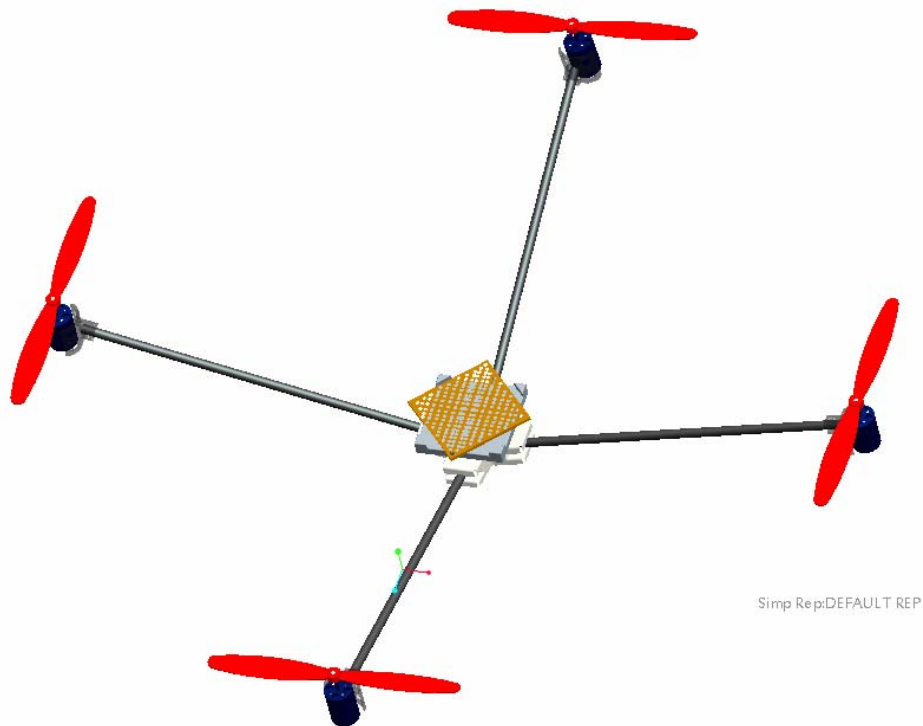


Quad-rotor Unmanned Aerial Vehicle

Final Report
May 2, 2007



GROUP #2

Carlo Canetta
Jonathan Chin
Sevan Mehrabian
Ludguier Montejo
Hendrik Thompson

Executive Summary

Unmanned aerial vehicles are aircrafts capable of flight without an on-board operator. Such vehicles can be controlled remotely by an operator on the ground, or autonomously via a pre-programmed flight path. UAVs are already being used by the military for reconnaissance and search and rescue operations.

Our project is to design and build a quad-rotor UAV. Our proposed design is for a lightweight, nimble craft that can be operated both indoors and out. The project poses particular challenges in terms of weight reduction and controllability. Quad-rotor crafts generally support only a light payload, as they are required to carry the weight of the power supply, a heavy battery, onboard. Thus, weight reduction of all components is essential in order to allow for sufficient lift force. The controls system for such a craft is also complex, as it requires the synchronization of four individual motors. These motors must be closely controlled in order to account for variations between the motors (one motor slightly more powerful than the others will upset the equilibrium of the craft), and in order to effectively dampen external disturbances.

We succeeded in building a craft structure that was both stiff and lightweight, using carbon fiber and ABS plastic. We incorporated in our vehicle a closed loop, proportional control system which measured in-flight dynamics via a single axis gyro and two dual axis tilt-sensors. A microcomputer took these sensor inputs and performed proportional control on the four motors by varying PWM signals sent to each of the motors.

The result of our project is an advanced prototype for autonomous flight. The controls software is fully operative, though it is hindered by unreliable operation of the tilt-sensors. Future work will allow refinements in the control system hardware that will ultimately lead to stable, untethered flight.

TABLE OF CONTENTS

Executive Summary	1
Introduction.....	4
Scope.....	4
Design Proposal	4
Goals	5
Parameters.....	5
Background Information.....	7
Design Details.....	9
Design Overview	9
Structure.....	9
Controls.....	12
Analysis.....	15
Dynamic Simulation	15
Finite Element Stress Analysis	22
Experiments	32
Hardware Experiments.....	32
Software Experiments.....	33
Cost Assessment	34
Design Strengths and Weaknesses.....	34
Conclusions.....	35
Future Improvements	36
References.....	37
Appendix.....	38
Project Timeline.....	A.1
Parts List	A.2
CAD Drawing Sheets.....	A.3
Circuit Diagrams.....	A.4
Experimental Results: Motor thrust plots	A.5
MATLAB Simulation Code.....	A.6
C-Code for Proportional Control System	A.7

LIST OF FIGURES

Figure 1: 3D model of the physical structure.....	9
Figure 2: Control system schematic.....	12
Figure 3: Quad-rotor schematic	16
Figure 4: Results of first MATLAB simulation.....	20
Figure 5: Results of second MATLAB simulation	21
Figure 6: Results of third MATLAB simulation.....	22
Figure 7: Von Mises stress and displacement on smaller diameter carbon fiber rod	25
Figure 8: Von Mises stress distribution along larger diameter rod.....	26
Figure 9: Maximum von Mises stress for larger diameter rod.....	26
Figure 10: Analysis convergence.....	27
Figure 11: Von Mises stress in rod undergoing 20 times the static load	29
Figure 12: Deflection of rod undergoing 20 times the static load	30
Figure 13: Von Mises stress in rod undergoing 50 times the static load	31
Figure 14: Motor thrust test setup	32

LIST OF TABLES

Table 1: Main structural components	9
Table 2: Physical parameters used in MATLAB analysis	18
Table 3: Predicted moments of inertia	18
Table 4: Initial conditions used to solve equations in MATLAB simulation	19
Table 5: Properties for Carbon Fiber Panex 33 48K.....	24
Table 6: Convergence values	28
Table 7: Motor thrust data for final design configuration.....	33

Introduction

Scope

Unmanned aerial vehicles (UAVs) are crafts capable of flight without an onboard pilot. They can be controlled remotely by an operator, or can be controlled autonomously via preprogrammed flight paths. Such aircraft have already been implemented by the military for reconnaissance flights. Further use for UAVs by the military, specifically as tools for search and rescue operations, warrant continued development of UAV technology.

A quad-rotor helicopter is an aircraft whose lift is generated by four rotors. Control of such a craft is accomplished by varying the speeds of the four motors relative to each other. Quad-rotor crafts naturally demand a sophisticated control system in order to allow for balanced flight. Uncontrolled flight of a quad-rotor would be virtually impossible by one operator, as the dynamics of such a system demand constant adjustment of four motors simultaneously.

The goal of our project was to design and construct a quad-rotor vehicle capable of indoor and outdoor flight and hover. Through the use of an integrated control system, this vehicle would be capable of autonomous operation, including take-off, hover, and landing capabilities, through pre-programmed flight paths.

Design Proposal

Our project showcases important control capabilities which allow for autonomous balancing of a system which is otherwise dynamically unstable. A quad-rotor poses a more challenging control problem than a single-rotor or dual-rotor inline helicopter because the controls demands include accounting for subtle variations which exist between the motors and

cause each motor to provide a slightly different level of lift. In order for the quad-rotor craft to be stable, the four motors must all provide the same amount of lift, and it is the task of the control system to account for variations between motors by adjusting the power supplied to each one.

We deemed the control of a quad-rotor craft as a valuable challenge to pursue. The benefits of such a craft warrant the design challenges, as a quad-rotor craft is more efficient and nimble than a single-rotor craft. Unlike a single-rotor craft, which uses a second, smaller vertical propeller to change direction, the quad-rotor craft's directional motion is generated by the same four motors that are providing lift. Also, the quad-rotor can change direction without having to reorient itself – there is no distinction between front and back of the craft. In the quad-rotor, every rotor plays a roll in direction and balance of the vehicle as well as lift, unlike the more traditional single rotor helicopter designs in which each rotor has a specific task - lift or directional control - but never both.

Goals

We set the following goals for our quad-rotor UAV:

- 1) Stable, autonomous hover with active controls system taking dynamic information from sensors mounted on craft.
- 2) Stable hover, with addition of variations in altitude based on human input for thrust.
- 3) Stable hover, with addition of variations in altitude and directional movement based on human inputs for thrust, pitch, roll, yaw.

Parameters

Our project was developed within the following parameters:

- Total concept development cost of under \$400. An expenditure allowance of \$50 exists for the purchase of spare parts and backup material.
- Time constraint of 15 weeks, from preliminary brainstorming through to finished product.
- Ability to fabricate in-house (Columbia University Mechanical Engineering Department laboratory) or to purchase all necessary component parts, without the need to out-source any fabrication processes.
- Limit of 10 unique parts to be fabricated in-house.
- Size constraint: finished product should fit within a 3 foot cube.

Background Information

There is a fair amount of published research with regards to quad-rotor aircraft. In fact, there are many patents for designs similar to ours. Among them are a few “Four Propeller Helicopter” designs (Dammar, Michael. "Four Propeller Helicopter." US Patent D465196. November 2002.), some “Quad Tiltrotor” designs (DeTore, John A., Richard F. Spivey, Malcolm P. Foster, and Tom L. Wood. "Quad Tiltrotor." US Patent D453317. February 2002.), and various vertical lift aircrafts (Smart, R.C. "Vertical Lift Aircraft." US Patent 3185410. May 1965.) While the above mentioned four propeller helicopter and the quad tilt-rotor patent applications do not include much information aside from the purpose of the craft and the orientation of the rotors, it is without a doubt that the crafts are similar to the quad-rotor UAV. R.C. Smart's “Vertical Lift Aircraft,” on the other hand, includes operational details, such as basic information on the particular dynamics of that craft.

In the world of higher education, there are a few members of academia who have published research on quad-rotor UAVs. Among them are Joseph F. Horn and Wei Guo of Pennsylvania State University (“Modeling and Simulation for the Development of a Quad-Rotor UAV Capable of Indoor Flight”), Ming Chen and Mihai Huzmezan of the University of British Columbia (“A Simulation Model and H_∞ Loop Shaping Control of a Quad Rotor Unmanned Air Vehicle”), and Eryk Brian Nice of Cornell University (“Design of a Four Rotor Hovering Vehicle”).

An attempt to search for similar projects on the market did not yield many results. Aside from a few overachieving hobbyists, there exist only a few commercially available products which take advantage of similar quad-rotor flight: the Silverlit X-UFO, the Draganflyer V Ti, and the Microdrones GmbH MD4-200. All three of these products use four rotors in conjunction

with a control system that consists of three gyroscopes for feedback. The Microdrones GmbH MD4-200 and a particular model of the Draganflyer V Ti additionally have an onboard camera for reconnaissance purposes. However, as these crafts are designed as high-end hobbyist crafts, they also come with a fairly steep price tag, as the Draganflyer and the Microdrones products are upwards of \$1500 or more.

On a much larger, industrial scale, there is currently a project in development named the Bell Boeing Quad TiltRotor. It is a large-scale, government-sponsored, quad-rotor aircraft currently in development as a joint venture between Bell Helicopter Textron and Boeing Integrated Defense Systems. The project is the largest-scale of all the existing projects, and with a capacity of upwards of 150 passengers, far exceeds the size and span of any other similar project.

Design Details

Design Overview

Structure

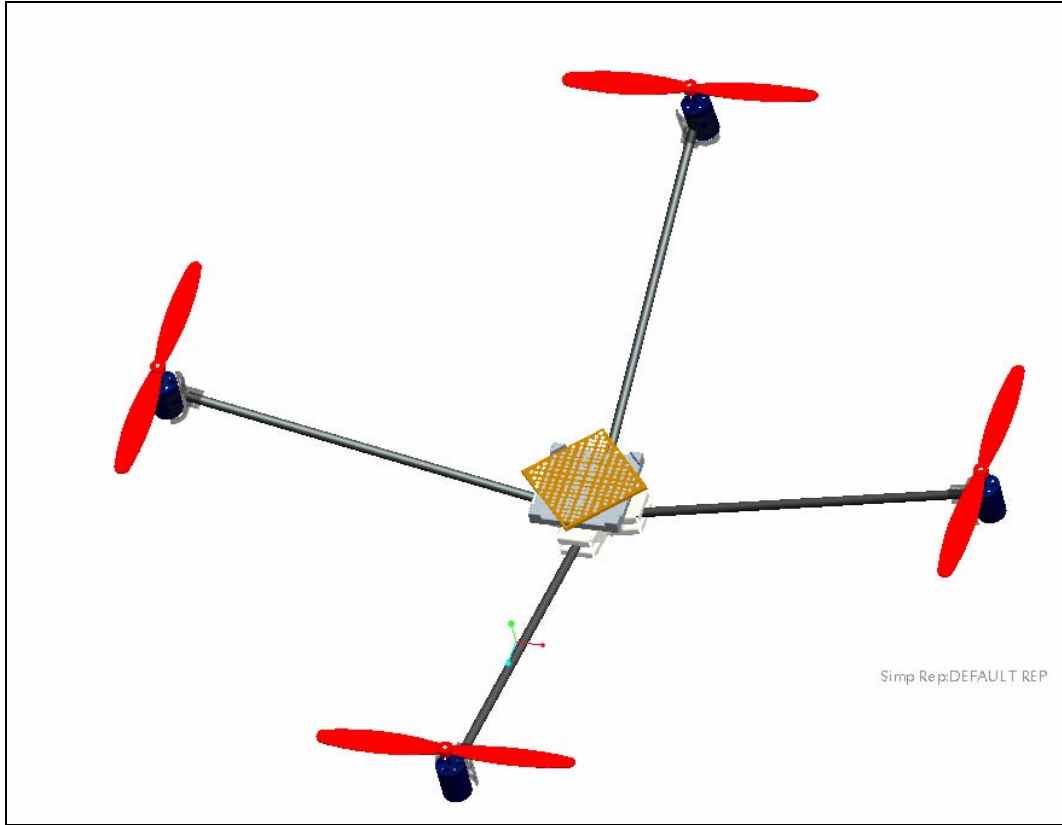


Figure 1: 3D model of the physical structure

The following table summarizes the main structural components of our quad-rotor craft design.

Table 1: Main structural components

<i>Qty</i>	<i>Name</i>
1	Central Hub
4	Arm
4	Motor mount
4	DC motor
4	Gearbox
2	Pusher Propeller
2	Tractor Propeller

Motors

The motors are cobalt, brushed, DC motors rated for 12 V, 15 amps. The DC, brushed motor configuration was desired for ease of control (ability to control via PWM). The cobalt motors use strong rare earth magnets and provide the best power to weight ratio of the hobby motors available for model aircraft. We were limited to these hobby motors by our design budget. As a result, the rest of our structural design revolves around the selection of these motors and the allowable weight of the craft based on the lift provided by these motors (approximately 350g of lift from each motor).

Propellers

The propellers are 10" from tip to tip. Two are of the tractor style, for clockwise rotation, and the other two are of the pusher style, for counterclockwise rotation. For our design, a propeller with a shallow angle of attack was necessary as it provided the vertical lift for stable hovering. The propellers we used were steeper than the ideal design because of limited availability of propellers that are produced in both the tractor and pusher styles.

Gearboxes

The gearboxes have a 2.5:1 gear ratio. They reduce the speed of the prop compared to the speed of the motor, allowing the motors to exert more torque on the propellers while drawing much less current than in a direct drive configuration.

Arms

The arms of our quad-rotor design needed to be light and strong enough to withstand the

stress and strain caused by the weight of the motors and the central hub at their opposite ends. Carbon fiber was deemed the best choice because of its weight to strength ratio. The thickness of the tube was chosen to be the smallest possible to lower its weight. The length of each arm (10") was chosen based on the propellers. The propellers used are 10" long each so we had to allow enough room for them to spin without encountering turbulence from one another. Since such a phenomenon would be quite complex to analyze, we simply distanced the motors far enough apart to avoid the possibility of turbulence interference among rotors.

Battery

The battery was selected on the basis of power requirements for the selected motor/gearbox combination. We opted for a battery of the lithium-polymer variety, despite the fact that it was considerably more expensive than other batteries providing the same power, because this battery provided the best power-to-weight ratio. Our battery choice was a 1450mah 12.0V 12C Li-polymer battery. (Note: Because we did not have enough time to integrate the circuitry of the controls system on-board, and thus performed only tethered flight, we did not ultimately purchase the battery.)

Central Hub

The central hub carries all of the electronics, sensors, and battery. It sits lower than the four motors in order to bring the center of gravity downwards for increased stability. We manufactured it using a rapid prototyping machine (Stratasys, Inc. FDM 2000 in the Columbia University Mechanical Engineering lab). Considering our design for the hub, the rapid prototyping machine was ideal because of its ability to produce relatively complex details, for

example the angled holes which allow for the central hub to sit lower than the surrounding motors. The thermoplastic polymer used in rapid prototyping has good strength to weight ratio.

Motor Mounts

The motor mounts connect the motors to the carbon fiber arms. Because of their complex details, they were manufactured using the rapid prototyping machine and therefore made of thermoplastic polymer.

Controls

The following schematic depicts our controls system. The diagram represents how the control system interacts with the physical system for controlled quad-rotor flight.

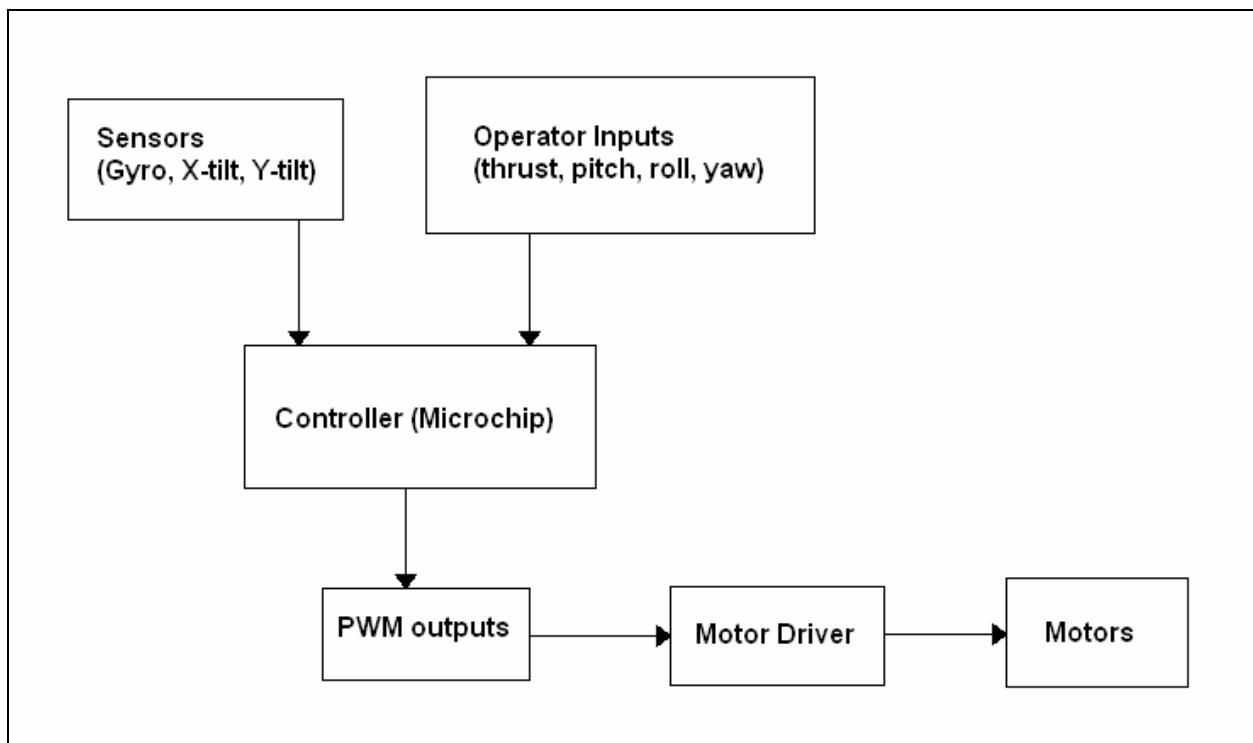


Figure 2: Control system schematic

Sensors

Gyroscope

We mounted a single-axis gyroscope in the center of the craft in order to measure the yaw rate while in flight.

XY Accelerometer

We mounted two dual-axis tilt sensors on the central hub of the craft. These tilt sensors each provided X- and Y- analog output signals. The outputs from the two sensors were averaged for higher sensitivity.

Software

The Microchip microcomputer in our design acted as the controller for the system. A proportional, closed-loop control system was chosen for its simplicity, both in implementation and function, as well as its effectiveness. Since such a system was chosen, the sole feedback for the system is proportional to the error, or the difference between the command and the output, as read by the onboard sensors. This error is then amplified by a gain so as to adjust the sensitivity of the control system.

Ultimately, such a control system was implemented by taking advantage of two particular modules on the Microchip pic18f4431 processor: the onboard high-speed A/D converter and the power pulse width modulation (PWM) module. The A/D converter is used to measure seven analog signals; four command signals representing overall thrust, roll, pitch, and yaw, and three onboard sensors, which monitor the tilt along two axes as well as the yaw rotation of the entire body. This data is then interpolated by the processor to calculate the error. This error is then amplified by a fixed gain, and since the hypothetical responses of the dynamics of the vehicle are

known, new output signals are then calculated and outputted to the four motors via the PWM module using the following equations:

$$\begin{aligned} outMotorA = & (kThrust*(inThrust) - (kElevator*(inElevator-512) + kTiltX*(inTiltX-512)) - \\ & (kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512))) \end{aligned} \quad (Eqn 1)$$

$$\begin{aligned} outMotorB = & (kThrust*(inThrust) + (kAilerons*(inAilerons-512) + kTiltY*(inTiltY-512)) + \\ & (kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512))) \end{aligned} \quad (Eqn 2)$$

$$\begin{aligned} outMotorC = & (kThrust*(inThrust) - (kAilerons*(inAilerons-512) + kTiltY*(inTiltY-512)) + \\ & (kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512))) \end{aligned} \quad (Eqn 3)$$

$$\begin{aligned} outMotorD = & (kThrust*(inThrust) + (kElevator*(inElevator-512) - kTiltX*(inTiltX-512)) - \\ & (kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512))) \end{aligned} \quad (Eqn 4)$$

These equations factor in seven gains (represented by the variable names beginning with “k”) corresponding to their respective inputs, as well as the fact that for all of the sensors and inputs, except for thrust, the values allow for both positive and negative variation from the rest value. There also exists a fail-safe where the level of thrust for a particular motor never falls below a given limit. This fail-safe prevents the motors, which require a fair amount of power to start from idle, from stalling in mid-air. The end result is a microcontroller which not only enables the UAV to fly, but provides stability in doing so.

Circuit

The schematics for the circuits used in our project are provided in Appendix A.4. Namely, they detail the circuit used to amplify the sensor signals and send them to the microcomputer, as well as the circuit used to amplify the PWM outputs in order to drive the motor. There are also schematics for the individual connections made to each of the sensors and the microcomputer.

Analysis

Dynamic Simulation

System of Differential Equations

The dynamics of the quad-rotor UAV are determined from a set of equations of motion. The complexity of the equations of motion increases with increased accuracy. A highly complex set of equations and their derivation can be found in *PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor* by Bouabdallah, et al. A simplified version of these equations is presented by Altug, et al in *Control of a Quadrotor Helicopter Using Visual Feedback*. The set of equations presented model the motion of the craft based on the amount of lift delivered by each individual motor without taking into account the aerodynamics of the craft.

Figure 3 is a top-down view of the craft. Motion is obtained by varying the amount of lift each motor provides. The amount of lift each motor provides is controlled by the amount of power delivered to each motor. Depending on the motor and gearbox used, the relationship can be linear, parabolic, or a combination of various other trigonometric functions.

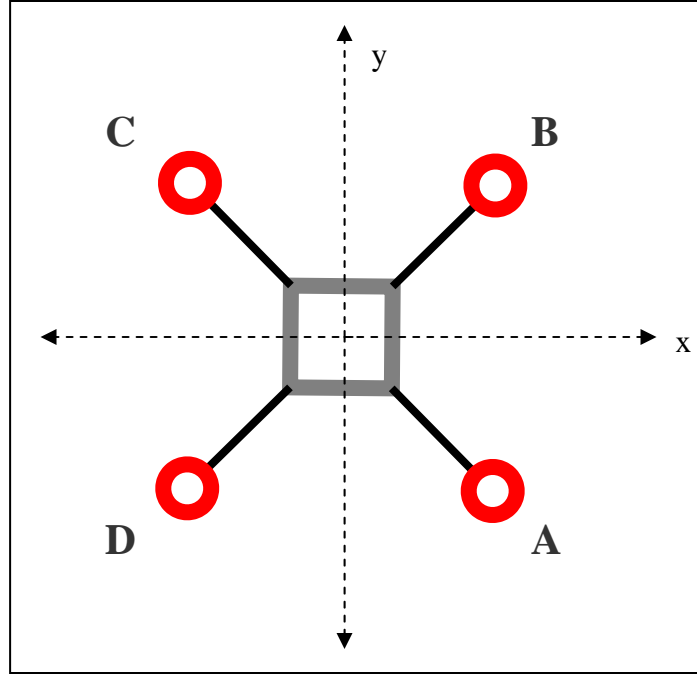


Figure 3: Quad-rotor schematic

For example, the craft will move in the positive x-direction by reducing the thrust from motor A (by reducing the power) and simultaneously increasing the thrust (by increasing the power) from motor C. The thrust from motor B and D must be increased so that the craft maintains constant altitude while moving along the desired path. More complex movements can be achieved by varying the power delivered to all four motors and will be studied in the following subsection. The equations of motion that govern the dynamics of the craft are listed below. (*Altug, et al, 2002*)

$$\ddot{x} = \frac{\left(\sum_{i=1}^4 F_i \right) (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) - K_1 \dot{x}}{m} \quad (\text{Eqn 5})$$

$$\ddot{y} = \frac{\left(\sum_{i=1}^4 F_i \right) (\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi) - K_2 \dot{y}}{m} \quad (\text{Eqn 6})$$

$$\ddot{z} = \frac{\left(\sum_{i=1}^4 F_i\right)(\cos \phi \cos \psi) - mg - K_3 \dot{z}}{m} \quad (\text{Eqn 7})$$

$$\ddot{\theta} = \frac{L(-F_1 - F_2 + F_3 + F_4 - K_4 \dot{\theta})}{J_1} \quad (\text{Eqn 8})$$

$$\ddot{\psi} = \frac{L(-F_1 + F_2 + F_3 - F_4 - K_5 \dot{\psi})}{J_2} \quad (\text{Eqn 9})$$

$$\ddot{\phi} = \frac{L(-M_1 - M_2 + M_3 - M_4 - K_6 \dot{\phi})}{J_3} \quad (\text{Eqn 10})$$

θ , ψ , and ϕ are the pitch, roll, and yaw respectively. The forces on the motors are given by the F_i terms. The moments of inertia of the craft with respect to the axes are given by the J_i terms (where x corresponds to 1, y corresponds to 2, and z corresponds to 3). The K_i terms represent the drag coefficients, which can be ignored for simplicity. C is a force to moment scaling factor. The center of gravity is assumed to be on the origin.

Simulations

The craft can increase in altitude by simultaneously increasing the thrust from all motors. Likewise, the craft can descend (if already airborne) by simultaneously decreasing the thrust from all motors.

We used MATLAB to solve the system of six differential equations that simulate the craft's motion. We used the packaged ODE45 program as the solver. Please see Appendix A.6 to see the two sets of code that solves the system. Please see the topic "ODE45" under the help menu to get help in solving ordinary differential equations using the ODE45 package.

Simulations were performed to ensure the validity of our basic assumptions about the behavior of the craft when power is varied to the motors. When possible, the basic parameters

were estimated. Parameters that could not be estimated without a detailed study of the craft and the air around it were estimated to unity.

The moments of inertia were determined by modeling the central hub as a square, the motors as cylinders, and the propellers as rods. We used the parallel axis theorem to sum the moments of inertia due to each individual motor. Table 2 summarizes the parameters used in these calculations. Table 3 summarizes the predicted moments of inertia.

Table 2: Physical parameters used in MATLAB analysis

Parameter	Value	Description
W [m]	0.1	Width of hub
D [m]	0.1	Depth of hub
H [m]	0.02	Height of hub
Q [kg]	0.09	Mass of motor
P [kg]	0.1	Mass of hub
R [m]	0.012	Radius of motor
H [m]	0.036	Height of motor
R [m]	0.25	Distance from motor to hub

Table 3: Predicted moments of inertia

	Equation	Value [kg*m ²]
J ₁	$(1/3)*Q*(3*r^2+h^2)+4*Q*R^2+(1/12)*P*(H^2+D^2)$	0.026
J ₂	$(1/3)*Q*(3*r^2+h^2)+4*Q*R^2+(1/12)*P*(H^2+D^2)$	0.026
J ₃	$2*Q*r^2+4*Q*R^2+(1/12)*P*(W^2+D^2)$	0.027

Table 4 summarizes the values of the initial conditions we used to solve the system of equations along with the parameters in Table 2 and Table 3.

Table 4: Initial conditions used to solve equations in MATLAB simulation

	Name in Code	Value	Description	Units
$x(t=0)$	x_1	0	Initial position along x-axis	[m]
$y(t=0)$	x_3	0	Initial position along y-axis	[m]
$z(t=0)$	x_5	0	Initial position along z-axis	[m]
$\theta(t=0)$	x_7	0	Initial pitch	[deg]
$\phi(t=0)$	x_9	0	Initial roll	[deg]
$\psi(t=0)$	x_{11}	0	Initial yaw	[deg]
$\dot{x}(t=0)$	x_2	0	Initial velocity along x-axis	[m/s]
$\dot{y}(t=0)$	x_4	0	Initial velocity along y-axis	[m/s]
$\dot{z}(t=0)$	x_6	0	Initial velocity along z-axis	[m/s]
$\dot{\theta}(t=0)$	x_8	0	Initial rate of change in pitch	[deg/s]
$\dot{\phi}(t=0)$	x_{10}	0	Initial rate of change in roll	[deg/s]
$\dot{\psi}(t=0)$	x_{12}	0	Initial rate of change in yaw	[deg/s]

Simulation 1:

When all motors are left at the same thrust (1.3N), the craft rises because the total thrust is larger than the weight of the craft. This is pictured in Figure 4. The average climbing speed is 3 m/s.

There is no velocity in the xy-plane because the pitch, roll, and yaw are zero.

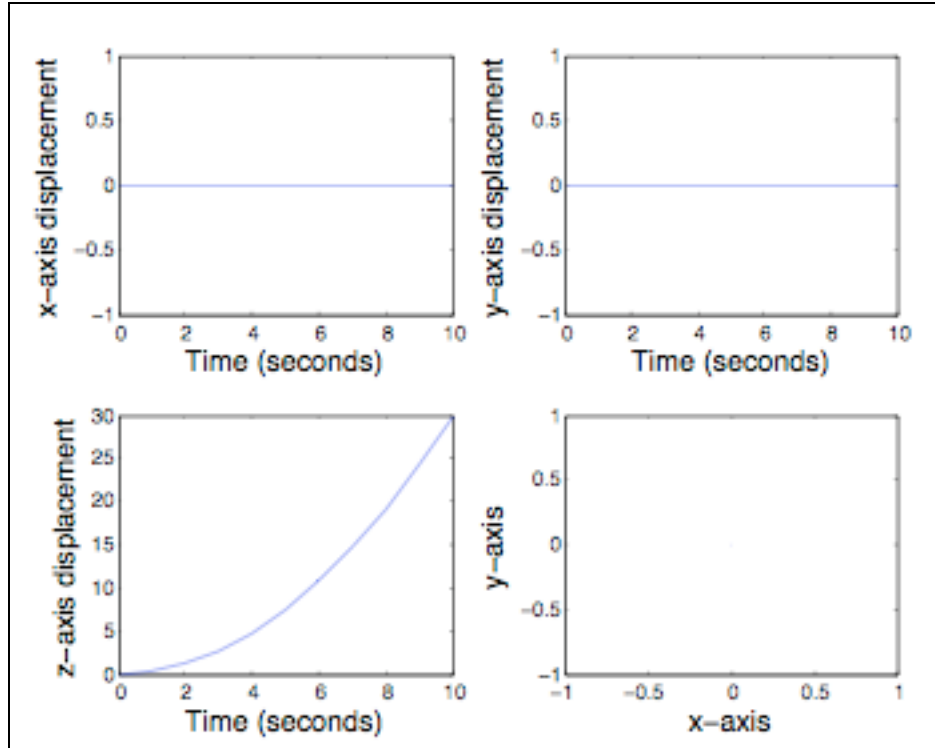


Figure 4: Results of first MATLAB simulation

Simulation 2:

We simulate movement of the craft when the thrust from motor A is increased by 10% and the thrust from motor C is decreased by 10%. The thrust from motors B and D remains constant.

Figure 5 displays the expected behavior of the craft.

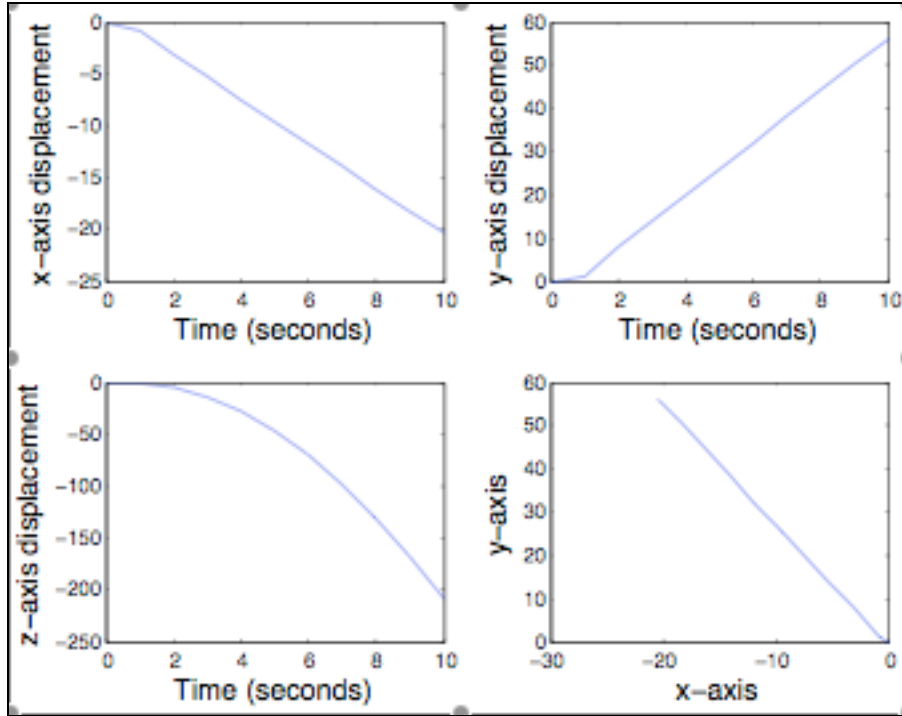


Figure 5: Results of second MATLAB simulation

The average velocity in the xy-plane is about 5 m/s. The average vertical velocity is 20 m/s in the negative z-direction (falling). It is noteworthy to point out that the craft moved in the xy-plane in the direction predicted from theoretical analysis; however, the craft was also vertically displaced. The drop in height was surprising because we expected the craft to maintain most of its vertical thrust.

Simulation 3:

We determined that significant velocity can be obtained without a corresponding loss in altitude. Figure 6 displays the results of simulated flight when the thrust in motor A is increased by 0.01% and the thrust in motor C is decreased by 0.01%. The average velocity in the xy-plane is about 2 m/s. In this case, the craft continues rising at the same speed as in Simulation 1 (about 3 m/s).

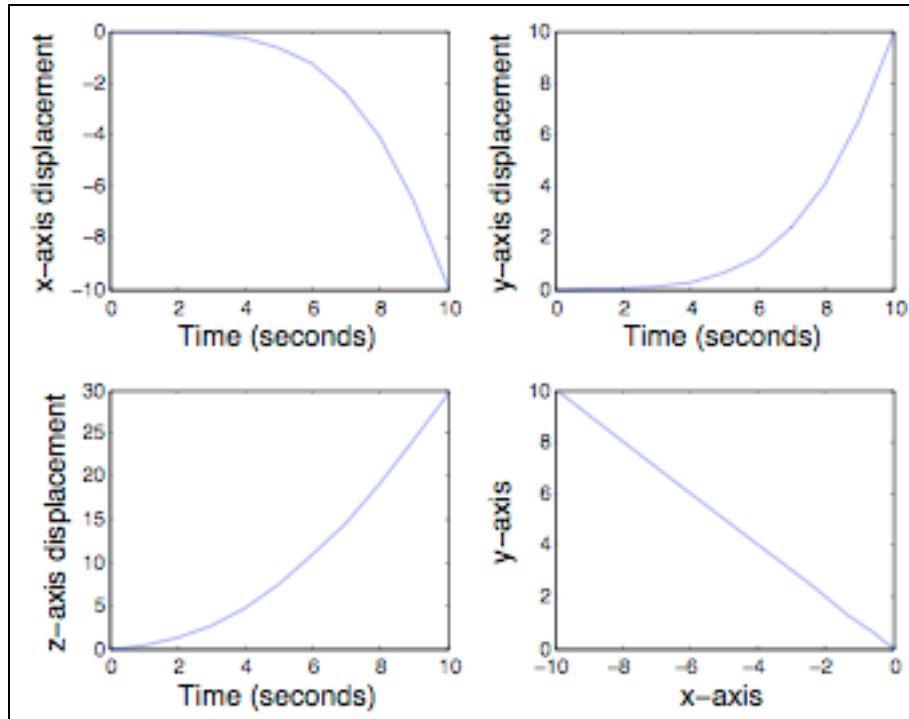


Figure 6: Results of third MATLAB simulation

Figure 5 and Figure 6 imply that the craft must be maneuvered with relatively small changes in thrust from each motor. The change in thrust in Simulation 2 is 1000 times larger than the change in thrust in Simulation 3, yet the average velocity in the xy-plane is only 2 or 3 times larger. In addition, the craft continues to rise in simulation 3 but falls fast in simulation 2.

Finite Element Stress Analysis

Carbon Fiber Arms

A number of materials were considered before deciding to use hollow carbon fiber tubes to build the arms of the craft. Once carbon fiber was chosen for its lightweight properties, the diameter and length were considered using FEA to select the configuration with the smallest mass, while maintaining robustness of design. First, a safety factor was selected based on the

use of the material. Use falls under *better-known materials that are to be used in uncertain environments or subject to uncertain stresses*. This gives a SF of 3-4, but to assure that failure does not occur over a long period a larger value was sought.¹ This larger SF takes into account that significant dynamic forces can act on this craft and that a long period of strenuous testing and evaluation will be required to achieve reliability of the control system. In addition, all forces cannot be predicted for this type of vehicle. Since it is built to represent a complex dynamic flight system there is a chance that stresses outside the ordinary range will occur. This can be seen in many flight systems that experience degrees of overcorrection in flight, outside disturbances, and incidental contact with solid objects. The student filled, enclosed test environment also creates an additional hazard. A failure can turn a piece of the craft or other equipment into a projectile. Quick changes in rotor speed may apply additional forces along the transverse axis of the mount arm.

The hollow carbon fiber rod with inner diameter of 0.158” and outer diameter of 0.254” was selected based on the fact that it is widely available, that it has a negligible deflection magnitude for a length of 10.75”, and that it has the ability to handle large transverse forces at its ends.

Although a carbon fiber rod of much smaller diameter would be able to operate safely under known lift forces, both the prospect of unknown dynamic forces and the large deflection of smaller diameter rod necessitated the choice of the more robust configuration.

Predicted Maximum Static Load

Failure criterion for this application was selected based on a life cycle with regular replacement as fatigue was not considered in this analysis. The analysis was performed by

¹ <http://www.mech.uwa.edu.au/DANotes/SSS/safety/safety.html>

placing a transverse force on one end of the rod while the other end remained fixed. This analysis would have been performed by break tests in the lab if the cost of carbon fiber rod was not limiting. The material used is similar to that described in the chart below. The maximum failure stress was determined to be 551 ksi and the maximum stress occurs near the fixed end.

Table 5: Properties for Carbon Fiber Panex 33 48K²

Physical Properties	Metric	English
Density	1.81 g/cc	0.0654 lb/in ³
Mechanical Properties		
Tensile Strength, Ultimate	3800 MPa	551000 psi
Modulus of Elasticity	228 GPa	33100 ksi

Judging from an analysis of a slightly smaller carbon fiber rod, it is clearly seen that the magnitude of deformation is excessive and would adversely affect the thrust vector produced by a mounted rotor. Additionally, it would introduce unnecessary vibration into the system.

² <http://www.matweb.com/search/SpecificMaterial.asp?bassnum=ECZO00>

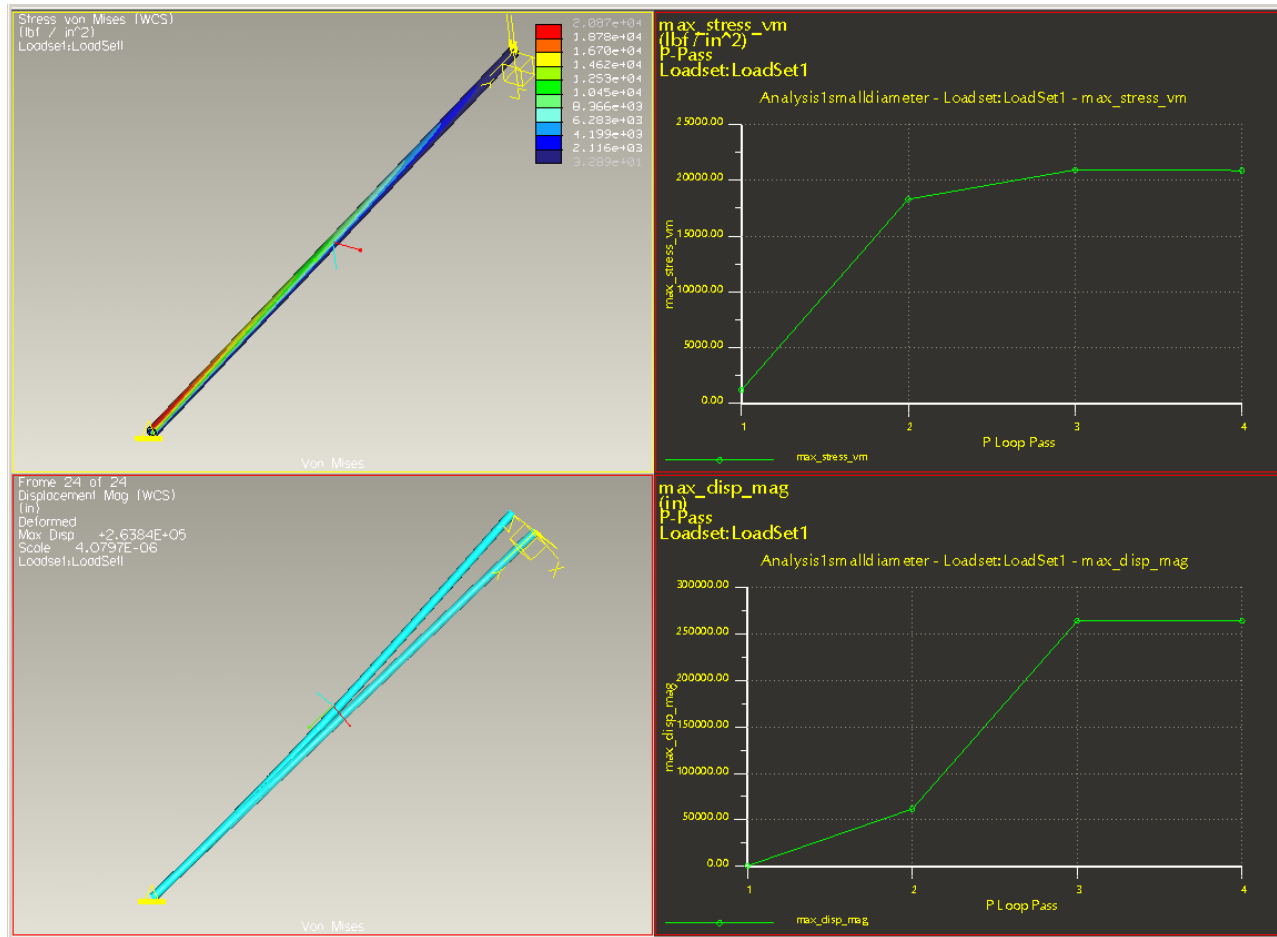


Figure 7: Von Mises stress and displacement on smaller diameter carbon fiber rod

Above are charts and pictorial representations of the maximum von Mises stress and magnitude of displacement on a small diameter hollow carbon fiber rod under expected static loading. Note that the displacement is several orders of magnitude larger than that predicted in the static test of the larger diameter rod given below.

For the larger diameter rod, as the below chart shows, the max von Mises stress occurs near the fixed end of the rod. The max stress reached is less than 10 ksi as shown on the legend in the upper right corner.

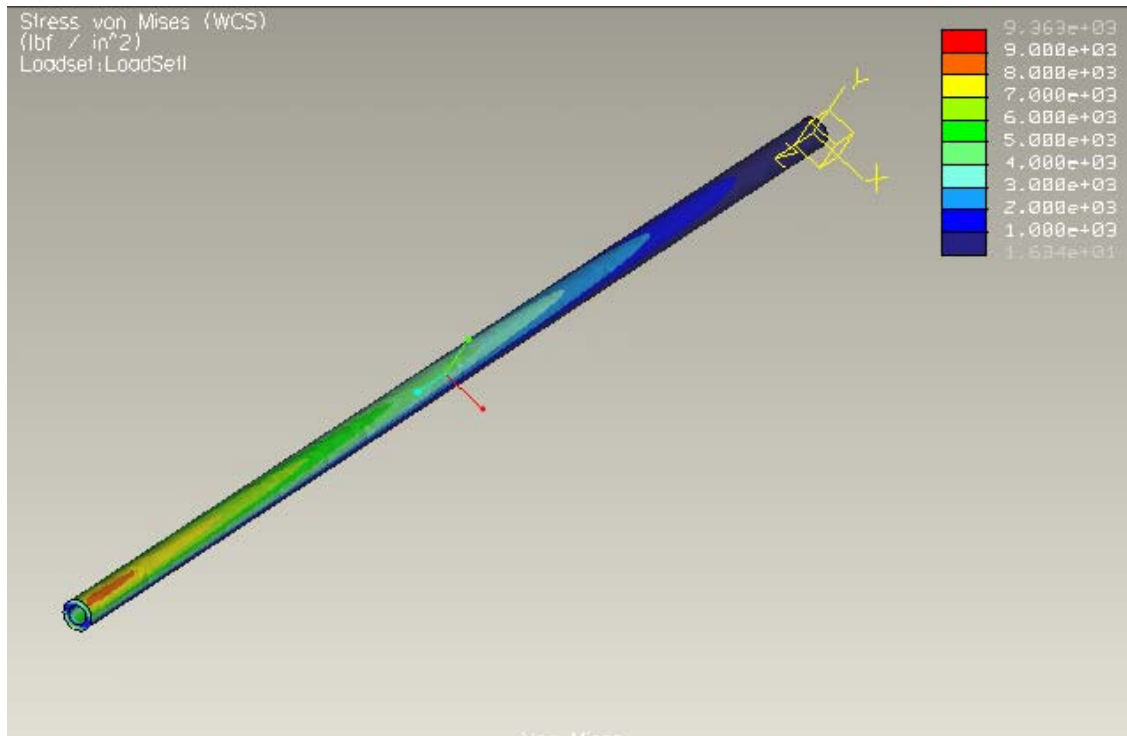


Figure 8: Von Mises stress distribution along larger diameter rod

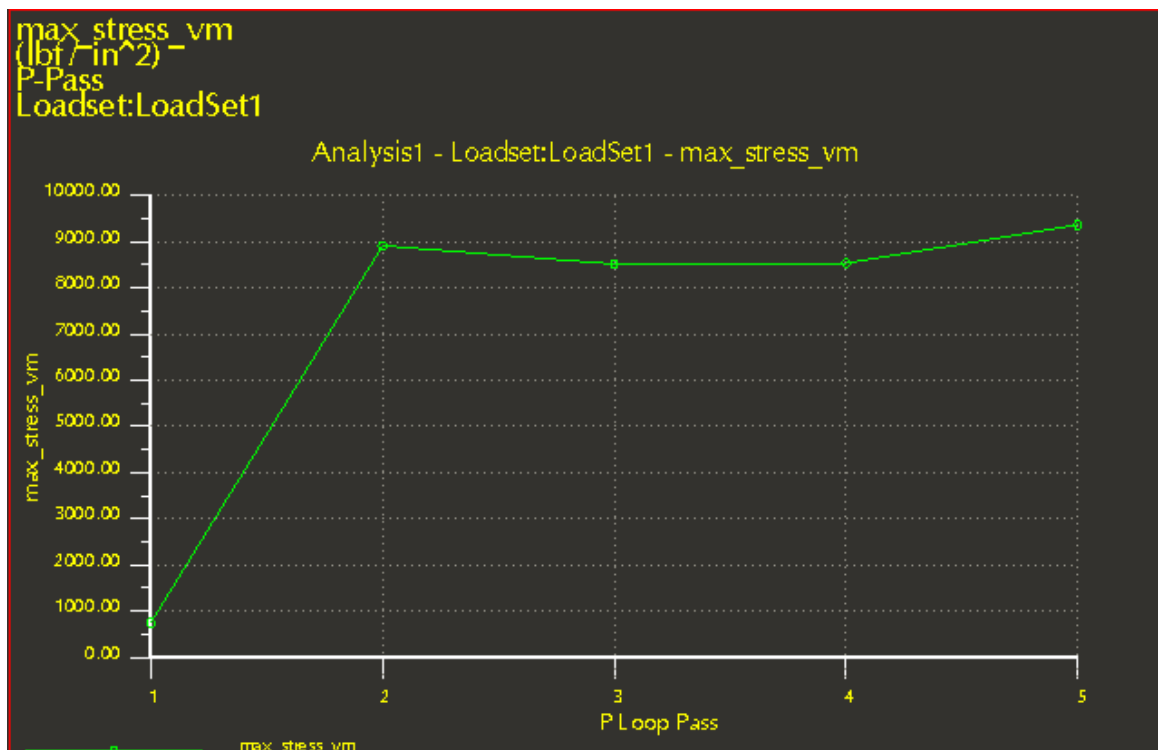


Figure 9: Maximum von Mises stress for larger diameter rod

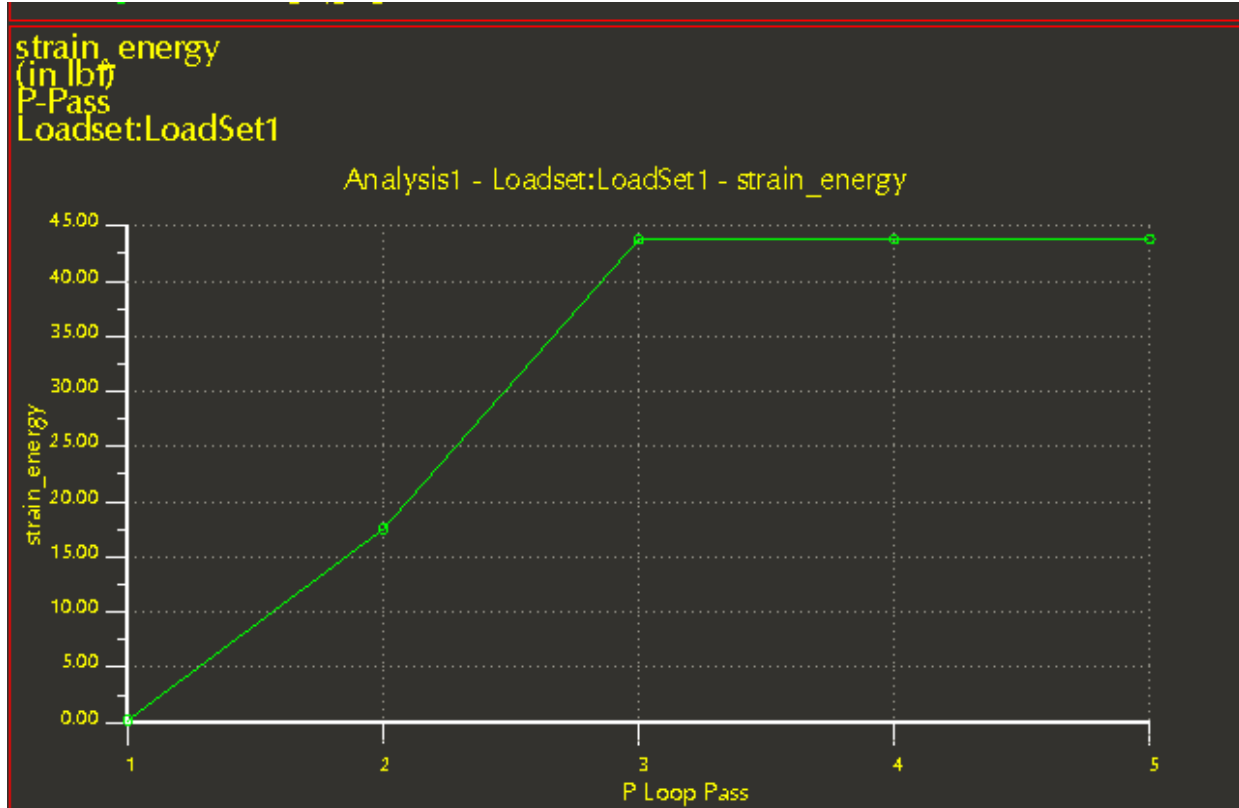


Figure 10: Analysis convergence

The analysis algorithm converges to less than 5% within 9.4 ksi by pass 5 and the strain energy is seen to converge at around 44 in lbf as shown in Figure 10 above.

Table 6: Convergence values

max_disp_mag:	7.946422e+01	0.0%
max_disp_x:	-4.944648e-03	0.6%
max_disp_y:	7.945180e+01	0.0%
max_disp_z:	-1.407400e+00	0.0%
max_prin_mag:	1.422705e+04	9.8%
max_rot_mag:	0.000000e+00	0.0%
max_rot_x:	0.000000e+00	0.0%
max_rot_y:	0.000000e+00	0.0%
max_rot_z:	0.000000e+00	0.0%
max_stress_prin:	1.422705e+04	9.8%
max_stress_vm:	9.363115e+03	8.8%
max_stress_xx:	5.548397e+03	8.7%
max_stress_xy:	-6.098574e+02	4.3%
max_stress_xz:	2.241477e+03	10.7%
max_stress_yy:	5.692178e+03	8.0%
max_stress_yz:	-3.414498e+03	14.7%
max_stress_zz:	1.283528e+04	9.0%
min_stress_prin:	-1.273572e+04	7.8%
strain_energy:	4.379018e+01	0.0%

The exact values produced by the static analysis along with their percent convergence can be seen above.

Predicted Maximum Dynamic Load

To test the limits of the carbon fiber arm, analyses were performed at both 20 times the expected static load and 50 times the static load. A force similar to that of a 20 lb or 50 lb mass was suspended from the end of the rod.

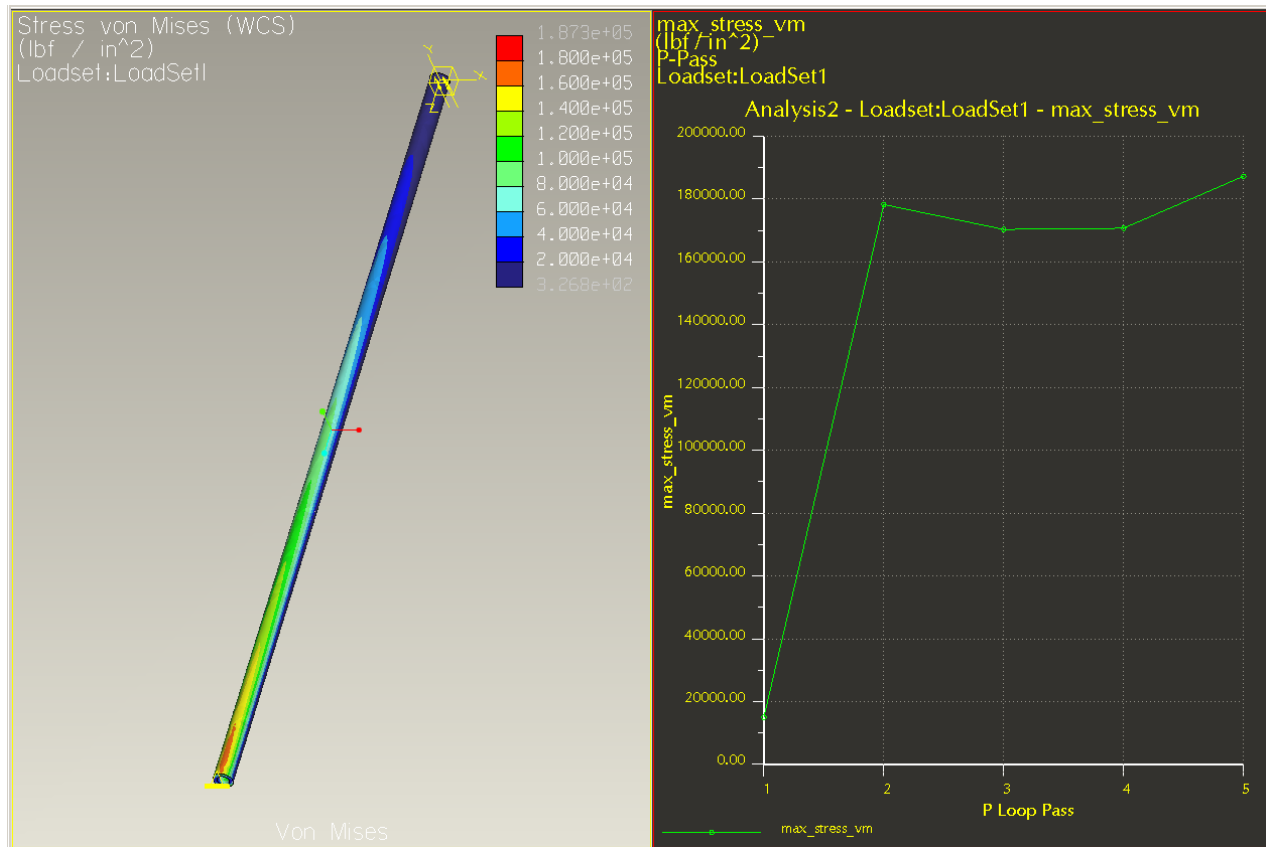


Figure 11: Von Mises stress in rod undergoing 20 times the static load

With nearly 20 lbs of force applied to the end of the arm, we are still within the failure limits of the material. Significant deflection can be seen (see Figure 12 below). The magnitude of deflection of the arm is more than two orders of magnitude greater than that achieved in the static load test. However, since this deflection is not enough to snap the arm, this part is still acceptable for this application.

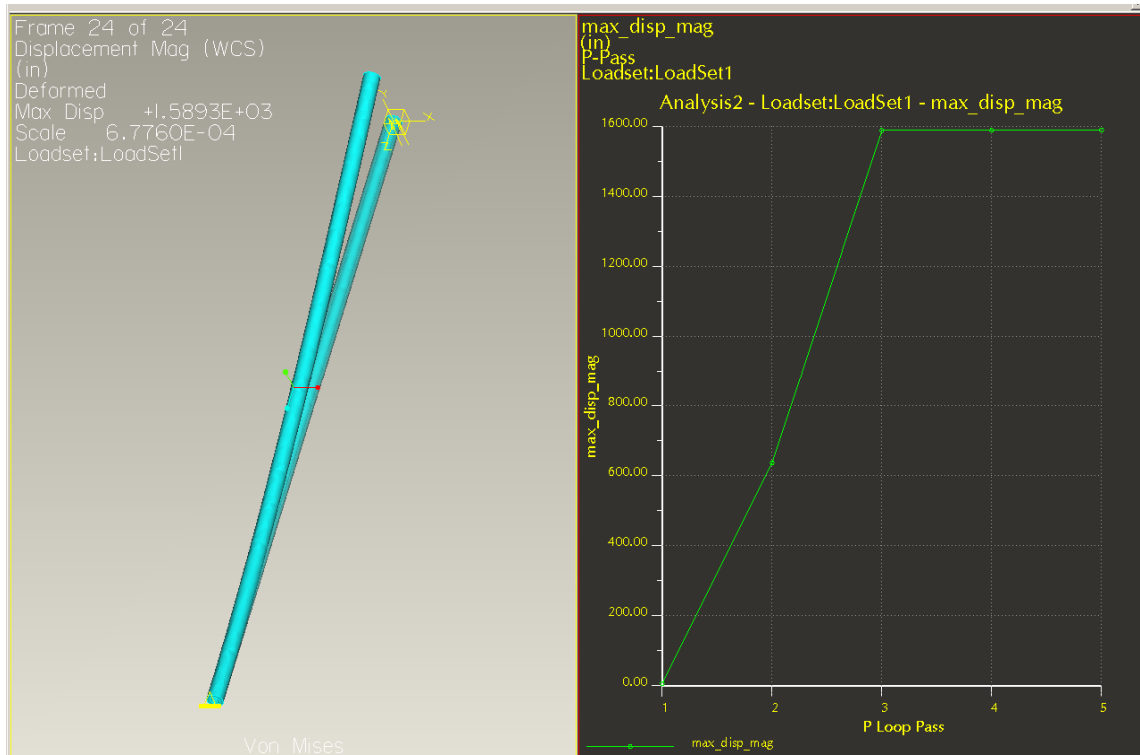


Figure 12: Deflection of rod undergoing 20 times the static load

Only by increasing the static load magnitude by 50 times do we reach the neighborhood of failure limits of the carbon fiber arm (see Figure 13).

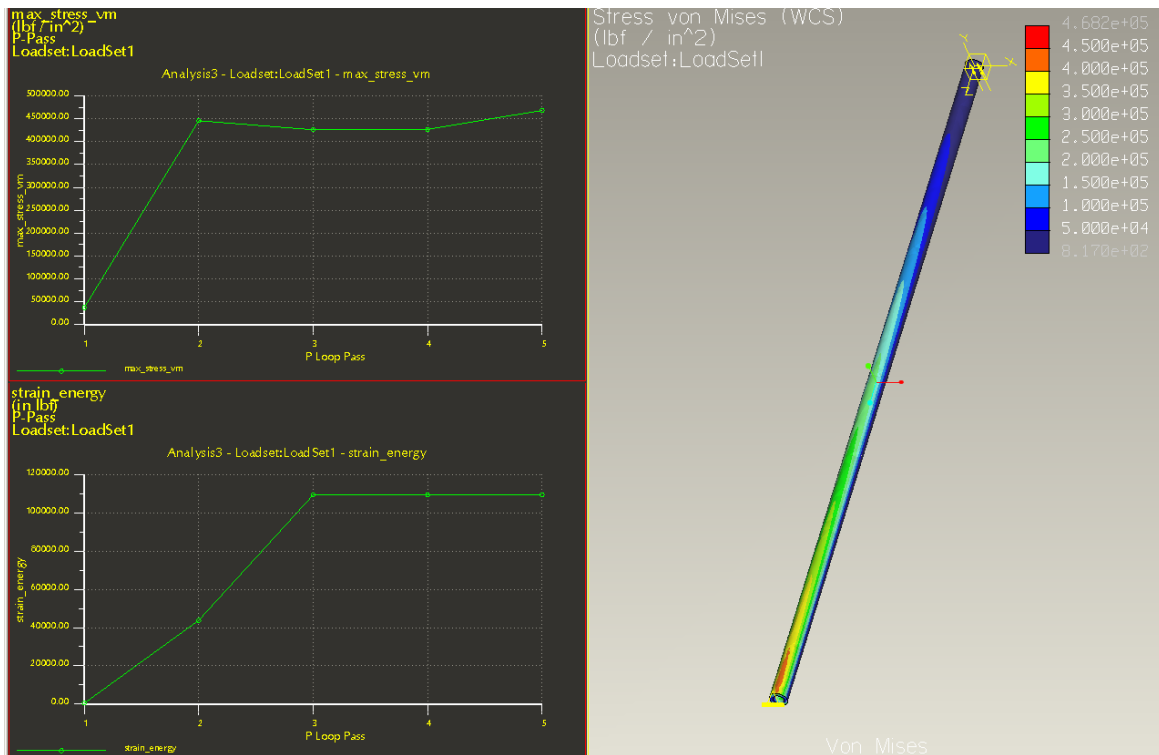


Figure 13: Von Mises stress in rod undergoing 50 times the static load

It is apparent that the carbon fiber arm design is very robust. This design with such a high safety factor can be carried out because of the lightweight nature of carbon fiber. The robustness is necessary because of unpredictability of flight conditions, the possibility for rough landings, and the cost prohibitive nature of break testing. Using the results above, we can estimate that nearly 50 times the force modeled would have to be applied in order to snap the rigid carbon fiber. This figure assumes an ultimate stress around 550 ksi. In other words, a 50 lb force applied to the end of the arms approaches the failure limit for this structural component.

Experiments

Hardware Experiments

We experimented with various motors, gears, and propellers to determine the configuration that provided the most lift with the best possible combination of low weight and low power requirements. Figure 14 illustrates the setup used to test the amount of lift provided by each motor/gearbox/propeller combination through a range of voltage and current levels.

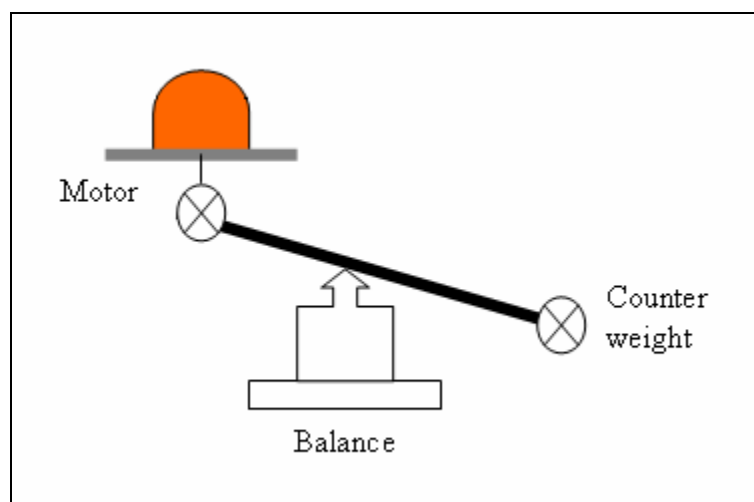


Figure 14: Motor thrust test setup

The data for the motor/gearbox/propeller combination used in our final design is summarized in Table 7. The data for other motor/gearbox/propeller configurations tested during the design analysis are summarized in Appendix A.5.

Table 7: Motor thrust data for final design configuration

Motor/Gearbox/Propeller Combination: Global Super Cobalt/10x8 Pusher Propeller/Electric Gear Drive		
Voltage [Volts]	Current [Amps]	Lift [grams]
3	5	60
5	6	120
6	7	260
7	7	220
7.5	9	280
8	10	310
8	15	360

Software Experiments

Through trial and from results of MATLAB simulations we determined that the controls can rectify the crafts' movements better if small gains are used ($k = 0.8$) rather than large gains (larger than 2). With gain values less than 1, the controls system is able to stabilize the craft at low voltage/current.

Using the information gained from our simulations, we tried a wide range of gains in the actual aircraft, ranging from $k = 0.1$ all the way to $k = 5.0$. When the gains were too high ($k > 2.0$), the craft would overcorrect itself too rapidly, causing it to become highly unstable. With gains from $k = 0.8$ to $k = 2.0$, the craft would still overcorrect itself and would seemingly remain stable for a short while before becoming unstable. For the gains $k = 0.2$ and $k = 0.5$, the craft was much more stable in hovering, was stable in response to given pitch/roll/yaw commands and altogether was much more stable in all-around flight, despite taking longer to correct given errors.

Cost Assessment

The total cost of the final project was assessed at \$207. A part-by-part cost analysis is included in the detailed parts list provided in Appendix A.2. The only significant cost not included in the assessment was the cost of the gyro and tilt sensors; these were omitted as they were provided as educational samples from Analog Devices. Market price for these sensors is roughly \$30 for the gyro, and \$10 each for the tilt sensors, for a total project cost including sensors of \$257.

Design Strengths and Weaknesses

The proposed concept for an aerial vehicle is very flexible because of the four rotors' direct involvement in the direction, lift and balance duties of the craft. All the component parts of this design can easily be replaced, adjusted, or modified. The concept benefits highly from its simplicity. More complex control problems can be incorporated into the project as time allows, namely the addition of ultrasonic and RFID sensors.

A weakness of our quad-rotor concept is the strict limits on payload. Motors falling within our price constraints are limited in terms of the thrust they produce. Thus, we are working with a very limited weight for the total craft. Linked to this issue, a second weakness of our design is the short flight time allotted for by the batteries. Again, the batteries we can use are limited by the two main factors of price considerations and weight limits.

Conclusions

The aim of our project was to have a fully functional quad-rotor helicopter capable of autonomous hover and directional motion based on operator inputs. We barely achieved our first goal of autonomous hover. During our test runs we achieved lift of the craft, and some level of autonomous hover, with visible corrections from the control system. However, noise from the tilt sensors provided enough uncertainty that we were not willing to attempt flight without a cable to secure the craft from flipping over and losing altitude.

Despite not achieving untethered flight, we are pleased with the results we did obtain. We feel that our design and construction of the physical structure was a success both in terms of stiffness and strength of the craft, and in terms of weight reduction. Our final craft weighed in at 720 grams. Allowing for an additional 300 grams for an on-board power supply, the total vehicle weight still falls well below the total lift of 1400 grams provided by the four motors.

We also succeeded in writing fully functional software for the control of the four motors. We implemented this software code on a commercially available microcomputer and obtained visible results with an oscilloscope set up to read the sensor and operator inputs and PWM outputs.

Another success was the cost of our project. We remained comfortably below the \$400 budget of the project. While revised sensors and battery would increase the cost of the project roughly another \$150, the cost of our craft is still well below that of commercially available quad-rotor crafts which demand \$1500 premiums.

Our goals for designing and building our UAV were ambitious given the timeframe of 15 weeks, and now we recognize that another 15 weeks would be in order for experimentation with

various sensors to refine a proper control system. However, we feel the progress that we have outlined above was significant along the way to our goal of autonomous flight.

Future Improvements

Given the ambitious nature of the project and the strict time limitations on arriving at a final product, there are many future improvements we would pursue having more time:

Our dynamic analysis shows that in producing motion other than up/down, the craft will lose altitude. This occurs because the increased lift from the motor which speeds up does not compensate entirely for the reduced lift of the opposite motor which slows down. Based on this analysis, a future improvement would be to incorporate an accelerometer to quantify losses in altitude. The necessary corrections based on this sensor information could easily be incorporated into the existing code.

A more immediate improvement would be to improve the existing circuit and sensors for quantifying X- and Y-tilt. Specifically, the existing tilt sensors were plagued by excess noise. We would revise the circuit to minimize noise. Also, we would potentially change sensors. One theory was that our tilt sensors' outputs were disturbed by vibrations of the structure when in flight, and perhaps less sensitive tilt sensors could yield better results of the control system. We attempted to average the readings of the tilt sensors in our code in order to filter out signal fluctuations but this method was unsuccessful as the time lag was too long for an airborne system.

References

- Allug, E. et al. *Control of a Quadrotor Helicopter Using Visual Feedback*. International Conference on Robotics & Automation, Washington, DC May 2002. <<http://ieeexplore.ieee.org/iel5/7916/21826/01013341.pdf>>.
- Bouabdallah, S. et al. *PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor*. <<http://asl.epfl.ch/aslInternalWeb/ASL/publications/uploadedFiles/330.pdf>>.
- Chen, M. and Mihai Huzmezan. *A Simulation Model and H-infinity Loop Shaping Control of a Quad Rotor Unmanned Aerial Vehicle*. Proceedings of the IASTED International Conference on Modelling, Simulation and Optimization, Banff, Canada, July 2-4, 2003.
- Guo, W. and J. Horn. *Modeling and Simulation for the Development of a Quad-Rotor UAV Capable of Indoor Flight*. Department of Aerospace Engineering, Pennsylvania State University, University Park, PA, 2006.
- Nice, Eryk B. *Design of Four Rotor Hovering Vehicle*. Thesis presented for the degree of Masters of Science, Cornell University, May 2004.
- Wright, Douglas. *Stress, Strength, and Safety*. 2005. <<http://www.mech.uwa.edu.au/DANotes/SSS/safety/safety.html>>.
- Zoltek Panex 33 48K Continuous PAN Carbon Fiber Properties. 2007. <<http://www.matweb.com/search/SpecificMaterial.asp?bassnum=ECZO00>>.

Appendix

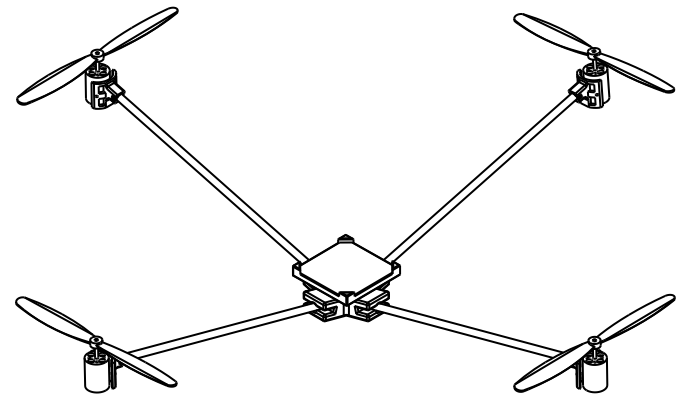
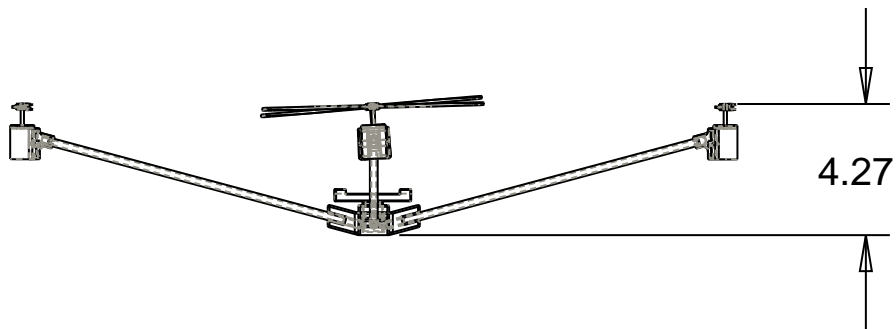
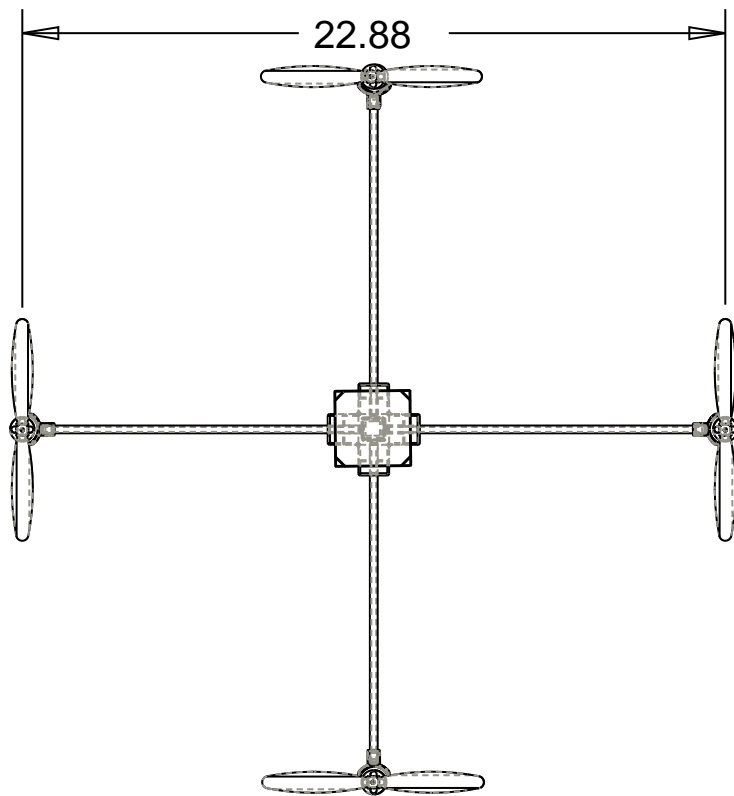
A.1: Project Timeline

	ID	Name	Start	Finish	Jan 2007			February 2007				March 2007				April 2007					
					15	22	29	05	12	19	26	05	12	19	26	02	09	16	23	30	
		Brainstorming	1/16/2007	1/25/2007																	
		Initial Design	1/18/2007	2/13/2007																	
		Component Research	1/18/2007	2/20/2007																	
		Component Testing	1/30/2007	2/20/2007																	
		Design Review #1	2/22/2007	2/22/2007																	
		Fabrication & Assembly	2/23/2007	3/9/2007																	
		Design Review #2	3/5/2007	3/5/2007																	
		Testing, Analysis, Tweaking	3/5/2007	4/17/2007																	
		Technical Evaluation	3/27/2007	3/27/2007																	
		Fabrication & Assembly Evaluation	4/3/2007	4/3/2007																	
		Web Site Construction	4/5/2007	4/19/2007																	
		System Evaluation	4/17/2007	4/17/2007																	
		In-House Demonstration	4/26/2007	4/26/2007																	
		Final Reports	5/4/2007	5/4/2007																	
		Design Expo	5/6/2007	5/6/2007																	

A.2: Parts List

COMPREHENSIVE PARTS LIST							
Part Name	Manufacturer	Part Number	Unit Wt (g)	Unit Price	Qty	Total Wt (g)	Price
Pusher Propeller 10x8	APC Props	LP-1008P	18	\$ 4.54	2	36	\$ 9.08
Tractor Propeller 10x8	APC Props	LP-10080	18	\$ 2.63	2	36	\$ 5.26
Electric Gear Drive	ElectriFly	GPMG0850	10	\$ 12.99	4	40	\$ 51.96
Global Super Cobalt 400 27T Motor							
Bottom of Form		Super 400					
	WattAge	27T	110	\$ 29.99	4	440	\$ 119.96
Carbon fiber tube .254" OD	McMaster-Carr	2153T35	40	\$ 20.71	1	40	\$ 20.71
Dual-Axis Accelerometer	Analog Devices	ADXL202	0.5	sample	2	1	-
Single Chip Rate Gyro	Analog Devices	ADXRS150	0.5	sample	1	0.5	-
Single Chip Rate Gyro Evaluation Board	Analog Devices	ADXRS150EB	0	sample	1	0	-
		PIC18f4431					
pic18f4431 Microcontroller	Microchip	I/P	2	sample	1	2	-
MOSFET Transistor	Fairchild	50N06L	0	sample	4	0	-
30A, 600V HyperFast Diode	Fairchild	RHRP3060	0	sample	4	0	-
	Texas						
Drive Amplifier	Instruments	TLV2462CP	0	sample	1	0	-
	Texas						
Darlington Transistor Array	Instruments	ULN2003AN	0	sample	1	0	-
Central hub	rapid prototyping	-	65	-	1	65	-
Circuit board mount	rapid prototyping	-	18	-	1	18	-
Motor mount	rapid prototyping	-	10	-	4	40	-
TOTALS:						718.5	\$ 206.97

A.3: CAD Drawing Sheets



SCALE 0.160

Vehicle Assembly

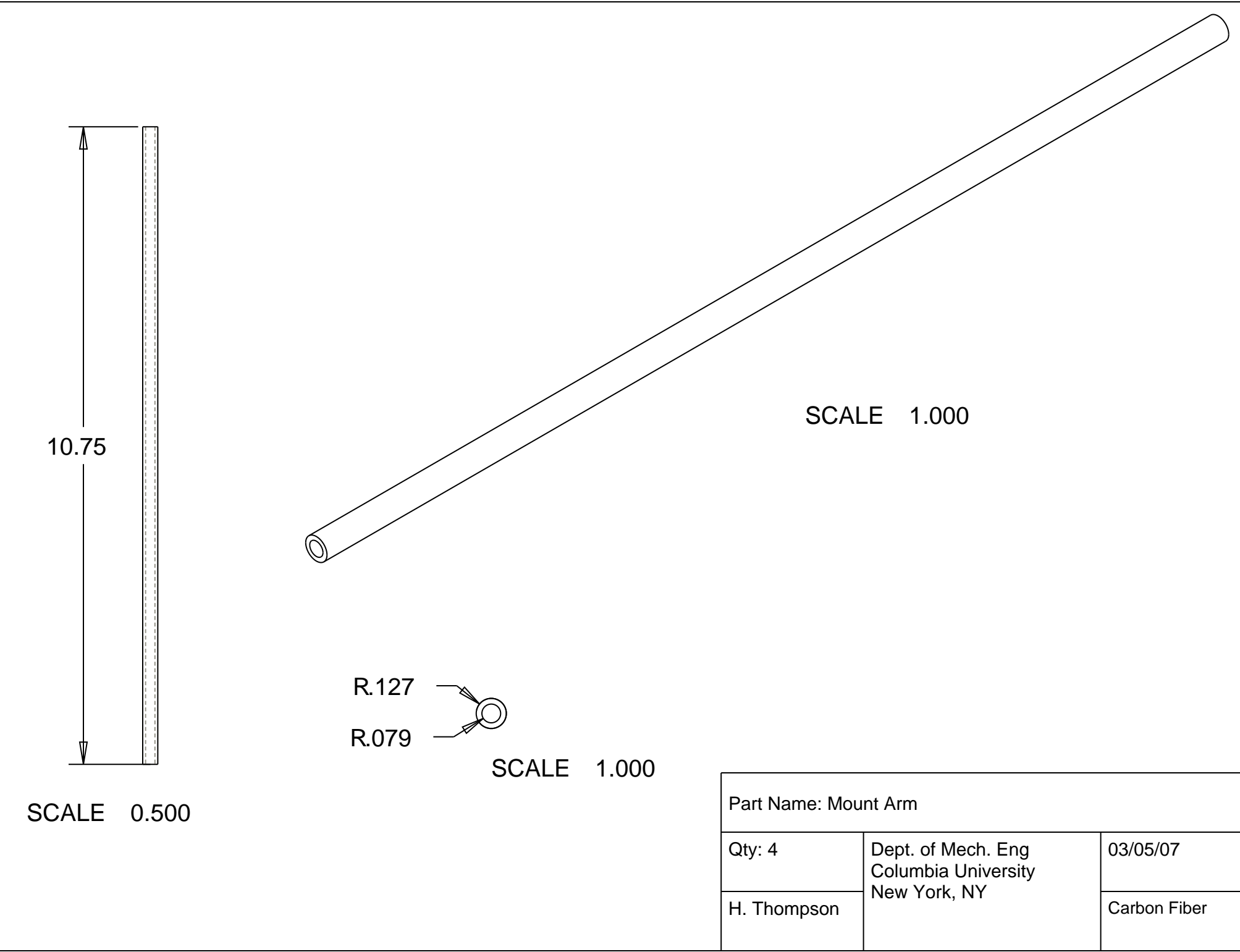
Qty. 1

Dept. of Mech. Eng.
Columbia University
New York, NY

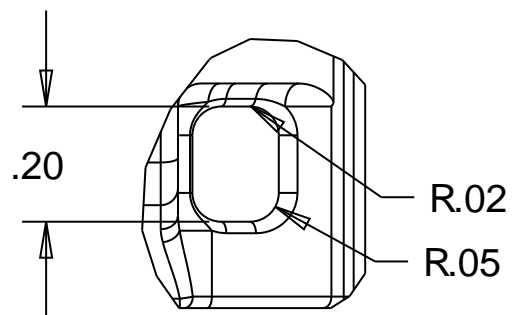
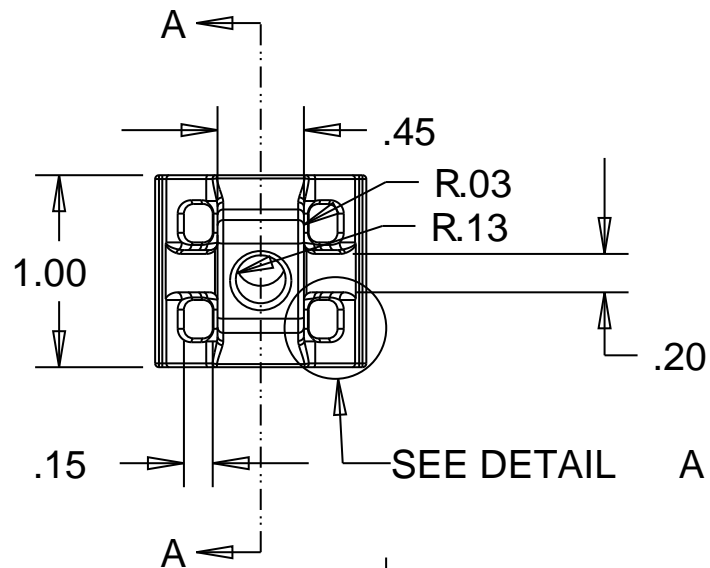
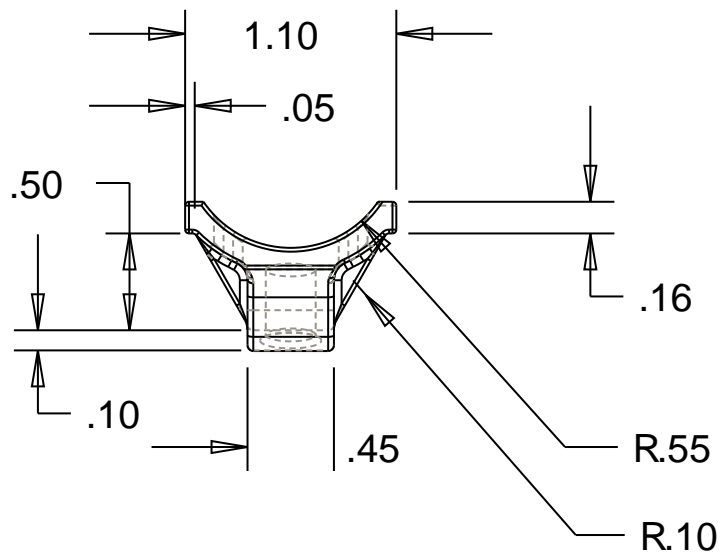
04/15/07

L. Montejo

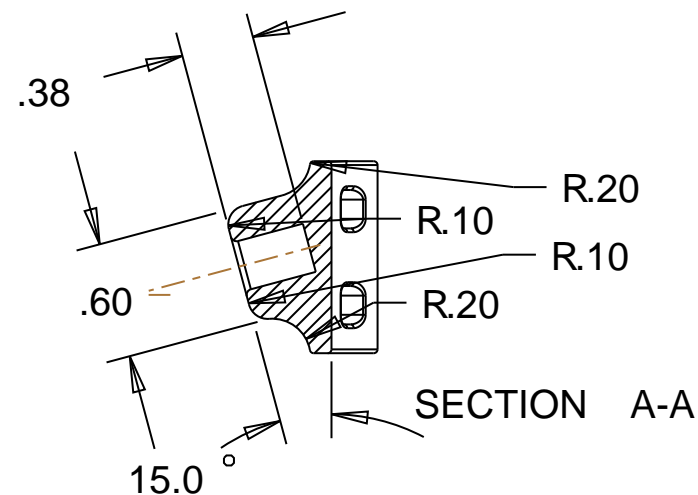
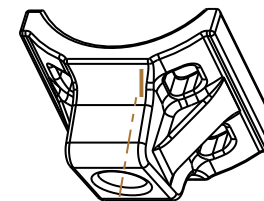
Q-UAV



Part Name: Mount Arm		
Qty: 4	Dept. of Mech. Eng Columbia University New York, NY	03/05/07
H. Thompson		Carbon Fiber



DETAIL A
SCALE 3.000



Part Name: Motor Mount

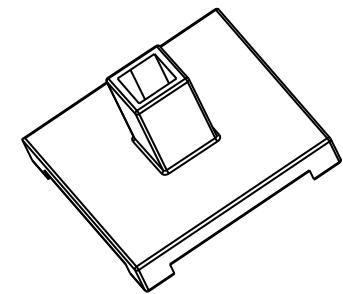
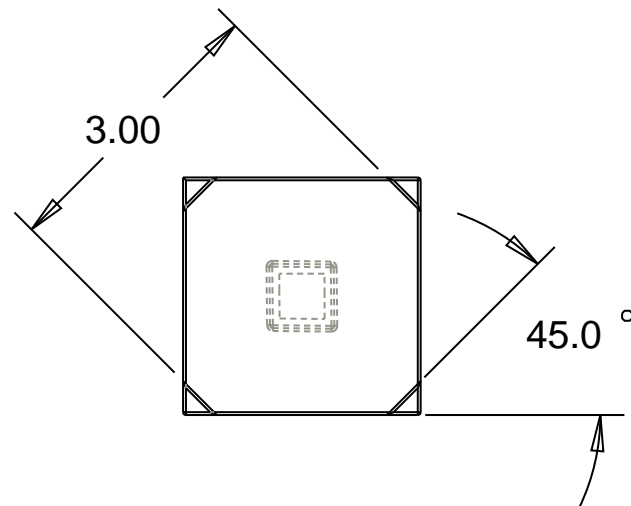
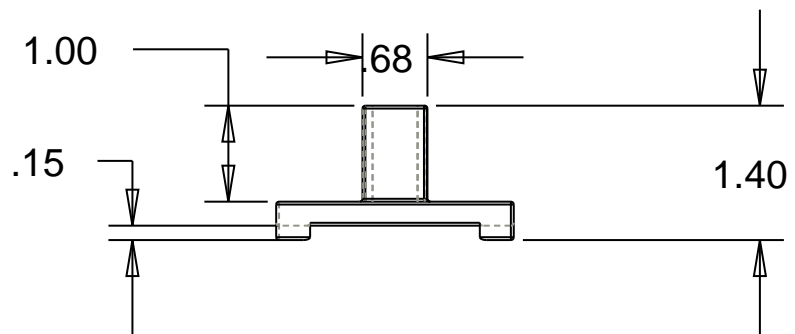
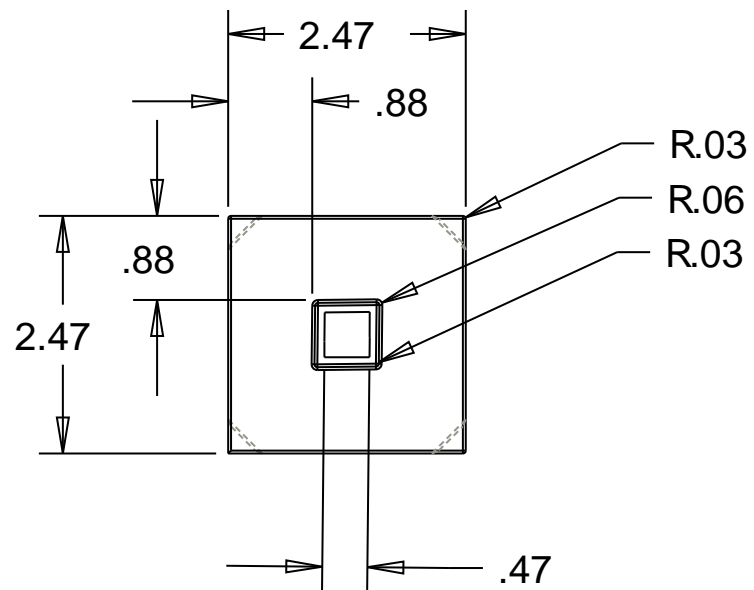
Qty. 4

Dept. of Mech.
Eng. Columbia
University New
York, NY

04/11/07

H.Thompson

ABS



Part Name: Circuit Mount

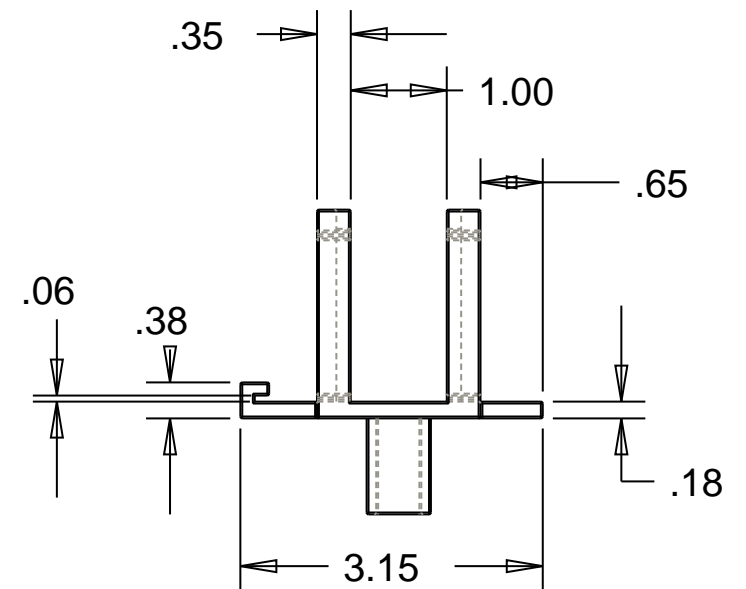
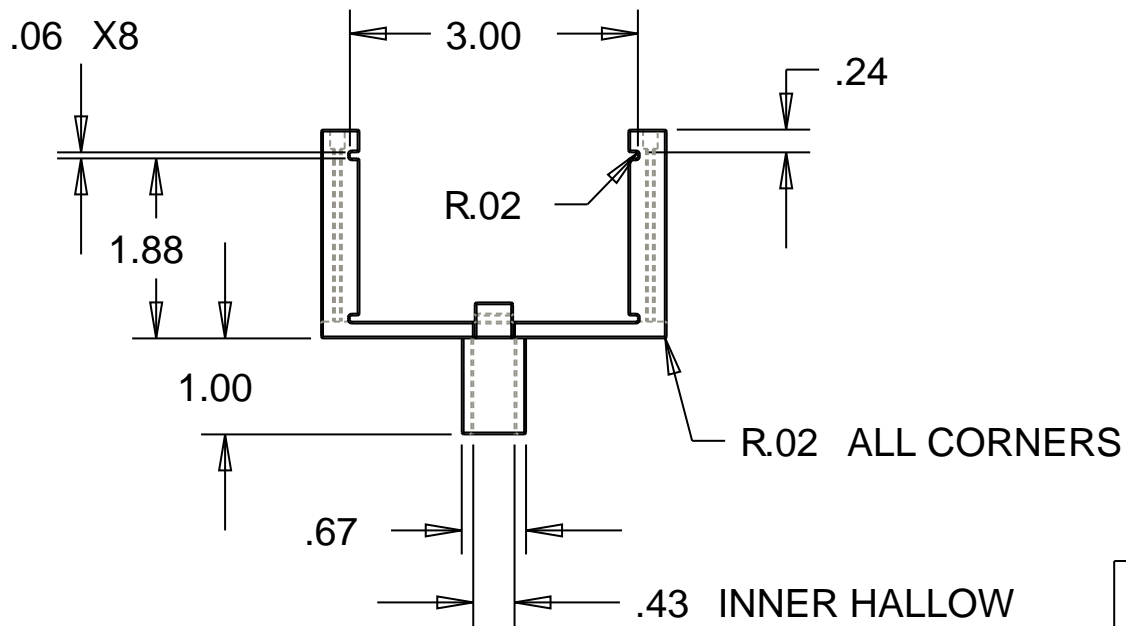
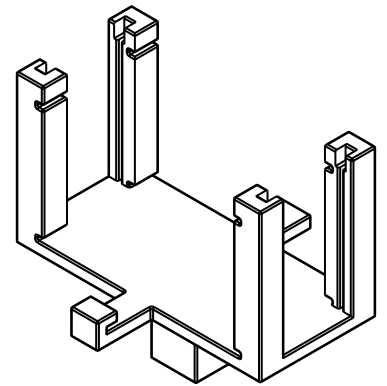
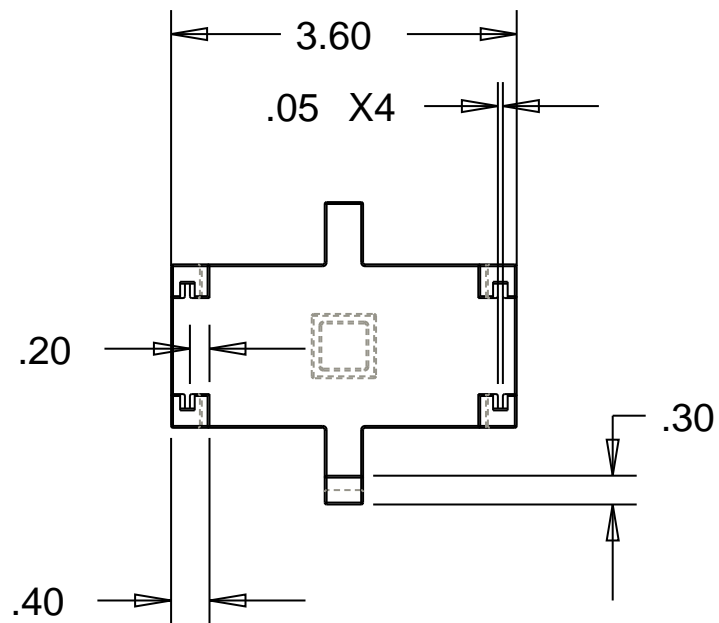
Qty. 1

Dept. of Mech.
Eng. Columbia
University New
York, NY

04/16/07

L. Montejo

ABS



PART NAME: CUSTOM CIRCUIT MOUNT

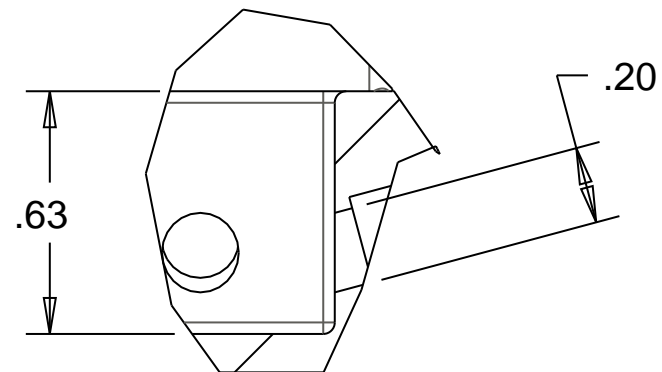
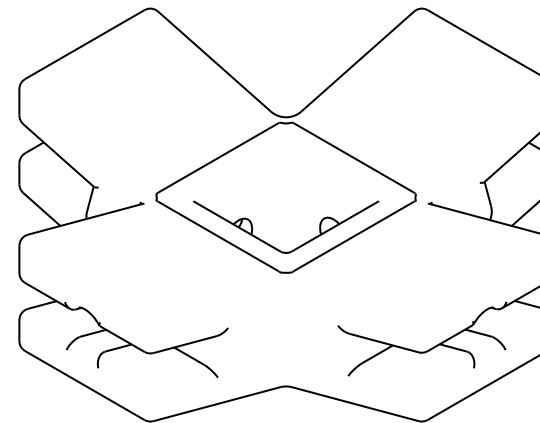
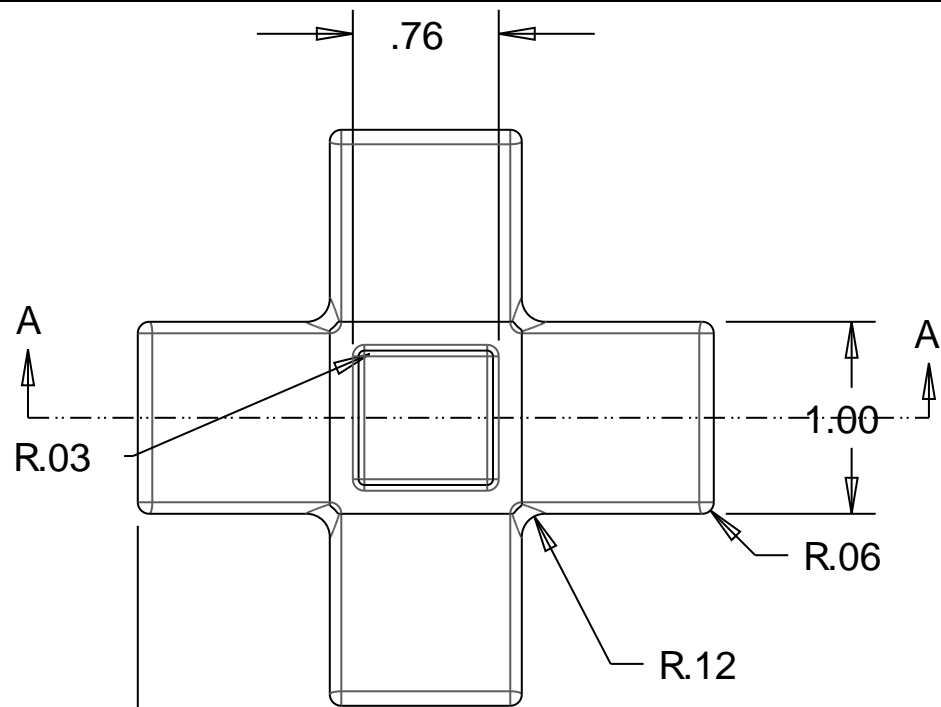
QTY. 1

Dept. of Mech.
Eng. Columbia
University New
York, NY

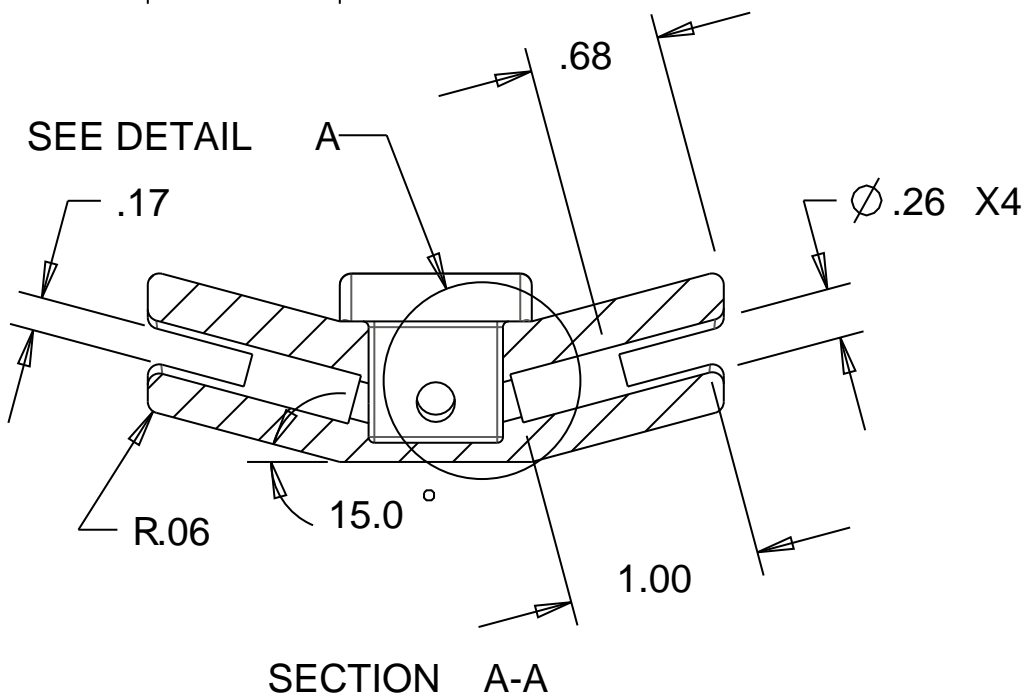
04/27/07

H.THOMPSON

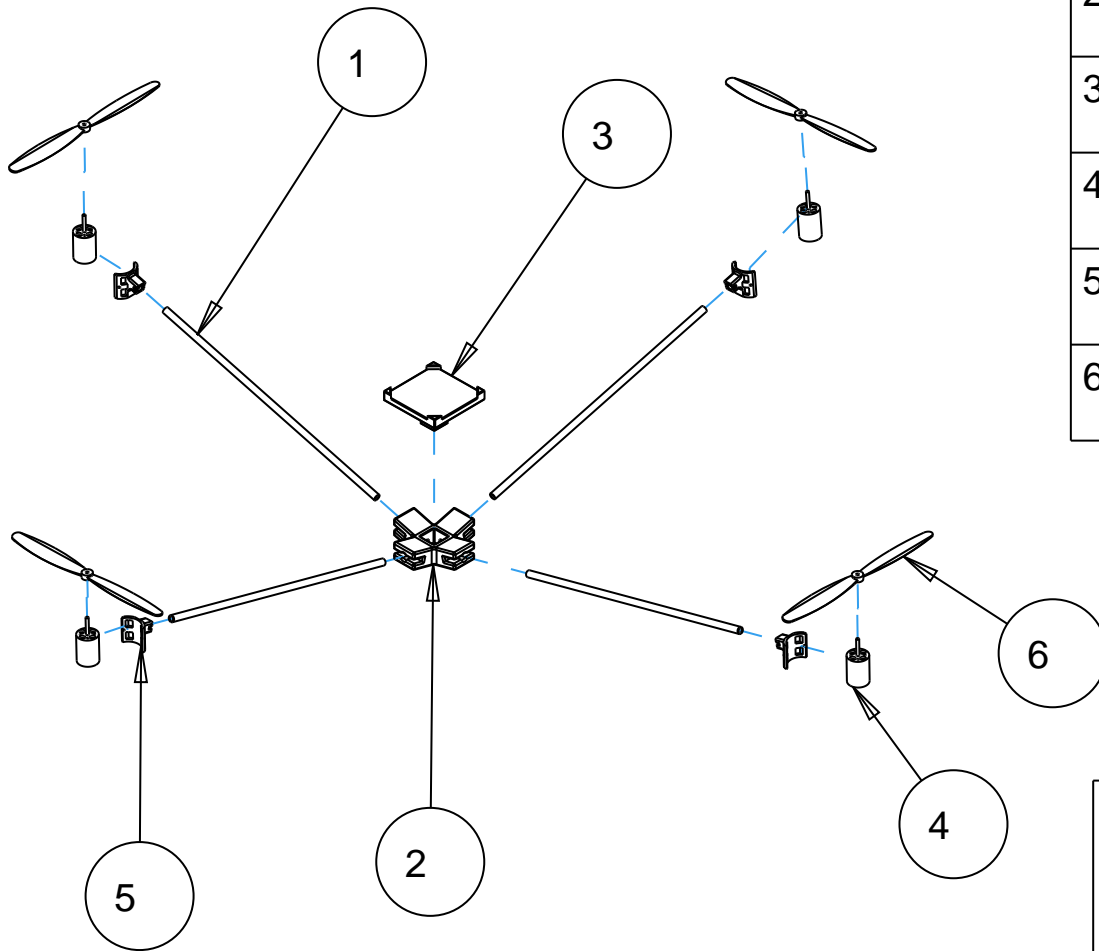
ABS



DETAIL A
SCALE 2.000
Gen Tol ± 0.01 inches



Part Name: Central Hub		
Qty: 1	Dept. of Mech. Eng. Columbia University New York, NY	04/05/07
H. Thompson		ABS



SCALE 0.150

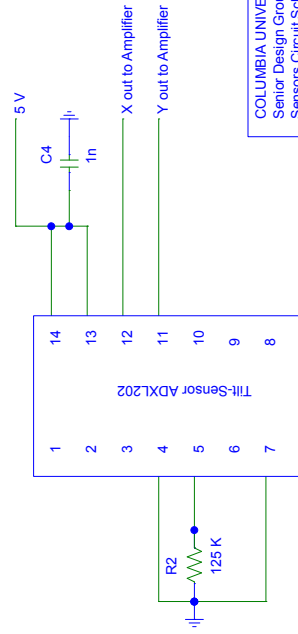
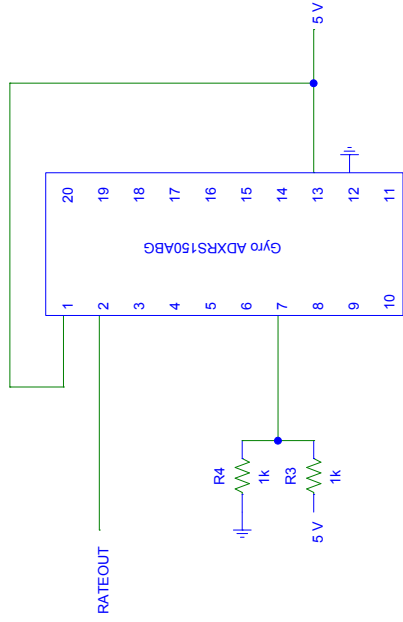
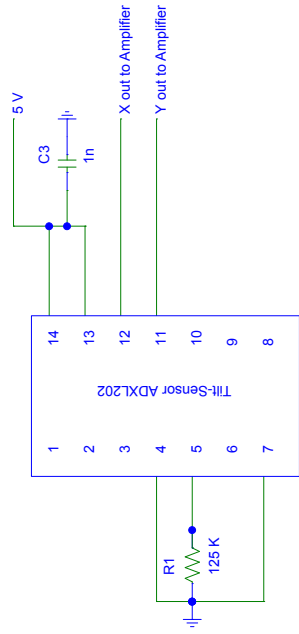
Index	Part Name	Qty
1	Mount Arm	4
2	Central Hub	1
3	Circuit Mount	1
4	Motor	4
5	Motor Mount	4
6	Prop	4

Scale: 0.150	Vehicle Assembly
	Department of Mechanical Engineering Columbia University New York, NY
04/30/07	Drawing No. : 1
View A	H.Thompson

A.4: Circuit Diagrams

1

2



B

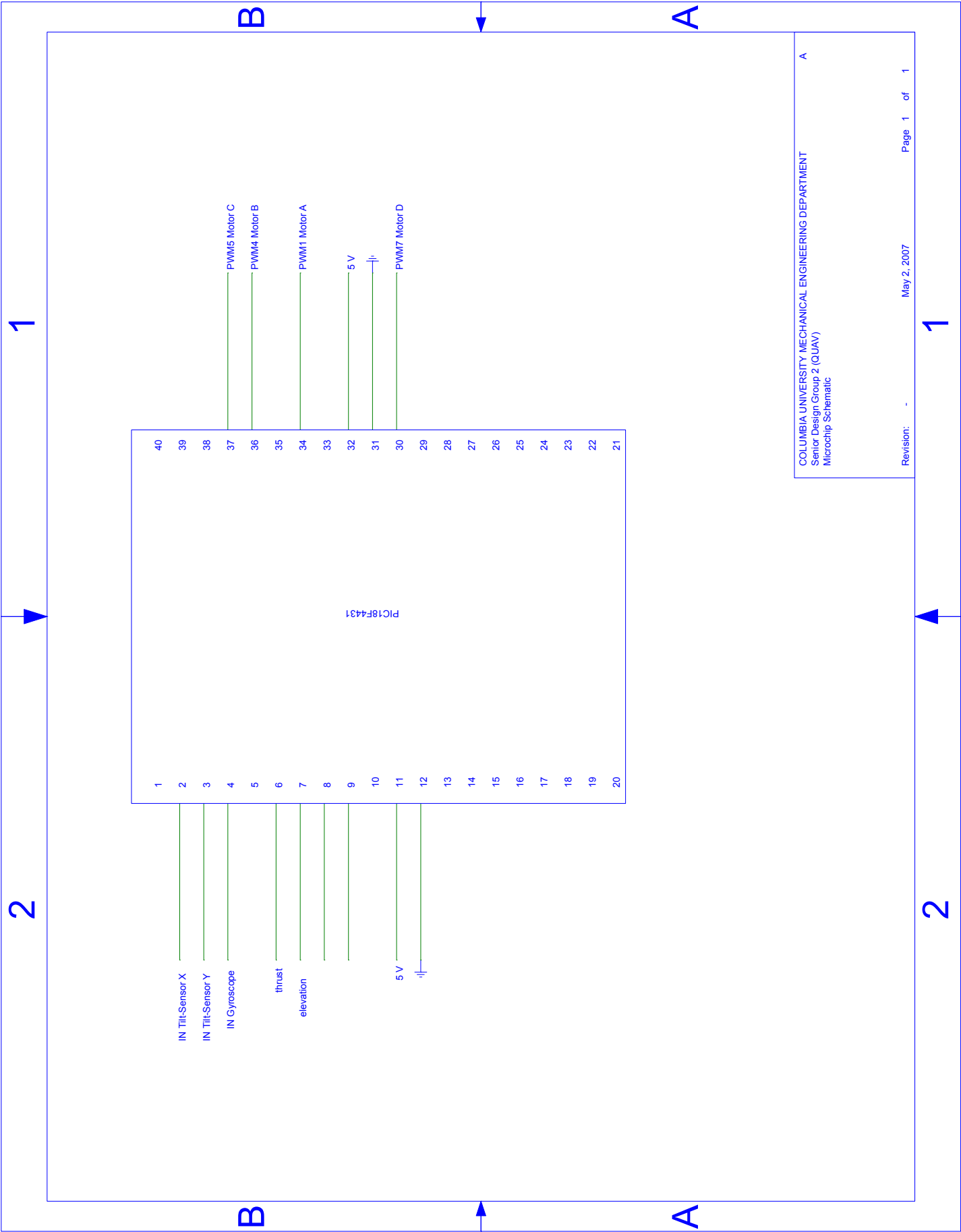
A

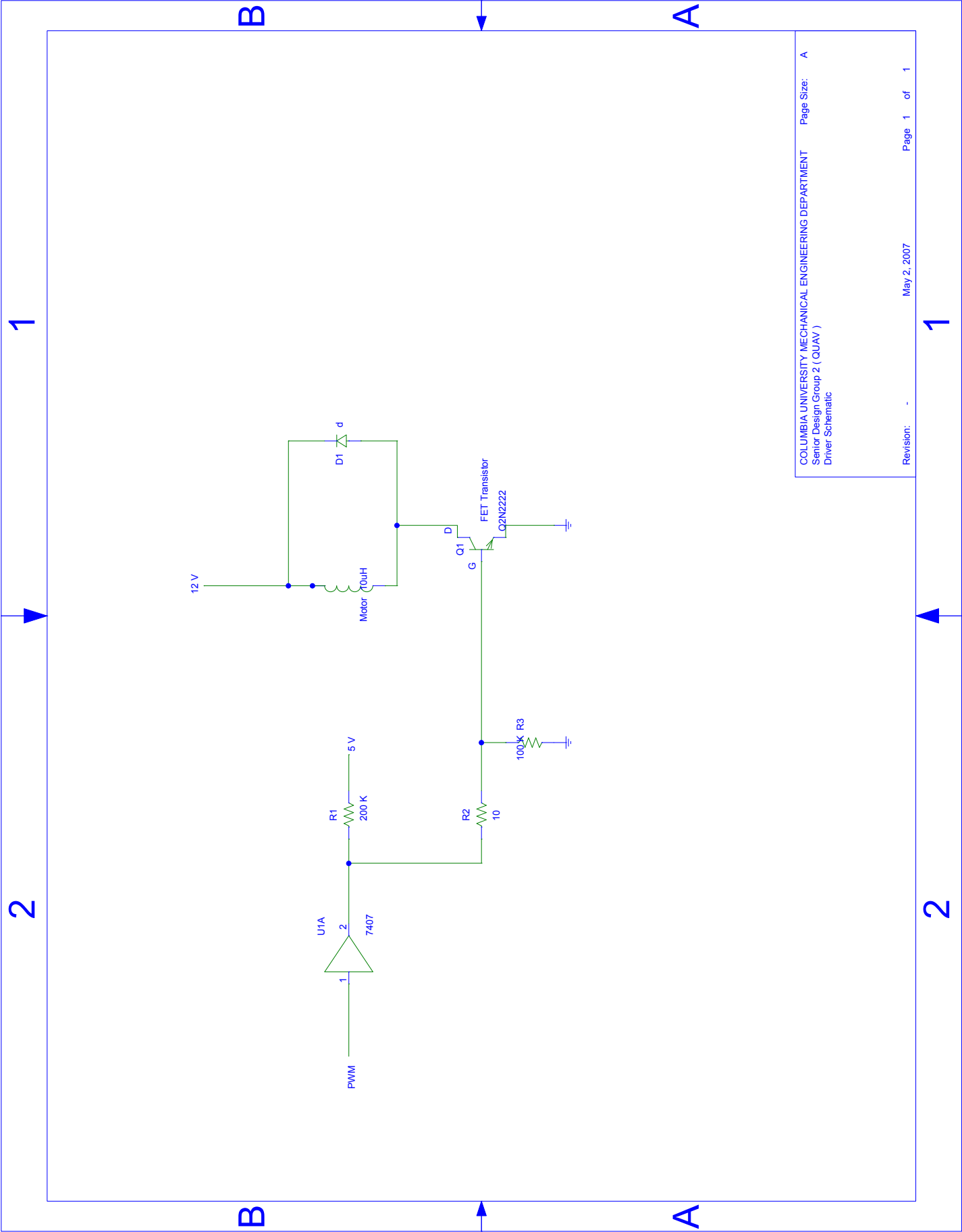
B

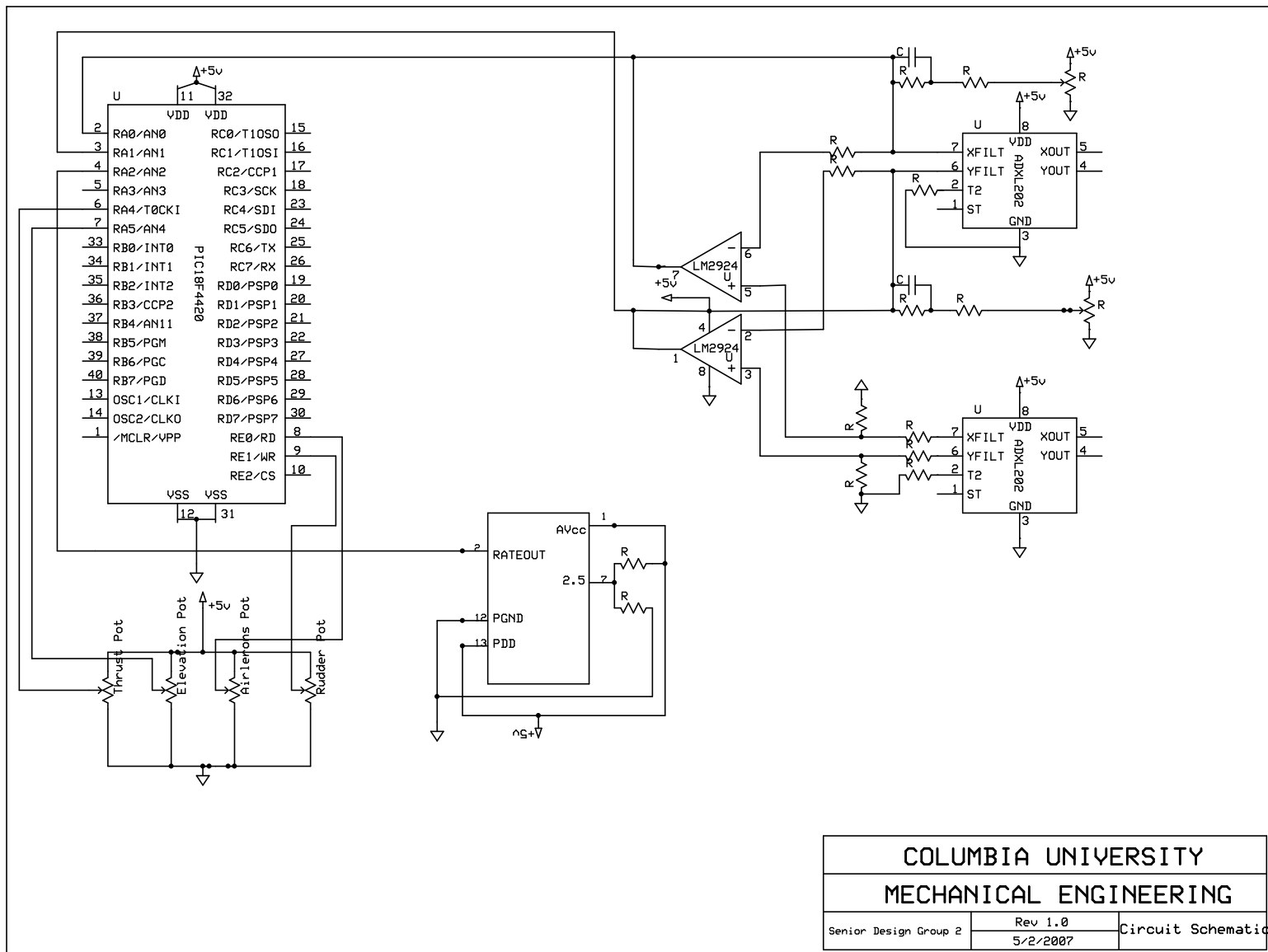
A

1

2







A.5: Experimental Results: Motor thrust plots

Table A.5.1: GWS EM400/GWS EPS400C/GWS Triblade

Motor/Gearbox/Propeller Combination: GWS EM400/GWS EPS400C/GWS Tri-rotor			
Voltage [Volts]	Current [Amps]	Lift [grams]	RPM [rev/minute]
3.0	0	38	1120
4.5	0	78	1350
5.5	3	108	1600
6.0	3	128	1760
6.5	5	143	1880
7.0	6	168	2000
7.5	6	193	2160
8	6.5	218	2280

Table A.5.2: GWS EM400/GWS EPS400C/APC 9x6 Propeller

Motor/Gearbox/Propeller Combination: GWS EM400/GWS EPS400C/APC 9x6 Propeller			
Voltage [Volts]	Current [Amps]	Lift [grams]	RPM [rev/minute]
3.0	3	35	1580
4.5	3	90	1760
5.5	3	125	1980
6.0	5	140	2080
6.5	5	160	2260
7.0	6	185	2420
7.5	6.5	205	2540
8.0	7	225	2680

Table A.5.3: GWS EM400/GWS EPS400C/GWS EP0843 (orange blades)

Motor/Gearbox/Propeller Combination: GWS EM400/GWS EPS400C/GWS EP0843 (orange blades)			
Voltage [Volts]	Current [Amps]	Lift [grams]	RPM [rev/minute]
3.0	0.0	35	1640
4.5	0.0	72	1760
5.5	0.5	102	2140
6.0	1.5	112	2260
6.5	2.0	132	2420
7.0	2.0	152	2580
7.5	3.0	172	2740
8.0	4.0	192	2860

A.6: MATLAB Simulation Code

```
%Driver file

clear all;
close all;

global C g L m J1 J2 J3 F1 F2 F3 F4 nF1 nF2 nF3 nF4 %define global variables

W=0.1;%width of hub
D=W;%depth of hub
H=0.02;%height of hub
Q=0.09;%mass of motor
P=0.1;%mass of hub
r=0.012;%radius of motor
h=0.036;%height of motor
R=0.25;%distance from motor to hub

%constants used in the equations of motion
C=1;
g=9.8;
L=0.2;
m=0.5;
J1=(1/3)*Q*(3*r^2+h^2)+4*Q*R^2+(1/12)*P*(H^2+D^2)
J2=J1
J3=2*Q*r^2+4*Q*R^2+(1/12)*P*(W^2+D^2)

tf = 10; %length of time the ODE's will be solved for

%following is the IC's code, but it wont be use here because there is code
%for it later down, but its kept for refenrece in future use
% x10=100; x20=0; x30=100; x40=0; x50=0; x60=0;
% x70=0; x80=0; x90=0; x100=0; x110=0; x120=0;
% f1a=1.4;f2a=1.4;f3a=1.42;f4a=1.42;

%command code for the craft to go on a trapezoidal path
for j=1:5
if j==1 f1a=1.1;f2a=1;f3a=0.9;f4a=1; end
if j==2 f1a=1.1;f2a=1.1;f3a=0.9;f4a=0.9; end
if j==3 f1a=1;f2a=1.1;f3a=1;f4a=0.9; end
if j==4 f1a=0.9;f2a=0.9;f3a=1.1;f4a=1.1; end
if j==5 f1a=0.9;f2a=0.9;f3a=1.1;f4a=1.1; end
```

```
%magnitude of the thrust force
```

```
F1=1.3*f1a;
```

```
F2=1.3*f2a;
```

```
F3=1.3*f3a;
```

```
F4=1.3*f4a;
```

```
%Initial Conditions actually used
```

```
if j==1
```

```
    x10=0; x20=0; x30=0; x40=0; x50=0; x60=0;
```

```
    x70=0; x80=0; x90=0; x100=0; x110=0; x120=0;
```

```
end
```

```
if j>1
```

```
    x10=x1(tf+1,1); x20=0; x30=x1(tf+1,3); x40=0; x50=0; x60=0;
```

```
    x70=0; x80=0; x90=0; x100=0; x110=0; x120=0;
```

```
end
```

```
%File of ODE's
```

```
function dx=equations(t,x)
```

```
global C g L m J1 J2 J3 F1 F2 F3 F4 nF1 nF2 nF3 nF4
```

```
dx=zeros(12,1);
```

```
N=1;
```

```
dx(1) = x(2);%x dot
```

```
dx(2) = ((F1+F2+F3+F4)/m)*(cos(x(11))*sin(x(7))*cos(x(9))+sin(x(11))*sin(x(9)));%x double dot
```

```
dx(3) = x(4);%y dot
```

```
dx(4) = ((F1+F2+F3+F4)/m)*(sin(x(11))*sin(x(7))*cos(x(9))-cos(x(11))*sin(x(9)));%y double dot
```

```
dx(5) = x(6);%z dot
```

```
dx(6) = ((F1+F2+F3+F4)/m)*(cos(x(7))*cos(x(9)))-g;%z double dot
```

```
dx(7) = x(8);%theta dot
```

```
dx(8) = ((-F1-F2+F3+F4)/J1)*L;%theta double dot
```

```
dx(9) = x(10);%psi dot
```

```
dx(10) = ((-F1+F2+F3-F4)/J2)*L;%psi double dot
```

```
dx(11) = x(12);%phi dot
```

```
dx(12) = C*(F1-F2+F3-F4)/J3;%phi double dot
```

A.7: C-Code for Proportional Control System

```
#include <p18f4431.h>
```

```
/******  
pic184431_commented_final.c
```

This program allows for the control of a quadrotor helicopter.
In particular, the program controls four DC motors via four individual PWM modules.

The PWM output values for the four motors are determined based on A/D readings from four inputs and four sensors. The inputs characterize the desired motion of the craft (thrust, elevator, aileron, rudder controls). The sensors characterize the actual motion of the craft (pitch, roll, yaw, altitude). The code contains equations which account for these 8 inputs to calculate for the appropriate outputs to each motor.

The program makes use of 8 of the 9 available A/D inputs on the pic18f4431, as well as the odd PWM output pins (1,3,5,7) for four unique PWM outputs.

```
*****/
```

```
// variable declarations
```

```
signed char      kThrust, kElevator, kAilerons, kRudder;  
signed char      kTiltX, kTiltY, kGyroZ, kSonicZ;  
signed int       inTiltX=0, inTiltY=0, inGyroZ=0, inSonicZ=0;  
signed int       inThrust=0, inElevator=0, inAilerons=0, inRudder=0;  
signed int       outMotorA=0, outMotorB=0, outMotorC=0, outMotorD=0;  
signed int       testAH=0, testAL=0, testBH=0, testBL=0, testCH=0, testCL=0, testDH=0, testDL=0;  
unsigned int     samplecount, limit;
```

```
signed int       inTiltXsum=0, inTiltYsum=0, inGyroZsum=0, inSonicZsum=0, i, samples;
```

```
void DelayFn(unsigned int cnt)           // time delay function  
{  
    while (cnt>=1)  
    {  
        cnt--;  
    }  
}
```

```
void AtoDinit(void)  
{  
    RCONbits.IPEN = 0;           // disables priority levels on interrupts  
    INTCONbits.GIE = 1;         // enables global interrupts
```

```

    INTCONbits.PEIE = 1;           // enables peripheral interrupts
    PIE1bits.ADIE = 1;           // enables A/D interrupt
    PIR1bits.ADIF = 0;           // ensures that A/D interrupt flag is cleared

    ADCON0 = 0b00010100;         // enables A/D Module and sets A/D to single shot, multi-
channel, sequential mode, with GO/DONE bit ADCON0<1> cleared, (waiting to be set to start A/D conversions)
    ADCON1 = 0b00011000;         // sets internal Vref+ and Vref-, buffer on
    ADCON2 = 0b10110000;         // sets A/D acquisition time to 12Tad, clock sets acq time to
Fosc/2, sets right-justified A/D result
    ADCON3 = 0b10000000;         // interrupt generated on 4th word written to buffer, triggers
disabled
    ANSEL0 = 0b11110111;         // sets only analog inputs AN0-AN2,AN4-AN7 as enabled,
other pins remain digital I/O status (if ultrasonic used, AN3 needs to be enabled)
    ANSEL1 = 0b00000000;         // see above-- disables analog input AN8 to maintain digital
I/O status

    ADCON0bits.ADON = 1;         // turns on A/D module
    DelayFn(500);                // module requires delay before sampling can be done
}

void AtoDreadSensors(void)       // channel select => read sensors (AN0,AN1,AN2,AN3)
{
    ADCHS = 0b00000000;         // sets analog input pins to their respective sampling groups to
read sensors
    DelayFn(10);                // time delay necessary when switching channels of the A/D
    ADCON0bits.GO_DONE = 1;      // starts A/D sampling

    while (ADCON0bits.GO_DONE==1) {} // A/D still converting

    testAH = ADRESH;
    testAL = ADRESL;
    inTiltXsum = inTiltXsum + 256*testAH + testAL; // stores reading from tilt sensor X output
    testBH = ADRESH;
    testBL = ADRESL;
    inTiltYsum = inTiltYsum + 256*testBH + testBL; // stores reading from tilt sensor Y output
    testCH = ADRESH;
    testCL = ADRESL;
    inGyroZsum = inGyroZsum + 256*testCH + testCL; // stores reading from yaw rate gyro
    testDH = ADRESH;
    testDL = ADRESL;
    inSonicZsum = inSonicZsum + 256*testDH + testDL; // stores reading from sonic sensor
}

void AtoDreadInputs(void)        // channel select => read inputs (AN4,AN5,AN6,AN7)
{
    ADCHS = 0b01010101;         // sets analog input pins to their respective sampling groups to
read inputs
    DelayFn(10);                // time delay necessary when switching channels of the A/D
    ADCON0bits.GO_DONE = 1;      // starts A/D sampling

    while (ADCON0bits.GO_DONE==1) {} // A/D still converting

    testAH = ADRESH;
    testAL = ADRESL;

```

```

    inThrust = 256*testAH + testAL;           // stores reading from thrust input
    testBH = ADRESH;
    testBL = ADRESL;
    inElevator = 256*testBH + testBL; // stores reading from elevator input
    testCH = ADRESH;
    testCL = ADRESL;
    inAilerons = 256*testCH + testCL; // stores reading from aileron input
    testDH = ADRESH;
    testDL = ADRESL;
    inRudder = 256*testDH + testDL;          // stores reading from rudder input
}

void    PWMinit(void)
{
    PTCON0 = 0b00000000;           // 1:1 Prescale & 1:1 Postscale
    PTCON1 = 0b00000000;           // Time Base Timer options PTCON1<7>=PTEN = time base
enable bit [makes it go]
    PWMCON0 = 0b01111111;           // enables all PWM pins (RB1,RB3,RB5,RD7) and
sets independent mode
    PWMCON1 = 0b00000000;           // other pwm inits

    PTPERH = 0x00;                  // sets PTPER to FFh (10-bit PWM resolution, 39.6 kHz
frequency)
    PTPERL = 0xFF;
}

void    PWMout(void)
{
    //for the tilt+gyro inputs, the equilibrium value is 2.5V/5V-- as such, a no-change tilt/gyro reading should
be read as .5(10-bit resolution) = 512.
    //for the inElevator, inAilerons, inRudder inputs, equilibrium value is 2.5V/5V, for the inThrust, the
equilibrium value is 0V/5V.

    outMotorA = (kThrust*(inThrust) - (kElevator*(inElevator-512) + kTiltX*(inTiltX-512)) -
(kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512)));

    if (inThrust < limit)
    {
        outMotorA=0;
        PDC0L = 0b00000000;
        PDC0H = 0b00000000;
    }

    else if (inThrust > limit && outMotorA < limit)
    {
        if (limit > 255)
        {
            PDC0L = limit;
            PDC0H = 0b00000001;
        }
        else if (limit < 255)
        {
            PDC0L = limit;
            PDC0H = 0b00000000;
        }
    }
}

```

```

    }
}
else if (inThrust > limit && outMotorA > 1023)
{
    PDC0L = 0b11111111;
    PDC0H = 0b00000011;
}
else if (inThrust > limit && outMotorA > limit && outMotorA < 256)
{
    PDC0L = outMotorA;
    PDC0H = 0b00000000;
}
else if (inThrust > limit && outMotorA > 255 && outMotorA < 512)
{
    PDC0L = outMotorA - 256;
    PDC0H = 0b00000001;
}
else if (inThrust > limit && outMotorA > 511 && outMotorA < 768)
{
    PDC0L = outMotorA - 512;
    PDC0H = 0b00000010;
}
else if (inThrust > limit && outMotorA > 767 && outMotorA < 1024)
{
    PDC0L = outMotorA - 768;
    PDC0H = 0b00000011;
}

outMotorB = (kThrust*(inThrust) + (kAilerons*(inAilerons-512) + kTiltY*(inTiltY-512)) +
(kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512)));

if (inThrust < limit)
{
    outMotorB=0;
    PDC1L = 0b00000000;
    PDC1H = 0b00000000;
}

else if (inThrust > limit && outMotorB < limit)
{
    if (limit > 255)
    {
        PDC1L = limit;
        PDC1H = 0b00000001;
    }
    else if (limit < 255)
    {
        PDC1L = limit;
        PDC1H = 0b00000000;
    }
}
else if (inThrust > limit && outMotorB > 1023)
{
    PDC1L = 0b11111111;
    PDC1H = 0b00000011;
}

```

```

    }
    else if (inThrust > limit && outMotorB > limit && outMotorB < 256)
    {
        PDC1L = outMotorB;
        PDC1H = 0b00000000;
    }
    else if (inThrust > limit && outMotorB > 255 && outMotorB < 512)
    {
        PDC1L = outMotorB - 256;
        PDC1H = 0b00000001;
    }
    else if (inThrust > limit && outMotorB > 511 && outMotorB < 768)
    {
        PDC1L = outMotorB - 512;
        PDC1H = 0b00000010;
    }
    else if (inThrust > limit && outMotorB > 767 && outMotorB < 1024)
    {
        PDC1L = outMotorB - 768;
        PDC1H = 0b00000011;
    }

    outMotorC = (kThrust*(inThrust) - (kAilerons*(inAilerons-512) + kTiltY*(inTiltY-512)) +
(kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512)));

    if (inThrust < limit)
    {
        outMotorC=0;
        PDC2L = 0b00000000;
        PDC2H = 0b00000000;
    }

    else if (inThrust > limit && outMotorC < limit)
    {
        if (limit > 255)
        {
            PDC2L = limit;
            PDC2H = 0b00000001;
        }
        else if (limit < 255)
        {
            PDC2L = limit;
            PDC2H = 0b00000000;
        }
    }
    else if (inThrust > limit && outMotorC > 1023)
    {
        PDC2L = 0b11111111;
        PDC2H = 0b00000011;
    }
    else if (inThrust > limit && outMotorC > limit && outMotorC < 256)
    {
        PDC2L = outMotorC;
        PDC2H = 0b00000000;
    }

```

```

else if (inThrust > limit && outMotorC > 255 && outMotorC < 512)
{
    PDC2L = outMotorC - 256;
    PDC2H = 0b00000001;
}
else if (inThrust > limit && outMotorC > 511 && outMotorC < 768)
{
    PDC2L = outMotorC - 512;
    PDC2H = 0b00000010;
}
else if (inThrust > limit && outMotorC > 767 && outMotorC < 1024)
{
    PDC2L = outMotorC - 768;
    PDC2H = 0b00000011;
}

outMotorD = (kThrust*(inThrust) + (kElevator*(inElevator-512) - kTiltX*(inTiltX-512)) -
(kRudder*(inRudder-512) + kGyroZ*(inGyroZ-512)));

if (inThrust < limit)
{
    outMotorD=0;
    PDC3L = 0b00000000;
    PDC3H = 0b00000000;
}

else if (inThrust > limit && outMotorD < limit)
{
    if (limit > 255)
    {
        PDC3L = limit;
        PDC3H = 0b00000001;
    }
    else if (limit < 255)
    {
        PDC3L = limit;
        PDC3H = 0b00000000;
    }
}
else if (inThrust > limit && outMotorD > 1023)
{
    PDC3L = 0b11111111;
    PDC3H = 0b00000011;
}
else if (inThrust > limit && outMotorD > limit && outMotorD < 256)
{
    PDC3L = outMotorD;
    PDC3H = 0b00000000;
}
else if (inThrust > limit && outMotorD > 255 && outMotorD < 512)
{
    PDC3L = outMotorD - 256;
    PDC3H = 0b00000001;
}
else if (inThrust > limit && outMotorD > 511 && outMotorD < 768)

```


[illegible]