Optimization of a Hybrid Motor Propelled Model Rocket

by

Ted Kapustka

A Project Submitted to the Graduate Faculty of Rensselaer Polytechnic Institute in Partial Fulfillment of the Requirements for the degree of

MASTER OF ENGINEERING IN MECHANICAL ENGINEERING

Approved:

Professor Ernesto Gutierrez-Miravete, Project Adviser

Rensselaer Polytechnic Institute
Hartford, CT
April, 2011
(For Graduation May 2011)
© Copyright 2011
by
Ted Kapustka
All Rights Reserved
## CONTENTS

Optimization of a Hybrid Motor Propelled Model Rocket ................................................. i

LIST OF TABLES ............................................................................................................. v

LIST OF FIGURES ........................................................................................................... vi

LIST OF KEYWORDS ..................................................................................................... viii

GLOSSARY ...................................................................................................................... ix

ACKNOWLEDGMENT ..................................................................................................... x

ABSTRACT ....................................................................................................................... 1

1. Introduction .................................................................................................................. 2
   1.1 High Power Model Rocket ................................................................................. 2
   1.2 Subsonic, Transonic, and Supersonic Drag ...................................................... 3
   1.3 The Goddard Problem ..................................................................................... 4

2. Methodology ................................................................................................................ 8
   2.1 Overview ........................................................................................................... 8
   2.2 Stability Calculation ....................................................................................... 8
   2.3 Drag Modeling ............................................................................................... 9
   2.4 Flight Simulation ........................................................................................... 10

3. Results and Discussion............................................................................................... 12
   3.1 Stability Analysis ......................................................................................... 12
   3.2 Drag Model ................................................................................................. 12
   3.3 Performance Simulation and Calibration ....................................................... 15
   3.4 Optimum Performance from a Solid Rocket Motor ....................................... 18
   3.5 Optimum Hybrid Motor Design – Physical Characteristics ........................... 19
   3.6 Optimum Hybrid Motor – Thrust Curve ....................................................... 21

4. Conclusions ................................................................................................................ 25

5. References .................................................................................................................. 26

Appendix 1 – Solid and Hybrid Motors .......................................................................... 27
Appendix 2 – Drag Calculations........................................................................................................28
Appendix 3 – Stability Calculations ...............................................................................................36
Appendix 4 – Flight Simulation........................................................................................................41
Appendix 5 – Standard Atmosphere and Misc. Functions..............................................................47
LIST OF TABLES

Table 1 – Comparison of Mass, Center of Pressure, Center of Gravity, and Caliber for Project Simulation and RASAero Software.............................................................12
Table 2 - K-class Solid Motor Performance and Flight Simulation Results...............18
Table 3 - Physical Characteristics and Thrust for Solid and Hybrid Motors..............27
Table 4 – Variables used in Drag Analysis..............................................................28
Table 5 – Variables used in Stability Analysis .......................................................36
LIST OF FIGURES

Figure 1: Tripoli High-Power Rocket Altitude Records Classed by Total Impulse ........ 3
Figure 2 – Drag Coefficient vs. Mach number for 130mm Sounding Rocket .............. 4
Figure 3 - Optimum Thrust Curve with Thrust Constraint – No Mach number Effects –
per reference [2].......................................................................................................... 5
Figure 4 – Variation of Thrust T with Time – With Mach number Effects – per reference
[3]..................................................................................................................................... 5
Figure 5 – Air Density and Gravitational Acceleration Normalized to MSL vs. Altitude
for a Rocket Reaching Geosynchronous Orbit .............................................................. 6
Figure 6 — Air Density and Gravitational Acceleration Normalized to MSL vs. Altitude
for a High-Power Model Rocket................................................................................ 6
Figure 7 – Geometric Parameters used in Drag Calculation........................................ 10
Figure 8 - Thrust vs. Time for Solid K-class Motors with Total Impulse > 2400 N*s.... 11
Figure 9 – Predicted and Measured Drag Coefficient for L65931 Rocket ................. 13
Figure 10 – Predicted and Measured Drag Coefficients for 130 mm Sounding Rocket in
both Burning and Coasting States............................................................................. 13
Figure 11 – Predicted and Measured Drag Coefficients for Aerobee 150A Second Stage
in both Burning and Coasting States......................................................................... 14
Figure 12 – Comparison to RASAero Cd Prediction for LOC Precision/Caliber ISP
Model Rocket.............................................................................................................. 14
Figure 13 - Altitude vs. Time for the LOC Precision/Caliber Isp™ ......................... 15
Figure 14 – Predicted Altitude vs. Time for Baseline Rocket with K300 Solid Motor and
K-class Solid Record................................................................................................. 16
Figure 15 – Predicted Altitude vs. Time K-240 Propelled Rocket and K-class Hybrid
Record....................................................................................................................... 17
Figure 16 – C_d vs. Mach number for K-240 Propelled K-class Hybrid.................... 17
Figure 17 – Altitude vs. Average Thrust for Baseline Rocket Propelled by Selected K-
class Solid Motors.................................................................................................... 19
Figure 18 – K-class Motor Total Impulse vs. Weight.................................................. 20
Figure 19 – Total Impulse vs. Motor Length Normalized to 54mm diameter.......... 20
Figure 20 – Total Impulse vs. Motor Length Normalized to 64mm diameter.......... 21
Figure 21 – Altitude vs. Thrust for Rockets using 54mm and 64mm Hybrid Motors with Single-Level Thrust Curves ................................................................. 22

Figure 22 - Two-Level Thrust Curve Optimized for 54mm Rocket......................... 22

Figure 23 – Maximum Altitude vs. Secondary Thrust Level for the 54mm Motor........ 23

Figure 24 - Maximum Altitude vs. Secondary Thrust Level for the 64mm Motor ....... 23

Figure 25 - Altitude vs. Time for Optimized Hybrid Propelled Rocket .................... 24

Figure 26 - Velocity vs. Time for Optimized Hybrid Propelled Rocket................... 24

Figure 28 – Altitude, Velocity, Acceleration, and Jerk vs. Time .............................. 41
LIST OF KEYWORDS

Center of pressure
Drag coefficient
Flight simulation
High power rocket
Hybrid rocket motor
Stability
Tripoli Rocketry Association
GLOSSARY

Acrylonitrile butadiene styrene (ABS) – a common thermoplastic
Ammonium perchlorate (AP) – An oxidizer commonly used in solid rocket motors
Center of pressure (C.P.) - The point on a body where the total sum of a pressure field acts, causing a force and no moment about that point
Center of gravity (C.G.) – The point in a body or system around which its mass or weight is evenly distributed or balanced and through which the force of gravity acts.
Drag coefficient \( C_d \) – A dimensionless quantity used to quantify the drag or resistance of an object moving through a fluid.
Geosynchronous orbit – An orbit around the earth with an orbital period that matches the Earth’s sidereal rotation speed
K-class – High power rocket or motor with less than 2560Ns of total impulse
LOC Precision/Caliber Isp – Model rocket kit manufactured by LOC Precision
Tripoli Rocketry Association, Inc – a non-profit organization dedicated to the advancement and operation of amateur high power rocketry.
ACKNOWLEDGMENT

I wish to acknowledge the many individuals who have contributed to the wealth of model rocketry knowledge available on the web. Special thanks to Jon Champion, Robert A. Braeunig, and Rick Newlands whose work I found particularly helpful for this project.
ABSTRACT

This report documents design and flight simulation tools that can be used to optimize a high-power model rocket and the hybrid motor that propels it. The motivation of this project is beating the existing K-class rocket records in both hybrid and solid classes. Existing altitude records are 13,040 feet (3974.6 m) for hybrid propelled rockets and 29,266 feet (8920.3 m) for solid propelled rockets. The design tools calculate the center of gravity and center of pressure which are used to predict the stability of the rocket. A flight simulation is performed to optimize the physical characteristics of the rocket and thrust curve of the motor to maximize the altitude at apogee. The simulation includes a Mach-number dependant drag prediction that is based on the rocket’s dimensions, and is calibrated to drag coefficients obtained through test. An optimum thrust curve that employs two thrust levels is proposed.
1. Introduction

1.1 High Power Model Rocket

In recent years, high power model rockets have become increasingly popular, and thanks to media attention, the sport is becoming somewhat less obscure. In 2009, the successful launching of a 36-ft.-tall (11.0 m), 1648 pound (748 kg) scale model of the historic Saturn-V gained media attention when it broke both size and weight records for amateur rockets. Not all high-power rockets are as audacious---in fact a high-power rocket can weigh as little as 1.5 kg providing it has one or more motors with more than 62.5 grams of propellant or more than 160 N-s of total impulse. These rockets compete for various records including time to 1000 feet (304.8 m) and maximum altitude.

High-power model rockets routinely reach altitudes of 10,000 feet (3048 m) with some rockets reaching altitudes in excess of 45,000 feet (13,716 m). High power rocket altitude records are certified by Tripoli Rocket Association, Inc. For the purpose of establishing altitude records; rockets are grouped based on the total impulse of their motor and the type of motor. Motor types include black powder, other solid fuel such as ammonium perchlorate/aluminum, and hybrid. In a hybrid motor the fuel and oxidizer are different phases, e.g. gaseous nitrous oxide ($N_2O$) as the oxidizer and solid ABS plastic as a fuel. The fuel and oxidizer are kept separate until they are mixed in the combustion chamber. As shown in Figure 1, record altitudes for the solid-propelled rockets far exceed the record altitudes for hybrid-powered rockets. The motivation of this project is beating the existing K-class rocket records in both hybrid and solid classes. Existing altitude records are 13,040 feet (3974.6 m) for hybrid propelled rockets and 29,266 foot (8920.3 m) for solid propelled rockets.
1.2 Subsonic, Transonic, and Supersonic Drag

Flow through a compressible medium is divided into four ranges based on Mach number: subsonic, transonic, supersonic, and hypersonic. The transition from subsonic to transonic flow occurs at a Mach number of approximately 0.8 and the transition to supersonic occurs at a Mach number of approximately 1.2. Mach numbers greater than 5 are hypersonic and will not be considered here. The following simplified explanation of the mechanisms that influence drag is based on [1]. At subsonic speeds, the drag on a rocket is composed of two main components: skin friction and pressure drag. Induced drag (drag due to lift) is generally small for a well designed and constructed rocket. As a rocket transitions from subsonic to transonic flight there is a substantial increase in the total drag due to changes in the pressure distribution acting on the rocket. This increase in drag is called wave drag. Wave drag is due to the formation of shock waves which transform energy into heat and to flow separation downstream of the shocks. Figure 2 shows drag coefficient ($C_d$) vs. Mach number data for a rocket. The drag coefficient in the transonic range is higher than in the supersonic range because of erratic shock formation and flow instabilities. The drag coefficient is reduced as flow transitions from transonic to supersonic. For rockets, $C_d$ is typically based on the maximum diameter.
\[ Drag = \frac{1}{2} \rho V^2 C_d A = \frac{\pi \rho V^2 D^2}{8} \]

where,

\[ \rho = \text{Air density, kg/m}^3 \]
\[ V = \text{Velocity, m/s} \]
\[ D = \text{Diameter, m} \]

![Figure 2 – Drag Coefficient vs. Mach number for 130mm Sounding Rocket](image)

1.3 The Goddard Problem

The problem of maximizing the altitude of a rocket with a fixed amount of propellant is widely known as the Goddard Problem. Since Robert Goddard first examined this problem in his 1919 publication, “A Method of Reaching Extreme Altitudes”, there has been considerable attention in literature. In [2] the problem is solved both with a thrust constraint (thrust must not exceed some fixed maximum value) and the case when thrust and dynamic pressure are constrained. Consider the problem with only thrust constraint. Drag is a function of altitude (density variation) and the square of velocity ---it is independent of Mach number. The resulting optimum thrust vs. time curve is shown in Figure 3. Note, thrust and time are shown as dimensionless
quantities. The curve consists of three segments: maximum thrust, reduced thrust with ramp-up, and coast.

Figure 3 - Optimum Thrust Curve with Thrust Constraint – No Mach number Effects – per reference [2]

Reference [3] provides a solution that includes Mach-number-dependant drag by sharply increasing drag in the transonic region. The authors conclude that the optimum thrust curve may be more complex than the classical three-segment curve. Figure 4 shows one such solution where five segments are used. Again, thrust and time are dimensionless quantities.

Figure 4 – Variation of Thrust T with Time – With Mach number Effects – per reference [3]
These two examples of optimized thrust curves are for rockets intended to reach earth orbit, which for a geosynchronous orbit is 35,786 km. As shown in Figure 5, the large rocket experiences drag over a very small portion of its flight (approximately the width of the line representing the y-axis). It also experiences a significant change in acceleration due to gravity. The model rocket sees nearly constant gravitational acceleration, and although the air density is reduced to approximately 1/3 at apogee, drag is always important.

![Figure 5 – Air Density and Gravitational Acceleration Normalized to MSL vs. Altitude for a Rocket Reaching Geosynchronous Orbit](image)

![Figure 6 — Air Density and Gravitational Acceleration Normalized to MSL vs. Altitude for a High-Power Model Rocket](image)
The thrust curves of Figure 3 and Figure 4 are attainable by large, complex commercial hybrid motors which have demonstrated throttling capabilities in excess of 8:1, and the ability to stop and restart [4]. Since the model rocket will see less variation in drag than its full-size brother, the thrust curve can be less complex.
2. Methodology

2.1 Overview

Two tools are developed to enable optimization of the model rocket. The first calculates center of gravity and center of pressure, so that the stability of a candidate rocket design can be verified. The second is a flight simulation tool that allows the performance of the model rocket to be evaluated while varying its various physical characteristics, such as mass, length, diameter, and fin dimensions. The flight simulation allows a variety of thrust curves to be evaluated to determine the curve that produces the maximum altitude. The goal is to choose a thrust curve that maintains Mach number less than 0.9, which reduces drag loss and maximizes altitude performance [5].

The tools are developed in Visual Basic and run in an MS Excel spreadsheet. The tools are launched from a form from which the rocket and motor are selected. Pushbuttons are used to execute a Cd vs. Mach number analysis, stability analysis, or flight simulation.

With the dimensions and weight of commercially available high-performance rocket kits, as a starting point, a “baseline” rocket is designed. The rocket is designed around the K300 motor since it was used to set the K-class altitude record. The diameter of the rocket is just large enough to hold the K300 motor and is just long enough to contain the motor, recovery system, and altimeter. A commercially available 5:1 (length:diameter) ogive nose is used. The body tube and fins are carbon fiber. The flights are simulated with the baseline rocket to verify reasonable accuracy of the simulator. Other rocket configurations that are necessary for the project are scaled from the baseline.

2.2 Stability Calculation

In order to maintain stable flight the center of gravity (C.G.) must be located forward of the aerodynamic center of pressure (C.P.). A widely used criterion for model rockets is to maintain the C.G. forward of the C.P. by a distance equal to or greater than the maximum diameter, or caliber, of the rocket. It is common practice to describe the stability of a rocket in caliber, e.g. if the C.G is forward of the C.P. by 1.5 diameters, the rocket has 1.5 caliber of stability.
The Barrowman method [10] is used to determine the location of the center of pressure. This method has been shown to be accurate for Mach numbers below 0.6 [11]. This method accommodates conical and ogive noses, cylindrical bodies, conical shoulders and boat tails, and fins. Due to drag concerns conical noses, shoulders, and boat tails should be avoided, so these aspects of the stability calculations are not implemented.

The center of gravity of the rocket is also calculated. The center of gravity calculation includes the nose, body tube, altimeter (payload), recovery system, fins, motor mount (10% of mass), motor, and 10% additional mass to account for items such as bulkheads, recovery shock cord, recovery fittings, etc. The mass and center of gravity of the individual rocket components are scaled from the baseline rocket. The nose and body are scaled linearly with length and diameter. The recovery system is scaled by burnout mass. Fin mass is calculated from fin volume.

Since the motor mass is significant, the center of gravity is calculated for a specific rocket and motor combination. The stability is calculated both for the launch and burnout motor masses. The rocket can be made more stable by increasing the size of the fins, but this increases drag. Ideally, the fins should be just large enough to maintain stability for all expected C.G. locations. The rocket nose tip is used for the reference datum. Details of the stability calculations and code are found in Appendix 3.

2.3 Drag Modeling

An important element of a flight simulation is a model that calculates drag for subsonic, transonic, and supersonic flight as a function of altitude. The drag calculations provided in [6] agree very well with wind tunnel data for the L65931 rocket for Mach numbers between 0.7 and 1.5. Details of the L65931 rocket and test data are available in [7]. In addition, these procedures show excellent agreement with wind tunnel data for a 130mm sounding rocket for both burning and coasting conditions. This method is used to calculate the components of drag vs. Mach number and to calculate drag in the flight simulation. Two additional rocket geometries are used to calibrate the drag model: 1) The Aerobe 150A rocket geometry and drag data that are available in [8], and 2) The
LOC Precision/Caliber Isp™ rocket geometry and drag coefficient prediction using RASAero Aerodynamic Analysis and Flight Simulation Software available in [9]. The drag calculations account for air density using the standard atmosphere for a standard day. Equations used in [6] are based on feet and inches. A number of constants were modified for SI units. Geometric parameters for the drag calculation are shown in Figure 7, and the equations and computer code are found in Appendix 1.

2.4 Flight Simulation

Two solid-propelled rocket flights are simulated to verify the simulation. First the LOC Precision/Caliber Isp™ for which the rocket geometry, mass, and engine model are available in [9]. Secondly, the K-class record flight is simulated using the baseline rocket and the motor used for the record flight. The K-class hybrid record flight is also simulated using a rocket designed for the K240 hybrid motor.
The “baseline” rocket is flown using the simulator with the available K-class solid motors including the record setting K300 to verify that altitude predictions are reasonably accurate. As shown in Figure 8, there is considerable variation in the average thrust and burn time for motors with similar total impulse.

![Figure 8 - Thrust vs. Time for Solid K-class Motors with Total Impulse > 2400 N*s](image)

Commercially available hybrid motors are longer and have greater mass than solid motors of similar total impulse. In order to simulate a hybrid-propelled rocket, the length and mass of the baseline model are increased accordingly. Ground rules for the hybrid design are: 1) the thrust variation with time must be accomplished as simply as possible and not require valves or electronic controls, 2) only two thrust levels are used, and 3) the maximum turn-down ratio is 8:1.
3. Results and Discussion

3.1 Stability Analysis

The LOC Precision/Caliber Isp\textsuperscript{TM} rocket is modeled to check the accuracy of the stability calculations. A comparison of my results and the RASAero software are shown in Table 1. RASAero is commercially available software for analysis of model rocket performance. The prediction accuracy is good with the error in stability of approximately 0.3 caliber.

Table 1 – Comparison of Mass, Center of Pressure, Center of Gravity, and Caliber for Project Simulation and RASAero Software

<table>
<thead>
<tr>
<th></th>
<th>Project Simulation</th>
<th>RASAero</th>
<th>Error</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Launch Burnout</td>
<td>Magnitude Units</td>
<td></td>
</tr>
<tr>
<td>Mass</td>
<td>1.572 1.348</td>
<td>1.55</td>
<td>1.4 %</td>
</tr>
<tr>
<td>COP</td>
<td>1.19 1.96</td>
<td>1.19</td>
<td>0.006 m</td>
</tr>
<tr>
<td>CG</td>
<td>1.007 1.05</td>
<td>1.02</td>
<td>0.03 m</td>
</tr>
<tr>
<td>Caliber</td>
<td>2.4 1.9</td>
<td>2.2</td>
<td>-0.301 caliber</td>
</tr>
</tbody>
</table>

3.2 Drag Model

Prior to using the drag model in the flight simulation several rockets were modeled and the results compared to drag coefficients obtained experimentally. As shown in Figure 9, the predictions match the L-65931 rocket data very well in all three ranges of Mach number. Drag predictions for the 130mm sounding rocket, Figure 10, and the Aerobee 150A stage 2, Figure 11, agree well in for subsonic flow. The measured Cd in the transonic range is higher than predicted. A comparison of the predicted drag coefficients used in the flight simulator and the RASAero software [9] is shown in Figure 12. The result is similar to the 130mm sounding rocket and the Aerobee 150A where transonic drag coefficient is under predicted. Because the drag model used in the simulation may under predict drag coefficients for transonic flight, actual drag on rockets which fly in this range will higher than predicted, and the benefit of keeping below this range will be greater than predicted.
Figure 9 – Predicted and Measured Drag Coefficient for L65931 Rocket

Figure 10 – Predicted and Measured Drag Coefficients for 130 mm Sounding Rocket in both Burning and Coasting States
Figure 11 – Predicted and Measured Drag Coefficients for Aerobee 150A Second Stage in both Burning and Coasting States

Figure 12 – Comparison to RASAero Cd Prediction for LOC Precision/Caliber ISP Model Rocket

Of the four rockets for which the drag coefficient was calculated, only one---the LOC Precision/Caliber Isp\textsuperscript{TM} has a launch lug. The launch lug extends out of the side of the rocket and engages a vertical rail that extends from the launch pad, enabling the rocket to launch in the intended direction. Initially, the drag model produced very high subsonic drag coefficients due to the launch lug. This was reduced to match the average subsonic Cd from the RASAero results. Launch lugs are not necessary with certain types of launch equipment---none will be used in the flight simulations.
3.3 Performance Simulation and Calibration

The flight simulation is calibrated against flight data for a rocket of known configuration and motor. In addition, the K-class solid and hybrid record flights will be used to calibrate the simulation. Only the maximum altitude and motor used are known, so the rocket configurations are optimized around the motors.

Flight data for a LOC Precision/Caliber Isp™ propelled with a Cesaroni I205 motor is used to calibrate the performance simulation. Figure 13 shows good agreement between the simulation and flight data with an error in maximum altitude of 8%.

![Altitude vs. Time](image)

Figure 13 - Altitude vs. Time for the LOC Precision/Caliber Isp™

The dimensions and weight of commercially available high-performance rocket kits were used as a starting point for the optimized rocket. The flight simulation is used to optimize the rocket configuration. For the K-class, solid-motor record the rocket is based on a 54mm airframe diameter which is just large enough to fit the motor. Overall length is reduced by reducing the payload section to that required to hold the altimeter (10cm). The fins provide 2.8 caliber of stability. A stability of greater than one caliber is acceptable. Figure 14 shows the resulting altitude vs. time and the altitude record. The flight simulation predicts the maximum altitude within 2% of the actual.
An additional performance calibration is done using the K-class, hybrid record of 3975m (13,040ft), which was set using a Hypertec K240 motor. The K240 is 81mm in diameter and is 55cm long. This necessitates an increase in rocket OD from 57.2mm to 84.2mm and an increase in the size of the fins to maintain positive stability. The overall length can be reduced from 1.219m to 1.119m. Figure 15 shows the K-class hybrid record altitude and prediction for the K240 motor. The prediction is approximately 13% low. Replacing the K240 hybrid motor, which has 1292 Ns of total thrust, with the lighter, 2546-Ns K300 increases the maximum predicted altitude to 5210m, closing gap to the solid motor record by about 1/3. The remainder of the disparity between the K-class solid and hybrid records is due to its larger diameter and fins---changes that were made to accommodate the hybrid motor’s diameter and mass. For peak performance the hybrid motor must be dimensioned to minimize airframe drag.
Figure 15 – Predicted Altitude vs. Time K-240 Propelled Rocket and K-class Hybrid Record

The simulation shows that only 4.4s of the 24.7s to apogee are above a Mach number of 0.8. Figure 16 gives some insight into the components of drag that affect the performance of the K240 propelled rocket. Below Mach 0.8 the components of drag are dominated by body friction and base drag. The size of the fins can be changed to maintain stability without significantly changing the overall drag.

Figure 16 – $C_d$ vs. Mach number for K-240 Propelled K-class Hybrid
3.4 Optimum Performance from a Solid Rocket Motor

Table 2 shows flight simulation results for the baseline rocket with various K-class motors with total impulse greater than 2400 Ns. These motors are a subset of the available K-class motors selected for their high total impulse. The total impulse variation within this group is less than 8%. This data suggests that the current K-class record, which was set using a K300, could have been slightly higher using a K250. As expected, motors with the highest average thrust had the highest maximum acceleration and Mach number. They would likely require structural strengthening and protection from high stagnation temperatures. The additional mass would reduce maximum altitude particularly for the K660 and K1050. The average thrust varies from 250 N to 1050 N as indicated in the numeric portion of the motor designation. As shown in Figure 17, the highest altitudes are reached by motors with the lowest average thrust. All of these motors had sufficient thrust to maintain a significant amount of time in transonic and supersonic flight and reach Mach numbers of 2.0 or higher.

<table>
<thead>
<tr>
<th>Motor</th>
<th>Total impulse</th>
<th>Average Thrust</th>
<th>Maximum Acceleration</th>
<th>Maximum Mach number</th>
<th>Altitude</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ns</td>
<td>N</td>
<td>g</td>
<td>m</td>
<td></td>
</tr>
<tr>
<td>K250</td>
<td>2353</td>
<td>250</td>
<td>15.0</td>
<td>2.3</td>
<td>9471</td>
</tr>
<tr>
<td>K300</td>
<td>2546</td>
<td>304</td>
<td>17.9</td>
<td>2.2</td>
<td>9254</td>
</tr>
<tr>
<td>K575</td>
<td>2493</td>
<td>578</td>
<td>19.3</td>
<td>2.0</td>
<td>6901</td>
</tr>
<tr>
<td>K660</td>
<td>2437</td>
<td>659</td>
<td>40.8</td>
<td>2.8</td>
<td>8080</td>
</tr>
<tr>
<td>K1050</td>
<td>2530</td>
<td>1050</td>
<td>59.5</td>
<td>3.0</td>
<td>7644</td>
</tr>
</tbody>
</table>
3.5 Optimum Hybrid Motor Design – Physical Characteristics

There are numerous K-class hybrid motors commercially available today. Key physical characteristics and performance of these motors are analyzed to determine how the rocket geometry must be modified to accommodate the motor and how much the mass will change relative to the K300 solid motor. The best achieved performance based on the available motors is scaled to the maximum allowable total impulse and used for the Optimum Motor. We start with mass. Figure 18 shows mass vs. total impulse for the motors. The motor with the best achieved mass/total impulse is scaled to 2560 Ns total impulse. The resulting hybrid motor mass is 3.5 kg---1.2 kg greater than the K300 motor. In a similar manner, the minimum length for the hybrid motor is determined. The volume of each hybrid motor is calculated, and with either 54mm or 64mm diameter, used to calculate an equivalent length. The resulting lengths are plotted on Figure 19 and Figure 20. The resulting minimum motor lengths are 1.08m for a 54mm diameter motor and 0.75m for a 64mm diameter motor; this increases the rocket length by 0.43m and 0.1m respectively. Detailed motor data is shown in Appendix 1.
Figure 18 – K-class Motor Total Impulse vs. Weight

Figure 19 – Total Impulse vs. Motor Length Normalized to 54mm diameter
3.6 Optimum Hybrid Motor – Thrust Curve

The length, liftoff mass, and burnout mass of the 54mm and 64mm motors are used to determine the required rocket dimensions. The resulting rocket/engine combination is checked for stability and the fins modified as necessary. The simulation is used to determine the maximum achievable altitude using a single-level thrust curve. The results, which are shown in Figure 21, fall far short of the K-class solid record of 8950m. For the 54mm motor, the best performance is at a thrust level of 125N with the altitude falling sharply at lower levels. The altitude for a 43N thrust level would be zero since this is the weight of the rocket at launch.
A simple, two-level thrust curve as shown in Figure 22 is proposed for the hybrid motor. The first segment is 400N. To maximize altitude the time of the first segment and the thrust level of the second segment are varied. The time of the second segment is adjusted to maintain 2560Ns of total impulse. As shown in Figure 23, there is a significant improvement in performance relative to a single thrust level; in fact the maximum attained altitude for the 54mm motor is 8421m which is only 529m less than the K-class solid record. Results for the 64mm rocket are slightly less favorable with a maximum altitude of 8337m---613m short of the record. Results for the 64mm motor are shown in Figure 24.
Figure 23 – Maximum Altitude vs. Secondary Thrust Level for the 54mm Motor

Figure 24 - Maximum Altitude vs. Secondary Thrust Level for the 64mm Motor

The Comparing altitude vs. time for the 54mm motor with single-level and two-level thrust curves as in Figure 25 is useful. All of the altitude advantage for two-thrust levels is obtained in the first 12-13 seconds. This advantage can be explained if we look at the velocity vs. time plot in Figure 26. The two-level motor reaches a velocity of 250m/s in the first three seconds of flight which allows subsequent thrust to do more work than for the slower single-level motor. Consider the velocity at five seconds. The
rate of work done by the single-level motor is $125N \times 100m/s = 12,500Nm/s$. The two-level motor is providing work at a rate of $65N \times 250m/s = 16,250Nm/s$. More power for half the thrust.

![Altitude vs. Time for Optimized Hybrid Propelled Rocket](image1)

Figure 25 - Altitude vs. Time for Optimized Hybrid Propelled Rocket

![Velocity vs. Time for Optimized Hybrid Propelled Rocket](image2)

Figure 26 - Velocity vs. Time for Optimized Hybrid Propelled Rocket
4. Conclusions

Drag – Minimizing drag is key to attaining high altitudes from model rockets. An important element of low drag is minimum diameter. The motor designer must make it a priority to minimize the motors diameter since it limits the minimum diameter or the rocket.

Stability – The relative position of the center of pressure and center of gravity determine the stability. A rocket with inadequate stability must be modified either by moving the C.G. forward or the C.P. rearward. Moving the C.P. rearward by increasing the size of the fins increases drag.

Hybrid Motor Optimization – There is wide variation in the design and performance of commercially available hybrid motors. When the lightest and lowest volume motors are scaled to the K-class maximum 2560Ns total impulse the resulting motor is heavier and longer than its solid equivalent.

Thrust curve optimization – A motor with an optimized two-level thrust curve improves the performance of the rocket over what can be attained with single-level thrust and the same total impulse.

Altitude Records – This study has shown that the K-class hybrid altitude record can easily be broken by a motor and rocket optimized for low drag and with an optimized thrust curve. The optimized hybrid propelled rocket fall short of the overall K-class record, and since solid motors enjoy both mass and volume advantages, they probably will remain dominant. They to can employ two-level thrust by incorporating a multi-fin grain.
5. References


[7] Flight Investigation at Mach Numbers from 0.8 to 1.5 to Determine the Effect of Nose Bluntness on the Total Drag of two Fin-Stabilized Bodies of Revolution, Roger G. Hart, NACA Tech Note 3549, 1955


[10] The Theoretical Prediction of the Center of Pressure, Presented as a Research and Development Project at NARAM-8, James S. Barrowman and Judith A. Barrowman, August 18, 1966

## Appendix 1 – Solid and Hybrid Motors

### Table 3 - Physical Characteristics and Thrust for Solid and Hybrid Motors

<table>
<thead>
<tr>
<th>Mfg</th>
<th>Name</th>
<th>Type</th>
<th>Dia.</th>
<th>Length</th>
<th>Volume</th>
<th>Length normalized to 54mm</th>
<th>Length normalized to 64mm</th>
<th>Isp</th>
<th>Official Total Wt</th>
<th>Nitrous Wt</th>
<th>Total Launch Wt (w/ N2O)</th>
<th>Recovery Wt</th>
<th>Grain Burned</th>
<th>Fuel Grain Length</th>
<th>Fuel Grain Wt</th>
<th>Total Impulse</th>
<th>Max thrust</th>
<th>Nitrous Wt</th>
<th>Oxidizer Fuel Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cesaroni</td>
<td>K300</td>
<td>Reload</td>
<td>54</td>
<td>65</td>
<td>1486</td>
<td>65</td>
<td>46</td>
<td>2270</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aerotech</td>
<td>K250</td>
<td>Single Use</td>
<td>54</td>
<td>67</td>
<td>1541</td>
<td>67</td>
<td>48</td>
<td>2211</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AeroTech</td>
<td>K485</td>
<td>hybrid</td>
<td>54</td>
<td>70</td>
<td>1601</td>
<td>70</td>
<td>50</td>
<td>2220</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contrail</td>
<td>K234</td>
<td>hybrid</td>
<td>54</td>
<td>126</td>
<td>2893</td>
<td>126</td>
<td>90</td>
<td>110</td>
<td>2063</td>
<td>1147</td>
<td>3210</td>
<td>1678</td>
<td>385</td>
<td>30</td>
<td>450</td>
<td>1657</td>
<td>478</td>
<td>1490</td>
<td>3.0</td>
</tr>
<tr>
<td>SkyRipper</td>
<td>K257</td>
<td>hybrid</td>
<td>54</td>
<td>91</td>
<td>2093</td>
<td>91</td>
<td>65</td>
<td>184</td>
<td>1541</td>
<td>870</td>
<td>2411</td>
<td>1451</td>
<td>90</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RATT</td>
<td>K240</td>
<td>hybrid</td>
<td>64</td>
<td>91</td>
<td>2921</td>
<td>128</td>
<td>91</td>
<td>143</td>
<td>2814</td>
<td>1218</td>
<td>4032</td>
<td>2814</td>
<td>96</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contrail</td>
<td>K404</td>
<td>hybrid</td>
<td>75</td>
<td>102</td>
<td>4489</td>
<td>196</td>
<td>140</td>
<td>131</td>
<td>4150</td>
<td>1579</td>
<td>5729</td>
<td>3832</td>
<td>318</td>
<td>25</td>
<td>703</td>
<td>2441</td>
<td>1036</td>
<td>2050</td>
<td>5.0</td>
</tr>
<tr>
<td>WestCoast</td>
<td>K460</td>
<td>hybrid</td>
<td>75</td>
<td>76</td>
<td>3366</td>
<td>147</td>
<td>105</td>
<td>149</td>
<td>3650</td>
<td>980</td>
<td>3650</td>
<td>2290</td>
<td>380</td>
<td>25</td>
<td>1988</td>
<td>768</td>
<td>2.6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hypertek</td>
<td>K240</td>
<td>hybrid</td>
<td>81</td>
<td>55</td>
<td>2844</td>
<td>124</td>
<td>88</td>
<td>167</td>
<td>1807</td>
<td>(1)</td>
<td>2508</td>
<td>1009</td>
<td>789</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- nitrous density = 0.77 g/cc at 70F
- K460 Certified by CAR - Official wt includes Nitrous
- (1) - nitrous weight estimated from propellant weight at 8:1
Appendix 2 – Drag Calculations

Drag Calculations derived from work of Jon Champion – Converted to SI units

Table 4 – Variables used in Drag Analysis

<table>
<thead>
<tr>
<th>Reference Variable</th>
<th>Code Variable</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ν</td>
<td>.KVis</td>
<td>m²/s</td>
<td>Kinematic viscosity</td>
</tr>
<tr>
<td>λ</td>
<td>.Lambda</td>
<td>---</td>
<td>Fin chord ratio</td>
</tr>
<tr>
<td>ΔCdₚ</td>
<td>delCds</td>
<td>---</td>
<td>Supersonic drag rise</td>
</tr>
<tr>
<td>ΔCdₜ</td>
<td>delCdt</td>
<td>---</td>
<td>Transonic drag rise</td>
</tr>
<tr>
<td>A</td>
<td>.Xpro</td>
<td>m²</td>
<td>Maximum cross-section area of protuberance</td>
</tr>
<tr>
<td>a</td>
<td>Not used</td>
<td>m/s</td>
<td>Speed of sound</td>
</tr>
<tr>
<td>a, b, c, g, Fts, x</td>
<td>a, b, c, g, Fts, Xts</td>
<td>---</td>
<td>Trivial variables used in transonic drag calc</td>
</tr>
<tr>
<td>aM</td>
<td>Vptr</td>
<td>m/s</td>
<td>Velocity (a*Mach)</td>
</tr>
<tr>
<td>Cₚ</td>
<td>.D</td>
<td>---</td>
<td>Rocket drag coefficient</td>
</tr>
<tr>
<td>Cₚₙ</td>
<td>Cdb</td>
<td>---</td>
<td>Base drag coefficient</td>
</tr>
<tr>
<td>Cₚₚ</td>
<td>Cdf</td>
<td>---</td>
<td>Total friction and interference drag coefficient</td>
</tr>
<tr>
<td>Cₚₚ(body)</td>
<td>CdfBody</td>
<td>---</td>
<td>Body coefficient of drag due to friction</td>
</tr>
<tr>
<td>Cₚₚ(fins)</td>
<td>CdfFins</td>
<td>---</td>
<td>Coefficient of friction drag for all fins</td>
</tr>
<tr>
<td>Cₚₚ</td>
<td>CdPro</td>
<td>---</td>
<td>Drag coefficient of protuberance due to friction</td>
</tr>
<tr>
<td>Cf</td>
<td>CF</td>
<td>---</td>
<td>Compressible skin friction coefficient</td>
</tr>
<tr>
<td>Cfₖ</td>
<td>CfLambda</td>
<td>---</td>
<td>Average flat plate skin friction coefficient for each fin panel</td>
</tr>
<tr>
<td>Cf(final)</td>
<td>CfFinal</td>
<td>---</td>
<td>Greater of Cf*(term) and Cf(term)</td>
</tr>
<tr>
<td>Cf(term)</td>
<td>CfTerm</td>
<td>---</td>
<td>Compressible skin friction coefficient with roughness</td>
</tr>
<tr>
<td>Cf*</td>
<td>CfStar</td>
<td>---</td>
<td>Incompressible skin friction coefficient</td>
</tr>
<tr>
<td>Cf*(term)</td>
<td>CfStarTerm</td>
<td>---</td>
<td>Incompressible skin friction coefficient with roughness</td>
</tr>
<tr>
<td>Cₚₚ</td>
<td>CfPro</td>
<td>---</td>
<td>Friction coefficient of protuberance</td>
</tr>
<tr>
<td>Cₚₚ</td>
<td>.FinRootChord</td>
<td>m</td>
<td>Fin root chord</td>
</tr>
<tr>
<td>Cₚₚ</td>
<td>.FinTipChord</td>
<td>m</td>
<td>Fin tip chord</td>
</tr>
<tr>
<td>d</td>
<td>.D</td>
<td>m</td>
<td>Maximum diameter of rocket</td>
</tr>
<tr>
<td>dₙ</td>
<td>.dB, .dnC</td>
<td>m</td>
<td>Diameter at base of rocket (burn and coast)</td>
</tr>
<tr>
<td>fₖ</td>
<td>fb</td>
<td>---</td>
<td>Base drag function for Mach &gt; 0.6</td>
</tr>
<tr>
<td>K</td>
<td>.Kb, .FinK, .Kp</td>
<td>m</td>
<td>Surface roughness</td>
</tr>
<tr>
<td>Kₙ</td>
<td>Kbase</td>
<td>---</td>
<td>Constant of proportionality</td>
</tr>
<tr>
<td>Kₑ</td>
<td>Ke</td>
<td>---</td>
<td>Coefficient for excrescencies drag incurment</td>
</tr>
<tr>
<td>L</td>
<td>.L</td>
<td>m</td>
<td>Total body length</td>
</tr>
<tr>
<td>Reference Variable</td>
<td>Code Variable</td>
<td>Units</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------</td>
<td>-------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>M</td>
<td>Mno</td>
<td>---</td>
<td>Mach number</td>
</tr>
<tr>
<td>M_D</td>
<td>Md</td>
<td>---</td>
<td>Transonic divergence Mach number</td>
</tr>
<tr>
<td>M_F</td>
<td>Mf</td>
<td>---</td>
<td>Final Mach number of transonic region</td>
</tr>
<tr>
<td>n</td>
<td>n</td>
<td>---</td>
<td>Exponent</td>
</tr>
<tr>
<td>n_t</td>
<td>.nFin</td>
<td>---</td>
<td>Number of fins</td>
</tr>
<tr>
<td>R_n</td>
<td>Rn</td>
<td>---</td>
<td>Incompressible Reynolds number</td>
</tr>
<tr>
<td>R_n*</td>
<td>RnStar</td>
<td>---</td>
<td>Compressible Reynolds number</td>
</tr>
<tr>
<td>S_b</td>
<td>.Sb</td>
<td>m²</td>
<td>Surface area of body</td>
</tr>
<tr>
<td>S_f</td>
<td>.FinSurfaceArea</td>
<td>m²</td>
<td>Total wetted surface area of each fin</td>
</tr>
<tr>
<td>S_pro</td>
<td>.Spro</td>
<td>m²</td>
<td>Surface area of protuberance</td>
</tr>
<tr>
<td>S_r</td>
<td>Not used</td>
<td>m²</td>
<td>Total surface area of rocket</td>
</tr>
<tr>
<td>t</td>
<td>.FinMaxThickness</td>
<td>m</td>
<td>Maximum thickness of fin at root</td>
</tr>
<tr>
<td>X_t/c</td>
<td></td>
<td>---</td>
<td>X_t/c / Cr</td>
</tr>
<tr>
<td>X_c/c</td>
<td>.FinXtc</td>
<td>m</td>
<td>Distance from fin leading edge to maximum thickness</td>
</tr>
</tbody>
</table>

**Body Friction Drag**

\[ R_n^* = \frac{aML}{\nu} \left( 1 + 0.0283M - 0.043M^2 + 0.2107M^3 - 0.03829M^4 + 0.002709M^5 \right) \]

\[ C_f^* = 0.037036R_n^{*-0.155079} \]

\[ C_f = C_f^* \left( 1 + 0.00798M - 0.1813M^2 + 0.0632M^3 - 0.00933M^4 + 0.000549M^5 \right) \]

\[ C_f^* (\text{term}) = \frac{1}{0.142 + 1.62 \log_{10} \left( \frac{L}{K} \right)}^{2.5} \]

\[ C_f (\text{term}) = \frac{C_f^* (\text{term})}{1 + 0.2044M^2} \]

\[ C_f (\text{final}) = \max(C_f, C_f (\text{term})) \]

\[ C_d_f (\text{body}) = C_f (\text{final}) \left[ 1 + \frac{60}{(L/d)^3} + 0.0025(L/d) \right] \cdot \frac{4S_b}{\pi d^2} \]
Fin Friction Drag

\[ Rn^* = \frac{aMC_r}{\nu} \left( 1 + 0.0283M - 0.043M^2 + 0.2107M^3 - 0.03829M^4 + 0.002709M^5 \right) \]

\[ Cf^* = 0.037036 Rn^*^{-0.155079} \]

\[ Cf = Cf^* \left( 1 + 0.00798M - 0.1813M^2 + 0.0632M^3 - 0.00933M^4 + 0.000549M^5 \right) \]

\[ Cf^* (\text{term}) = \frac{1}{0.142 + 1.62 \log_{10} \left( \frac{L}{K} \right) ^{2.5}} \]

\[ Cf (\text{term}) = \frac{Cf^* (\text{term})}{1 + 0.2044M^2} \]

\[ Cf (\text{final}) = \max(Cf, Cf (10\text{term})) \]

\[ Rn = \frac{aMC_r}{\nu} \]

\[ \lambda = \frac{C_r}{C_r} \]

\[ Cf_\lambda = Cf (\text{final}) \left[ 1 + \frac{0.5646}{\log_{10} (Rn)} \right], \text{if } \lambda = 0.0 \]

or

\[ Cf_\lambda = Cf (\text{final}) \left[ \frac{\log_{10} (Rn)}{\lambda^2 - 1} \right]^{2.6} \left[ \frac{\lambda^2}{[\log_{10} (Rn\lambda)]^{2.6}} - \frac{1}{[\log_{10} (Rn)]^{2.6}} \right] + 0.5646 \left[ \frac{\lambda^2}{[\log_{10} (Rn\lambda)]^{3.6}} - \frac{1}{[\log_{10} (Rn)]^{3.6}} \right] \]

\[ Cd_\lambda (\text{fins}) = Cf (\text{final}) \left[ 1 + \frac{60}{(L/d)^3} + 0.0025(L/d) \right] \frac{4S_b}{\pi d^2} \]
Protuberance Friction Drag

(Protuberances are components that are found on the exterior of the rocket – launch lugs, etc.)

\[
Rn^* = \frac{aML_p}{v} \left(1 + 0.0283M - 0.043M^2 + 0.2107M^3 - 0.03829M^4 + 0.002709M^5 \right)
\]

\[
Cf^* = 0.037036Rn^*^{-0.155079}
\]

\[
Cf = Cf^* \left(1 + 0.00798M - 0.1813M^2 + 0.0632M^3 - 0.00933M^4 + 0.000549M^5 \right)
\]

\[
Cf^* (\text{term}) = \frac{1}{0.142 + 1.62 \log_{10} \left( \frac{L_p}{K} \right)}^{2.5}
\]

\[
Cf (\text{term}) = \frac{Cf^* (\text{term})}{1 + 0.2044M^2}
\]

\[
Cf (\text{final}) = \max(Cf, Cf (\text{term}))
\]

\[
Cf_{pro} = 0.8151Cf (\text{final}) \left( \frac{a}{L_p} \right)^{-0.1243}
\]

\[
Cd_{pro} = Cf_{pro} \left[1 + 1.798 \left( \frac{\sqrt{A}}{L_p} \right)^{3/2} \right] \frac{4S_{pro}}{\pi d^2}
\]

Drag due to Excrescencies

(Scratches, gouges, rivets, joints, etc.)

\[
Cd_e = K_e \frac{4S_e}{\lambda d^2}
\]

where,

\[
K_e = 0.00038, \text{ for } M < 0.78
\]

\[
K_e = -0.4501M^4 + 1.5954M^3 - 2.1062M^2 + 1.2288M - 0.26717, \text{ for } 0.78 \leq M \leq 1.04
\]

\[
K_e = 0.0002M^2 - 0.0012M + 0.0018, \text{ for } M > 1.04
\]

Total Friction and Interference Drag

\[
Cd_f = Cd_f (body) + 1.04Cd_f (fins) + 1.04Cd_{pro} + Cd_e
\]
Base Drag – $M < 0.6$

$$Cd_b = K_b \left( \frac{d_b}{d} \right)^n \sqrt{Cd_f}$$

$$K_b = 0.0274 \tan^{-1} \left( \left( \frac{L_{o}}{d} \right) + 0.0116 \right)$$

$$n = 3.6542 \left( \frac{L_{v}}{d} \right)^{-0.2733}$$

Base Drag – $M > 0.6$

$$Cd_b = KCd_b(M = 0.6)f_{b}$$

Where,

$$f_b = 1.0 + 215.8(M - 0.6), \text{ for } 0.6 < M < 1.0$$

$$f_b = 2.0881(M - 1)^3 - 3.7938(M - 1)^2 + 1.4618(M - 1) + 1.883917, \text{ for } 1.0 < M < 2.0$$

$$f_b = 0.297(M - 2)^3 - 0.7937(M - 2)^2 - 0.1115(M - 2) + 1.64006, \text{ for } M > 2.0$$

Transonic Wave Drag Coefficient

$$M_D = -0.0156 \left( \frac{L_N}{d} \right)^2 + 0.136 \left( \frac{L_N}{d} \right) + 0.6817$$

$$M_F = a \left( \frac{L_N}{d} \right)^b + 1.0275$$

Where,

$$a = 2.4, \text{ for } \left( \frac{L_N}{L_c} \right) < 0.2$$

'DragModule - Drag calculations'

Public Type DragType
    CdTotal As Single
    CdBody As Single
    CdPro As Single
    CdExc As Single
    CdFric As Single
    CdFin As Single
    CdBase As Single
    CdTS As Single
    D As Single
End Type
Public Function Drag(Vptr As Single, Alt As Single, AF As AirframeType, Burn As Boolean) As DragType

Dim Mno As Single
Dim Cdf As Single           'drag coefficient for body
Dim Rn As Single            'incompressible Reynolds number
Dim RnStar As Single        'compressible Reynolds number
Dim CfStar As Single        'incompressible skin friction coefficient
Dim CF As Single            'compressible skin friction coefficient
Dim CfStarTerm As Single    'incompressible skin friction coefficient with roughness
Dim CTerm As Single         'compressible skin friction coefficient with roughness
Dim CFinal As Single        'final skin friction coefficient
Dim CdfBody As Single       'body coefficient of drag due to friction
Dim CfLambda As Single      'average flat plate skin friction coefficient for each fin panel
Dim CdfFins As Single       'coefficient of friction drag for all fins
Dim Cfpro As Single         'friction coefficient of protuberance
Dim CdPro As Single         'drag coefficient of protuberance due to friction
Dim Cde As Single           'drag coefficient due to excrescencies
Dim Ke As Single            'coefficient for excrescencies drag increment
Dim Cdb As Single           'coefficient for base drag
Dim Kbase As Single         'constant of proportionality
Dim n As Single             'exponent
Dim fb As Single            'transonic base drag function
Dim Md As Single            'transonic drag divergence mach number
Dim Mf As Single            'final mach number of transonic region
Dim Lne As Single           '=af.Ln/af.Le
Dim Lnbb As Single          '=af.Ln/af.Lb
Dim a As Single
Dim b As Single
Dim C As Single
Dim g As Single
Dim delCdmax As Single      'max drag rise over transonic region
Dim delCdt As Single        'transonic wave drag coefficient
Dim delDts As Single        'supersonic wave drag coefficient
Dim Xts As Single
Dim Fts As Single
Dim V As Single
Dim dbUse As Single

If Burn = True Then
    dbUse = AF.dbB
Else
    dbUse = AF.dbC
End If

V = Abs(Vptr)           'Positive value of velocity
If V = 0 Then
    Drag.CdBody = 0
    Drag.CdFin = 0
    Drag.CdPro = 0
    Drag.CdExc = 0
    Drag.Cdf = 0
    Drag.Cdbase = 0
    Drag.CdT = 0
    Drag.Cdtotal = 0
    Drag.D = 0
Else
    Mno = V / StdAt(Alt).a
Body Cd Calculations
\[ R_n = V \times \frac{AF.L}{StdAt(Alt).KVis} \]
\[ R_n^{\text{Star}} = R_n \times (1 + 0.0283 \times M_{no} - 0.043 \times M_{no}^2 + 0.2107 \times M_{no}^3 - 0.03829 \times M_{no}^4 + 0.002709 \times M_{no}^5) \]
\[ \text{Cf}^{\text{Star}} = 0.037036 \times R_n^{\text{Star}}^{-0.155079} \]
\[ \text{CfStarTerm} = 1 / (0.142 + 1.62 \times \log_{10}(AF.L / AF.Kb))^{2.5} \]
If \( \text{Cf} > \text{CfTerm} \) Then
\[ \text{CfFinal} = \text{Cf} \]
Else
\[ \text{CfFinal} = \text{CfTerm} \]
End If
\[ \text{CdfBody} = \text{CfFinal} \times (1 + 60 / (AF.L / AF.D)^3 + 0.0025 \times (AF.L / AF.D)) \times 4 \times AF.Sb / (\Pi \times AF.D^2) \]

Fin Cd Calculations
\[ R_n = V \times \frac{AF.FinRootChord}{StdAt(Alt).KVis} \]
\[ R_n^{\text{Star}} = R_n \times (1 + 0.0283 \times M_{no} - 0.043 \times M_{no}^2 + 0.2107 \times M_{no}^3 - 0.03829 \times M_{no}^4 + 0.002709 \times M_{no}^5) \]
\[ \text{Cf}^{\text{Star}} = 0.037036 \times R_n^{\text{Star}}^{-0.155079} \]
\[ \text{CfStarTerm} = 1 / (0.142 + 1.62 \times \log_{10}(AF.FinRootChord / AF.FinKb))^{2.5} \]
If \( \text{Cf} > \text{CfTerm} \) Then
\[ \text{CfFinal} = \text{Cf} \]
Else
\[ \text{CfFinal} = \text{CfTerm} \]
End If
\[ \text{CdfFins} = \text{CfFinal} \times (1 + 60 \times (AF.FinMaxThickness / AF.FinRootChord)^4 + 0.8 \times (1 + 5 \times (AF.Xtc / AF.FinRootChord)^2) \times (AF.FinMaxThickness / AF.FinRootChord)) \times 4 \times AF.nFin \times AF.FinSurfaceArea / (\Pi \times AF.D^2) \]

Protuberance Cd Calculations
\[ R_n = V \times \frac{AF.Lp}{StdAt(Alt).KVis} \]
\[ R_n^{\text{Star}} = R_n \times (1 + 0.0283 \times M_{no} - 0.043 \times M_{no}^2 + 0.2107 \times M_{no}^3 - 0.03829 \times M_{no}^4 + 0.002709 \times M_{no}^5) \]
\[ \text{Cf}^{\text{Star}} = 0.037036 \times R_n^{\text{Star}}^{-0.155079} \]
\[ \text{CfStarTerm} = 1 / (0.142 + 1.62 \times \log_{10}(AF.Lp / AF.Kp))^{2.5} \]
If \( \text{Cf} > \text{CfTerm} \) Then
\[ \text{CfFinal} = \text{Cf} \]
Else
\[ \text{CfFinal} = \text{CfTerm} \]
End If
\[ \text{Cfpro} = 0.8151 \times \text{CfFinal} \times (AF.Ap / AF.Lp)^{-0.1243} \]
\[ \text{CdPro} = \text{Cfpro} \times (1 + 1.798 \times (AF.Xpro^0.5 / AF.Lp)^1.5) \times 4 \times AF.Spro / (\Pi \times AF.D^2) \]

Drag due to Excrescences
If \( M_{no} < 0.78 \) Then
\[ K_e = 0.00038 \]
ElseIf \( M_{no} < 1.04 \) Then
\[ K_e = -0.4501 \times M_{no}^4 + 1.5954 \times M_{no}^3 - 2.1062 \times M_{no}^2 + 1.2288 \times M_{no} - 0.26717 \]
Else
\[ K_e = 0.0002 \times M_{no}^2 - 0.0012 \times M_{no} + 0.0018 \]
End If
\[ \text{Cde} = K_e \times 4 \times (AF.Sb + AF.FinSurfaceArea + AF.Spro) / (\Pi \times AF.D^2) \]

Total friction and interference drag coefficient
\[ \text{Cdf} = \text{CdfBody} + 1.04 \times \text{CdfFins} + 1.04 \times \text{CdPro} + \text{Cde} \]
'base drag coefficient for Mach < 0.6
Kbase = 0.0274 * Atan(AF.Lo / AF.D) + 0.0116
n = 3.6542 * (AF.Lo / AF.D) ^ -0.2733
Cdb = Kbase * (dbUse / AF.D) ^ n / (Cdf) ^ 0.5

'base drag for Mach > 0.6
If Mno < 0.6 Then
    fb = 1#
ElseIf Mno < 1# Then
    fb = 1 + 215.8 * (Mno - 0.6) ^ 6#
ElseIf Mno < 2# Then
    fb = 2.0881 * (Mno - 1) ^ 3 - 3.7938 * (Mno - 1) ^ 2 + 1.4618 * (Mno - 1) + 1.883917
Else
    fb = 0.297 * (Mno - 2) ^ 3 - 0.7937 * (Mno - 2) ^ 2 - 0.1115 * (Mno - 2) + 1.64006
End If
Cdb = fb * Cdb

'transonic wave drag coefficient
Md = -0.0156 * (AF.Ln / AF.D) ^ 2 + 0.136 * (AF.Ln / AF.D) + 0.6817
Lne = AF.Ln / AF.Le
If Lne < 0.2 Then
    a = 2.4
    b = -1.05
Else
    a = -321.94 * (Lne) ^ 2 + 264.07 * (Lne) - 36.348
    b = 19.634 * (Lne) ^ 2 - 18.369 * (Lne) + 1.7434
End If
Mf = a * (AF.Le / AF.D) ^ b + 1.0275
Lnb = AF.Ln / AF.Lb
C = 50.676 * (Lnb) ^ 2 - 51.734 * (Lnb) + 15.642
g = -2.2538 * (Lnb) ^ 2 + 1.3108 * (Lnb) - 1.7344
If AF.Le / AF.D >= 6# Then
    delCdmax = C * (AF.Le / AF.D) ^ g
Else
    delCdmax = C * (6) ^ g
End If
Xts = ((Mno - Md) / (Mf - Md))
Fts = -8.3474 * Xts ^ 5 + 24.543 * Xts ^ 4 - 24.946 * Xts ^ 3 + 8.6321 * Xts ^ 2 + 1.1195 * Xts
If Mno < Md Then
    delCdt = 0
    delCds = 0
ElseIf Mno < Mf Then
    delCdt = delCdmax * Fts
    delCds = 0
Else
    delCdt = 0
    delCds = delCdmax
End If
Drag.CdBody = CdBody
Drag.CdFin = CdFins * 1.04
Drag.CdPro = CdPro * 1.04
Drag.CdExc = Cde
Drag.CdFric = CdF
Drag.CdBase = Cdb
Drag.CdTS = delCdt + delCds
Drag.CdTotal = CdF + Cdb + delCdt + delCds
Drag.D = (0.5 * StdAt(Alt).Dens * V ^ 2) * Drag.CdTotal * (Pi * AF.D ^ 2 / 4)
End If
End Function
Appendix 3 – Stability Calculations

Center of Pressure Calculation

![Fin Geometry for Stability Calculation](image)

Figure 27 – Fin Geometry for Stability Calculation

Table 5 – Variables used in Stability Analysis

<table>
<thead>
<tr>
<th>Reference Variable</th>
<th>Code Variable</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>((C_{Na})_f)</td>
<td>FinCnAlpha</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>((C_{Na})_N)</td>
<td>NoseCnAlpha</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>(C_r)</td>
<td>FinRootChord</td>
<td>m</td>
<td>Root chord</td>
</tr>
<tr>
<td>(C_t)</td>
<td>FinTipChord</td>
<td>m</td>
<td>Tip chord</td>
</tr>
<tr>
<td>(d)</td>
<td>(d)</td>
<td>m</td>
<td>Body diameter</td>
</tr>
<tr>
<td>(L)</td>
<td>(Ln)</td>
<td>m</td>
<td>Length of rocket nose</td>
</tr>
<tr>
<td>(l)</td>
<td>FinMidChordLength</td>
<td>m</td>
<td>Length of fin at mid chord</td>
</tr>
<tr>
<td>(r^*)</td>
<td>Not used</td>
<td>m</td>
<td>Body radius at fin (d/2)</td>
</tr>
<tr>
<td>(s)</td>
<td>FinSpan</td>
<td>m</td>
<td>Fin span</td>
</tr>
<tr>
<td>(X_{bar_f})</td>
<td>FinXbar</td>
<td>m</td>
<td>Fin center of pressure – relative to fin root L.E.</td>
</tr>
<tr>
<td>(X_{bar_N})</td>
<td>NoseXbar</td>
<td>m</td>
<td>Nose center of pressure</td>
</tr>
<tr>
<td>(X_f)</td>
<td>(X_f)</td>
<td>m</td>
<td>Distance from nose to fin root L.E.</td>
</tr>
<tr>
<td>(X_t)</td>
<td>FinTipX</td>
<td>m</td>
<td>Distance from fin root L.E. to tip L.E.</td>
</tr>
</tbody>
</table>
\[ (C_{Na})_N = 2 \]
\[ X_N = 0.466 \times L \]
\[ (C_{Na})_f = \frac{16\left(\frac{s}{d}\right)^2}{1 + \sqrt{1 + \left(\frac{2l}{C_r + C_t}\right)^2}} \]
\[ X_F = \frac{X_t \left(\frac{C_r + 2C_t}{C_r + C_t}\right) + \frac{1}{6} \left[C_r + C_t - \frac{C_r C_t}{C_r + C_t}\right]}{\left[-\frac{2l}{C_r + C_t}\right]} \]
\[ Kt(\beta) = 1 + \frac{r^*}{s + r^*} \]
\[ X = \frac{(C_{Na})_N \times X_N + (C_{Na})_f \times Kt(\beta) \times (X_F + X_F)}{(C_{Na})_N + (C_{Na})_f \times Kt(\beta)} \]

Center of Gravity Calculation
\[ X_{cg} = \frac{\sum X_i M_i}{\sum M_i} \]

'COP_CG Module - Center of pressure and center of gravity calculations'

Dim X(10) As Single
Dim Mass(10) As Single
Dim MassX(10) As Single
Dim ItemName(10) As String

'sub CalcXopXcg - calls CalcXCOP (center of pressure) and CalcXCG (center of gravity subroutines)
' and writes output to airframe sheet.

Public Sub CalcXcopXcg(AF As AirframeType, Motor As MotorCurveType)

Call CalcXCOP(AF)
Call CalcXCG(AF, Motor)
AF.CaliberL = (AF.XCOP - AF.XCGL) / AF.D
AF.CaliberB = (AF.XCOP - AF.XCGB) / AF.D
Sheets(AirframeSheet).Select
R = 38
Cells(R + 1, 2).Value = "Empty"
Cells(R + 1, 3).Value = "Launch"
Cells(R + 1, 4).Value = "Burnout"
Cells(R + 2, 1).Value = "Mass ="
Cells(R + 2, 2).Value = AF.MStructural
Cells(R + 2, 3).Value = AF.MStructural + Motor.TotalMass
Cells(R + 3, 1).Value = "COP ="
Cells(R + 3, 2).Value = AF.XCOP
Cells(R + 4, 1).Value = "CG ="

37
Public Sub CalcXCOP(AF As AirframeType)

    Dim NoseXbar As Single
    Dim NoseCnAlpha As Single
    Dim FinXbar As Single
    Dim FinCnAlpha As Single
    Dim FinMidChordLength As Single
    Dim Ktb As Single
    Dim XCOP As Single

    With AF

        'Ogive Nose
        NoseCnAlpha = 2
        NoseXbar = 0.466 * .Ln

        'Fins
        FinMidChordLength = Sqr(.FinSpan ^ 2 + (.FinRootChord / 2 - .FinTipChord / 2) ^ 2)
        FinCnAlpha = 4 * _nFin * (.FinSpan / .D) ^ 2 / (1 + Sqr((2 * FinMidChordLength / (.FinRootChord + .FinTipChord)) ^ 2))
        FinXbar = FinXTip / 3 * (.FinRootChord + 2 * .FinTipChord) / (.FinRootChord + .FinTipChord) + 1 / 6 * (.FinRootChord + .FinTipChord - (.FinRootChord * .FinTipChord) / (.FinRootChord + .FinTipChord))
        FinXbar = FinXbar + .FinXBase
        Ktb = 1 + AF.D / 2 / (AF.FinSpan + AF.D / 2)
        FinCnAlpha = FinCnAlpha * Ktb
        AF.XCOP = (NoseCnAlpha * NoseXbar + FinCnAlpha * FinXbar) / (NoseCnAlpha + FinCnAlpha)
    End With

End Sub

Public Sub CalcXCG(AF As AirframeType, Motor As MotorCurveType)

    Dim VolPerFin As Single
    Dim SumMass As Single
    Dim SumMassX As Single

    With AF

        'nose - scale by nose length and dia

ItemName(1) = "Nose"
X(1) = .Ln * 0.75
Mass(1) = 0.112 * (.Ln / 0.241) * (.D / 0.057)
MassX(1) = X(1) * Mass(1)
SumMass = Mass(1)
SumMassX = MassX(1)
'tube - scale by tube length and dia
ItemName(2) = "Tube"
If .Power = "MP" Then
   X(2) = (.Ln + .L) / 2
   Mass(2) = 0.226 * (.D / 0.0762) * (.L - .Ln) / 0.864
Else
   X(2) = (.Ln + .L) / 2
   Mass(2) = 0.2 * (.D / 0.057) * (.L - .Ln) / 0.919
End If
MassX(2) = X(2) * Mass(2)
SumMass = SumMass + Mass(2)
SumMassX = SumMassX + MassX(2)
'fins
ItemName(3) = "Fins"
       .FinTipChord * .FinRootChord + .FinRootChord ^ 2) / (3 * (.FinTipChord + .FinRootChord))
X(3) = .FinXBase + X(3)
VolPerFin = (.FinTipChord + .FinRootChord) / 2 * .FinSpan * .FinMaxThickness / 0.00163
Mass(3) = VolPerFin * .nFin * 1.626 * 1.8
MassX(3) = X(3) * Mass(3)
SumMass = SumMass + Mass(3)
SumMassX = SumMassX + MassX(3)
'Recovery - based on SkyAngle classic
ItemName(4) = "Recovery"
X(4) = .Ln + .Lpay + 0.25 / 2
Mass(4) = 0.083 + 0.027 * (.MStructural + Motor.TotalMass - Motor.FuelMass)
MassX(4) = X(4) * Mass(4)
SumMass = SumMass + Mass(4)
SumMassX = SumMassX + MassX(4)
'alimiter
ItemName(5) = "Alimiter"
X(5) = .Ln + .Lpay / 2
Mass(5) = 0.096
MassX(5) = X(5) * Mass(5)
SumMass = SumMass + Mass(5)
SumMassX = SumMassX + MassX(5)
'motor mount
ItemName(6) = "Motor Mount"
X(6) = .L - Motor.Length / 2
Mass(6) = SumMass * 0.1
MassX(6) = X(6) * Mass(6)
SumMass = SumMass + Mass(6)
SumMassX = SumMassX + MassX(6)
'misc
ItemName(7) = "Misc"
X(7) = AF.L / 2
Mass(7) = SumMass * 0.1
MassX(7) = X(7) * Mass(7)
SumMass = SumMass + Mass(7)
SumMassX = SumMassX + MassX(7)
AF.MStructural = SumMass
'motor - launch
ItemName(8) = "Motor - Launch"
X(8) = .L - Motor.Length / 2 + 0.01
Mass(8) = Motor.TotalMass
MassX(8) = X(8) * Mass(8)
SumMass = SumMass + Mass(8)
SumMassX = SumMassX + MassX(8)
AF.XCGL = SumMassX / SumMass
'motor - Burnout
  ItemName(9) = "Motor - Burnout"
  X(9) = .L - Motor.Length / 2 + 0.01
  MassX(9) = X(9) * Mass(9)
  SumMass = SumMass + Mass(9)
  SumMassX = SumMassX + MassX(9)
  XCGB = SumMassX / SumMass
End With
End Sub
Appendix 4 – Flight Simulation

Equations of motion:

\[ J = \text{cons} \tan t \]
\[ a_i = a_{(i-1)} + J \times dT \]
\[ J = \frac{a_i - a_{(i-1)}}{dT} \]
\[ V_i = V_{(i-1)} + a_{(i-1)} \times dT + \frac{1}{2} \times J \times dT^2 \]
\[ Alt_i = Alt_{(i-1)} + V_{(i-1)} \times dT + \frac{1}{2} \times a_{(i-1)} \times dT^2 + \frac{1}{6} \times J \times dT^3 \]

Figure 28 – Altitude, Velocity, Acceleration, and Jerk vs. Time

Flight Simulation Module

Const SIM_TIME = 60 'Total time of simulation
Const nINC = 1000
Type AirframeType
    Kb As Single 'surface finish roughness
    L As Single 'length
    Lo As Single
    Ln As Single
    Le As Single
    Lb As Single
    D As Single 'maximum diameter
    Sb As Single 'total wetted surface area of body
    dBb As Single
    dBC As Single
    nFin As Integer 'number of fins
    FinK As Single 'surface finish
    FinTipChord As Single 'fin tip chord
    FinRootChord As Single 'fin root chord
    FinXtc As Single 'distance from LE to max thickness
FinSpan As Single     'fin span
FinMaxThickness As Single    'max thickness of fin at root
Lambda As Single           'tip chord/root chord
FinXbar As Single           'FinXtc/Cr
FinSurfaceArea As Single    'wetted surface of a single fin
FinXTip As Single           'mid chord sweep angle - deg
FinXBase As Single          'tip to L.E. base
Lp As Single                'length of protuberance
Ap As Single                'distance from nose to protuberance L.E.
Kp As Single                'surface finish
Xpro As Single              'cross sectional area of protuberance
Spro As Single              'surface area
MStructural As Single
Mtotal As Single
XCGL As Single              'center of gravity - launch
XCGB As Single              'center of gravity - burnout
XCOP As Single              'center of pressure
CaliberL As Single          'stability in caliber
CaliberB As Single
Lpay As Single
Power As String
End Type
Type LaunchType
  Time As Single
  Alt As Single
  Vel As Single
  Accel As Single
  Drag As Single
  Thrust As Single
End Type

Dim Launch(10000) As LaunchType
Public WhatToDo As String
Public MotorSheet As String
Public AirframeSheet As String
Public Airframe As AirframeType
Dim AirframeCtr As Integer
'
Sub main()
  RunSelectForm
  Select Case WhatToDo
    Case "Cd"
      ReadRocket
      CalcCdvsMno
    Case "COPCG"
      ReadRocket
      ReadMotor
      Call CalcXcopXcg(Airframe, Motor)
    Case "Hybrid"
      ReadMotor
      'motorsim
    Case "Rocket"
      ReadRocket
      ReadMotor
      RunLaunch
  End Select
End Select
End Sub

Sub RunSelectForm()
  Dim SheetCtr As Integer
  Dim SheetType(50) As String
  'Dim Motors(50) As String
  'Dim Motors(50) As String
Dim MotorCtr As Integer

MotorCtr = 0
AirframeCtr = 0
For SheetCtr = 1 To Sheets.Count
    Sheets(SheetCtr).Select
    Range("A1").Select  'eliminates problem when sheet left with object selected
    SheetType(SheetCtr) = Cells(1, 1).Value
    If SheetType(SheetCtr) = "MOTOR" Then
        MotorCtr = MotorCtr + 1
        'Motors(MotorCtr) = Sheets.Application.ActiveSheet.Name
        SimControlForm.MotorSelectBox.AddItem Sheets.Application.ActiveSheet.Name
    ElseIf SheetType(SheetCtr) = "AIRFRAME" Then
        AirframeCtr = AirframeCtr + 1
        SimControlForm.AirframeSelectBox.AddItem Sheets.Application.ActiveSheet.Name
    End If
Next SheetCtr
X = 1
Sheets("Simulation").Select
SimControlForm.Show
X = MotorSheet
y = SimControlForm.AirframeSelectBox.Value
z = 1
End Sub

Sub ReadMotor()
Const MotorRowOffset = 5
' read motor data
    Sheets(MotorSheet).Select
    If Cells(1, 2).Value = "CURVE" Then
        Motor.Name = Cells(2, 1).Value
        Motor.TotalMass = Cells(5, 6).Value
        Motor.Length = Cells(5, 8).Value
        Motor.ImpTime(0) = 0
        Motor.Impulse(0) = 0
        For i = 1 To 50
            Motor.ImpTime(i) = Cells(i + MotorRowOffset, 1).Value
            Motor.Impulse(i) = Cells(i + MotorRowOffset, 2).Value
            If i > 1 And Motor.ImpTime(i) = 0 Then
                Motor.BurnTime = Motor.ImpTime(i - 1)
                Exit For
            End If
        Next i
        MotorSetup
    ElseIf Cells(1, 2).Value = "SIM" Then
        'read
        Else
        'error
    End If
End Sub

Sub ReadRocket()
' read airframe data
    Sheets(AirframeSheet).Select
    With Airframe
        .Kb = Cells(5, 6).Value
        .L = Cells(6, 6).Value
        .Lo = Cells(7, 6).Value
        .Ln = Cells(8, 6).Value
        .Le = Cells(9, 6).Value
        .Lb = Cells(10, 6).Value
    End With
End Sub
.D = Cells(11, 6).Value
.Sh = Cells(12, 6).Value
.dbB = Cells(13, 6).Value
.dbC = Cells(14, 6).Value
.nFin = Cells(16, 6).Value
.FinK = Cells(17, 6).Value
.FinTipChord = Cells(18, 6).Value
.FinRootChord = Cells(19, 6).Value
.FinXtc = Cells(20, 6).Value
.FinSpan = Cells(21, 6).Value
.FinMaxThickness = Cells(22, 6).Value
.FinXTip = Cells(23, 6).Value
.FinXbar = Cells(24, 6).Value
.Lp = Cells(26, 6).Value
.Ap = Cells(27, 6).Value
.Kp = Cells(28, 6).Value
.Xpro = Cells(29, 6).Value
.Spro = Cells(30, 6).Value
.MStructural = Cells(32, 6).Value
.Lpay = Cells(35, 6).Value
.Power = Cells(36, 6).Value
.Lambda = .FinTipChord / .FinRootChord
.FinXbar = .FinXtc / .FinRootChord
.FinSurfaceArea = .FinSpan * (.FinRootChord + .FinTipChord)
End With
i = 1
End Sub

Sub CalcCdvsMno()
Dim ColOff As Integer
Dim V As Single
Dim row As Single
Dim Mach As Single
ColOff = 12
row = 2
Sheets(AirframeSheet).Select
' BURN
Cells(row, ColOff).Value = "Mach"
Cells(row, ColOff + 1).Value = "Body"
Cells(row, ColOff + 2).Value = "Fin"
Cells(row, ColOff + 3).Value = "Pro"
Cells(row, ColOff + 4).Value = "Exc"
Cells(row, ColOff + 5).Value = "Base"
Cells(row, ColOff + 6).Value = "T/S"
Cells(row, ColOff + 7).Value = "Total"
row = 3
For i = 2 To 25
Mach = i / 10
V = Mach * StdAt(0).a
Cells(row, ColOff).Value = Mach
Cells(row, ColOff + 1).Value = Drag(V, 0#, Airframe, True).CdBody
Cells(row, ColOff + 2).Value = Drag(V, 0#, Airframe, True).CdFin
Cells(row, ColOff + 3).Value = Drag(V, 0#, Airframe, True).CdPro
Cells(row, ColOff + 4).Value = Drag(V, 0#, Airframe, True).CdExc
Cells(row, ColOff + 5).Value = Drag(V, 0#, Airframe, True).CdBase
Cells(row, ColOff + 6).Value = Drag(V, 0#, Airframe, True).CdTS
Cells(row, ColOff + 7).Value = Drag(V, 0#, Airframe, True).CdTotal
row = row + 1
Next i
' COAST
row = 2
ColOff = 20
Cells(row, ColOff).Value = "Mach"
Cells(row, ColOff + 1).Value = "Body"
Cells(row, ColOff + 2).Value = "Fin"
Cells(row, ColOff + 3).Value = "Pro"
Cells(row, ColOff + 4).Value = "Exc"
Cells(row, ColOff + 5).Value = "Base"
Cells(row, ColOff + 6).Value = "T/S"
Cells(row, ColOff + 7).Value = "Total"
row = 3
For i = 2 To 25
    Mach = i / 10
    V = Mach * StdAt(0).a
    Cells(row, ColOff).Value = Mach
    Cells(row, ColOff + 1).Value = Drag(V, 0#, Airframe, False).CdBody
    Cells(row, ColOff + 2).Value = Drag(V, 0#, Airframe, False).CdFin
    Cells(row, ColOff + 3).Value = Drag(V, 0#, Airframe, False).CdPro
    Cells(row, ColOff + 4).Value = Drag(V, 0#, Airframe, False).CdExc
    Cells(row, ColOff + 5).Value = Drag(V, 0#, Airframe, False).CdBase
    Cells(row, ColOff + 6).Value = Drag(V, 0#, Airframe, False).CdTS
    Cells(row, ColOff + 7).Value = Drag(V, 0#, Airframe, False).CdTotal
    row = row + 1
Next i
End Sub

Sub RunLaunch()
    Dim dT As Single
    Dim J As Single
    Dim Weight As Single
    Dim OnPad As Boolean
    Application.ScreenUpdating = False
    Sheets("Simulation").Select
    Cells(1, 1).Value = "Time"
    Cells(1, 2).Value = "Thrust"
    Cells(1, 3).Value = "Drag"
    Cells(1, 4).Value = "Accel-g"
    Cells(1, 5).Value = "Vel"
    Cells(1, 6).Value = "Mach no."
    Cells(1, 7).Value = "Alt"
    Cells(1, 8).Value = "Mtotal"
    Erase Launch
    dT = SIM_TIME / nINC
    OnPad = True
    For inc = 1 To nINC
        Launch(inc).Time = inc * dT
        'acceleration
        Launch(inc).Thrust = Thrust(Launch(inc).Time)
        Airframe.Mtotal = Airframe.MStructural + MotorMass(Launch(inc).Time)
        Weight = Airframe.Mtotal * 9.8
        Launch(inc).Drag = Drag(Launch(inc - 1).Vel, Launch(inc - 1).Alt, Airframe, Burn(Launch(inc).Time)).D
        Launch(inc).Drag = Launch(inc).Drag * Sgn(Launch(inc - 1).Vel)
        If OnPad And Launch(inc).Accel < 0 Then Launch(inc).Accel = 0    'still on launch pad
        J = (Launch(inc).Accel - Launch(inc - 1).Accel) / dT
        Launch(inc).Vel = Launch(inc - 1).Vel + (Launch(inc - 1).Accel * dT) + (0.5 * J * dT ^ 2)
        Launch(inc).Alt = Launch(inc - 1).Alt + (Launch(inc - 1).Vel * dT) + _
            (0.5 * Launch(inc - 1).Accel * dT ^ 2) + (1 / 6 * J * dT ^ 3)
        If Launch(inc).Alt < 0 Then Exit Sub
        If Launch(inc).Alt > 0 Then OnPad = False
        If Launch(inc).Time > 17.44 Then
            s = 1
        End If
    Next inc
End Sub
Cells(inc + 1, 1).Value = Launch(inc).Time
Cells(inc + 1, 2).Value = Launch(inc).Thrust
Cells(inc + 1, 3).Value = Launch(inc).Drag
Cells(inc + 1, 4).Value = Launch(inc).Accel / 9.8
Cells(inc + 1, 5).Value = Launch(inc).Vel
Cells(inc + 1, 6).Value = Abs(Launch(inc).Vel / StdAt(Launch(inc).Alt).a)
Cells(inc + 1, 7).Value = Launch(inc).Alt
Cells(inc + 1, 8).Value = Airframe.Mtotal
Next inc
End Sub

Sub WriteOutput()
Sheets("Simulation").Select
For row = 1 To 1000
    cell(row, 1).Value = Time(row)
Next row
End Sub
Appendix 5 – Standard Atmosphere and Misc. Functions

Misc. Function Module

Option Base 1
Public Const Pi = 3.14159265358979
'
' Function LinInterpVB performs linear interpolation on arrays in VBA
' xs and xy are (n) arrays
' xs must be monotonically increasing
'
Public Function LinInterpVB(X As Single, XTable() As Single, YTable() As Single)
Dim Lb As Integer
Dim UB As Integer
Dim Msg As String

Lb = LBound(XTable, 1)
UB = UBound(XTable, 1)
If Lb <> LBound(YTable, 1) Or UB <> UBound(YTable, 1) Then
    MsgBox "X-Table and Y-Table are not same size"
    LinInterpVB = 0
    Exit Function
End If

If XTable(Lb) > X Then
    Msg = "Extrapolating on X" + "test"
    MsgBox (Msg)
    LinInterpVB = YTable(Lb) - ((XTable(Lb) - X) / (XTable(Lb + 1) - XTable(Lb)) * 
        (YTable(Lb + 1) - YTable(Lb)))
    Exit Function
End If

For i = Lb + 1 To UB
    If XTable(i) >= X Then
        LinInterpVB = YTable(i - 1) - ((XTable(i - 1) - X) / (XTable(i) - XTable(i - 1)) * 
            (YTable(i) - YTable(i - 1)))
        Exit Function
    End If
Next i
Msg = "Extrapolating on X" + "test"
MsgBox (Msg)
i = UBound(XTable, 1)
LinInterpVB = YTable(i - 1) - ((XTable(i - 1) - X) / (XTable(i) - XTable(i - 1)) * 
      (YTable(i) - YTable(i - 1)))
End Function
'
'Log10 - returns log base 10
'
Public Function Log10(n As Single)
    Log10 = Log(n) / Log(10#)
End Function

Standard Atmosphere Module

Option Base 1
Public Type StdAtmosphereType
    Sigma As Single
    Delta As Single
    Theta As Single
    t As Single
End Type
P As Single
Dens As Single
a As Single  'velocity of sound
Vis As Single
KVis As Single
End Type
'StdAt returns standard atmosphere data given an altitude
Static Function StdAt(Altitude As Single) As StdAtmosphereType
Dim AllSet As Boolean
' Standard Atmosphere data
Dim Alt(45) As Single
Dim Sigma(45) As Single
Dim Delta(45) As Single
Dim Theta(45) As Single
Dim Temp(45) As Single
Dim Press(45) As Single
Dim Dens(45) As Single
Dim a(45) As Single
Dim Visc(45) As Single
Dim K_Visc(45) As Single
'Set up standard atmosphere data arrays
If AllSet = False Then
Alt(1) = -2000: Sigma(1) = 1.207: Delta(1) = 1.261: Theta(1) = 1.0451
Alt(2) = 0: Sigma(2) = 1#: Delta(2) = 1#: Theta(2) = 1#
Alt(3) = 2000: Sigma(3) = 0.8217: Delta(3) = 0.7846: Theta(3) = 0.9549
Alt(4) = 4000: Sigma(4) = 0.6689: Delta(4) = 0.6085: Theta(4) = 0.9098
Alt(5) = 6000: Sigma(5) = 0.5389: Delta(5) = 0.466: Theta(5) = 0.8648
Alt(6) = 8000: Sigma(6) = 0.4292: Delta(6) = 0.3519: Theta(6) = 0.8198
Alt(7) = 10000: Sigma(7) = 0.3376: Delta(7) = 0.2615: Theta(7) = 0.7748
Alt(8) = 12000: Sigma(8) = 0.2546: Delta(8) = 0.1915: Theta(8) = 0.7519
Alt(9) = 14000: Sigma(9) = 0.186: Delta(9) = 0.1399: Theta(9) = 0.7519
Alt(10) = 16000: Sigma(10) = 0.1359: Delta(10) = 0.1022: Theta(10) = 0.7519
Alt(11) = 18000: Sigma(11) = 0.0993: Delta(11) = 0.07466: Theta(11) = 0.7519
Alt(12) = 20000: Sigma(12) = 0.07258: Delta(12) = 0.05457: Theta(12) = 0.7519
Alt(13) = 22000: Sigma(13) = 0.05266: Delta(13) = 0.03995: Theta(13) = 0.7585
Alt(14) = 24000: Sigma(14) = 0.03832: Delta(14) = 0.02933: Theta(14) = 0.7654
Alt(15) = 26000: Sigma(15) = 0.02796: Delta(15) = 0.0216: Theta(15) = 0.7723
Alt(16) = 28000: Sigma(16) = 0.02047: Delta(16) = 0.01595: Theta(16) = 0.7792
Alt(17) = 30000: Sigma(17) = 0.01503: Delta(17) = 0.01181: Theta(17) = 0.7861
Alt(18) = 32000: Sigma(18) = 0.01107: Delta(18) = 0.008774: Theta(18) = 0.793
Alt(19) = 34000: Sigma(19) = 0.008071: Delta(19) = 0.006547: Theta(19) = 0.8112
Alt(20) = 36000: Sigma(20) = 0.005925: Delta(20) = 0.00492: Theta(20) = 0.8304
Alt(21) = 38000: Sigma(21) = 0.004381: Delta(21) = 0.003722: Theta(21) = 0.8496
Alt(22) = 40000: Sigma(22) = 0.003262: Delta(22) = 0.002834: Theta(22) = 0.8688
Alt(23) = 42000: Sigma(23) = 0.002445: Delta(23) = 0.002171: Theta(23) = 0.888
Alt(24) = 44000: Sigma(24) = 0.001844: Delta(24) = 0.001673: Theta(24) = 0.9072
Alt(25) = 46000: Sigma(25) = 0.001399: Delta(25) = 0.001296: Theta(25) = 0.9263
Alt(26) = 48000: Sigma(26) = 0.001075: Delta(26) = 0.00101: Theta(26) = 0.9393
Alt(27) = 50000: Sigma(27) = 0.0008382: Delta(27) = 0.0007873: Theta(27) = 0.9393
Alt(28) = 52000: Sigma(28) = 0.0006578: Delta(28) = 0.000614: Theta(28) = 0.9336
Alt(29) = 54000: Sigma(29) = 0.0005216: Delta(29) = 0.000477: Theta(29) = 0.9145
Alt(30) = 56000: Sigma(30) = 0.0004118: Delta(30) = 0.0003687: Theta(30) = 0.8954
Alt(31) = 58000: Sigma(31) = 0.0003234: Delta(31) = 0.0002834: Theta(31) = 0.8763
Alt(32) = 60000: Sigma(32) = 0.0002528: Delta(32) = 0.0002167: Theta(32) = 0.8573
Alt(33) = 62000: Sigma(33) = 0.0001965: Delta(33) = 0.0001647: Theta(33) = 0.8382
Alt(34) = 64000: Sigma(34) = 0.0001519: Delta(34) = 0.0001244: Theta(34) = 0.8191
Alt(35) = 66000: Sigma(35) = 0.0001167: Delta(35) = 0.0000935: Theta(35) = 0.8001
Alt(36) = 68000: Sigma(36) = 0.0000891: Delta(36) = 0.00006959: Theta(36) = 0.7811
Alt(37) = 70000: Sigma(37) = 0.0000676: Delta(37) = 0.00005152: Theta(37) = 0.762
Alt(38) = 72000: Sigma(38) = 0.00005091: Delta(38) = 0.00003785: Theta(38) = 0.7436
| Alt(39) | 74000: Sigma(39) = 0.00003786: Delta(39) = 0.00002764: Theta(39) = 0.73 |
| Alt(40) | 76000: Sigma(40) = 0.000028: Delta(40) = 0.00002006: Theta(40) = 0.7164 |
| Alt(41) | 78000: Sigma(41) = 0.0000206: Delta(41) = 0.00001448: Theta(41) = 0.7029 |
| Alt(42) | 80000: Sigma(42) = 0.00001506: Delta(42) = 0.00001038: Theta(42) = 0.6893 |
| Alt(43) | 82000: Sigma(43) = 0.00001095: Delta(43) = 0.0000074: Theta(43) = 0.6758 |
| Alt(44) | 84000: Sigma(44) = 0.00000578: Delta(44) = 0.000003684: Theta(44) = 0.6623 |

| Temp(1) | 301.2: Press(1) = 127800#: Dens(1) = 1.478 |
| Temp(2) | 288.1: Press(2) = 101300#: Dens(2) = 1.225 |
| Temp(3) | 275.2: Press(3) = 79500#: Dens(3) = 1.007 |
| Temp(4) | 262.2: Press(4) = 61660#: Dens(4) = 0.8193 |
| Temp(5) | 249.2: Press(5) = 47220#: Dens(5) = 0.6601 |
| Temp(6) | 236.2: Press(6) = 35650#: Dens(6) = 0.5258 |
| Temp(7) | 223.3: Press(7) = 26500#: Dens(7) = 0.4135 |
| Temp(8) | 216.6: Press(8) = 19400#: Dens(8) = 0.3119 |
| Temp(9) | 216.6: Press(9) = 14710#: Dens(9) = 0.2279 |
| Temp(10) | 216.6: Press(10) = 10350#: Dens(10) = 0.1665 |
| Temp(11) | 216.6: Press(11) = 7565#: Dens(11) = 0.1216 |
| Temp(12) | 216.6: Press(12) = 5529#: Dens(12) = 0.08891 |
| Temp(13) | 216.6: Press(13) = 4047#: Dens(13) = 0.06451 |
| Temp(14) | 216.6: Press(14) = 2972#: Dens(14) = 0.04694 |
| Temp(15) | 216.6: Press(15) = 2188#: Dens(15) = 0.03426 |
| Temp(16) | 216.6: Press(16) = 1616#: Dens(16) = 0.02508 |
| Temp(17) | 216.6: Press(17) = 1197#: Dens(17) = 0.01841 |
| Temp(18) | 216.6: Press(18) = 889#: Dens(18) = 0.01355 |
| Temp(19) | 216.6: Press(19) = 663.4: Dens(19) = 0.009887 |
| Temp(20) | 216.6: Press(20) = 498.5: Dens(20) = 0.007257 |
| Temp(21) | 216.6: Press(21) = 377.1: Dens(21) = 0.005366 |
| Temp(22) | 216.6: Press(22) = 287.1: Dens(22) = 0.003995 |
| Temp(23) | 216.6: Press(23) = 220#: Dens(23) = 0.002995 |
| Temp(24) | 216.6: Press(24) = 169.5: Dens(24) = 0.002259 |
| Temp(25) | 216.6: Press(25) = 113.3: Dens(25) = 0.001714 |
| Temp(26) | 216.6: Press(26) = 70.3: Dens(26) = 0.001317 |
| Temp(27) | 216.6: Press(27) = 39.77: Dens(27) = 0.001027 |
| Temp(28) | 216.6: Press(28) = 26.21: Dens(28) = 0.0008055 |
| Temp(29) | 216.6: Press(29) = 18.33: Dens(29) = 0.0006389 |
| Temp(30) | 216.6: Press(30) = 12.6: Dens(30) = 0.0005044 |
| Temp(31) | 216.6: Press(31) = 8.72: Dens(31) = 0.0003962 |
| Temp(32) | 216.6: Press(32) = 5.22: Dens(32) = 0.0002407 |
| Temp(33) | 216.6: Press(33) = 4.835: Dens(33) = 0.0002407 |
| Temp(34) | 216.6: Press(34) = 4.835: Dens(34) = 0.000186 |
| Temp(35) | 216.6: Press(35) = 4.835: Dens(35) = 0.0001429 |
| Temp(36) | 216.6: Press(36) = 4.835: Dens(36) = 0.0001429 |
| Temp(37) | 216.6: Press(37) = 4.835: Dens(37) = 0.0001429 |
| Temp(38) | 216.6: Press(38) = 4.835: Dens(38) = 0.0001429 |
| Temp(39) | 216.6: Press(39) = 4.835: Dens(39) = 0.0001429 |
| Temp(40) | 216.6: Press(40) = 4.835: Dens(40) = 0.0001429 |
| Temp(41) | 216.6: Press(41) = 4.835: Dens(41) = 0.0001429 |
| Temp(42) | 216.6: Press(42) = 4.835: Dens(42) = 0.0001429 |
| Temp(43) | 216.6: Press(43) = 4.835: Dens(43) = 0.0001429 |
| Temp(44) | 216.6: Press(44) = 4.835: Dens(44) = 0.0001429 |
| Temp(45) | 216.6: Press(45) = 4.835: Dens(45) = 0.0001429 |

| a(1) | 347.9: Visc(1) = 18.51: K_Visc(1) = 0.0000125 |
| a(2) | 340.3: Visc(2) = 17.89: K_Visc(2) = 0.0000146 |
| a(3) | 332.5: Visc(3) = 17.26: K_Visc(3) = 0.0000171 |
| a(4) | 324.6: Visc(4) = 16.61: K_Visc(4) = 0.0000203 |
| a(5) | 316.5: Visc(5) = 15.95: K_Visc(5) = 0.0000242 |
| a(6) | 308.1: Visc(6) = 15.27: K_Visc(6) = 0.000029 |
| a(7) | 299.5: Visc(7) = 14.58: K_Visc(7) = 0.0000353 |
| a(8) | 295.1: Visc(8) = 14.22: K_Visc(8) = 0.0000456 |
| a(9) | 295.1: Visc(9) = 14.22: K_Visc(9) = 0.0000624 |
| a(10) | 295.1: Visc(10) = 14.22: K_Visc(10) = 0.0000854 |
a(11) = 295.1: Visc(11) = 14.22: K_Visc(11) = 0.000117
a(12) = 295.1: Visc(12) = 14.22: K_Visc(12) = 0.00016
a(13) = 296.4: Visc(13) = 14.32: K_Visc(13) = 0.000222
a(14) = 297.7: Visc(14) = 14.43: K_Visc(14) = 0.000307
a(15) = 299.1: Visc(15) = 14.54: K_Visc(15) = 0.000424
a(16) = 300.4: Visc(16) = 14.65: K_Visc(16) = 0.000584
a(17) = 301.7: Visc(17) = 14.75: K_Visc(17) = 0.000801
a(18) = 303#: Visc(18) = 14.86: K_Visc(18) = 0.0011
a(19) = 306.5: Visc(19) = 15.14: K_Visc(19) = 0.00153
a(20) = 310.1: Visc(20) = 15.43: K_Visc(20) = 0.00213
a(21) = 313.7: Visc(21) = 15.72: K_Visc(21) = 0.00293
a(22) = 317.2: Visc(22) = 16.01: K_Visc(22) = 0.00401
a(23) = 320.7: Visc(23) = 16.29: K_Visc(23) = 0.00544
a(24) = 324.1: Visc(24) = 16.57: K_Visc(24) = 0.00734
a(25) = 327.5: Visc(25) = 16.85: K_Visc(25) = 0.00983
a(26) = 329.8: Visc(26) = 17.04: K_Visc(26) = 0.0129
a(27) = 329.8: Visc(27) = 17.04: K_Visc(27) = 0.0166
a(28) = 328.8: Visc(28) = 16.96: K_Visc(28) = 0.021
a(29) = 325.4: Visc(29) = 16.68: K_Visc(29) = 0.0261
a(30) = 322#: Visc(30) = 16.4: K_Visc(30) = 0.0325
a(31) = 318.6: Visc(31) = 16.12: K_Visc(31) = 0.0407
a(32) = 315.1: Visc(32) = 15.84: K_Visc(32) = 0.0511
a(33) = 311.5: Visc(33) = 15.55: K_Visc(33) = 0.0646
a(34) = 308#: Visc(34) = 15.26: K_Visc(34) = 0.082
a(35) = 304.4: Visc(35) = 14.97: K_Visc(35) = 0.105
a(36) = 300.7: Visc(36) = 14.67: K_Visc(36) = 0.134
a(37) = 297.1: Visc(37) = 14.38: K_Visc(37) = 0.174
a(38) = 293.4: Visc(38) = 14.08: K_Visc(38) = 0.226
a(39) = 290.7: Visc(39) = 13.87: K_Visc(39) = 0.299
a(40) = 288#: Visc(40) = 13.65: K_Visc(40) = 0.398
a(41) = 285.3: Visc(41) = 13.43: K_Visc(41) = 0.532
a(42) = 282.5: Visc(42) = 13.21: K_Visc(42) = 0.716
a(43) = 279.7: Visc(43) = 12.98: K_Visc(43) = 0.968
a(44) = 276.9: Visc(44) = 12.76: K_Visc(44) = 1.32
a(45) = 274.1: Visc(45) = 12.53: K_Visc(45) = 1.8
AllSet = True
End If
StdAt.Sigma = LinInterpVB(Altitude, Alt, Sigma)
StdAt.Delta = LinInterpVB(Altitude, Alt, Delta)
StdAt.Theta = LinInterpVB(Altitude, Alt, Theta)
StdAt.t = LinInterpVB(Altitude, Alt, Temp)
StdAt.P = LinInterpVB(Altitude, Alt, Press)
StdAt.Dens = LinInterpVB(Altitude, Alt, Dens)
StdAt.a = LinInterpVB(Altitude, Alt, a)
StdAt.Vis = LinInterpVB(Altitude, Alt, Visc)
StdAt.KVis = LinInterpVB(Altitude, Alt, K_Visc)
End Function