSMain.SetGroundLevel1Click(Sender: TObject);
var
    Answer: Integer;
begin
    if Application.MessageBox('Do you want to change the Ground Reference Level?',
        'Warning',MB.YesNo)= mrYes then
        begin
            GRLFlag:=True;
        end;
end;

procedure TGPSMain.SetGPSEstationPosition1Click(Sender: TObject);
var
    Answer: Integer;
begin
    if Application.MessageBox('Do you want to set GPS Monitor Ground Station position?',
        'Warning',MB.YesNo)= mrYes then
        begin
            GPSMonFlag:=True;
        end;
end;

procedure TGPSMain.SetBatteriesCapacities1Click(Sender: TObject);
begin
    FBatCap.Show;
end;

procedure TGPSMain.ShowDataBase1Click(Sender: TObject);
begin
    FVCMonitorDB.Show;
end;

procedure TGPSMain.StopRecording1Click(Sender: TObject);
begin
    VCFlag:=False;
    StatBar.Panels[8].Text:='VCDB:No Rec.';
end;

procedure TGPSMain.StartRecording1Click(Sender: TObject);
begin
    CTime:=Now;
    VCFlag:=True;
    StatBar.Panels[8].Text:='VCDB Rec: '+ IntToStr(TVCMonitor.RecordCount);
end;

procedure TGPSMain.EraseDataBase1Click(Sender: TObject);
begin
    if Not TVCMonitor.IsEmpty then
        begin
            if TVCMonitor.Exists then

```

```

begin
    TVCMonitor.Close;
    TVCMonitor.EmptyTable;
    TVCmonitor.Open;
    FVCMonitorDB.EdRecords.Clear;
    Application.MessageBox( 'Data base Errased.',
                            'Voltage and Current Data Base', MB.OK);
end;
end
else
    Application.MessageBox( 'Data base is already Empty.',
                            'Voltage and Current Data Base', MB.OK);
end;

// This procedure is calle by some ruotines like StartHelicopterTx1Click,
// HelicopterStartRecording1Click and HelicopterStopRecording1Click
// and is done because StartHelicopterTx1Click cannot be called by
// HelicopterStartRecording1Click or HelicopterStopRecording1Click
Procedure TGPSMain.Dummy;
begin
    Tx_Ack_Comm;
    StopHelicopterTx1.Enabled:=True;
    ReqTx.Enabled:=True;
    TxFlag:=True;
    StartHelicopterTx1.Enabled:=False;
end;

procedure TGPSMain.StartHelicopterTx1Click(Sender: TObject);
begin
    Dummy;
end;

procedure TGPSMain.StopHelicopterTx1Click(Sender: TObject);
begin
    StartHelicopterTx1.Enabled:=True;
    ReqTx.Enabled:=False;
    TxFlag:=False;
    StopHelicopterTx1.Enabled:=False;
    Puerto.FlushRX;
end;

procedure TGPSMain.ReqTxTimer(Sender: TObject);
begin
    Tx_Ack_Comm;
end;

procedure TGPSMain.StartRecordingData;
var
    TxData: String;
begin
    If Puerto.EstaAbierto then
        begin
            Puerto.FlushTX;

```

```

    TxData:= 'U'+ 'R'+ #00+ #10+ #13;
    puerto. Enviar (Pchar (TxData), length (TxData));
    TDelay. Interval:=75;           // 75 ms of Delay then active TX calling Dummy
    TDelay. Enabled:=True;
    CommandF:=0;
end;
end;

procedure TGPSMain. HelicopterStartRecording1Click (Sender: TObject);
begin
    CommandF:=1;
end;

procedure TGPSMain. StopRecordingData;
var
    TxData: String;
begin
    if puerto. EstaAbierto then
    begin
        Puerto. FlushTX;
        TxData:= 'U'+ 'S'+ #00+ #10+ #13;
        puerto. Enviar (Pchar (TxData), length (TxData));
        TDelay. Interval:=75; // 75 ms of Delay then active TX calling Dummy
        TDelay. Enabled:=True;
        CommandF:=0;
    end;
end;

procedure TGPSMain. HelicopterStopRecording1Click (Sender: TObject);
begin
    CommandF:=2;
end;

procedure TGPSMain. StatBarDrawPanel (StatusBar: TStatusBar;
    Panel: TStatusPanel; const Rect: TRect);
begin
    with StatBar. Canvas Do
    begin
        case Panel. Index of
            0: begin
                Brush. Color:=RGB(254,219,137);
                Font. Color:=clBlack;
                // Font. Style:=[fsBold];
            end;
            1: begin
                Brush. Color:=RGB(254,219,137);
                Font. Color:=clBlack;
                // Font. Style:=[fsItalic];
            end;
            2: begin
                Brush. Color:=RGB(254,219,137);
                Font. Color:=clBlack;
                // Font. Style:=[fsItalic];
            end;
        end;
    end;
end;

```

```
end;
3:begin
  if Not FCom then
  begin
    Brush . Color:=RGB(254,219,137);
    Font . Color:=clBlack;
  end
  else
  begin
    Brush . Color:=ClGreen;
    Font . Color:=clWhite;
    Font . Style:=[fsBold];
  end;
end;
4:begin
  Brush . Color:=RGB(254,219,137);
  Font . Color:=clBlack;
  // Font . Style:=[fsItalic];
end;
5:begin
  Brush . Color:=ClMoneyGreen;
  Font . Color:=clBlack;
  // Font . Style:=[fsItalic];
end;
6:begin
  Brush . Color:=ClMoneyGreen;
  Font . Color:=clBlack;
  // Font . Style:=[fsItalic];
end;
7:begin
  Brush . Color:=ClMoneyGreen;
  Font . Color:=clBlack;
  // Font . Style:=[fsItalic];
end;
8:begin
  if Not VCFlag then
  begin
    Brush . Color:=ClMoneyGreen;
    Font . Color:=clBlack;
  end
  else
  begin
    Brush . Color:=ClRed;
    Font . Color:=clWhite;
    Font . Style:=[fsBold];
  end
end;
9:begin
  Brush . Color:=RGB(251,189,199);
  Font . Color:=clBlack;
  // Font . Style:=[fsItalic];
end;
10:begin
  If Not FHRec then
  begin
```

```

        Brush.Color:=RGB(251,189,199);
        Font.Color:=clBlack;
    end
    else
    begin
        Brush.Color:=ClRed;
        Font.Color:=clWhite;
        Font.Style:=[fsBold];
    end;
end;
11:begin
    Brush.Color:=RGB(251,189,199);
    Font.Color:=clBlack;
    // Font.Style:=[fsItalic];
end;
end;
FillRect(Rect);
TextRect(Rect,2+Rect.Left,Rect.Top,Panel.Text);
end;
end;

procedure TGPSMain.TDelayTimer(Sender: TObject);
begin
    if puerto.EstaAbierto then
        Dummy;
        TDelay.Enabled:=False;
end;

procedure TGPSMain.Save2Text;
var
    Data:String;
    k,n:Integer;
begin
    TVCMonitor.First;
    n:=TVCMonitor.RecordCount;
    if SaveD.Execute then
    begin
        FileName:=SaveD.FileName;
        AssignFile(FName,FileName);
        Rewrite(FName);
        for k:=1 to n do
        begin
            Data:=FloatToStr(TimeStampToMsecs(DateTimeToTimeStamp(
                TVCMonitor.FieldByName('Time').AsDateTime))) + #09 +
            FloatToStr(TVCMonitor.FieldByName('Avionic V').AsFloat) + #09 +
            FloatToStr(TVCMonitor.FieldByName('Avionic I').AsFloat) + #09 +
            FloatToStr(TVCMonitor.FieldByName('Rotor V').AsFloat) + #09 +
            FloatToStr(TVCMonitor.FieldByName('Rotor I').AsFloat) + #09 +
            FloatToStr(TVCMonitor.FieldByName('Servo V').AsFloat) + #09 +
            FloatToStr(TVCMonitor.FieldByName('Servo I').AsFloat);
            writeln(Fname,Data);
            TVCMonitor.Next;
        end;
    end;
end;

```

```

        end;
        CloseFile (FName);
    end;
end;

procedure TGPSMain.ExporttoTextFiledat1Click(Sender: TObject);
begin
    Save2Text;
end;

end.

*****
*                                     Unit Port Settings
*
*****
unit UPortSettings;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls, ComSerie;

type
    TFPortSettings = class(TForm)
        BDone: TButton;
        BCancel: TButton;
        StaticText1: TStaticText;
        StaticText2: TStaticText;
        StaticText3: TStaticText;
        ComPuertos: TComboBox;
        ComVelocidad: TComboBox;
        ComParidad: TComboBox;
        procedure BCancelClick(Sender: TObject);
        procedure BDoneClick(Sender: TObject);
        procedure ComPuertosChange(Sender: TObject);
        procedure ComVelocidadChange(Sender: TObject);
        procedure ComParidadChange(Sender: TObject);
        procedure FormCreate(Sender: TObject);
    private
        { Private declarations }
    public
        { Public declarations }
    end;

var
    FPortSettings: TFPortSettings;
    Puerto: TPuertoSerie;

implementation

```

```

uses GPS;

{$R *.dfm}

procedure TFPortSettings.BCancelClick(Sender: TObject);
begin
    Close;
end;

procedure TFPortSettings.BDoneClick(Sender: TObject);
begin
    Close;
end;

procedure TFPortSettings.ComPuertosChange(Sender: TObject);
begin
    GPSMain.StatBar.Panels[0].Text:= 'Port: '+ComPuertos.Text;
end;

procedure TFPortSettings.ComVelocidadChange(Sender: TObject);
begin
    GPSMain.StatBar.Panels[1].Text:= 'Speed: '+ComVelocidad.Text;
end;

procedure TFPortSettings.ComParidadChange(Sender: TObject);
begin
    GPSMain.StatBar.Panels[2].Text:= 'Parity: '+ComParidad.Text;
end;

procedure TFPortSettings.FormCreate(Sender: TObject);
begin
    GPSMain.StatBar.Panels[0].Text:= 'Port: ' + FPortSettings.ComPuertos.Text;
    GPSMain.StatBar.Panels[1].Text:= 'Speed: ' + FPortSettings.ComVelocidad.Text;
    GPSMain.StatBar.Panels[2].Text:= 'Parity: ' + FPortSettings.ComParidad.Text;
    GPSMain.StatBar.Panels[3].Text:= 'Status: Close';
end;

end.

*****
*                                     Unit UIPD
*
*****
unit UIPD;

interface

uses
    Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
    Dialogs, StdCtrls;

type
    TFIPD = class(TForm)

```

```

    GB_Angles: TGroupBox;
    Label7: TLabel;
    Label6: TLabel;
    Label5: TLabel;
    EdRoll: TEdit;
    EdPitch: TEdit;
    EdYaw: TEdit;
    StaticText2: TStaticText;
    StaticText3: TStaticText;
    StaticText4: TStaticText;
    GB_Rates: TGroupBox;
    Label10: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    EdXRate: TEdit;
    EdYRate: TEdit;
    EdZRate: TEdit;
    StaticText5: TStaticText;
    StaticText6: TStaticText;
    StaticText7: TStaticText;
    GB_Accelerations: TGroupBox;
    Label11: TLabel;
    Label12: TLabel;
    Label13: TLabel;
    EdXAcc: TEdit;
    EdYAcc: TEdit;
    EdZAcc: TEdit;
    StaticText8: TStaticText;
    StaticText9: TStaticText;
    StaticText10: TStaticText;
    BClose: TButton;
    procedure BCloseClick(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;

var
    FIPD: TFIPD;

implementation

{$R *.dfm}

procedure TFIPD.BCloseClick(Sender: TObject);
begin
    Close;
end;

end.

```

```

*****
*                                     Unit GPS Data

```

```

*
*****
unit UGPSData;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls;

type
  TFGPSData = class(TForm)
    GB_Position: TGroupBox;
    Altitude: TLabel;
    Lat: TLabel;
    Long: TLabel;
    EdAltitude: TEdit;
    EdLatitude: TEdit;
    EdLongitude: TEdit;
    StaticText11: TStaticText;
    StaticText12: TStaticText;
    StaticText13: TStaticText;
    GB_Velocity: TGroupBox;
    North: TLabel;
    East: TLabel;
    Down: TLabel;
    EdNorth: TEdit;
    EdEast: TEdit;
    EdDown: TEdit;
    StaticText14: TStaticText;
    StaticText15: TStaticText;
    StaticText16: TStaticText;
    BClose: TButton;
    procedure BCloseClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  FGPSData: TFGPSData;

implementation

{$R *.dfm}

procedure TFGPSData.BCloseClick(Sender: TObject);
begin
  Close;
end;

end.

```

```

*****
*                                     Unit Battery Capacity
*
*****
unit UBatCap;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls;

type
  TFBatCap = class(TForm)
    AvBc: TLabeledEdit;
    RotBc: TLabeledEdit;
    SerBc: TLabeledEdit;
    StaticText1: TStaticText;
    StaticText2: TStaticText;
    StaticText3: TStaticText;
    BDone: TButton;
    BCancel: TButton;
    procedure BDoneClick(Sender: TObject);
    procedure AvBcKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure RotBcKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure SerBcKeyUp(Sender: TObject; var Key: Word;
      Shift: TShiftState);
    procedure FormShow(Sender: TObject);
    procedure BCancelClick(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  FBatCap: TFBatCap;
  KeyValue: Integer;
implementation

uses
  GPS;

{$R *.dfm}

procedure TFBatCap.BDoneClick(Sender: TObject);
begin
  Gps.AvBCap:=StrToInt(AvBc.text);GPSMain.AvBc.Caption:='Bc='+AvBc.Text+ ' mAh';
  Gps.RotBCap:=StrToInt(RotBc.text);GPSMain.RotBc.Caption:='Bc='+RotBc.Text+ ' mAh';
  Gps.SerBCap:=StrToInt(SerBc.text);GPSMain.SerBc.Caption:='Bc='+SerBc.Text+ ' mAh';
  GPSMain.AVCap.Value:=100.0;GPSMain.RotCap.Value:=100.0;GPSMain.SerCap.Value:=100.0;

```

```

GPS.AQd:=0.0;GPS.RQd:=0.0;GPS.SQd:=0.0;
GPS.iA1:=0;GPS.iA1:=0;GPS.iR1:=0;GPS.iR1:=0;GPS.iS1:=0;GPS.iS1:=0;
GPS.AQt:=StrToInt(AvBc.Text)*3.6; GPS.RQt:=StrToInt(RotBc.Text)*3.6;
GPS.SQt:=StrToInt(SerBc.Text)*3.6;
GPSMain.AvCap.ColorMin:=clLime;
GPSMain.RotCap.ColorMin:=clLime;
GPSMain.SerCap.ColorMin:=clLime;

Close;
end;

procedure TFBatCap.AvBcKeyUp(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if Key = VK_RETURN then
  begin
    try
      StrToInt(AvBc.Text);
      RotBc.SetFocus;
    except;
      KeyValue:=Application.messageBox('Battery Capacity must be an Integer','Error',
        MB_ICONWARNING or MB_OK);

      AvBc.Clear;AvBc.SetFocus;
    end;
  end;
end;

procedure TFBatCap.RotBcKeyUp(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if (Key = VK_RETURN) or (Key = VK_TAB) then
  begin
    try
      StrToInt(RotBc.Text);
      SerBc.SetFocus;
    except;
      Application.messageBox('Battery Capacity must be an Integer','Error',
        MB_ICONWARNING or MB_OK);

      RotBc.Clear;RotBc.SetFocus;
    end;
  end;
end;

procedure TFBatCap.SerBcKeyUp(Sender: TObject; var Key: Word;
  Shift: TShiftState);
begin
  if (Key = VK_RETURN) or (Key = VK_TAB) then
  begin
    try
      StrToInt(SerBc.Text);
      BDone.Enabled:=True;
      BDone.Default:=True;
    except;
      Application.messageBox('Battery Capacity must be an Integer','Error',

```

```

                                MB.ICONWARNING or MB.OK);
    SerBc.Clear; SerBc.SetFocus;
    BDone.Enabled:=False;
    BDone.Default:=False;
  end;
end;
end;

procedure TFBatCap.FormShow(Sender: TObject);
begin
  AvBc.Clear;
  RotBc.Clear;
  SerBc.Clear;
  BDone.Enabled:=False;
  BDone.Default:=False;
  AvBc.SetFocus;
end;

procedure TFBatCap.BCancelClick(Sender: TObject);
begin
  Close;
end;

end.

*****
*                               Unit Voltage Current Monitor DB
*
*****
unit UVCMonitorDB;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Grids, DBGrids, ExtCtrls, DBCtrls, StdCtrls;

type
  TFVCMonitorDB = class(TForm)
    DBGVC: TDBGrid;
    DBNVCMoitor: TDBNavigator;
    BClose: TButton;
    EdRecords: TEdit;
    StaticText1: TStaticText;
    procedure FormShow(Sender: TObject);
    procedure BCloseClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  FVCMonitorDB: TFVCMonitorDB;

```

```
implementation
uses
    GPS;
{$R *.dfm}

procedure TFVCMonitorDB.FormShow(Sender: TObject);
begin
    EdRecords.SelText:=IntToStr(GPSMain.TVCMonitor.RecordCount);
    BClose.SetFocus;
end;

procedure TFVCMonitorDB.BCloseClick(Sender: TObject);
begin
    Close;
end;

end.
```

C.6 Mechanical Data

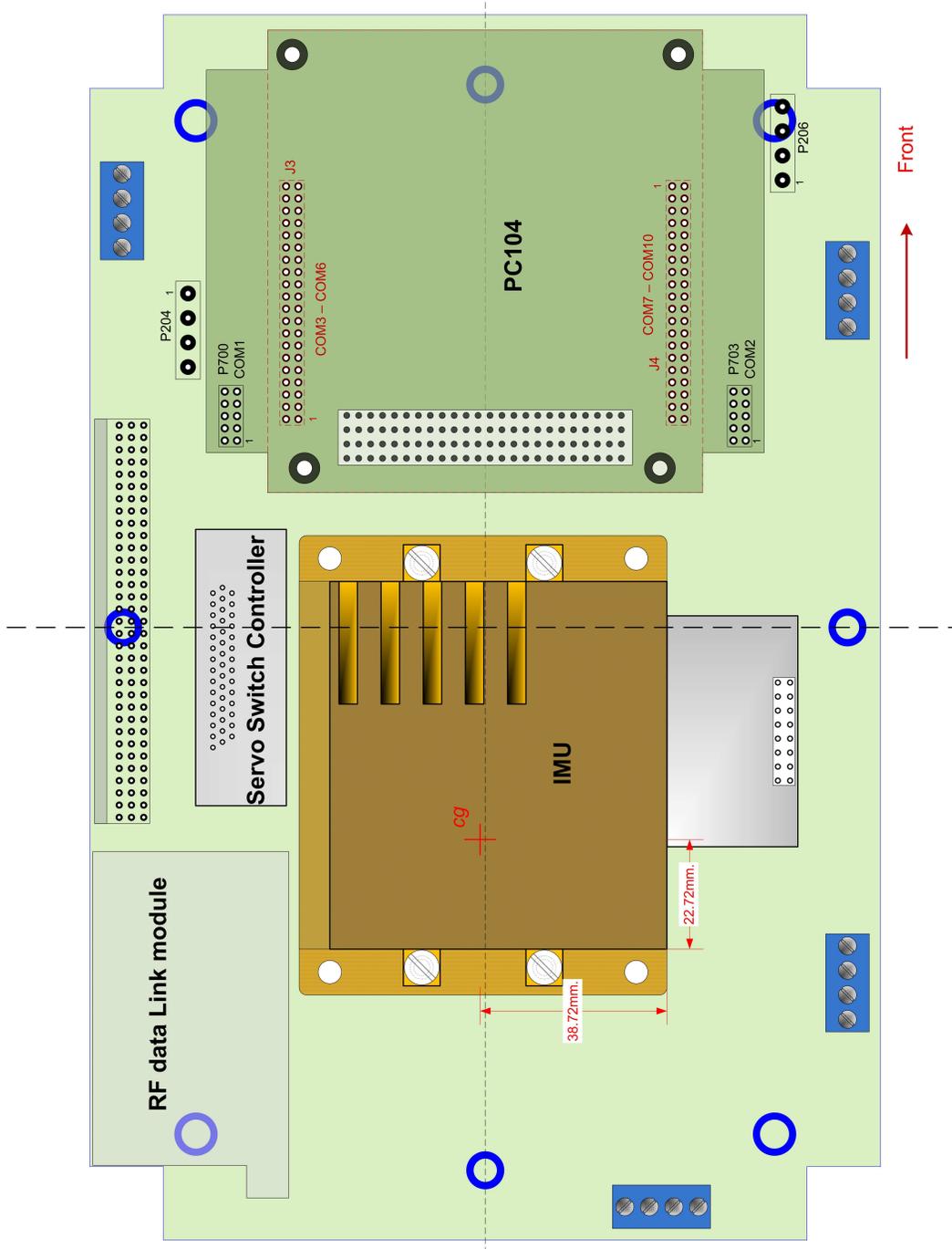


Figure C.22: ECICH General Layout.

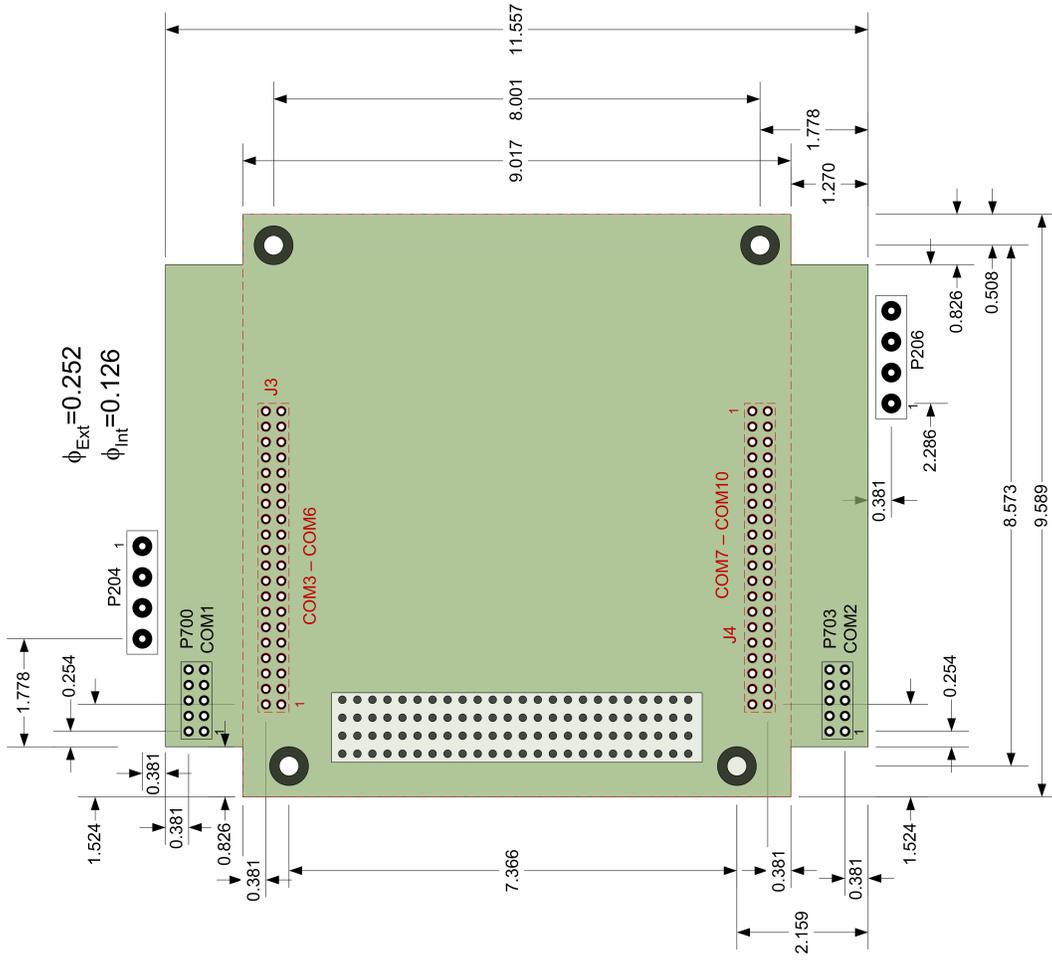


Figure C.23: PC/104™ Dimensions.

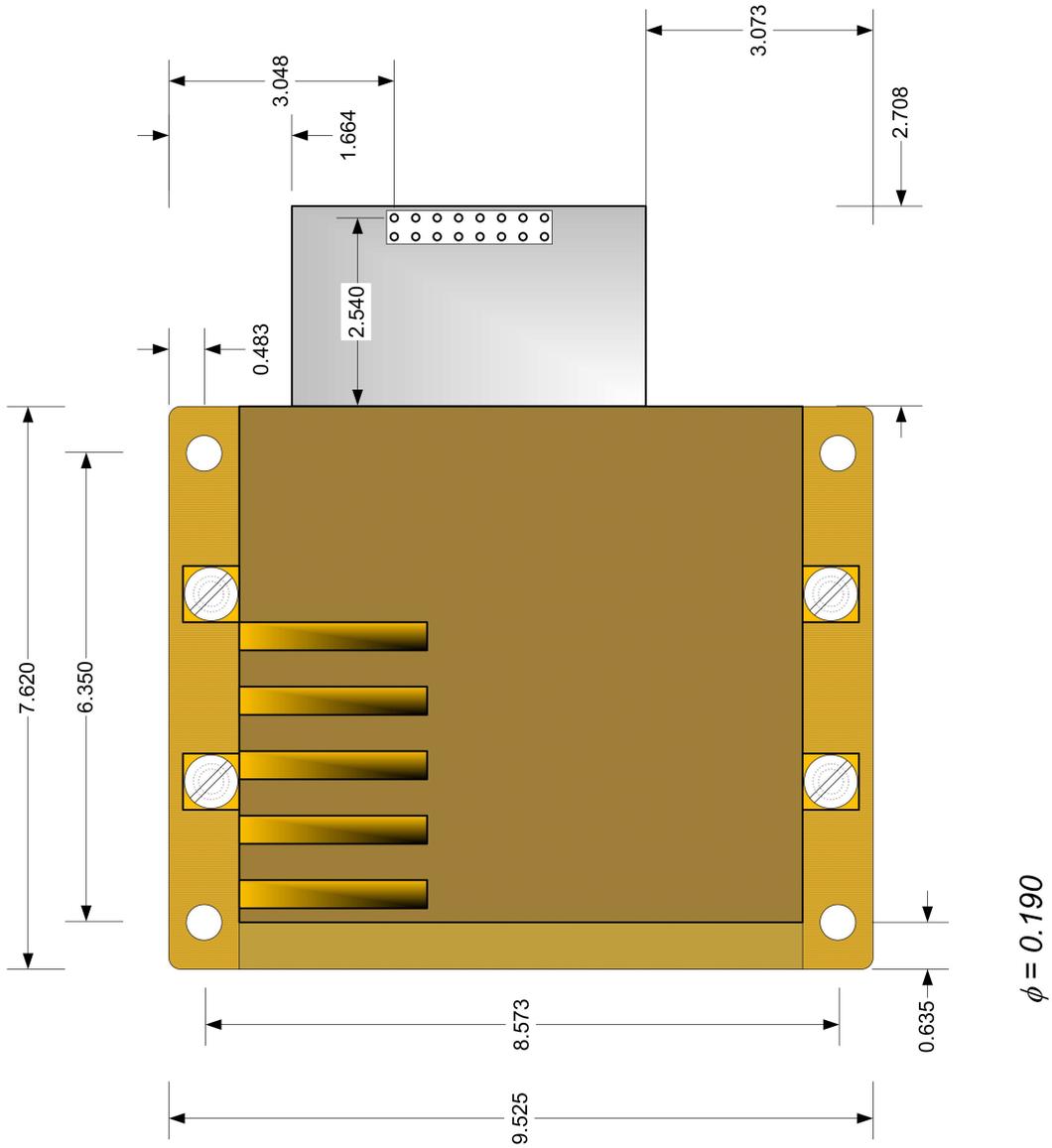


Figure C.24: NAV440™ Dimensions.

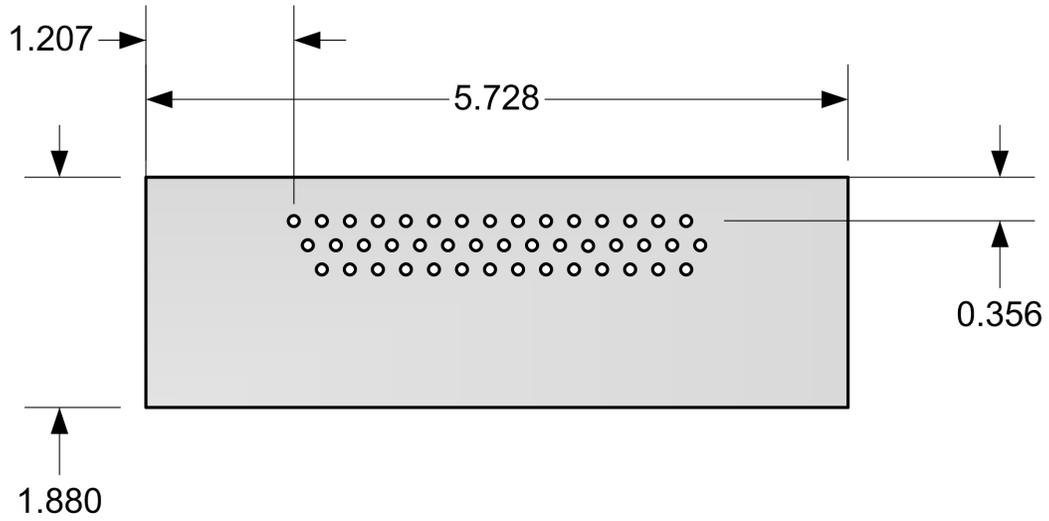


Figure C.25: *SSC Connector Dimensions.*

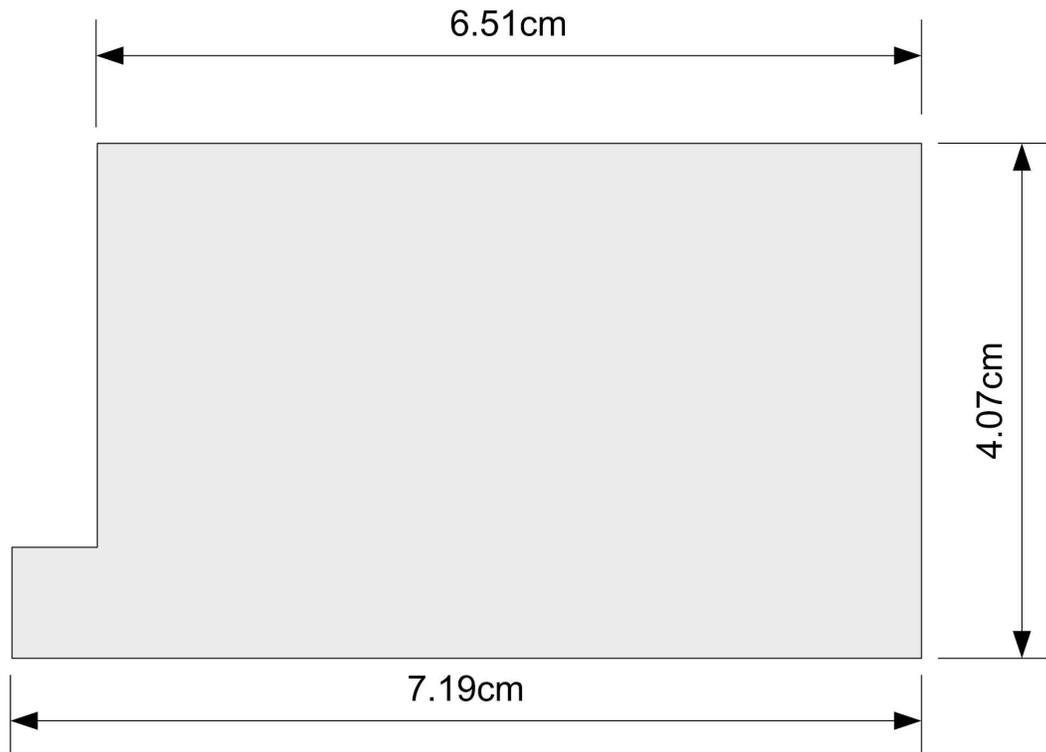


Figure C.26: *Radio Modem OEM module Dimensions.*

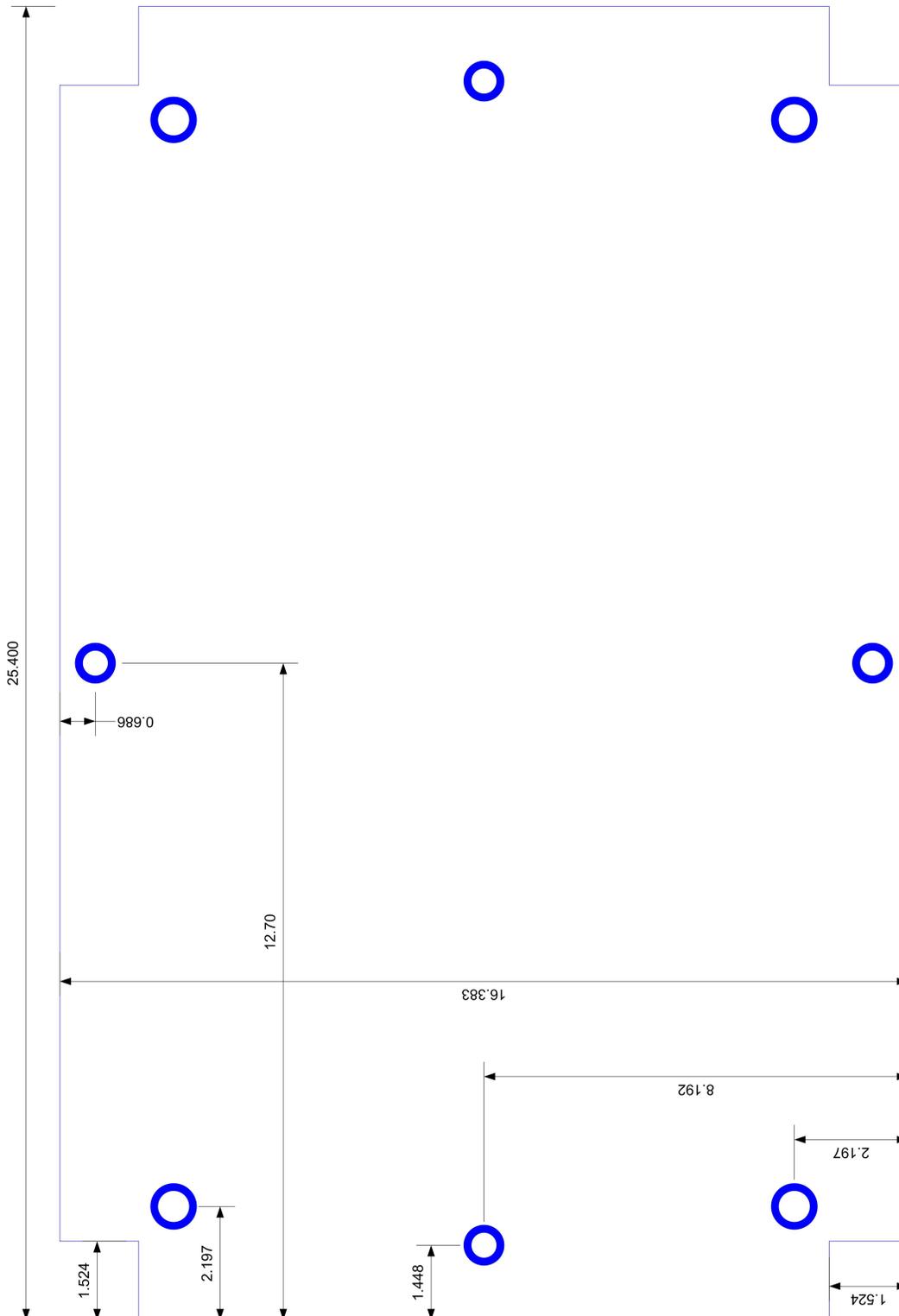


Figure C.27: PCB Dimensions.

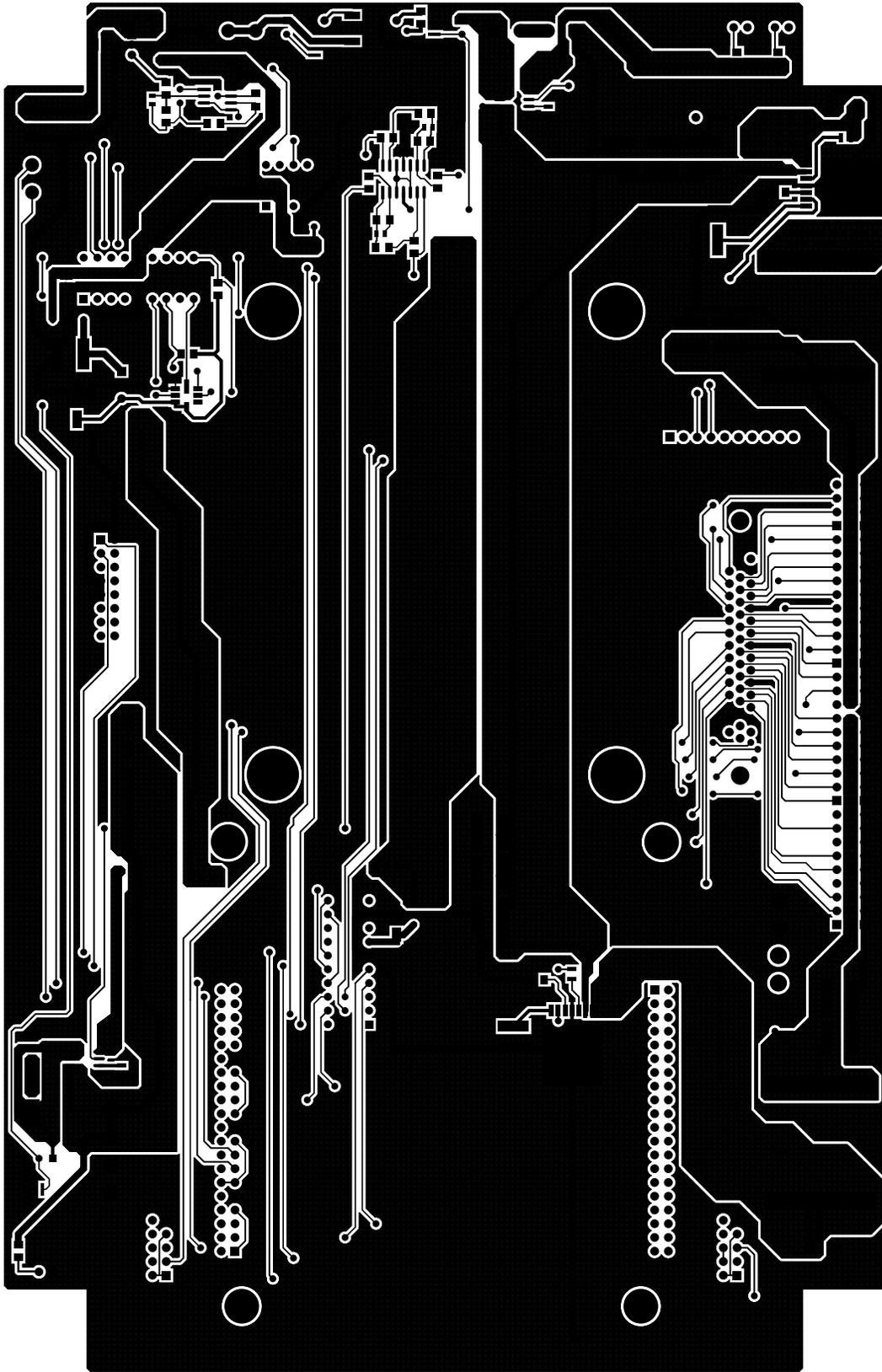


Figure C.28: PCB Top Side tracks. Not at scale.

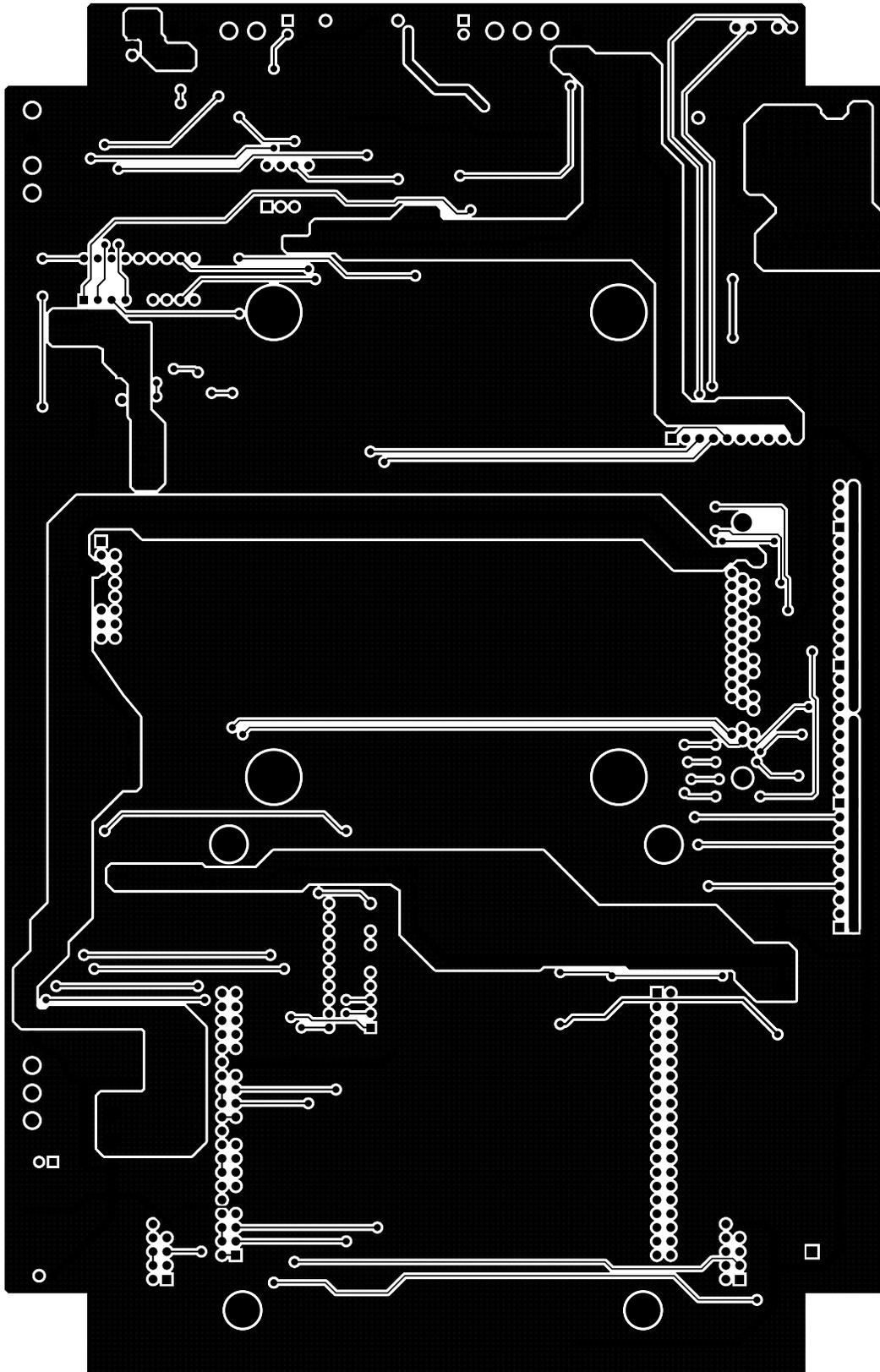
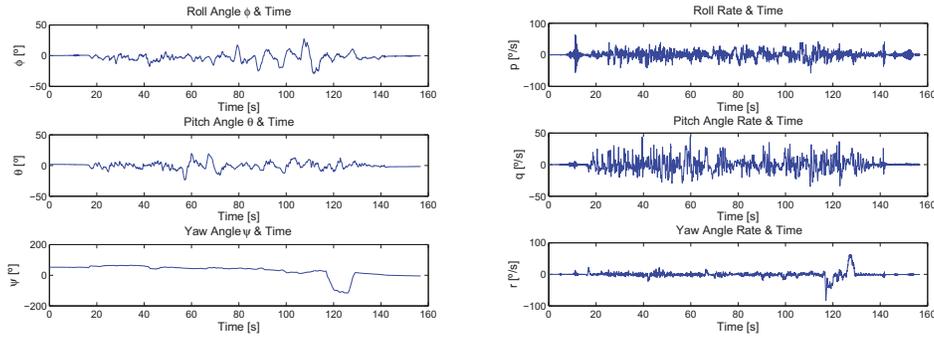


Figure C.29: *PCB Bottom Side tracks. Not at scale.*

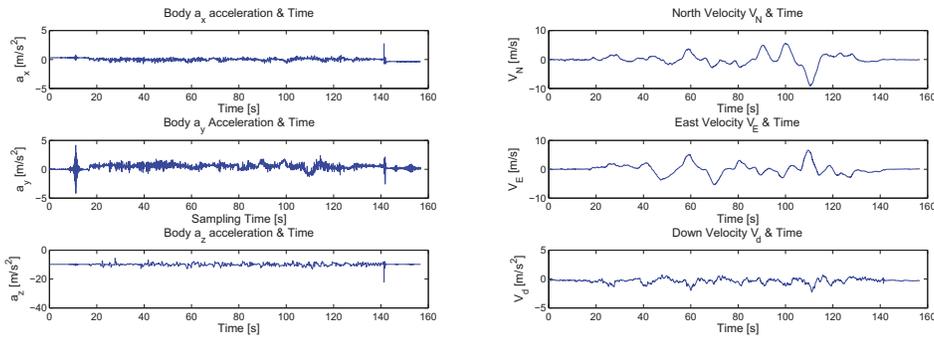
C.7 Data Flight Graphics

Figures C.31 to C.33 show the acquired data by the ECICH from the second of flights meanwhile figures C.34 to C.36 correspond to the fourth flight.



(a) Euler angles.

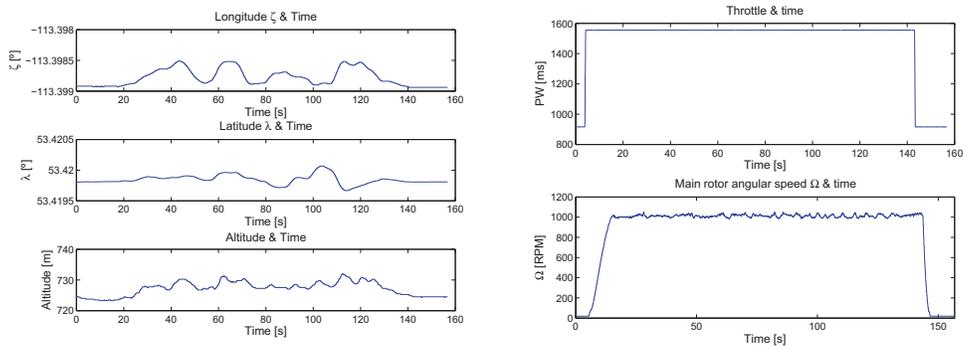
(b) Body angular rates.



(c) Body Accelerations.

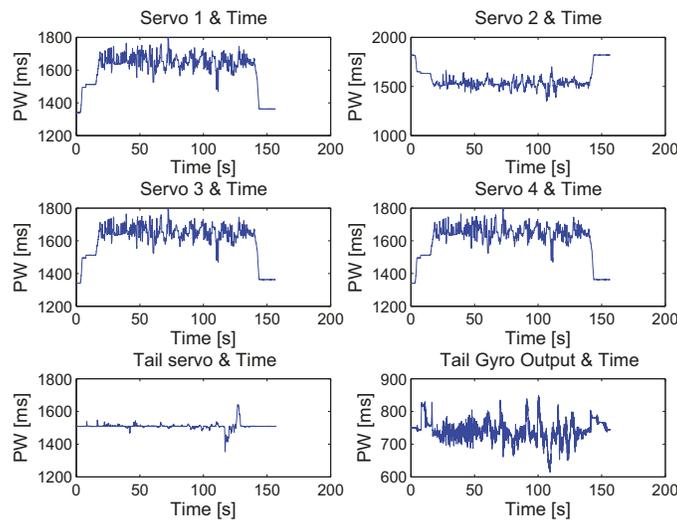
(d) Frame C_N velocities.

Figure C.31: Euler angles, body angular rates, body accelerations and NED velocities data from second flight.



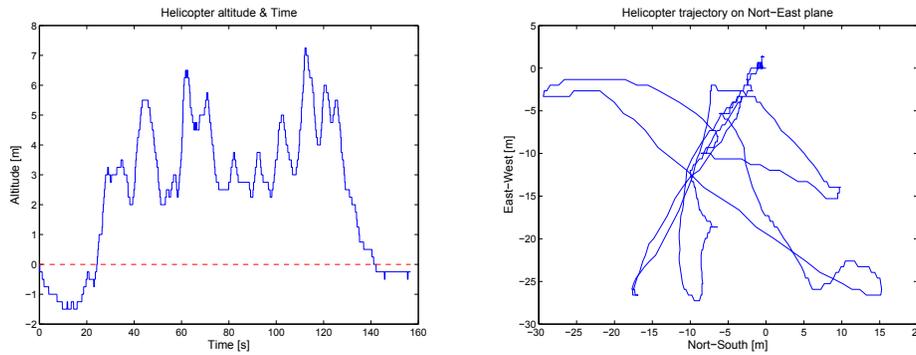
(a) GPS position.

(b) Throttle and main rotor RPM.



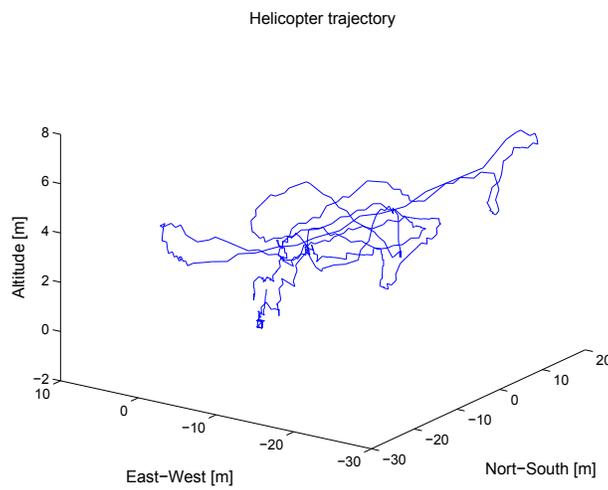
(c) Pilot servo commands.

Figure C.32: GPS position, Throttle command, main rotor RPM and pilot servo commands from second flight.



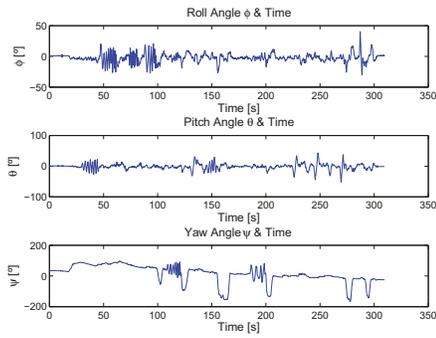
(a) Helicopter GPS altitude.

(b) Helicopter trajectory in the North-East plane.

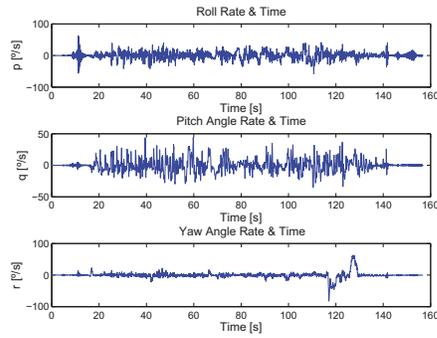


(c) Helicopter trajectory on space.

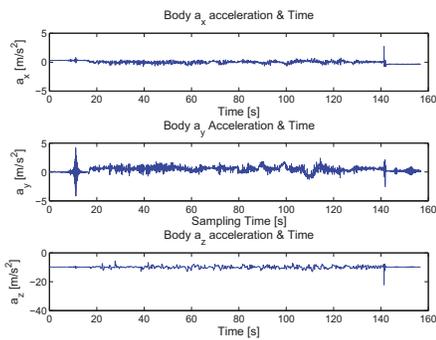
Figure C.33: Helicopter Trajectory and altitude from second flight.



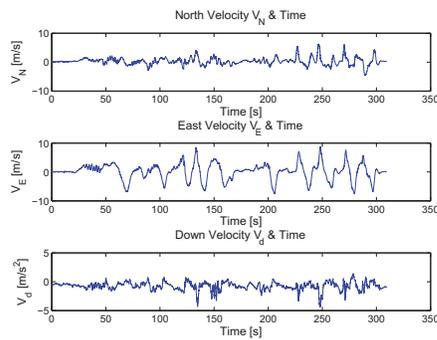
(a) Euler angles.



(b) Body angular rates.

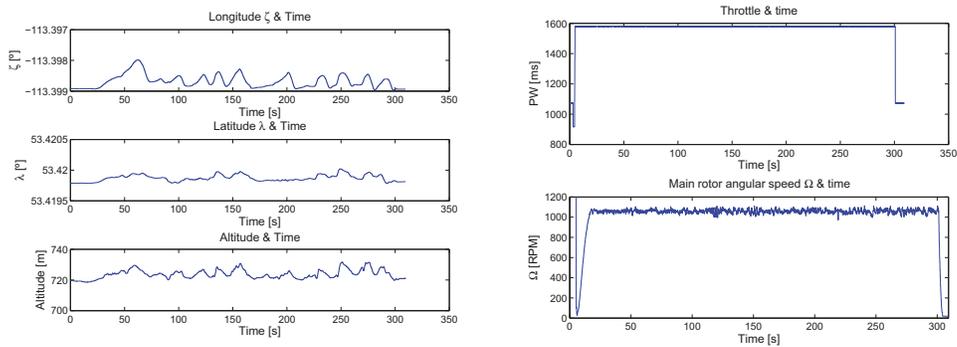


(c) Body Accelerations.



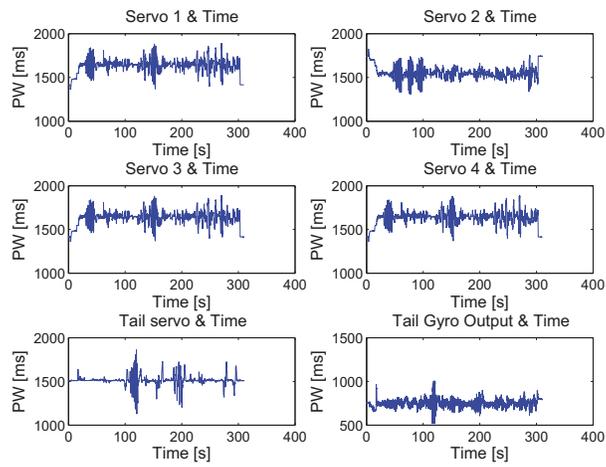
(d) Frame C_N velocities.

Figure C.34: Euler angles, body angular rates, body accelerations and NED velocities data from fourth flight.



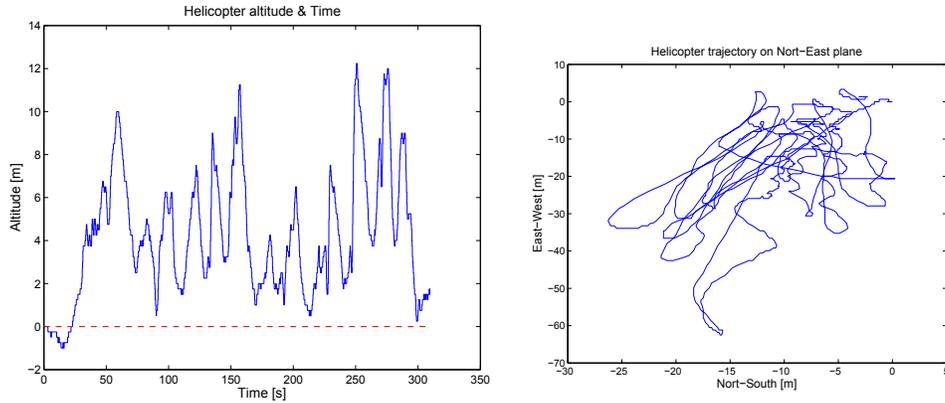
(a) GPS position.

(b) Throttle and main rotor RPM.



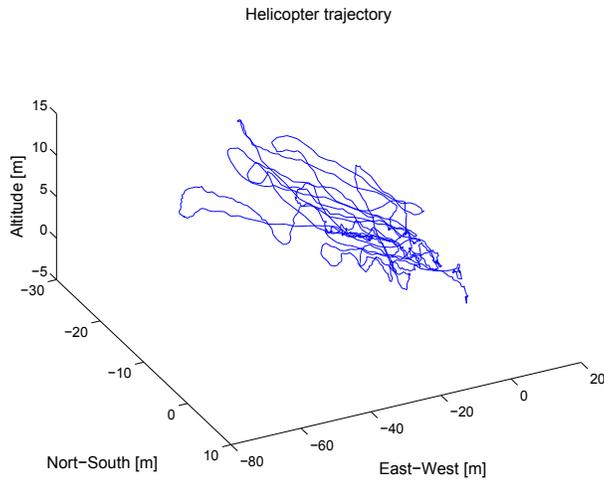
(c) Pilot servo commands.

Figure C.35: GPS position, Throttle command, main rotor RPM and pilot servo commands from fourth flight.



(a) Helicopter GPS altitude.

(b) Helicopter trajectory in the North-East plane.



(c) Helicopter trajectory on space.

Figure C.36: Helicopter Trajectory and altitude from fourth flight.

Appendix D

Helicopter Nonlinear Equations, Model Linearization and LQR Controller

D.1 Helicopter nonlinear equations.

Equations of force independent of the input commands:

$${}^b\mathbf{F}_o = \begin{bmatrix} -k_5 \left(\mu_p (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D_o} \mu_r u_{hub}}{C_{L\alpha}} \right) + \\ k_{13} \left(\pi e_t \mathcal{R}_t - C_{L\alpha_t} \right) \left(\mu_{p_t} \mathbf{v}_t + \frac{\pi e_t \mathcal{R}_t C_{D_o_t} \mu_{q_t} (u_t + \mu_{r_t} Y'_{tcg})}{C_{L\alpha_t}} \right) \\ \\ k_9 (\mu_{p_t} \mathbf{u}_t - 2\mu_{q_t} \mathbf{v}_t + \mu_{r_t} \mathbf{w}_t) + \\ k_5 \left(\mu_q (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D_o} \mu_r v_{hub}}{C_{L\alpha}} \right) \\ \\ -k_1 (2\mu_r (\mu_z - \lambda_i) + \mu_p \mu_x + \mu_q \mu_y) - \\ k_{13} \left((\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \mu_{r_t} \mathbf{v}_t - \frac{\pi e_t \mathcal{R}_t C_{D_o_t} \mu_{q_t} (\mathbf{w}_t - \mu_{p_t} Y'_{tcg})}{C_{L\alpha_t}} \right) \end{bmatrix}$$

Weight force:

$${}^b\mathbf{W} = m_T g \begin{bmatrix} -\sin \theta \\ \cos \theta \sin \phi \\ \cos \theta \cos \phi \end{bmatrix}$$

Drag force:

$${}^b\mathbf{F}_D = \begin{bmatrix} \frac{1}{2}\rho v_{h_i} S_{x_{f_u}} u \\ \frac{1}{2}\rho v_{h_i} S_{y_{f_u}} v \\ \frac{1}{2}\rho S_{v_f} \left[0.602 (V_\infty^2 + V_{v_{f_1}}^2) C_{D_{v_{f_1}}} + 0.38 (V_\infty^2 + V_{v_{f_2}}^2) C_{D_{v_{f_2}}} \right] \end{bmatrix}$$

Dependant force equations:

$$\begin{aligned}
{}^b\mathbf{F}_\theta = & \left[\begin{aligned}
& k_6 \left\{ \frac{C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{col}^2}{4} + \frac{C_{L\alpha} u_{hub} v_{hub} \delta_{col} \delta_{lat}}{4} - \right. \\
& \quad \left. \frac{C_{L\alpha} [\kappa_2 \mu_r^2 + \frac{3\kappa_4}{4} (3u_{hub}^2 + v_{hub}^2)] \delta_{col} \delta_{lon}}{6} + \right. \\
& \quad \frac{C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{lat}^2}{16} + \frac{3C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{lon}^2}{16} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{32} (\mu_q u_{hub} + \mu_p v_{hub}) \delta_{lat} - \\
& \quad \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left(\mu_r (\mu_z - \lambda_i) - \frac{3\mu_p \mu_x}{4} - \frac{\mu_q v_{hub}}{4} + \frac{3\mu_q \mu_p Z'_{rcg}}{4} \right) \delta_{lon} - \\
& \quad \left. \frac{C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{lat} \delta_{lon}}{8} + \frac{(\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left[-\frac{\kappa_2 \mu_q \mu_r}{3} + \kappa_4 u_{hub} (\mu_z - \lambda_i) \right] \delta_{col} \right\} \\
& \quad + k_{14} \left\{ \frac{\kappa_{t3} C_{L\alpha_t} \mu_{qt} (\mathbf{u}_t + \mu_{rt} Y'_{tcg}) \delta_{tail}^2}{4} \right. \\
& \quad \left. + \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left[-\frac{\kappa_{t2} \mu_{pt} \mu_{qt}}{3} + \kappa_{t4} (u_{hub}^2 + 3v_{hub}^2) \mathbf{v}_t \right] \delta_{tail} \right\} \\
& k_6 \left\{ -\frac{C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{col}^2}{4} - \frac{C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{lon}^2}{16} - \frac{C_{L\alpha} [\kappa_2 \mu_r^2 + \frac{3\kappa_4}{4} (u_{hub}^2 + 3v_{hub}^2)] \delta_{col} \delta_{lat}}{6} + \right. \\
& \quad \frac{C_{L\alpha} \kappa_4 u_{hub} v_{hub} \delta_{col} \delta_{lon}}{4} - \frac{3C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{lat}^2}{16} + \\
& \quad \frac{(\pi e \mathcal{R} - 2C_{L\alpha})}{4} \left[\frac{\kappa_2 \mu_q \mu_r}{3} - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right] \delta_{col} + \\
& \quad \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left(-\mu_r (\mu_z - \lambda_i) + \frac{3\mu_p \mu_y}{4} + \frac{\mu_p u_{hub}}{4} + \frac{3\mu_q \mu_p Z'_{rcg}}{4} \right) \delta_{lat} - \\
& \quad \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{32} (\mu_q u_{hub} + \mu_p v_{hub}) \delta_{lon} + \frac{C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{lat} \delta_{lon}}{8} \right\} + \\
& k_{10} \left\{ \frac{\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} \left[(\mathbf{u}_t + \mu_{rt} Y'_{tcg})^2 + (\mathbf{w}_t - \mu_{pt} Y'_{tcg})^2 \right] \right\} \delta_{tail} \\
& k_2 \left\{ \kappa_3 \mu_r v_{hub} \delta_{lat} - \kappa_3 \mu_r u_{hub} \delta_{lon} + \left[\frac{2\kappa_2 \mu_r^2}{3} + \kappa_4 (u_{hub}^2 + v_{hub}^2) \right] \delta_{col} \right\} + \\
& k_{14} \left\{ \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} (\mathbf{w}_t - \mu_{pt} Y'_{tcg}) \mathbf{v}_t \right) \delta_{tail} - \right. \\
& \quad \left. \frac{C_{L\alpha_t} \kappa_{t3} \mu_{qt} (\mathbf{w}_t - \mu_{pt} Y'_{tcg}) \delta_{tail}^2}{4} \right\}
\end{aligned} \right]
\end{aligned}$$

Total force:

$$\mathbf{F}_T = \mathbf{F}_o + \mathbf{F}_W + \mathbf{F}_D + \mathbf{F}_\theta \delta_\theta$$

State acceleration equations: $\dot{\mathbf{v}}_{cg} = \frac{\mathbf{F}_T}{m_T} - \boldsymbol{\omega} \times \mathbf{v}_{cg}$

$$\begin{aligned}
 & \left. \frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{col}^2}{4m_T} + \left\{ \frac{k_6 C_{L\alpha} u_{hub} v_{hub} \delta_{lat}}{4m_T} - \frac{k_6 C_{L\alpha} [\kappa_2 \mu_r^2 + \frac{3\kappa_4}{4} (3u_{hub}^2 + v_{hub}^2)] \delta_{lon}}{6} m_T + \frac{k_6 (\pi e \mathcal{R} - 2C_{L\alpha}) (-\frac{\kappa_2}{3} \mu_r \mu_r + \kappa_4 u_{hub} (\mu_z - \lambda_i))}{4m_T} \delta_{col} + \frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{lat}^2}{16m_T} \right\} \right. \\
 & \left. \left(-\frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{lon}}{8m_T} - \frac{k_6 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) (\mu_q u_{hub} + \mu_r v_{hub})}{32m_T} \right) \delta_{lat} + \frac{3k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{lon}^2}{16m_T} - \frac{k_6 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left(\mu_r (\mu_z - \lambda_i) - \frac{3\mu_r \mu_x}{4} - \frac{3\mu_q \mu_r Z'_{rcg}}{4} \right) \delta_{lon}}{8m_T} \right. \\
 & \left. + \frac{k_{14} \kappa_3 C_{L\alpha} \mu_{qt} (\mu_r + \mu_{rt} Y'_{tcg}) \delta_{tail}^2}{4m_T} + \frac{k_{14} (\pi e_t \mathcal{R}_t - 2C_{L\alpha t}) (-\frac{\kappa_{t2}}{3} \mu_{pt} \mu_{qt} + \kappa_{t4} (u_{hub}^2 + 3v_{hub}^2) V_t) \delta_{tail}}{4m_T} + \frac{k_5 \left[-\mu_p (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) + \frac{\pi e \mathcal{R} C_{D_0} \mu_r v_{hub}}{C_{L\alpha}} \right]}{m_T} \right. \\
 & \left. \frac{k_{13} (\pi e_t \mathcal{R}_t - C_{L\alpha t}) \left[\mu_{pt} v_t + \frac{\pi e_t \mathcal{R}_t C_{D_0 t} \mu_{qt} (u_t + \mu_{rt} Y'_{tcg})}{C_{L\alpha t}} \right]}{m_T} - g \sin \theta - qv + rv + \frac{\rho v_{h_i} S_{x_f u}}{2m_T} \right. \\
 & \left. \frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{col}^2}{4m_T} + \left(-\frac{k_6 C_{L\alpha} [\kappa_2 \mu_r^2 + \frac{3\kappa_4}{4} (u_{hub}^2 + 3v_{hub}^2)] \delta_{lat}}{6m_T} + \frac{k_6 C_{L\alpha} \kappa_4 u_{hub} v_{hub} \delta_{lon}}{4m_T} + \frac{k_6 (\pi e \mathcal{R} - 2C_{L\alpha}) \left[\frac{\kappa_2 \mu_q \mu_r}{3} - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right]}{4m_T} \right) \delta_{col} - \frac{3k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{lat}^2}{16m_T} \right. \\
 & \left. \left(\frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} \delta_{lon}}{8m_T} + \frac{k_6 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left(-\mu_r (\mu_z - \lambda_i) + \frac{3\mu_r \mu_x}{4} + \frac{3\mu_q \mu_r Z'_{rcg}}{4} \right)}{8m_T} \right) \delta_{lat} - \frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} \delta_{lon}^2}{16m_T} \right. \\
 & \left. \frac{k_{10} \left\{ \frac{2\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} \left[(u_t + \mu_{rt} Y'_{tcg})^2 + (w_t - \mu_{pt} Y'_{tcg})^2 \right] \right\} \delta_{tail}}{m_T} + \frac{k_5 \left[\mu_q (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D_0} \mu_r v_{hub}}{C_{L\alpha}} \right]}{m_T} \right. \\
 & \left. \frac{\rho v_{h_i} S_{y_f v}}{2m_T} \right. \\
 & \left. \frac{k_2 \left[\frac{2\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} (u_{hub}^2 + v_{hub}^2) \right] \delta_{col}}{m_T} + \frac{k_2 \kappa_3 \mu_r v_{hub} \delta_{lat}}{m_T} - \frac{k_2 \kappa_3 \mu_r u_{hub} \delta_{lon}}{m_T} - \frac{k_{14} C_{L\alpha t} \kappa_{t3} \mu_{qt} (w_t - \mu_{pt} Y'_{tcg}) \delta_{tail}^2}{4m_T} + \frac{k_{14} (\pi e_t \mathcal{R}_t - 2.0 C_{L\alpha t}) \left[\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} (w_t - \mu_{pt} Y'_{tcg}) v_t \right] \delta_{tail}}{4m_T} \right. \\
 & \left. \frac{[-2\mu_r (\mu_z - \lambda_i) - \mu_p \mu_x - \mu_q \mu_y] k_1}{m_T} + \left[-(\pi e_t \mathcal{R}_t - C_{L\alpha t}) \mu_{rt} v_t + \frac{\pi e_t \mathcal{R}_t C_{D_0 t} \mu_{qt} (w_t - \mu_{pt} Y'_{tcg})}{C_{L\alpha t}} \right] \frac{k_{13}}{m_T} + g \cos \theta \cos \phi - pv + qu + \right. \\
 & \left. \frac{\rho S_{v_f} (V_\infty^2 + V_{v_f1}^2) C_{D_{v_f1}} + 0.38 (V_\infty^2 + V_{v_f2}^2) C_{D_{v_f2}}}{2m_T} \right)
 \end{aligned}$$

Non-dependant Moment equations:

$$\mathbf{M}_{cg_o} = \left[\begin{array}{l} -k_7 \left[\kappa_3 (\mu_z - \lambda_i) u_{hub} + \frac{\kappa_1 \mu_p \mu_r}{2} \right] + k_9 z_{t_{cg}} (\mu_{p_t} \mathbf{u}_t - 2\mu_{q_t} \mathbf{v}_t + \mu_{r_t} \mathbf{w}_t) + \\ k_5 z_{r_{cg}} \left[\mu_q (\mu_z - \lambda_i) (C_{L_\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D_o} \mu_r v_{hub}}{C_{L_\alpha}} \right] + \\ k_{13} y_{t_{cg}} \left[(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \mu_{r_t} \mathbf{v}_t - \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} \mu_{q_t} (\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}})}{C_{L_{\alpha_t}}} \right] \\ \\ k_7 \left[\kappa_3 (\mu_z - \lambda_i) v_{hub} + \frac{\kappa_1 \mu_q \mu_r}{2} \right] + \\ k_5 z_{r_{cg}} \left(\mu_p (\mu_z - \lambda_i) (C_{L_\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D_o} \mu_r u_{hub}}{C_{L_\alpha}} \right) - \\ k_{13} x_{t_{cg}} \left[(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \mu_{r_t} \mathbf{v}_t - \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} \mu_{q_t} (\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}})}{C_{L_{\alpha_t}}} \right] \\ -k_{13} z_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left(\mu_{p_t} \mathbf{v}_t + \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} \mu_{q_t} (\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}})}{C_{L_{\alpha_t}}} \right) - \\ k_{11} \left\{ (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left(\frac{\kappa_{t_1}}{4} (\mu_{r_t}^2 + \mu_{p_t}^2) - \kappa_{t_3} \mathbf{v}_t^2 \right) + \right. \\ \left. \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} (\kappa_{t_1} \mu_{q_t}^2 + \kappa_{t_3} [(u_t + \mu_{r_t} Y'_{t_{cg}})^2 + (\mathbf{w}_t - \mu_{p_t} Y'_{t_{cg}})^2])}{2C_{L_{\alpha_t}}} \right\} \\ \\ k_3 \left\{ \frac{(C_{L_\alpha} - \pi e \mathcal{R})}{2} [4\kappa_3 (\mu_z - \lambda_i)^2 + \kappa_1 (\mu_p^2 + \mu_q^2)] - \right. \\ \left. \frac{\pi e \mathcal{R} C_{D_o} [\kappa_1 \mu_r^2 + \kappa_3 (u_{hub}^2 + v_{hub}^2)]}{C_{L_\alpha}} \right\} - k_9 x_{t_{cg}} (\mu_{p_t} \mathbf{u}_t - 2\mu_{q_t} \mathbf{v}_t + \mu_{r_t} \mathbf{w}_t) + \\ k_{13} y_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left(\mu_{p_t} \mathbf{v}_t + \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} \mu_{q_t} (\mathbf{u}_t + \mu_{r_t} Y'_{t_{cg}})}{C_{L_{\alpha_t}}} \right) \end{array} \right]$$

Flay bar dynamics:

$$\mathbf{M}_{f_b} = \begin{bmatrix} \alpha_2 \dot{\theta} - \alpha_1 \delta \theta_{lon} \\ -\alpha_4 \dot{\phi} + \alpha_3 \delta \theta_{lat} \\ 0 \end{bmatrix}$$

Dependant moment equations:

$$\mathbf{M}_{cg_\theta} = [M_{cg_{\theta_x}}, M_{cg_{\theta_y}}, M_{cg_{\theta_z}}]$$

where $M_{cg\theta_x}$, $M_{cg\theta_y}$, and $M_{cg\theta_z}$ are:

$$\begin{aligned}
 M_{cg\theta_x} = & -\frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{col}^2}{4} + \\
 & \left\{ \frac{k_6 C_{L\alpha} u_{hub} v_{hub} z_{rcg} \delta_{lon}}{4\kappa_4} - \frac{k_6 C_{L\alpha}}{6} \left[\kappa_2 \mu_r^2 + \frac{3\kappa_4}{4} (u_{hub}^2 + 3v_{hub}^2) \right] z_{rcg} \delta_{lat} - \right. \\
 & \left. k_8 \kappa_2 \mu_r u_{hub} + \frac{k_6 (\pi e AR - 2C_{L\alpha})}{4} \left[\frac{\kappa_2 \mu_q \mu_r}{3} - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right] z_{rcg} \right\} \delta_{col} - \\
 & \frac{3k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{lat}^2}{16} + \left\{ \frac{3k_8 \kappa_3 u_{hub} v_{hub}}{8} + \frac{k_6 \kappa_3 (\pi e AR - 2C_{L\alpha})}{8} [-\mu_r (\mu_z - \lambda_i) + \right. \\
 & \left. \frac{3\mu_q \mu_y}{4} + \frac{\mu_p u_{hub}}{4} + \frac{3\mu_q \mu_p Z'_{rcg}}{4}] z_{rcg} + \frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{lon}}{8} \right\} \delta_{lat} - \\
 & \frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{lon}^2}{3} + \left\{ -\frac{k_6 \kappa_3 (\pi e AR - 2C_{L\alpha}) (\mu_q u_{hub} + \mu_p v_{hub}) z_{rcg}}{32} + \right. \\
 & \left[-\frac{3\kappa_1 \mu_r^2}{8} - \frac{3\kappa_3 (3u_{hub}^2 + v_{hub}^2)}{16} \right] k_8 \left. \right\} \delta_{lon} + \\
 & \left\{ \frac{2\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} \left[(u_t + \mu_{rt} Y'_{tcg})^2 + (w_t - \mu_{pt} Y'_{tcg})^2 \right] \right\} z_{tcg} \delta_{tail} k_{10} + \\
 & \left\{ -\frac{(\pi e_t R_t - 2C_{L\alpha t})}{4} \left[\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} (w_t - \mu_{pt} Y'_{tcg}) v_t \right] y_{tcg} \delta_{tail} + \right. \\
 & \left. \frac{C_{L\alpha t} \kappa_{t3} (w_t + \mu_{pt} Y'_{tcg}) \mu_{qt} y_{tcg} \delta_{tail}^2}{4} \right\} k_{14}
 \end{aligned}$$

$$\begin{aligned}
 M_{cg\theta_y} = & -\frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{col}^2}{4} + \left\{ \frac{k_6 C_{L\alpha}}{6} \left[\kappa_2 \mu_r^2 + \frac{3\kappa_4 (3u_{hub}^2 + v_{hub}^2)}{4} \right] z_{rcg} \delta_{lon} - \right. \\
 & \left. \frac{k_6 C_{L\alpha} \kappa_4 u_{hub} v_{hub} z_{rcg} \delta_{lat}}{4} - k_8 \kappa_2 \mu_r u_{hub} - \right. \\
 & \left. \frac{k_6 (\pi e AR - 2C_{L\alpha}) \left[-\frac{\kappa_2 \mu_q \mu_r}{3} + \kappa_4 u_{hub} (\mu_z - \lambda_i) \right] z_{rcg}}{4} \right\} \delta_{col} - \frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{lat}^2}{16} + \\
 & \left\{ \frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{lon}}{8} + \frac{k_6 \kappa_3 (\pi e AR - 2C_{L\alpha})}{32} (\mu_q u_{hub} + \mu_p v_{hub}) z_{rcg} + \right. \\
 & \left[\frac{3\kappa_1 \mu_r^2}{8} + \frac{3\kappa_3}{16} (u_{hub}^2 + 3v_{hub}^2) \right] k_8 \left. \right\} \delta_{lat} - \frac{3k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{lon}^2}{16} + \\
 & \left\{ \frac{3k_8 \kappa_3 u_{hub} v_{hub}}{8} + \frac{\kappa_3 (\pi e AR - 2C_{L\alpha}) (\mu_r (\mu_z - \lambda_i) - 3/4 \mu_p \mu_x - 1/4 \mu_q v_{hub} + 3/4 \mu_q \mu_p Z'_{rcg}) z_{rcg}}{8} \right\} \delta_{lon} + \\
 & \left\{ \frac{\kappa_{t2} (\pi e_t R_t - 2C_{L\alpha t}) (\mu_{qt} v_t - \frac{\mu_{rt} w_t}{2} - \frac{\mu_{pt} u_t}{2}) \delta_{tail}}{6} - \right. \\
 & \left. \frac{C_{L\alpha t}}{8} \left(\kappa_{t1} \mu_{qt}^2 + \kappa_{t3} \left[(u_t + \mu_{rt} Y'_{tcg})^2 + (\mu_{pt} Y'_{tcg} - w_t)^2 \right] \right) \delta_{tail}^2 \right\} k_{12} + \\
 & \left\{ \left[\frac{(\pi e_t R_t - 2C_{L\alpha t}) \left(\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} (w_t - \mu_{pt} Y'_{tcg}) v_t \right) x_{tcg}}{4} - \right. \right. \\
 & \left. \frac{(\pi e_t R_t - 2C_{L\alpha t})}{4} (\kappa_{t4} (u_t - \mu_{rt} Y'_{tcg}) v_t - \frac{\kappa_{t2} \mu_{pt} \mu_{qt}}{3}) z_{tcg} \right] \delta_{tail} + \\
 & \left. \left[-\frac{C_{L\alpha t} \kappa_{t3} \mu_{qt} (w_t + \mu_{pt} Y'_{tcg}) x_{tcg}}{4} - \frac{C_{L\alpha t} \kappa_{t3} \mu_{qt} (u_t + \mu_{rt} Y'_{tcg}) z_{tcg}}{4} \right] \delta_{tail}^2 \right\} k_{14}
 \end{aligned}$$

$$\begin{aligned}
 M_{cg\theta_z} = & \frac{k_4 C_{L\alpha} \left[\kappa_1 \mu_r^2 + \kappa_3 (v_{hub}^2 + v_{hub}^2) \right] \delta_{col}^2}{8} + \\
 & \left\{ -\frac{k_4 C_{L\alpha} \kappa_2 \mu_r u_{hub} \delta_{lon}}{3} + \frac{k_4 C_{L\alpha} \kappa_2 \mu_r v_{hub} \delta_{lat}}{3} - \frac{k_4 \kappa_2 (\pi e AR - 2C_{L\alpha}) \left[(\lambda_i - \mu_z) \mu_r + \frac{\mu_p \mu_x}{2} + \frac{\mu_q \mu_y}{2} \right]}{6} \right\} \delta_{col} + \\
 & \frac{k_4 C_{L\alpha} \left[\kappa_1 \mu_r^2 + \frac{\kappa_3}{16} (u_{hub}^2 + 3v_{hub}^2) \right] \delta_{lat}^2}{16} + \left[-\frac{k_4 (\pi e AR - 2C_{L\alpha}) (\kappa_3 (\mu_z - \lambda_i) v_{hub} - \kappa_1 \mu_q \mu_r)}{8} - \right.
 \end{aligned}$$

$$\begin{aligned}
& \frac{k_4 C_{L\alpha} \kappa_3 u_{hub} v_{hub} \delta_{lon}}{8} \Big] \delta_{lat} + \frac{k_4 C_{L\alpha} \left[\kappa_1 \mu_r^2 + \frac{\kappa_3}{2} (3u_{hub}^2 + v_{hub}^2) \right] \delta_{lon}^2}{16} - \\
& \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) (\kappa_3 (\mu_z - \lambda_i) u_{hub} - \kappa_1 \mu_p \mu_r) \delta_{lon}}{8} - \\
& k_{10} \left\{ \frac{2\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} \left[(u_t + \mu_{rt} Y'_{tcg})^2 + (w_t - \mu_{pt} Y'_{tcg})^2 \right] \right\} x_{tcg} \delta_{tail} + \\
& k_{14} \left\{ \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) (\kappa_{t4} (u_t - \mu_{rt} Y'_{tcg}) v_t - \frac{\kappa_{t2}}{3} \mu_{pt} \mu_{qt}) y_{tcg} \delta_{tail}}{4} + \frac{C_{L\alpha_t} \kappa_{t3} (u_t + \mu_{rt} Y'_{tcg}) \mu_{qt} y_{tcg} \delta_{tail}^2}{4} \right\}
\end{aligned}$$

Total moment:

$$\mathbf{M}_{cg} = \mathbf{M}_{cg_0} + \mathbf{M}_{f_0} + (\mathbf{M}_{f_\theta} + \mathbf{M}_{cg_\theta}) \delta_\theta$$

State angular acceleration equation:

$$\dot{\boldsymbol{\omega}} = (\mathbf{I} + \mathbf{I}_s)^{-1} \left\{ \mathbf{M}_{cg} - \boldsymbol{\omega} \times [(\mathbf{I} + \mathbf{I}_s) \boldsymbol{\omega} + \mathbf{I}_s \boldsymbol{\Omega}] \right\} = [\dot{\omega}_x, \dot{\omega}_y, \dot{\omega}_z]$$

where $\dot{\omega}_x$, $\dot{\omega}_y$, and $\dot{\omega}_z$ are:

$$\begin{aligned}
\dot{\omega}_x = & -\frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{col}^2}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \left\{ \frac{-k_6 C_{L\alpha} \left[\kappa_2 \mu_r^2 + \frac{3\kappa_4}{4} (u_{hub}^2 + 3v_{hub}^2) \right] z_{rcg} \delta_{lat}}{3(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \frac{k_6 C_{L\alpha} u_{hub} v_{hub} z_{rcg} \delta_{lon}}{2(2I_x + I_{x_{bl}} + I_{y_{bl}}) \kappa_4} \right. \\
& \left. \frac{k_6 (\pi e \mathcal{R} - 2C_{L\alpha}) \left[\frac{\kappa_2 \mu_q \mu_r}{2} - \kappa_4 v_{hub} (\mu_z - \lambda_i) \right] z_{rcg} - \frac{2k_8 \kappa_2 \mu_r u_{hub}}{2I_x + I_{x_{bl}} + I_{y_{bl}}}}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})} \right\} \delta_{col} - \frac{3k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{lat}^2}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \\
& \left(\frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{lon}}{4(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \frac{k_6 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left(-\mu_r (\mu_z - \lambda_i) + \frac{3\mu_q \mu_y}{4} + \frac{\mu_p u_{hub}}{4} + \frac{3\mu_q \mu_p z'_{rcg}}{4} \right) z_{rcg}}{4(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \left. \frac{3k_8 \kappa_3 u_{hub} v_{hub}}{4(2I_x + I_{x_{bl}} + I_{y_{bl}})} \right) \delta_{lat} - \frac{2k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{lon}^2}{3(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \left\{ -\frac{k_6 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) (\mu_q u_{hub} + \mu_p v_{hub}) z_{rcg}}{16(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \left. \frac{2 \left[-\frac{3\kappa_1 \mu_r^2}{8} - \frac{3\kappa_3}{16} (3u_{hub}^2 + v_{hub}^2) \right] k_8 - \frac{2\alpha_1}{2I_x + I_{x_{bl}} + I_{y_{bl}}}}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \right\} \delta_{lon} + \frac{k_{14} C_{L\alpha_t} \kappa_{t3} (W_t + \mu_{pt} Y'_{tcg}) \mu_{qt} y_{tcg} \delta_{tail}^2}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \\
& \left(\frac{2k_{10} \left(\frac{2\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} \left[(u_t + \mu_{rt} Y'_{tcg})^2 + (w_t - \mu_{pt} Y'_{tcg})^2 \right] \right) z_{tcg}}{2I_x + I_{x_{bl}} + I_{y_{bl}}} - \right. \\
& \left. \frac{k_{14} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \left(\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} (w_t - \mu_{pt} Y'_{tcg}) v_t \right) y_{tcg}}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})} \right) \delta_{tail} + \\
& \frac{2z_{rcg} k_5 \left(\mu_q (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D_0} \mu_r v_{hub}}{C_{L\alpha}} \right)}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} + \frac{2(-\kappa_3 (\mu_z - \lambda_i) u_{hub} - \frac{\kappa_1 \mu_p \mu_r}{2}) k_7}{2I_x + I_{x_{bl}} + I_{y_{bl}}} + \\
& \frac{2z_{tcg} k_9 (\mu_{pt} u_t - 2\mu_{qt} v_t + \mu_{rt} w_t)}{2I_x + I_{x_{bl}} + I_{y_{bl}}} + \frac{2y_{tcg} k_{13} \left((\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \mu_{rt} v_t - \frac{\pi e_t \mathcal{R}_t C_{D_{\alpha_t}} \mu_{qt} (w_t - \mu_{pt} Y'_{tcg})}{C_{L\alpha_t}} \right)}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} +
\end{aligned}$$

$$\begin{aligned}
& 2 \frac{-q[(I_z + I_{z_{bl}})r - I_{z_{bl}}\Omega] + r\left(I_y + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2}\right)q + \alpha_2(q \cos \phi - r \sin \phi)}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \\
\dot{\omega}_y = & -\frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{col}^2}{2(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \left\{ -\frac{k_6 C_{L\alpha} \kappa_4 u_{hub} v_{hub} z_{rcg} \delta_{lat}}{3(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \frac{k_6 C_{L\alpha} [\kappa_2 \mu_r^2 + 3/4 \kappa_4 (3u_{hub}^2 + v_{hub}^2)] z_{rcg} \delta_{lon}}{3(2I_y + I_{x_{bl}} + I_{y_{bl}})} - \\
& \left. \frac{k_6 (\pi e \mathcal{R} - 2C_{L\alpha}) [-1/3 \kappa_2 \mu_q \mu_r + \kappa_4 u_{hub} (\mu_z - \lambda_i)] z_{rcg}}{2(2I_y + I_{x_{bl}} + I_{y_{bl}})} - 2 \frac{k_8 \kappa_2 \mu_r u_{hub}}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \right\} \delta_{col} - \\
& \frac{k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{lat}^2}{8(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \left\{ 1/4 \frac{k_6 C_{L\alpha} \kappa_3 \mu_r v_{hub} z_{rcg} \delta_{lon}}{4(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \frac{k_6 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) (\mu_q u_{hub} + \mu_p v_{hub}) z_{rcg}}{16(2I_y + I_{x_{bl}} + I_{y_{bl}})} + 2 \left[\frac{3\kappa_1 \mu_r^2}{8} + \frac{3\kappa_3}{16} (u_{hub}^2 + 3v_{hub}^2) \right] \frac{k_8}{2I_y + I_{x_{bl}} + I_{y_{bl}}} + \\
& \left. 2 \frac{\alpha_3}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \right\} \delta_{lat} - \frac{3k_6 C_{L\alpha} \kappa_3 \mu_r u_{hub} z_{rcg} \delta_{lon}^2}{8(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \left[\frac{3k_8 \kappa_3 u_{hub} v_{hub}}{4(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) \left(\mu_r (\mu_z - \lambda_i) - \frac{3\mu_p \mu_x}{4} - \frac{\mu_q v_{hub}}{4} + \frac{3\mu_q \mu_p z'_{rcg}}{4} \right) z_{rcg}}{4(2I_y + I_{x_{bl}} + I_{y_{bl}})} \right] \delta_{lon} + \\
& \left\{ -\frac{k_{12} C_{L\alpha_t} \left(\kappa_{t1} \mu_{qt}^2 + \kappa_{t3} \left[(u_t + \mu_{rt} Y'_{t_{cg}})^2 + (\mu_{pt} Y'_{t_{cg}} - w_t)^2 \right] \right)}{4(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \left. 2 \left(-\frac{C_{L\alpha_t} \kappa_{t3} \mu_{qt}}{4} (w_t + \mu_{pt} Y'_{t_{cg}}) x_{t_{cg}} - \frac{C_{L\alpha_t} \kappa_{t3} \mu_{qt}}{4} (u_t + \mu_{rt} Y'_{t_{cg}}) z_{t_{cg}} \right) \frac{k_{14}}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \right\} \delta_{tail}^2 + \\
& \left\{ \frac{k_{12} \kappa_{t2} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) (\mu_{qt} v_t - \frac{\mu_{rt} w_t}{2} - \frac{\mu_{pt} u_t}{2})}{3(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \right. \\
& \left. \frac{2}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \left[\frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \left(\frac{\kappa_{t2} \mu_{rt} \mu_{qt}}{3} - \kappa_{t4} (w_t - \mu_{pt} Y'_{t_{cg}}) v_t \right) x_{t_{cg}}}{4} \right. \right. \\
& \left. \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \left(\kappa_{t4} (u_t - \mu_{rt} Y'_{t_{cg}}) v_t - \frac{\kappa_{t2} \mu_{pt} \mu_{qt}}{3} z_{t_{cg}} \right)}{4} \right] k_{14} \right\} \delta_{tail} + \\
& \frac{2z_{rcg} k_5}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\mu_p (\mu_z - \lambda_i) (C_{L\alpha} - \pi e \mathcal{R}) - \frac{\pi e \mathcal{R} C_{D\alpha} \mu_r u_{hub}}{C_{L\alpha}} \right] + \\
& 2 \frac{k_7 (\kappa_3 (\mu_z - \lambda_i) v_{hub} + \frac{\kappa_1 \mu_q \mu_r}{2})}{2I_y + I_{x_{bl}} + I_{y_{bl}}} + \\
& \frac{2k_{11}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ -(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \left(\frac{\kappa_{t1}}{4} (\mu_{rt}^2 + \mu_{pt}^2) - \kappa_{t3} v_t^2 \right) - \right. \\
& \left. \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} \left(\kappa_{t1} \mu_{qt}^2 + \kappa_{t3} \left[(u_t + \mu_{rt} Y'_{t_{cg}})^2 + (w_t - \mu_{pt} Y'_{t_{cg}})^2 \right] \right)}{2C_{L\alpha_t}} \right\} + \\
& \frac{2k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ -x_{t_{cg}} \left[(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \mu_{rt} v_t - \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} \mu_{qt} (w_t - \mu_{pt} Y'_{t_{cg}})}{C_{L\alpha_t}} \right] - \right. \\
& \left. z_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \left(\mu_{pt} v_t + \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} \mu_{qt} (u_t + \mu_{rt} Y'_{t_{cg}})}{C_{L\alpha_t}} \right) \right\} + \\
& 2 \frac{p[(I_z + I_{z_{bl}})r - I_{z_{bl}}\Omega] - r\left(I_x + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2}\right)p - \alpha_4 [p + (q \sin \phi + r \cos \phi) \tan \theta]}{2I_y + I_{x_{bl}} + I_{y_{bl}}}
\end{aligned}$$

$$\begin{aligned}
\dot{\omega}_z = & \frac{k_4 C_{L\alpha} (\kappa_1 \mu_r^2 + \kappa_3 (u_{hub}^2 + v_{hub}^2)) \delta_{col}^2}{8(I_z + I_{z_{bl}})} + \left(\frac{k_4 C_{L\alpha} \kappa_2 \mu_r v_{hub} \delta_{lat}}{3(I_z + I_{z_{bl}})} - \frac{k_4 C_{L\alpha} \kappa_2 \mu_r u_{hub} \delta_{lon}}{3(I_z + I_{z_{bl}})} \right. \\
& \left. \frac{k_4 \kappa_2 (\pi e \mathcal{R} - 2C_{L\alpha}) (\lambda_i - \mu_z) \mu_r + \frac{\mu_p \mu_x}{2} + \frac{\mu_q \mu_y}{2}}{6(I_z + I_{z_{bl}})} \right) \delta_{col} + \\
& \frac{k_4 C_{L\alpha} (\kappa_1 \mu_r^2 + \frac{\kappa_3}{2} (u_{hub}^2 + 3v_{hub}^2)) \delta_{lat}^2}{16(I_z + I_{z_{bl}})} + \left(-\frac{k_4 C_{L\alpha} \kappa_3 u_{hub} v_{hub} \delta_{lon}}{8(I_z + I_{z_{bl}})} - \right. \\
& \left. \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) (\kappa_3 (\mu_z - \lambda_i) v_{hub} - \kappa_1 \mu_q \mu_r)}{8(I_z + I_{z_{bl}})} \right) \delta_{lat} + \frac{k_4 C_{L\alpha} (\kappa_1 \mu_r^2 + \frac{\kappa_3}{2} (3u_{hub}^2 + v_{hub}^2)) \delta_{lon}^2}{16(I_z + I_{z_{bl}})} - \\
& \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) (\kappa_3 (\mu_z - \lambda_i) u_{hub} - \kappa_1 \mu_p \mu_r) \delta_{lon}}{8(I_z + I_{z_{bl}})} + \frac{k_{14} C_{L\alpha} \kappa_{t3} (u_t + \mu_{rt} Y'_{tcg}) \mu_{qt} y_{tcg} \delta_{tail}^2}{4(I_z + I_{z_{bl}})} + \\
& \left(-\frac{k_{10} \left(\frac{2\kappa_{t2} \mu_{qt}^2}{3} + \kappa_{t4} \left[(u_t + \mu_{rt} Y'_{tcg})^2 + (w_t - \mu_{pt} Y'_{tcg})^2 \right] \right) x_{tcg}}{I_z + I_{z_{bl}}} + \right. \\
& \left. \frac{k_{14} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) (\kappa_{t4} (u_t - \mu_{rt} Y'_{tcg}) v_t - \frac{\kappa_{t2} \mu_{pt} \mu_{qt}}{3}) y_{tcg}}{4(I_z + I_{z_{bl}})} \right) \delta_{tail} + \\
& \frac{k_3}{(I_z + I_{z_{bl}})} \left\{ \frac{(C_{L\alpha} - \pi e \mathcal{R}) [4\kappa_3 (\mu_z - \lambda_i)^2 + \kappa_1 (\mu_p^2 + \mu_q^2)]}{2} - \frac{\pi e \mathcal{R} C_{D\alpha} (\kappa_1 \mu_r^2 + \kappa_3 (u_{hub}^2 + v_{hub}^2))}{C_{L\alpha}} \right\} - \\
& \frac{x_{tcg} k_9 (\mu_{pt} u_t - 2\mu_{qt} v_t + \mu_{rt} w_t)}{I_z + I_{z_{bl}}} + \frac{y_{tcg} k_{13} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \left(\mu_{pt} v_t + \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} \mu_{qt} (u_t + \mu_{rt} Y'_{tcg})}{C_{L\alpha_t}} \right)}{(I_z + I_{z_{bl}})} + \\
& -pq \left[\left(I_y + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) + \left(I_x + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) \right]
\end{aligned}$$

State Euler rates:

$$\dot{\phi} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \mathbf{C}_b^i \boldsymbol{\omega} = \begin{bmatrix} p + q \sin \phi \tan \theta + r \cos \phi \tan \theta \\ q \cos \phi - r \sin \phi \\ q \sin \phi \sec \theta + r \cos \phi \sec \theta \end{bmatrix}$$

State linear velocities in the inertial frame:

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \end{bmatrix} = \mathbf{C}_b^i \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} u c\psi c\theta + v (-s\psi c\phi + c\psi s\theta s\phi) + w (s\psi s\phi + c\psi s\theta c\phi) \\ u s\psi c\theta + v (c\psi c\phi + s\psi s\theta s\phi) + w (-c\psi s\phi + s\psi s\theta c\phi) \\ -u s\theta + v c\theta s\phi + w c\theta c\phi \end{bmatrix}$$

D.2 Jacobian derivatives

Computation of the Jacobian for the state matrices \mathbf{A} and \mathbf{B} evaluated at the equilibrium point

$$\mathbf{A} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial u} & \frac{\partial \dot{u}}{\partial v} & \frac{\partial \dot{u}}{\partial w} & \frac{\partial \dot{u}}{\partial p} & \frac{\partial \dot{u}}{\partial q} & \frac{\partial \dot{u}}{\partial r} & \frac{\partial \dot{u}}{\partial \phi} & \frac{\partial \dot{u}}{\partial \theta} & \frac{\partial \dot{u}}{\partial \psi} \\ \frac{\partial \dot{v}}{\partial u} & \frac{\partial \dot{v}}{\partial v} & \frac{\partial \dot{v}}{\partial w} & \frac{\partial \dot{v}}{\partial p} & \frac{\partial \dot{v}}{\partial q} & \frac{\partial \dot{v}}{\partial r} & \frac{\partial \dot{v}}{\partial \phi} & \frac{\partial \dot{v}}{\partial \theta} & \frac{\partial \dot{v}}{\partial \psi} \\ \frac{\partial \dot{w}}{\partial u} & \frac{\partial \dot{w}}{\partial v} & \frac{\partial \dot{w}}{\partial w} & \frac{\partial \dot{w}}{\partial p} & \frac{\partial \dot{w}}{\partial q} & \frac{\partial \dot{w}}{\partial r} & \frac{\partial \dot{w}}{\partial \phi} & \frac{\partial \dot{w}}{\partial \theta} & \frac{\partial \dot{w}}{\partial \psi} \\ \frac{\partial \dot{p}}{\partial u} & \frac{\partial \dot{p}}{\partial v} & \frac{\partial \dot{p}}{\partial w} & \frac{\partial \dot{p}}{\partial p} & \frac{\partial \dot{p}}{\partial q} & \frac{\partial \dot{p}}{\partial r} & \frac{\partial \dot{p}}{\partial \phi} & \frac{\partial \dot{p}}{\partial \theta} & \frac{\partial \dot{p}}{\partial \psi} \\ \frac{\partial \dot{q}}{\partial u} & \frac{\partial \dot{q}}{\partial v} & \frac{\partial \dot{q}}{\partial w} & \frac{\partial \dot{q}}{\partial p} & \frac{\partial \dot{q}}{\partial q} & \frac{\partial \dot{q}}{\partial r} & \frac{\partial \dot{q}}{\partial \phi} & \frac{\partial \dot{q}}{\partial \theta} & \frac{\partial \dot{q}}{\partial \psi} \\ \frac{\partial \dot{r}}{\partial u} & \frac{\partial \dot{r}}{\partial v} & \frac{\partial \dot{r}}{\partial w} & \frac{\partial \dot{r}}{\partial p} & \frac{\partial \dot{r}}{\partial q} & \frac{\partial \dot{r}}{\partial r} & \frac{\partial \dot{r}}{\partial \phi} & \frac{\partial \dot{r}}{\partial \theta} & \frac{\partial \dot{r}}{\partial \psi} \\ \frac{\partial \dot{\phi}}{\partial u} & \frac{\partial \dot{\phi}}{\partial v} & \frac{\partial \dot{\phi}}{\partial w} & \frac{\partial \dot{\phi}}{\partial p} & \frac{\partial \dot{\phi}}{\partial q} & \frac{\partial \dot{\phi}}{\partial r} & \frac{\partial \dot{\phi}}{\partial \phi} & \frac{\partial \dot{\phi}}{\partial \theta} & \frac{\partial \dot{\phi}}{\partial \psi} \\ \frac{\partial \dot{\theta}}{\partial u} & \frac{\partial \dot{\theta}}{\partial v} & \frac{\partial \dot{\theta}}{\partial w} & \frac{\partial \dot{\theta}}{\partial p} & \frac{\partial \dot{\theta}}{\partial q} & \frac{\partial \dot{\theta}}{\partial r} & \frac{\partial \dot{\theta}}{\partial \phi} & \frac{\partial \dot{\theta}}{\partial \theta} & \frac{\partial \dot{\theta}}{\partial \psi} \\ \frac{\partial \dot{\psi}}{\partial u} & \frac{\partial \dot{\psi}}{\partial v} & \frac{\partial \dot{\psi}}{\partial w} & \frac{\partial \dot{\psi}}{\partial p} & \frac{\partial \dot{\psi}}{\partial q} & \frac{\partial \dot{\psi}}{\partial r} & \frac{\partial \dot{\psi}}{\partial \phi} & \frac{\partial \dot{\psi}}{\partial \theta} & \frac{\partial \dot{\psi}}{\partial \psi} \end{bmatrix} \quad \begin{array}{l} \mathbf{x} = \mathbf{x}_{eq} \\ \mathbf{u} = \mathbf{u}_{eq} \end{array}$$

$$\mathbf{B} = \begin{bmatrix} \frac{\partial \dot{u}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{u}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{u}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{u}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{v}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{v}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{v}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{v}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{w}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{w}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{w}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{w}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{p}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{p}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{p}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{p}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{q}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{q}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{q}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{q}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{r}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{r}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{r}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{r}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{\phi}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{\theta}}{\partial \delta_{\theta_{tail}}} \\ \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{col}}} & \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{lat}}} & \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{lon}}} & \frac{\partial \dot{\psi}}{\partial \delta_{\theta_{tail}}} \end{bmatrix} \quad \begin{array}{l} \mathbf{x} = \mathbf{x}_{eq} \\ \mathbf{u} = \mathbf{u}_{eq} \end{array}$$

Partial derivatives from the Jacobian for the dynamics matrix \mathbf{A} evaluated at the equilibrium point vector \mathbf{x}_{eq} and \mathbf{u}_{eq}

\dot{u} derivatives:

$$\begin{aligned} \frac{\partial \dot{u}}{\partial u} = & \frac{\pi e \mathcal{R} C_{D_o} (\Omega - r_o) k_5}{m_T C_{L_\alpha} \Omega^2 R} + \frac{k_6}{m_T} \left(\frac{C_{L_\alpha} \kappa_3 (\Omega - r_o) \delta_{lat_o}^2}{16 \Omega^2 R} + \frac{3 C_{L_\alpha} \kappa_3 (\Omega - r_o) \delta_{lon_o}^2}{16 \Omega^2 R} + \right. \\ & \frac{C_{L_\alpha} (v_o - v_w + p_o z_{rcg}) \delta_{col_o} \delta_{lat_o}}{4 R^2 \Omega^2} + \frac{C_{L_\alpha} \kappa_3 (\Omega - r_o) \delta_{col_o}^2}{4 \Omega^2 R} + \\ & \frac{(\pi e \mathcal{R} - 2 C_{L_\alpha}) \kappa_4 \left(\frac{w_o - w_w}{R \Omega} - \frac{v_i}{R \Omega} \right) \delta_{col_o}}{4 R \Omega} - \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) q_o \delta_{lat_o}}{32 \Omega^2 R} + \\ & \left. \frac{3 \kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) p_o \delta_{lon_o}}{32 \Omega^2 R} - \frac{3 C_{L_\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) \delta_{col_o} \delta_{lon_o}}{4 R^2 \Omega^2} \right) + \\ & \frac{(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \pi e_t \mathcal{R}_t C_{D_{o_t}} (q_o + \Omega_t) k_{13}}{m_T C_{L_{\alpha_t}} \Omega_t^2 R_t} + \\ & \frac{k_{14}}{m_T} \left(\frac{(\pi e_t \mathcal{R}_t - 2 C_{L_{\alpha_t}}) \kappa_{t_4} (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{it}) \delta_{tail_o}}{2 R^2 \Omega^2 R_t \Omega_t} + \right. \\ & \left. \frac{\kappa_{t_3} C_{L_{\alpha_t}} (q_o + \Omega_t) \delta_{tail_o}^2}{4 \Omega_t^2 R_t} \right) + \frac{1}{2} \frac{\rho v h_i S_{x_{fu}}}{m_T} \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{u}}{\partial v} = & \frac{k_6}{m_T} \left(- \frac{C_{L_\alpha} \kappa_4 (p_o z_{rcg} + v_o - v_w) \delta_{col_o} \delta_{lon_o}}{4 R^2 \Omega^2} - \frac{C_{L_\alpha} \kappa_3 (\Omega - r_o) \delta_{lat_o} \delta_{lon_o}}{8 \Omega^2 R} - \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) p_o \delta_{lat_o}}{32 \Omega^2 R} + \right. \\ & \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) q_o \delta_{lon_o}}{32 \Omega^2 R} + \frac{C_{L_\alpha} (-q_o z_{rcg} + u_o - u_w) \delta_{col_o} \delta_{lat_o}}{4 R^2 \Omega^2} \right) + \frac{(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) p_o k_{13}}{m_T \Omega_t^2 R_t} + \\ & \frac{k_{14} (\pi e_t \mathcal{R}_t - 2 C_{L_{\alpha_t}})}{4 m_T} \left[\frac{\kappa_{t_4}}{\Omega_t R_t} \left(\frac{(-q_o z_{rcg} + u_o - u_w)^2}{R^2 \Omega^2} + 3 \frac{(p_o z_{rcg} + v_o - v_w)^2}{R^2 \Omega^2} \right) + \right. \\ & \left. \frac{6 \kappa_{t_4} (p_o z_{rcg} + v_o - v_w) (p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{it})}{R^2 \Omega^2 R_t \Omega_t} \right] \delta_{tail_o} + r_o \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{u}}{\partial w} = & \frac{k_6}{m_T} \left(\frac{(\pi e \mathcal{R} - 2 C_{L_\alpha}) \kappa_4 (-q_o z_{rcg} + u_o - u_w) \delta_{col_o}}{4 R^2 \Omega^2} - \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) (\Omega - r_o) \delta_{lon_o}}{8 \Omega^2 R} \right) - \\ & \frac{k_5 (-\pi e \mathcal{R} + C_{L_\alpha}) p_o}{m_T \Omega^2 R} - q_o \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{u}}{\partial p} = & - \frac{k_5 \left(-\frac{v_i}{R \Omega} + \frac{w_o - w_w}{R \Omega} \right) (-\pi e \mathcal{R} + C_{L_\alpha})}{m_T \Omega} + \frac{k_6}{m_T} \left[- \frac{C_{L_\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lat_o} \delta_{lon_o}}{8 \Omega^2 R} - \right. \\ & \left. \frac{C_{L_\alpha} \kappa_4 (p_o z_{rcg} + v_o - v_w) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4 R^2 \Omega^2} - \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) \left(\frac{p_o z_{rcg}}{\Omega^2 R} + \frac{p_o z_{rcg} + v_o - v_w}{\Omega^2 R} \right) \delta_{lat_o}}{32} \right. \\ & \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha}) \left(q_o z_{rcg} - \frac{3(u_o - u_w)}{2} \right) \delta_{lon_o}}{16 \Omega^2 R} + \frac{C_{L_\alpha} (-q_o z_{rcg} + u_o - u_w) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4 R^2 \Omega^2} \right] + \\ & \frac{k_{13}}{m_T} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left(\frac{p_o z_{tcg}}{\Omega_t^2 R_t} + \frac{p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{it}}{\Omega_t^2 R_t} \right) + \\ & \frac{k_{14} (\pi e_t \mathcal{R}_t - 2 C_{L_{\alpha_t}})}{4 m_T} \left[\left(\frac{(-q_o z_{rcg} + u_o - u_w)^2}{R^2 \Omega^2} + 3 \frac{(p_o z_{rcg} + v_o - v_w)^2}{R^2 \Omega^2} \right) \frac{\kappa_{t_4} z_{tcg}}{\Omega_t R_t} - \right. \end{aligned}$$

$$\left. \frac{(q_o + \Omega_t) \kappa_{t_2}}{3\Omega_t^2} + \frac{6(p_o z_{rcg} + v_o - v_w)(p_o z_{t_{cg}} - x_{tr_o} + v_o - v_w - v_{i_t}) \kappa_{t_4} z_{rcg}}{\Omega_t R_t \Omega^2 R^2} \right] \delta_{tail_o}$$

$$\frac{\partial \dot{u}}{\partial q} = -\frac{\pi e \mathcal{R} C_{D_o} (\Omega - r_o) z_{rcg} k_5}{m_T C_{L_\alpha} \Omega^2 R} + \frac{k_6}{m_T} \left[-\frac{3 C_{L_\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lon_o}^2}{16 \Omega^2 R} - \frac{C_{L_\alpha} z_{rcg} (p_o z_{rcg} + v_o - v_w) \delta_{col_o} \delta_{lat_o}}{4 R^2 \Omega^2} - \frac{C_{L_\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{col_o}^2}{4 \Omega^2 R} - \frac{C_{L_\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lat_o}^2}{16 \Omega^2 R} - \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha})}{32 \Omega^2 R} [(u_o - u_w) - 2 q_o z_{rcg}] \delta_{lat_o} - \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha})}{32 \Omega^2 R} [2 p_o z_{rcg} - (v_o - v_w)] \delta_{lon_o} + \frac{3 C_{L_\alpha} \kappa_4 (-q_o z_{rcg} + u_o - u_w) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4 R^2 \Omega^2} + \frac{(\pi e \mathcal{R} - 2 C_{L_\alpha})}{4} \left(-\left(-\frac{v_i}{\Omega R} + \frac{w_o - w_w}{\Omega R} \right) \frac{z_{rcg} \kappa_4}{\Omega R} - \frac{(\Omega - r_o) \kappa_2}{3 \Omega^2} \right) \delta_{col_o} \right] + \frac{k_{13} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}})}{m_T} \left[\pi \left(\frac{y_{t_{cg}} r_o}{\Omega_t^2 R_t} + \frac{-q_o z_{t_{cg}} + u_o - u_w}{\Omega_t R_t} \right) \frac{C_{D_{o_t}} e_t \mathcal{R}_t}{C_{L_{\alpha_t}} \Omega_t} - \frac{\pi (q_o + \Omega_t) C_{D_{o_t}} z_{t_{cg}} e_t \mathcal{R}_t}{C_{L_{\alpha_t}} \Omega_t^2 R_t} \right] + \left[\frac{(\pi e_t \mathcal{R}_t - 2 C_{L_{\alpha_t}})}{4} \left(-\frac{\kappa_{t_2} p_o}{3 \Omega_t^2} - \frac{2(-q_o z_{rcg} + u_o - u_w)(p_o z_{t_{cg}} - x_{t_{cg}} r_o + v_o - v_w - v_{i_t}) \kappa_{t_4} z_{rcg}}{\Omega_t R_t \Omega^2 R^2} \right) \delta_{tail_o} + \frac{\kappa_{t_3} C_{L_{\alpha_t}}}{4 \Omega_t} \left(\frac{y_{t_{cg}} r_o}{\Omega_t^2 R_t} + \frac{-q_o z_{t_{cg}} + u_o - u_w}{\Omega_t R_t} \right) \delta_{tail_o}^2 - \frac{\kappa_{t_3} C_{L_{\alpha_t}} (q_o + \Omega_t) z_{t_{cg}} \delta_{tail_o}^2}{4 \Omega_t^2 R_t} \right] \frac{k_{14}}{m_T} - w_o$$

$$\frac{\partial \dot{u}}{\partial r} = -\frac{\pi e \mathcal{R} C_{D_o} (-q_o z_{rcg} + u_o - u_w) k_5}{m_T C_{L_\alpha} \Omega^2 R} + \frac{k_6}{m_T} \left[-\frac{C_{L_\alpha} \kappa_3 (-q_o z_{rcg} + u_o - u_w) \delta_{col_o}^2}{4 \Omega^2 R} - \frac{C_{L_\alpha} \kappa_3 (-q_o z_{rcg} + u_o - u_w) \delta_{lat_o}^2}{16 \Omega^2 R} - \frac{3 C_{L_\alpha} \kappa_3 (-q_o z_{rcg} + u_o - u_w) \delta_{lon_o}^2}{16 \Omega^2 R} + \frac{C_{L_\alpha} \kappa_3 (p_o z_{rcg} + v_o - v_w) \delta_{lat_o} \delta_{lon_o}}{8 \Omega^2 R} + \frac{(\pi e \mathcal{R} - 2 C_{L_\alpha}) \kappa_2 q_o \delta_{col_o}}{12 \Omega^2} + \frac{\kappa_3 (\pi e \mathcal{R} - 2 C_{L_\alpha})}{8 \Omega} \left(-\frac{v_i}{\Omega R} + \frac{w_o - w_w}{\Omega R} \right) \delta_{lon_o} + \frac{C_{L_\alpha} \kappa_2 (\Omega - r_o) \delta_{col_o} \delta_{lon_o}}{3 \Omega^2} \right] + \frac{k_{13} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}})}{m_T} \left(-\frac{p_o x_{t_{cg}}}{\Omega_t^2 R_t} + \frac{\pi (q_o + \Omega_t) C_{D_{o_t}} y_{t_{cg}} e_t \mathcal{R}_t}{C_{L_{\alpha_t}} \Omega_t^3 R_t} \right) + \frac{k_{14}}{m_T} \left[\frac{\kappa_{t_3} C_{L_{\alpha_t}} (q_o + \Omega_t) y_{t_{cg}} \delta_{tail_o}^2}{4 \Omega_t^3 R_t} - \frac{(\pi e_t \mathcal{R}_t - 2 C_{L_{\alpha_t}}) \kappa_{t_4}}{4} \left(\frac{(-q_o z_{rcg} + u_o - u_w)^2}{\Omega^2 R^2} + 3 \frac{(p_o z_{rcg} + v_o - v_w)^2}{\Omega^2 R^2} \right) \frac{x_{t_{cg}} \delta_{tail_o}}{R_t \Omega_t} \right] + v_o$$

$$\frac{\partial \dot{u}}{\partial \phi} = 0$$

$$\frac{\partial \dot{u}}{\partial \theta} = -g \cos \theta_o$$

$$\frac{\partial \dot{u}}{\partial \psi} = 0$$

\dot{v} derivatives:

$$\frac{\partial \dot{v}}{\partial u} = \left(-\frac{C_{L\alpha} \kappa_4 (-q_0 z_{rcg} + u_o - u_w) \delta_{col_o} \delta_{lat_o}}{4\Omega^2 R^2} + \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) \delta_{lat_o} \delta_{lon_o}}{8\Omega^2 R} + \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) p_o \delta_{lat_o}}{32\Omega^2 R} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) q_o \delta_{lon_o}}{32\Omega^2 R} + \frac{C_{L\alpha} \kappa_4 (p_o z_{rcg} + v_o - v_w) \delta_{col_o} \delta_{lon_o}}{4\Omega^2 R^2} \right) \frac{k_6}{m_T} + \frac{p_o k_9}{m_T \Omega_t^2 R_t} + \frac{2k_{10} \kappa_{t4}}{m_t \Omega_t R_t} \left(\frac{y_{tcg} r_o}{\Omega_t^2 R_t} + \frac{-q_0 z_{tcg} + u_o - u_w}{R_t \Omega_t} \right) \delta_{tail_o} - r_o$$

$$\frac{\partial \dot{v}}{\partial v} = -\frac{\pi e \mathcal{R} C_{D_o} (\Omega - r_o) k_5}{m_T C_{L\alpha} \Omega^2 R} + \frac{k_6}{m_T} \left(-\frac{C_{L\alpha} \kappa_3 (\Omega - r_o) \delta_{lon_o}^2}{16\Omega^2 R} - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) \delta_{col_o}^2}{4\Omega^2 R} - \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) \delta_{lat_o}^2}{16\Omega^2 R} + \frac{3\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) p_o \delta_{lat_o}}{32\Omega^2 R} - \frac{3C_{L\alpha} \kappa_4 (p_o z_{rcg} + v_o - v_w) \delta_{col_o} \delta_{lat_o}}{4\Omega^2 R^2} + \frac{C_{L\alpha} \kappa_4 (-q_0 z_{rcg} + u_o - u_w) \delta_{col_o} \delta_{lon_o}}{4\Omega^2 R^2} - \frac{(\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_4}{4\Omega R} \left(-\frac{v_i}{\Omega R} + \frac{w_o - w_w}{\Omega R} \right) \delta_{col_o} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) p_o \delta_{lon_o}}{32\Omega^2 R} \right) - \frac{2(q_o + \Omega_t) k_9}{m_T \Omega_t^2 R_t} + \frac{\rho v_{h_i} S_{yfu}}{2m_T}$$

$$\frac{\partial \dot{v}}{\partial w} = \frac{q_o (-\pi e \mathcal{R} + C_{L\alpha}) k_5}{m_T \Omega^2 R} + \frac{k_6}{m_T} \left(-\frac{(\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_4 (p_o z_{rcg} + v_o - v_w) \delta_{col_o}}{4\Omega^2 R^2} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) (\Omega - r_o) \delta_{lat_o}}{8\Omega^2 R} \right) + \frac{r_o k_9}{m_T \Omega_t^2 R_t} + \frac{2k_{10} \kappa_{t4}}{m_T \Omega_t R_t} \left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \mathcal{X} v_i + w_o - w_w}{R_t \Omega_t} \right) \delta_{tail_o} + p_o$$

$$\frac{\partial \dot{v}}{\partial p} = -\frac{\pi e \mathcal{R} C_{D_o} (\Omega - r_o) z_{rcg} k_5}{m_T C_{L\alpha} \Omega^2 R} + \frac{k_6}{m_T} \left(-\frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{col_o}^2}{4\Omega^2 R} - \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lat_o}^2}{16\Omega^2 R} - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lon_o}^2}{16\Omega^2 R} - \frac{3C_{L\alpha} \kappa_4 (p_o z_{rcg} + v_o - v_w) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} + \frac{C_{L\alpha} \kappa_4 (-q_0 z_{rcg} + u_o - u_w) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4R^2 \Omega^2} - \frac{(\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_4 z_{rcg}}{4\Omega R} \left(-\frac{v_i}{R\Omega} + \frac{w_o - w_w}{R\Omega} \right) \delta_{col_o} + \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left(\frac{3q_o z_{rcg}}{4\Omega^2 R} + \frac{3(v_o - v_w)}{4\Omega^2 R} + \frac{-q_0 z_{rcg} + u_o - u_w}{4\Omega^2 R} \right) \delta_{lat_o} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{2p_o z_{rcg} + v_o - v_w}{\Omega^2 R} \right) \delta_{lon_o} \right) + \frac{k_9}{m_T} \left(-\frac{2(q_o + \Omega_t) z_{tcg}}{\Omega_t^2 R_t} + \frac{-q_0 z_{tcg} + u_o - u_w}{\Omega_t^2 R_t} \right) - \frac{2k_{10} \kappa_{t4}}{m_t \Omega_t^2 R_t} \left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \mathcal{X} v_i + w_o - w_w}{\Omega_t R_t} \right) y_{tcg} \delta_{tail_o} + w_o$$

$$\frac{\partial \dot{v}}{\partial q} = \frac{k_5}{m_T \Omega} \left(-\frac{v_i}{R\Omega} + \frac{w_o - w_w}{R\Omega} \right) (-\pi e \mathcal{R} + C_{L\alpha}) + \frac{k_6}{m_T} \left(\frac{(\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_2 (\Omega - r_o) \delta_{col_o}}{12\Omega^2} + \frac{C_{L\alpha} \kappa_4 (-q_0 z_{rcg} + u_o - u_w) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lat_o} \delta_{lon_o}}{8\Omega^2 R} + \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha}) p_o z_{rcg} \delta_{lat_o}}{32\Omega^2 R} \right)$$

$$\begin{aligned}
& \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L\alpha})}{32\Omega^2 R} \left(-2q_0 z_{rcg} + u_o - u_w \right) \delta_{lon_o} - \frac{C_{L\alpha} \kappa_4 (p_o z_{rcg} + v_o - v_w) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4R^2 \Omega^2} \Big) + \\
& \frac{k_9}{m_T} \left(-\frac{p_o z_{tcg}}{\Omega_t^2 R_t} + \frac{x_{tcg} r_o}{\Omega_t^2 R_t} - 2 \frac{p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{i_t}}{\Omega_t^2 R_t} \right) + \\
& \frac{k_{10}}{m_T} \left\{ \kappa_{t_4} \left[-\frac{2z_{tcg}}{\Omega_t R_t} \left(\frac{y_{tcg} r_o}{\Omega_t^2 R_t} + \frac{-q_o z_{tcg} + u_o - u_w}{\Omega_t R_t} \right) + \frac{2x_{tcg}}{\Omega_t R_t} \left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t R_t} \right) \right] + \right. \\
& \left. \frac{4(q_o + \Omega_t) \kappa_{t_2}}{3\Omega_t^2} \right\} \delta_{tail_o} \\
\frac{\partial \dot{v}}{\partial r} = & \frac{\pi e \mathcal{R} C_{D_o} (p_o z_{rcg} + v_o - v_w) k_5}{m_T C_{L\alpha} \Omega^2 R} + \frac{k_6}{m_T} \left(\frac{C_{L\alpha} \kappa_3 (p_o z_{rcg} + v_o - v_w) \delta_{col_o}^2}{4\Omega^2 R} + \right. \\
& \frac{3C_{L\alpha} \kappa_3 (p_o z_{rcg} + v_o - v_w) \delta_{lat_o}^2}{16\Omega^2 R} + \frac{C_{L\alpha} \kappa_3 (p_o z_{rcg} + v_o - v_w) \delta_{lon_o}^2}{16\Omega^2 R} + \frac{C_{L\alpha} \kappa_2 (\Omega - r_o) \delta_{col_o} \delta_{lat_o}}{3\Omega^2} - \\
& \frac{C_{L\alpha} \kappa_3 (-q_o z_{rcg} + u_o - u_w) \delta_{lat_o} \delta_{lon_o}}{8\Omega^2 R} - \frac{(\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_2 q_o \delta_{col_o}}{12\Omega^2} + \\
& \left. \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L\alpha})}{8\Omega} \left(-\frac{v_i}{\Omega R} + \frac{w_o - w_w}{\Omega R} \right) \delta_{lat_o} \right) + \frac{k_9}{m_T} \left(\frac{2(q_o + \Omega_t) x_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t^2 R_t} \right) + \\
& \frac{2k_{10} \kappa_{t_4}}{m_T \Omega_t^2 R_t} \left(\frac{y_{tcg} r_o}{\Omega_t^2 R_t} + \frac{-q_o z_{tcg} + u_o - u_w}{\Omega_t R_t} \right) y_{tcg} \delta_{tail_o} - u_o
\end{aligned}$$

$$\frac{\partial \dot{v}}{\partial \phi} = g \cos \theta_o \cos \phi_o$$

$$\frac{\partial \dot{v}}{\partial \theta} = -g \sin \theta_o \sin \phi_o$$

$$\frac{\partial \dot{v}}{\partial \psi} = 0$$

\dot{w} derivatives:

$$\frac{\partial \dot{w}}{\partial u} = -\frac{p_o k_1}{m_T \Omega^2 R} + \frac{k_2}{m_T} \left(\frac{2\kappa_4 (-q_o z_{rcg} + u_o - u_w) \delta_{col_o}}{R^2 \Omega^2} - \frac{\kappa_3 (\Omega - r_o) \delta_{lon_o}}{\Omega^2 R} \right) + q_o$$

$$\begin{aligned}
\frac{\partial \dot{w}}{\partial v} = & -\frac{q_o k_1}{m_T \Omega^2 R} + \frac{k_2}{m_T} \left(\frac{2\kappa_4 (p_o z_{rcg} + v_o - v_w) \delta_{col_o}}{R^2 \Omega^2} + \frac{\kappa_3 (\Omega - r_o) \delta_{lat_o}}{\Omega^2 R} \right) - \\
& \frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) r_o k_{13}}{m_T \Omega_t^2 R_t} - \frac{k_{14} (\pi \mathcal{R}_t r_t - 2C_{L\alpha_t}) \kappa_{t_4}}{4m_T \Omega_t R_t} \left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t R_t} \right) \delta_{tail_o} - p_o
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \dot{w}}{\partial w} = & -\frac{2(\Omega - r_o) k_1}{m_T \Omega^2 R} + \frac{\pi e_t \mathcal{R}_t C_{D_o_t} (q_o + \Omega_t) k_{13}}{m_T C_{L\alpha_t} \Omega_t^2 R_t} + \\
& \frac{k_{14}}{m_T} \left(-\frac{\kappa_{t_3} C_{L\alpha_t} (q_o + \Omega_t) \delta_{tail_o}^2}{4\Omega_t^2 R_t} - \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \kappa_{t_4} (p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{i_t}) \delta_{tail_o}}{4R_t^2 \Omega_t^2} \right)
\end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{w}}{\partial p} = & -\frac{(u_o - u_w)k_1}{m_T \Omega^2 R} + \frac{k_2}{m_T} \left(\frac{2\kappa_4(p_o z_{rcg} + v_o - v_w)z_{rcg} \delta_{col_o}}{R^2 \Omega^2} + \frac{\kappa_3(\Omega - r_o)z_{rcg} \delta_{lat_o}}{\Omega^2 R} \right) + \\ & \frac{k_{13}}{m_T} \left(-\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t})z_{tcg} r_o}{\Omega_t^2 R_t} - \frac{\pi(q_o + \Omega_t)C_{D_{ot}} y_{tcg} e_t \mathcal{R}_t}{C_{L\alpha_t} \Omega_t^3 R_t} \right) + \\ & \left\{ \frac{\kappa_{t3} C_{L\alpha_t} (q_o + \Omega_t) y_{tcg} \delta_{tail_o}^2}{4\Omega_t^3 R_t} + \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left[\left(\frac{p_o y_{tcg}}{\Omega_t^2 R_t} - \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t R_t} \right) \frac{\kappa_{t4} z_{tcg}}{\Omega_t R_t} + \right. \right. \\ & \left. \left. \frac{(p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{it}) \kappa_{t4} y_{tcg}}{\Omega_t^3 R_t^2} \right] \delta_{tail_o} \right\} \frac{k_{14}}{m_T} - v_o \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{w}}{\partial q} = & -\frac{(v_o - v_w)k_1}{m_T \Omega^2 R} + \frac{k_2}{m_T} \left(\frac{-2\kappa_4(-q_o z_{rcg} + u_o - u_w)z_{rcg} \delta_{col_o}}{R^2 \Omega^2} + \frac{\kappa_3(\Omega - r_o)z_{rcg} \delta_{lon_o}}{\Omega^2 R} \right) + \\ & \frac{k_{13}}{m_T} \left[\left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t R_t} \right) \frac{\pi C_{D_{ot}} e_t \mathcal{R}_t}{C_{L\alpha_t} \Omega_t} + \frac{\pi(q_o + \Omega_t)C_{D_{ot}} x_{tcg} e_t \mathcal{R}_t}{C_{L\alpha_t} \Omega_t^2 R_t} \right] + \\ & \frac{k_{14}}{m_T} \left[-\frac{\kappa_{t3} C_{L\alpha_t}}{4\Omega} \left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t R_t} \right) \delta_{tail_o}^2 - \frac{\kappa_{t3} C_{L\alpha_t} (q_o + \Omega_t) x_{tcg} \delta_{tail_o}^2}{4\Omega_t^2 R_t} + \right. \\ & \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(\frac{\kappa_{t2} r_o}{3\Omega_t^2} - \frac{(p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{it}) \kappa_{t4} x_{tcg}}{\Omega_t^2 R_t^2} \right) \delta_{tail_o} \right] + u_o \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{w}}{\partial r} = & \frac{2k_1}{m_T \Omega} \left(-\frac{v_i}{\Omega R} + \frac{w_o - w_w}{\Omega R} \right) + \frac{k_2}{m_T} \left(-\frac{4\kappa_2(\Omega - r_o)\delta_{col_o}}{3\Omega^2} - \frac{\kappa_3(p_o z_{rcg} + v_o - v_w)\delta_{lat_o}}{\Omega^2 R} + \right. \\ & \left. \frac{\kappa_3(-q_o z_{rcg} + u_o - u_w)\delta_{lon_o}}{\Omega^2 R} \right) + \frac{k_{13}}{m_T} \left(-\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t})(p_o z_{tcg} - x_{tcg} r_o + v_o - v_w - v_{it})}{\Omega_t^2 R_t} + \right. \\ & \left. \frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t})x_{tcg} r_o}{\Omega_t^2 R_t} \right) + \\ & \frac{k_{14}(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4m_T} \left[\left(-\frac{p_o y_{tcg}}{\Omega_t^2 R_t} + \frac{q_o x_{tcg} - \varkappa v_i + w_o - w_w}{\Omega_t R_t} \right) \frac{\kappa_{t4} x_{tcg}}{\Omega_t R_t} + \frac{(q_o + \Omega_t)\kappa_{t2}}{3\Omega_t^2} \right] \delta_{tail_o} \end{aligned}$$

$$\frac{\partial \dot{w}}{\partial \phi} = -g \cos \theta_o \sin \phi_o$$

$$\frac{\partial \dot{w}}{\partial \theta} = -g \sin \theta_o \cos \phi_o$$

$$\frac{\partial \dot{w}}{\partial \psi} = 0$$

\dot{p} derivatives:

$$\begin{aligned} \frac{\partial \dot{p}}{\partial u} = & \frac{k_6 C_{L\alpha} (\Omega - r_o) z_{rcg} \delta_{lat_o} \delta_{lon_o} k_3}{4(2I_x + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} + \frac{2k_6}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \left(\frac{C_{L\alpha} (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4\kappa_4 R^2 \Omega^2} - \right. \\ & \left. \frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} + \frac{\kappa_3 (\pi e_t \mathcal{R}_t - 2C_{L\alpha}) p_o z_{rcg} \delta_{lat_o}}{32\Omega^2 R} - \frac{\kappa_3 (\pi e_t \mathcal{R}_t - 2C_{L\alpha}) q_o z_{rcg} \delta_{lon_o}}{32\Omega^2 R} \right) - \\ & \frac{2k_7 \kappa_3}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) R \Omega} \left(\frac{w_o - w_w}{R \Omega} - \frac{v_i}{R \Omega} \right) + \end{aligned}$$

$$\begin{aligned}
& \frac{2k_8}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{\kappa_2(\Omega - r_o)\delta_{col_o}}{\Omega^2 R} + \frac{3\kappa_3(v_o - v_w + p_o z_{rcg})\delta_{lat_o}}{8R^2\Omega^2} - \frac{9\kappa_3(u_o - u_w - q_o z_{rcg})\delta_{lon_o}}{8R^2\Omega^2} \right) + \\
& \frac{2p_o z_{tcg} k_9}{(2I_x + I_{x_{bl}} + I_{y_{bl}})\Omega_t^2 R_t} + \frac{4k_{10}\kappa_{t_4}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})R_t\Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t\Omega_t} + \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) z_{tcg} \delta_{tail_o} \\
\frac{\partial \dot{p}}{\partial v} = & -2 \frac{\pi e \mathcal{R} C_{D_o}(\Omega - r_o) z_{rcg} k_5}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) C_{L_\alpha} \Omega^2 R} + \frac{2k_6}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{C_{L_\alpha} \kappa_3(\Omega - r_o) z_{rcg} \delta_{col_o}^2}{4\Omega^2 R} - \right. \\
& \frac{C_{L_\alpha} \kappa_3(\Omega - r_o) z_{rcg} \delta_{lon_o}^2}{3\Omega^2 R} - \frac{3C_{L_\alpha} \kappa_4(v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4R^2\Omega^2} - \frac{3C_{L_\alpha} \kappa_3(\Omega - r_o) z_{rcg} \delta_{lat_o}^2}{16\Omega^2 R} - \\
& \frac{\kappa_4(\pi e \mathcal{R} - 2C_{L_\alpha})}{4\Omega R} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{rcg} \delta_{col_o} + \frac{3\kappa_3(\pi e \mathcal{R} - 2C_{L_\alpha}) q_o z_{rcg} \delta_{lat_o}}{32\Omega^2 R} - \\
& \left. \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L_\alpha}) p_o z_{rcg} \delta_{lon_o}}{32\Omega^2 R} + \frac{C_{L_\alpha}(u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4\kappa_4 R^2 \Omega^2} \right) + \\
& \frac{6k_8}{8(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{\kappa_3(u_o - u_w - q_o z_{rcg})\delta_{lat_o}}{R^2\Omega^2} - \frac{\kappa_3(v_o - v_w + p_o z_{rcg})\delta_{lon_o}}{R^2\Omega^2} \right) - \\
& \frac{4(q_o + \Omega_t) z_{tcg} k_9}{(2I_x + I_{x_{bl}} + I_{y_{bl}})\Omega_t^2 R_t} + \frac{2y_{tcg}(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) r_o k_{13}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})\Omega_t^2 R_t} + \\
& \frac{k_{14}\kappa_{t_4}(\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}})}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})R_t\Omega_t} \left(\frac{w_o - w_w + q_o x_{tcg} - k v_i}{R_t\Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) y_{tcg} \delta_{tail_o} \\
\frac{\partial \dot{p}}{\partial w} = & 2 \frac{q_o z_{rcg} (C_{L_\alpha} - \pi e \mathcal{R}) k_5}{(2I_x + I_{x_{bl}} + I_{y_{bl}})\Omega^2 R} + \\
& \frac{-2k_6}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{(\pi e \mathcal{R} - 2C_{L_\alpha}) \kappa_4(v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o}}{4R^2\Omega^2} + \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L_\alpha})(\Omega - r_o) z_{rcg} \delta_{lat_o}}{8\Omega^2 R} \right) - \\
& \frac{2\kappa_3(u_o - u_w - q_o z_{rcg}) k_7}{(2I_x + I_{x_{bl}} + I_{y_{bl}})R^2\Omega^2} + \frac{2z_{tcg} r_o k_9}{(2I_x + I_{x_{bl}} + I_{y_{bl}})\Omega_t^2 R_t} + \\
& \frac{4k_{10}\kappa_{t_4}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})R_t\Omega_t} \left(\frac{w_o - w_w + q_o x_{tcg} - k v_i}{R_t\Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) z_{tcg} \delta_{tail_o} - \frac{2y_{tcg} \pi e_t \mathcal{R}_t C_{D_{o_t}}(q_o + \Omega_t) k_{13}}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) C_{L_{\alpha_t}} \Omega_t^2 R_t} + \\
& \frac{k_{14}}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \left(\frac{(\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \kappa_{t_4}(v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_i t) y_{tcg} \delta_{tail_o}}{2R_t^2 \Omega_t^2} + \frac{C_{L_{\alpha_t}} \kappa_{t_3}(q_o + \Omega_t) y_{tcg} \delta_{tail_o}^2}{2\Omega_t^2 R_t} \right) \\
\frac{\partial \dot{p}}{\partial p} = & -2 \frac{z_{rcg}^2 \pi e \mathcal{R} C_{D_o}(\Omega - r_o) k_5}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) C_{L_\alpha} \Omega^2 R} + \frac{2k_6}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{C_{L_\alpha} \kappa_3(\Omega - r_o) z_{rcg}^2 \delta_{col_o}^2}{4\Omega^2 R} - \right. \\
& \frac{3C_{L_\alpha} \kappa_4(v_o - v_w + p_o z_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lat_o}}{4R^2\Omega^2} - \frac{3C_{L_\alpha} \kappa_3(\Omega - r_o) z_{rcg}^2 \delta_{lat_o}^2}{16\Omega^2 R} - \\
& \frac{C_{L_\alpha} \kappa_3(\Omega - r_o) z_{rcg}^2 \delta_{lon_o}^2}{3\Omega^2 R} + \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L_\alpha})}{8} \left(\frac{u_o - u_w - q_o z_{rcg}}{4\Omega^2 R} + \frac{3q_o z_{rcg}}{4\Omega^2 R} \right) z_{rcg} \delta_{lat_o} - \\
& \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L_\alpha})}{32} \left(\frac{v_o - v_w + p_o z_{rcg}}{\Omega^2 R} + \frac{p_o z_{rcg}}{\Omega^2 R} \right) z_{rcg} \delta_{lon_o} + \frac{C_{L_\alpha}(u_o - u_w - q_o z_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lon_o}}{4\kappa_4 R^2 \Omega^2} - \\
& \left. \frac{(\pi e \mathcal{R} - 2C_{L_\alpha}) \kappa_4 z_{rcg}^2}{4R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \delta_{col_o} \right) - \frac{\kappa_1(\Omega - r_o) k_7}{(2I_x + I_{x_{bl}} + I_{y_{bl}})\Omega^2} + \\
& \frac{6k_8}{8(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{\kappa_3(u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lat_o}}{R^2\Omega^2} - \frac{\kappa_3(v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lon_o}}{R^2\Omega^2} \right) +
\end{aligned}$$

$$\begin{aligned}
& \frac{2z_{tcg}k_9}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left(\frac{u_o-u_w-q_oz_{tcg}}{\Omega_t^2 R_t} - 2 \frac{(q_o+\Omega_t)z_{tcg}}{\Omega_t^2 R_t} \right) - \\
& \frac{4k_{10}\kappa_{t4}}{(2I_x+I_{x_{bl}}+I_{y_{bl}})\Omega_t^2 R_t} \left(\frac{w_o-w_w+q_ox_{tcg}-\varkappa v_i}{R_t\Omega_t} - \frac{p_oy_{tcg}}{\Omega_t^2 R_t} \right) y_{tcg} z_{tcg} \delta_{tail_o} + \\
& \frac{2y_{tcg}k_{13}}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left(\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t})r_oz_{tcg}}{\Omega_t^2 R_t} + \frac{\pi e_t \mathcal{R}_t C_{D_o t} (q_o+\Omega_t)y_{tcg}}{C_{L\alpha_t} \Omega_t^3 R_t} \right) + \\
& \frac{2k_{14}}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left[\frac{C_{L\alpha_t} \kappa_{t3} y_{tcg}^2 (q_o+\Omega_t) \delta_{tail_o}^2}{4\Omega_t^3 R_t} - \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(\frac{\kappa_{t4} y_{tcg} (v_o-v_w+p_oz_{tcg}-r_ox_{tcg}-v_{it})}{\Omega_t^3 R_t^2} \right. \right. \\
& \left. \left. \frac{\kappa_{t4}}{R_t\Omega_t} \left(\frac{w_o-w_w+q_ox_{tcg}-kv_i}{R_t\Omega_t} - \frac{p_oy_{tcg}}{\Omega_t^2 R_t} \right) z_{tcg} \right) y_{tcg} \delta_{tail_o} \right] \\
\frac{\partial \dot{p}}{\partial q} = & -\frac{k_6 C_{L\alpha} (\Omega - r_o) z_{rcg}^2 \delta_{lat_o} \delta_{lon_o} k_3}{4(2I_x+I_{x_{bl}}+I_{y_{bl}})\Omega^2 R} + \frac{2k_5 z_{rcg} (C_{L\alpha} - \pi e \mathcal{R})}{(2I_x+I_{x_{bl}}+I_{y_{bl}})\Omega} \left(\frac{w_o-w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) + \\
& \frac{2k_6}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left(\frac{(\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_2 (\Omega - r_o) z_{rcg} \delta_{col_o}}{12\Omega^2} - \frac{C_{L\alpha} z_{rcg}^2 (v_o-v_w+p_oz_{rcg}) \delta_{col_o} \delta_{lon_o}}{4\kappa_4 R^2 \Omega^2} + \right. \\
& \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left(\frac{3v_o-v_w}{4\Omega^2 R} + \frac{p_oz_{rcg}}{2\Omega^2 R} \right) z_{rcg} \delta_{lat_o} - \right. \\
& \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{u_o-u_w-q_oz_{rcg}}{\Omega^2 R} - \frac{q_oz_{rcg}}{\Omega^2 R} \right) z_{rcg} \delta_{lon_o} + \frac{C_{L\alpha} \kappa_4 (u_o-u_w-q_oz_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} \right) + \\
& \frac{2\kappa_3}{\kappa_7 (2I_x+I_{x_{bl}}+I_{y_{bl}})\Omega R} \left(\frac{w_o-w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{rcg} + \\
& \frac{2k_8}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left(\frac{\kappa_2 (\Omega - r_o) z_{rcg} \delta_{col_o}}{\Omega^2 R} - \frac{3\kappa_3 (v_o-v_w+p_oz_{rcg}) z_{rcg} \delta_{lat_o}}{8R^2 \Omega^2} + \frac{9\kappa_3 (u_o-u_w-q_oz_{rcg}) z_{rcg} \delta_{lon_o}}{8R^2 \Omega^2} \right) + \\
& \frac{2z_{tcg}k_9}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left(-\frac{p_oz_{tcg}}{\Omega_t^2 R_t} - 2 \frac{v_o-v_w+p_oz_{tcg}-r_ox_{tcg}-v_{it}}{\Omega_t^2 R_t} + \frac{r_ox_{tcg}}{\Omega_t^2 R_t} \right) + \\
& \frac{2k_{10}}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left\{ \frac{4\kappa_{t2} (q_o+\Omega_t)}{3\Omega_t^2} + \kappa_{t4} \left[-\frac{2}{R_t\Omega_t} \left(\frac{u_o-u_w-q_oz_{tcg}}{R_t\Omega_t} + \frac{r_oy_{tcg}}{\Omega_t^2 R_t} \right) z_{tcg} + \right. \right. \\
& \left. \left. \frac{2}{R_t\Omega_t} \left(\frac{w_o-w_w+q_ox_{tcg}-\varkappa v_i}{R_t\Omega_t} - \frac{p_oy_{tcg}}{\Omega_t^2 R_t} \right) x_{tcg} \right] \right\} z_{tcg} \delta_{tail_o} + \\
& \frac{2y_{tcg}k_{13}}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left[\frac{-\pi e_t \mathcal{R}_t C_{D_o t}}{C_{L\alpha_t} \Omega_t} \left(\frac{w_o-w_w+q_ox_{tcg}-\varkappa v_i}{R_t\Omega_t} - \frac{p_oy_{tcg}}{\Omega_t^2 R_t} \right) - \frac{\pi e_t \mathcal{R}_t C_{D_o t} (q_o+\Omega_t)x_{tcg}}{C_{L\alpha_t} t\Omega_t^2 R_t} \right] + \\
& \frac{2k_{14}}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left[\frac{C_{L\alpha_t} \kappa_{t3}}{4\Omega_t} \left(\frac{w_o-w_w+q_ox_{tcg}-kv_i}{R_t\Omega_t} + \frac{p_oy_{tcg}}{\Omega_t^2 R_t} \right) y_{tcg} \delta_{tail_o}^2 + \frac{C_{L\alpha_t} \kappa_{t3} x_{tcg} (q_o+\Omega_t) y_{tcg} \delta_{tail_o}^2}{4\Omega_t^2 R_t} - \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(\frac{\kappa_{t2} r_o}{3\Omega_t^2} - \frac{\kappa_{t4} x_{tcg} (v_o-v_w+p_oz_{tcg}-r_ox_{tcg}-v_{it})}{R_t^2 \Omega_t^2} \right) y_{tcg} \delta_{tail_o} \right] + \\
& 2 \frac{I_{z_{bl}} \Omega - (I_z + I_{z_{bl}}) r_o + r_o \left(I_y + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) + \alpha_2 \cos(\phi_o)}{2I_x+I_{x_{bl}}+I_{y_{bl}}} \\
\frac{\partial \dot{p}}{\partial r} = & -\frac{k_6 \kappa_3 C_{L\alpha} (u_o-u_w-q_oz_{rcg}) z_{rcg} \delta_{lat_o} \delta_{lon_o}}{4(2I_x+I_{x_{bl}}+I_{y_{bl}})\Omega^2 R} + \frac{2z_{rcg} \pi e \mathcal{R} C_{D_o} (v_o-v_w+p_oz_{rcg}) k_5}{(2I_x+I_{x_{bl}}+I_{y_{bl}}) C_{L\alpha} \Omega^2 R} + \\
& \frac{2k_6}{(2I_x+I_{x_{bl}}+I_{y_{bl}})} \left[\frac{C_{L\alpha} \kappa_3 (v_o-v_w+p_oz_{rcg}) z_{rcg} \delta_{lon_o}^2}{3\Omega^2 R} + \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8\Omega} \left(\frac{w_o-w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{rcg} \delta_{lat_o} + \right.
\end{aligned}$$

$$\begin{aligned}
& \frac{C_{L\alpha} \kappa_2 (\Omega - r_o) z_{rcg} \delta_{col_o} \delta_{lat_o}}{3\Omega^2} + \frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o}^2}{4\Omega^2 R} + \\
& \left. \frac{3C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lat_o}^2}{16\Omega^2 R} - \frac{(\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_2 q_o z_{rcg} \delta_{col_o}}{12\Omega^2} \right] + \\
& \frac{\kappa_1 p_o k_7}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) \Omega^2} + \frac{2k_8}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{\kappa_2 (u_o - u_w - q_o z_{rcg}) \delta_{col_o}}{\Omega^2 R} + \frac{3\kappa_1 (\Omega - r_o) \delta_{lon_o}}{4\Omega^2} \right) + \\
& \frac{2k_9 z_{t_{cg}}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(2 \frac{(q_o + \Omega_t) x_{t_{cg}}}{\Omega_t^2 R_t} + \frac{w_o - w_w + q_o x_{t_{cg}} - \mathcal{X} v_i}{\Omega_t^2 R_t} \right) + \\
& \frac{4k_{10} \kappa_{t_4}}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) \Omega_t^2 R_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) y_{t_{cg}} z_{t_{cg}} \delta_{tail_o} + \\
& \frac{2y_{t_{cg}} k_{13}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t})}{\Omega_t^2 R_t} - \frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) r_o x_{t_{cg}}}{\Omega_t^2 R_t} \right) - \\
& \frac{k_{14} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{2(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{\kappa_{t_2} (q_o + \Omega_t)}{3\Omega_t^2} + \frac{\kappa_{t_4}}{R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \mathcal{X} v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) x_{t_{cg}} \right] y_{t_{cg}} \delta_{tail_o} + \\
& 2 \frac{-\alpha_2 \sin(\phi_o) + \left(I_y + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) q_o - q_o (I_z + I_{z_{bl}})}{2I_x + I_{x_{bl}} + I_{y_{bl}}}
\end{aligned}$$

$$\frac{\partial \dot{p}}{\partial \phi} = - \frac{2\alpha_2 (q_o \sin \phi_o + r_o \cos \phi_o)}{2I_x + I_{x_{bl}} + I_{y_{bl}}}$$

$$\frac{\partial \dot{p}}{\partial \theta} = 0$$

$$\frac{\partial \dot{p}}{\partial \psi} = 0$$

\dot{q} derivatives:

$$\begin{aligned}
\frac{\partial \dot{q}}{\partial u} = & -2 \frac{\pi e_t \mathcal{R} C_{D_o} (\Omega - r_o) z_{rcg} k_5}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L\alpha} \Omega^2 R} + \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(- \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{col_o}^2}{4\Omega^2 R} - \right. \\
& \left. \frac{C_{L\alpha} \kappa_4 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} - \frac{(\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_4}{4R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{rcg} \delta_{col_o} - \right. \\
& \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lon_o}^2}{16\Omega^2 R} + \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha}) q_o z_{rcg} \delta_{lat_o}}{32\Omega^2 R} + \\
& \left. \frac{3C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lon_o}^2}{4R\Omega^2} - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lat_o}^2}{16\Omega^2 R} \right) + \\
& \frac{2k_8}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(- \frac{\kappa_2 (\Omega - r_o) \delta_{col_o}}{\Omega^2 R} + \frac{3\kappa_3 (u_o - u_w - q_o z_{rcg}) \delta_{lat_o}}{8R^2 \Omega^2} + \frac{3\kappa_3 (v_o - v_w + p_o z_{rcg}) \delta_{lon_o}}{8R^2 \Omega^2} \right) - \\
& \frac{2\pi e_t \mathcal{R}_t C_{D_{o_t}} k_{11} \kappa_{t_3}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L\alpha_t} R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) + \\
& \frac{2k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(- \frac{\kappa_{t_2} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) p_o \delta_{tail_o}}{12\Omega_t^2 R_t} - \frac{C_{L\alpha_t} \kappa_{t_3}}{4R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) \delta_{tail_o}^2 \right) - \\
& 2 \frac{z_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \pi e_t \mathcal{R}_t C_{D_{o_t}} (q_o + \Omega_t) k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L\alpha_t} \Omega_t^2 R_t} +
\end{aligned}$$

$$\begin{aligned}
& \frac{k_{14}}{2(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{C_{L\alpha_t} \kappa_{t3} (q_o + \Omega_t) z_{tcg} \delta_{tail_o}^2}{\Omega_t^2 R_t} - \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \kappa_{t4} (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_i_t) z_{tcg} \delta_{tail_o}}{R_t^2 \Omega_t^2} \right) - \frac{3\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha}) p_o z_{rcg} \delta_{lon_o}}{16(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} \\
\frac{\partial \dot{q}}{\partial v} = & \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} + \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha}) p_o z_{rcg} \delta_{lat_o}}{32\Omega^2 R} + \right. \\
& \left. \frac{C_{L\alpha} \kappa_4 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o} \delta_{lon_o}}{4R^2 \Omega^2} + \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg} \delta_{lat_o} \delta_{lon_o}}{8\Omega^2 R} \right) + \\
& \frac{2k_7}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) R \Omega} \kappa_3 \left(\frac{w_o - w_w}{R \Omega} - \frac{v_i}{R \Omega} \right) + \\
& \frac{2k_8}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{3\kappa_3 (u_o - u_w - q_o z_{rcg}) \delta_{lon_o}}{8R^2 \Omega^2} + \frac{9\kappa_3 (v_o - v_w + p_o z_{rcg}) \delta_{lat_o}}{8R^2 \Omega^2} \right) + \\
& 4 \frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \kappa_{t3} (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_i_t) k_{11}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) R_t^2 \Omega_t^2} + \\
& 1/3 \frac{\kappa_{t2} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) (q_o + \Omega_t) \delta_{tail_o} k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega_t^2 R_t} + \\
& \frac{2k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) r_o x_{tcg}}{\Omega_t^2 R_t} - \frac{z_{tcg} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) p_o}{\Omega_t^2 R_t} \right) + \\
& \frac{2k_{14}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[-\frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \kappa_{t4}}{4R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{tcg} - k v_i}{R_t \Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) x_{tcg} - \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \kappa_{t4}}{4R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} - \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) z_{tcg} \right] \delta_{tail_o} - \\
& \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha}) q_o z_{rcg} \delta_{lon_o}}{16(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} \\
\frac{\partial \dot{q}}{\partial w} = & 2 \frac{p_o z_{rcg} (C_{L\alpha} - \pi e_t \mathcal{R}) k_5}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} - \frac{(\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_4 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{col_o} k_6}{2(2I_y + I_{x_{bl}} + I_{y_{bl}}) R^2 \Omega^2} + \\
& 2 \frac{\kappa_3 (v_o - v_w + p_o z_{rcg}) k_7}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) R^2 \Omega^2} - \frac{k_{11} 2\pi e_t \mathcal{R}_t C_{D_{ot}} \kappa_{t3}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L\alpha_t} R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{tcg} - k v_i}{R_t \Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) + \\
& \frac{2k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{C_{L\alpha_t} \kappa_{t3}}{4R_t \Omega_t} \left(\frac{p_o y_{tcg}}{\Omega_t^2 R_t} - \frac{w_o - w_w + q_o x_{tcg} - k v_i}{R_t \Omega_t} \right) \delta_{tail_o}^2 - \frac{\kappa_{t2} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) r_o \delta_{tail_o}}{12\Omega_t^2 R_t} \right) + \\
& 2 \frac{x_{tcg} \pi e_t \mathcal{R}_t C_{D_{ot}} (q_o + \Omega_t) k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L\alpha_t} \Omega_t^2 R_t} + \frac{2k_{14}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{C_{L\alpha_t} \kappa_{t3} (q_o + \Omega_t) x_{tcg} \delta_{tail_o}^2}{4\Omega_t^2 R_t} - \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \kappa_{t4} (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_i_t) x_{tcg} \delta_{tail_o}}{4R_t^2 \Omega_t^2} \right) + \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha}) (\Omega - r_o) z_{rcg} \delta_{lon_o}}{4(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} \\
\frac{\partial \dot{q}}{\partial p} = & \frac{2k_5 z_{rcg}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega} \left(\frac{w_o - w_w}{R \Omega} - \frac{v_i}{R \Omega} \right) (C_{L\alpha} - \pi e_t \mathcal{R}) + \\
& 2k_6 (2I_y + I_{x_{bl}} + I_{y_{bl}}) \left(-\frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} + \right. \\
& \left. \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{v_o - v_w + p_o z_{rcg}}{\Omega^2 R} + \frac{p_o z_{rcg}}{\Omega^2 R} \right) z_{rcg} \delta_{lat_o} + \frac{C_{L\alpha} \kappa_4 (v_o - v_w + p_o z_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lon_o}}{4R^2 \Omega^2} + \right.
\end{aligned}$$

$$\begin{aligned}
& \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg}^2 \delta_{lat_o} \delta_{lon_o}}{8\Omega^2 R} \Big) + \frac{2k_7 \kappa_3}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) R \Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{rcg} + \\
& \frac{2k_8}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{9\kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lat_o}}{8R^2 \Omega^2} + \frac{3\kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lon_o}}{8R^2 \Omega^2} \right) + \\
& \frac{2k_{11}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[- (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \left(\frac{\kappa_{t_1} p_o}{2\Omega_t^2} - 2 \frac{\kappa_{t_3} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{it}) z_{t_{cg}}}{R_t^2 \Omega_t^2} \right) \right] + \\
& \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} \kappa_{t_3}}{C_{L\alpha_t} \Omega_t^2 R_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) y_{t_{cg}} \Big] + \\
& \frac{2k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{C_{L\alpha_t} \kappa_{t_3}}{4\Omega_t^2 R_t} \left(\frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} - \frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} \right) y_{t_{cg}} \delta_{tail_o}^2 + \right. \\
& \left. \frac{\kappa_{t_2}}{6} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \left(\frac{(q_o + \Omega_t) z_{t_{cg}}}{\Omega_t^2 R_t} - 1/2 \frac{u_o - u_w - q_o z_{t_{cg}}}{\Omega_t^2 R_t} \right) \delta_{tail_o} \right] + \\
& \frac{2k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[-x_{t_{cg}} \left(\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) r_o z_{t_{cg}}}{\Omega_t^2 R_t} + \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} (q_o + \Omega_t) y_{t_{cg}}}{C_{L\alpha_t} \Omega_t^3 R_t} \right) - \right. \\
& \left. z_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \left(\frac{v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{it}}{\Omega_t^2 R_t} + \frac{p_o z_{t_{cg}}}{\Omega_t^2 R_t} \right) \right] + \\
& \frac{2k_{14}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ \left(\frac{\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}}{4} \right) \left[\frac{\kappa_{t_4} y_{t_{cg}} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{it})}{\Omega_t^3 R_t^2} - \right. \right. \\
& \left. \frac{\kappa_{t_4}}{R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) z_{t_{cg}} \right] x_{t_{cg}} - \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left[\frac{\kappa_{t_4}}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} - \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) z_{t_{cg}} - \frac{\kappa_{t_2} (q_o + \Omega_t)}{3\Omega_t^2} \right] z_{t_{cg}} \right\} \delta_{tail_o} - \\
& \frac{C_{L\alpha_t} \kappa_{t_3} (q_o + \Omega_t) y_{t_{cg}} x_{t_{cg}} \delta_{tail_o}^2}{4\Omega_t^3 R_t} \Big\} + \frac{2}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[-I_{z_{bl}} \Omega - r_o \left(I_x + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) + \right. \\
& \left. (I_z + I_{z_{bl}}) r_o - \alpha_4 + \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha})}{8} \left(-3/4 \frac{u_o - u_w}{\Omega^2 R} + 1/2 \frac{q_o z_{rcg}}{\Omega^2 R} \right) z_{rcg} \delta_{lon_o} \right] \\
\frac{\partial \dot{q}}{\partial q} = & 2 \frac{z_{rcg}^2 \pi e_t \mathcal{R} C_{D\alpha} (\Omega - r_o) k_5}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L\alpha} \Omega^2 R} + \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg}^2 \delta_{col_o}^2}{4\Omega^2 R} - \right. \\
& \frac{3C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lon_o}}{4R^2 \Omega^2} + \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg}^2 \delta_{lon_o}^2}{16\Omega^2 R} + \\
& \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{u_o - u_w - q_o z_{rcg}}{\Omega^2 R} - \frac{q_o z_{rcg}}{\Omega^2 R} \right) z_{rcg} \delta_{lat_o} + \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) z_{rcg}^2 \delta_{lat_o}^2}{16\Omega^2 R} - \\
& \frac{(\pi e_t \mathcal{R} - 2C_{L\alpha})}{4} \left(-\frac{\kappa_2 (\Omega - r_o)}{3\Omega^2} - \frac{\kappa_4 z_{rcg}}{R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \right) z_{rcg} \delta_{col_o} + \\
& \frac{C_{L\alpha} \kappa_4 (v_o - v_w + p_o z_{rcg}) z_{rcg}^2 \delta_{col_o} \delta_{lat_o}}{4R^2 \Omega^2} \Big] + \frac{\kappa_1 (\Omega - r_o) k_7}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2} + \\
& \frac{2k_8}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{\kappa_2 (\Omega - r_o) z_{rcg} \delta_{col_o}}{\Omega^2 R} - \frac{3\kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lat_o}}{8R^2 \Omega^2} - \right. \\
& \left. \frac{3\kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lon_o}}{8R^2 \Omega^2} \right) -
\end{aligned}$$

$$\begin{aligned}
& \frac{k_{11}\pi e_t \mathcal{R}_t C_{D_{O_t}}}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L_{\alpha_t}}} \left\{ 2 \frac{\kappa_{t_1}(q_o + \Omega_t)}{\Omega_t^2} + \kappa_{t_3} \left[-\frac{2}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) z_{t_{cg}} + \right. \right. \\
& \left. \left. \frac{2}{R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) x_{t_{cg}} \right] \right\} + \\
& 2 \left\{ \frac{\kappa_{t_2}}{16} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left(\frac{v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t}}{\Omega_t^2 R_t} - \frac{r_o x_{t_{cg}}}{2\Omega_t^2 R_t} + \frac{p_o z_{t_{cg}}}{2\Omega_t^2 R_t} \right) \delta_{tail_o} - \right. \\
& \left. \frac{C_{L_{\alpha_t}}}{8} \left[2 \frac{\kappa_{t_1}(q_o + \Omega_t)}{\Omega_t^2} + \kappa_{t_3} \left(-\frac{2}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) z_{t_{cg}} - \right. \right. \right. \\
& \left. \left. \frac{2}{R_t \Omega_t} \left(\frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} - \frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} \right) x_{t_{cg}} \right] \delta_{tail_o}^2 \right\} \frac{k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \\
& 2 \left\{ -x_{t_{cg}} \left[\frac{-\pi e_t \mathcal{R}_t C_{D_{O_t}}}{C_{L_{\alpha_t}} \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) - \frac{\pi e_t \mathcal{R}_t C_{D_{O_t}} (q_o + \Omega_t) x_{t_{cg}}}{C_{L_{\alpha_t}} \Omega_t^2 R_t} \right] - \right. \\
& z_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left[\frac{\pi e_t \mathcal{R}_t C_{D_{O_t}}}{C_{L_{\alpha_t}} \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) - \right. \\
& \left. \left. \frac{\pi e_t \mathcal{R}_t C_{D_{O_t}} (q_o + \Omega_t) z_{t_{cg}}}{C_{L_{\alpha_t}} \Omega_t^2 R_t} \right] \right\} \frac{k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \\
& 2 \left\{ \left[\frac{(\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}})}{4} \left(\frac{\kappa_{t_2} r_o}{3\Omega_t^2} - \frac{\kappa_{t_4} x_{t_{cg}} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t})}{R_t^2 \Omega_t^2} \right) x_{t_{cg}} - \right. \right. \\
& \left. \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}})}{4} \left(-\frac{\kappa_{t_4} z_{t_{cg}} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t})}{R_t^2 \Omega_t^2} - \frac{\kappa_{t_2} p_o}{3\Omega_t^2} \right) z_{t_{cg}} \right] \delta_{tail_o} + \right. \\
& \left[-\frac{C_{L_{\alpha_t}} \kappa_{t_3}}{4\Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} + \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) x_{t_{cg}} - \frac{C_{L_{\alpha_t}} \kappa_{t_3} (q_o + \Omega_t) x_{t_{cg}}^2}{4\Omega_t^2 R_t} - \right. \\
& \left. \frac{C_{L_{\alpha_t}} \kappa_{t_3}}{4\Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{(\Omega - r_o) y_{t_{cg}}}{\Omega R_t \Omega_t} \right) z_{t_{cg}} + \frac{C_{L_{\alpha_t}} \kappa_{t_3} (q_o + \Omega_t) z_{t_{cg}}^2}{4\Omega_t^2 R_t} \right] \delta_{tail_o}^2 \right\} \frac{k_{14}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} + \\
& \frac{2}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{\kappa_3 (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}})}{8} \left(-\frac{v_o - v_w + p_o z_{r_{cg}}}{4\Omega^2 R} + \frac{3p_o z_{r_{cg}}}{4\Omega^2 R} \right) z_{r_{cg}} \delta_{lon_o} - \alpha_4 \sin(\phi_o) \tan(\theta_o) \right] \\
\frac{\partial \dot{q}}{\partial r} = & 2 \frac{z_{r_{cg}} \pi e_t \mathcal{R} C_{D_o} (u_o - u_w - q_o z_{r_{cg}}) k_5}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) C_{L_{\alpha}} \Omega^2 R} + \\
& \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{3C_{L_{\alpha}} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) z_{r_{cg}} \delta_{lon_o}^2}{16\Omega^2 R} + \frac{C_{L_{\alpha}} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) z_{r_{cg}} \delta_{col_o}^2}{4\Omega^2 R} + \right. \\
& \frac{C_{L_{\alpha}} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) z_{r_{cg}} \delta_{lat_o}^2}{16\Omega^2 R} - \frac{(\pi e_t \mathcal{R} - 2C_{L_{\alpha}}) \kappa_2 q_o z_{r_{cg}} \delta_{col_o}}{12\Omega^2} - \frac{C_{L_{\alpha}} \kappa_2 (\Omega - r_o) z_{r_{cg}} \delta_{col_o} \delta_{lon_o}}{3\Omega^2} - \\
& \left. \frac{C_{L_{\alpha}} \kappa_3 (v_o - v_w + p_o z_{r_{cg}}) z_{r_{cg}} \delta_{lat_o} \delta_{lon_o}}{8\Omega^2 R} \right] - \frac{\kappa_1 q_o k_7}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2} + \\
& \frac{2k_8}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(-\frac{3\kappa_1 (\Omega - r_o) \delta_{lat_o}}{4\Omega^2} + \frac{\kappa_2 (u_o - u_w - q_o z_{r_{cg}}) \delta_{col_o}}{\Omega^2 R} \right) + \\
& \frac{2k_{11}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[-(\pi e_t \mathcal{R}_t - C_{L_{\alpha_t}}) \left(\frac{\kappa_{t_1} r_o}{2\Omega_t^2} + 2 \frac{\kappa_{t_3} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t}) x_{t_{cg}}}{R_t^2 \Omega_t^2} \right) - \right. \\
& \left. \frac{\pi e_t \mathcal{R}_t C_{D_{O_t}} \kappa_{t_3}}{C_{L_{\alpha_t}} \Omega_t R_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) y_{t_{cg}} \right] + \\
& \frac{2k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{\kappa_{t_2}}{6} (\pi e_t \mathcal{R}_t - 2C_{L_{\alpha_t}}) \left(-\frac{(q_o + \Omega_t) x_{t_{cg}}}{\Omega_t^2 R_t} - \frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{2\Omega_t^2 R_t} \right) \delta_{tail_o} - \right. \\
& \left. \frac{C_{L_{\alpha_t}} \kappa_{t_3}}{4\Omega_t^2 R_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) y_{t_{cg}} \delta_{tail_o}^2 \right] +
\end{aligned}$$

$$\begin{aligned}
& \frac{2k_{13}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[-x_{t_{cg}} \left(\frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t})(v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t})}{\Omega_t^2 R_t} - \frac{(\pi e_t \mathcal{R}_t - C_{L\alpha_t}) r_o x_{t_{cg}}}{\Omega_t^2 R_t} \right) - \right. \\
& z_{t_{cg}} \left(\pi e_t \mathcal{R}_t - C_{L\alpha_t} \right) \left(-\frac{p_o x_{t_{cg}}}{\Omega_t^2 R_t} + \frac{\pi e_t \mathcal{R}_t C_{D\alpha_t} (q_o + \Omega_t) y_{t_{cg}}}{C_{L\alpha_t} \Omega_t^3 R_t} \right) \left. \right] + \\
& \frac{2k_{14}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ \frac{C_{L\alpha_t} \kappa_{t3} (q_o + \Omega_t) y_{t_{cg}} z_{t_{cg}} \delta_{tail_o}^2}{4\Omega_t^2 \Omega R_t} + \right. \\
& \left[\frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(\frac{\kappa_{t2} (q_o + \Omega_t)}{3\Omega_t^2} + \frac{\kappa_{t4}}{x_{t_{cg}} R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - k v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) \right) x_{t_{cg}} - \right. \\
& \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(-\frac{\kappa_{t4} y_{t_{cg}} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t})}{\Omega_t^3 R_t^2} - \right. \\
& \left. \left. \frac{\kappa_{t4}}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} - \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) x_{t_{cg}} \right) z_{t_{cg}} \right] \delta_{tail_o} \left. \right\} + \\
& \frac{2}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[p_o (I_z + I_{z_{bl}}) - \frac{\kappa_3 (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{8\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{r_{cg}} \delta_{lon_o} - \right. \\
& \left. \left(I_x + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) p_o - \alpha_4 \cos(\phi_o) \tan(\theta_o) \right]
\end{aligned}$$

$$\frac{\partial \dot{q}}{\partial \phi} = -2 \frac{\alpha_4 (q_o \cos \phi_o - r_o \sin \Phi_o) \tan \theta_o}{2I_y + I_{x_{bl}} + I_{y_{bl}}}$$

$$\frac{\partial \dot{q}}{\partial \theta} = -2 \frac{\alpha_4 (q_o \sin \phi_o + r_o \cos \phi_o) (1 + \tan^2 \theta_o)}{2I_y + I_{x_{bl}} + I_{y_{bl}}}$$

$$\frac{\partial \dot{q}}{\partial \psi} = 0$$

\dot{r} derivatives:

$$\begin{aligned}
\frac{\partial \dot{r}}{\partial u} = & -2 \frac{\pi e_t \mathcal{R} C_{D\alpha} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) k_3}{(I_z + I_{z_{bl}}) C_{L\alpha} R^2 \Omega^2} + \\
& \frac{k_4}{(I_z + I_{z_{bl}})} \left[\frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) \delta_{col_o}^2}{4R^2 \Omega^2} + \frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) \delta_{lat_o}^2}{16R^2 \Omega^2} - \right. \\
& \frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{r_{cg}}) \delta_{lat_o} \delta_{lon_o}}{8R^2 \Omega^2} - \frac{\kappa_2 (\pi e_t \mathcal{R} - 2C_{L\alpha}) p_o \delta_{col_o}}{12\Omega^2 R} + \frac{3C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{r_{cg}}) \delta_{lon_o}^2}{16R^2 \Omega^2} - \\
& \left. \frac{C_{L\alpha} \kappa_2 (\Omega - r_o) \delta_{col_o} \delta_{lon_o}}{3\Omega^2 R} \right] - \frac{p_o x_{t_{cg}} k_9}{(I_z + I_{z_{bl}}) \Omega_t^2 R_t} - \frac{2k_{10} \kappa_{t4}}{(I_z + I_{z_{bl}}) R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) x_{t_{cg}} \delta_{tail_o} + \\
& \frac{y_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) \pi e_t \mathcal{R}_t C_{D\alpha_t} (q_o + \Omega_t) k_{13}}{(I_z + I_{z_{bl}}) C_{L\alpha_t} \Omega_t^2 R_t} + \\
& \frac{k_{14}}{(I_z + I_{z_{bl}})} \left[\frac{C_{L\alpha_t} \kappa_{t3} (q_o + \Omega_t) y_{t_{cg}} \delta_{tail_o}^2}{4\Omega_t^2 R_t} + \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t}) \kappa_{t4} (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t}) y_{t_{cg}} \delta_{tail_o}}{4R_t^2 \Omega_t^2} \right] -
\end{aligned}$$

$$\begin{aligned}
& \frac{k_4 \kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8(I_z + I_{z_{bl}})R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \delta_{lon_o} \\
\frac{\partial \dot{r}}{\partial v} = & -2 \frac{\pi e \mathcal{R} C_{D_o} \kappa_3 (v_o - v_w + p_o z_{rcg}) k_3}{(I_z + I_{z_{bl}}) C_{L\alpha} R^2 \Omega^2} + \\
& \frac{k_4}{(I_z + I_{z_{bl}})} \left[\frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) \delta_{col_o}^2}{4R^2 \Omega^2} + \frac{3C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) \delta_{lat_o}^2}{16R^2 \Omega^2} - \right. \\
& \left. \frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{rcg}) \delta_{lat_o} \delta_{lon_o}}{8R^2 \Omega^2} - \frac{\kappa_2 (\pi e \mathcal{R} - 2C_{L\alpha}) q_o \delta_{col_o}}{12\Omega^2 R} + \frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) \delta_{lon_o}^2}{16R^2 \Omega^2} \right] + \\
& 2 \frac{(q_o + \Omega_t) x_{t_{cg}} k_9}{(I_z + I_{z_{bl}}) \Omega_t^2 R_t} + \frac{y_{t_{cg}} (\pi e_t \mathcal{R}_t - C_{L\alpha_t}) p_o k_{13}}{(I_z + I_{z_{bl}}) \Omega_t^2 R_t} + \\
& \frac{k_{14} \kappa_{t_4} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4(I_z + I_{z_{bl}}) R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} - \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) y_{t_{cg}} \delta_{tail_o} + \\
& - \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_3}{8R\Omega(I_z + I_{z_{bl}})} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \delta_{lat_o} + \frac{k_4 C_{L\alpha} \kappa_2 (\Omega - r_o) \delta_{col_o} \delta_{lat_o}}{3\Omega^2 R (I_z + I_{z_{bl}})} \\
\frac{\partial \dot{r}}{\partial w} = & \frac{4k_3 \kappa_3 (C_{L\alpha} - \pi e \mathcal{R})}{(I_z + I_{z_{bl}}) R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) + \frac{\kappa_2 (\pi e \mathcal{R} - 2C_{L\alpha}) (\Omega - r_o) \delta_{col_o} k_4}{6(I_z + I_{z_{bl}}) \Omega^2 R} - \frac{r_o x_{t_{cg}} k_9}{(I_z + I_{z_{bl}}) \Omega_t^2 R_t} - \\
& \frac{2k_{10} \kappa_{t_4}}{(I_z + I_{z_{bl}}) R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \chi v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) x_{t_{cg}} \delta_{tail_o} - \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_3 (u_o - u_w - q_o z_{rcg}) \delta_{lon_o}}{8(I_z + I_{z_{bl}}) R^2 \Omega^2} \\
\frac{\partial \dot{r}}{\partial p} = & \frac{k_3}{(I_z + I_{z_{bl}})} \left(\frac{(C_{L\alpha} - \pi e \mathcal{R}) \kappa_1 p_o}{\Omega^2} - 2 \frac{\pi e \mathcal{R} C_{D_o} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg}}{C_{L\alpha} R^2 \Omega^2} \right) + \\
& \frac{k_4}{(I_z + I_{z_{bl}})} \left[\frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o}^2}{4R^2 \Omega^2} + \frac{3C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lat_o}^2}{16R^2 \Omega^2} + \right. \\
& \left. \frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lon_o}^2}{16R^2 \Omega^2} - \frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lat_o} \delta_{lon_o}}{8R^2 \Omega^2} - \right. \\
& \left. \frac{\kappa_2 (\pi e \mathcal{R} - 2C_{L\alpha}) (u_o - u_w) \delta_{col_o}}{12\Omega^2 R} \right] - \frac{x_{t_{cg}} k_9}{(I_z + I_{z_{bl}})} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{\Omega_t^2 R_t} - 2 \frac{(q_o + \Omega_t) z_{t_{cg}}}{\Omega_t^2 R_t} \right) + \\
& \frac{2k_{10} \kappa_{t_4}}{(I_z + I_{z_{bl}}) \Omega_t^2 R_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \chi v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) y_{t_{cg}} x_{t_{cg}} \delta_{tail_o} + \\
& \frac{y_{t_{cg}} k_{13} (\pi e_t \mathcal{R}_t - C_{L\alpha_t})}{(I_z + I_{z_{bl}})} \left(\frac{v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{it}}{\Omega_t^2 R_t} + \frac{p_o z_{t_{cg}}}{\Omega_t^2 R_t} \right) + \\
& \frac{k_{14} (\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4(I_z + I_{z_{bl}})} \left[\frac{\kappa_{t_4}}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} - \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) z_{t_{cg}} - \frac{\kappa_{t_2} (q_o + \Omega_t)}{3\Omega_t^2} \right] y_{t_{cg}} \delta_{tail_o} + \\
& \frac{1}{(I_z + I_{z_{bl}})} \left[q_o \left(I_x + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) - \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_3 z_{rcg}}{8R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \delta_{lat_o} + \right. \\
& \left. \frac{k_4 (\pi e \mathcal{R} - 2C_{L\alpha}) \kappa_1 (\Omega - r_o) \delta_{lon_o}}{8\Omega^2} - \left(I_y + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) q_o + \frac{k_4 C_{L\alpha} \kappa_2 (\Omega - r_o) z_{rcg} \delta_{col_o} \delta_{lat_o}}{3\Omega^2 R} \right] \\
\frac{\partial \dot{r}}{\partial q} = & \frac{k_3}{(I_z + I_{z_{bl}})} \left(\frac{(C_{L\alpha} - \pi e \mathcal{R}) \kappa_1 q_o}{\Omega^2} + 2 \frac{\pi e \mathcal{R} C_{D_o} \kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg}}{C_{L\alpha} R^2 \Omega^2} \right) + \\
& \frac{k_4}{(I_z + I_{z_{bl}})} \left[- \frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{col_o}^2}{4R^2 \Omega^2} - \frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lat_o}^2}{16R^2 \Omega^2} + \right.
\end{aligned}$$

$$\begin{aligned}
& \frac{C_{L\alpha} \kappa_3 (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lat_o} \delta_{lon_o}}{8R^2 \Omega^2} - \frac{\kappa_2 (\pi e_t \mathcal{R} - 2C_{L\alpha}) (v_o - v_w) \delta_{col_o}}{12\Omega^2 R} \\
& \left. \frac{3C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lon_o}^2}{16R^2 \Omega^2} + \frac{C_{L\alpha} \kappa_2 (\Omega - r_o) z_{rcg} \delta_{col_o} \delta_{lon_o}}{3\Omega^2 R} \right] - \\
& \frac{x_{tcg} k_9}{(I_z + I_{z_{bl}})} \left(-\frac{p_o z_{tcg}}{\Omega_t^2 R_t} - 2 \frac{v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{i_t}}{\Omega_t^2 R_t} + \frac{r_o x_{tcg}}{\Omega_t^2 R_t} \right) - \\
& \frac{k_{10}}{(I_z + I_{z_{bl}})} \left\{ \frac{4\kappa_{t_2} (q_o + \Omega_t)}{3\Omega_t^2} + \kappa_{t_4} \left[-\frac{2}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) z_{tcg} + \right. \right. \\
& \left. \left. \frac{2}{R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) x_{tcg} \right] \right\} x_{tcg} \delta_{tail_o} + \\
& \frac{y_{tcg} k_{13} (\pi e_t \mathcal{R}_t - C_{L\alpha_t})}{(I_z + I_{z_{bl}})} \left[\frac{\pi e_t \mathcal{R}_t C_{D_{o_t}}}{C_{L\alpha_t} \Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) - \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} (q_o + \Omega_t) z_{tcg}}{C_{L\alpha_t} \Omega_t^2 R_t} \right] + \\
& \frac{k_{14}}{(I_z + I_{z_{bl}})} \left[\frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(-\frac{\kappa_{t_4} z_{tcg} (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{i_t})}{R_t^2 \Omega_t^2} - \frac{\kappa_{t_2} p_o}{3\Omega_t^2} \right) y_{tcg} \delta_{tail_o} - \right. \\
& \left. \frac{C_{L\alpha_t} \kappa_{t_3} z_{tcg} (q_o + \Omega_t) y_{tcg} \delta_{tail_o}^2}{4R_t \Omega_t^2} + \frac{C_{L\alpha_t} \kappa_{t_3}}{4\Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{R_t \Omega_t^2} \right) y_{tcg} \delta_{tail_o}^2 \right] + \\
& \frac{1}{(I_z + I_{z_{bl}})} \left[\frac{k_4 (\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_3}{8R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) z_{rcg} \delta_{lon_o} + \frac{k_4 (\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_1 (\Omega - r_o) \delta_{lat_o}}{8\Omega^2} + \right. \\
& \left. \left(I_x + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) p_o - p_o \left(I_y + \frac{I_{x_{bl}}}{2} + \frac{I_{y_{bl}}}{2} \right) \right]
\end{aligned}$$

$$\begin{aligned}
\frac{\partial \dot{r}}{\partial r} &= 2 \frac{\pi e_t \mathcal{R} C_{D_o} \kappa_1 (\Omega - r_o) k_3}{(I_z + I_{z_{bl}}) C_{L\alpha} \Omega^2} + \\
& \frac{k_4}{(I_z + I_{z_{bl}})} \left[-\frac{C_{L\alpha} \kappa_1 (\Omega - r_o) \delta_{lat_o}^2}{8\Omega^2} - \frac{C_{L\alpha} \kappa_1 (\Omega - r_o) \delta_{col_o}^2}{4\Omega^2} + \right. \\
& \left. \frac{\kappa_2 (\pi e_t \mathcal{R} - 2C_{L\alpha})}{6\Omega} \left(\frac{v_i}{R\Omega} - \frac{w_o - w_w}{R\Omega} \right) \delta_{col_o} - \frac{C_{L\alpha} \kappa_1 (\Omega - r_o) \delta_{lon_o}^2}{8\Omega^2} + \right. \\
& \left. \frac{C_{L\alpha} \kappa_2 (u_o - u_w - q_o z_{rcg}) \delta_{col_o} \delta_{lon_o}}{3\Omega^2 R} \right] - \frac{x_{tcg} k_9}{(I_z + I_{z_{bl}})} \left(2 \frac{(q_o + \Omega_t) x_{tcg}}{R_t \Omega_t^2} + \frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega_t^2} \right) - \\
& \frac{2k_{10} \kappa_{t_4}}{(I_z + I_{z_{bl}}) \Omega_t^2 R_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{R_t \Omega_t^2} \right) y_{tcg} x_{tcg} \delta_{tail_o} + \\
& \frac{y_{tcg} k_{13} (\pi e_t \mathcal{R}_t - C_{L\alpha_t})}{(I_z + I_{z_{bl}})} \left(-\frac{p_o x_{tcg}}{R_t \Omega_t^2} + \frac{\pi e_t \mathcal{R}_t C_{D_{o_t}} (q_o + \Omega_t) y_{tcg}}{C_{L\alpha_t} \Omega_t^3 R_t} \right) + \left[\frac{C_{L\alpha_t} \kappa_{t_3} y_{tcg}^2 (q_o + \Omega_t) \delta_{tail_o}^2}{4\Omega_t^3 R_t} + \right. \\
& \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left(-\frac{\kappa_{t_4} y_{tcg} (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{i_t})}{\Omega_t^3 R_t^2} - \right. \\
& \left. \frac{\kappa_{t_4}}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} - \frac{r_o y_{tcg}}{R_t \Omega_t^2} \right) x_{tcg} \right) y_{tcg} \delta_{tail_o} \left. \right] \frac{k_{14}}{(I_z + I_{z_{bl}})} + \\
& \frac{1}{(I_z + I_{z_{bl}})} \left(-\frac{k_4 (\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_1 p_o \delta_{lon_o}}{8\Omega^2} - \frac{k_4 (\pi e_t \mathcal{R} - 2C_{L\alpha}) \kappa_1 q_o \delta_{lat_o}}{8\Omega^2} - \right. \\
& \left. \frac{k_4 C_{L\alpha} \kappa_2 (v_o - v_w + p_o z_{rcg}) \delta_{col_o} \delta_{lat_o}}{3\Omega^2 R} \right)
\end{aligned}$$

$$\frac{\partial \dot{r}}{\partial \phi} = 0$$

$$\frac{\partial \dot{r}}{\partial \theta} = 0$$

$$\frac{\partial \dot{r}}{\partial \psi} = 0$$

Partial derivatives of the Euler angles.

$\dot{\phi}$ derivatives:

$$\frac{\partial \dot{\phi}}{\partial u} = 0$$

$$\frac{\partial \dot{\phi}}{\partial v} = 0$$

$$\frac{\partial \dot{\phi}}{\partial w} = 0$$

$$\frac{\partial \dot{\phi}}{\partial p} = 0$$

$$\frac{\partial \dot{\phi}}{\partial q} = \sin \phi_o \tan \theta_o$$

$$\frac{\partial \dot{\phi}}{\partial r} = \cos \phi_o \tan \theta_o$$

$$\frac{\partial \dot{\phi}}{\partial \phi} = q_o \cos \phi_o \tan \theta_o - r_o \sin \phi_o \tan \theta_o$$

$$\frac{\partial \dot{\phi}}{\partial \theta} = q_o \sin \phi_o (1 + \tan^2 \theta_o) + r_o \cos \phi_o (1 + \tan^2 \theta_o)$$

$$\frac{\partial \dot{\phi}}{\partial \psi} = 0$$

$\dot{\theta}$ derivatives:

$$\frac{\partial \dot{\theta}}{\partial u} = 0$$

$$\frac{\partial \dot{\theta}}{\partial v} = 0$$

$$\frac{\partial \dot{\theta}}{\partial w} = 0$$

$$\frac{\partial \dot{\theta}}{\partial p} = 0$$

$$\frac{\partial \dot{\theta}}{\partial q} = \cos \phi_o$$

$$\frac{\partial \dot{\theta}}{\partial r} = -\sin \phi_o$$

$$\frac{\partial \dot{\theta}}{\partial \phi} = -q_o \sin \phi_o - r_o \cos \phi_o$$

$$\frac{\partial \dot{\theta}}{\partial \theta} = 0$$

$$\frac{\partial \dot{\theta}}{\partial \psi} = 0$$

$\dot{\psi}$ derivatives:

$$\frac{\partial \dot{\psi}}{\partial u} = 0$$

$$\frac{\partial \dot{\psi}}{\partial v} = 0$$

$$\frac{\partial \dot{\psi}}{\partial w} = 0$$

$$\frac{\partial \dot{\psi}}{\partial p} = 0$$

$$\frac{\partial \dot{\psi}}{\partial q} = \sin \phi_o \sec \theta_o$$

$$\frac{\partial \dot{\psi}}{\partial r} = \cos \phi_o \sec \theta_o$$

$$\frac{\partial \dot{\psi}}{\partial \phi} = q_o \cos \phi_o \sec \theta_o - r_o \sin \phi_o \sec \theta_o$$

$$\frac{\partial \psi}{\partial \theta} = q_o \sin \phi_o \sec \theta_o \tan \theta_o + r_o \cos \phi_o \sec \theta_o \tan \theta_o$$

$$\frac{\partial \psi}{\partial \psi} = 0$$

Derivatives for the matrix **B**.

u derivatives:

$$\frac{\partial \dot{u}}{\partial \delta_{\theta_{col}}} = \left\{ \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) \delta_{col_o}}{2\Omega^2 R} + \frac{C_{L\alpha} (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) \delta_{lat_o}}{4R^2 \Omega^2} - \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + \frac{3\kappa_4}{4} \left(3 \frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] \delta_{lon_o} + \frac{k_6 (\pi e \mathcal{R} - 2C_{L\alpha})}{4m_T} \left[-\frac{\kappa_2 q_o (\Omega - r_o)}{3\Omega^2} + \frac{\kappa_4}{R\Omega} (u_o - u_w - q_o z_{rcg}) \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \right] \right\}$$

$$\frac{\partial \dot{u}}{\partial \delta_{\theta_{lat}}} = \frac{k_6}{m_T} \left\{ \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) \delta_{lat_o}}{8\Omega^2 R} + \frac{C_{L\alpha} (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) \delta_{col_o}}{4R^2 \Omega^2} - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) \delta_{lon_o}}{8\Omega^2 R} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{q_o (u_o - u_w - q_o z_{rcg})}{\Omega^2 R} + \frac{p_o (v_o - v_w + p_o z_{rcg})}{\Omega^2 R} \right) \right\}$$

$$\frac{\partial \dot{u}}{\partial \delta_{\theta_{lon}}} = \frac{k_6}{m_T} \left\{ \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) \delta_{lon_o}}{8\Omega^2 R} - \frac{C_{L\alpha}}{6} \left(\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + \frac{3\kappa_4}{4} \left[3 \frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right] \right) \delta_{col_o} - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) \delta_{lat_o}}{8\Omega^2 R} - \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left(\left(\frac{\Omega - r_o}{\Omega} \right) \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) - \frac{3p_o (u_o - u_w)}{4\Omega^2 R} - \frac{q_o (v_o - v_w + p_o z_{rcg})}{4\Omega^2 R} + \frac{3q_o p_o z_{rcg}}{4\Omega^2 R} \right) \right\}$$

$$\frac{\partial \dot{u}}{\partial \delta_{\theta_{tail}}} = \frac{k_{14}}{m_T} \left\{ \frac{\kappa_{t3} C_{L\alpha_t} (q_o + \Omega_t)}{2\Omega_t} \left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) \delta_{tail_o} + \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left[-\frac{\kappa_{t2} p_o (q_o + \Omega_t)}{3\Omega_t^2} + \frac{\kappa_{t4}}{R_t \Omega_t} \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t}) \right] \right\}$$

\dot{v} derivatives:

$$\begin{aligned} \frac{\partial \dot{v}}{\partial \delta_{\theta_{col}}} &= \frac{k_6}{m_T} \left\{ -\frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) \delta_{col_o}}{2\Omega^2 R} - \right. \\ &\quad \left. \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + \frac{3\kappa_4}{4} \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] \delta_{lat_o} + \right. \\ &\quad \left. \frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) \delta_{lon_o}}{4R^2 \Omega^2} + \right. \\ &\quad \left. \frac{(\pi e R - 2C_{L\alpha})}{4} \left[\frac{\kappa_2 q_o (\Omega - r_o)}{3\Omega^2} - \frac{\kappa_4}{R\Omega} (v_o - v_w + p_o z_{rcg}) \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \right] \right\} \\ \frac{\partial \dot{v}}{\partial \delta_{\theta_{lat}}} &= \frac{k_6}{m_T} \left\{ -\frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) \delta_{lat_o}}{8\Omega^2 R} - \right. \\ &\quad \left. \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + \frac{3\kappa_4}{4} \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] \delta_{col_o} + \right. \\ &\quad \left. \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) \delta_{lon_o}}{8\Omega^2 R} + \right. \\ &\quad \left. \frac{\kappa_3 (\pi e R - 2C_{L\alpha})}{8} \left(-\frac{(\Omega - r_o)}{\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) + \frac{3p_o (v_o - v_w)}{4\Omega^2 R} + \frac{p_o (u_o - u_w - q_o z_{rcg})}{4\Omega^2 R} + \frac{3q_o p_o z_{rcg}}{4\Omega^2 R} \right) \right\} \\ \frac{\partial \dot{v}}{\partial \delta_{\theta_{lon}}} &= \frac{k_6}{m_T} \left\{ -\frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) \delta_{lon_o}}{8\Omega^2 R} + \frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) \delta_{col_o}}{4R^2 \Omega^2} + \right. \\ &\quad \left. \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) \delta_{lat_o}}{8\Omega^2 R} - \frac{\kappa_3 (\pi e R - 2C_{L\alpha})}{32} \left(\frac{q_o (u_o - u_w - q_o z_{rcg})}{\Omega^2 R} + \frac{p_o (v_o - v_w + p_o z_{rcg})}{\Omega^2 R} \right) \right\} \\ \frac{\partial \dot{v}}{\partial \delta_{\theta_{tail}}} &= \frac{k_{10}}{m_T} \left\{ \frac{2\kappa_2 (q_o + \Omega t)^2}{3\Omega t^2} + \kappa_{t4} \left[\left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega t} + \frac{r_o y_{tcg}}{\Omega t^2 R_t} \right)^2 + \left(\frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega t} - \frac{p_o y_{tcg}}{\Omega t^2 R_t} \right)^2 \right] \right\} \end{aligned}$$

\dot{w} derivatives:

$$\begin{aligned} \frac{\partial \dot{w}}{\partial \delta_{\theta_{col}}} &= \frac{k_2}{m_T} \left\{ \frac{2\kappa_2 (\Omega - r_o)^2}{3\Omega^2} + \kappa_4 \left[\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right] \right\} \\ \frac{\partial \dot{w}}{\partial \delta_{\theta_{lat}}} &= \frac{k_2 \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg})}{m_T \Omega^2 R} \\ \frac{\partial \dot{w}}{\partial \delta_{\theta_{lon}}} &= -\frac{k_2 \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg})}{m_T \Omega^2 R} \\ \frac{\partial \dot{w}}{\partial \delta_{\theta_{tail}}} &= \frac{k_{14}}{m_T} \left\{ -\frac{C_{L\alpha_t} \kappa_{t3} (q_o + \Omega t)}{2\Omega} \left(\frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega t} - \frac{p_o y_{tcg}}{\Omega t^2 R_t} \right) \delta_{tail_o} + \right. \end{aligned}$$

$$\left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left[\frac{\kappa_{t_2} r_o (q_o + \Omega_t)}{3\Omega_t^2} - \frac{\kappa_{t_4}}{R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \varkappa v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{i_t}) \right] \right\}$$

\dot{p} derivatives:

$$\begin{aligned} \frac{\partial \dot{p}}{\partial \delta_{\theta_{col}}} &= \frac{2k_6}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left\{ -1/2 \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o}}{\Omega^2 R} - \right. \\ &\quad \left. \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + 3/4 \kappa_4 \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] z_{rcg} \delta_{lat_o} + \right. \\ &\quad \left. \frac{C_{L\alpha} (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lon_o}}{4\kappa_4 R^2 \Omega^2} + \right. \\ &\quad \left. \frac{(\pi e \mathcal{R} - 2C_{L\alpha})}{4} \left(\frac{\kappa_2 q_o (\Omega - r_o)}{3\Omega^2} - \frac{\kappa_4}{R\Omega} (v_o - v_w + p_o z_{rcg}) \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \right) z_{rcg} \right\} - \\ &\quad 2 \frac{k_8 \kappa_2 (\Omega - r_o) (u_o - u_w - q_o z_{rcg})}{(2I_x + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{p}}{\partial \delta_{\theta_{lat}}} &= \frac{2k_6}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left\{ - \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lat_o}}{8\Omega^2 R} - \right. \\ &\quad \left. \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + 3/4 \kappa_4 \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] z_{rcg} \delta_{col_o} + \right. \\ &\quad \left. \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lon_o}}{8\Omega^2 R} + \right. \\ &\quad \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left[- \frac{(\Omega - r_o)}{\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) + \frac{3q_o (v_o - v_w)}{4\Omega^2 R} + \right. \right. \\ &\quad \left. \left. \frac{p_o (u_o - u_w - q_o z_{rcg})}{4\Omega^2 R} + \frac{3q_o p_o z_{rcg}}{4\Omega^2 R} \right] z_{rcg} \right\} + \frac{3k_8 \kappa_3 (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg})}{4(2I_x + I_{x_{bl}} + I_{y_{bl}}) R^2 \Omega^2} \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{p}}{\partial \delta_{\theta_{lon}}} &= \frac{2k_6}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left\{ - \frac{2C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{lon_o}}{3\Omega^2 R} + \right. \\ &\quad \left. \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) z_{rcg} \delta_{lat_o}}{8\Omega^2 R} - \right. \\ &\quad \left. \frac{\kappa_3 (\pi e \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{q_o (u_o - u_w - q_o z_{rcg})}{\Omega^2 R} + \frac{p_o (v_o - v_w + p_o z_{rcg})}{\Omega^2 R} \right) z_{rcg} + \right. \\ &\quad \left. \frac{C_{L\alpha} (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) z_{rcg} \delta_{col_o}}{4\kappa_4 R^2 \Omega^2} \right\} + \\ &\quad \frac{2k_8}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left[- \frac{3\kappa_1 (\Omega - r_o)^2}{8\Omega^2} - \frac{3\kappa_3}{16} \left(3 \frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] - \\ &\quad 2 \frac{\alpha_1}{2I_x + I_{x_{bl}} + I_{y_{bl}}} \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{p}}{\partial \delta_{\theta_{tail}}} = & \frac{2k_{10}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left\{ \frac{2\kappa_{t2}(q_o + \Omega_t)^2}{3\Omega_t^2} + \kappa_{t4} \left[\left(\frac{u_o - u_w - q_o z_{t_{cg}}}{R_t \Omega_t} + \frac{r_o y_{t_{cg}}}{\Omega_t^2 R_t} \right)^2 + \right. \right. \\ & \left. \left. \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \varkappa v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right)^2 \right] \right\} z_{t_{cg}} + \\ & \frac{2k_{14}}{(2I_x + I_{x_{bl}} + I_{y_{bl}})} \left\{ \frac{C_{L\alpha} \kappa_{t3}}{2\Omega} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \varkappa v_i}{R_t \Omega_t} + \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) (q_o + \Omega_t) y_{t_{cg}} \delta_{tail_o} - \right. \\ & \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha})}{4} \left[\frac{\kappa_{t2} r_o (q_o + \Omega_t)}{3\Omega_t^2} - \right. \right. \\ & \left. \left. \frac{\kappa_{t4}}{R_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{t_{cg}} - \varkappa v_i}{R_t \Omega_t} - \frac{p_o y_{t_{cg}}}{\Omega_t^2 R_t} \right) (v_o - v_w + p_o z_{t_{cg}} - r_o x_{t_{cg}} - v_{it}) \right] y_{t_{cg}} \right\} \end{aligned}$$

\dot{q} derivatives:

$$\begin{aligned} \frac{\partial \dot{q}}{\partial \delta_{\theta_{col}}} = & \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{r_{cg}}) z_{r_{cg}} \delta_{col_o}}{2\Omega^2 R} - \right. \\ & \frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{r_{cg}}) (v_o - v_w + p_o z_{r_{cg}}) z_{r_{cg}} \delta_{lat_o}}{4R^2 \Omega^2} + \\ & \left. \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + \frac{3\kappa_4}{4} \left(3 \frac{(u_o - u_w - q_o z_{r_{cg}})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{r_{cg}})^2}{R^2 \Omega^2} \right) \right] z_{r_{cg}} \delta_{lon_o} - \right. \\ & \left. \frac{(\pi e_t \mathcal{R} - 2C_{L\alpha})}{4} \left(- \frac{\kappa_2 q_o (\Omega - r_o)}{3\Omega^2} + \frac{\kappa_4 (u_o - u_w - q_o z_{r_{cg}})}{R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) \right) z_{r_{cg}} \right\} - \\ & 2 \frac{k_8 \kappa_2 (\Omega - r_o) (u_o - u_w - q_o z_{r_{cg}})}{(2I_y + I_{x_{bl}} + I_{y_{bl}}) \Omega^2 R} \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{q}}{\partial \delta_{\theta_{lat}}} = & \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ - \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{r_{cg}}) z_{r_{cg}} \delta_{lat_o}}{8\Omega^2 R} + \right. \\ & \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{r_{cg}}) z_{r_{cg}} \delta_{lon_o}}{8\Omega^2 R} + \\ & \frac{\kappa_3 (\pi e_t \mathcal{R} - 2C_{L\alpha})}{32} \left(\frac{q_o (u_o - u_w - q_o z_{r_{cg}})}{\Omega^2 R} + \frac{p_o (v_o - v_w + p_o z_{r_{cg}})}{\Omega^2 R} \right) z_{r_{cg}} - \\ & \left. \frac{C_{L\alpha} \kappa_4 (u_o - u_w - q_o z_{r_{cg}}) (v_o - v_w + p_o z_{r_{cg}}) z_{r_{cg}} \delta_{col_o}}{4R^2 \Omega^2} \right\} + \\ & \frac{2k_8}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left[\frac{3\kappa_1 (\Omega - r_o)^2}{8\Omega^2} + \frac{3\kappa_3}{16} \left(\frac{(u_o - u_w - q_o z_{r_{cg}})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{r_{cg}})^2}{R^2 \Omega^2} \right) \right] + \\ & \frac{2\alpha_3}{2I_y + I_{x_{bl}} + I_{y_{bl}}} \end{aligned}$$

$$\begin{aligned} \frac{\partial \dot{q}}{\partial \delta_{\theta_{lon}}} = & \frac{2k_6}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ \frac{C_{L\alpha}}{6} \left[\frac{\kappa_2 (\Omega - r_o)^2}{\Omega^2} + \frac{3\kappa_4}{4} \left(3 \frac{(u_o - u_w - q_o z_{r_{cg}})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{r_{cg}})^2}{R^2 \Omega^2} \right) \right] z_{r_{cg}} \delta_{col_o} + \right. \\ & \left. \frac{C_{L\alpha} \kappa_3 (\Omega - r_o) (v_o - v_w + p_o z_{r_{cg}}) z_{r_{cg}} \delta_{lat_o}}{8\Omega^2 R} - \frac{3C_{L\alpha} \kappa_3 (\Omega - r_o) (u_o - u_w - q_o z_{r_{cg}}) z_{r_{cg}} \delta_{lon_o}}{8\Omega^2 R} \right\} + \end{aligned}$$

$$\begin{aligned}
& \frac{3\kappa_3(u_o - u_w - q_o z_{rcg})(v_o - v_w + p_o z_{rcg})k_8}{8(2I_y + I_{x_{bl}} + I_{y_{bl}})R^2\Omega^2} + \\
& \frac{\kappa_3(\pi e \mathcal{R} - 2C_{L\alpha})}{4(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left(\frac{(\Omega - r_o)}{\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) - \frac{3p_o(u_o - u_w)}{4\Omega^2 R} - \frac{q_o(v_o - v_w + p_o z_{rcg})}{4\Omega^2 R} + \frac{3q_o p_o z_{rcg}}{4\Omega^2 R} \right) z_{rcg} \\
\frac{\partial \dot{q}}{\partial \delta_{\theta_{tail}}} = & \frac{2k_{12}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ -\frac{C_{L\alpha t}}{4} \left[\frac{\kappa_{t1}(q_o + \Omega_t)^2}{\Omega_t^2} + \right. \right. \\
& \left. \left. \kappa_{t3} \left(\left[\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right]^2 + \left[\frac{p_o y_{tcg}}{\Omega_t^2 R_t} - \frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega_t} \right]^2 \right) \right] \delta_{tail_o} + \right. \\
& \frac{\kappa_{t2}(\pi e_t \mathcal{R}_t - 2C_{L\alpha t})}{6} \left[\frac{(q_o + \Omega_t)(v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{it})}{\Omega_t^2 R_t} - \right. \\
& \left. \left. \frac{r_o(w_o - w_w + q_o x_{tcg} - \varkappa v_i)}{2\Omega_t^2 R_t} - \frac{p_o(u_o - u_w - q_o z_{tcg})}{2\Omega_t^2 R_t} \right] \right\} + \\
& \frac{2k_{14}}{(2I_y + I_{x_{bl}} + I_{y_{bl}})} \left\{ 2 \left[-\frac{C_{L\alpha t} \kappa_{t3}(q_o + \Omega_t)}{4\Omega} \left(\frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega_t} + \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) x_{tcg} - \right. \right. \\
& \left. \left. \frac{C_{L\alpha t} \kappa_{t3}(q_o + \Omega_t)}{4\Omega} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{(\Omega - r_o)y_{tcg}}{\Omega_t R_t \Omega_t} \right) z_{tcg} \right] \delta_{tail_o} + \right. \\
& \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha t})}{4} \left[\frac{\kappa_{t2} r_o (q_o + \Omega_t)}{3\Omega_t^2} - \right. \\
& \left. \left. \frac{\kappa_{t4}}{r_t \Omega_t} \left(\frac{w_o - w_w + q_o x_{tcg} - \varkappa v_i}{R_t \Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right) (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{it}) \right] x_{tcg} - \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha t})}{4} \left[\frac{\kappa_{t4}(v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{it})}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} - \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) - \right. \right. \\
& \left. \left. \frac{\kappa_{t2} p_o (q_o + \Omega_t)}{3\Omega_t^2} \right] z_{tcg} \right\}
\end{aligned}$$

\dot{r} derivatives:

$$\begin{aligned}
\frac{\partial \dot{r}}{\partial \delta_{\theta_{col}}} = & \frac{k_4}{(I_z + I_{z_{bl}})} \left\{ \frac{C_{L\alpha}}{4} \left[\frac{\kappa_1(\Omega - r_o)^2}{\Omega^2} + \kappa_3 \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] \delta_{col_o} + \right. \\
& \frac{C_{L\alpha} \kappa_2(\Omega - r_o)(v_o - v_w + p_o z_{rcg}) \delta_{lat_o}}{3\Omega^2 R} - \frac{C_{L\alpha} \kappa_2(\Omega - r_o)(u_o - u_w - q_o z_{rcg}) \delta_{lon_o}}{3\Omega^2 R} - \\
& \left. \frac{\kappa_2(\pi e \mathcal{R} - 2C_{L\alpha})}{6} \left[\left(\frac{v_i}{R\Omega} - \frac{w_o - w_w}{R\Omega} \right) \frac{(\Omega - r_o)}{\Omega} + \frac{p_o(u_o - u_w)}{2\Omega^2 R} + \frac{q_o(v_o - v_w)}{2\Omega^2 R} \right] \right\} \\
\frac{\partial \dot{r}}{\partial \delta_{\theta_{lat}}} = & \frac{k_4}{(I_z + I_{z_{bl}})} \left\{ \frac{C_{L\alpha}}{8} \left[\frac{\kappa_1(\Omega - r_o)^2}{\Omega^2} + \frac{\kappa_3}{2} \left(\frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + 3 \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] \delta_{lat_o} + \right. \\
& \left. \frac{C_{L\alpha} \kappa_2(\Omega - r_o)(v_o - v_w + p_o z_{rcg}) \delta_{col_o}}{3\Omega^2 R} - \frac{C_{L\alpha} \kappa_3(u_o - u_w - q_o z_{rcg})(v_o - v_w + p_o z_{rcg}) \delta_{lon_o}}{8R^2 \Omega^2} \right\}
\end{aligned}$$

$$\begin{aligned}
& \left. \frac{(\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left[\frac{\kappa_3}{R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) (v_o - v_w + p_o z_{rcg}) - \frac{\kappa_1 q_o (\Omega - r_o)}{\Omega^2} \right] \right\} \\
\frac{\partial \dot{r}}{\partial \delta_{\theta_{lon}}} &= \frac{k_4}{(I_z + I_{z_{bl}})} \left\{ \frac{C_{L\alpha}}{8} \left[\frac{\kappa_1 (\Omega - r_o)^2}{\Omega^2} + \frac{\kappa_3}{2} \left(3 \frac{(u_o - u_w - q_o z_{rcg})^2}{R^2 \Omega^2} + \frac{(v_o - v_w + p_o z_{rcg})^2}{R^2 \Omega^2} \right) \right] \delta_{lon_o} - \right. \\
& \left. \frac{C_{L\alpha} \kappa_2 (\Omega - r_o) (u_o - u_w - q_o z_{rcg}) \delta_{col_o}}{3\Omega^2 R} - \frac{C_{L\alpha} \kappa_3 (u_o - u_w - q_o z_{rcg}) (v_o - v_w + p_o z_{rcg}) \delta_{lat_o}}{8R^2 \Omega^2} - \right. \\
& \left. \frac{(\pi e \mathcal{R} - 2C_{L\alpha})}{8} \left[\frac{\kappa_3}{R\Omega} \left(\frac{w_o - w_w}{R\Omega} - \frac{v_i}{R\Omega} \right) (u_o - u_w - q_o z_{rcg}) - \frac{\kappa_1 p_o (\Omega - r_o)}{\Omega^2} \right] \right\} \\
\frac{\partial \dot{r}}{\partial \delta_{\theta_{tail}}} &= -\frac{k_{10}}{(I_z + I_{z_{bl}})} \left\{ \frac{2\kappa_{t2} (q_o + \Omega_t)^2}{3\Omega_t^2} + \right. \\
& \left. \kappa_{t4} \left[\left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right)^2 + \left(\frac{w_o - w_w + q_o x_{tcg} - \kappa v_i}{R_t \Omega_t} - \frac{p_o y_{tcg}}{\Omega_t^2 R_t} \right)^2 \right] \right\} x_{tcg} + \\
& \frac{k_{14}}{(I_z + I_{z_{bl}})} \left\{ \frac{C_{L\alpha_t} \kappa_{t3}}{2\Omega} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} + \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) (q_o + \Omega_t) y_{tcg} \delta_{tail_o} + \right. \\
& \left. \frac{(\pi e_t \mathcal{R}_t - 2C_{L\alpha_t})}{4} \left[\frac{\kappa_{t4} (v_o - v_w + p_o z_{tcg} - r_o x_{tcg} - v_{it})}{R_t \Omega_t} \left(\frac{u_o - u_w - q_o z_{tcg}}{R_t \Omega_t} - \frac{r_o y_{tcg}}{\Omega_t^2 R_t} \right) - \right. \right. \\
& \left. \left. \frac{\kappa_{t2} p_o (q_o + \Omega_t)}{3\Omega_t^2} \right] y_{tcg} \right\}
\end{aligned}$$

$\dot{\phi}$ derivatives:

$$\frac{\partial \dot{\phi}}{\partial \delta_{\theta_{col}}} = 0$$

$$\frac{\partial \dot{\phi}}{\partial \delta_{\theta_{lat}}} = 0$$

$$\frac{\partial \dot{\phi}}{\partial \delta_{\theta_{lon}}} = 0$$

$$\frac{\partial \dot{\phi}}{\partial \delta_{\theta_{tail}}} = 0$$

$\dot{\theta}$ derivatives:

$$\frac{\partial \dot{\theta}}{\partial \delta_{\theta_{col}}} = 0$$

$$\frac{\partial \dot{\theta}}{\partial \delta_{\theta_{lat}}} = 0$$

$$\frac{\partial \dot{\theta}}{\partial \delta_{\theta_{lon}}} = 0$$

$$\frac{\partial \dot{\theta}}{\partial \delta_{\theta_{tail}}} = 0$$

$\dot{\phi}$ derivatives:

$$\frac{\partial \dot{\psi}}{\partial \delta_{\theta_{col}}} = 0$$

$$\frac{\partial \dot{\psi}}{\partial \delta_{\theta_{lat}}} = 0$$

$$\frac{\partial \dot{\psi}}{\partial \delta_{\theta_{lon}}} = 0$$

$$\frac{\partial \dot{\psi}}{\partial \delta_{\theta_{tail}}} = 0$$

D.3 Riccati Equation Derivation

In chapter 5, the proposed energy index function for the Air Star Evolution Helicopter is:

$$J_{LQR} = \frac{1}{2} \mathbf{x}^T(t_f) \mathbf{P}(t_f) \mathbf{x}(t_f) + \frac{1}{2} \int_{t_0}^{t_f} (\|\mathbf{z}\|^2 + \rho \|\mathbf{u}\|^2) dt$$

Where the parameter ρ is a positive constant to mediate between the energy of the controlled states \mathbf{z} and the energy of the controller outputs \mathbf{u} .

Using the quadratic form, the energy index is written as

$$J_{LQR} = \frac{1}{2} \mathbf{x}^T(t_f) \mathbf{P}(t_f) \mathbf{x}(t_f) + \frac{1}{2} \int_{t_0}^{t_f} (\mathbf{z}^T \mathbf{Q} \mathbf{z} + \rho \mathbf{u}^T \mathbf{R} \mathbf{u}) dt$$

where $\mathbf{Q} \in \mathbb{R}^{9 \times 9}$ and $\mathbf{R} \in \mathbb{R}^{4 \times 4}$ are positive definite matrices. Substituting (5.7) into the energy index results in

$$J_{LQR} = \frac{1}{2} \mathbf{x}^T(t_f) \mathbf{P}(t_f) \mathbf{x}(t_f) + \frac{1}{2} \int_{t_0}^{t_f} \begin{bmatrix} \mathbf{x}^T & \mathbf{u}^T \end{bmatrix} \begin{bmatrix} \bar{\mathbf{Q}} & \bar{\mathbf{N}} \\ \bar{\mathbf{N}}^T & \bar{\mathbf{R}} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{u} \end{bmatrix} dt \quad (\text{D.1})$$

where

$$\begin{aligned} \bar{\mathbf{Q}} &= \mathbf{G}^T \mathbf{Q} \mathbf{G} \\ \bar{\mathbf{R}} &= \mathbf{H}^T \mathbf{Q} \mathbf{H} + \rho \mathbf{R} \\ \bar{\mathbf{N}} &= \mathbf{G}^T \mathbf{Q} \mathbf{H} \end{aligned} \quad (\text{D.2})$$

Matrices \mathbf{Q} and \mathbf{R} are estimated using the Bryson's rule

$$\mathbf{Q}_{ii} = \frac{1}{\max\{\mathbf{z}_i^2\}}$$

and

$$\mathbf{R}_{jj} = \frac{1}{\max\{\mathbf{u}_j^2\}}$$

From [67] – [82], the cost index J_{LQR} can be minimized using the Linear Quadratic Neighboring-Optimal method, under linear dynamic constraint

$$\dot{\mathbf{x}} = \mathbf{A} \mathbf{x} + \mathbf{B} \mathbf{u}$$

Therefore, the augmented index J_A is [71, 72]

$$J_A = \frac{1}{2}\phi[\mathbf{x}(t_f)] + \frac{1}{2} \int_{t_0}^{t_f} (\mathcal{L}[\mathbf{x}(t), \mathbf{u}(t)] + \boldsymbol{\lambda}^T [\mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) - \dot{\mathbf{x}}(t)]) dt$$

and the Hamiltonian \mathcal{H} associated to the cost function is [71, 72]

$$\mathcal{H} = [J_{LQR} + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u})]$$

where $\boldsymbol{\lambda}$ is the Lagrange multiplier. By expanding the integral part of (D.1) the resulting Hamiltonian is

$$\mathcal{H} = (\mathbf{x}^T \bar{\mathbf{Q}}\mathbf{x} + \mathbf{u}^T \bar{\mathbf{R}}\mathbf{u} + 2\mathbf{x}^T \bar{\mathbf{N}}\mathbf{u}) + \boldsymbol{\lambda}^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}) \quad (\text{D.3})$$

After integrating by parts the term $\boldsymbol{\lambda}^T \dot{\mathbf{x}}(t)$ and introducing a variations in the augmented cost function the following three equations are derived –see [72]

$$\dot{\boldsymbol{\lambda}} = - \left(\frac{\partial \mathcal{H}(\cdot)}{\partial \mathbf{x}} \right)^T = -\mathbf{A}^T \boldsymbol{\lambda} - \left(\frac{\partial \mathcal{L}(\cdot)}{\partial \mathbf{x}} \right)^T = -\bar{\mathbf{Q}}\mathbf{x} - \bar{\mathbf{N}}\mathbf{u} - \mathbf{A}^T \boldsymbol{\lambda} \quad (\text{D.4})$$

$$\left(\frac{\partial \mathcal{H}(\cdot)}{\partial \mathbf{u}} \right)^T = \left(\frac{\partial \mathcal{L}(\cdot)}{\partial \mathbf{u}} \right)^T + \mathbf{B}^T \boldsymbol{\lambda} = \bar{\mathbf{N}}^T \mathbf{x} + \bar{\mathbf{R}}\mathbf{u} + \mathbf{B}^T \boldsymbol{\lambda} = 0 \quad (\text{D.5})$$

$$\boldsymbol{\lambda}_f = \left. \frac{\partial \phi(\cdot)}{\partial \mathbf{x}} \right|_{t=t_f} \quad (\text{D.6})$$

Solving (D.5) for \mathbf{u} , results in

$$\mathbf{u} = -\bar{\mathbf{R}}^{-1} [\bar{\mathbf{N}}^T \mathbf{x} + \mathbf{B}^T \boldsymbol{\lambda}] \quad (\text{D.7})$$

As can see from (D.5), the second derivative of \mathcal{H} respect to \mathbf{u} is equal to $\bar{\mathbf{R}}$ which is positive definite, therefore this is a minimum.

By setting

$$\boldsymbol{\lambda}_f = \mathbf{P}(t_f)\mathbf{x}(t_f) \quad (\text{D.8})$$

for the final time and the whole interval –see [72] the optimal control law results in

$$\mathbf{u} = -\bar{\mathbf{R}}^{-1} (\bar{\mathbf{N}}^T + \mathbf{B}^T \mathbf{P}) \mathbf{x} \quad (\text{D.9})$$

Where \mathbf{P} is a unique positive definite matrix that can be determine by taking the time derivative of equation (D.8) and solving the resulting differential

equation for \mathbf{P} , that is

$$\dot{\mathbf{P}}\mathbf{x} = \dot{\boldsymbol{\lambda}} - \mathbf{P}\dot{\mathbf{x}} \quad (\text{D.10})$$

Substituting (D.8) and (D.9) into (D.4) and substituting (D.9) into the state equation yields

$$\dot{\boldsymbol{\lambda}} = (\overline{\mathbf{N}}\overline{\mathbf{R}}^{-1}\overline{\mathbf{N}}^T - \overline{\mathbf{Q}})\mathbf{x} - (\mathbf{A} - \mathbf{B}\overline{\mathbf{R}}^{-1}\overline{\mathbf{N}}^T)^T \mathbf{P}\mathbf{x}$$

and

$$\dot{\mathbf{x}} = (\mathbf{A} - \mathbf{B}\overline{\mathbf{R}}^{-1}\overline{\mathbf{N}}^T)\mathbf{x} - \mathbf{B}\overline{\mathbf{R}}^{-1}\mathbf{B}^T \mathbf{P}\mathbf{x}$$

Substituting these two expressions into (D.10) the resulting equation for \mathbf{P} is

$$-\dot{\mathbf{P}} = \mathbf{A}^T \mathbf{P} + \mathbf{P}\mathbf{A} + \overline{\mathbf{Q}} - (\overline{\mathbf{N}} + \mathbf{P}\mathbf{B})\overline{\mathbf{R}}^{-1}(\overline{\mathbf{N}}^T + \mathbf{B}^T \mathbf{P}) \quad (\text{D.11})$$

Equation (D.11) is known as the Differential Riccati Equation (DRE). Because matrix \mathbf{P} is constant, $\dot{\mathbf{P}}$ is zero reducing equation (D.11) to an Algebraic Riccati Equation (ARE).

References

- [1] Charles Coulston Gillispie. *The Montgolfier Brothers and the invention of Aviation*. Princeton University Press, 1983.
- [2] The Free Encyclopedia Wikipedia. History of unmanned aerial vehicles. http://en.wikipedia.org/wiki/History_of_unmanned_aerial_vehicles, June 2011.
- [3] Jay Gundlach. *Designing Unmanned Aircraft Systems A Comprehensive Approach*. American Institute of Aeronautics and Astronautics, 2012.
- [4] URL http://en.wikipedia.org/wiki/List_of_unmanned_aerial_vehicles.
- [5] V. Gavrilets, B. Mettler, and E. Feron. Dynamic model for a miniature aerobatic helicopter. In *unknown*, page 22, unknown.
- [6] Mark B. Tischler. System identification method for aircraft flight control development and validation. Technical Report 95-A-007, NASA and US Army, Moffet Field, CA 94035-1000, October 1995.
- [7] Bernard Mettler, Takeo Kanade, and Mark B. Tischler. System identification modeling of a model-scale helicopter. page 25.
- [8] S. k. Kim and D. M. Tilbury. Mathematical modeling and experimental identification of an unmanned helicopter robot with flybar dynamics. *Journal of Robotics System*, 21:19, 2004.
- [9] S. K. Kim and D. M. Tilbury. Mathematical modeling and experimental identification of a model helicopter. *Submitted to the AIAA Journal of Guidance, Control, and Dynamics*, pages 1–33, 2000.

- [10] Marco La Civita, William Messner, and Takeo Kanade. Modeling of small-scale helicopters with integrated first-principle and system-identification techniques. *American Helicopter Society 58th Annual Forum*, page 25, June 2001.
- [11] John C. Morris, Michiel van Nieuwstadt, and Pascale Bendotti. Identification and control of a model helicopter in hover. In *American Control Conference Proceedings*, page 4, June 1994.
- [12] Seiji Hashimoto, Tomonori Ogawa, Shuichi Adachi, Anzhong Tan, and Gou Miyamori. System identification experiments on a large-scale unmanned helicopter for autonomous flight. In *Proceedings of the 2000 IEEE International Conference on Control Applications*, page 5, September 2000.
- [13] Byeongil Kim, Yushin Chang, and Man Hyung Lee. System identification and 6-dof hovering controller design of unmanned model helicopter. *JSME International Journal*, 4:9, 2006.
- [14] S. Suresh, M. Vijaya Kumar, S. N. Omkar, V. Mani, and Prasad Sampath. Neural networks based identification of helicopter dynamics using flight data. In *Proceedings of the 9th International Conference on Neural Information Processing*, volume 1, page 4.
- [15] David Hyunchul Shim, Hyoun Jin Kim, and Shankara Sastry. Control system design for rotorcraft-based unmanned aerial vehicles using time-domain system identification. In *Proceedings of the 2000 IEEE International Conference on Control Applications*, page 5, September 2000.
- [16] Bernard Mettler, Mark B. Tischler, and Takeo Kanade. System identification of small-size unmanned helicopter dynamics. In *Presented at the American Society 55th*, page 12, May 1999.
- [17] A. Budiyo, T. Sudiyanto, and H. Lesmana. First principle approach to modeling of small scale helicopter. In *ICIUS*, page 10, 2007.
- [18] Guowei Cai, Alvin K. Cai, Ben M. Chen, and Tong H. Lee. Construction, modeling and control of a mini autonomous uav helicopter. In *Proceedings of the IEEE International Conference on Automation and Logistics*, page 4, September 2008.

- [19] Guowei Cai, Ben M. chen, Kemao Peng, Miaobo Dong, and Tong H. Lee. Modeling and control system design for a uav helicopter. page 6.
- [20] Mark B. Tischler. System identification requirements for high-bandwidth rotorcraft flight control system design. page 9, Moffett Field. CA 94035-1000. U.S Army Aviation Research & Technology Activity.
- [21] Dongwon Jung and Panagiotis Tsiotras. Modeling and hardware-in-the-loop simulation for a small unmanned aerial vehicle. page 13. American Institute of Aeronautics and Astronautics.
- [22] Sepehr P. Khaligh. *Control-Oriented Modeling and System Identification for Nonlinear Trajectory Tracking Control of a Small-Scale Unmanned Helicopter*. PhD thesis, University of Alberta, Edmonton, Alberta, Canada, 2014.
- [23] Government of Canada. Transport canada. URL www.tc.gc.ca/eng/menu.hlmt.
- [24] M. Saffarian and F. Fahimi. A comprehensive kinematic analysis of a model helicopter's actuating mechanism. Technical report, University of Alberta, 2007.
- [25] NIMA. Department of defense world geodetic system 1984. Technical report, National Imagery and Mapping Agency, January 2003.
- [26] Daniel J. Biezad. *Integrated Navigation and Guidance Systems*. Education Series. American Institute of Aeronautics and Astronautics, 1801 Alexander Bell Drive, Reston, VA 20191, 1999.
- [27] Brian L. Stevens and Frank L. Lewis. *Aircraft Control and Simulation*. John Wiley & Sons, Inc., second edition, 2003.
- [28] B.R. Bowring. Transformation from spatial to geographical coordinates. *Surv. Rev.*, 23(3181):323/327, 1976.
- [29] Mathworks. Ecef position to lla, 2012. URL <http://www.mathwork.com/help/toolbox/aeroblks/ecefpositiontolla.html>.
- [30] Wikipedia. Geodetic systems, 2012. URL http://en.wikipedia.org/wiki/Geodetic_system.

- [31] Warren F. Phillips. *Mechanics of Flight*. Jhon Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, first edition, 2004.
- [32] Dan B. Marghitu. *Mechanisms and Robots Analysis with MATLAB*. Springer-Verlag London Limited, 2009.
- [33] A. R.S. Bramwell. *Helicopter Dynamics*. Edward Arnold Ltd., 3 East Street, Baltimore, Meryland 21201, U.S.A., first edition, 1976.
- [34] J. Seddon and Simon Newman. *Basic Helicopter Aerodynamics*. American Institute of Aeronautics and Astronautics, 1801 Alexandel Bell Drive, Reston, Va 20191, second edition, 2010.
- [35] O. Rand. A phenomonological modification for glauert’s classical induced velocity equation. *Journal of the American Helicopter Society*, 51(3):279–282, 2006.
- [36] H. Glauert and M. A. A general theory of the autogyro. Report and Memorandum No. 1111 1111, Director of Scientific Research Air Minister, November 1926.
- [37] Mohammad H. Sadraey. *Aircraft Performance Analysis*. VDM Verlag DR Muller, 2011.
- [38] Boundary layer theory. URL <http://www.freestudy.co.uk/fluid%20mechanics/t3203.pdf>.
- [39] Prieto Catalano, Meng Wang, Gianluca Iaccarino, and Parviz Morin. Numerical simulation of the flow around a circular cylinder at high reynolds numbers. *International Journal of Heat and Fluid Flow*, 24:463 – 469, 2003.
- [40] Meng Wang, Prieto Catalano, and Gianluca Iaccarino. Prediction of high reynolds number flow over a circular cylinder using les with wall modeling. Technical report, Center for Turbulent Research, 2001.
- [41] K. Klausmann, T. Wacker, and B. Ruck. The flow around porously coated circular cylinder. 2011.
- [42] Anatol Roshko. Experiments on the flow past a circular cylinder at very hig reynolds number. Technical report, Guggenheim Aeronautical Laboratory, California Institute of technology, 1960.

- [43] S. P. Singh and S. Mittal. Flow past a cylinder shear layer instability and drag crisis. Technical report, Indian Institute of Technology. Department of Aerospace Engineering.
- [44] C. Wieselsberger. New data on the laws of fluid resistance. Technical report, National Advisory Committee for Aeronautics, 1922.
- [45] Jonathan Lucas Heseltine. Flow around a circular cylinder with a free end. Master's thesis, University of Saskatchewan, Department of Mechanical Engineering, 2003.
- [46] Hongwei an, Liang Cheng, and Ming Zhao. Steady streaming around a circular cylinder near a plane boundary due to oscillatory flow. *Journal of Hydraulics Engineering*, pages 23–33, 2011. doi: 10.1061/(ASCE)HY.1943-7900.0000258.
- [47] Momchilo M. Zdravkovich. A critical remark on use of the drag coefficient at low Reynolds numbers. Technical report, Recueil des travaux de l'Institut Mathematique., 1979.
- [48] Crossbow-Technology-Inc. *440 Series User's Manual*, December 2008. Document 7430-0131-01 Revision C.
- [49] Omni-Instruments. *Servo Switch/Controller Manual*, March 2005.
- [50] Naser Hossein Motlagh. Frequency hopping spread spectrum: An effective way to improve wireless communication performance, advanced trends in wireless communications. Technical report, Department of Information Technology, Vaasa University of Applied Sciences Finland, 2011.
- [51] K. H. Torvmark. Frequency hopping systems. application note an014. Technical report, Chipcon Products from Texas Instruments.
- [52] Theresa C. Maxino and Philip J. Koopman. The effectiveness of the checksums for embedded control networks. *IEEE Transactions on Dependable and Secure Computing*, 6(1):59–72, January-March 2009.
- [53] Theresa Maxino. Revisiting Fletcher and Adler checksums. In *DSN 2006 Student Forum*, 2006.

- [54] Plettenberg[®]. Hp 370/40/ heli datasheet, 2005. URL http://www.plettenberg-motoren.com/Datenblaetter/370_40/370-40-A2_Heli.pdf.
- [55] Schulze. Electronic speed controller. URL [www.http://rc.runryder.com/helicopter/t536454p1/](http://www.rc.runryder.com/helicopter/t536454p1/).
- [56] Limited Yineng-China-Holdings-Group-Co. Lipo battery, 2012. URL <http://www.e6180.com/>.
- [57] Hokuyo-Automatic-Co.-LTD. *Scanning Laser Range Finder URG-04LX Specifications*, October 2005.
- [58] National-Semiconductors. *Simple Switcher[®] 5A Step-Down Voltage Regulator with Adjustable Current Limit*, February 2008.
- [59] Advanced-Digital-logic[™]. *ADL855PC Manual Revision 3.2*, .
- [60] Advanced-Digital-logic[™]. *ADLPS104CF Manual Rev. 1.3*, .
- [61] Diamod-System-Corporation. *EMERALD-MM-8 8-Channel Serial port PC/104 Module User Manual V2.42*, 2005.
- [62] Natural-Resources-Canada. Magnetic declination calculator, 2012. URL <http://geomac.nrcan.gc.ca/cal/mdcal-eng.phb>.
- [63] Earth-Measurement-Consulting. Gps accuracy and limitations, 2005. URL http://earthmeasurement.com/GPS_accuracy.html.
- [64] Novatel. Gps position accuracy measures. Technical Report APN-029 Rev 1, Novatel Position Leadership, December 2003.
- [65] Ingo Harre. A standard algorithm for the determination of position errors by the example of gps with and without ‘selective availability. Technical Report ??, STN Atlas Elektronik, May 2001.
- [66] Daniel S Wilkis. *Statistical Methods in the Atmospheric Sciences, Volume 100 (International Geophysics Series)*. Academic Press, The Boulevard, Langford Lane, Kidlington, Oxford, OX512GB, UK, third edition, 2011.
- [67] Hassan K. Khalil. *Nonlinear Systems*. Prentice Hall, 2002.

- [68] Richard C. Dorf and Robert H. Bishop. *Modern Control Systems*. Prentice Hall, 2011.
- [69] A. Fermelia, D. A. Gyorog, and V. J. Flanigan. Helicopter motion: Equation linearization. *IEEE Transaction on Aerospace and Electronics systems*, AES-12:767–782, 1976.
- [70] A. J. Jordan. Linearilinear of non-linear state equation. *Bulletine of the polish academy of sciences*, 54:63–73, 2006.
- [71] Donald E. Kirk. *Optimal Control Theory. An Introduction*. Dover Publications, Inc., 2004.
- [72] Robert F. Stengel. *Optimal Control and Estimation*. Dover Publications, Inc., 1994.
- [73] Surgud Skogetad and Ian Postlethwaite. *Multivariable Feedback Control, Analysis and Design*. John Wilwy & Sons, 1996.
- [74] D. G. Luenberger. Observing the states of a linear system. *IEEE Transactions on Military Electronics*, pages 70–80, 1964.
- [75] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill International Editions, 1984.
- [76] Charles L. Phillips and H. Troy Nagel. *Digital Control System Analysis and Design*. Prentice Hall, 1995.
- [77] Katsuhiko Ogata. *Discrete-Time Control Systems*. Prentice Hall, 1995.
- [78] Katsuhiko Ogata. *Ingenieria de Control Moderna*. Prentice Hall, 1993.
- [79] Benjamin C. Kuo. *Sistemas de Control Automatico*. Prentice Hall, 1996.
- [80] Chi-Tsong Chen. *Linear Systems Theory and Design*. Oxford, 1999.
- [81] J. J. D’Azzo and C. H. Houpis. *Feedback Control Systems, Analysis & Synthesis*. McGraw-Hill International Editions, 1966.
- [82] Horacio J. Marquez. *Nonlinear Control Systems, Analysis and Design*. Wiley Intescience, 2003.