

Using Presagis Simulation Software to Model UAV Aircraft in a Humanitarian Mission Configuration

January 2016

by

Aaron Sotto, B.Eng, Ryerson University, 2013

A Major Research Project Paper
presented to Ryerson University

in partial fulfillment of the
requirements of the degree of
Master of Engineering
in the program of
Aerospace Engineering

Toronto, Ontario, Canada, 2016
©Aaron Sotto 2016

Using Presagis Simulation Software to Model UAV Aircraft in a Humanitarian Mission Configuration

Master of Engineering, 2016
Aaron Sotto
Aerospace Engineering, Ryerson University

Abstract

Aircraft simulation software was used together to simulate a humanitarian variant of the MQ-9 Reaper drone as well as its longitudinal stability response upon dropping an aid payload. This project derives stability derivatives from the MQ-9 dimensions using the mass moments of inertia and approximate airfoil shape using Athena Vortex Lattice (AVL) code. The stability derivatives, aircraft properties, weights and control systems were modelled with Presagis FlightSim 14 to approximate the MQ-9 flight model. A graphics model was also built using Presagis Creator and the flight model and graphics model were unified into a virtual environment. Its longitudinal short period and phugoid responses as well as the lateral Dutch mode after dropping a 200 kg payload was recorded and analysed. The older Ryan Navion was also modelled using the same method which was used to model the MQ-9. The same dynamic responses were compared to real Navion flight test and calculated data in order to validate the aforementioned modelling method.

List of Tables

3.1	Radii of gyration values for a single propeller aircraft [17]	19
3.2	Navion calculated mass moments of inertia	19
3.3	Navion AVL-computed longitudinal stability derivatives comparison	20
3.4	Navion AVL-computed lateral stability derivatives comparison	20
3.5	MQ-9 AVL-computed longitudinal and lateral stability derivatives	22
3.6	MQ-9 AVL-computed forces and moments at $\alpha=0$ deg.	22
5.1	Error between FLSim and real Navion aircraft	51

List of Figures

1.1	Presagis M&S Program Suite integration	2
1.2	Screenshot of Creator Models from the Presagis website [16]	3
1.3	Screenshot of an F-16 using an FLSim flight model [16]	3
1.4	Screenshot of an environment built with Stage [16]	4
1.5	Logic diagram for modelling aircraft using the Presagis 2014 Suite	5
2.1	MQ-9 dims and flight	6
2.2	UN Salex Falco unarmed surveillance UAV [13]	7
2.3	Northrop-Grumman RQ-4 and Israel Aerospace Industries Heron drones	8
3.1	Completed 3-view construction planes	9
3.2	Hierarchy pane	10
3.3	Creating a rectangular face/plane	10
3.4	Building construction planes for each aircraft view	11
3.5	Combining coplanar faces	11
3.6	Applying textures to a face	12
3.7	Applying a texture to a selected body	12
3.8	Creating a finer grid spacing	13
3.9	Airfoil polygon	14
3.10	Using the wall plus tool to extrude the airfoil	14
3.11	Applying the <i>lattice deform</i> tool to the wing	15
3.12	Selecting lattice nodes to deform	15
3.13	Deforming shape with selected lattice nodes	16
3.14	MQ-9 fuselage model undergoing lattice deformation	16
3.15	CAD 3-view of Guardian in Creator 14	17
3.16	MQ-9 Guardian Creator Model with fixed Radar Payload	17
3.17	Ryan Navion dimensions	18
3.18	MQ-9 Reaper geometry in AVL	21
3.19	Eppler 374 airfoil	21
3.20	Flight Sim 14 aircraft reference lines [8]	23
3.21	Flight Sim 14 <i>Modeler</i> mode	24
3.22	Flight Sim 14 <i>Run-time</i> mode	25
3.23	The weight and balance screen with values for the MQ-9	26

3.24	Landing gear definition and braking friction curve	27
3.25	Gear installation specification	27
3.26	Weight and balance of components on FLSim Guardian	28
3.27	The <i>propulsion</i> menu in <i>Modeler</i> mode	29
3.28	The <i>Engines</i> modelling tab in <i>Modeler</i> mode	30
3.29	Locating and defining the wing and its aerodynamic center	30
3.30	autothrottle control logic	31
3.31	longitudinal control surface logic	31
3.32	yaw control logic	31
3.33	lateral control surface logic	32
3.34	Total C_L vs α curve for the MQ-9 Guardian	32
3.35	Defining payload properties	33
3.36	Selecting and saving payload configuration	33
3.37	Applying the payload configuration to the aircraft	34
3.38	Tracking aircraft climb rate and α as a function of time	35
3.39	FLSim simulation cockpit view	36
3.40	New project (database) in Stage	37
3.41	Defining a unit platform	38
3.42	Locating Toronto in Stage	39
3.43	Defining aircraft	40
3.44	Guardian aircraft platform being selected	41
3.45	Defining the aircraft model	42
3.46	Defining the aircraft configuration	42
3.47	Defining the MQ-9 initial conditions	43
3.48	Stage in the scenario editor mode	44
3.49	Selecting the Guardian unit for viewing	44
3.50	Untextured Guardian in flight in Stage	45
4.1	Flight Sim 14/Creator Guardian in Stage environment	46
4.2	A Canadian CP-140 dropping a SKAD package	46
5.1	FLSim Navion Phugoid mode (bottom) comparison with real Navion [2] (top)	48
5.2	FLSim Navion short period mode (bottom) comparison with real Navion [2] (top)	49
5.3	FLSim Navion Phugoid mode (bottom) comparison with real Navion [6] (top)	50
5.4	MQ-9 phugoid motion with aid payload	52
5.5	MQ-9 phugoid motion with dropped aid payload	53
5.6	MQ-9 short period mode with payload	54
5.7	MQ-9 Dutch roll response with payload	54
A1	Ryan Navion Derivatives from Nelson [15]	58
A2	Ryan Navion dimensions and mass properties [15]	58
A3	Ryan Navion NACA-Calculated Lateral Stability Derivatives [19]	59
A4	Ryan Navion NACA-Calculated Longitudinal Stability Deriva- tives [19]	60
A5	Navion statistically-derived longitudinal stability derivatives [22]	61
A6	Navion statistically-derived lateral stability derivatives [22]	62

Contents

Abstract	ii
List of Tables	iii
List of Figures	iii
Nomenclature	vi
1 Introduction	1
Purpose	1
Modelling Unmanned Aerial Vehicles in New Roles	1
Presagis 2014 Modelling and Simulation Suite	2
2 Benchmark Aircraft	5
3 Methods of Modelling	8
Creator 14 Model	8
Athena Vortex Lattice - AVL	18
Flight Sim 14 Model	22
Stage 14 Integration	37
4 Humanitarian Configuration Changes	45
5 Method Validation and Evaluation of Guardian Flight Model	47
Navion Flight Model Validation	47
MQ-9 Guardian Dynamic Modes	52
6 Conclusion	55
Future Work and Improvements	55
Bibliography	56
Appendices	58

Nomenclature

Symbol	Units	Description
α	<i>deg or rad</i>	Angle of attack
β	<i>deg or rad</i>	Side slip angle
δ	–	Logarithmic decrement
ϵ	<i>deg or rad</i>	Downwash angle
ζ	–	Damping Ratio
η	–	Dimensionless tail efficiency factor
θ	<i>deg or rad</i>	Pitch angle
κ	–	Wing airfoil lift curve to ideal lift curve ratio
σ	<i>deg or rad</i>	Sidewash angle
ϕ	<i>deg or rad</i>	roll angle
A	–	Wing aspect ratio
a_h	rad^{-1}	Horizontal tail lift curve slope
a_v	rad^{-1}	Vertical tail lift curve slope
a_w	rad^{-1}	Wing lift curve slope
\bar{c}	m	Mean aerodynamic wing chord
C_D	–	Drag coefficient
C_L	–	Lift coefficient
$C_{L\alpha}$	rad^{-1}	$\frac{\partial C_L}{\partial \alpha}$
$C_{Y\beta}$	rad^{-1}	$\frac{\partial C_L}{\partial \beta}$
C_l	–	Rolling moment coefficient
$C_{l\beta}$	rad^{-1}	$\frac{\partial C_l}{\partial \beta}$
C_{lp}	rad^{-1}	$\frac{\partial C_l}{\partial \frac{pb}{2V}}$
C_{lr}	rad^{-1}	$\frac{\partial C_l}{\partial \frac{rb}{2V}}$
C_m	–	Pitching moment coefficient
$C_{m\alpha}$	rad^{-1}	$\frac{\partial C_m}{\partial \alpha}$
C_{mq}	rad^{-1}	$\frac{\partial C_m}{\partial \frac{p^2}{2V}}$
C_n	–	Yawing moment coefficient
$C_{n\beta}$	rad^{-1}	$\frac{\partial C_n}{\partial \beta}$
C_{np}	rad^{-1}	$\frac{\partial C_n}{\partial \frac{pb}{2V}}$
C_{nr}	rad^{-1}	$\frac{\partial C_n}{\partial \frac{rb}{2V}}$
D_f	m	Fuselage depth
I_{xx}	$kg \cdot m^2$	Moment of inertia about x -axis
I_{yy}	$kg \cdot m^2$	Moment of inertia about y -axis
I_{zz}	$kg \cdot m^2$	Moment of inertia about z -axis

Ma	–	Mach number
S_h	m^2	Horizontal tail area
S_w	m^2	Wing area
V_h	m^3	Horizontal tail volume
V_v	m^3	Vertical tail volume
x_{cg}	m	Centre of gravity location from wing leading edge
x_{NP}	m	Neutral point location from wing leading edge
Z_{wf}	m	Vertical tail height above wing

1 Introduction

Purpose

The objective of this project was to create a well-approximated flight model of a relatively new and proprietary aircraft for use in the simulation of a Search and Rescue (SAR) mission. Validation of the modelling method using Presagis modelling and Simulation software by CAE simulation and training corporation in Montréal was provided by first modelling a well-documented aircraft, the Ryan Navion. The Navion stability derivatives were taken from flight tests, wind-tunnel data and calculations from multiple sources. Its flight model was then built and flown in a simulated environment and compared to data recorded from real world Navion flight tests. A comparison was made between the Navion flight model constructed in this project and the data collected. This created the basis to build the MQ-9 Reaper aircraft using the same methods. A humanitarian configuration of the MQ-9 Reaper, the Guardian, was built using Presagis software. A symmetrical 200 kg payload was attached to a centerline hard-point and dropped. The aircraft's long period phugoid, short period and lateral Dutch roll responses were recorded and observed. The capability for the modelling method documented in this report displays that useful and realistic flight models with good approximations to performance and stability can be built for aircraft with known dimensions and weights but whose performance information is not readily available.

Modelling Unmanned Aerial Vehicles in New Roles

Unmanned Aerial Vehicles (UAV's) are versatile, remote-controlled aircraft which are used primarily by military organizations and increasingly by paramilitary and police departments around the world. Their cheap operating cost, increasing reliability and lack of onboard crew means they can be used often without endangering the life of the aircraft operator or friendly forces. The work accomplished and documented in this project illustrates a possible humanitarian role for UAV's in which they can be just as effective as they are in an offensive, military role. UAV's offer the same advantages in humanitarian Search And Rescue (SAR) roles as they do as attack drones. Their lack of onboard operator means that they can be flown for long hours without fatiguing a crew, while operating in areas that could be dangerous for larger aircraft with humans onboard. Additionally, large numbers of these SAR UAV's can be deployed for greater coverage of an area.

Adapting a UAV for a new role requires time and funding for new variant development, but software like Presagis can be used to simulate aircraft in new roles for which a UAV was not originally designed. The work documented in this report develops and validates a method by which an aircraft can be modelled.

Presagis 2014 Modelling and Simulation Suite

The aircraft in the project were modelled primarily with the Presagis 2014 modelling and Simulation (M&S) software suite, designed to model, simulate and evaluate aircraft performance scenarios and human-machine interface (HMI) systems. Presagis software tools have been in development as a part of CAE simulation and training corporations located in Montréal, Canada. Presagis software allows the user to develop aircraft graphic polygon models in Creator 14 as well as their flight models in Flight Sim 14 (FLSim 14).

Terra Vista is another program created by Presagis which generates terrain for the simulation scenario and allows the user to customize a map. Vega Prime is the integration of all systems and is the primary visualization tool for the Presagis suite. No Terra Vista or Vega Prime files were modified directly during the course of this project, though the loading and launching of simulation environments in Stage automatically used Vega Prime default terrain and Vega Prime simulation screen for visualization. Figure 1.1 shows the overall integration of Presagis M&S programs.

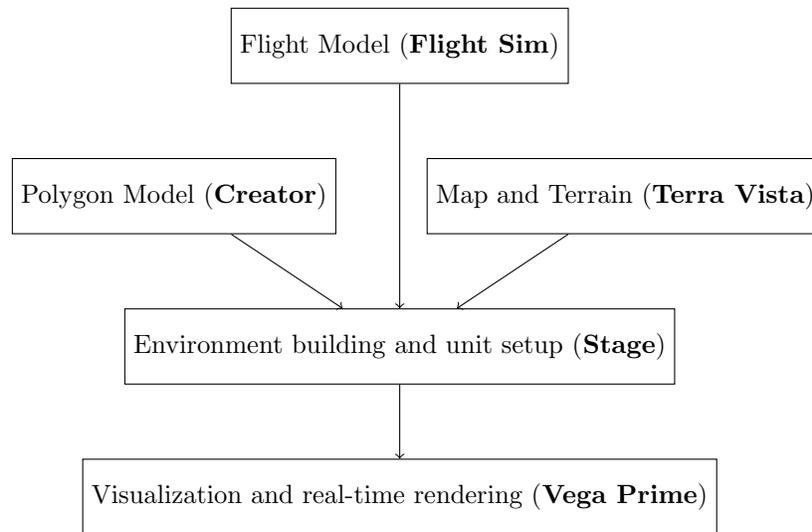


Figure 1.1: Presagis M&S Program Suite integration



Figure 1.2: Screenshot of Creator Models from the Presagis website [16]

Creator is built with real-time rendering in mind. The polygon model can be made extremely accurate requiring thousands of polygons or made very basic allowing for faster computing time during render. The versatility of Creator is such that a very detailed model can be adjusted into a lower polygon version. With Creator the user can:

- build simple objects using versatile, virtual "sculpting" tools
- adjust polygon faces, edges and properties to create the desired image
- apply textures to models for a realistic and detailed model skin
- model buildings, train stations, airports, cities and other ground objects for a realistic simulation environment



Figure 1.3: Screenshot of an F-16 using an FLSim flight model [16]

The user has the freedom to tailor specific aircraft aerodynamic, avionics and dimensional properties in order to make a flight model. FLSim can build a flight model as accurate as the user data allows. Variables can be approximated where there is data missing, but can be made very accurate over an aircraft flight envelope if enough data exists. This makes FLSim a powerful and versatile tool for creating working models of different fidelity levels. Features of FLSim 14 include the ability to:

- make and manipulate stability and force coefficient curves versus any variable like α , Ma
- create control logic diagrams that use user-defined curves for aircraft stability and control

- define number, locations and changes in properties versus time for fuel tanks, engines and landing gear
- define aircraft dimensions, component centers of gravity, mass moments of inertia and more
- test aircraft in flight through automated inputs, or through manual flight
- create payload drop scenarios by defining an additional load



Figure 1.4: Screenshot of an environment built with Stage [16]

Stage 14 is the simulation environment program whose primary function is to integrate many aircraft, ground vehicles and other systems in a real time, automated or interactive simulated scenario. With Stage, the user can observe the interactions and outcomes between simulated units, their environment and the surrounding structures and obstacles. Since none of the aircraft modelled in this project was interacting with other units, Stage was only used to observe the virtual aircraft as the union of graphic and flight models.

Figure 1.5 illustrates the process behind building and simulating an aircraft using the Presagis 2014 M&S suite. The stability must be acquired through calculation or research. The control system logic - steady state equations governing aircraft control as well as control system block diagram logic with respect to stability derivatives - must also be attained and input into FLSim. Aircraft mass, mass moments of inertia and component definition (i.e. fuel tank locations, engine center of gravity) must also be known. Finally, a flyable aircraft model is constructed and its various dynamic modes can be observed and recorded. Separately, the aircraft graphic model is built from aircraft dimensions in Creator and integrated with the FLSim flight model into the Presagis suite via Stage 2014. Stage is a simulation environment through which aircraft can be observed. Stage can simultaneously simulate various terrain features as well as ground-based and airborne models.

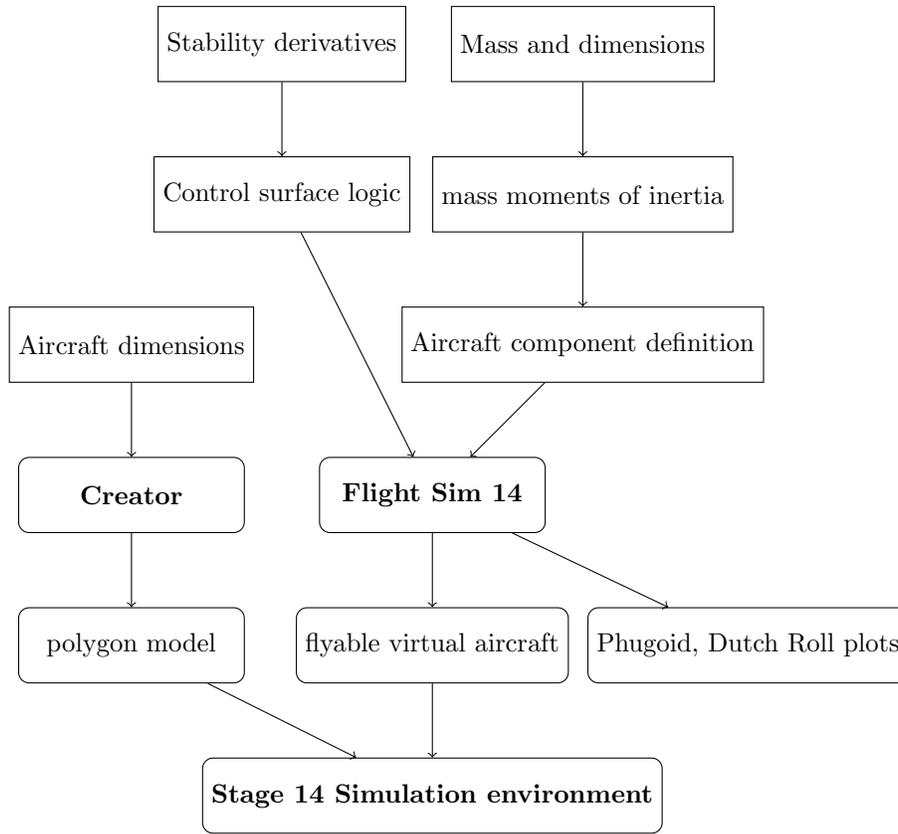
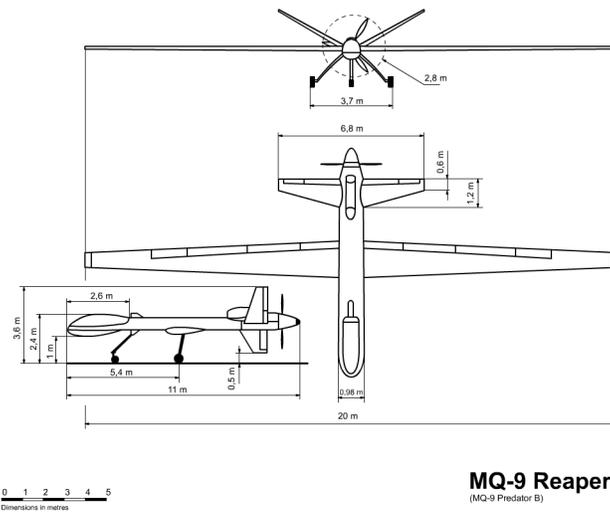


Figure 1.5: Logic diagram for modelling aircraft using the Presagis 2014 Suite

2 Benchmark Aircraft

The General Atomics MQ-9 Guardian aircraft is a modified, civil maritime patrol version of the military MQ-9 Reaper attack and surveillance drone. The MQ-9 was chosen as an possible Search and Rescue aircraft alternative to crewed helicopters and search planes. It was also considered more suitable than the larger RQ-4 Global Hawk and smaller Israel Aerospace Industries Heron. The Guardian has proven its ability to fly with a large radar payload system attached to a centerline hardpoint under and behind the wings which could possibly hold some kind of rescue payload like a raft or supplies. The Guardian's dimensions are nearly identical to those of the Reaper, with particular exception to the centerline-mounted avionics set.



(a) MQ-9 Reaper dimensions in metres [25]



(b) Guardian in flight [9]

Figure 2.1: MQ-9 Reaper dimensions and Guardian variant

The MQ-9's ability to loiter for a large amount of time, carry a payload, survey and cruise at medium altitudes makes it an ideal candidate as an SAR drone. An MQ-9 would be able to deliver cargo to victims stranded in an ocean or injured people in an area hit by a tornado. The payload can consist of medical supplies, food and water which can help victims survive until they can be extracted if necessary. UAV's are already being used by the United Nations to survey disaster-stricken areas and to monitor aid supply distribution [14]. One such drone is the Italian-built Saalex Falco shown in Figure 2.2 which was used in early 2013 to survey and support operations in the Democratic Republic of Congo.



Figure 2.2: UN Saalex Falco unarmed surveillance UAV [13]

The other two aircraft considered were the RQ-4 Global Hawk and the IAI Heron. Both aircraft have been used in service since the 1990's to early 2000's and have served with renowned air forces for extremely long loiter surveillance missions. The RQ-4 is powered by a turbofan and is designed primarily for high altitude flight. The RQ-4 has the capability to hold a multitude of different sensors, but is considered too large for the SAR mission. It would certainly be able to carry an aid package but its large size and significant aspect ratio would make it cumbersome to operate, especially on small or rough airfields. According to IAI, the Heron is capable of loitering for 36 hours; nine more hours than the MQ-9. It is far smaller however, and is unlikely to carry as much payload as the MQ-9.



(a) RQ-4 Global Hawk [24]



(b) IAI Heron [12]

Figure 2.3: Northrop-Grumman RQ-4 and Israel Aerospace Industries Heron drones

SAR aircraft therefore have to be small enough to be able to take off of small or ill-prepared runways while being large enough to fly quickly over a large distance with a sizable payload. Such aircraft can be used for disaster relief scenarios, possibly flying in conditions and areas that would too dangerous for an aircraft with a human crew. Of the three aircraft, the MQ-9 is the most optimized for the SAR mission.

3 Methods of Modelling

The goal of the modelling method presented in this project is to demonstrate the creation of a realistic flight model of a given aircraft using Presagis software and the Athena Vortex Lattice (AVL) executable which calculates stability derivatives and flight forces based on geometry and mass properties. The aircraft was subjected to validation by first creating a well-known, flight-tested aircraft whose information is readily available. This project aims to build the MQ-9 Reaper/Guardian which has proprietary performance and airfoil information, while the old 1947 Ryan Navion serves as the aircraft which validates the MQ-9 flight model construction method.

The Presagis Simulation suite provides the user with the ability to model an aircraft flight model as well as its graphic polygon representation. In order to accurately find the stability derivatives of an aircraft configuration of a certain mass, geometry and mass moments of inertia, the Athena Vortex Lattice (AVL) executable can be used to find the aircraft stability derivatives. These values are plotted in Presagis FLSim. The graphics model can then be assigned to the flight model in the Presagis Stage environment simulator, allowing the user to observe and plot properties of the aircraft in flight.

Creator 14 Model

The MQ-9 Guardian polygon model was constructed using the Presagis Creator 14 executable. This 2014 version of Creator allows construction of polygon

models using manipulation. Creator is a polygon modelling program capable of creating complex geometries optimized for a flight simulator environment, where computing time is a valuable resource. Multiple static and moving objects must be modelled without compromising computational lag and Creator is capable of building models around that necessity.

The method used to create graphic models involved first creating three orthogonal planes, each of which would be constructed with their dimensions corresponding to the maximum dimensions of the aircraft in that plane. These planes would then be textured with images of the MQ-9 from different perspectives. The three planes would then serve as a template upon which the aircraft is constructed. Figure 3.1 illustrates a completed plane structure that can be used as a visual reference when building the aircraft model.

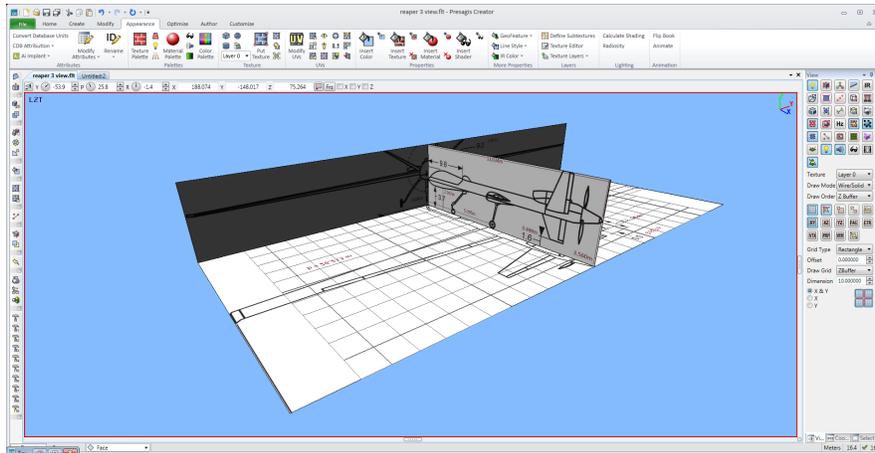


Figure 3.1: Completed 3-view construction planes

It is imperative that the view planes are designated as a separate object or group from the aircraft that is to be built. The user may pull the part hierarchy list up by left-clicking and dragging the pane border from the bottom of the screen. The user can then select the parent group (already selected by default) by holding *Alt* then left-clicking the desired group. Any body built next becomes a part of the selected parent group. By using the hierarchy, the user is able to create new groups, objects and other bodies independent from each other. This allows the user to operate on a selected body without affecting other parts of the model. Figure 3.2 shows a hierarchy tree with a database at the top, followed by 2 groups and 2 objects. Bodies such as these groups, objects and faces can be constructed by selecting the *Create* tab and clicking the button corresponding to the desired feature.

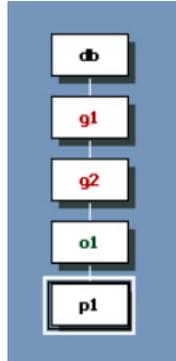


Figure 3.2: Hierarchy pane

When the MQ-9 is viewed from the side, its length is about $l = 11$ metres with a height of $h = 3.6$ metres. Viewed from the top, the aircraft's maximum dimensions are its length and its semi-span, $\frac{b}{2} = 10$ metres. Therefore, a plane was constructed on the XZ -plane by selecting the XZ -plane on the *View* menu on the right of the screen then using the *polygon* tool under the *Create* tab with dimensions 3.6 m by 11 m. The YZ -plane was then selected on the *View* pane, and a plane with dimensions 3.6 m by 10 m was created and attached to the first plane. Finally, a 10m by 11 m plane was constructed on the XY -plane.

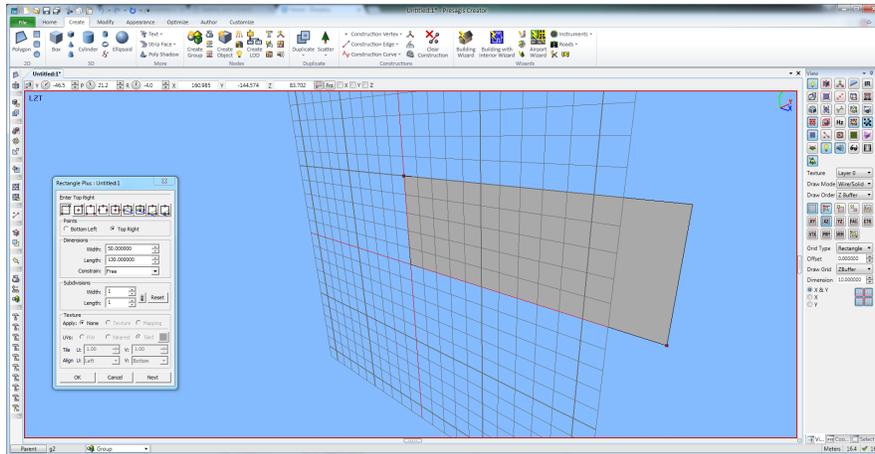
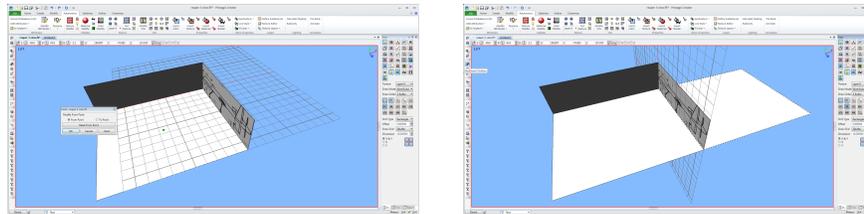


Figure 3.3: Creating a rectangular face/plane

It was necessary to mirror the second and third planes about the first in order to apply the front view and top view images in their entirety to the construction planes. Creator selection was put in face mode by selecting *face* in the expandable list at the bottom of the Creator screen. The third plane was selected by left-clicking it and then copied and pasted using the *Control + C*

and *Control + V* keyboard shortcuts. A new window appears that prompts the user to copy and paste the new plane in place or move the copy to another location. The plane was kept in its place. Upon its creation, the copied plane is automatically selected and then it is mirrored about the first plane using the *mirror* tool under the *Modify* tab. Figure 3.4 illustrates the copying and mirroring process.



(a) Copying the third plane

(b) Mirroring the copied plane

Figure 3.4: Building construction planes for each aircraft view

The original third plane and its copy were both selected by holding *Shift* and selecting both. Right clicking anywhere in the virtual construction space opens a modification menu as shown in Figure 3.5. The same process was done for the second plane.

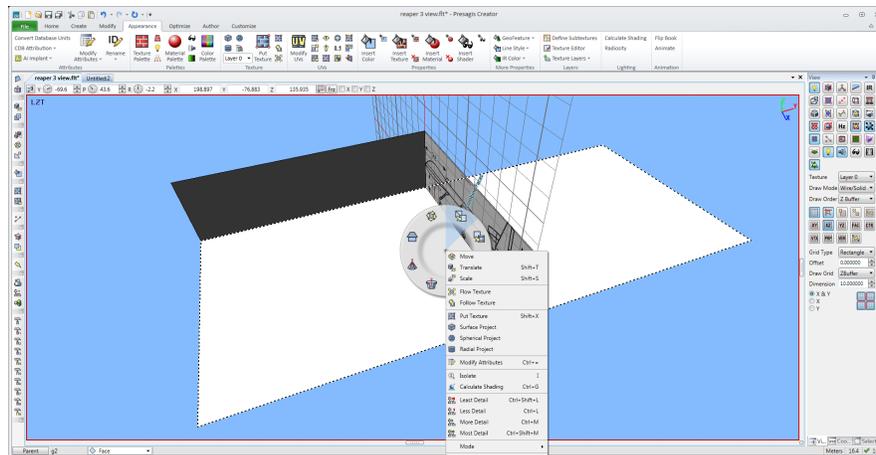
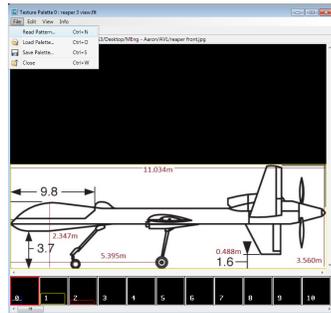


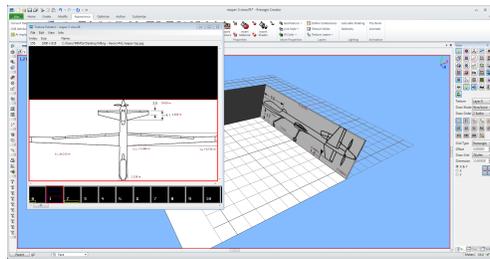
Figure 3.5: Combining coplanar faces

The newly combined faces could now be textured with the aircraft view pictures. Under the *Appearance* tab, the *Texture Palette* was chosen. A *Texture Palette* window appears and the user can then select any .jpeg or .png image by clicking a frame on the bottom of the palette window. The user then clicks *File > Read Pattern* as shown in Figure 3.6 and selects the desired image from a computer drive. The user may choose to read all the necessary patterns and

assign them to a different frame on the bottom of the texture palette window. The chosen image is then selected by clicking its corresponding frame and the texture palette window is closed.



(a) Loading image to the palette



(b) Selecting loaded texture

Figure 3.6: Applying textures to a face

A face in the geometry is selected and then the *Put Texture* tool is selected. The *Put Texture* window opens indicating options for configuring the image to the selected geometry. A thumbnail of the selected image appears in this window indicating three points (indicated in red, green and blue). When the user clicks any point in the geometry, this point corresponds to the red point in the thumbnail. The next two clicks correspond to the green and blue points in the thumbnail. This allows the user to apply the texture in the desired fashion to the selected faces, shown below in Figure 3.7.

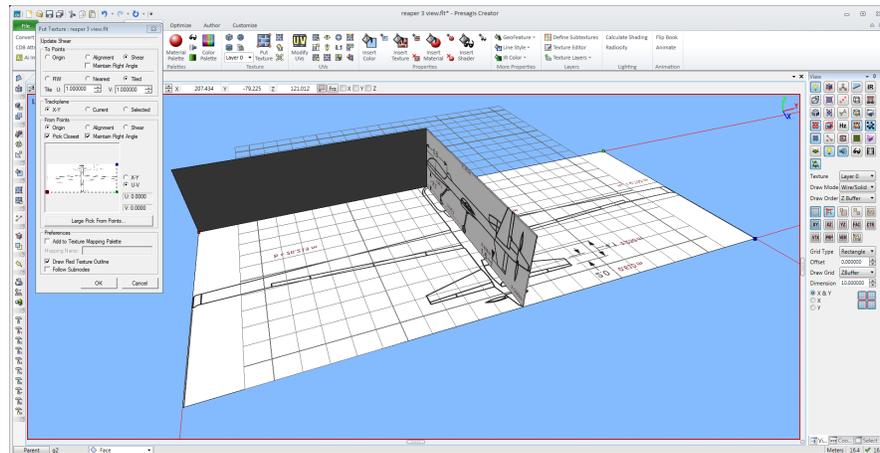


Figure 3.7: Applying a texture to a selected body

The images are then used to build one half of the aircraft. After this point, the entire model can be mirrored about the appropriate plane to complete the

entire aircraft model. The MQ-9 model was built beginning with the wing. The XZ -plane was selected so that a polygon could be drawn on it. A default feature of Creator is that any point designated by the user will snap to a grid corner. If the user creates a rectangle and clicks in the construction space, the clicked point will appear on the closest grid corner. Therefore, the view pane grid dimensions were changed into a finer spacing as shown in Figure 3.8. The *polygon* tool was then selected to trace an airfoil on the MQ-9 side view image.

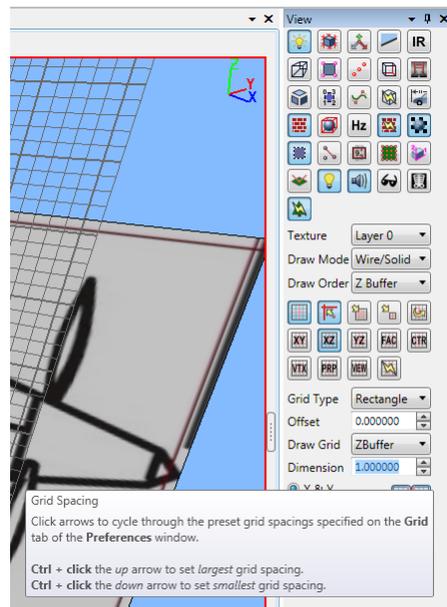


Figure 3.8: Creating a finer grid spacing

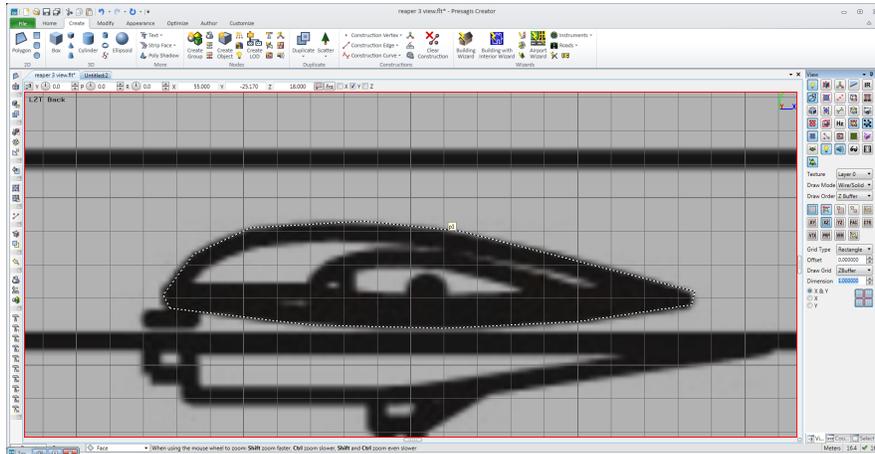


Figure 3.9: Airfoil polygon

The airfoil polygon can now be extruded using the *wall plus* tool under the *Modify* tab as shown in Figure 3.10. It is recommended that at this point, the user selects a new parent node in the hierarchy tree so that the newly constructed wing is built into a different group as the construction planes. When the user puts Creator into *object* mode and then selects the airfoil drawing as the object, the airfoil can now be extruded by clicking the *wall plus* tool. A *wall plus* window appears with the option of creating sections in the direction of extrusion. It is necessary to create sections so that the wing can be molded later using the *deform lattice* tool, also under the *modify* tab.

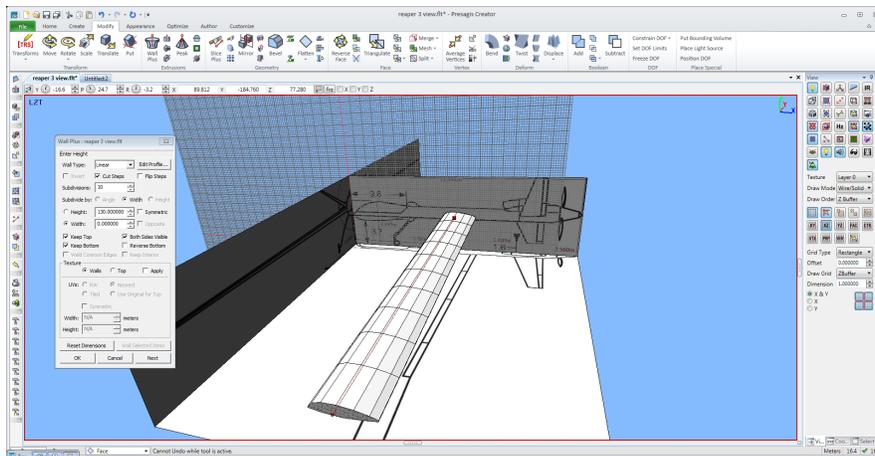


Figure 3.10: Using the wall plus tool to extrude the airfoil

With the wing constructed, manipulation is now possible using the *lattice*

deform. This tool encases the selected object in a three-dimensional gridded lattice. The user must first divide the lattice grid into as many parts as necessary before deforming the lattice. This can be done by increasing or decreasing the number of sections in the *X*, *Y* or *Z* directions, as indicated in the *lattice deform* window. As the lattice is manipulated in shape, the encased object is manipulated to the same degree. The deformation process is shown in Figures 3.11 to 3.12.

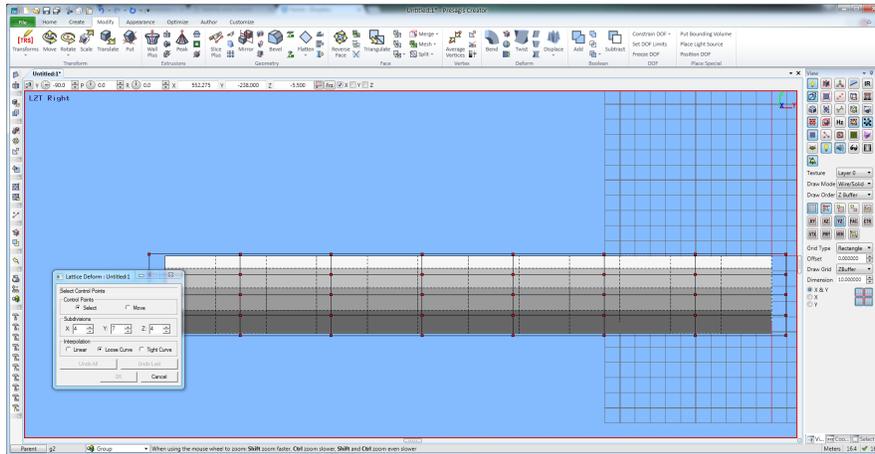


Figure 3.11: Applying the *lattice deform* tool to the wing

The user can then select a single node by left clicking it or multiple nodes by left clicking and dragging a box around multiple nodes. The nodes can then be manipulated by selecting the *move* option in the lattice deform window.

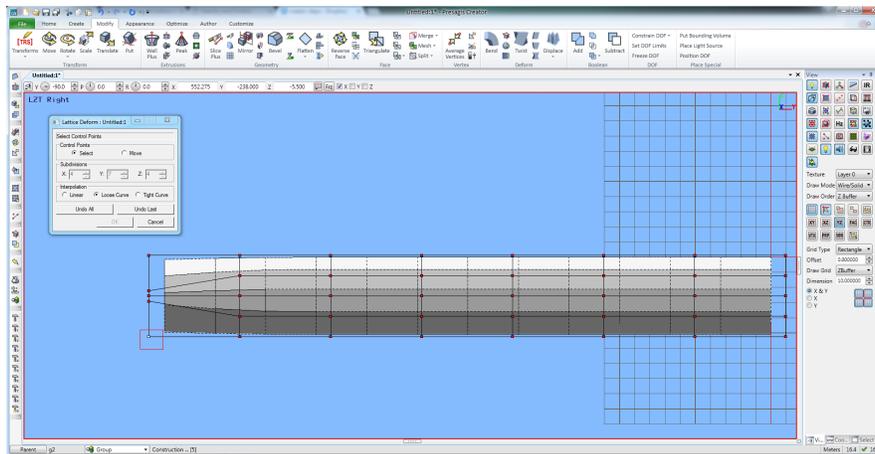


Figure 3.12: Selecting lattice nodes to deform

The user can choose to move the nodes by dragging them with the cursor, which will move to the nodes to the grid corner closest to the cursor position. Figure 3.13 illustrates how an object is deformed by moving nodes.

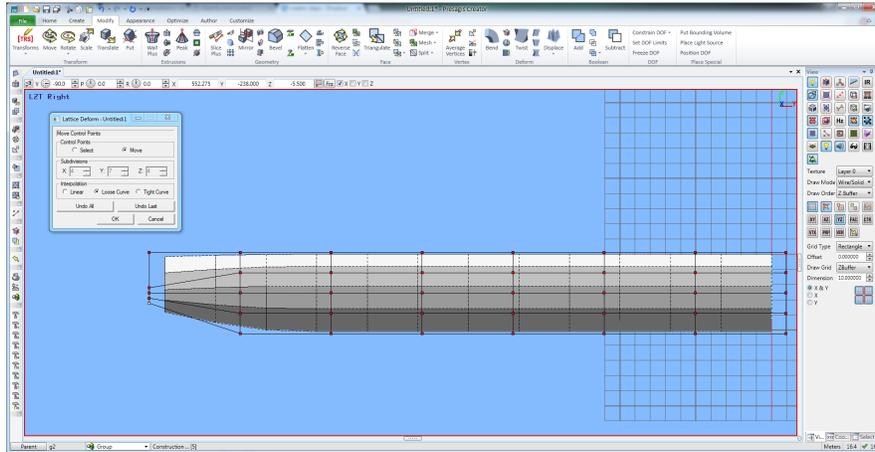


Figure 3.13: Deforming shape with selected lattice nodes

Lattice deform is the primary means by which the aircraft is sculpted. The MQ-9 sensor section in the front of the fuselage, as well as the swept tail surfaces and wing taper were all created using the *lattice deform* tool.

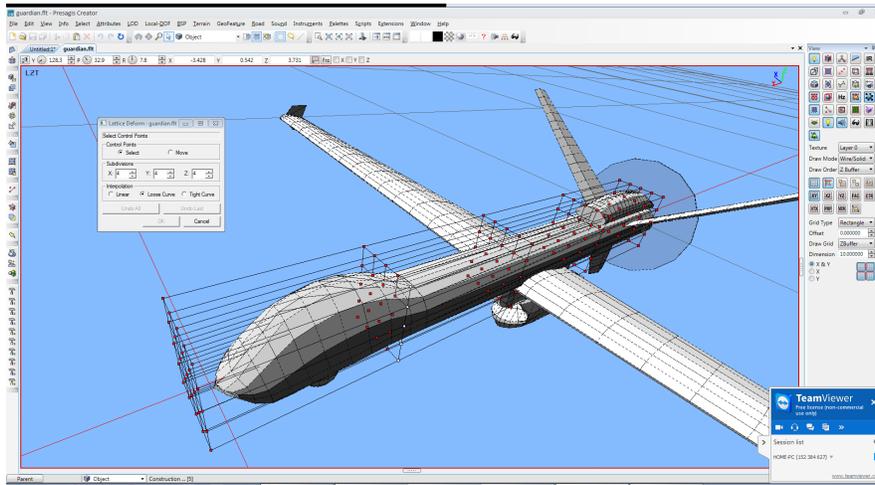


Figure 3.14: MQ-9 fuselage model undergoing lattice deformation

If necessary, the user has the option to scale the model in the X , Y or Z directions as desired. A *scale* option exists in the *view* pane, or *Shift + S* can be used as a shortcut. The view can then be adjusted to focus on the new object

by pressing V .

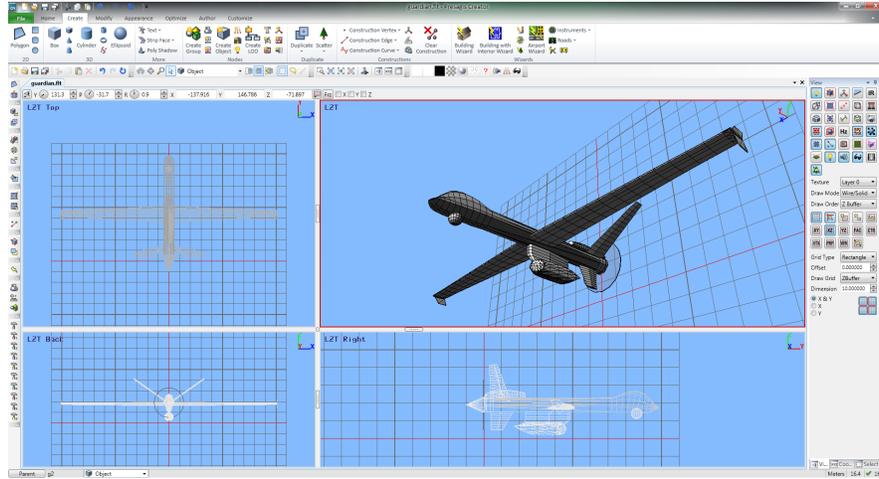


Figure 3.15: CAD 3-view of Guardian in Creator 14

Upon completion of half the aircraft, the model is mirrored in the appropriate plane to create a full aircraft. It can then be textured, coloured and used in the FLSim program as a model in a simulated scenario. Figures 3.15 and 3.16 are screenshots of the final MQ-9 Guardian polygon model.

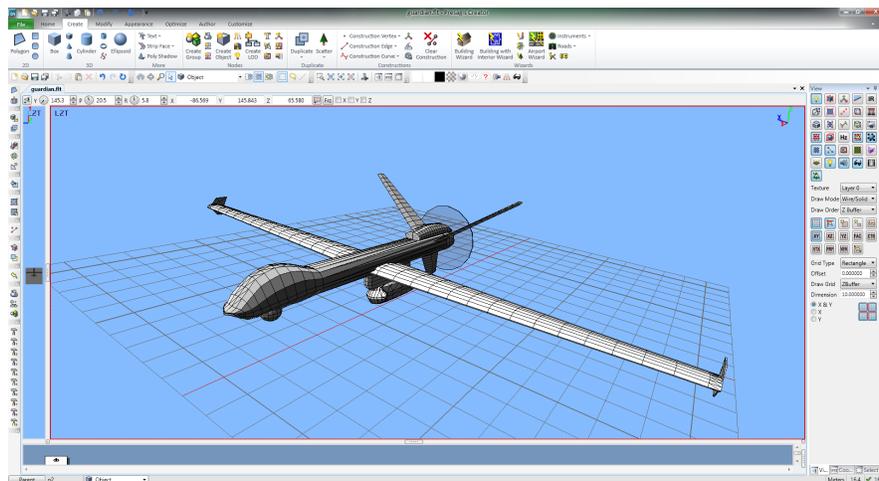
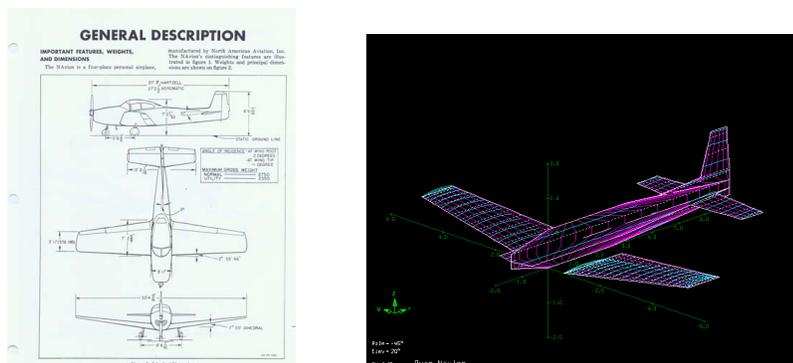


Figure 3.16: MQ-9 Guardian Creator Model with fixed Radar Payload

Athena Vortex Lattice - AVL

The Athena Vortex Lattice (AVL) program written by Mark Drela of MIT is a simple yet powerful executable, which uses vortex lattice aerodynamic calculations on a user-specified geometry and uses ideal flow for calculations, and therefore assumes inviscid conditions as stated by the AVL guide text file. The geometry of the MQ-9 Reaper drone was used for the Guardian, as they share the same airframe.

It was necessary to first validate the AVL stability derivative approximations through direct comparison with an aircraft whose geometries, stability derivatives and mass moments of inertia are well-documented. The Ryan Navion was chosen as the validation aircraft. The Ryan Navion dimensions, weights and mass moments of inertia were provided in [7] and were programmed into AVL to yield stability derivatives. Longitudinal stability derivatives were of key importance since aircraft handling qualities from a symmetrical payload drop was primarily investigated.



(a) Ryan Navion 3-view geometries

(b) Ryan Navion AVL geometry

Figure 3.17: Ryan Navion dimensions

The AVL-computed stability derivatives for the Ryan Navion flying at $Ma=0.2$ and $\alpha=0$ are shown below. Mass moments of inertia can be calculated using [17]. Equations 3.1 to 3.3 below are Raymer's [17] approximations to finding mass moments of inertia of a given aircraft type of length L , wingspan b , mass M as well as the nondimensional radii of gyration about each axis R_x , R_y and R_z . For a single-engine propeller-driven aircraft, these radii of gyration are assumed to be the values shown in Table 3.1.

Radius of gyration	Value
R_x	0.25
R_y	0.38
R_z	0.39

Table 3.1: Radii of gyration values for a single propeller aircraft [17]

$$I_{xx} = \frac{b^2 MR_x^2}{4} \quad (3.1)$$

$$I_{yy} = \frac{L^2 MR_y^2}{4} \quad (3.2)$$

$$I_{zz} = \frac{\left(\frac{b+L}{2}\right)^2 MR_z^2}{4} \quad (3.3)$$

The Navion has a wing span of $b=10.2$ m and a length of $l=8.35$ m and weighs 1336 kg with a single pilot and full fuel [22]. Using these values and Equations 3.1 to 3.3, the Navion mass moments of inertia are estimated as shown in Table 3.2. The calculated mass moments of inertia compare favourably to the true values as published in [22].

Moment of inertia	calculated value (kg·m ²)	measured values (kg·m ²) [22]
I_{xx}	2170	1742
I_{yy}	3360	3762
I_{zz}	4366	4389

Table 3.2: Navion calculated mass moments of inertia

The AVL-calculated stability derivatives compare favourably to the wind tunnel and USAF Datcom data from [7]. The stability derivatives were calculated at low altitude (5000 ft) and at a Mach number of $Ma=0.158$ at that altitude which was approximately 103 knots. These conditions closely match the Navion flight conditions in [19]. Additional data from flight testing from 1972 documented in Suit’s report [22] is provided for an additional comparison where data was provided. Suit’s report provides stability derivatives which were extracted based on the least amount of error deviation from flight test data. The stability derivatives used were assumed to be the sole values across the flight envelope being tested since both the Navion and the MQ-9 were flown at low speeds. Sample calculations for some of these derivatives can be found in the Appendix in Equations A1 to A7.

Derivative	AVL	Flight/wind tunnel Data [19]	Flight Statistics [22]	Hand calculations
$C_{L\alpha}$	4.0545	4.52/6.04	6.04	4.22
$C_{m\alpha}$	-0.8615	-0.95/-	-0.77	-0.717
C_{mq}	-13.5578	-/-	-18.3*	-12.360

(a) * Only $C_{mq} + C_{m\dot{\alpha}}$ could be found for flight data

Table 3.3: Navion AVL-computed longitudinal stability derivatives comparison

Derivative	AVL	Flight/wind tunnel Data [19]	Flight Statistics [22]
$C_{Y\beta}$	-0.4203	-0.77/-0.61	-0.6
$C_{l\beta}$	-0.1116	-0.067/-0.097	0.07
C_{lp}	-0.4482	-/-0.46	-0.49
C_{lr}	0.1257	0.270/-	0.11
$C_{n\beta}$	0.0585	0.086/0.109	0.073
C_{np}	-0.0148	-0.038/-	-0.04
C_{nr}	-0.1389	-0.163/-	-0.09

Table 3.4: Navion AVL-computed lateral stability derivatives comparison

The user enters various parameters specifying the aircraft geometry as well as its weights, mass moments of inertia and flight conditions. AVL will then quickly run cases defined by the user and give stability derivative outputs as well as forces and moments on the aircraft in the given flight conditions. Note that some of the stability derivatives do not closely match. Data between wind tunnel tests, flight tests and the statistical analysis from Suit [22] are all different, and the AVL approximations also vary. It is assumed that a range of values for these derivatives is sufficient since the values do not correlate closely. The correct fuselage dimensions for the Navion were unavailable and AVL is particularly limited in its ability to easily define fuselage dimensions. The combination of these limitations are what likely caused the low values for $C_{Y\beta}$ (which was calculated as shown in the Appendix as -0.525) and $C_{n\beta}$. It must also be noted that studies published within the same relative time as shown by Tables 3.3 and 3.4 frame show inconsistent values for stability derivatives.

Since most of the stability derivatives for the AVL Navion closely matched the wind tunnel tests from [19] and the statistically-chosen flight test data from [22], it was decided to use the same method to find the stability derivatives of the MQ-9 Reaper.

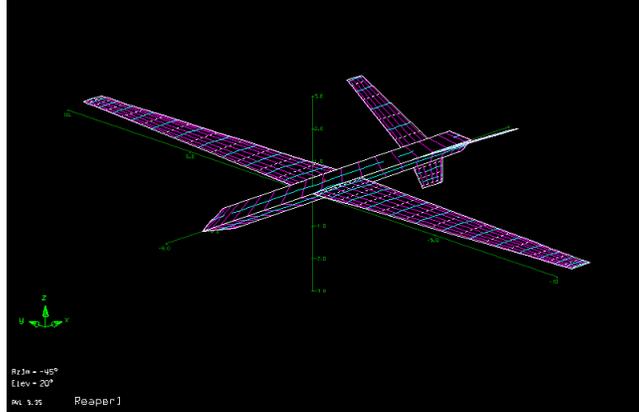


Figure 3.18: MQ-9 Reaper geometry in AVL

The MQ-9 Reaper airfoil is proprietary information. An approximation based primarily on visual inspection of the Guardian was the only way to find a suitable airfoil for the simulation. Figure 2.1b shows a Guardian in flight, with its wing tip airfoil section clearly visible between the wing fences. An airfoil, similar in appearance, was chosen as the airfoil for the whole wing as an approximation. The Eppler 374 airfoil was chosen as a place holder for the proprietary MQ-9 airfoil, and was chosen based on its visual similarity to the wingtip airfoil visible in Figure 2.1b.

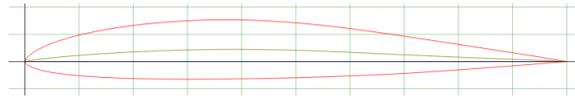


Figure 3.19: Eppler 374 airfoil

Table 3.5 shows the AVL-calculated stability derivatives of the MQ-9 at 120 knots, level flight at sea level. These flight conditions were chosen based on the conditions a low-medium altitude SAR aircraft would likely experience. The derivatives closely matched wind tunnel data for the Navion and were therefore chosen for the MQ-9 due to their accuracy. Table 3.6 shows the lift, drag and moment coefficients for the Reaper at the given flight conditions. Because the aircraft is symmetrical along its fuselage axis, roll and yaw are assumed and calculated by AVL to be 0 in steady level flight.

The Guardian mass mass moments of inertia (excluding the payload) were calculated using Equations 3.1 to 3.3 solely from the dimensions and maximum take-off weight of the Reaper. The Guardian is a low Reynolds Number propeller aircraft and can therefore have its radii of gyration approximated using Raymer's values as was done for the Navion. The mass mass moments of inertia were found to be $I_{xx}=23480 \text{ kg}\cdot\text{m}^2$, $I_{yy}=9770 \text{ kg}\cdot\text{m}^2$ and $I_{zz}=29297 \text{ kg}\cdot\text{m}^2$.

Derivative	AVL
$C_{L\alpha}$	6.151
$C_{Y\beta}$	-0.244
$C_{m\alpha}$	-1.952
C_{mq}	-20.362
$C_{l\beta}$	-0.057
C_{lp}	-0.595
C_{lr}	0.163
$C_{n\beta}$	0.034
C_{np}	-0.054
C_{nr}	-0.027

Table 3.5: MQ-9 AVL-computed longitudinal and lateral stability derivatives

Force or moment coefficient	AVL
C_L	0.1510
C_D	0.0005
C_m	-0.0255

Table 3.6: MQ-9 AVL-computed forces and moments at $\alpha=0$ deg.

Flight Sim 14 Model

Presagis Flight Sim 14 (FLSim 14) is a program used to create aircraft flight models for simulations. The aircraft must be defined in its geometry, weights, mass moments of inertia, control logic for its control surfaces as well as its stability and force derivatives. All of these pertinent parameters can be constructed and adjusted through the FLSim *Modeler* mode. Individual curves for each stability derivative, or each change in center of mass location for each component can be plotted versus Mach number, angle of attack, time in the air or any other variable and function the user desires.

The weights and locations of important components were defined first. FLSim uses the Fuselage Station (FS), Waterline (WL) and Buttline (BL) references in order to locate components of aircraft in 3D space as shown in Figure 3.20. Figure 3.23 illustrates the FLSim weight and balance screen. This is where the user enters mass moments of inertia, empty weight and the aircraft overall centre of gravity. Figure 3.20 illustrates the convention. Since default FLSim aircraft were built with their noses intercepting the WL axis, the Navion and MQ-9 were built in the same fashion.

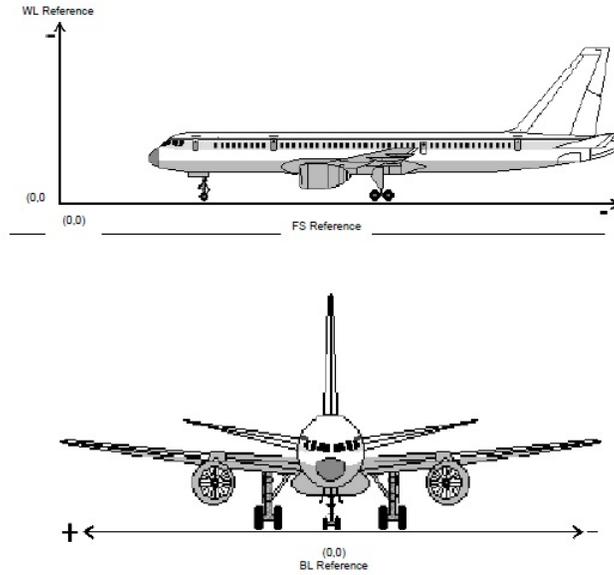


Figure 3.20: Flight Sim 14 aircraft reference lines [8]

FLSim 14 can be used in either *Modeler* mode or *Run-time* mode. FLSim *Modeler* mode is used to build the flight model and the aircraft properties into FLSim while *Run-time* mode is used to observe flight, monitor parameters such as fuel state, phugoid motion, weight changes, performance metrics and more. Within each mode, there are tabs displayed on the left menu. *Modeler* mode contains tabs for flight models, engines and additional loads the aircraft can carry. *Run-time* mode contains the options to setup the aircraft flight scenario and conditions, plots that monitor and display parameters the user chooses, tests which cause the aircraft to encounter automated conditions such as engine failure, and communications which the user uses to define the means by which the aircraft is controlled in the simulation.

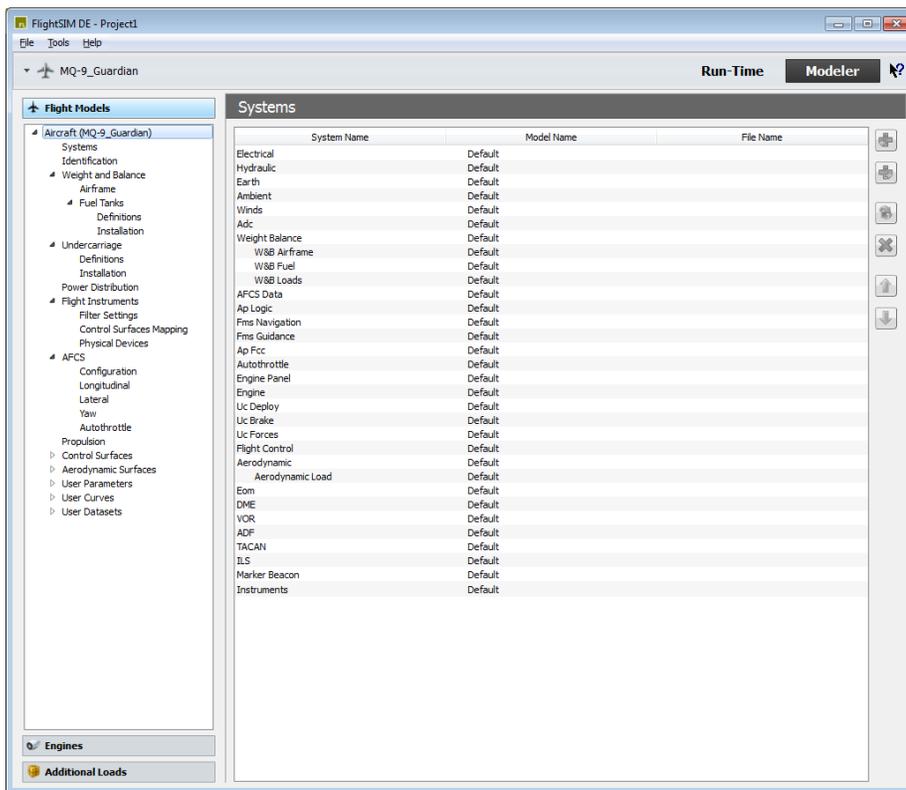


Figure 3.21: Flight Sim 14 *Modeler* mode

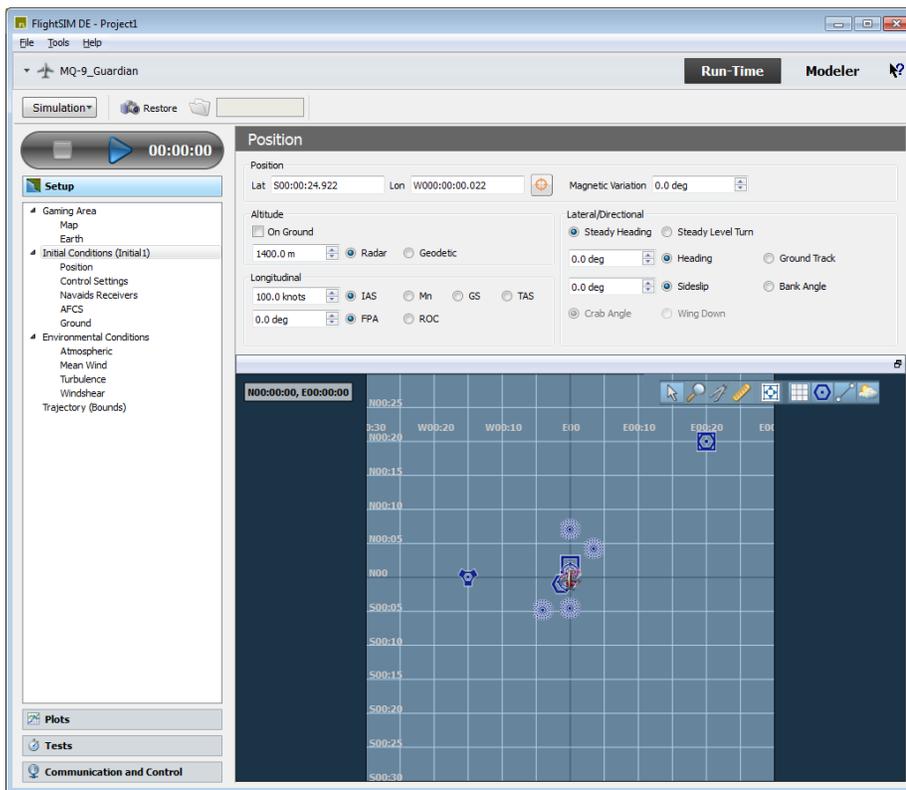


Figure 3.22: Flight Sim 14 *Run-time* mode

Each of the mode tabs contain their own menus and sub-menus which allow the definition of detailed properties. The *Flight models* tab contains all the necessary information for the flight model of the aircraft. Engines, weights, mass moments of inertia, centers of gravity and flight force and stability derivative curves are all defined by the user. The first menu used for defining the MQ-9 was the *weights and balance* menu. The MQ-9 masses were taken from [23] and mass moments of inertia were calculated based on its dimensions and Raymer’s approximations shown in Equations 3.1 to 3.3. The centre of mass was assumed to be approximately 25 percent of mean aerodynamic chord. This was adjusted by inputting numbers under the *Center of Mass (CoM)* section as shown in the Figure 3.23.

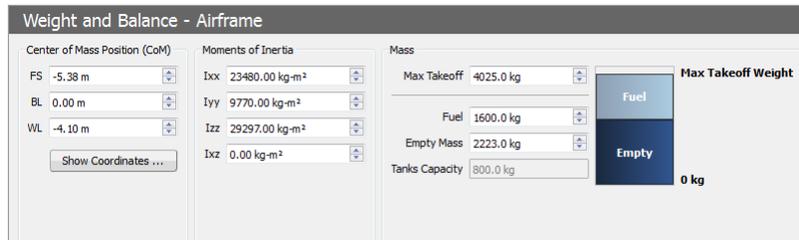


Figure 3.23: The weight and balance screen with values for the MQ-9

Other components, like the aircraft landing gear and engine, can have their locations specified as well. In addition, any change in moment of inertia due to the system (i.e. landing gear retraction) can be modelled. The landing gear friction braking coefficient can be defined as well. Figures 3.24 and 3.25 display the *landing gear* modelling section.

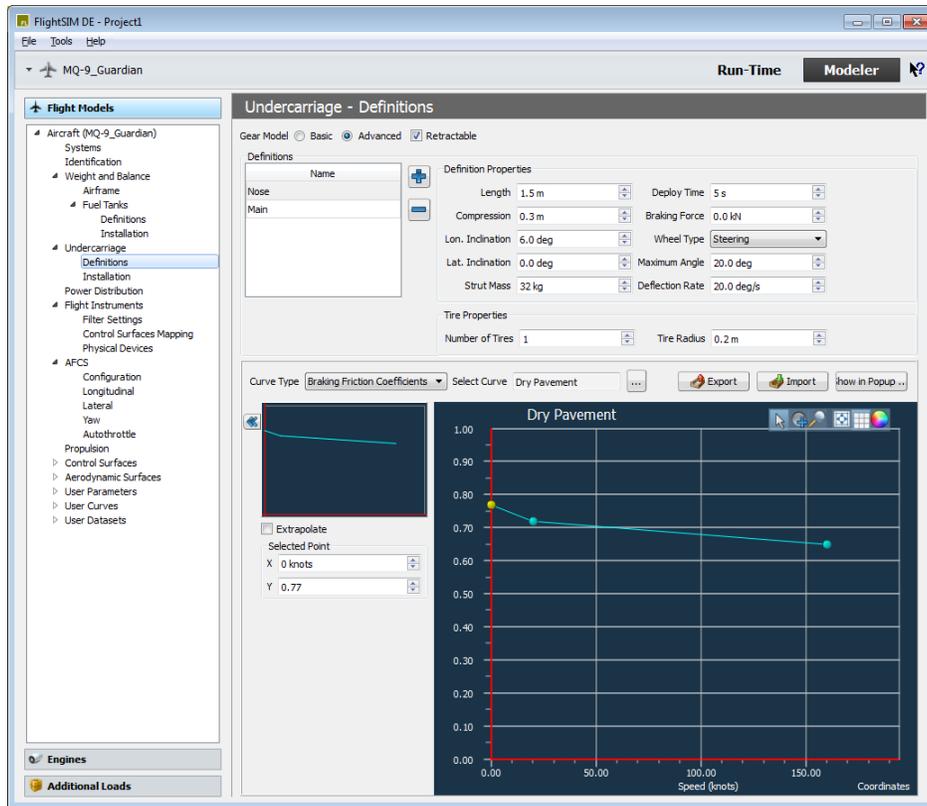


Figure 3.24: Landing gear definition and braking friction curve

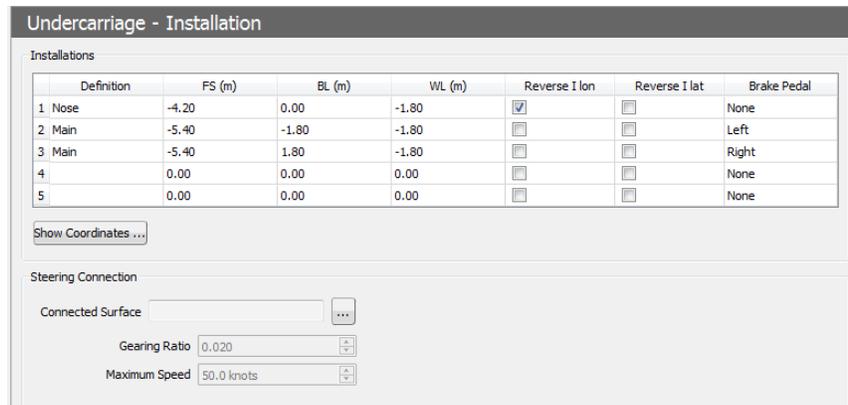


Figure 3.25: Gear installation specification

A graphical representation of the locations of each component center of mass

is accessible from any part installation menu and appears as shown in Figure 3.26.

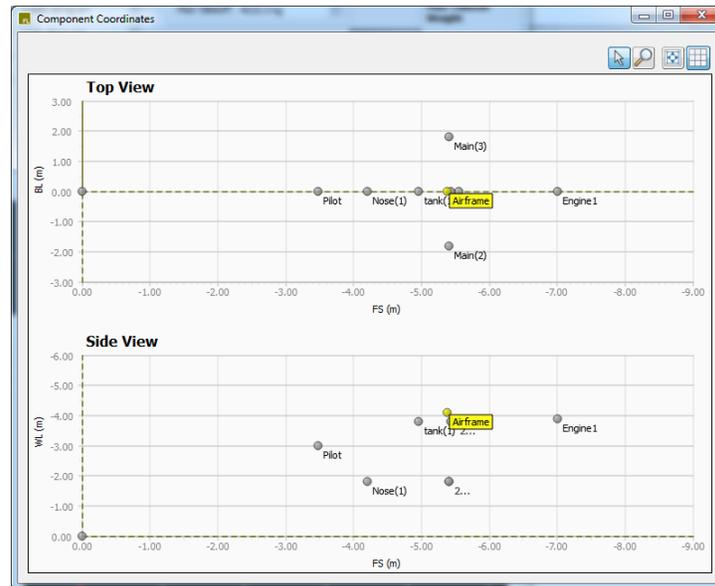


Figure 3.26: Weight and balance of components on FLSim Guardian

The *propulsion* menu in the *Flight models* tab allows the user to select pre-defined engines. The MQ-9 uses a default PT-6 turboprop engine, modelled in FLSim as a default engine as a stand-in to the newer Honeywell TPE331-10GD turboprop due to the similar size and power output of the two engines. The user may define the location and number of engines in this menu as shown in Figure 3.27, but engine modelling can only be done in the *Engines* tab. The user may define engine fuel consumption, blade angles as a function of forward airspeed, power output and other parameters as curves in that tab if necessary. Figure 3.28 displays a user-defined curve with N1 turbine speed as a function of engine pressure ratio; just one of the engine curves that can be defined in FLSim.

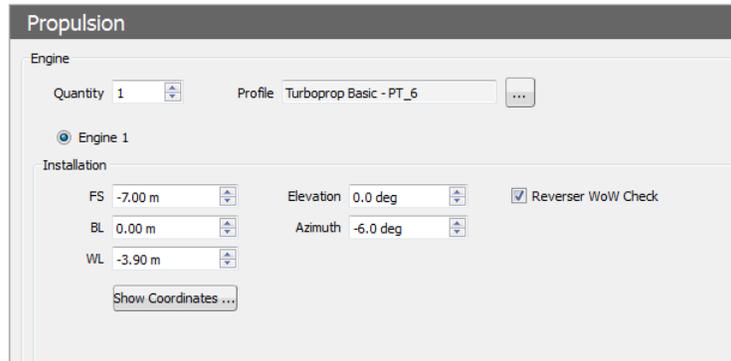


Figure 3.27: The *propulsion* menu in *Modeler* mode

Since engine performance was not considered as a factor affecting longitudinal performance, the default, predefined PT-6 engine was considered sufficient for the MQ-9. The same engine was used on the Navion validation model, but full throttle was not used since the Navion's stock Continental E185 piston engine produced far less power than the PT-6 turboprop.

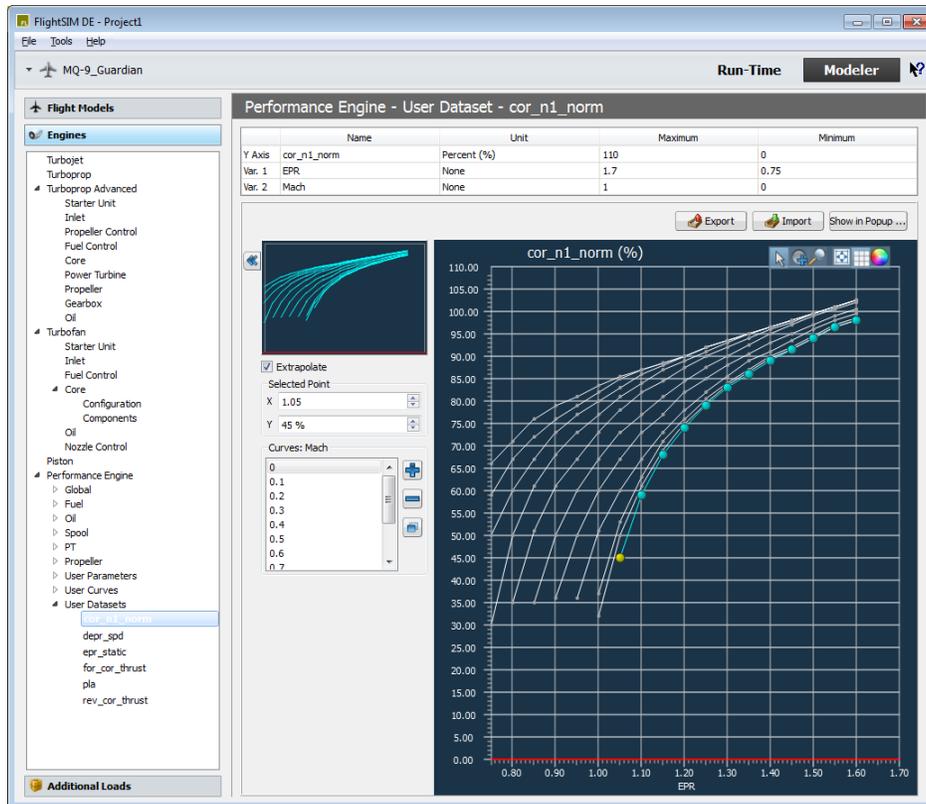


Figure 3.28: The *Engines* modelling tab in *Modeller* mode

The *aerodynamic surfaces* menu allows the user to define the aerodynamic centre of the aircraft wing as well as the wing itself via wing area, chord and span in metres. The location of the aerodynamic centre is defined in the same way the centers of mass are defined for the landing gear and engines.

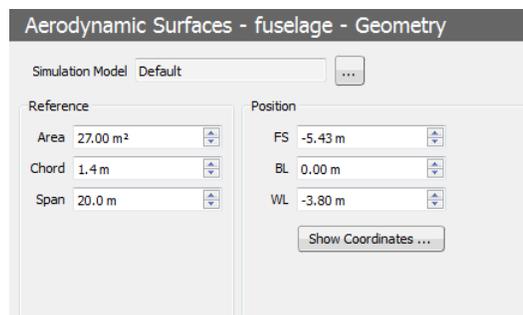


Figure 3.29: Locating and defining the wing and its aerodynamic center

The FLSim default UAV models used the control logic diagrams shown in Figures 3.30 to 3.33. The Navion and MQ-9 Guardian UAV flight models were built with the FLSim default control logic. With the derivatives available, the MQ-9 graphic model was built in Creator and united with the FLSim model in the Presagis Stage simulation environment.

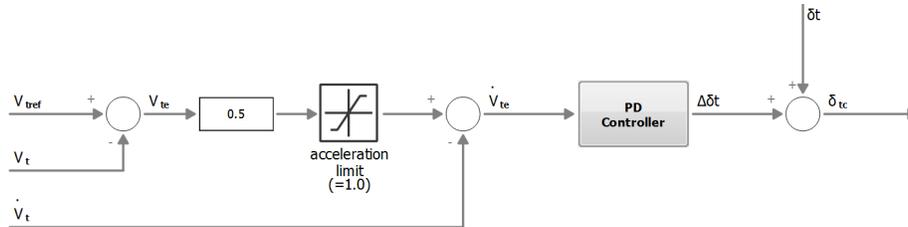


Figure 3.30: autothrottle control logic

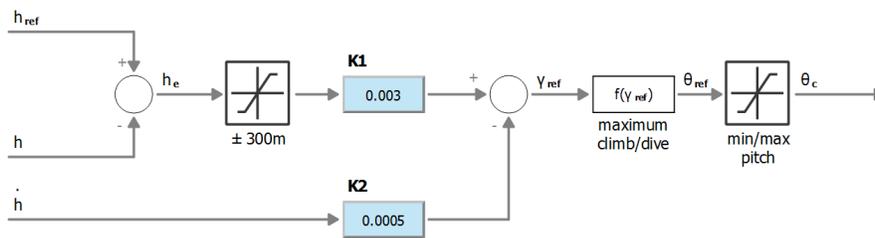


Figure 3.31: longitudinal control surface logic

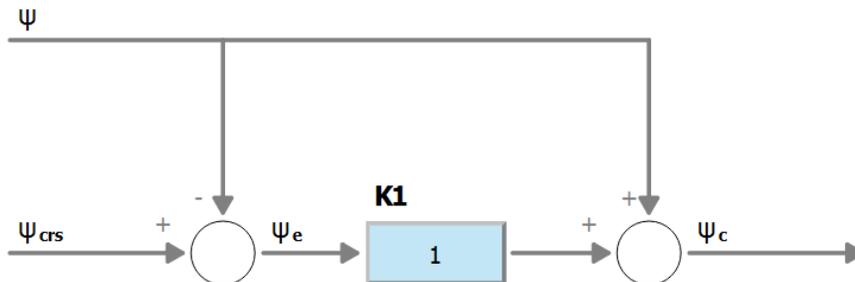


Figure 3.32: yaw control logic

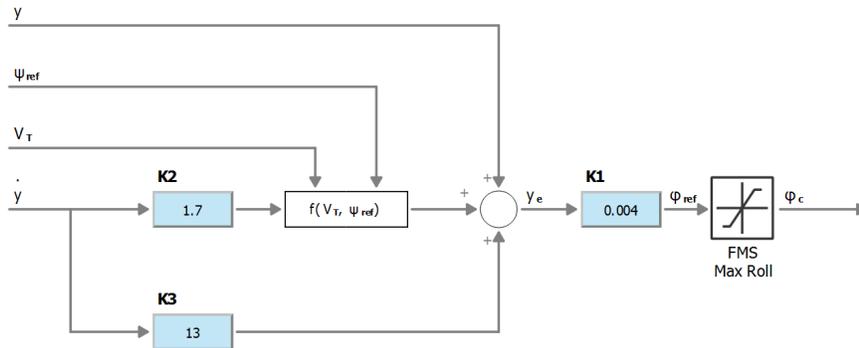


Figure 3.33: lateral control surface logic

The most important component of an aircraft flight model is its stability derivative and its lift and drag curves. The stability derivatives and forces at defined flight conditions were extracted from AVL and applied as *user defined curves* in *Modeler* mode. The derivatives and forces from Tables 3.5 and 3.6 were used to construct the appropriate curves that would define the the MQ-9 stability and flight characteristics. Force coefficients such as C_L , C_D and C_m as well as their derivatives ($C_{L\alpha}$, $C_{D\alpha}$ and $C_{m\alpha}$) found in AVL were plotted directly into the FLSim user curve tree in *Modeler* mode. These curves were then used in the definition of control Equations in the Control Surfaces tree. Figure 3.34 below shows the different C_L curves for the RAD 2014 RC aircraft with different elevator deflections.

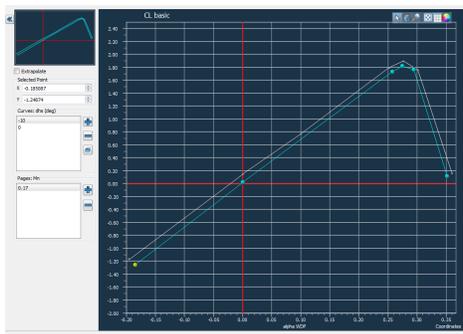


Figure 3.34: Total C_L vs α curve for the MQ-9 Guardian

The MQ-9 Guardian was modelled with a 200 kg, cube-shaped payload mounted on the centerline, just aft of the wing trailing edge. This was done using the *Additional loads* tab. The mass moments of inertia of the payload were calculated using 3.1 to 3.3. The user may first create individual payload properties which include its mass, mass moments of inertia and any lift or drag

it may produce. A profile can then be made which defines the location of the payload on the aircraft. Figures 3.35 and 3.36 illustrate how the load is defined and then configured on the aircraft.

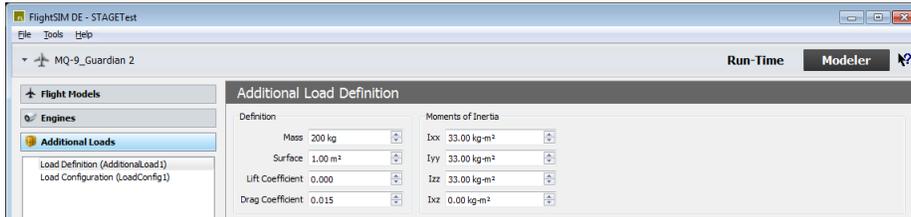


Figure 3.35: Defining payload properties

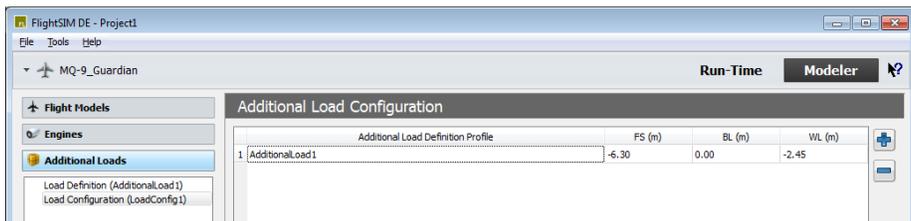


Figure 3.36: Selecting and saving payload configuration

It is imperative that the user then returns to the *Initial conditions* tab and selects the *control settings* menu. Here, the load configuration that was defined can be selected under *Additional load configurations*. This allows the payload to affect the aircraft as soon as the simulation begins.

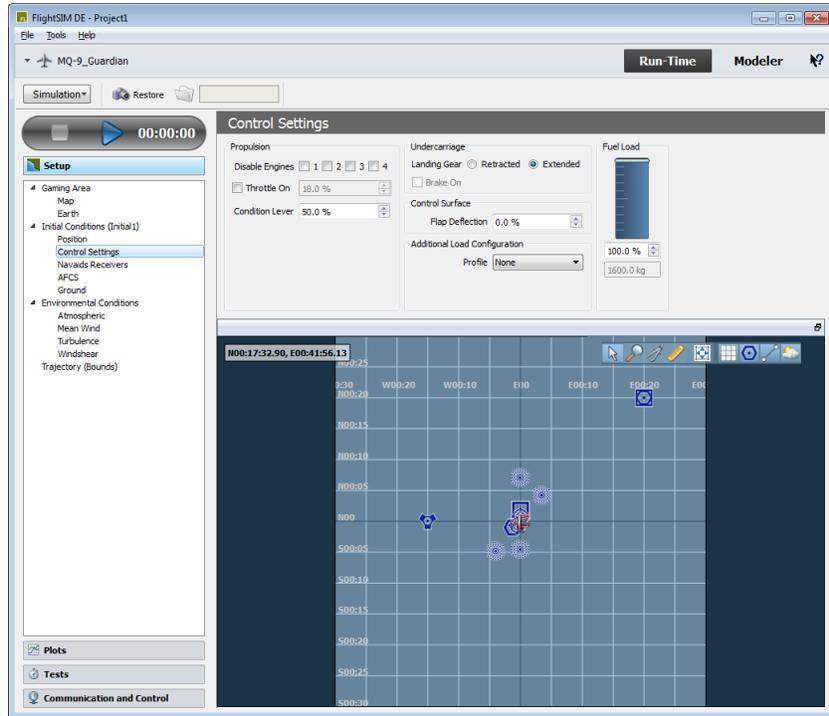


Figure 3.37: Applying the payload configuration to the aircraft

With the aircraft defined to the user's liking, the aircraft may be simulated in flight. As previously mentioned, FLSim 14 in *Run-time* mode allows the user to modify the aircraft's initial altitude and velocity. The *communications and control* tab can be used to define the whether the aircraft is controlled by FLSim default controls involving the mouse and keyboard, or a joystick. If the user prefers to use the latter, the user can select the PCJOYSTICK option for the desired functions. The user may define variables to track as the simulation runs by modifying and adding variables to track in the *plots* tab.

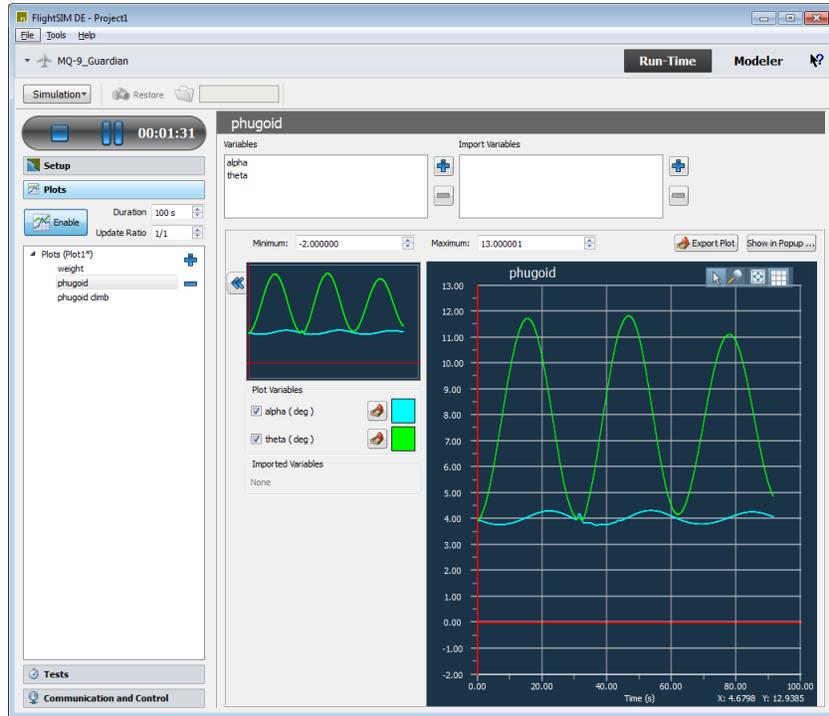


Figure 3.38: Tracking aircraft climb rate and α as a function of time

Finally, the play button located on the top left of the screen can be pressed and FLSim will proceed to run the simulation. Figure 3.38 shows an example of an aircraft angle of attack and climb rate being tracked over 100 seconds of flight. Figure 3.38 above shows the screen where these initial conditions are defined, and where the aircraft flight parameters are tracked. In this case, the human pilot is pulling and pushing the stick to oscillate the aircraft.

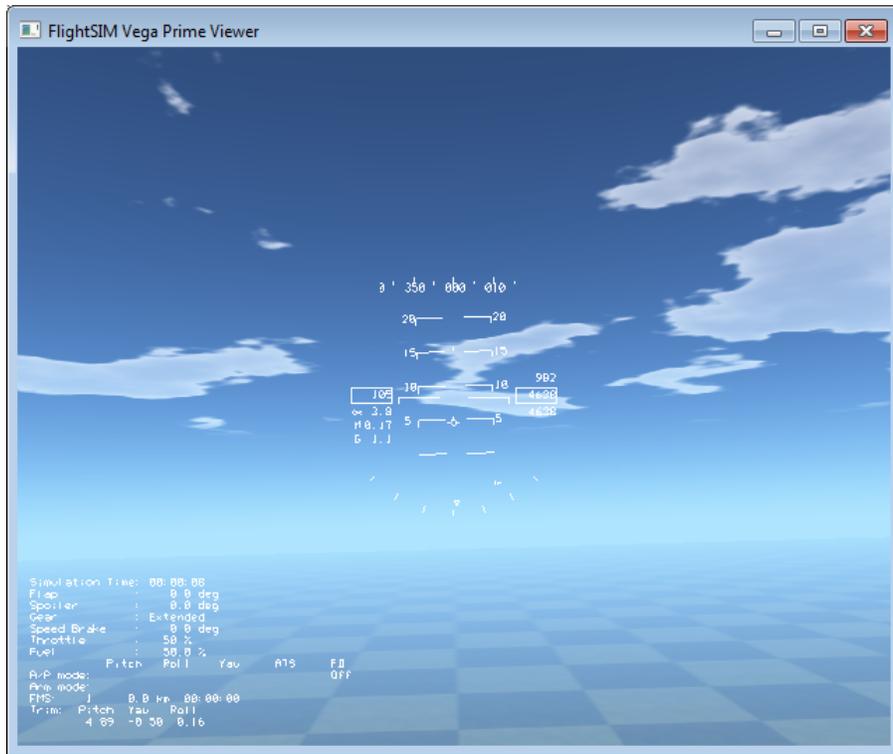


Figure 3.39: FLSim simulation cockpit view

Figure 3.39 displays the virtual cockpit view of the simulated MQ-9 Guardian. A standard HUD shows the flight path vector mark as a small circle with three small lines as well as the angle of climb indicators as horizontal lines. Speed in knots is shown on the box to the left with an angle of attack, Mach and normal loading indicators directly beneath the speed indicator. Climb rate in feet per minute is shown on the right. Other relevant information is shown in the bottom left corner of the screen.

Stage 14 Integration

The Stage 14 simulation environment was used primarily to observe the integration of the flight and polygon models of the MQ-9 Guardian. The user has the option to create a multitude of aircraft models which may interact with each other, though this was not done as it was beyond the scope of this project. The following section demonstrates how the MQ-9 was integrated into the Stage environment.

The first step is to create a new project by pressing *File > New* in the upper left-hand corner. The user will then be shown multiple windows. The Earth will be shown as the current simulation environment.

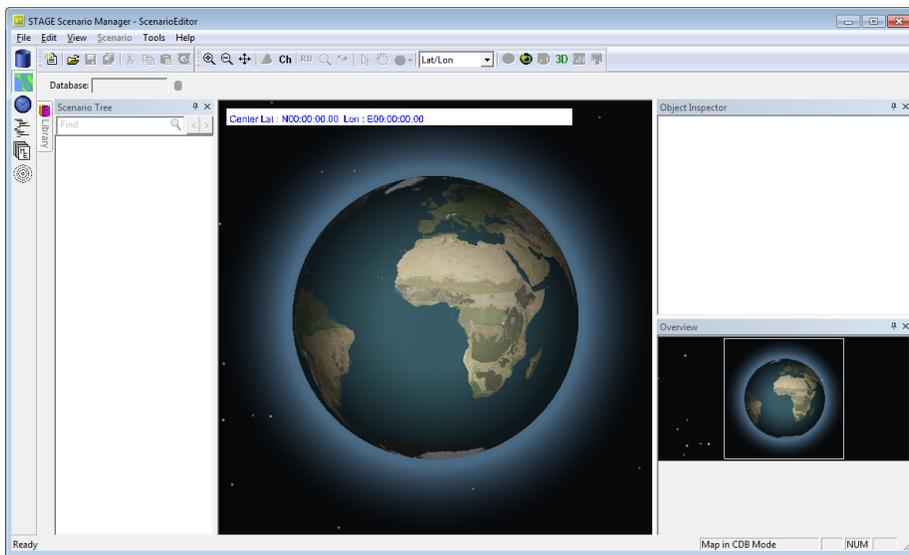


Figure 3.40: New project (database) in Stage

Left clicking the database editor button (as shown highlighted in red in Figure 3.41) allows the user to define properties of units that will be simulated in the environment. The units dimensions and mass can be edited in the primary window and its name can be changed. The following figures use a platform renamed to *Guardian*. The polygon model which represents the unit can also be selected here under *3D Object*. The Guardian Creator polygon model was selected as shown in Figure 3.41.

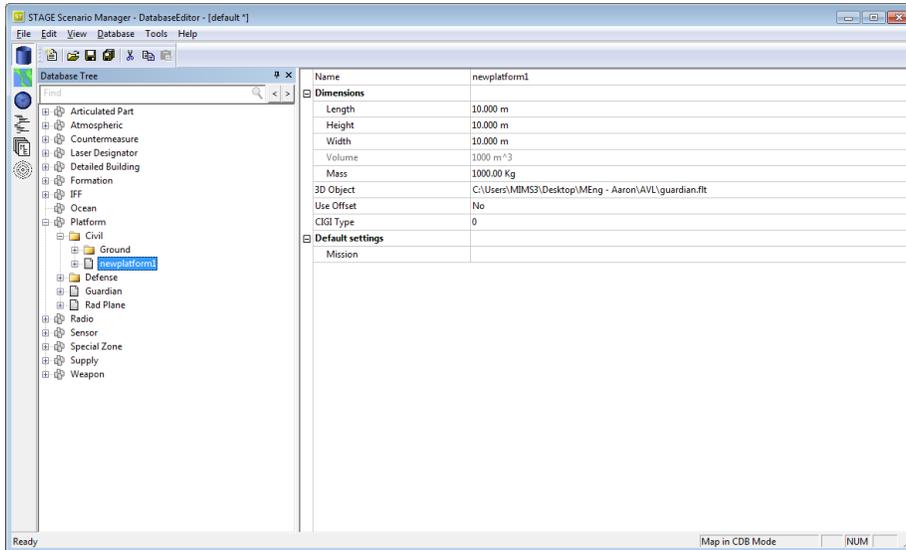


Figure 3.41: Defining a unit platform

The user can then click the scenario editor button on the left pane of the screen, as highlighted in red in the following Figure. Here, the user can enter latitude and longitude coordinates to choose a more specific map location for the simulation. A scale is also supplied when *Map parameters* is selected to show a closer or further view of the map being shown. Toronto is chosen as shown below with a small portion of the city rendered as a texture.

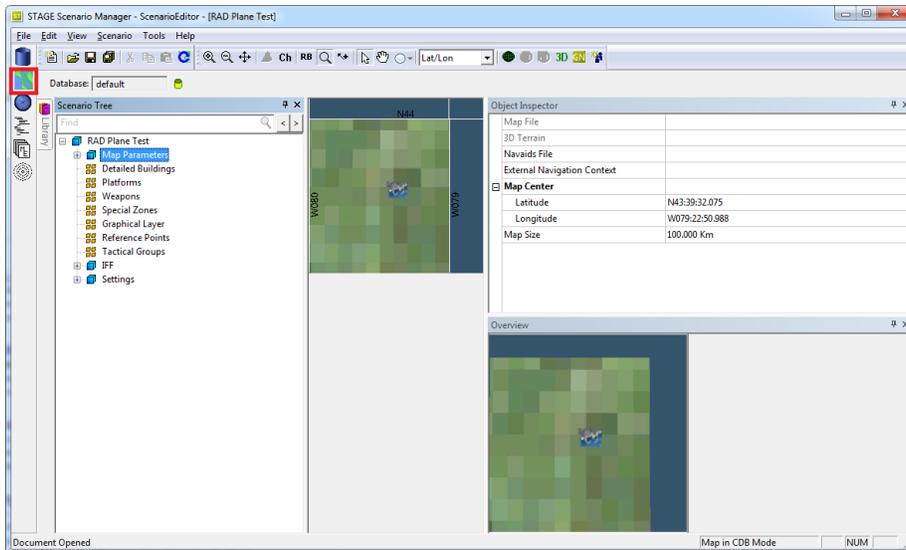


Figure 3.42: Locating Toronto in Stage

Figure 3.43 illustrates the process of adding a plane in a project titled *RAD Plane Test*. A series of expandable menus in the *Scenario Tree* pane allows the user to select and define an aircraft by right clicking *Platforms* and then clicking *Add*. Here, *item1* is the newly added item shown under *Platforms*. The user has the option to create many units defined by many different platforms available in Stage. Many platforms are built into Stage, but more can be defined as shown in Figure 3.41. Ground vehicles, soldiers and other dynamic systems can be made if necessary. These units can be given commands in order to interact with one another. This project does not cover such commands since the Guardian is the sole unit being simulated.

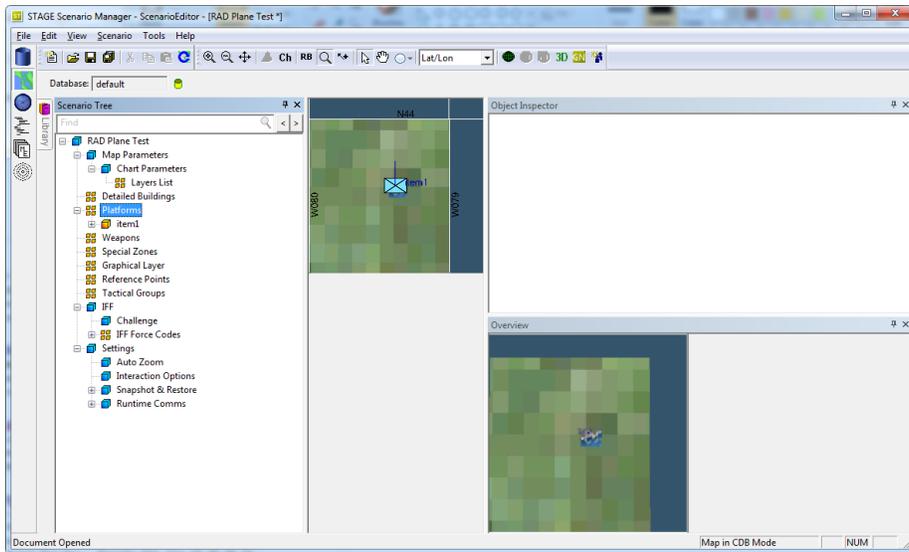


Figure 3.43: Defining aircraft

The user can highlight the item by left clicking its name. With *item1* highlighted in the *Scenario Tree*, the user must define the user can edit the item's properties in the *Object Inspector* window. The item name can be changed (in this case to *Guardian*) and its platform profile and dynamic model can be chosen. The platform profile is defined by double clicking the item to the right of *platform profile* and a new window opens showing available platforms. The platform acts as the unit type in Stage and has no information on what real-time model simulates its actions during simulation.

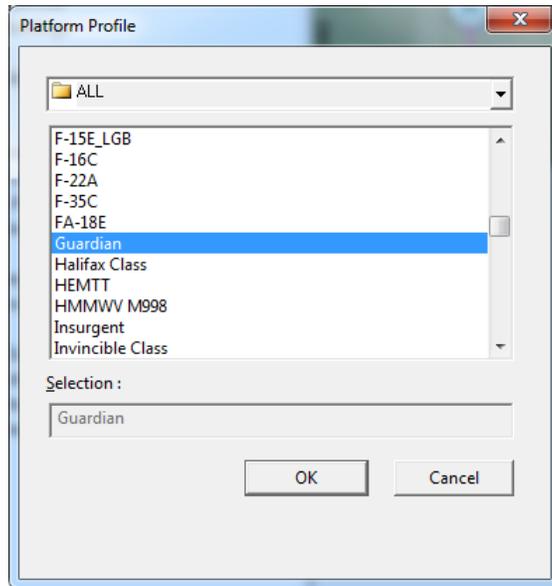


Figure 3.44: Guardian aircraft platform being selected

The unit is now defined as a Guardian aircraft. In this case, the dynamic model source is FlightSim (FLSim). The user must define the specific FLSim model being used by highlighting the new *Flightsim Configuration* tree menu. Referring to Figure 3.46 the user selects the *MQ-9.Guardian 2* flight model for the *Aircraft model* section in the *Object Inspector* window. *PilotInput1* is selected for under *Pilot Input* and *Cigi1* selected under *CIGI configuration* if these configurations exist. If they are not defined, the boxes are left empty and the aircraft cannot be controlled by the user even though it can still be simulated along a trajectory. The definition of these settings is not covered in this project.

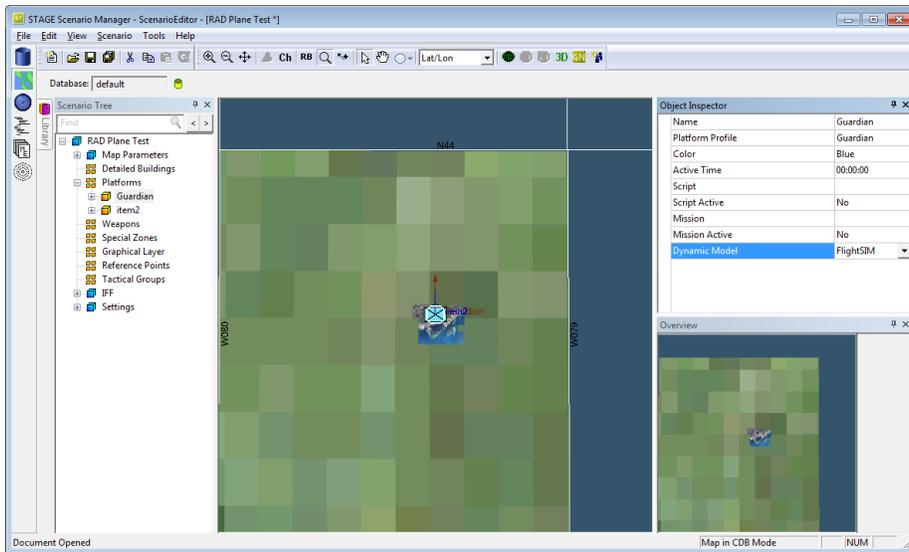


Figure 3.45: Defining the aircraft model

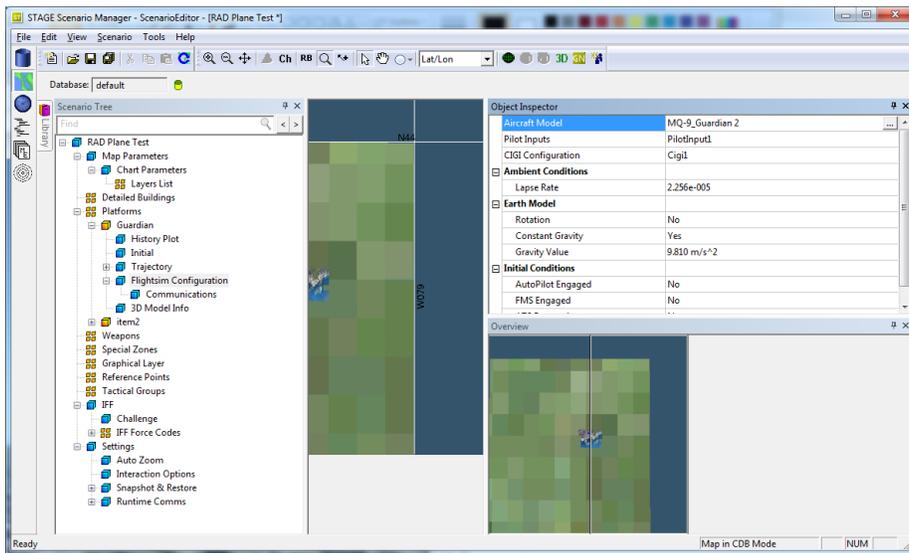


Figure 3.46: Defining the aircraft configuration

The initial conditions of the aircraft can now be set. *Initial* under the expanded Guardian tree can be highlighted and changed in the *Object Inspector* window.

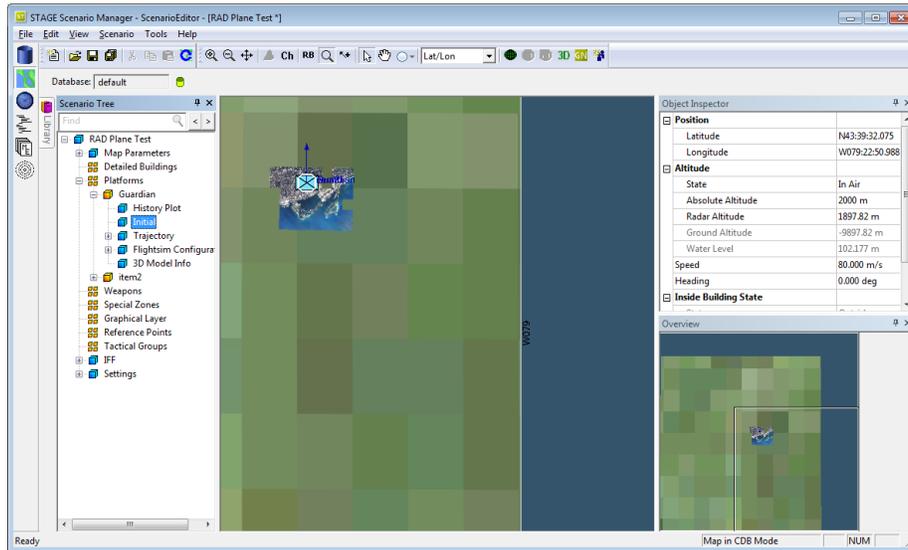


Figure 3.47: Defining the MQ-9 initial conditions

Finally the user can begin simulation by clicking the runtime environment button as highlighted in red in the following image. Highlighted in orange are the simulation controls. The play button begins the simulation and the button with the 3D icon opens a new *3D Viewer Control* window which can be used to set the camera on a specific entity (Figure 3.49) once the simulation has started.

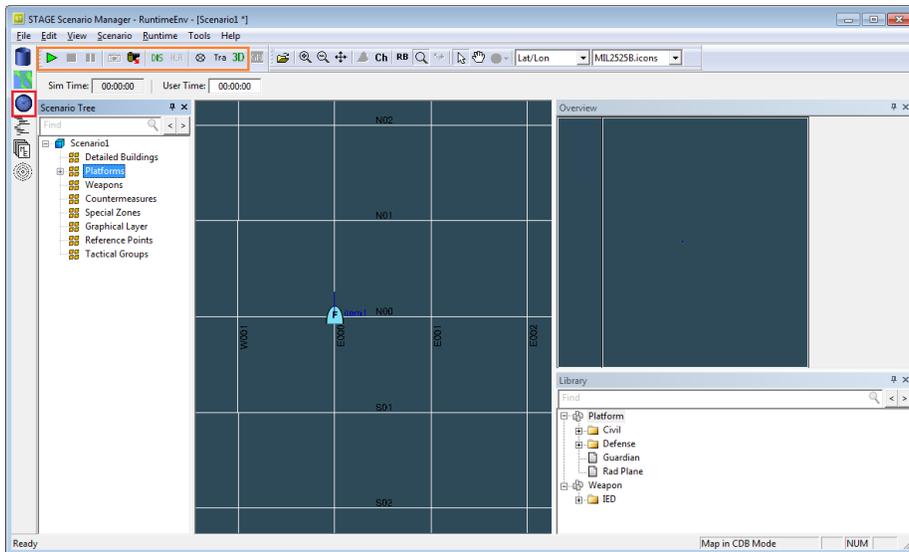


Figure 3.48: Stage in the scenario editor mode

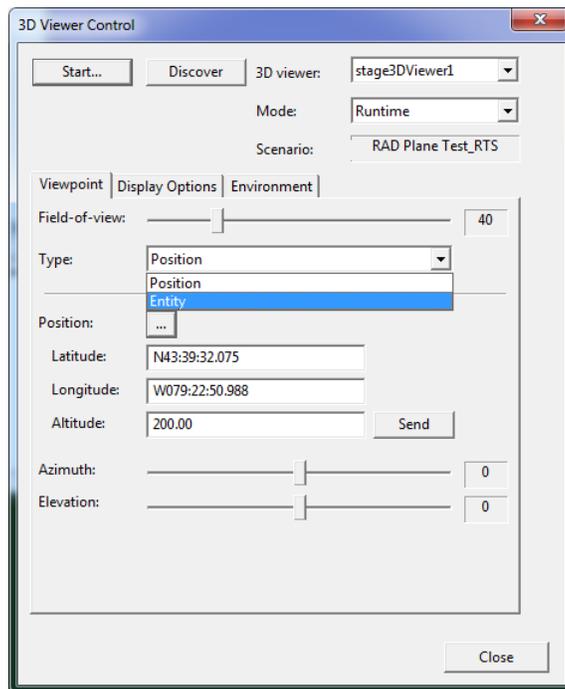


Figure 3.49: Selecting the Guardian unit for viewing

The Guardian can then be observed in flight, under the control of the user if the pilot inputs were set accordingly. The various view controls are located in the *3D Viewer Control* window.



Figure 3.50: Untextured Guardian in flight in Stage

4 Humanitarian Configuration Changes

The primary difference between the MQ-9 Reaper and Guardian is the latter's role as a maritime patrol vehicle using a fixed avionics payload under the fuselage. A possible humanitarian configuration change involves the Guardian's ability to carry this payload as a SAR aid package that can be jettisoned. It may possibly carry a life-raft, food and water supplies, a homing beacon and a parachute to allow safe delivery. In the simulation, it was modelled to be a two metre-long cylinder with a one metre radius and a mass of 200 kg, yielding mass moment of inertia values $I_{xx}=25.0 \text{ kg}\cdot\text{m}^2$, $I_{yy}=266.0 \text{ kg}\cdot\text{m}^2$ and $I_{zz}=25.0 \text{ kg}\cdot\text{m}^2$.

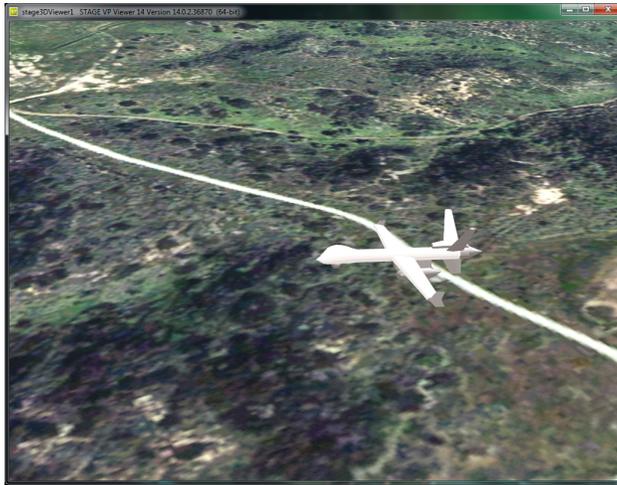


Figure 4.1: Flight Sim 14/Creator Guardian in Stage environment

The RCAF CP-140 maritime and reconnaissance aircraft and is capable of dropping the arctic Survival Kit Air Droppable (SKAD) [5] as shown in Figure 4.2. Multiple modified Guardians could be deployed to complement or replace this kind of aircraft without the need for a crew.



Figure 4.2: A Canadian CP-140 dropping a SKAD package

5 Method Validation and Evaluation of Guardian Flight Model

Navion Flight Model Validation

The MQ-9 Guardian dynamic modes were first validated by observing the Ryan Navion model in flight, and concluding that the Navion flight model and its construction accurately portrayed the true flight stability of the Navion. The Ryan Navion flight model simulated phugoid motion, Dutch roll and short period modes were compared to data provided in [2] and [6]. The plots of the simulated and flight-tested Navion aircraft were compared and values for damping ratio were estimated from the peak and trough values in these plots. Figure 5.1 illustrates the similarities between the phugoid and short period motion damping ratios, frequencies and amplitudes shown in [2] of the Ryan Navion at 105 mph at 5000 feet. The Dutch roll of the simulated Navion was compared to computed and flight test results of the Navion as documented by Ebbert [6]. The damping ratio of any decaying sinusoidal variable can be approximated using the following equation [3]:

$$\zeta = \frac{\ln \delta}{2\pi} \quad (5.1)$$

where δ is the logarithmic decrement [3] given by:

$$\delta = \frac{1}{n} \ln \frac{y_0}{y} \quad (5.2)$$

where each y_0 and y correspond to the amplitude at a given peak and any subsequent peak respectively and n represents the amount of period cycles in between the peaks with amplitude y_0 and y .

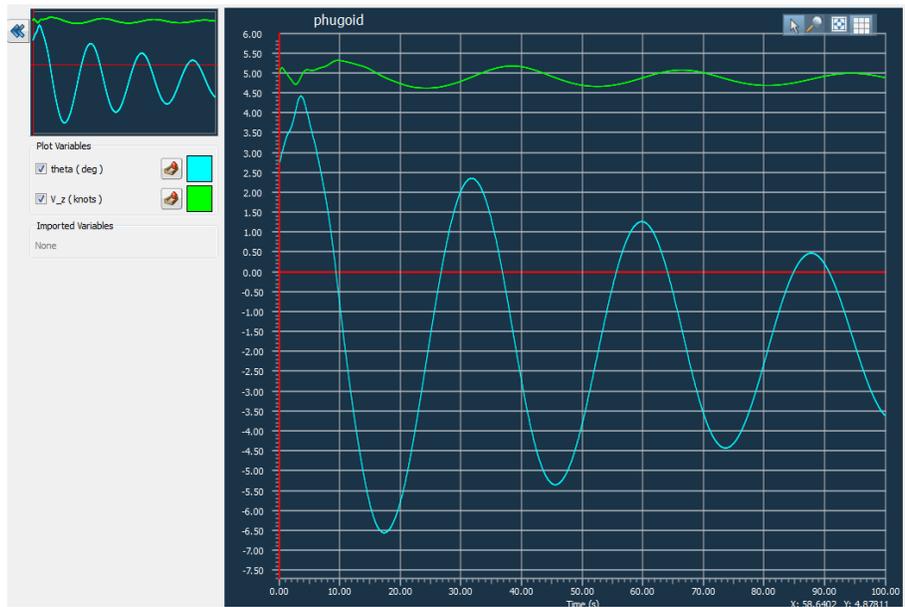
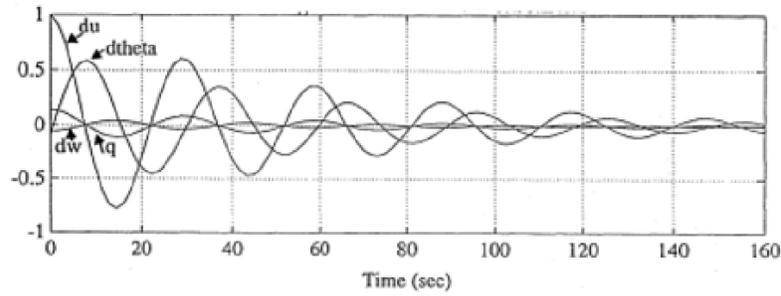


Figure 5.1: FLSim Navion Phugoid mode (bottom) comparison with real Navion [2] (top)

Using equation 5.1, the damping ratio between the second and third troughs of the Navion’s phugoid motion was $\zeta_{phug}=0.0748$ for the actual Navion and $\zeta_{phug}=0.0661$ for the FLSim Navion in Figure 5.1. Additionally, the period of each cycle was about 33 seconds for the real Navion and 34 seconds for the FLSim Navion. The blue and green curves display the pitch angle, θ in degrees and the climb rate in knots respectively. A summary of the dynamic mode damping ratios and errors between the real and FLSim Navion is shown after Figures 5.3 in Table 5.1.

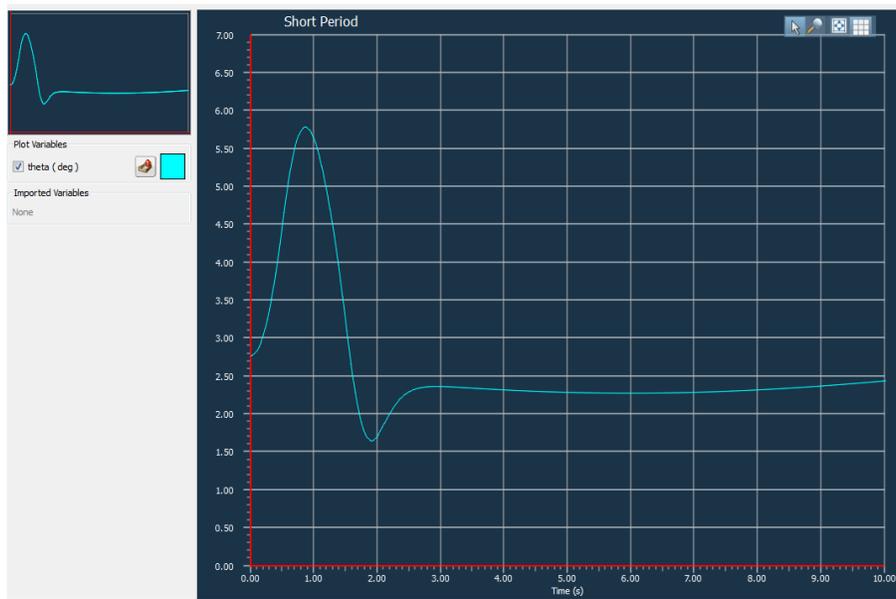
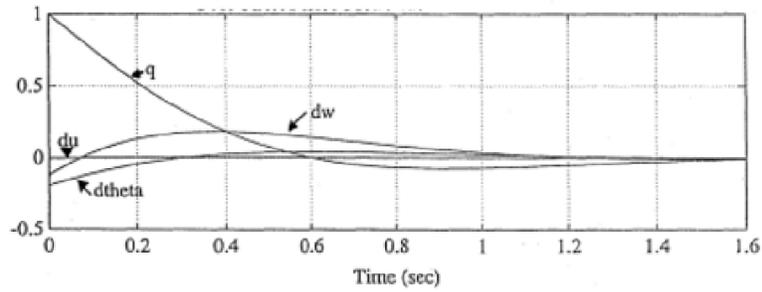


Figure 5.2: FLSim Navion short period mode (bottom) comparison with real Navion [2] (top)

A comparison was also made for the short period mode of the Navion. The FLSim Short period mode was induced with a sharp pull then push of the stick followed by a return to centre. The resulting amplitude of the final longitudinal input and natural stick-fixed response of the Navion yielded a damping ratio of $\zeta_{SP}=0.3665$ between the second and third troughs. Estimating the real Navion short period motion peak values from [2] yields a damping ratio of $\zeta_{SP}=0.4020$.

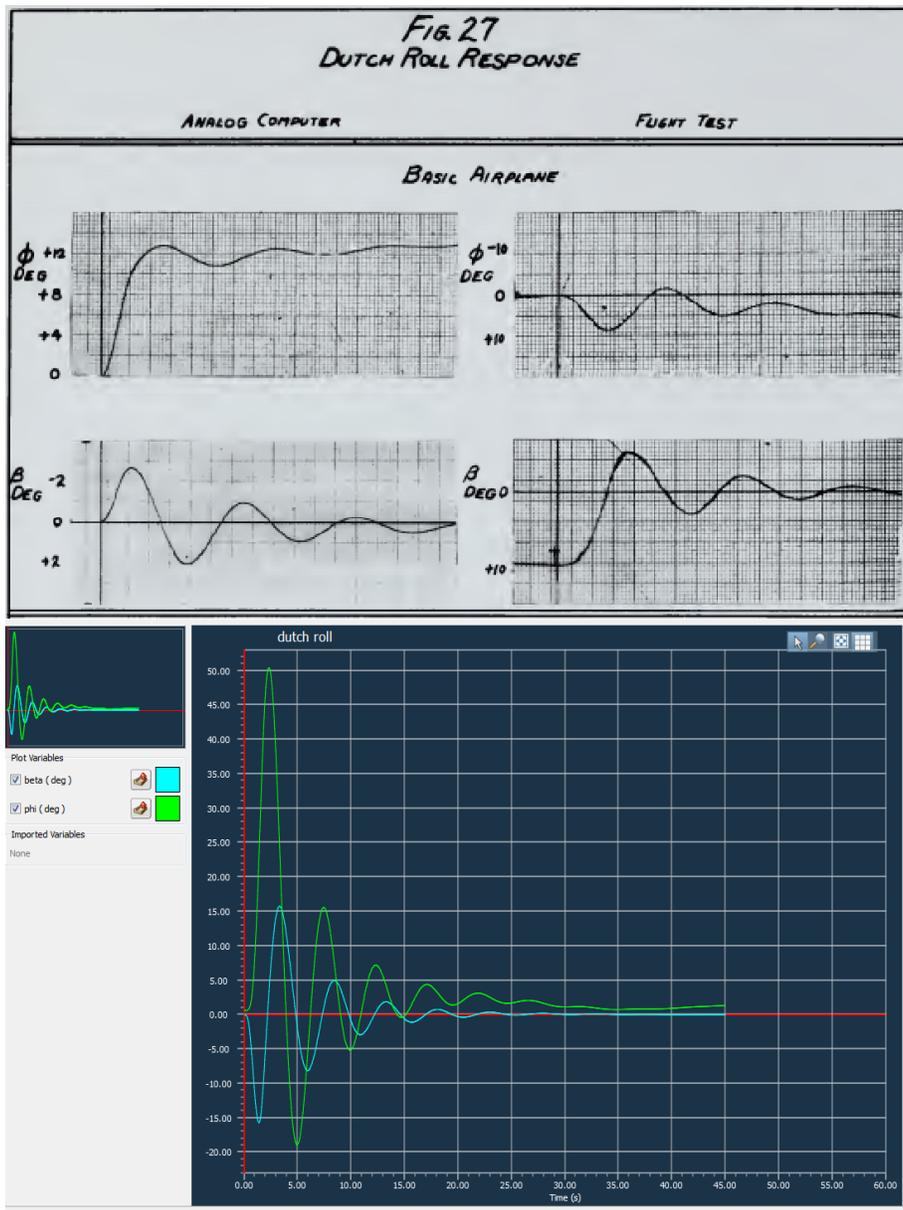


Figure 5.3: FLSim Navion Phugoid mode (bottom) comparison with real Navion [6] (top)

The Dutch roll mode the FLSim Navion was compared to the the real Navion Dutch roll mode as documented in [6] by comparing the second and third peaks of the roll and yaw cycles. Figure 27 in reference [6] displays both analog com-

puter data and a flight test recording of the yawing and rolling motions of a real Navion in Dutch roll mode. Referring to the top of from the top of Figure 5.3, the flight test plots show an approximate yaw damping ratio of $\zeta_{DR,yaw}=0.1916$ and the analog computer approximates yaw damping as $\zeta_{DR,yaw}=0.2104$. The FLSim Navion Dutch roll was induced at $t= 1$ s by inputting 70% right foot pedal and then 70% left foot pedal. The yaw damping ratio is $\zeta_{DR,yaw}=0.1851$ as shown in Figure 5.3 where the blue curve is the sideslip angle, β and the green curve is the roll angle, ϕ . The roll damping was approximated from the plots in [6] as $\zeta_{DR,roll}=0.1103$ and $\zeta_{DR,roll}=0.1213$ for the FLSim Navion. The period of the Dutch roll for the Navion in [6] was approximately 6 seconds and 5 seconds for the FLSim Navion.

Mode Damping Ratio	FLSim prediction	Flight data, Ebbert [6]	% Error
ζ_{phug}	0.0661	0.0748	11%
ζ_{SP}	0.3665	0.4020	9%
$\zeta_{DR,yaw}$	0.2104	0.1916	10%
$\zeta_{DR,roll}$	0.1213	0.1103	10%

Table 5.1: Error between FLSim and real Navion aircraft

The largest error between the FLSim data and the real world Navion plots was found when comparing the phugoid damping ratios. This error reached about 11% which was considered acceptable when approximating data points by reading values of peaks and troughs from plots. The similarity in the FLSim Navion and flight data from the real Navion was the basis used when building the FLSim MQ-9 Guardian.

MQ-9 Guardian Dynamic Modes

The following figures illustrate the MQ-9 Guardian configuration with an expendable aid payload attached where the maritime sensor suite would normally be placed. Figures 5.4 and 5.5 respectively illustrate the long period phugoid motion of the MQ-9 with the payload attached and with the payload dropped just after 20 seconds into flight. Phugoid motion period is observed to be approximately 30 seconds with a damping ratio of $\zeta_{phug}=0.0680$ calculated from the second and third peaks of the diagram and using equation 5.1.

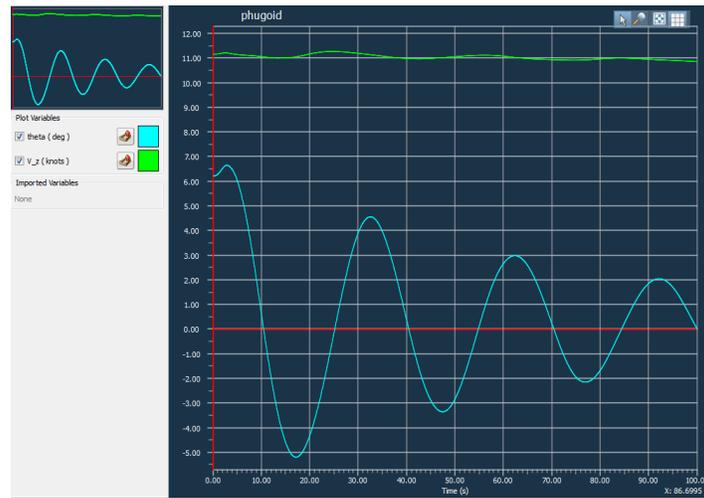


Figure 5.4: MQ-9 phugoid motion with aid payload

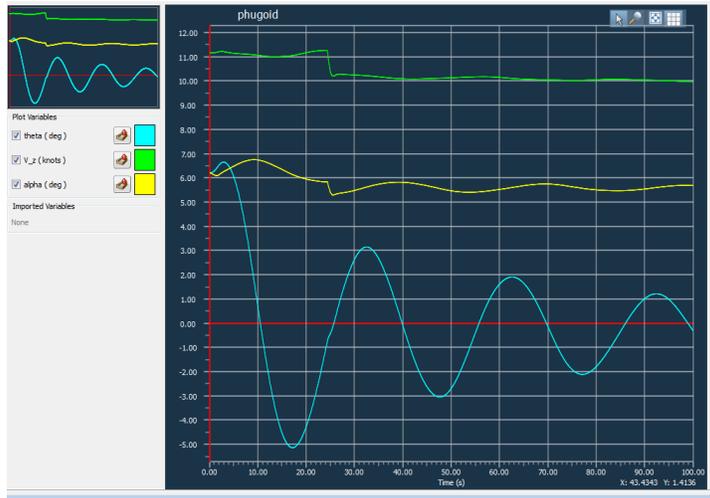


Figure 5.5: MQ-9 phugoid motion with dropped aid payload

The payload drop shows a decrease in α (displayed in yellow), which is expected since the aircraft now needs to create less lift to attain the same speed and attitude. The green and blue curves show vertical speed and pitch angle respectively. Phugoid motion is barely distinguishable and the aid payload's location on the aircraft does not seem to affect the flight of the aircraft in any major way. The short period motion of the Guardian was induced with a single short, sharp pull of the stick and appeared unaffected by the payload drop. After the second trough, the Guardian is so heavily damped that it dives into phugoid motion. The short period plot is shown below with a damping ratio of $\zeta_{SP}=0.5123$ between the very first and second trough.

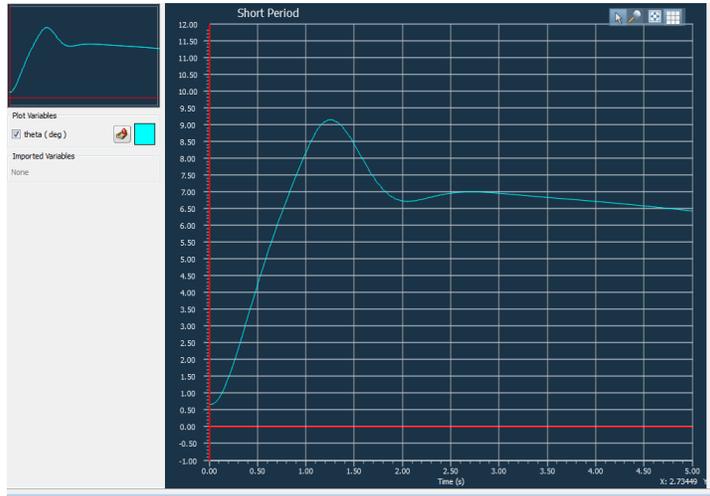


Figure 5.6: MQ-9 short period mode with payload

Dutch roll was induced by a sharp yaw to the left and then to the right of the same magnitude. The Guardian Dutch roll yaw damping ratio calculated from the second and third peaks in Figure 5.7 is approximately $\zeta_{DR,yaw}=0.0813$ and the roll damping ratio is approximately $\zeta_{DR,roll} = 0.0722$. The cycle period is about 9 seconds for the Guardian’s Dutch roll response.

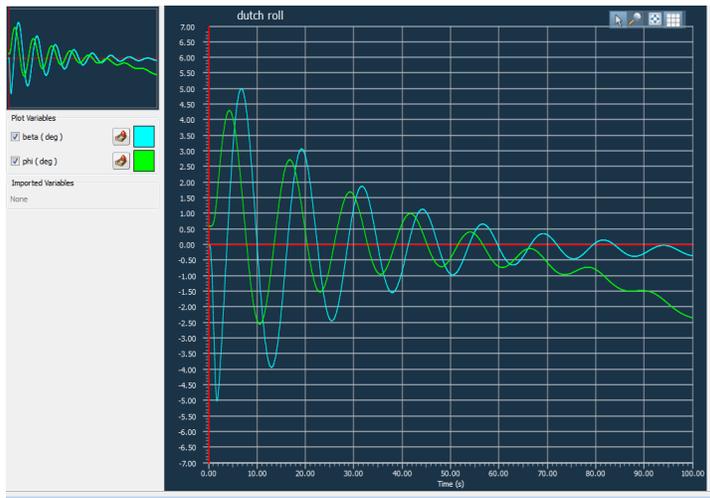


Figure 5.7: MQ-9 Dutch roll response with payload

6 Conclusion

The goal of this project was to build realistic flight models of a given aircraft like the MQ-9 Reaper/Guardian by first finding stability derivatives with AVL and then creating and unifying polygon and flight models using Presagis simulation software. The method was validated by building a well-documented Ryan Navion aircraft using the same method and comparing the model to actual Navion flight data. The combination of evaluating stability derivatives from AVL, finding masses, mass moments of inertia and geometry and inputting data into FLSim creates a high-fidelity aircraft model approximation that can be used to evaluate new aircraft configurations. The Ryan Navion was modelled accurately from a longitudinal stability derivative perspective, and so the same methods were used to model the MQ-9 Guardian as an SAR aircraft carrying a 200 kg aid payload. The Guardian payload drop barely affected its flight performance, and acted as expected. Because the MQ-9 mass moments of inertia and airfoils are proprietary, however, its flight model accuracy is unlikely to be as accurate as the Navion whose properties are well-documented due to its age and time spent as an obsolete aircraft. This method, however, creates an well-approximated flight model with realistic dynamic modes and flight performance.

Future Work and Improvements

More information on the MQ-9 or any other aircraft would yield more realistic results, as the Presagis Flight Sim program is able to model more of a given flight envelope if more data points from other sources can be provided. The modelling method documented in this project could be improved through more stringent analysis of aircraft dimensions, and moment of inertia approximation, which would yield more realistic stability derivatives. Other programs could be used to determine stability derivatives in addition to AVL and other flight envelopes could be explored to determine stability derivatives over a greater range of the aircraft performance.

Bibliography

- [1] Abbot, I., Doenhoff, A. *Theory of Wing Sections Including a Summary of Airfoil Data*. Dover Publications, New York, United States, 1959.
- [2] Bryson, A. *Control of Spacecraft and Aircraft*. Princeton University Press, 1994.
- [3] Buchholdt, H., Moossavi, N. *Structural Dynamics for Engineers, 2nd Ed.* ICE Publishing, London, UK, 2012.
- [4] Caughey, D. *Introduction to Aircraft Stability and Control Course Notes*. Sibley School of Mechanical & Aerospace Engineering Cornell University, Ithaca, New York, 2011.
- [5] CBC News. In Depth Search and Rescue. CBC News Website [Online]. October, 2005. Available: http://www.cbc.ca/news2/background/search_rescue/rescue_aircraft.html. Accessed January 2016.
- [6] Ebbert, E., O'Hara, J. *A Preliminary Evaluation of the Navion as a Lateral-directional Flight Simulator for Use in the Investigation of Flying Qualities Criteria*. Prince University, June, 1960.
- [7] Etkin, B., Reid, L. *Dynamics of Stability and Flight: Stability and Control. 3rd Ed.* John Wiley and Sons, Inc., 1996.
- [8] *FlightSim 14 Training Course Notes* Presagis, 2014.
- [9] GA-ASI's Guardian UAV image [Online]. Available: <http://www.unmannedsystemstechnology.com/2012/10/ga-asi-demonstrates-ads-b-surveillance-system-on-guardian-uav/>. Accessed November, 2015.
- [10] Goldberg, J. *Effects of Spring and Inertia Devices on the Longitudinal Stability of Aircraft*. WADC Technical Report No. 53-350. Wright Air Development Center, USAF, 1953.
- [11] Haber, J. *Enhancing the Functional Design of a Multi-touch UAV Ground Control Station*. Unpublished MaSc Thesis, Ryerson University, 2015.
- [12] Israel Aerospace Industries. IAI Heron image. IAI Website [Online]. <http://www.iai.co.il/Templates/Gallery/GalleryPopup.aspx?docID=37204&lang=en&tabNum=1&num=2>. Israel Aerospace Industries. 2002.
- [13] Karlsrud, J. Norwegian Centre for Humanitarian Studies. June, 2015 [Online]. Available: <http://www.humanitarianstudies.no/2015/09/21/karlsrud-highlights-the-opportunities-and-challenges-linked-to-the-use-of-new-technologies-in-un-peacekeeping/>. Accessed December 2015.

- [14] Meier, P. *The Rise of the Humanitarian Drone: Giving Content to an Emerging Concept*. July 8, 2014 [Online]. Available: <http://isnblog.ethz.ch/international-relations/the-rise-of-the-humanitarian-drone-giving-content-to-an-emerging-concept>. Accessed November, 2015.
- MQ-9 Reaper 3-view drawing. Wikipedia encyclopedia, 2015. [Online]. Available: https://commons.wikimedia.org/wiki/File:MQ-9_Reaper_dimensioned_sketch.png-Reaperdimensions. Accessed December 2015.
- [15] Nelson, R. *Flight Stability and Automatic Control, 2nd Ed.* WCB/McGraw-Hill, United States, 1998.
- [16] Presagis Website. Content Creation images [Online]. 2015. Available: http://www.presagis.com/products_services/products/modelling-simulation/content_creation/. Accessed December, 2015.
- [17] Raymer, D. *Aircraft Design: A Conceptual Approach. 5th Ed.* AIAA Inc., Virginia, 2012.
- [18] Sforza, P. *Commercial Airplane Design Principles*. Butterworth-Heinemann, Oxford, UK, 2014.
- [19] Seckel, E., Morris, J.J. *The Stability Derivatives of the Navion Aircraft Estimated by Various Methods and Derived from Flight Test Data*. Department of Transportation, FAA, Washington, 1971.
- [20] Selig, M., Guglielmo, J., Broeren A., Giguère P. *Summary of Low-Speed Airfoil Data* SoarTech Publications, Virginia, 1995.
- [21] Struett, R. *Empennage Sizing and Aircraft Stability Using MATLAB*. California Polytechnic State University, 2011.
- [22] Suit, W. *Aerodynamic Parameters of the Navion Airplane Extracted from Flight Data*. NASA, Washington, 1972.
- [23] U.S. Air Force. *USAF MQ-9 Reaper Fact Sheet*. September 2015 [Online]. Available: <http://www.af.mil/AboutUs/FactSheets/Display/tabid/224/Article/104470/mq-9-reaper.aspx>. Accessed November, 2015.
- [24] U.S. Air Force. RQ-4 image. USAF fact sheet, USAF website [Online]. Oct 27, 2014. Available: <http://www.af.mil/AboutUs/FactSheets/Display/tabid/224/Article/104516/rq-4-global-hawk.aspx>. Accessed December, 2015.
- [25] Wikimedia Commons Image Database. MQ-9 Reaper Dimensioned Sketch [Online]. 2008. Available: https://commons.wikimedia.org/wiki/File:MQ-9_Reaper_dimensioned_sketch.png. Accessed November, 2015.

Appendices

General aviation airplane: NAVION

Longitudinal	C_L	C_D	$C_{L\alpha}$	$C_{D\alpha}$	$C_{m\alpha}$	$C_{L\dot{\alpha}}$	$C_{m\dot{\alpha}}$	C_{Lq}	C_{mq}	$C_{L\dot{M}}$	$C_{D\dot{M}}$	$C_{m\dot{M}}$	$C_{L\dot{\delta}_e}$	$C_{m\dot{\delta}_e}$
M = 0.158														
Sea level	0.41	0.05	4.44	0.33	-0.683	0.0	-4.36	3.8	-9.96	0.0	0.0	0.0	0.355	-0.923
Lateral	$C_{Y\beta}$	$C_{l\beta}$	$C_{n\beta}$	C_{Yp}	C_{lp}	C_{nr}	C_{lr}	$C_{l\dot{\alpha}}$	$C_{n\dot{\alpha}}$	$C_{Y\dot{\delta}_e}$	$C_{l\dot{\delta}_e}$	$C_{n\dot{\delta}_e}$		
M = 0.158														
Sea level	-0.564	-0.074	-0.071	-0.410	-0.0575	0.107	-0.125	-0.134	-0.0035	0.157	0.107	-0.072		

Note: All derivatives are per radian.

Figure A1: Ryan Navion Derivatives from Nelson [15]

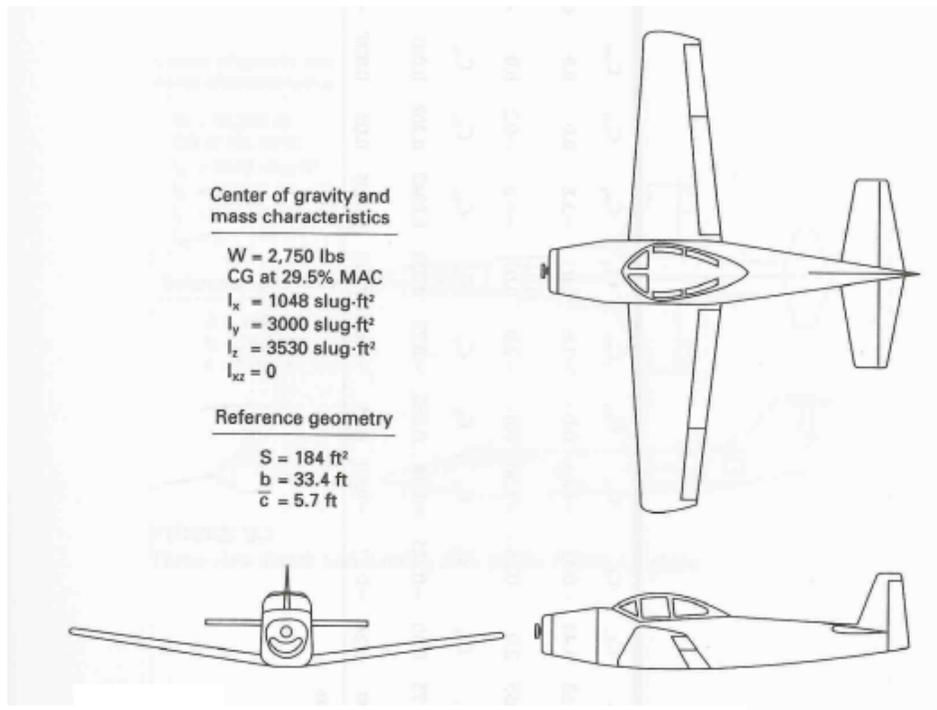


Figure A2: Ryan Navion dimensions and mass properties [15]

TABLE II - LATERAL DIRECTIONAL DYNAMIC DERIVATIVES

Derivative	Textbooks			NACA TR 1098	USAF Datcom	Wind Tunnel	Flight Test
	Perkins and Hage	Etkin	Seckel				
$C_{y\beta}$	-	-.28	-.20	-	-.35	-.77	-.61
$(C_{y\beta})_{vt}$	-	-	-.18	-	-	-	-.61 ^a
$C_{l\beta}$	-.062	-.103	-.077	-.072	-.100	-.109	-.067
C_{l_r}	.070	.084	.067	.062	.130	-.097	.069
C_{l_p}	-.46	-.39	-.42	-.42	-.30	-	-.46
$C_{n\beta}$.073	.094	.033	-	.082	.109	.085
$(C_{n\beta})_{vt}$.109	.139	.090	-	-	.109	.084
C_{n_r}	-.087	-.141	-.065	-	-.120	-	-.088
$(C_{n_r})_{vt}$	-.084	-.139	-.061	-	-	-	-.163
C_{n_p}	-.035	-.017	-.010	-.010	-.030	-	-.038
$C_{l\delta a}$	-	-	-	-	-	.152	.152 ^a
$C_{n\delta a}$	-	-	-	-	-	-.0047	-.0047 ^a
$C_{n\delta r}$	-	-	-	-	-	-.077	-.075
							-.093

NOTES

1. In each box, the number in the upper left corner is for Condition I and the number in the lower right corner is for Condition II.
2. The enboxed numbers are estimates which are considered to be unacceptably erroneous.
3. The asterisks signify numbers which were not found independently during the analog matching procedure (see Discussion).

Figure A3: Ryan Navion NACA-Calculated Lateral Stability Derivatives [19]

TABLE III - LONGITUDINAL DYNAMIC DERIVATIVES

Derivative	Textbooks			USAF Datcom	Wind Tunnel	Flight Test
	Perkins and Hage	Etkin	Seckel			
$C_{L\alpha}$	4.36	4.25	4.54	5.50	4.52	6.04 6.40
$C_{m\alpha}$	-.83	-.715 -.872	-.545 -.545	-1.24	-.95 -1.03	-
$C_{m\dot{\alpha}}$	-3.0	-4.91 -5.23	-4.91 -4.91	-6.58	-	-
$C_{m\dot{q}}$	-9.6	-9.75 -9.75	-9.50 -9.50	-13.29	-	-
$C_{m\dot{\delta}_e}$	-	-	-	-	-1.42 -1.55	-1.42 -1.55 ^o
$C_{m\dot{t}}$	-1.82	-1.68 -1.68	-1.72 -1.72	-2.87	-2.03 -2.19	-
$C_{m\ddot{\alpha}} + \frac{C_{L\alpha} C_{Mq}}{4u}$	-1.08	-1.97 -1.12	-1.81 -.81	-1.68	-1.27* -1.35 ^o	-1.55 -1.37
$C_{m\dot{\alpha}} + C_{m\dot{q}}$	-12.6	-14.7	-14.4 -14.4	-19.8	-	-18.3 -15.5

NOTES

1. In each box, the number in the upper left corner is for Condition I and the number in the lower right corner is for Condition II.
2. The boxed numbers are estimates which are considered to be unacceptably erroneous.
3. The asterisks signify numbers which were not found independently (see Discussion).

Figure A4: Ryan Navion NACA-Calculated Longitudinal Stability Derivatives [19]

TABLE VII.- LONGITUDINAL AERODYNAMIC PARAMETERS OF THE NAVION AIRPLANE EXTRACTED FROM FLIGHT DATA

(a) $V_O = 73.2$ m/sec (240 ft/sec); $\delta_f = 0^\circ$

Parameter	Value	Standard deviation
$C_{X\alpha}$	0.262	0.046
$C_{Z\alpha}$	-4.33	0.007
C_{Zq}	-15.9	2.28
$C_{Z\delta_e}$	-0.511	
$C_{m\alpha}$	-0.63 (-0.77 corrected to 25% c.g.)	0.006
$C_{m\dot{\alpha}}$	-6.5	0.52
C_{mq}	-18.1	
$C_{m\delta_e}$	-1.42	

(b) $V_O = 43.9$ m/sec (144.1 ft/sec); $\delta_f = 20^\circ$

Parameter	Value	Standard deviation
$C_{X\alpha}$	1.37	0.09
$C_{Z\alpha}$	-4.86	0.09
C_{Zq}	-27.13	1.75
$C_{Z\delta_e}$	-0.52	
$C_{m\alpha}$	-0.70 (-0.84 corrected to 25% c.g.)	0.011
$C_{m\dot{\alpha}}$	-6.0	0.25
C_{mq}	-16.4	0.035
$C_{m\delta_e}$	-1.55	

Figure A5: Navion statistically-derived longitudinal stability derivatives [22]

TABLE XI.- LATERAL PARAMETERS

Parameter	V = 73.2 m/sec (240 ft/sec); $\delta_f = 0^\circ$		V = 43.9 m/sec (144.1 ft/sec); $\delta_f = 20^\circ$	
	Value	Standard deviation	Value	Standard deviation
$C_{Y\beta}$	-0.6	0.078	-0.74	0.059
$C_{Y\delta_r}$	0.33	0.05	0.68	0.03
$C_{l\beta}$	-0.07	0.0016	-0.053	0.0012
$C_{l\dot{p}}$	-0.49	0.003	-0.53	0.0037
$C_{l\dot{r}}$	0.11	0.008	0.114	0.0068
$C_{l\delta_a}$	0.154	0.0004	0.16	0.0004
$C_{l\delta_r}$	0.026	0.0018	0.0007	0.0013
$C_{n\beta}$	0.073	0.0004	0.083	0.0005
$C_{n\dot{p}}$	-0.04	0.0005	-0.147	0.0016
$C_{n\dot{r}}$	-0.09	0.005	-0.108	0.0044
$C_{n\delta_a}$	-0.004	8×10^{-6}	-0.0015	0.0001
$C_{n\delta_r}$	-0.063	0.0007	-0.067	0.0005

Figure A6: Navion statistically-derived lateral stability derivatives [22]

Estimates from [18] for $\kappa = \frac{C_{l\alpha}}{(2\pi)(1-Ma^2)^{1/2}} \approx 0.923$ with an ideal wing lift curve slope of 5.730 per radian [1] for the following equation [18].

$$C_{L\alpha} = \frac{2\pi A}{2 + \left[\frac{A^2(1-Ma^2)}{\kappa^2} + 4 \right]^{1/2}} \quad (A1)$$

$$= \frac{2\pi 6}{2 + \left[\frac{6^2(1-0.158^2)}{0.772^2} + 4 \right]^{1/2}} = 4.22$$

Estimates from [17] give $\frac{d\epsilon}{d\alpha} \approx 0.5$. The following equation is from [4]. The following equation is given by [4]. Approximating efficiency for a horizontal tail placed relatively low above the wing and affected by downwash gives $\eta_h \approx 0.6$. Lift slopes are given by [1] and $x_{cg} \approx 0.57$ m at 25% root chord.

$$x_{NA} = x_{cg} + \eta V_h \left(\frac{a_h}{a_w} \right) \left(1 - \frac{d\epsilon}{d\alpha} \right) \quad (A2)$$

$$= 0.57 + 0.60.648 \left(\frac{5.729}{5.729} \right) (1 - 0.5) = 0.864$$

The following equations are given by [4].

$$\begin{aligned} C_{M\alpha} &= \left(\frac{x_{cg}}{\bar{c}} - \frac{x_{NP}}{\bar{c}} \right) C_{L\alpha} \\ &= \left(\frac{0.57}{1.73} - \frac{0.864}{1.73} \right) (4.22) = -0.717 \end{aligned} \quad (\text{A3})$$

The following equation yields tail volume where $l \approx 4.8$ m is the length from the wing aerodynamic centre to the horizontal tail aerodynamic centre.

$$\begin{aligned} V_h &= \frac{l S_h}{\bar{c} S_w} \\ &= \frac{(4.8)(4)}{(1.73)(17.122)} = 0.648 \end{aligned} \quad (\text{A4})$$

Using a similar equation to Equation A4, $V_z \approx 0.041$ [21].

The following equation is given by [4].

$$\begin{aligned} C_{Mq} &= -2\eta_h \frac{l}{\bar{c}} (V_h) a_h \\ &= -2(0.6) \left(\frac{4.8}{1.73} \right) (0.648) (5.729) = -12.3603 \end{aligned} \quad (\text{A5})$$

The following equation is given by [17]. η_v is assumed to be 1 for the lack of a better approximation.

$$\begin{aligned} \frac{d\sigma}{d\beta} &= \eta_v^{-1} \left(0.724 + \frac{3.06 \frac{S_{vs}}{S_w}}{1 + \cos\Lambda} - 0.4 \frac{Z_{wf}}{D_f} + 0.009 A_{wing} \right) \\ &= 0.724 + \frac{3.06 \left(\frac{1.163}{17.122} \right)}{1 + \cos(0)} - 0.4 \frac{2}{1.5} + 0.009(6) = 0.348 \end{aligned} \quad (\text{A6})$$

The following equation is given by [4].

$$\begin{aligned} C_{Y\beta} &= -\eta_v \frac{S_v}{S_w} a_v \left(1 + \frac{d\sigma}{d\beta} \right) \\ &= -\frac{1.163}{17.122} (5.729) (1 + 0.348) = -0.525 \end{aligned} \quad (\text{A7})$$