Design Optimization of a Quad-Rotor Capable of Autonomous Flight

A Major Qualifying Project Report
Submitted to the Faculty
of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the
Degree of Bachelor of Science
In Aerospace Engineering

Submitted by:

Mark Dupuis
mdupuis@wpi.edu

Jonathan Gibbons
jonored@WPI.EDU

Maximillian Hobson-Dupont
ltdan@wpi.edu

Alex Knight
koohash@wpi.edu

Artem Lepilov
artem@wpi.edu

Michael Monfreda
monfreda@wpi.edu

George Mungai
gmungai@wpi.edu

April 24, 2008

Advisors:
Michael A. Demetriou
mdemetri@wpi.edu

David J. Olinger
olinger@wpi.edu

Abstract

An autonomous quad-rotor is an aerial helicopter with four horizontal rotors designed in a square configuration capable of locating lost or jeopardized victims, gathering military intelligence, and surveillance. The project team designed a miniaturized quad-rotor able to determine its own attitude through an onboard sensor system. A computer program using formulated control equations and an onboard processing system enables the quad-rotor to fly to a pre-determined position while correcting its attitude, which results in steady level autonomous flight.

Acknowledgments

Contents

List of Figures

List of Tables

Chapter 1

Introduction

An autonomous quad-rotor is an unmanned helicopter with four horizontal fixed rotors designed in a square, symmetric configuration, with the front and back rotors rotating counter-clockwise and the side rotors clockwise. Quad-rotors were conceptualized by Dr. George de Bothezat and Ivan Jerome in the 1920s. Early designs never achieved a hovering height greater than five meters [1]. Few applicable uses arose for the quad-rotor over the years. Recently, they have become a very popular consumer product as a remote control (RC) helicopter. The quad-rotor is such a proficient RC helicopter due to the fixed rotor design, which consequently reduces the chance of failure in comparison with the actuating rotor design of single rotor RC helicopters. In addition, having four motors versus one motor allows for an increased potential thrust. Therefore the quad-rotor has impressive maneuvering capabilities that make it a great indoor and/or outdoor RC helicopter. An effective autonomous miniature quad-rotor would have many applications ranging from locating fire and avalanche victims to military surveillance and intelligence. The advantages to developing an autonomous miniature quad-rotor are numerous and make it a worthy research topic.

Developing an autonomous miniature quad-rotor yielded many difficulties. Initially there was research into different types of motors, airfoils, batteries and control schemes in an effort to develop task specifications to accomplish flight. The motors and airfoil must produce enough thrust to lift all the components into flight and maneuver. The batteries need to have a large enough capacity to allow for sufficient flight time. Algorithms for

the dynamic control equations used to correct the attitude of the quad-rotor and create steady level flight have to be formulated. At the same time, components for movement, position, and a navigation system must be chosen, calibrated and implemented into a circuit. Also, a hub and frame must be constructed to hold the motors and all the electronic components. Once the final construction of the quad-rotor has been completed, a program to use the algorithms will be created to allow the computer to fly the quad-rotor. There are many facets in constructing a quad-rotor producing many challenges in integrating all the components.

The goal is to design a miniaturized autonomous quad-rotor capable of taking off from a landed position, maneuvering to a point determined by a programmed iRobot®, hovering, and landing at its take off point. The team designed a quad-rotor based on the RC helicopter configurations, except the provided quad-rotor's thrust capabilities needed improvement as well as the battery operating time to accomplish the goals. The team designed and machined a frame to hold all the components of the quad-rotor in a balanced and efficient manner. Algorithms were developed to map the dynamics of the miniature quad-rotor to implement them in a program capable of flying the helicopter. Finally, the team programed an iRobot® to communicate its position and the flight plan of the quad-rotor.

1.1 History

The idea of a quad-rotor aircraft has existed since early in the 20th century. Throughout the 20th century very few distinct rotor-craft designs had been developed. The earliest workable designs for a quad-rotor were developed by George DeBothezat, Etienne Oemichen and D.H. Kaplan.

Oemichen's quad-rotor design is the earliest mention of a complete four-rotor hovering vehicle in history. Oemichen's first design in 1920 failed in the initial attempt to become airborne, thereby requiring Oemichen to add additional lifting power and stability of a helium-filled balloon. After a number of recalculations and redesigns, Oemichen was able to come up with a design that actually was capable of lift off and even established world helicopter flight records of the time, remaining airborne for up to 14 minutes at a time by 1923.
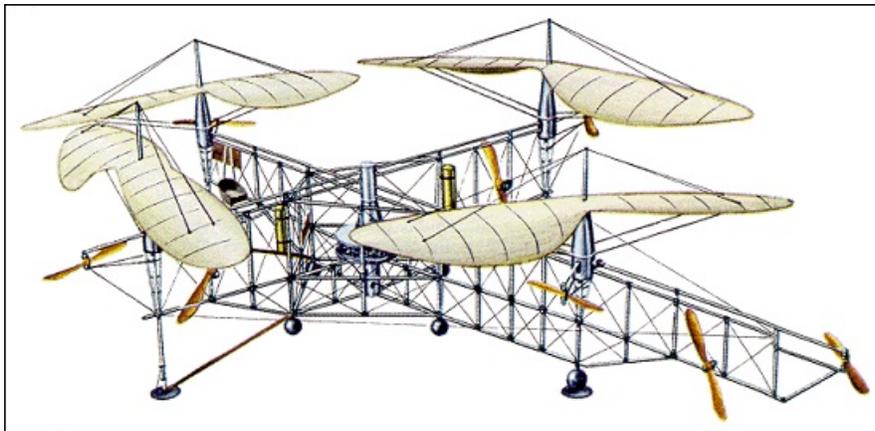
Figure 1.1: DeBothezats quad-rotor Design, 1922

DeBothezat's design was created for a 1921 contract with the US Air Corps. After working on his design for over 2 years, he was able to develop a fairly capable helicopter, which was able to take on a payload of up to 3 people in addition to the pilot. His design was deemed underpowered, unresponsive and susceptible to reliability problems. In addition, instead of the calculated 100 meters cruising altitude, his craft was only capable of reaching a height of roughly 5m.

3
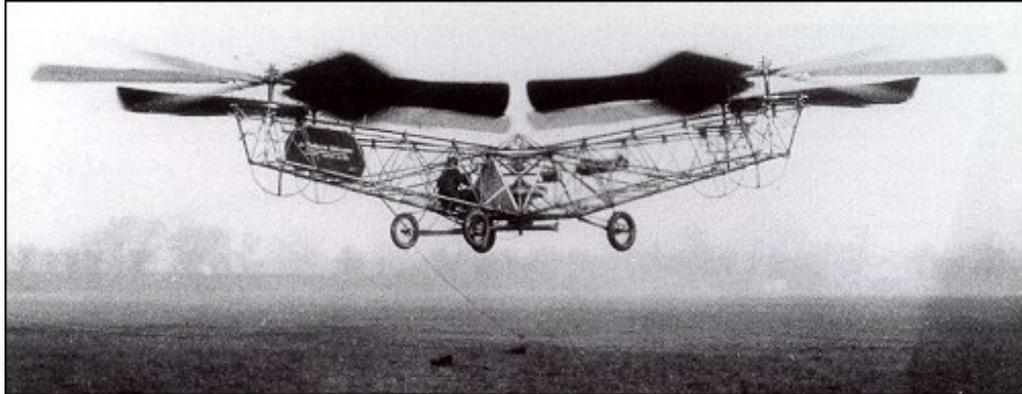
Figure 1.2: Oeminchen No.2 Quad-rotor Design, 1922

The early designs were propelled by additional rotors located somewhere on the rear or the front of the craft, perpendicular to the main rotors. Thus, they are not true quad-rotor designs. It was not until the mid-1950s that a true quad-rotor helicopter flew, which was designed by Marc Adam Kaplan. Kaplan's quad-rotor design, Convertawings Model "A" quad-rotor, is arguably the most successful of the early designs of the rotor-craft. The prototype first flew in 1956, and did so with great success. The 2200 pounds craft was able to hover and maneuver using its two 90 horsepower motors, each capable of driving all four rotors in backup mode. Control in this case did not call for additional rotors on the sides of the craft, but was obtained by varying thrust between rotors. This also was the first quad-rotor design that was able to fly forward successfully.

Despite these early proofs-of-concepts, people saw little practical use for quad-rotors. They simply were not competitive with the performance specifications (speed, payload, range, etc.) of more conventional aircrafts. No production contracts were awarded and interest in quad-rotors waned.

Recently however, there has been renewed interest in quad-rotors from hobbyists. This old idea is returning with great potential after RC quad-rotors have shown that there is definite potential in the quad-rotor platform.
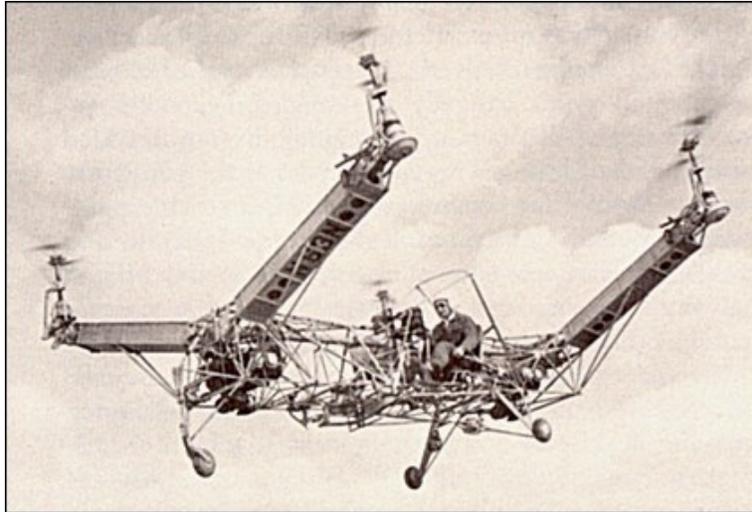
Figure 1.3: Convertawings Model A quad-rotor Design, 1956

1.2 Potential Applications for an Innovative Base Design

The goal of this project is to design and build a quad-rotor that is controlled completely autonomously. Written programs have the craft turn on, lift off, hover, translate, and land, or any other combination of these tasks. More difficult tasks, such as a positioning or navigational system require additional components and more advanced programming, but are easily adapted to this project's quad-rotor. The features of an unmanned, lightweight, quick and small craft capable of carrying a small load of up to a kilogram has many applications, from recreational and exploratory missions, to rescue or military objectives.

The quad-rotor in this project is based on the design of many RC helicopters already available on the internet and in stores. The top brand name models, such as Draganflyer®, start around $800. The budget of this project allows for a quad-rotor worth about $1200. However, the cost of the additional electrical sensors and processors needed for autonomous flight make Draganflyer® models too expensive to be a viable solution. Therefore the quad-rotor in this project is completely built from scratch and has been optimized to be light and powerful. It is as fast and powerful as any currently available top end models, and has a longer flight time. A skilled hobbyist

can build this quad-rotor from the schematics that are available throughout this report. There are diagrams of the circuit board that can be used to reproduce the quad-rotor in this project. The board is designed in a manner where the on board components can be easily swapped or upgraded.

A practical use for the autonomous quad-rotor would be to explore rough terrain that can not be easily accessed by a person. Photos and aerial videos could be acquired of volcanoes, glaciers or caverns that are susceptible to collapse and otherwise inaccessible. If a person is trapped in a mine and cannot be reached because of the danger of the mine shaft collapsing, a quad-rotor could locate the trapped miners and bring them food and water. If there is a missing person on a mountain, heat sensors could be integrated onto the board to help find them. Any search and rescue mission that covers a large area, such as the open ocean, could use a large group of the craft communicating wirelessly with the rescue team to increase the chances of rescue considerably. Its capabilities in smaller areas with obstructions, such as indoors, are more limited, but with enough development and time it would be possible to redesign for specific environments.

It is also possible to have multiple crafts linked together and controlled from a single computer. This makes it possible to use the same method of controlling for all the crafts from the computer. In fact, an entire heterogeneous system could be created, consisting of many quad-rotors all working in unison for a specific mission. In a military setting, the craft could be further optimized with more expensive materials and equipped with much more sophisticated sensors. This would allow missions requiring very precise maneuvers in tight settings, or long range missions employing many crafts at once to be performed easily and without risk to any troops.

Chapter 2

Dynamics

2.1 Quad-rotor Physics

A quad-rotor is an aerial vehicle with four rotors arranged in a symmetric, square configuration around a central hub, which houses the battery and processing components. While flying, the quad-rotor is positioned with a propeller in front and back. Moving counter-clockwise from the front propeller, let F_i be the force of each rotor i for $i = 1, 2, 3, 4$ such that F_1 and F_3 rotate counter-clockwise and F_2 and F_4 rotate clockwise. To perform a stationary hover, all four rotors rotate at the same rate and the total thrust of the craft is equal to its mass, m. Since both pairs of rotors spin in opposite directions, the net torque on the craft due to drag from the propellers is zero.

To create yaw movement in a counter-clockwise direction, F_1 and F_3 are sped up inversely proportional to F_2 and F_4. As a result, the net torque on the craft is negative via the right hand rule and it will yaw while remaining at the same altitude. If F_1 and F_3 do not increase proportionally to F_2 and F_4 decreasing, the craft will move in the z-direction because the net thrust will no longer equal zero. To yaw clockwise F_2 and F_4 must increase proportionally to F_1 and F_3's decrease. A roll to the left is accomplished by decreasing the speed of F_2 while increasing F_4. Again, F_2 must decrease at the same rate that F_4 increases to maintain zero net torque. Increasing F_2 and decreasing F_4 results in rolling to the right. Pitching forward and back is done in the same fashion as rolling, but with F_1 and F_3. The principle for maintaining an equal rate of change for the two
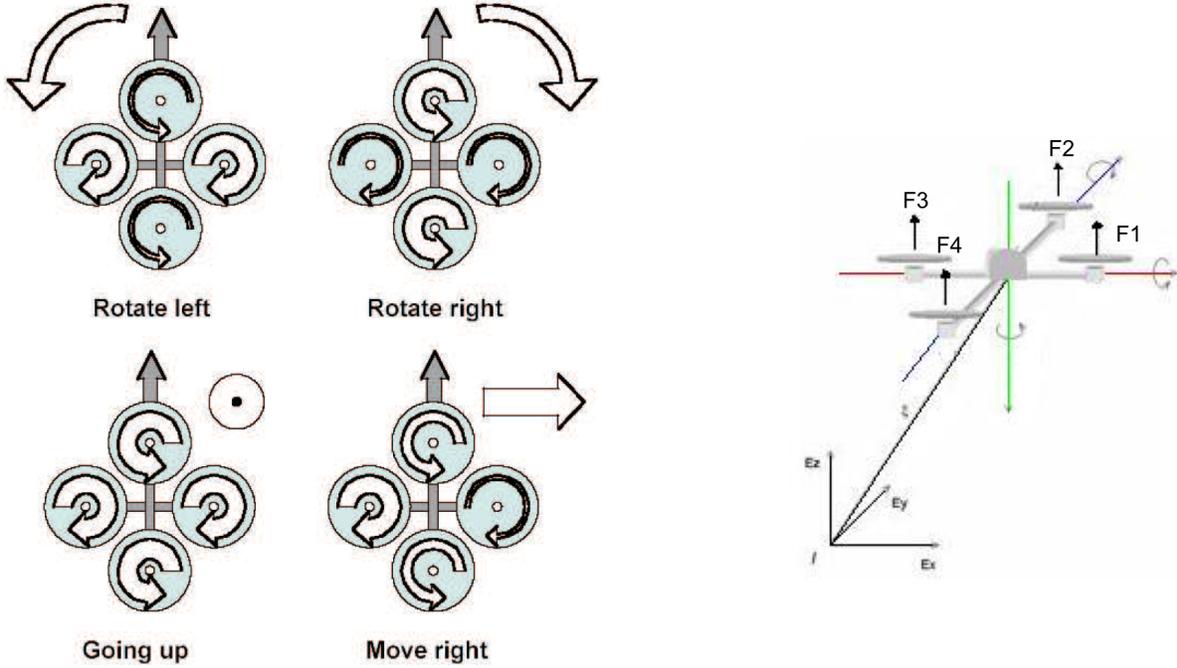
Figure 2.1: The dynamics of the quad-rotor [13].

opposing rotors is how the translation of the craft is determined. Due to either a pitch or a roll, the lift force is displaced in the x and y planes by angles θ and ϕ respectively, resulting in a horizontal force component that will translate the craft. The altitude of the quad-rotor is altered by changing the rate of all rotors collectively by the same amount.

The thrust from each rotor exists only in the z-direction with respect to the body frame. The total thrust can be represented by $u = F_1 + F_2 + F_3 + F_4$ and F_i is the force of rotor i.

2.2 Equations of Motion

The derivation for the dynamics follows the Lagrangian concept for mechanics [1]

$$L = T_{trans} + T_{rot} - U = \frac{m}{2}\dot{\xi}^T\dot{\xi} + \frac{1}{2}\dot{\eta}^T J\dot{\eta} - mgz \tag{2.1}$$

where

$$\xi = (x, y, z), \qquad \eta = (\psi, \theta\phi),$$
$$J = W^T I W.$$

J is the moment of inertia matrix calculated in the inertial coordinate system after being transformed from the body frame, I, by matrix W

$$W \;=\; \begin{bmatrix} -\sin(\theta) & 0 & 1 \\ \cos(\theta)\sin(\psi) & \cos(\psi) & 0 \\ \cos(\theta)\cos(\psi) & -\sin(\psi) & 0 \end{bmatrix}.$$

To derive translational dynamics, the thrust vector in the body fixed frame must be transformed to an inertial frame via the matrix rotation, R. The forces in the inertial and body frame are given by

$$F_i = RF \qquad F_b = [0, 0, u]^T$$

where

$$R = \begin{bmatrix} \cos(\theta)\cos(\psi) & \sin(\theta)\sin(\psi) & -\sin(\theta) \\ \cos(\psi)\sin(\theta)\sin(\phi) - \sin(\psi)\cos(\phi) & \sin(\psi)\sin(\theta)\sin(\phi) + \cos(\psi)\cos(\phi) & \cos(\theta)\sin(\phi) \\ \cos(\psi)\sin(\theta)\cos(\phi) + \sin(\psi)\sin(\phi) & \sin(\psi)\sin(\theta)\cos(\phi) - \cos(\psi)\sin(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix}.$$

After the rotation matrix has been applied, the translational equations of motion with respect to an inertial frame are given by

$$\begin{aligned} m(\ddot{x}) &= -u\sin(\theta) \\ m(\ddot{y}) &= u\cos(\theta)\sin(\phi) \\ m(\ddot{z}) &= u\cos(\theta)\cos(\phi) - mg. \end{aligned} \qquad (2.2)$$

Following the Euler-Lagrange equations, the rotational dynamics include torques and coriolis terms. The coriolis term, $C(\eta, \dot{\eta})\dot{\eta}$, defines the gyroscopic effects on the system when the craft yaws. τ represents the vector of torques applied to the system.

$$C(\eta, \dot{\eta})\dot{\eta} \triangleq \dot{J}\dot{\eta} - \frac{\partial}{\partial \eta}(\dot{\eta}^T J \dot{\eta}) \qquad (2.3)$$

$$J\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} = \tau \tag{2.4}$$

$$\tau = J\tilde{\tau} + C(\eta, \dot{\eta})\dot{\eta} \tag{2.5}$$

It follows that

$$\ddot{\eta} = [\tilde{\tau}_\psi, \, \tilde{\tau}_\theta, \, \tilde{\tau}_\phi]^T = \tilde{\tau}.$$

Combined with Eq. (2.2) the translational and rotational dynamics can be summarized as

$$\begin{aligned}
\ddot{\phi} &= \tilde{\tau}_\phi \\
\ddot{\theta} &= \tilde{\tau}_\theta \\
\ddot{\psi} &= \tilde{\tau}_\psi.
\end{aligned} \tag{2.6}$$

2.3 Control Strategy

2.3.1 Altitude Control

To control the altitude of the quad-rotor, the system dynamics can be modeled to represent a mass damper spring (MDS) system. To do so, the control parameter p_1 is introduced such that

$$u = \frac{p_1 + mg}{\cos\theta\cos\phi}.$$

Introducing this term into the translational equation of motion for $\ddot{z}$ yields

$$m(\ddot{z}) = p_1.$$

The variable r_1 is then defined to model the MDS system, where

$$p_1 = -a_{z_1}\dot{z} - a_{z_2}(z - z_d) \tag{2.7}$$

such that

$$m\ddot{z} + a_{z_1}\dot{z} + a_{z_2}(z - z_d) = 0 \tag{2.8}$$

which is of the form

$$A\lambda^2 + B\lambda + C = \lambda^2 + 2\zeta\omega_n\lambda + \omega_n{}^2 = 0 \tag{2.9}$$

where

$$\zeta = \frac{a_{z_1}}{2\omega_n m}$$

$$\omega_n = \sqrt{\frac{a_{z_2}}{m}}.$$

Given a desired settling time of t_s, the quad-rotor can reach within 2% of its desired altitude, when

$$t_s = \frac{4}{\zeta \omega_n}.$$

From this, the altitude control parameters a_{z_1} and a_{z_2} can be calculated to match performance criteria. For example, based on the assigned parameters below, and assigning ζ to have a value of 0.7, a_{z_1} and a_{z_1} are calculated to be 0.96 and 0.7837, respectively.

Mass (m)	600 grams
Initial Height (z_0)	0 m
Initial Velocity ($\dot{z}_0$)	0 m/s
Desired Height (z_d)	2 m

Table 2.1: Control scheme.
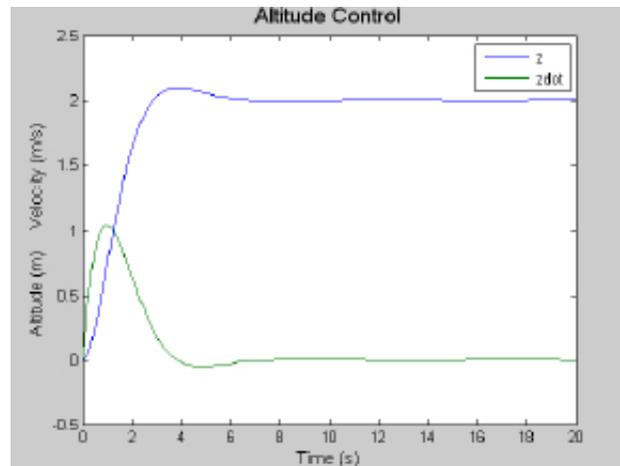
Figure 2.2: Altitude control system.

2.3.2 Yaw Control

Similarly, yaw control follows the same method outlined for altitude control except that there is no mass term in the derivation. It simplifies to

$$\ddot{\psi} = -a_{\psi_1}\dot{\psi} - a_{\psi_2}(\psi - \psi_d) = \tilde{\tau}_\psi \tag{2.10}$$

2.3.3 Roll and Pitch Control

It is necessary to control both roll and pitch to prevent the quad-rotor from exceeding critical angles that will cause it to flip. To achieve this goal, it is essential to limit the torque on the system such that angular accelerations remain low. These limitations appear in the boundaries of the saturation functions $\sigma_i(s)$, resulting in the nonlinear controllers below [1].

$$
\begin{aligned}
\tilde{\tau}_\phi &= -\sigma_{\phi_1}\left(\dot{\phi} + \sigma_{\phi_2}\left(\dot{\phi} + \phi + \sigma_{\phi_3}\left(\dot{\phi} + 2\phi + \frac{\dot{y}}{g} + \sigma_{\phi_4}\left(\dot{\phi} + 3\phi + 3\frac{\dot{y}}{g} + \frac{y - y_{desired}}{g}\right)\right)\right)\right) \\
\tilde{\tau}_\theta &= -\sigma_{\theta_1}\left(\dot{\theta} + \sigma_{\theta_2}\left(\dot{\theta} + \theta + \sigma_{\theta_3}\left(\dot{\theta} + 2\theta - \frac{\dot{x}}{g} + \sigma_{\theta_4}\left(\dot{\theta} + 3\theta - 3\frac{\dot{x}}{g} - \frac{x - x_{desired}}{g}\right)\right)\right)\right).
\end{aligned} \tag{2.11}
$$

2.4 PWM Motor Control

The vector term $\tilde{\tau}$ directly equates to the angular accelerations that are governed by the pitch, roll, and yaw controllers. Once computed, they are then transformed to output the torque terms seen in the τ vector by using the relationship in Eq. (2.6). Problems arise in the computation of the coriolis term, $C(\eta, \dot{\eta})\dot{\eta}$, due to limited onboard processing capacity. To remedy the problem this term is neglected, reducing the dynamics equation to $\tau = J\tilde{\tau}$. Analysis shows the magnitude of the difference is small enough to be assumed negligible and can be used throughout the onboard computations of the dynamics. Increasing the settling time for yaw displacement can reduce this error. This can be seen in Appendix B.1 through B.6 where the settling time was increased from five seconds to ten seconds and a reduction in the magnitude of the coriolis term is apparent.

For a desired maneuver to a specified x, y, z, or ψ position, the processor computes the necessary collective lift, u, and the three applied torques. This represents a system of four equations and four unknowns

$$
\begin{aligned}
u &= F_1 + F_2 + F_3 + F_4 & (2.12) \\
\tau_\psi &= c\left(-F - 1 + F_2 - F_3 + F_4\right) & (2.13) \\
\tau_\theta &= \left(F_4 - F_2\right) l & (2.14) \\
\tau_\psi &= \left(F_1 - F_3\right) l, & (2.15)
\end{aligned}
$$

where the constant l is the distance of the lever arm from the center of the hub to the center of the propellers which is 0.20353 meters.

After several tests were conducted to determine c from Eq. (2.14) the torque acting on the vehicle around the z-axis was determined. This was established by placing a scale vertically against a spar with one motor operating and varying the PWM signal. Another test was performed to identify the relationship between the PWM signal and the lift produced by the motors. Here the scale was placed horizontally under the quad-rotor and the PWM signal was varied to created the relationship. Comparing these two values provided a direct relationship between force and thrust. After a few simplifications, the relationships are as follows

$$
\begin{aligned}
\tau_\psi &= 0.0176\, PWM + 0.001 & (2.16) \\
F &= 0.004\, PWM + 0.001 & (2.17) \\
\tau_\psi &= 4.4\, F. & (2.18)
\end{aligned}
$$

The system can be written in the following matrix form,

$$
\begin{bmatrix} u \\ \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -c & c & -c & c \\ 0 & -l & 0 & l \\ l & 0 & -l & 0 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix}. \tag{2.19}
$$

Taking the inverse determines the amount of lift required of each motor for a particular maneuver such that the following holds true.

$$
\begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} = \begin{bmatrix} 0.25 & -0.6286 & 0 & 2.4556 \\ 0.25 & 0.6286 & -2.4556 & 0 \\ 0.25 & -0.6286 & 0 & -2.4556 \\ 0.25 & 0.6286 & 2.4566 & 0 \end{bmatrix} \begin{bmatrix} u \\ \tau_\psi \\ \tau_\theta \\ \tau_\phi \end{bmatrix} . \tag{2.20}
$$

Once the four required forces are calculated from solving Eq. (2.20), the signal sent to each motor in milliseconds can be calculated as

$$
\begin{bmatrix} PWM_1 \\ PWM_2 \\ PWM_3 \\ PWM_4 \end{bmatrix} = 0.004 \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} + 0.001. \tag{2.21}
$$

2.5 MATLAB Simulation

A simulation using MATLAB was conducted to evaluate the theoretical performance of the control theory governing the flight of the quad-rotor. The initial conditions and the four inputs, x, y, z, and ψ, were varied to simulate different maneuvers. The convention in Figure 2.3 contains the initial conditions followed by desired conditions in the order of x, y, z, and ψ. The corresponding PWM signals for the maneuvers were then computed and are shown.

Varying the constants of the pitch and roll controllers determined the time it took to reach the desired values for translational flight. For this first maneuver, the controls limited the translational velocity appropriately to reduce any unwanted excitement of the other motion variables that are held at zero. The PWM signals varied appropriately and converge to a constant value near the end of the simulation. Take note of the periods of positive and negative angles that correspond to acceleration and deceleration of the quad-rotor as it approaches its final value.

The second simulation demonstrates the altitude controller and the effect it has on the trajectory of the quad-rotor as it moves from ground to a one meter hover in Figure 2.4. The constants of the controller can be tweaked to change the dampening effect on the system and its responsiveness. Note that the all four PWM signals are
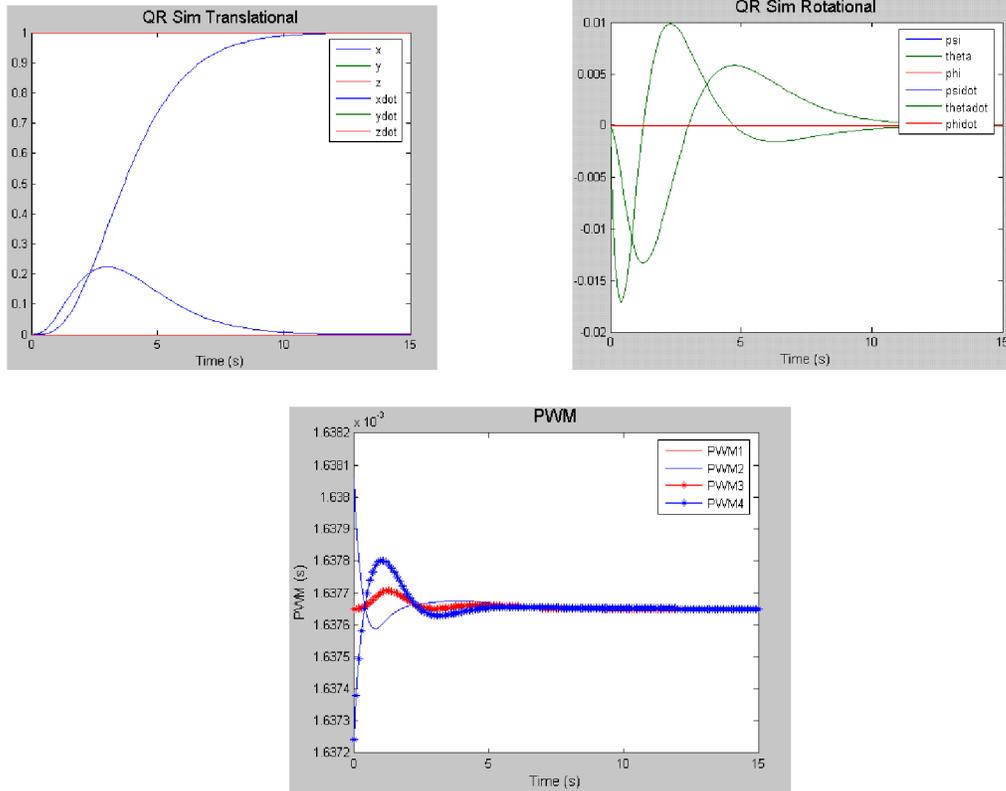
Figure 2.3: X-translation at a hover, (0, 0, 1, 0) to (1, 0, 1, 0)

identical throughout the duration of the simulation because there are no applied torques.

The final simulation is of the yaw controller and is similar in structure to the altitude controller. Likewise, its constants can be changed to alter the characteristic motion of the quad-rotor. The yaw angular velocity is also low enough as to reduce the excitement of the other unwanted motion variables. Also, the settling time is long enough as to reduce the error associated with disregarding the coriolis term as previously mentioned. Note again that the PWM signals converge upon the completion of the maneuver.
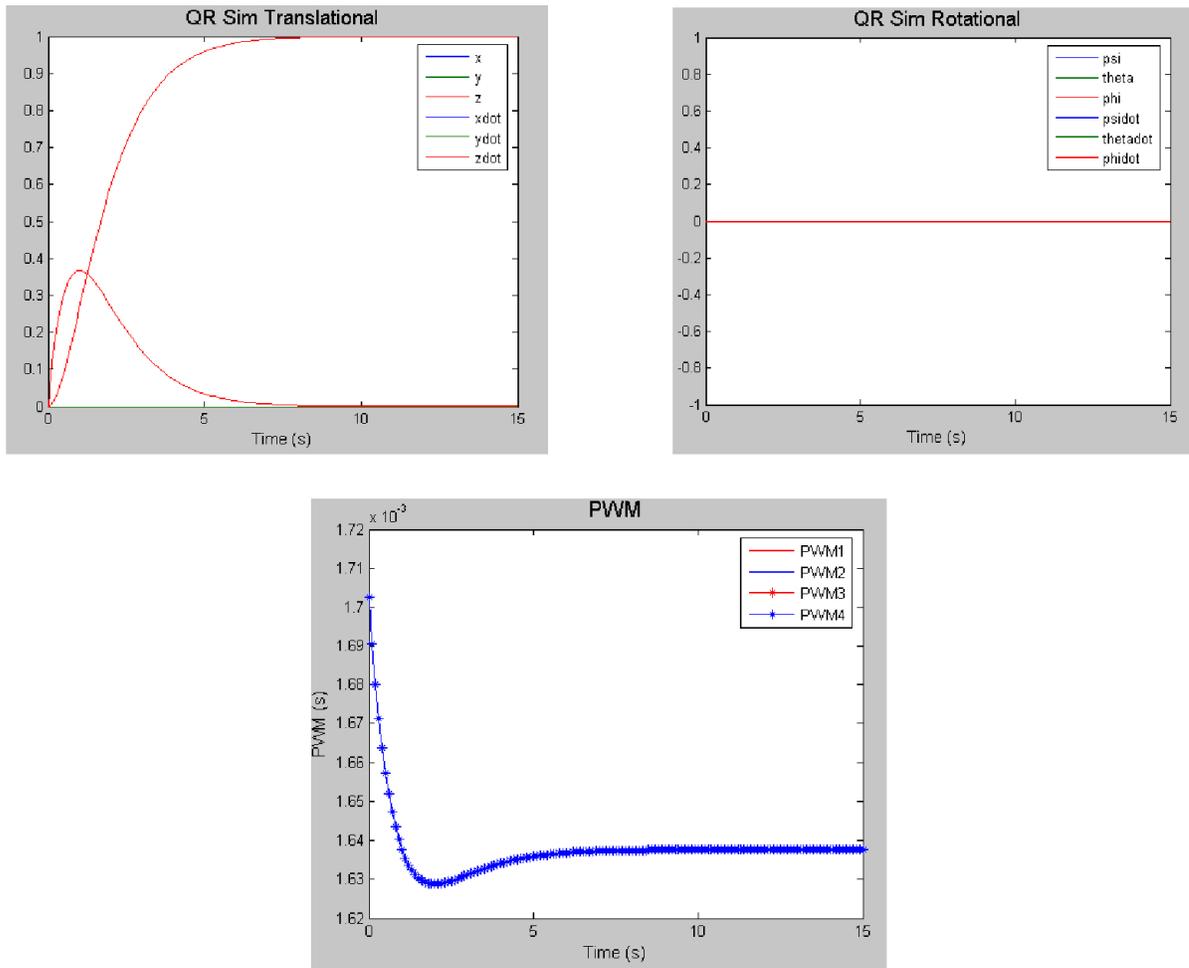
Figure 2.4: Ground to hover, (0, 0, 0, 0) to (0, 0, 1, 0)
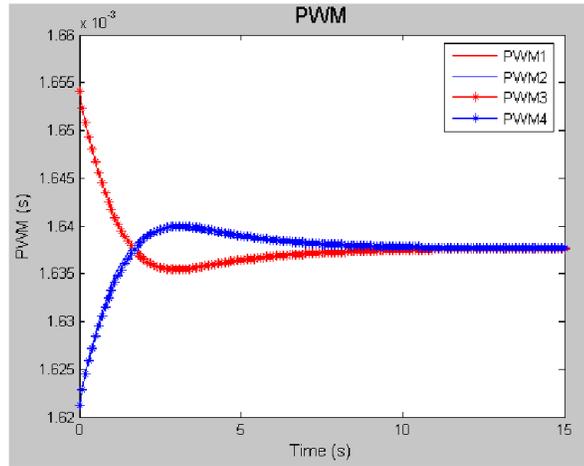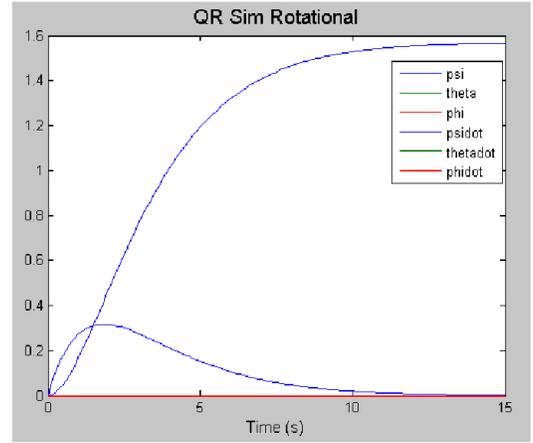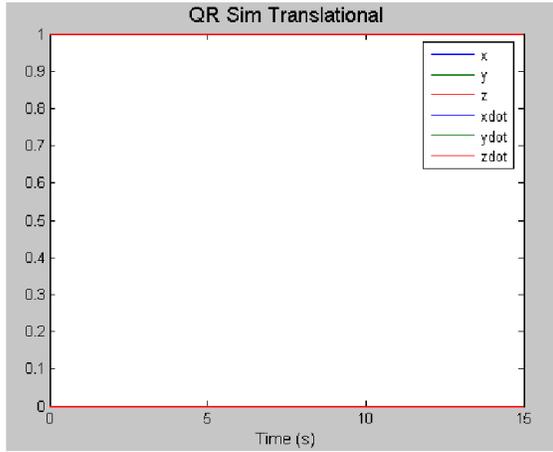
Figure 2.5: 90 ° yaw at a hover (0, 0, 1, 0) to (0, 0, 1, $\frac{pi}{2}$)

Chapter 3

Design of Quadrotor

3.1   The Silverlit RC X-UFO

The autonomous quad-rotor designed during this project is an improvement on the design of a fully functional

remote controlled quad-rotor called the Silverlit RC X-UFO, which was donated to the project at its inception. In

terms of quality and price the Silverlit quad-rotor is a low end model of the commercially available quad-rotors.

The X-UFO's manufacturer's suggested retail price is $232.99. which is low compared with the Draganflyer® V

Ti price of $1049.95, which is considered to be the best hobbyist quad-rotor commercially available. After a

close examination of the X-UFO, it can be seen that the lower price is reflected in the quality of its components.

These components are sufficient for the Silverlit to fly, but are insufficient for the requirements of this project.

Inefficient brushed motors for propulsion do not provide sufficient thrust required to lift additional electrical

components. The bulky, mechanical gyro used for stabilization, seen in Figure 3.1, fails when the quad-rotor is

tilted at more than $30\,^{\circ}$. The low-performance nickel-metal hydride (NiMH) battery with a low capacity of 350

mAh is inadequate for this project. The estimated energy requirements of the final quad-rotor demands a battery

of at least 3500 mAh, which can be achieved with Lithium Polymer battery packs. Finally, the power trans-

mission system is made of low precision plastic parts creating low tolerances, leading to unpredictable power

loss. Several hobbyists have actually published articles on the inadequacies of the Silverlit RC X-UFO. These

inferior components detrimentally affect the quad-rotor's performance as seen in the flight-testing discussed in

subsequent sections of the report.

Figure 3.1: Low quality mechanical gyro from X-UFO.

3.1.1 The Silverlit RC X-UFO Flight Potential

A series of flights tests conducted in a controlled indoor environment demonstrate the flight capabilities of the X-UFO. These tests prove that the X-UFO is difficult to fly, and has an extremely limited operating endurance and severely low payload potential.

Control of the X-UFO is achieved just as any commercially available quad-rotor using a four-axis remote control system with trimming capabilities on each of the axes. Initially, the trimming controls are used in an attempt to maintain hovering flight. The X-UFO has a tendency to yaw, which makes manual control difficult because the orientation is constantly changing. Translational flight only exacerbates this yaw problem. Hours of practice yield improvements in flight stability, but mastering the remote control flight of the Silverlit quad-rotor would take many additional hours. Videos of Draganflyer®quad-rotors display the advanced maneuvering capabilities of the Draganflyer which uses the advanced electrical gyroscope, thus exemplifying the poor quality of the mechanical gyroscope of the X-UFO. The low-cost plastic gimbal housing is partially responsible for the delay in orientation sensing. A simple washer is used as a flywheel, which is not ideal and probably selected to reduce cost. Literature review revealed that replacing the mechanical gyro with a MEMS device reduces drift,

results in greater sample rates and increased accuracy, which will all lead to a more stable flight.[1].
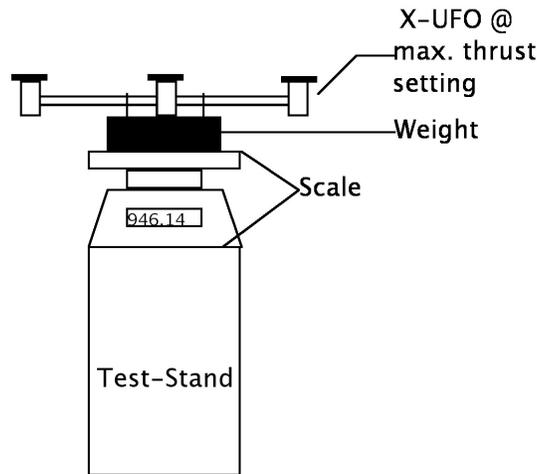
Figure 3.2: Maximum thrust experimental setup

Apart from control issues, there is an extremely limited operating endurance. Initial flight-testing yielded operating times of three to five minutes, which is inadequate for almost all applications. To quantify these inadequacies of the X-UFO, a thrust test was designed and thrust readings taken.

To calculate the thrust and endurance of the X-UFO, the craft was fixed to a weight which it could not lift. Then the X-UFO and attached weight were placed on a digital scale with a 0.1 grams resolution. The weight of the two were recorded as the base value, as shown in Figure 3.2. The scale was placed on a tall, skinny stand to minimize any ground-effect. Next, the X-UFO was set to its maximum thrust and readings were taken at the predetermined intervals. The effective thrust is simply the difference between the base value and the reading taken at the maximum thrust setting.

The results can be seen in the thrust and payload curves in Figure 3.3 and 3.4 below. The X-UFO's maximum net thrust is 224 grams, and it quickly drops off as seen in the figures. At approximately 2.6 minutes the X-UFO can no longer lift its own weight. In free flight this is the point when the Silverlit starts to lose altitude. It can be seen in Figure 3.4 that the hovering endurance decreases exponentially as time increases. For a modest payload
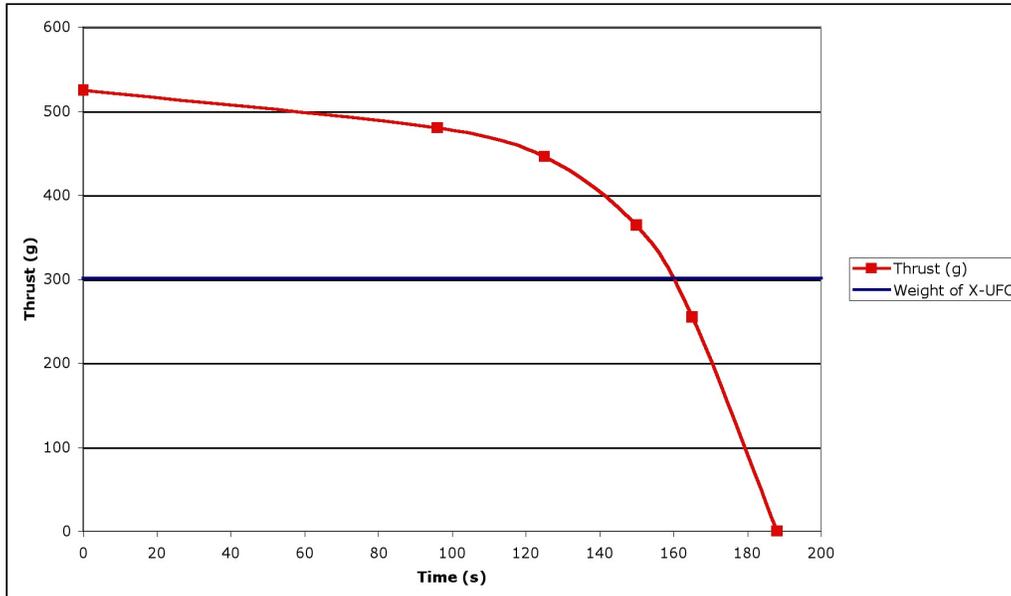
Figure 3.3: Maximum X-UFO thrust vs. time

of 150 grams the hovering endurance is roughly two minutes, and for a payload of 200 grams the hovering endurance is less than one minute. However, if 100% of the thrust is used to maintain hover, no thrust is left over for altitude changes and translation.

Vehicle testing determined that the Silverlit X-UFO is only suitable for recreational flight. Control issues make navigating through a real-world environment exceedingly difficult, and putting a high-tech and expensive sensor array designed to facilitate important, time-critical operations on an inherently unstable platform is unwise. The tiny payload severely limits the types of sensors that can be used, and the almost non-existent operational endurance further limits the available sensors. Due to all these limitations the X-UFO was redesigned with high performance components.
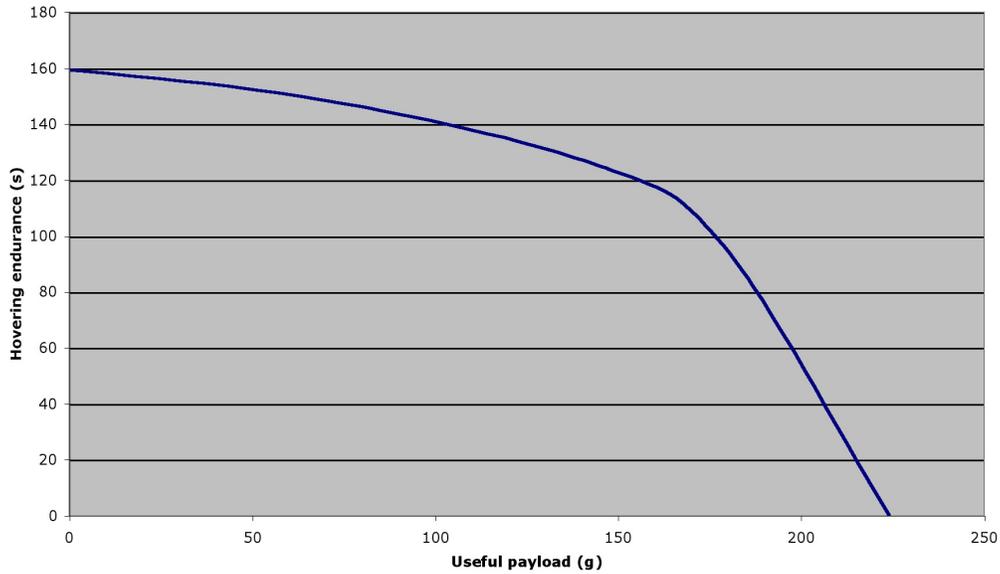
Figure 3.4: Useful payload with corresponding hovering endurance

3.2 The Redesigned Quad-Rotor

3.2.1 Motors and Electronic Speed Controllers

A miniaturized quad-rotor's driving force is produced by four servos and a electric speed controller (ESC) for each motor.

Servos

A servo is the component used to provide the driving torque. In our case, the servos drive each of the four propeller blades. There are two types of servos: brushed and brushless. The brushed DC motor is the conventional type of servo. The brushes make mechanical contact with a set of electrical contacts on the rotor, called the commutator. This forms an electrical circuit between the DC electrical source and the armature coil-windings. As the armature rotates on the axis, the stationary brushes come into contact with different sections of the rotating commutator. The commutator and brush system form a pair of electrical switches, each firing in sequence. Consequently, the armature coil closest to the stationary stator, which is a permanent magnet, constantly has

22

Table 3.1: Quad-Rotor thrust output.

Time (s)	Thrust (g)	Useful Payload (g)	Initial Weight (kg)	Recorded Weight (kg)
0	525	223.97	1.955	1.43
96	480	178.97	1.955	1.475
125	446	144.97	1.955	1.509
150	364	62.97	1.955	1.591
165	255	-42.034	1.955	1.7
188	0	-301.034	1.955	1.955

electrical power flowing through it.

In a brushless DC motor, the electromagnets are stationary. The permanent magnets rotate and the armature remains static. This gets around the problem of how to transfer current to a moving armature. To do this, the brush-system/commutator assembly is replaced by an intelligent electronic controller. The controller performs the same power distribution found in a brushed DC motor, but uses a solid-state circuit rather than a commutator/brush system.

Comparison of Brushed and Brushless DC Motors   Brushless DC (BLDC) motors offer several advantages over brushed DC motors. Brushless motors have higher efficiency and reliability, reduced noise, elimination of ionizing sparks from the commutator, overall reduction of electromagnetic interference (EMI) and no brush erosion which creates a longer lifetime. The maximum power that can be applied to a BLDC motor is exceptionally high, limited almost exclusively by heat, which can damage the magnets. To reduce excessive heating, the servos and their controllers are strategically positioned in the down-wash of the propeller blades which offer a cooling effect. Brushless DC motors' primary disadvantage is their high cost compared to brushed DC motors, which is due to two issues. First, the BLDC motors require complex ESCs to run. Brushed DC motors are regulated by a variable resistor and potentiometer or rheostat, which is inefficient but satisfactory and cost effective. Second, few practical uses have been developed for commercial purposes. For example, RC hobby

helicopters' commercial brushless motors are often hand-wound, whereas brushed motors use armature coils, which are inexpensively machine-wound.

Brushless DC motors are more efficient than brushed DC motors. The absence of friction in the brushes of a Brushless DC motor will convert more electrical energy into mechanical power than a brushed motor. Brushless DC motors are designed to operate over a broad range of speeds and have the advantage of reduced maintenance since there are no brushes or commutators. In conclusion, DC brushless motors offer increased speed and superior performance with higher torque-to-inertia ratio.
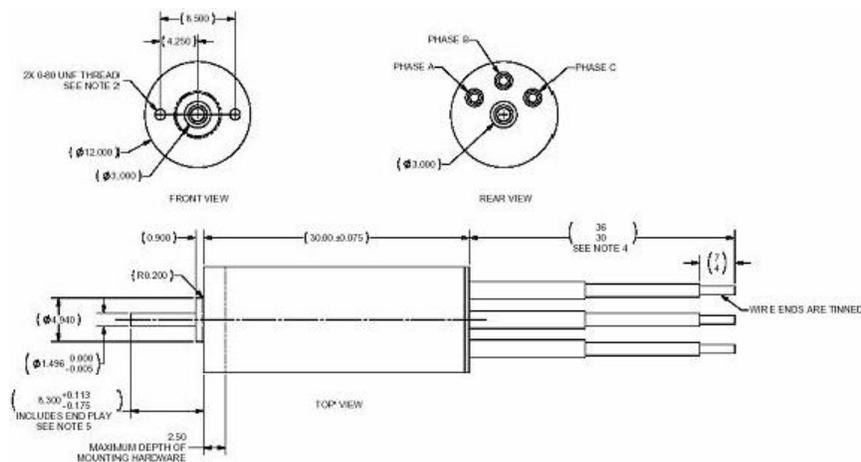
Figure 3.5: Schematic drawing of MR-012-030-4000

MR-012-030-4000

After much deliberation the Medusa Research MR-012-030-4000 Brushless DC motor was chosen to be used on the quad-rotor being designed for this project. The decision was based on dimensions, cost, output power rating, efficiency and amperage requirements. The specifications of the servo are shown in Table 3.2 and a diagram of the servo is shown in Figure 3.5.

Table 3.2: Properties of servo

Properties	Values
Dimensions (DXL)	12 mm x 30 mm
Shaft Diameter	1.5 mm
Weight	15 g
Kv	4000 RPM/V
I_o @ 8 V	0.3 A
Rm	300 mOhm
Pin	44 W
I/I_max	5 A / 6 A
V_max	15 V
Thrust	200 g

Electronic Speed Controller

An ESC directs the servo, which in turn controls rotor rotation. There are two ways to determine the rotor's position relative to the stator coils and the rpm. One way is to use a controller with a rotary encoder and directly measure the rotor's position and RPM. The other method, which was employed for this project, uses an open loop encoder whereby the pulse width module (PWM) is monitored to determine the motor rpm. This method is less costly, but not as accurate because the motors' RPMs, in this case, are being approximated.

3.2.2 Batteries and PSUs

Quad-Rotor's Initial Battery

Choosing the correct battery for a quad-rotor is difficult. The batteries that are generally used are NiMH (nickel). Nickel batteries present several problems. They do not contain enough energy to give an extended flight time. Roughly five minutes of flight time is produced during normal flight before the battery fails. Operating the quad-rotor takes practice. An operator may find themselves charging the battery three or more times before even achieving a stable hover. Another disadvantage of nickel batteries is inconsistent discharging from numerous battery drains. This results in quick drops in voltage toward the end of the battery's life, causing a lot of instability in the four motors. The last major consideration when choosing a battery is its weight. The battery is

the heaviest component on the craft, possibly outweighing the rest of the components on the vehicle.

Increasing the Battery's Life

Several options to increase flight time are available. One option is called Zapping. Several highly charged capacitors are discharged in parallel through the cells in the battery pack. If the current is high enough, the connections on the batteries will spot-weld themselves to their connectors, lowering the battery's overall internal resistance. In pervious studies this process increases battery performance by as much as ten percent. Another way is to create a larger battery pack, placing several additional batteries in parallel. For instance, create three 'hubs,' made of three battery cells placed in series, then connecting the hubs in parallel. There are other configurations that are useful for increased battery life. This process is not useful for nickel batteries because they self-discharge when connected in parallel. In contrast, when two Li-poly (lithium ion polymer) packs are connected two in parallel, they double their capacitance, while delivering half of the current draw through each cell. One workable pack configuration has three cells wired in series with two of these three cell configurations wired in parallel, which is known as 3s 2p packs.

Lithium Polymer Batteries versus Nickel Batteries

The development of Lithium ion polymer packs is an improvement in battery technology, slowly rendering its cheaper counterpart, nickel, obsolete. The Li-poly packs test very consistently, with random drops in voltage a thing of the past. Li-poly batteries are also three times more energy dense than nickel batteries, consequently, they have a lower weight, which increases flight time drastically. A Li-poly battery solves the majority of the issues with nickel batteries, although it introduces some new ones.

The lithium batteries are very expensive. A model with 11.1 volts and 6,000 mAh costs about $210. Even with educational discounts the cheapest battery found was about $185. To avoid high costs, modification of a cheaper pack can be explored. The Li-poly batteries are so energy dense there is a possibility they will catch fire

because they operate very hot, which will ruin the battery. Running a high current through such an energy dense battery can be very dangerous. Another option is to connect several smaller batteries in a configuration like the one described above in the 3s 2p packs. The issue with that configuration is the batteries need to be discharged at the exact same rates. If one cell performs slightly better then the others and drains slower, the cell becomes unbalanced, accumulating charge with each charge and discharge. This can result in the pack actually catching fire in addition to lowering performance on the unbalanced cells, however with a computing system the battery's charge can be controlled.
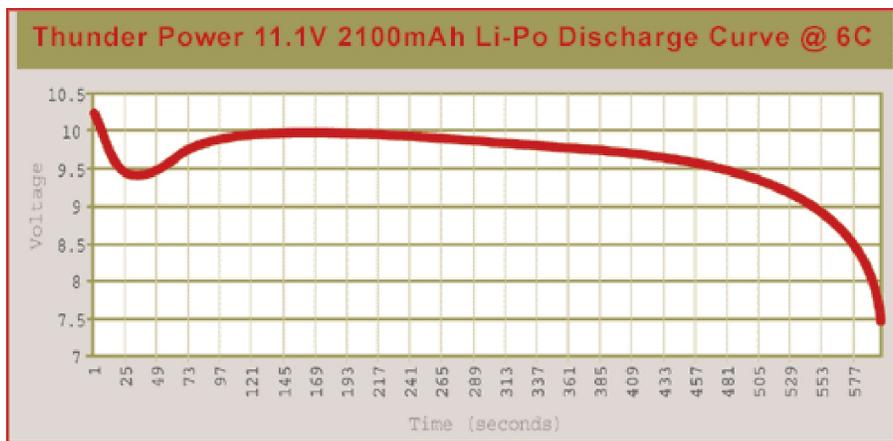
Figure 3.6: Voltage vs. Time as a Li-Poly Battery is Drained.

Nickel batteries need to be completely discharged often to retain their charging capacity because they develop memory or voltage depression characteristics. An easy way to assure the battery is fully discharged is to allow the RC vehicle to continue running after the battery's power has been depleted past flight capability. Contradictorily, Li-poly packs cannot be discharged too much. There is a cutoff point where the battery should not be used anymore even though there is still a little energy left. To avoid the cells being discharged too much, they are equipped with a device that cuts off all current from the cell once it has reached its minimum charge. Due to these new complications an already made, standard, Li-poly battery is worth the added cost and the best selection for the project.

Battery Type	Price	mAh	Rating Continuous	Rating Burst	Weight (g)	Dimensions (mm)
Pro-Lite	$210	6000	10-12C (60A)	18C (108A)	381	23x50x185
MaxAmps	$145	5000	20C (100A)	50C (250A)	345	45x26x138
RC-Extreme	$180	5000	25C (125A)	50C (250A)	388	25x48x160
RC-Extreme	$165	4600	22C (101A)	50C (230A)	364	18x48x195
RC-Extreme	$160	4500	25C (112.5A)	50C (225A)	359	23x48x160
Pro-Lite	$160	4200	15C (63A)	24C (100A)	284	30x44x100
MaxAmps	$115	4000	20C (80A)	50C (200A)	275	44x20x137
Pro-Lite	$150	4000	12C (48A)	18C (72A)	255	17x50x185
RC-Extreme	$140	3850	25C (96A)	50C (192A)	306	22x48x148
RC-Extreme	$125	3400	22C (75A)	50C (170A)	289	14x48x195
Pro-Lite	$110	2650	13C (34A)	20C (54A)	168	34x66x36
Max Force	$55	2200	20C (44A)		190	40x120x19
RC-Extreme	$85	2200	25C (55A)	50C (110A)	170	25x33x107
Pro-Lite	$73	2000	12C (24A)	20C (40A)	120	20x50x65

Table 3.3: A Comparison of lithium polymer batteries with voltage of 11.1 volts. *many of the cheaper batteries have been discontinued*.

Charger

The last issue with the batteries to be addressed was selecting the charger for the lithium packs and understanding how they worked differently than the previously selected nickel battery's charger. A Li-poly charger functions by charging each cell in the pack to 3.6V, at a constant current charge rate of about 75% of the cell's capacity. At this point the battery is usually about 80% charged. To bring the battery to full capacity, the charger switches to a constant voltage charge of 3.6V per cell. This process takes an hour to an hour and a half to complete.

Gear Ratio	5.86 : 1	7 : 1	8.6 : 1
Amperes	4.98	3.45	2.21
$\frac{F}{T}$ Endurance (LiPo 900 mAh) (minutes)	10.8	15.6	24.4
Efficiency	80.12%	81.22%	78.54%
Thrust (grams)	436	353	264

Table 3.4: Gear box ratios versus endurance and thrust.

3.2.3 Gear Box

The gear box is the component that connects the spars of the quad-rotor to the propeller and motor. The gear box also serves the purpose of increasing the gear ratio between the the motor and the propeller. A higher gear ratio increases the endurance, but decreases the thrust. Using an engineering calculator developed by Medusa Research [9] values for different gear ratios were obtained and are shown in Table 3.4. A '7 : 1' is the ideal gear ratio for the needs of this project because it provides enough thrust along with a long endurance.

3.2.4 Propellers

Figure 3.7: The quad-rotor propeller.

The propeller is the thrust producing component on the quad-rotor. Designs of the airfoil for the propeller are highly studied and coveted engineering secrets and have been for more than a century. Over the years many classes of airfoils have been developed. Each design created for different applications, producing different flight specifications. For the RC quad-rotor application there is only one class, the small-scale fixed-pitch rotor blade [7]. The miniature quad-rotor most commonly uses a twin-blade propeller for each of its motors. The twin-blade design is light and provides excellent mission endurance. Assuming compressibility effects are negligible the best airfoils have a high-aspect ratio, which increases lift. In aerodynamics however, maximizing lift is only as important as minimizing drag. A high lift-to-drag ratio is desirable. There are two types of drag, known as profile drag and separation drag, that must be be minimized. Profile drag is reduced by decreasing the camber and

reducing the cross-section. To reduce separation drag the trailing edge of the airfoil must be sharp, which creates a thin, highly acute angle. An additional consideration, is the moment force on the airfoil. A high moment force can increase wear, create unwanted vibrations, and increase drag. To reduce the moment, the center of pressure must be moved rearward. One can accomplish this by shifting the center of thickness and camber away from the leading edge, however this decreases lift. Consequently, there is a trade-off between minimizing the moment force and maximizing lift. Another way to reduce moments is by increasing the mechanical stiffness, which reduces angular deflection.

3.2.5 Electronics

To control the quad-rotor autonomously, it has a micro-controller, connected to an inertial measurement unit (IMU) and a wireless operations unit to drive the four main motors. To get the quad-rotor to hover or move in mid-air effectively and reliably, the craft must have a fairly accurate inertial measurement system, which sends all the necessary inputs to the micro-controller resulting in its ability to determine its instantaneous movement and correct for any instabilities in its flight.

Inertial Measurement Unit (IMU) Board

The IMU board is integrated in the main board of the quad-rotor. The IMU has inertial measurement components with 6 degrees of freedom (DoF). There is a 3-axis accelerometer, a 2-axis gyroscope and a 1-axis gyroscope. The accelerometer and the 2-axis gyroscope are a single unit with 5 DoF and the 1-axis gyroscope from SparkFun Electronics is the 6th DoF. The single axis gyroscope is the ADXRS300 gyroscope breakout board and the 5 DoF sensor is the SEN-00741 inertial measurement unit breakout board. The sensors are positioned as close to the center of mass of the craft as possible to simplify the calculations and produce more accurate results.

MaxSonar-EZ4

The MaxSonar-EZ4 is a high performance sonar range finder with analog and digital readouts complete with a breakout board for ease of integration. In analog mode with low signal noise and a shielded environment, it produces readings of voltage with a scaling factor of $(\frac{V_{cc}}{512})$ per inch. In this case, a supply of 5V yields $9.8\frac{mV}{in}$ and 3.3 volt yields $6.4\frac{mV}{in}$

Figure 3.8: General view of a MaxSonar-EZ4 range finder.

5 DoF Sparkfun Sensor SEN-00741

The 5 DoF unit is a pre-manufactured combination of an IDG300 dual-axis gyroscope and an Analog Devices ADXL330 triple axis accelerometer positioned on a breakout board for ease of integration.

IDG300 Dual-Axis Gyro   is an integrated dual-axis angular rate sensor, aka a gyroscope. It uses InvenSenses proprietary and patented MEMS technology with vertically driven, vibrating masses to make a functionally complete, lowcost, dual-axis angular rate sensor. All required electronics are integrated onto a single chip with the sensors. The gyroscope senses the tilt rates in 2 milli volts per every degree per second. For instance, if the angular rate is about $3\,^{\circ}$ per second, the readout is about 6 milli volts.

ADXL330 Triple Axis Accelerometer   The ADXL330 is a small, thin, low power, complete three axis accelerometer with signal conditioned voltage outputs, all on a single monolithic IC. The product measures acceleration with a minimum full-scale range of ± 3 g's. It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration. The output along every axis is the same, and the resultant output is about 300 milli volts per every $9.8 \frac{m}{s^2}$.
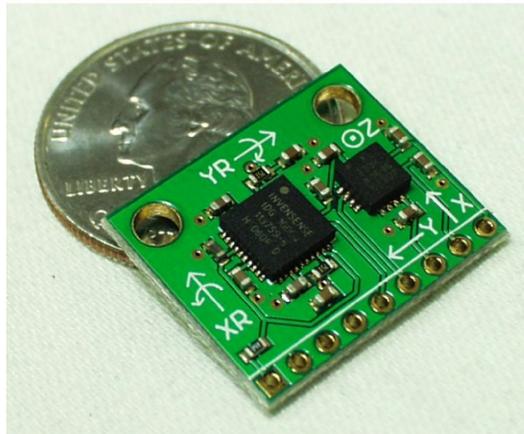
Figure 3.9: General view of a 5DOF SEN-00741 sensor.

1DOF Sparkfun Gyro ADXRS300   The ADXRS300 is a complete angular rate sensor (gyroscope) that uses Analog Devices surface-micromachining process to make a functionally complete and low cost angular rate sensor integrated with all of the required electronics on one chip. The manufacturing technique for this device is the same high volume BIMOS process used for high reliability automotive airbag accelerometers. The output signal, RATEOUT (1B, 2A), is a voltage proportional to angular rate about the axis normal to the top surface of the package as seen in Figure 3.10. A single external resistor can be used to lower the scale factor. An external capacitor is used to set the bandwidth. The results are delivered as 5mV every degree per second. (i.e. for $3\,^{\circ}$ per second, the output will be about 15 milli volts)

32
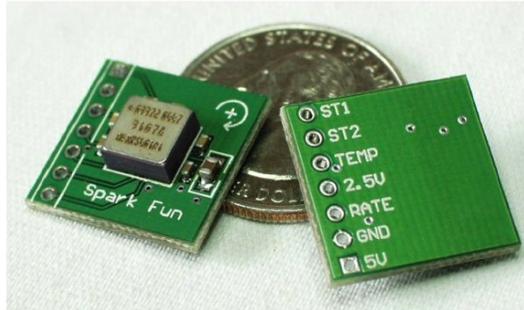
Figure 3.10: General view of an ADXRS300 with a breakout board.

The prototype board created by the team's electrical engineer is a proof of concept that the sensors can provide the outputs necessary for the quad-rotor's control system. A number of calibration experiments performed on the board tested for accuracy of the sensors and interference from a potential threat caused by the resultant noise of the high-speed blades and motors. The tests showed the sensors to all be working well and providing good, consistent, reliable, accurate outputs. Once the prototype was established as a working design, the team's electrical engineer was able to create a fully working model using the aforementioned sensors.

3.2.6   Sensors

The sensor suite of the board is described in section **??**. The sensor data is read by the MSP430 via it's 12-bit analog to digital converter in sequence mode; a sample from each sensor is copied to a distinct location in memory without intervention of the processor. An interrupt is triggered every cycle, and the samples are added to accumulators for those sensors, and after ten samples the results are sent to the uM-FPU for processing.
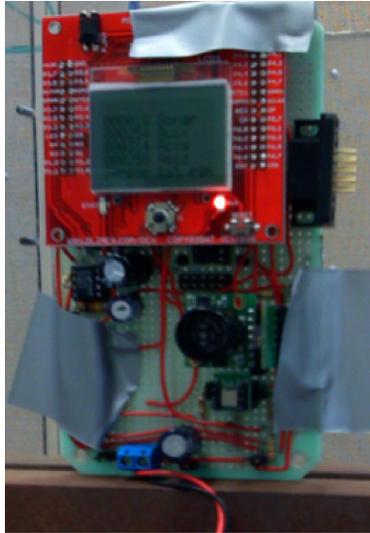
Figure 3.11: Picture of the Main Board Prototype rev. 1.0 by Alex Camilo.

3.2.7   XBee

To communicate with a base computer, an XBee module was included in the board design. This module provides either a point-to-point serial link or a more sophistocated mesh network; in our case, we only used the simple serial connection. For a multiple-robot swarm configuration, the mesh would be appropriate, and would only require code modifications.

3.2.8   Crystal Oscilators

A crystal oscillator is an electronic circuit that uses the resonance of a peizoelectric material to create a high precision frequency signal in a circuit. The board included two crystal oscilators to provide accurate timing sources for the MSP430. One produces 32.768 KHz, and the other provides 8.000 MHz.

3.2.9   Frame Design

Once all the readily available components selected for the quad-rotor are purchased, custom parts need to be designed and manufactured to properly configure the purchased components into a functioning quad-rotor .
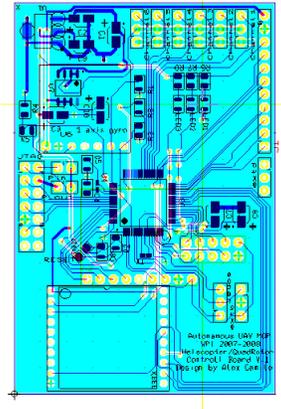
Figure 3.12: Schematic of the main board rev. 1.2.

Figure 3.13: View of the Main Board rev. 1.2.

Considering deadlines and minimal machining experience, it is essential that these parts are both simple to assemble and easy to manually manufacture. Additionally, an easy assembly design provides maximal flexibility when attaching necessary electrical components. The readily available components are:

- Gearboxes
- Propellers (clockwise and counter clockwise)
- Brushless motors
- Electrical motor controllers (ESCs)

One of each component is one driving assembly of the propulsion system. There are four driving assemblies on the quad-rotor, however these driving assemblies need to be connected to the controller board with the

Materials	Modulus of Elasticity (GPa)	Ultimate Tensile Strength (MPa)	Density ($\frac{g}{cm^3}$)
Nylon 6.6	2.61	82.8	1.14
Carbon Fiber (epoxy matrix)	220	760	1.7
Stainless Steel 440a (tempered)	200	1790	7.80
Wood (Red Oak)	14.1	112	0.67
Aluminum (7075 T6)	71	572	2.80

Table 3.5: A comparison of different material's mechanical properties.

IMU system. Therefore a frame needs to be constructed, which provides the necessary structural support for all components when the motors are exerting a force on the entire structure. The frame constitutes the main supporting structure of the quad-rotor, and is thus integral to its operation. The frame has to be durable and robust, in case of a crash or hard landing. It has to be light to maximize payload and flight performance. The frame must also be able to endure any possible environmental conditions, such as hot or cold environments, or solar radiation in outdoor applications. Considering these specifications the frame's material becomes an important factor.

Material Selection

There are a number of materials available for the frame design. Among them are plastics, carbon fiber, steel, wood, and aluminum. These materials have a wide array of different mechanical properties. Table 3.4.3 lists the top five material considered and lists the major mechanical properties taken into account.

Nylon is a polymer material that is very light and easy to manufacture. It is abundant and easy to procure, however, it is not very stiff, and its mechanical properties are incredibly sensitive to environmental conditions. Ultraviolet rays will break down the molecular structure, compromising its strength. Humidity also severely compromises its strength: a relative humidity of 50% cuts its ultimate tensile strength (UTS) in half.

Carbon Fiber Composite is comprised of carbon fibers in a matrix of another material, typically an epoxy resin. Carbon fibers are incredibly stiff, having the highest modulus of elasticity of all the materials investigated. It is very light and very strong: its UTS is 1.3 times aluminums yet it weighs 40% less. Unfortunately, its strength is directional with a UTS along its transverse axis (perpendicular) to the fibers) is only 28 MPa. Also, a carbon fiber composite fails with little deformation, consequently failure is often abrupt and violent, as opposed to most metals, which bend before breaking.

Steel is an alloyed metal, which has the highest UTS of the materials investigated, beating the competition by a factor of two or more, and has the second highest stiffness. It is however by far the heaviest, nearly three times as heavy as aluminum. It's increased density requires more time to machine making it a difficult material to use.

Wood is perhaps the most widely used structural material in the world. It is lightweight and is very easy to work with. Unfortunately, it is not as strong as the other surveyed materials, and its strength is directional like carbon fiber composite. Red oak's UTS along its transverse axis is merely 7.2 MPa. Not to mention the countless environmental issues to consider. Ultraviolet rays break down the fibers of the wood, and there is potential for rot.

Aluminum T6-7071 is an alloy of aluminum that has been developed especially for aerospace applications. It is a good compromise between strength, rigidity, and weight. It isnt as strong as stainless steel, but weighs considerably less, and it isnt as light as carbon fiber, but doesnt have the same directional load dependency.

Frame Components

To design an easy to manufacture frame, it is broken down into three components that are easy to manufacture individually:

- Central hub

- Spars

- Gearbox mounts

The frame is designed to be as light as possible to utilize as much of the thrust capabilities from the motors as possible. To accomplish that, the arm length are determined to be

$$D_{s_{min}} = \frac{D_p}{\sqrt{2}} \qquad (3.1)$$

where $D_{s_{min}}$ is the length of the spar and D_p is the diameter of the propeller. In the case of this quad-rotor design, $D_p = 9$" and $D_{a_{min}} = 6.364$". To assure the propellers do not touch the actual spar length is rounded up to 7". In addition, the spars had to be sufficiently strong to deal with the moment forces exerted on the arms.

Figure 3.14: One half of the post manufacture central hub.

The Central Hub is the main connection piece of the quad-rotor. It connects the four driving assemblies to the electronic housing, which contains all the sensors and processing ability, therefore the center hub must be strong and resilient to elastic deformation in all directions. To form strong and interchangeable connections between the hub and various components, the material must be capable of forming lasting threads and/or boils. In addition, the material must be light weight to increase the flight potential. Taking all these specifications into account aluminum is the most appropriate material.

Once a material is selected the next step is to design the center hub. Some considerations when building a quad-rotor are loose wires, weight, assembly and functionality. To keep the wires contained hollow spars and frame were necessary, therefore a two piece center hub is built. To keep the weight down, the thickness of the walls is minimized. For easy and quick assembly many hole are bore into the frame to attach any necessary components. Finally, functionality is accomplished with a simple cross as seen in Figure 3.14.
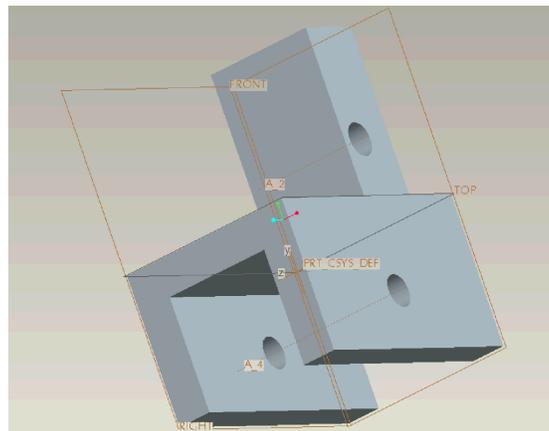
Figure 3.15: A Solid Works image of the gearbox mount.

The Gearbox Mounts   is the smallest piece manufactured by the design team. The mount is designed to connect the spars to the gearbox. Based on the sized of the spars and the gearbox the mount must be very small, therefore it needs to be strong in all directions to support the force from the motor as it is translated into the load on the longitudinal arms. Mounting hardware is required to connect the gearbox and spar together. The connection from the mount to the spar is especially important because the unidirectional load from the motor has to be translated into the load on the spars, therefore vibrations must be reduced. Consequently, all the dimensions for the fastened parts of the frame must be within very limited tolerances to function properly. Additionally, the mount must be light. Weight is especially important far away from the center of mass of the structure, the lower weight lowers the overall moment of inertia and will result in a more controllable machine. Based on these

specifications the best material is aluminum.
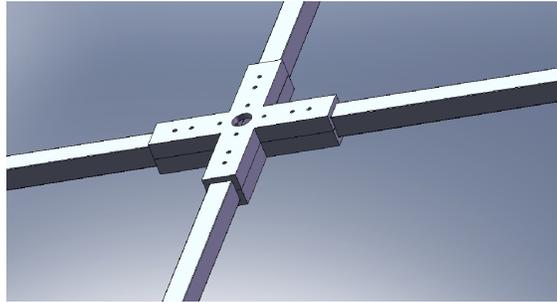
Figure 3.16: A Solid Works image of the structural arms.

Spars   There are many possible materials for the spars of the quad-rotor. Each comes with its own set of strengths and weaknesses. The material that had overall properties that best met the specifications is carbon fiber, sometimes called graphite fiber. It is comprised of many long strands of carbon fibers interwoven into tubes, rods, sheets, and many other shapes. This material had many qualities that helped optimized the design. One of the primary deciding factors is its extreme light weight. It is much less dense than steel or aluminum, and, out of all composite materials, has the highest compressive strength. The lift from the motors travels along the longitudinal axis of the arm, therefore carbon fiber is strong enough to support the in flight forces exerted by the servos and mass of the electronics that are attached to it. Carbon fiber composite also has a very high stiffness, which is important to lesson vibrations and limit changes in the thrust angles from deformation. Since it has a low coefficient of thermal expansion, it is unaffected by temperatures that the quad-rotor is designed to operate. The temperatures that the quad-rotor will be exposed to will range from 0°C up to 30°C.

One problem with carbon fiber is how hard it is to machine, which necessitates the use of a machine controlled cut. Another of the problems with the carbon fiber is its tendency to corrode aluminum. The four carbon fiber rods that will comprise the arms of the quad-rotor will be fastened together in the middle with an aluminum cross. The gearbox brackets that will mount at the tip of the rods are also aluminum. This long term problem

can be avoided by coating the places where the two materials will touch with an epoxy. The expected overall time of this project is too short and should not be an issue. However, any corrosion at all may affect the strength of the rods, so they will be coated.

The machining issues can be overcome through a couple of steps. The first step is creating a sort of jig that can hold the rods firmly in place during drilling. A 1 foot section of 2x4 can be used as the base. Two small long pieces of wood running the length of the 2x4 are nailed together at a distance apart slightly greater than the thickness of the rods. A rod is then wrapped several times around in masking tape. The tape is compressive enough to hold the rod firmly in the jig without crushing it. It also served to support the face that is being drilled along with the wooden walls and base. Drilling at high speeds, with a series of 3 relatively new bits to reach final diameter proved successful. The band-saw issue can also be overcome by taping the rods, a little thicker than before. It was fed through using a straight flat piece of wood to brace the back of the rod, keeping it from bending when the blade put pressure on it.

Once the Carbon fiber rods were cut they had to be finished. Another problem with the rods is they are fibrous in nature, they absorb liquids. This can seriously affect integrity. To prevent this, the rods can be treated with superglue (cyanoacrylates), epoxies, or even wood glue. Superglue seems to be the best choice. Before the cuts and holes can be treated any excess fibers that have been frayed or are hanging need to be cut with snips and then smoothened with scotch rite or fine grain sandpaper. After they are treated the rods are ready for use.

There are a total of five rods cut, with four coming out usable. Each rod weighs 5.4 grams, for a total of 21.6 grams. The aluminum spars weigh over 10.5 grams each, so the carbon fiber will cut the spar weight in more than half. One hole is drilled through the spar at the end for it to mount to the middle frame support. The other end is drilled through the opposite two faces as the first, and this is where the gearbox brackets will mount. Weight testing is done on 14 inch rods of the same diameter. They have no issue holding weights up to 1.5 kg, which is more than our whole craft will weigh. The 8 inch spars will have no problem supporting the torque or

weight from the motors.

Landing Gear   The landing gear is a key component of the quad-rotor. The landing gear must not only hold the aircraft up when the quad-rotor is at rest, but it must absorb any shock from a hard landing if there is a mechanical error during flight. There are several possible designs of landing gear that were considered. Initially, having a hard landing gear below each of the motors made of solid aluminum was considered. However, a solid piece of aluminum is stiff and heavy, which would be bad for the quad-rotor. To maximize the landing absorption ability it is decided to have flexible rods connecting each motor. To be stiff enough to support a hard landing the rods have to be made of stainless steel. One of the main problems with this design is connecting the rods to the motors. It is necessary to have removable landing gear for optimal flexibility to design of the quad-rotor down the road. Therefore, the use of epoxies and welding is avoided. In the end a thin plate is screwed to the bottom of each gear bracket and the stainless steel rods are rapped around holes drilled into the flat plate. This design provides easy assembly, proficient shock absorption and minimal weight making it the best design for the landing gear.

3.3   Manufacturing

The three major components of the frame are custom and required manual machining. Each member of the group familarized themselves with the available machines and created custom designs and processes to create unique piece. The main components manufactured are the center frame supports, the gearbox mounts, and the arms.

3.3.1   Central Hub

The center piece is created from two flat, square pieces of aluminum milled to the desired specifications and shape. These pieces serve to fasten the four arms of the quad-rotor together and hold the electronics hub off

the ground. The carbon fiber rods on the final design of the quad-rotor require two fasting points, where the test version of the quad-rotor only required a single piece. The additional fastening point is due to the possible weaknesses formed during the tricky processing stage. The frame's many constraints required a versatile and simple, for easy adjustments, center piece. The steps to manufacture the center piece are:

- Cut piece of aluminum into a flat, squared-off piece at the desired specifications all around

- Perform two center cuts per piece with several variable cut width, at a single depth pass, with an end-mill, using cutting fluid.

- Cut the excess material off with a band-saw resulting in a cross shaped piece that closely resembles the design.

- Drill holes with the desired placement, which can be determined accurately by the coordinate locator on many milling machines.

- The piece is finalized by milling the corners cut with the band-saw to the desired specifications all the way around to increase precision. A curve at the base of each support hub arm increases the strength.

3.3.2 Spars

Machining the rectangular carbon fiber rods has proven itself to be very difficult. Preliminary machining is done on test pieces to acclimate to the machine and material. The best tool for cutting carbon fiber rods is a diamond tipped jewelers saw, however this tool is expensive and unavailable. Consequently, other manufacturing machines were used. The band-saw catches the fibers rips the rod on the blade causing it to splinter. There are also problems drilling into carbon fiber rods. Since diamond tipped bits are also unavailable, when the non-diamond drill bit is lowered to the face of the material, the fibers would snag on the bit and split the rods. These issues can be over come by following a few steps:

- Create a sort of jig that can hold the rods firmly in place during drilling with a one foot section of 2x4 to be used as the base.

- Place two thinner, one foot pieces of wood running nailed to the 2x4 length wise, separated by a distance slightly greater than the thickness of the rods.

- Fix the rod in place using tape. The tape is compressive enough to hold the rod firmly to the jig, adding support without crushing it.

- Drilling at high speeds, with a series of 3 bits each increasing in size proved a capable method.

The carbon fiber splitting when using the band-saw can be overcome in a similar way as with drilling. Tape the rods in the same way as done when drilling the carbon fiber, however the tape should be thicker than before. The contraption is fed through the band-saw using a straight flat piece of wood to brace the back of the rods, preventing it from bending when the blade put pressure on it.

Once the Carbon fiber rods are cut they require a finish. The rods are fibrous in nature, as a result they absorb liquids. This can seriously affect the integrity. To prevent this, the rods must be treated with superglue (cyanoacrylates), epoxies, or even wood glue. Superglue is the most effective. Before the cuts and holes can be treated any excess fibers that have been frayed or are hanging need to be cut with snips and then smoothened with scotch rite or fine grain sandpaper. After the rods are treated, they are ready for use.

3.3.3 Gearbox Brackets

The gearbox mount is a small, but crucial piece of the quad-rotor. It has many small cross-sections and requires careful machining. The steps to machining the gearbox mount are:

- Cut a piece of aluminum that to specified maximum lengths of the gearbox mount design.
- The top notch is formed by milling out the entire area surrounding the piece.
- After the gearbox is nice and snug on the notch, the assembly is rotated upside down and a center cut is made for the arm to be easily placed and attached to the gearbox mount.
- Drill two holes on the side of the gearbox. One is to hold the arm in place and one to hold the gearbox in place.
- On the top drill two more holes. One for the driveshaft of the gearbox so it rotates freely without obstruction. One for a vertical support of the gearbox mount against the arms.

3.4 Calibrating

3.4.1 Sonar

To calibrate the sonar sensor two experiments were conducted. One to calibrate the voltage to distance ratio and the other to determine if the servos and propellors harmonic output interfere with the sonar's capability to
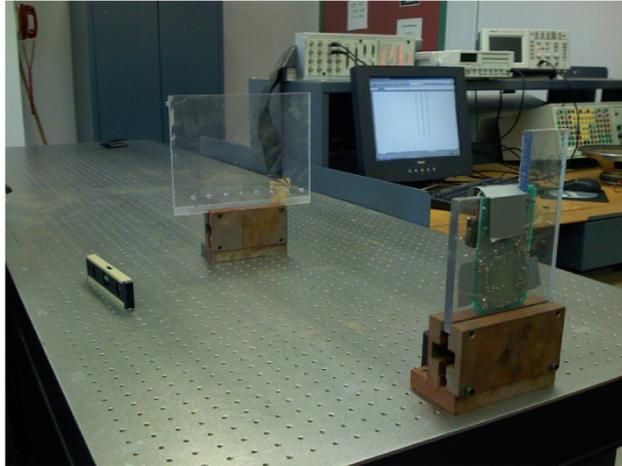
Figure 3.17: Sonar Calibration Set-Up.

measure distances. To calculate the voltage to distance ratio a stand is designed and constructed to hold the sonar at a $90\,^\circ$ angle to the table.

Figure 3.18 shows the IMU board with the sonar mounted on it. The IMU board has an LCD display which gives the output of the sonar readings in millivolts. The board is mounted at the mid-point of one end of the lab table. The x-axis lay on either side of the board and the y-axis is perpendicular to the board along the length of the lab table. Using a 6 x 6 x 0.25 inch perspex panel attached to the stand on the calibration table, the sonar's longitudinal and lateral sense capabilities are tested at 6 inch intervals along the x-axis and y-axis. Next, the experiment is repeated, however this time it is run with the quad-rotor turned on next to the sonar as the readings are taken.

Both experiments produced identical results regardless of the presence of running servos and propellers. The results for both experiments were tabulated and graphed in Figure 3.19. The results of the first experiment were superimposed by the results of the second experiment.

From the results, we arrived at the conclusion; noise generated from the servos and propellers will not interfere with the operation of the sonar. The frequency of the sound waves from the servos and propellers are
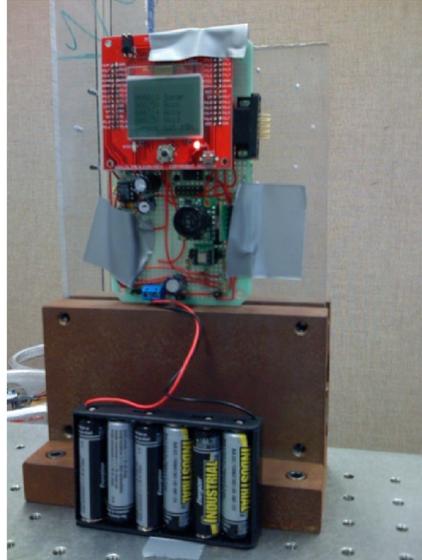
Figure 3.18: Prototype circuit board with accelerometer, gyroscope and sonar.
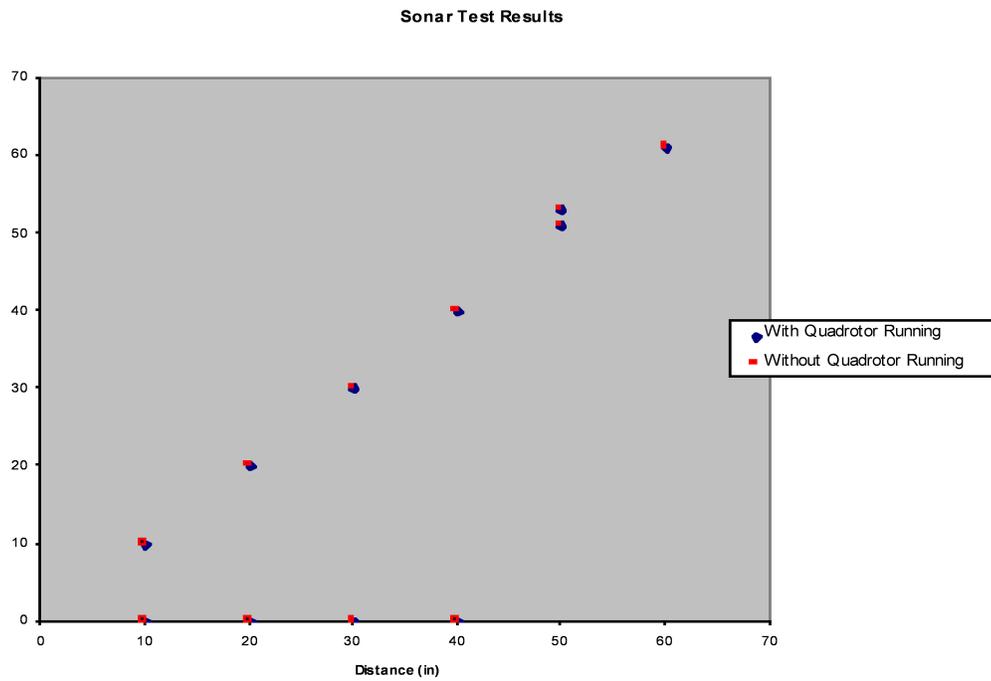
Figure 3.19: Sonar Calibration Graph.

lower then the ultrasonic frequency sound waves through which the sonar operates. In addition, we successfully calibrated the sonar.

3.4.2 Propeller Balancing

The quad-rotors propellers are designed to rotate as fast as 6,000 times per minute, making balance a crucial factor. The blades of the propeller may vary in weight due to imperfections and therefore, at certain revolutions per minute (RPMs), the propeller will vibrate uncontrollably. Balancing the propeller insures it will rotate without causing undue vibrations to the airframe and the electrical components. If left unchecked the vibrations may become destructive to the quad-rotor loosening its bolts and eventually damaging components to the point of failure.

First, the propellers and servos must be mounted symmetrically. Next, ensure both blades of the propeller are balanced over their span. There are more complicated balancing issues, with respect to the hub, however they apply to much larger propellers than those used on a miniature quad-rotor. The propellers focused on here are 9 in length and have 6° pitch angles, (9X6) propellers.

A specially designed propeller balance measurement device is used to balance the propeller. There are many types of model propeller balancers. Master Airscrew developed a propeller balancing device that suspends the propeller in a magnetic field with very low friction. Top Flites Power Point Precision Balancer seen in Figure 3.20 works in a similar manner. Great Planes Fingertip Propeller Balancer offer a propeller balancer for less than \$5.00. All of these model propeller balancers are suitable for the (9X6) propellers used on the quad-rotor.

To operate the balancer position the propeller on the spindle as done in Figure 3.20 above, and place the spindle on the balancer allowing only one side to contact the magnet. Rotate the propeller to a horizontal orientation and release the blade. If the propeller balances horizontally, rotate it 180^{circ} and check it again. If one blade is heavier than the other the propeller will rotate until the heavy blade is pointing down as in Figure 3.21 below.

Figure 3.20: Top Flites Power Point Precision Balancer.

Figure 3.21: An unbalanced propeller.

An unbalanced propeller is corrected by removing material from the trailing edge of the blade near the tip as

done in Figure 3.22. Be careful to preserve the airfoil shape while removing material conservatively. Very little

material should be removed between each balancing test as shown in Figure 3.23.

Figure 3.22: Balancing the propeller.

The propeller is balanced when a satisfactory orientation has been attained as illustrated in Figure 3.24
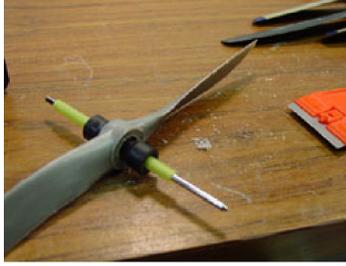
Figure 3.23: Shavings from the propeller.

Figure 3.24: Satisfactory orientation of a balanced propeller.

3.4.3 Gyroscope and Accelerometer

A gyroscope is an electronic device used to measure angular rates. When this sensor is mounted on the quad-rotor board it will output measurements that determine the orientation change, which consequently will enable the quad-rotor to maintain the desired orientation. An accelerometer is an electronic device used to measure the total external force on the quad-rotor, also referred to as the acceleration. This enables the quad-rotor to determine its change in motion along all three axis, which will allow the quad-rotor to determine its position with respect to its starting position. The goal of this experiment is to establish that the gyroscope and accelerometer sensors mounted on the board are in good working order and additionally to calibrate them. The experimental setup comprised of the following equipment:

- Dual-direction rotating, direct drive Numark turn-table with 33, 45 & 78 RPM setting
- Infra-red sensor
- Oscilloscope

Figure 3.25: Turn-table set-up

- Circuit board

- Battery pack

- Masking tape

To calibrate the sensors, it must be determined that the X-bee is transmitting the output of the sensors wirelessly in an effective manner. This is done by connecting the circuit board to the battery pack. If there are outputs to the computer then the first step is complete. Next, the accuracy of the rotational speed of the direct drive Numark turn-table with the circuit board and battery pack strapped onto the rotating plate must be verified. There is concern that the weight of these items might slow down the turn-table significantly, which would give inaccurate readings. The rotating plate of the turn table has 90 equidistant notches all around the outer edge. These notches are used to calculate the frequency of the rotation on the three different settings of 33, 45 and 78 revolutions per minute. This is done by connecting the infra-red sensor to an oscilloscope and mounting the infra-red sensor adjacent to the notches of the plate while securing and strapping the board and battery pack to the rotating plate using masking tape as illustrated below.

The outputs from the oscilloscope on the three settings, 33, 45,and 78 RPM, are equal to 49.5 Hz, 67.5 Hz and 117Hz, respectively. The infra red sensor reads the 90 notches on the side of the turn-table to determine the speed at which the turn-table is rotating. The rotational speed of the turn table is verified from the frequency

readings of the oscilloscope as outlined below:

$$\frac{49.5\,Hz}{90\,Notches}60 = 33\,RPM$$

$$\frac{67.5\,Hz}{90\,Notches}60 = 45\,RPM$$

$$\frac{117\,Hz}{90\,N0tches}60 = 78\,RPM$$

The turn-table appears to work well from the results. Next, the centripetal acceleration on the board is calculated using the equation below:

$$a_e = -\frac{V^2}{r}\mathbf{r} = -\frac{V^2}{r}\frac{\mathbf{r}}{r} = -\omega^2\mathbf{r} \tag{3.2}$$

in $\frac{m}{sec^2}$. Where

$$\omega = \frac{oscilloscope\,frequency}{corresponding\,RPM} \tag{3.3}$$

in $\frac{radians}{sec}$, and $\mathbf{r}$ is measured in meters.

The circuit board is oriented and strapped in various positions to measure the desired orientation outputs from the sensors. Prior to running the turn-table, the distance from the center of the turn-table to the sensor is measured. The data is collected by running the turn-table in both clockwise and counter-clockwise directions over durations of five minutes on each orientation to minimize any errors. However, it is not possible to calculate the accrued error from using poor quality sensors. The experiment is conducted based on the assumption that the sensors are ideal sensors. The data collected is then used to calibrate the sensors and code the output of the sensors onto the MSP430 processor. Since the number of notches is equal to 90, the following results are outlined in the Table 3.6.

3.4.4 Thrust and Torque Measurement Experiments

Each motor and propeller unit generates both a thrust along the shaft of the motor, and a torque around the shaft of the motor. For proper control of the craft, it is necessary to know how each of these values changes with respect to control input into the motor.

Corresponding RPM	Oscilloscope Frequency (Hz)	ω ($\frac{Hz}{notches}$)
33	49.5	0.55
45	67.5	0.75
78	117	1.3
X-axis	**Positive**	**Negative**
Radius (m)	0.10581	0.10472
Corresponding RPM	**Centripetal Acceleration** $\frac{m}{sec^2}$	**Centripetal Acceleration** $\frac{m}{sec^2}$
33	0.032007525	0.0316778
45	0.0595518125	0.058905
78	0.1788189	0.1769768
Y-axis	**Positive**	**Negative**
Radius (m)	0.09608	0.12212
Corresponding RPM	**Centripetal Acceleration** $\frac{m}{sec^2}$	**Centripetal Acceleration** $\frac{m}{sec^2}$
33	0.0290642	0.0369413
45	0.054045	0.0686925
78	0.1623752	0.2063828
Z-axis	**Positive**	**Negative**
Radius (m)	0.11438	0.13507
Corresponding RPM	**Centripetal Acceleration** $\frac{m}{sec^2}$	**Centripetal Acceleration** $\frac{m}{sec^2}$
33	0.03459995	0.040858675
45	0.06433875	0.075976875
78	0.1933022	0.2282683

Table 3.6: A comparison of different material's mechanical properties.

To obtain the relationship between torque exerted upon the craft and control input into the motor an experiment must be conducted. The experiment is set up with the quad-rotor affixed to a ball bearing with a bolt to reduce friction. The ball bearing is clamped in a vice, and the whole arrangement is placed on the edge of a shock table, with the motor assembly a few inches over the edge to reduce and ground effect. A digital scale is affixed vertically to a mount and the mount is bolted to the table, with the scale flush against the quad-rotor, on the opposite side from the spinning motor. Different control signals are sent to the quad-rotor and the measured force is recorded.

The result of the experiment is approximately a linear relationship between the input signal and the torque, as seen in Figure 3.27. Through this an expression for torque as a function of input signal is obtained.
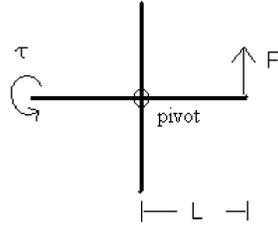
Figure 3.26: When the force F is measured by the scale, and a torque can be calculated with $T = LF$.
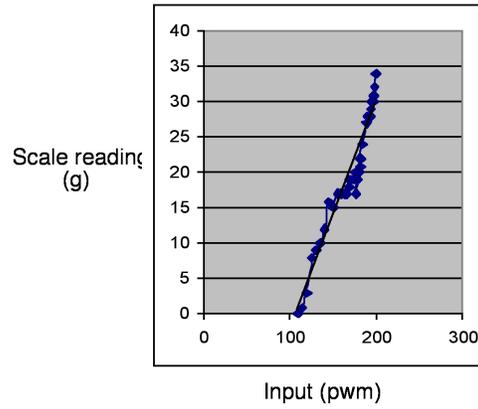
Figure 3.27: PWM input versus the scale reading in g

53
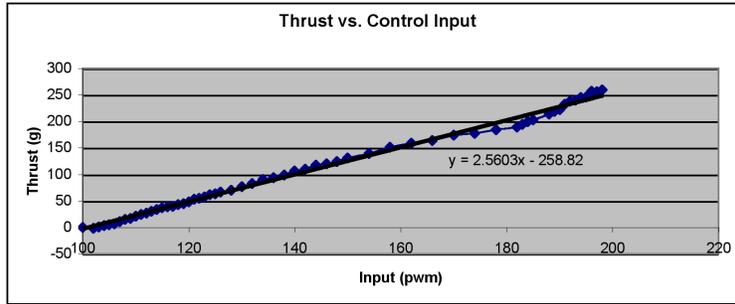
Figure 3.28: PWM versus Thrust in g.

To obtain units in Nm it is necessary to multiply the readings measuring in grams by $9.81\frac{L}{1000}$, which the model requires. This produces the following expression:

$$PWM = 0.0171T \tag{3.4}$$

where the PWM output is in seconds and the torque is in Nm. It is also necessary to obtain a thrust curve to determine how the thrust relates to the control input. A simple design for an experiment to measure thrust is set-up by placing the quad-rotor on a scale and measuring the thrust output corresponding to different control signals. Figure 3.28 shows the output from the experiment.

Using the curve in Figure 3.28, the following expression is obtained for the PWM signal:

$$PWM = 0.0004Thrust \tag{3.5}$$

where PWM is measured in seconds and the thrust is measured in newtons. These relations are crucial in autonomous computer control of the quad-rotor.

3.5 Vibration Reduction

One of the major issues that comes to light when the Quad-rotor is running off the computer is the amount of vibrations that occur. The servos themselves are quite powerful, and with the larger propellers and high RPMs, the motors themselves create a lot of vibrations. To reduce the vibrations several connection areas

54

between components need to be fastened using different techniques. The areas that require improvement are the propellers, gearbox mounts and the circuit board holders.

First, the propellers' for the vertical bars are too big and create a loose connection between the gear boxes and the proplellers. Fastening bolts and thin tape are added to the gearbox bar to reduce the wiggle in the propeller. Next, the gearbox bracket have a short vertical post that slides into the gearbox. A pin is then slid through the gearbox, into a hole drilled in the vertical post and then fastened with a nut on the other side.

The problem is the gearbox does not fit snug enough on the post. Masking tape is wrapped around the posts to try to tighten the connection, but it is too thick and does not stick to the aluminum. The posts has to be sanded down. After that each face has to be rubbed with the sticky side of a scrap piece of tape to remove any dust. Now the gearbox is able to fit over the tape and the tape didnt pull off. This small improvement makes a huge difference, decreasing the amount of vibrations drastically. Another method is to elastic bands connecting the spars together. The pull on the spars from the elastic bands helps to reduce any vibration. Finally, a lot of noise in the circuit board makes the sensors unreliable, therefore it is necessary to add padding between the aluminum frame and the circuit board. The padding drastically reduces the vibration experienced by the circuit board. All of these methods reduce the vibrations of the quad-rotor and reduce the noise experienced in the circuit board, which makes all the sensors more accurate.

3.6 Computer Control of Various Electronic Components

3.6.1 Motors

It is necessary to evaluate the performance of the electric motors, motor controllers and propeller configurations. The components requiring testing for functionality are the thrust and torque, which have to be measured, and the electronic speed controllers, which need to be programmed. To perform these functions, an interface system needs to be developed between the PC and the propulsion components.
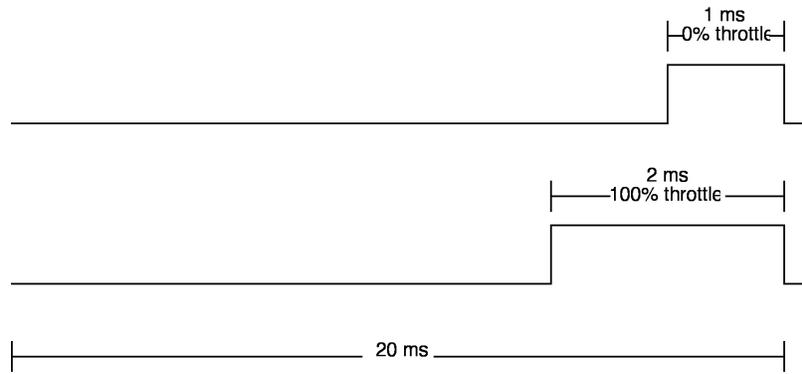
Figure 3.29: Motor Control Scheme.

The brushless DC motors require electronic speed controllers, which complicate the motor-controlling process. The electronic speed controllers are designed to receive a control input from a radio receiver. This signal is a 50 Hz PWM signal with pulse-widths between one and two milliseconds long. A one-millisecond pulse-width represents 0% throttle, while a two-millisecond pulse-width represents 100% throttle.

Initially, a dial-controlled signal generator is used. While, it could make the motors start turning, and change speed, the pulse-width could only be vaguely approximated using a oscilloscope. Therefore, It is deemed necessary to develop a much more robust control system.

Using MATLABs Simulink, a model can be produced of the control system, which merely consists of two program blocks. The first block is a PWM signal block, the inputs being output voltage, signal frequency, and pulse-width as a percentage of signal frequency. The second block is a Dspace digital in/analog out signal block, which takes the PWM signal and sent it to the motor controllers. It is possible to control the motors using this model, but every change in parameters required a rebuild of the model, which makes it difficult to test. Control Desk is then used to make a more useful control system. After building the model in Simulink, the parameters are linked in the PWM signal block to the control desk. A user-friendly layout is designed to make testing easier. It consists of multiple components. An on-off button is used, which changes the output from 0 to 5 volts. There are three ways to change the pulse-width of the signal. A slide-wheel is used for rapid changes, while a
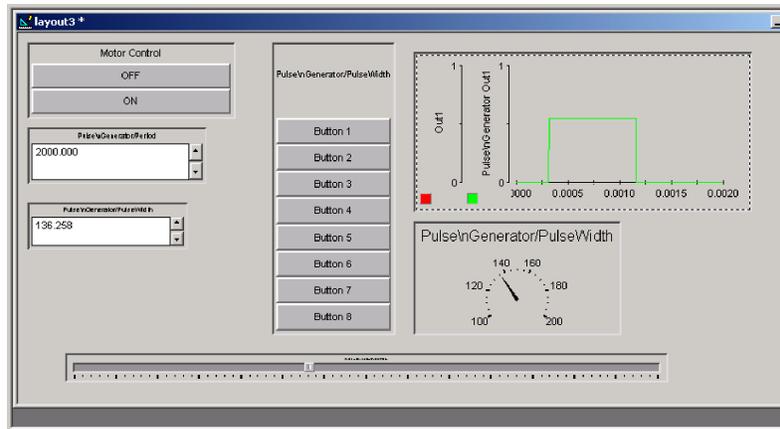
Figure 3.30: The motor control output layout.

numerical input is used for exact pulse-width values. Buttons are also used, set at exact pulse-widths.

3.6.2 Electronic Speed Controllers

Computer-interfacing with the electronic speed controllers makes it possible to control the motors. A series of control inputs changes the controllers into programming mode. The controller uses a series of flashing LEDs and beeps to ask the user what parameters to change and how they are altered. The user then inputs different pulse-widths to communicate a response. A pulse-width of 1 millisecond indicates a no, a pulse-width of 2 milliseconds indicates a yes and a pulse-width of 1.5 milliseconds indicates a next parameter response.

The electronic speed controllers are programmed to best suit the quad-rotor application. The controllers are programmed to be less sensitive to high-current draw, to prevent spontaneous motor shutoff, unsettling the flight dynamics. The electric braking function is disabled, as it has no use in hovering applications. The throttle setting is changed from auto-calibrate to a fixed configuration to ensure that the control inputs always correspond to the same throttle setting, crucial for consistent operation. The timing advance parameter is set to low producing slightly less thrust. However, the motors run cooler, with less amperage draw, will increase the flight lifetime. This is done because the thrust produced is much more than is necessary for hover, so any added efficiency will

only increase flight time. When battery voltage approaches a minimum specified value, the controllers cut off power from the motors. This changes the cutoff from a hard cutoff to a soft cutoff, which gradually powers down the motors, as opposed to an abrupt cutoff, causing the quad-rotor to plummet. Conversely, the start setting has changed from a soft start, to a hard start, allowing the motors to respond more quickly to inputs from the control board.

Chapter 4

Micro Controllers

4.1 Types of Micro Controllers

Autonomous flight calls for a number of different requirements. The vehicle must be able to maintain flight stability; it must be aware of its position relative to some coordinate system; it must have the ability to safely translate its position and change its orientation; it must have the ability to perform a specific mission; finally it must be able to land safely once it has completed its mission or commands. For this to be possible numerous sensors must be used, which include gyroscopes, accelerometers, a positioning system, as well as any mission-related sensors. A microprocessor must be able to interpret the inputs from the sensors and feed the appropriate commands to each motor individually. This must be done thousands of times per second, and the clock speed of the processor must be high, greater than 1 Mhz. The processor must also be lightweight and have a low power requirement. With these requirements in mind a number of different processor options were considered and are outlined below.

4.1.1 CuBLOC

For the quad-rotor to fly autonomously, it had to be able to control the motors individually, determine its location at any given time, and correct for any discrepancies with the given course. This called for a control system with a relatively high number of inputs and outputs, some of which had to have a PWM (Pulse Width Modu-

Figure 4.1: CuBLOC 280 Microcontroller.

lation) ability, to control the motors accurately. In a search for the perfect microcontroller initially, CuBLOC

Programmable Logic Controllers (PLCs) were chosen as a potential brain of the quad-rotor. However, upon a

more careful examination and after running some tests on the controllers, they were ruled out as being too slow

and primitive for our goals for the quad-rotor.

Another potential microcontroller for the craft is the MSC430 designed by texas instruments. It is a popular

microchip that is frequently used in robotic applications. This particular microcontroller was suitable for all our

needs and was fast enough to be utilized as a main controller on the quadrotor. The challenge in this case was

designing a circuit for the control of the aircraft and designing the code for the MSC430 to run.

4.1.2 MSP430

The microcontroller we decided to use for the project is the MSP430f1611 low-power chip made by Texas

Instruments. This microcontroller has (among other features) an $I2C$ interface, a 16-channel multiplexed 12-bit

ADC, a timer with seven capture-compare blocks (used for generating PWM signals without using CPU cycles).

The MSP430 Family of microcontrollers is ideal for wireless RF or battery applications. Its power requirement

can be as low as 1 microampere. At maximum power consumption, MSP430 can run at 16MHz. The particular

chip used in our application was an MSP430x1xx family chip which was a basic MSP430 without an embedded

LCD controller which required an external crystal to operate properly. For future development in the project, an MSP430x2xx family might be considered as that particular family operates at even lower power than the x1xx family without the necessity of an external crystal. The crystal oscillators were previously discussed in Section 3.2.8.

4.2 Hardware

4.2.1 Microcontroller

The microcontroller we decided to use for the project is the MSP430f1611 low-power chip made by Texas Instruments. This microcontroller has (among other features) an I2C interface, a 16-channel multiplexed 12-bit ADC, a timer with seven capture-compare blocks (used for generating PWM signals without using CPU cycles).

4.2.2 Coprocessor

Originally fixed-point math was going to be used on the MSP430, but it proved to be infeasible. Thus, a math coprocessor was added to run the modeling of the movement of the quad-rotor.

The math coprocessor does 32-bit floating-point arithmetic and matrix manipulation. Primarily, it is being used for matrix multiplication, sine, and cosine to compose differential rotation information from the sensors, rotate the sensor data into the inertial frame, and integrate. It runs off an internal 27MHz clock, and can perform approximately 100,000 floating-point operations per second.

The coprocessor communicates with the microcontroller via the I2C bus described in section 4.3.1. The address for the coprocessor is 100 (0x64) and the first byte sent in a write is a register number. A read is generally preceeded by a write to communicate which register is to be read.

4.3   Interfaces

4.3.1 I^2C

The I^2C interface was made available for expansions in the original design of the board and is used to connect the math coprocessor to the microcontroller. The I^2C bus uses a two-wire configuration, with one line carrying a data signal and the other carrying a clock signal. Both lines are configured as a wired-and, driven low by a chip signalling a zero and pulled up via a resistor when none is doing so. The data line is generally not permitted to change state while the clock is high, except in the case of a start condition (i.e. a master initiating a transaction). The packet consists of a start, seven bits of address, a read/write bit, and an arbitrary number of bytes of payload.

4.3.2   PWM

To control our on-board brushless motors via their controllers, a varying PWM signal is necessary. The PWM outputs are each, a square wave of 50 hertz with a high period of between 1 and 2 milliseconds, where 1ms is zero throttle, and 2ms is full throttle in a default configuration, for a fixed wing aircraft. Later on, the default setting was altered for one more suitable for a rotary wing aircraft, with a zero throttle at 1.3 milliseconds and full throttle at and above 1.8 milliseconds. Typically, for a servo actuator or reversible motor, 0-1 milliseconds signals are used for reverse motions of the motors rotor however, the Phoenix 10 motor controllers are not suited for that purpose.

4.3.3   UART

The Universal Asynchronous Reciever Transmitter is how the XBee module and the processor communicate. The UART sends a stream of bits without a clock signal (hence, asynchronous). It does this by sending a start bit, then eight data bits, then two stop bits. Timing is achieved by the recieving end listening for the start bit at all times, and both ends knowing the frequency to expect bits at. This does, however, make timing critical, as if the bit rate is off, then at best garbled data will be sent through.

4.3.4 Analog to Digital Converter

An analog to digital converter is an integrated circuit that converts any continuous signals into discrete-time digital signals. The other interface that he MSP430 used is the 12-bit analog-to-digital converter that it uses to read the inputs from the sensors.

4.4 Software

4.4.1 Processor Software

The built-in processor software primarily serves to coordinate between the various peripherals of the quadrotor; the same 50Hz pulse that generates the PWM signals for the motors controllers, triggers a number of actions in order to control the craft properly. First trigger is the transfer of the results from the control routine from the uM-FPU to the registers for the PWM output blocks. Then, it initiates the transfer of the sensor data and code for the model and control routines to the uM-FPU. Lastly, debugging information is sent to the computer via the X-Bee module. More sophisticated networking could also be done here or asynchronously via interrupts.

4.4.2 On the coprocessor

The coprocessor runs two routines, one which estimates the current location of the quad-rotor by composing the differential rotation from each tick with a running total, rotating the accelerometer data into the inertial frame, and

$$W_{t+1} = W_t \Delta W$$
$$X''_{t+1} = W_t X''_b$$
$$X'_{t+1} = X'_t + X''_t \Delta t$$
$$X_{t+1} = X_t + X'_t \Delta t$$

where
$\quad W \quad$ is the rotation from body coordinates to inertial coordinates,
$\quad \Delta W \quad$ is the rotation over this tick reported by the gyros,
$\quad X \quad$ is the position of the craft in inertial coordinates,
$\quad X''_b \quad$ is the acceleration reported by the accelerometers in body coordinates.

The other routine is an implementation of the control scheme detailed in section **??**.

4.4.3 How-To

To change the controller to use a different control scheme, the control routine for the uM-FPU should be edited. The control scheme is in the file controller.c, and is in uM-FPU assembly defined beginning with

```
char control_fpu[]={
```

and ending with the associated close brace. The state information should be in registers 115-127, with translational variables, then rotational, and then the derivatives thereof in the same order. The PWM signals should be stored as integer measurements in microseconds of duty cycle in registers 10-14; the microcontroller reads these registers to set the PWM output signals.

Similarly, the model code is in the file model.c and extends from the line

```
char model_fpu[]={
```

and the sensor inputs are loaded into registers starting at 114 and extending downwards before it is started.

Ideally, the uM-FPU code should be written for a compiler to it's assembly, but the existing compiler produced by micromega does not support the matrix math instructions of the uM-FPU and as such is not yet useful for this project.

4.5 Results

4.5.1 Computational requirements

Ultimately, the MSP430 proved insufficient on its own for handling the processing of sensor data, modeling, and control required for this project. While the MSP can execute upwards of 1 million instructions per second when running on the 8MHz clock, it lacks a hardware divider, and has only integer data types built-in, so each operation on an approximate real number takes between ten and fourty instructions.

During testing an activity indicator on the board showed that the processor was not completing a previous cycle before the next - which is to say, that the processor was insufficient for the task being put to it. This led

to the incorporation of the uM-FPU to take load off of the main microcontroller, but this was late in the project and difficulties integrating it and primarily with debugging the board as a whole proved sufficient to overrun the available time.

4.5.2 Clocking difficulties

One of the primary difficulties in getting the system to function properly was getting the 8MHz clock source to function properly. Given that the PWM outputs and the UART link to the radio rely on accurate timing, and that extra hardware was not included to provide a more reliable debugging connection to a computer, the amount of debugging necessary was too much to handle given the allotted time for the project. Complicating the matter was the clock which was malfunctioning at random times. The code to run the UART, therefore, never quite executed right at times, even when the code ran properly; the clock would cause it to be interpreted as an error by the handler. The issue with the clock is still not resolved, due to the limited high-frequency electrical background of the team.

Chapter 5

Conclusion

The goal of this project was to develop an autonomous four-rotor hovering device. Although autonomous flight was never achieved, the aerospace component of our project was completed successfully. Our project started as a handed-down, cheaply-made toy that was thoroughly tested, and determined to perform insufficiently to satisfy our project's requirements. It was completely redesigned, bearing only a slight likeness to the original craft. The original craft had a payload of 160 grams, with an operational endurance of 2 minutes, while the redesigned craft has a payload of slightly less than half a kilogram, with an operational endurance of 15 minutes. With no useful payload, such as in a hover test, the calculated endurance time is approximately 45-50 minutes. The dynamics equations for a quad-rotor were derived, and from these, a control scheme was conceived, developed, and simulated in MATLAB. This system was capable of outputting control signals to all four motors, simulating autonomous flight. An electronics package was developed that was capable of using this control system, with MEMS gyroscopic, inertial and ultrasonic instruments to calculate the required control output to the motors. The only remaining steps required for autonomous hovering flight are translating and compiling our control scheme from MATLAB code into a programming language supported by our flight control board, followed by final tweaking of the constants used in our equations for optimal flight behavior. Unfortunately, the latter depends on the former, which is out of the scope of our capabilities. This final development is more suited for computer science students than aerospace engineering students. In addition to flight testing, further development is needed

for the project to be effective in a heterogeneous sensor network. No navigation system is required for steady hovering flight, however it become crucial when translational movement is required.

5.1 Navigation and Guidance System
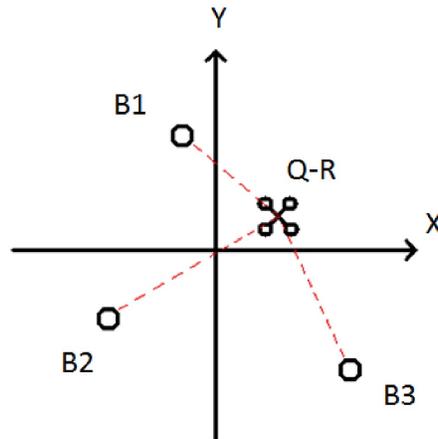
Figure 5.1: Navigation system on the (x, y)-axis.

Three systems were considered for the positioning system of the quad-rotor. The positioning system for the quad-rotor in this project is not completed past the initial prototypes. Therefore the main area to development is the positioning system.

5.1.1 Ultrasonic and RF

The first positioning system to consider is an ultrasonic and RF triangulation system. In this design the quad-rotor is the base station that sends both a RF and an ultrasonic pulse several times a second to at least three separate beacon stations, as seen in Figure 5.1. The beacon stations receive the RF signals first and start a time measurement, which stops when the ultrasonic signal is recieved. The difference in time between each signal reaching each station is used to determine the distance the quad-rotor is from each station. The RF signal travels to the stations in such a short time it can be assumed instantaneous, as it would take roughly $2.00138\,\mathrm{E}\,10^{-8}$
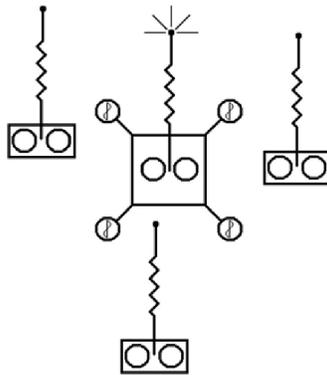
Figure 5.2: Navigation system Diagram.

seconds to travel 6 meters, roughly the length of an average room. The ultrasonic signal takes roughly 0.01744 seconds to travel the same distance under normal atmospheric conditions. Using modern equipment, it will be relatively easy to determine the distances within a reasonable accuracy. The stations must have a temperature sensor to accurately calculate the speed of sound. The distances calculated will represent three different radii, which symbolize circles around each beacon that the quad-rotor could be located. By determining the intersection point of all 3 circles combined with knowing each beacons' position and the altitude of the quad-rotor they would determine the quad-rotors exact 2D position.

5.1.2 Ultrasonic or RF

In addition to using both the ultrasonic and RF signals at the same time they can be used separately. Instead of calculating the difference in time between the ultrasonic and RF signal reaching the beacons, the quad-rotor has a receiver to calculate the time it takes for either signal to bounce off a wall or other solid objects. This method is cheaper than using the signals together, but it will be less accurate and may have increased error. If there are no walls to reflect the signal the quad-rotor has no reference point to position itself or if sound bounces off oddly shaped objects or down hallways and a second pulse of sound is sent out before the first pulse is received, errors

may occur. Accuracy may be decreased due to an increase in time for the signal to reflect off a solid object verses the time it takes two signals to reach a beacon requires a decreased number of pulses, limiting the quad-rotor's ability to determine its position. Another disadvantage is having a receiver on the quad-rotor increases its weight. Some advantages are the systems ability to detect intruding objects and work in areas where there are a number of objects to avoid. Also, having a completely self contained positioning system and not having to use beacon stations is a plus.

5.1.3 Camera

The final positioning system is recommended by the project team. It uses a camera mounted on the iRobot® and a camera mount that can determine its direction and angle in reference to the iRobot®. Since the iRobot®uses its own sensors and program to determine its position, it can calculate the position of the quad-rotor in reference to itself. It can track the quad rotor by putting several LEDs on the craft. The distances between the LEDs that is recorded by the camera are used to determine its distance. The distance measurement combined with the angle of the mount and direction of the mount is enough information to determine the height and position of the quad-rotor. Some disadvantages to this system is the need for light, which will effect out door operations when the sun is out. Also, the distances between the LEDs can be difficult to calculate because the direction the aerial craft is facing continually changes. A third disadvantage is the iRobot® must be in direct line of sight with the quad-rotor. The advantages are the technology and components are commercially available and cost less than the positioning systems discussed in Section [**?**] and [**?**]. Also, the programing is less mathematically complicated, therefore easier to create. The camera system was deemed the best positioning system because and the iRobot® is a very versatile unit, it is easy to implement., and it has an excellent position system already in place.
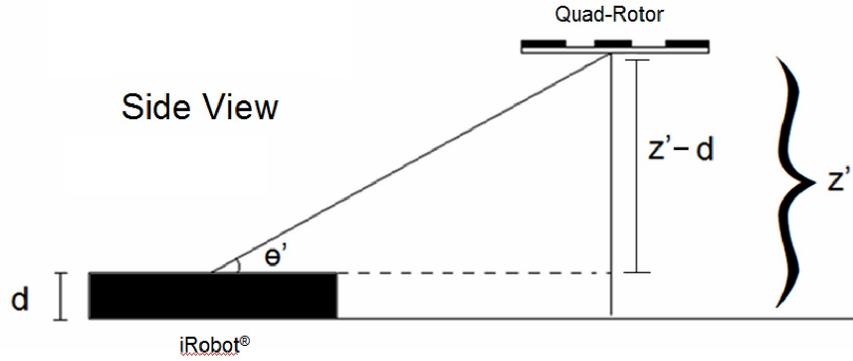
Figure 5.3: Navigation system from a side view.

5.2 iRobot® Communication System

The next step in subsequent projects should be communication of the quad-rotor with other systems. This project's communication platform is designed with this in mind. An iRobot® is the chosen system, though with a whole new year of work perhaps an additional quad-rotor would be more interesting. Some general calculations for the iRobot® to quad-rotor communications lead to the following equations,

$$x = \frac{(z - d)}{\tan(\dot{\theta})} \tag{5.1}$$

where Eq. 5.1 is related to Figure 5.3, and Equation 5.2 is related to Figure 5.4.

$$y = x \tan(\dot{\phi}) \tag{5.2}$$

5.3 Future Suggestions

In every project there are many obstacles to block a teams progress. It takes hard work and many hours to overcome any one of them. Any project to develop an autonomous vertical take-off and landing (VTOL) aircraft calls for a number of electrical and computer engineering students to be successful. At times, help will be required from an outside department. Keep in mind that professors in every department are always willing to help with the projects. Parts should be planned, machined, and ordered in advance. Once the components are
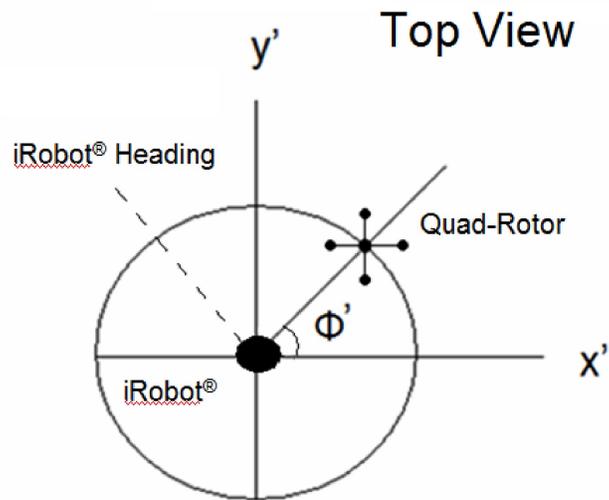
Figure 5.4: Navigation system from a top view.

gathered and things machined leave plenty of time for calibration, vibration reduction, and other unforeseen issues because they will come up and take a long time to solve. Most MQPs are designed to test what a team can do and often teams will not step up to the task, but if willing and if one has the foresight to plan and work ahead, great accomplishments can be achieved.

Bibliography

[1] Castillo, P., Dzul, A., Lozano, R., *Modelling and Control of Mini-Flying Machines*, Berlin: Springer, 2005.

[2] Castillo, P., Dzul, A., Lozano, R., *Real-Time Stabilization and Tracking of Four-Rotor Mini Rotorcraft*, **IEEE Transactions on Control Systems Technology**, Vol. 12, No. 4, July 2004.

[3] Castillo, P., Dzul, A., Lozano, R., *Stabilization of a Mini Rotorcraft with Four Rotors*, **IEEE Control Systems Magazine** December, 2005.

[4] Castillo P., Lara D., Lozano R., Sanchez A., *Real-Time Embedded Control System for VTOL, Aircrafts: Application to stabilize a quad-rotor helicopter*, **Proceedings of the 2006 IEEE Conference on Control Applications** Munich, Germany, October 4-6, 2006.

[5] DiBona C., 1. $http://egofood.blogspot.com/2006/02/silverlit-x-ufos-gyro-issues.html$.

[6] Escareño J., Lara D., Lozano R., Salazar-Cruz S., *Embedded Control System for a Four-Rotor UAV*, **International Journal of Adaptive Control and Signal Processing**, 21, 189-204, 2007.

[7] Mahony, R., Pounds, P., *Small-scale Aeroelastic Rotor Simulation, Design and Fabrication*, $http://www.araa.asn.au/acra/acra2005/papers/pounds.pdf$, 2005.

[8] Özhüner Ü., Xu R., *Sliding Mode Control of a Quadrotor Helicopter*, **Proceedings of the 45th IEEE Conference on Decision & Control** San Diego, CA, USA, December 13-15, 2006.

[9] $http://www.medusaproducts.com/motors/calculate.asp$.

[10] $http://avia.russian.ee/helicopters_eng/bothezat.php$.

[11] $http://avia.russian.ee/helicopters_eng/oemichen.php$.

[12] $http://www.cradleofaviation.org/exhibits/jet/quadrotor/index.html$.

[13] $http://asl.epfl.ch/aslInternalWeb/ASL/publications/uploadedFiles/325.pdf$

Appendix A

Microprocessor and MatLAB Code

A.1   MatLAB

```
%File QRODEF2.m

function dx=QRODEF2(t,x)

global m g az1 az2 apsi1 apsi2 psid I Ixx Ixy Ixz Iyx Iyy Iyz Izx Izy Izz xd yd zd z

%%Convention
% x=1
% y=2
% z=3
% psi=4
% theta=5
% phi=6
% xdot=7
% ydot=8
% zdot=9
% psidot=10
% thetadot=11
% phidot=12

%1,2,3,4 CCW


%Controllers

r1=-az1*x(9)-az2*(x(3)-zd);
u=(r1+m*g)/(cos(x(5))*cos(x(6)));

tau_tilde_psi=-apsi1*x(10)-apsi2*(x(4)-psid);
```

```matlab
tau_tilde_theta=-sat(x(11)+sat(x(5)+x(11)+sat(2*x(5)+x(11)-x(7)/g+sat(x(11)+3*x(5)-3
tau_tilde_phi=-sat(x(12)+sat(x(6)+x(12)+sat(2*x(6)+x(12)+x(8)/g+sat(x(12)+3*x(6)+3*(

tau_tilde=[tau_tilde_psi ; tau_tilde_theta; tau_tilde_phi];

%QR Dynamics
dx=[x(7:12);...
    -u*sin(x(5))/m;...
    u*cos(x(5))*sin(x(6))/m;...
    u*cos(x(5))*cos(x(6))/m-g;...
    tau_tilde];


%File: QRSOLVEF2.m

clc
close all
clear all


global m g az1 az2 apsi1 apsi2 psid I Ixx Ixy Ixz Iyx Iyy Iyz Izx Izy Izz xd yd zd z


%Moments of Inertia (kg*m^2)
Ixx=.00399055277;
Ixy=0;
Ixz=0;
Iyx=0;
Iyy=.00415903126;
Iyz=0;
Izx=0;
Izy=0;
Izz=.00790537454;

%Contants
m=0.650;
g=9.81;

%Initial Conditions
x0=0;
y0=0;
z0=1;
psi0=0;
theta0=0;
phi0=0;
```

```
xdot0=0;
ydot0=0;
zdot0=0;
psidot0=0;
thetadot0=0;
phidot0=0;
x0=[x0 y0 z0 psi0 theta0 phi0 xdot0 ydot0 zdot0 psidot0 thetadot0 phidot0];

%Desired Conditions
xd=0;
yd=0;
zd=1;
psid=pi/2;

%Z Control
scale_z=3;
zeta_z=1;

ts_z=scale_z*abs(zd-z0)+1;
omega_z=4/(ts_z*zeta_z);
az1=2*zeta_z*omega_z*m;
az2=m*(omega_z)^2;

%Psi Control
scale_psi=10;
zeta_psi=1;

ts_psi=scale_psi*abs(psid-psi0)+1;
omega_psi=4/(ts_psi*zeta_psi);
apsi1=2*zeta_psi*omega_psi;
apsi2=(omega_psi)^2;


%Translational Sim
%addtime1=1;
%time_end1=3*ts_z+addtime1;
time1=[0:.1:25];
[t,x]=ode45(@QRODEF2,time1,x0);
figure('Name','QR Sim Trans','NumberTitle','off')
plot(t,x(:,1:3))
hold on
plot(t,x(:,7:9))
xlabel('Time (s)','FontSize',11)
ylabel(' ','FontSize',11)
title('QR Sim Translational','FontSize',14)
```

```
h = legend('x','y','z','xdot','ydot','zdot',1);
%set(h,'Interpreter','none')

%Rotational Sim
figure('Name','QR Sim Rot','NumberTitle','off')
plot(t,x(:,4:6))
hold on
plot(t,x(:,10:12))
xlabel('Time (s)','FontSize',11)
ylabel(' ','FontSize',11)
title('QR Sim Rotational','FontSize',14)
h = legend('psi','theta','phi','psidot','thetadot','phidot',1);
%set(h,'Interpreter','none')


for i=1:length(t)

    r(i)=-az1*x(i,9)-az2*(x(i,3)-zd);
    u(i)=(r(i)+m*g)/(cos(x(i,5))*cos(x(i,6)));

    %tau_tilde (psi, theta, phi)

    tau_tilde(1,i)=-apsi1*x(i,10)-apsi2*(x(i,4)-psid);
    tau_tilde(2,i)=-sat(x(i,11)+sat(x(i,5)+x(i,11)+sat(2*x(i,5)+x(i,11)-x(i,7)/g+sat
    tau_tilde(3,i)=-sat(x(i,12)+sat(x(i,6)+x(i,12)+sat(2*x(i,6)+x(i,12)+x(i,8)/g+sat


    %tau_tilde=[tau_tilde_psi; tau_tilde_theta; tau_tilde_phi];

    W=[-sin(x(i,5)) 0 1; cos(x(i,5))*sin(x(i,4)) cos(x(i,4)) 0;
cos(x(i,5))*cos(x(i,4)) -sin(x(i,4)) 0];
    I=[Ixx Ixy Ixz; Iyx Iyy Iyz; Izx Izy Izz];        %Converted to kg*m^2
    J=W'*I*W;

    Cor(1:3,i)=[(-(-x(i,11)*cos(x(i,5))*Ixx+(-x(i,11)*sin(x(i,5))*sin(x(i,4))+x(i,10
 (x(i,5))*cos(x(i,4))-(-sin(x(i,5))*Ixz+cos(x(i,5))*sin(x(i,4))*Iyz+cos(x(i,5))*cos(
            (-(-x(i,10)*sin(x(i,4))*Iyx-x(i,10)*cos(x(i,4))*Izx)*sin(x(i,5))+(-x(i,10
            (-x(i,11)*cos(x(i,5))*Ixx+Ixy*(-x(i,11)*sin(x(i,5))*sin(x(i,4))+x(i,10)*c


    tau(1:3,i)=Cor(1:3,i)+J*tau_tilde(1:3,i);
    tauj(1:3,i)=J*tau_tilde(1:3,i);

    F1(i)=[0.25 -0.6286 0 2.4566]*[u(i); tau(1:3,i)];
    F2(i)=[0.25 0.6286 -2.4566 0]*[u(i); tau(1:3,i)];
```

```
    F3(i)=[0.25 -0.6286 0 -2.4566]*[u(i); tau(1:3,i)];
    F4(i)=[0.25 0.6286 2.4566 0]*[u(i); tau(1:3,i)];


%       F1(i)=[0.25 10.9987 0 2.4566]*[u(i); tau(1:3,i)];
%       F2(i)=[0.25 -10.9987 -2.4566 0]*[u(i); tau(1:3,i)];
%       F3(i)=[0.25 10.9987 0 -2.4566]*[u(i); tau(1:3,i)];
%       F4(i)=[0.25 -10.9987 2.4566 0]*[u(i); tau(1:3,i)];

%       PWM1(i)=(F1(i)*.0004+.001)/.02;
%       PWM2(i)=(F2(i)*.0004+.001)/.02;
%       PWM3(i)=(F3(i)*.0004+.001)/.02;
%       PWM4(i)=(F4(i)*.0004+.001)/.02;

    PWM1(i)=F1(i)*.0004+.001;
    PWM2(i)=F2(i)*.0004+.001;
    PWM3(i)=F3(i)*.0004+.001;
    PWM4(i)=F4(i)*.0004+.001;

    %Signal Gen
%[mod1, t]= modulate(PWM1,50,201,'pwm')
%y = modulate(x,fc,fs,'method',opt)
%[y,t] = modulate(x,fc,fs)


%pwmout= [t' (mod1'*5)]

end



figure('Name','PWM','NumberTitle','off')
plot(t,PWM1,'-r');
hold on
plot(t,PWM2,'-b');
hold on
plot(t,PWM3,'-r*');
hold on
plot(t,PWM4,'-b*');
hold on


xlabel('Time (s)','FontSize',11)
ylabel('PWM (s)','FontSize',11)
title('PWM','FontSize',14)
```

```
h = legend('PWM1','PWM2','PWM3','PWM4',1);
%set(h,'Interpreter','none')


figure('Name','tau','NumberTitle','off')
plot(t,tau);
hold on
figure('Name','tauj','NumberTitle','off')
plot(t, tauj);
figure('Name','cor','NumberTitle','off')
plot(t, Cor);
```
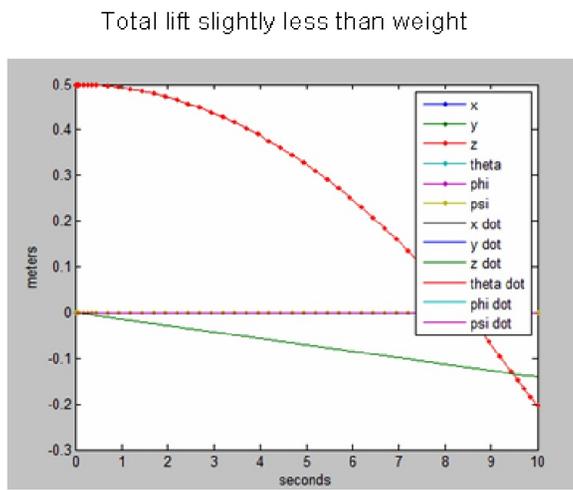

Figure A.1: MatLAB simulation of quad-rotor flight with T_{total} slightly less than weight.

Appendix B

Measurements

B.1 Coriolis Term Error for Settling Time of 10 Seconds
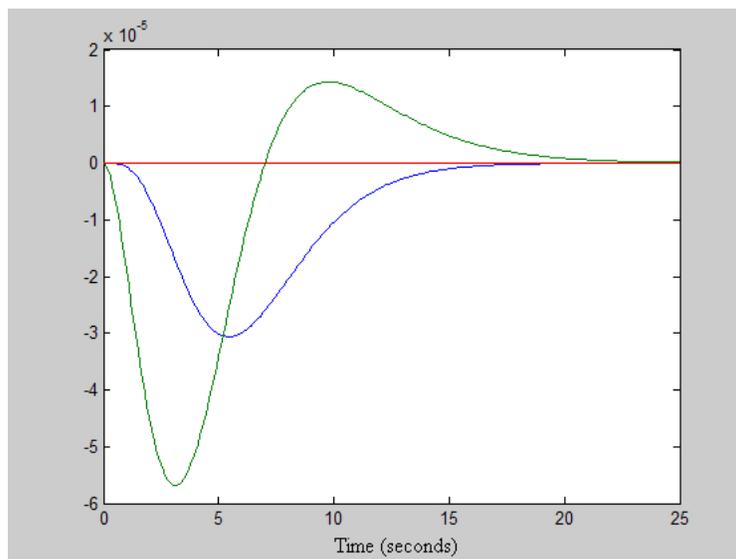
Figure B.1: Corliois term.
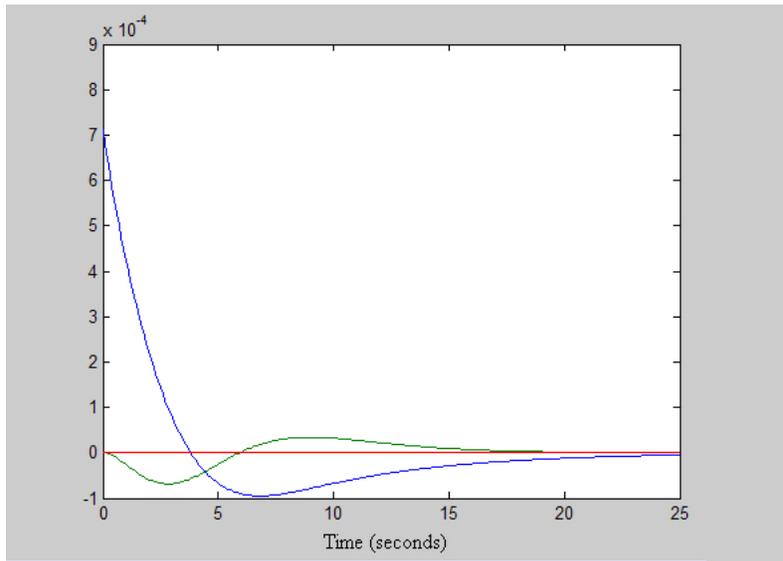
Figure B.2: τ term.
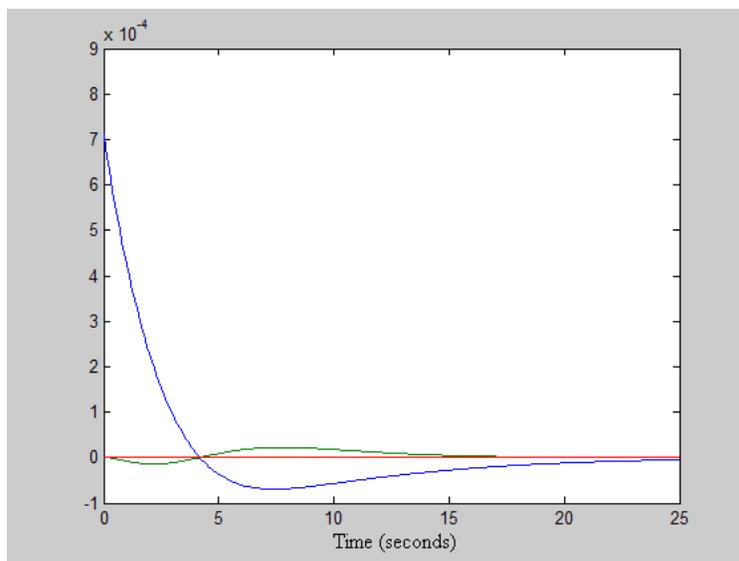
Figure B.3: $J\tilde{\tau}$ term.

B.2 Coriolis Term Error for Settling time of 5 Seconds
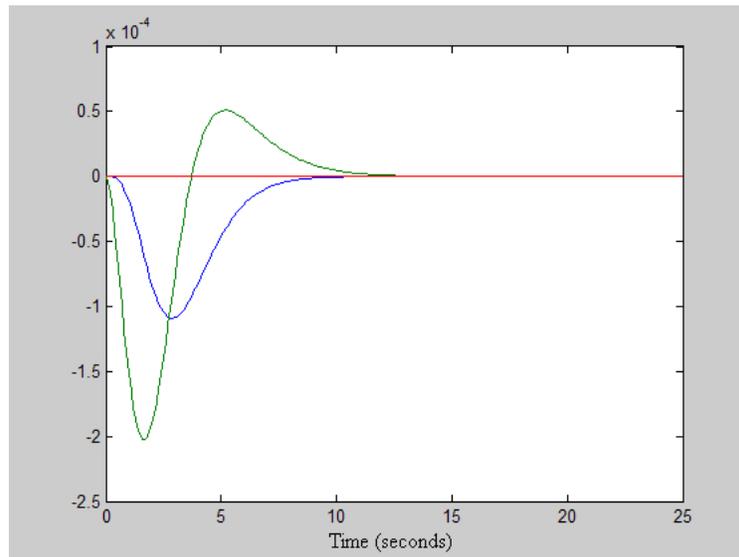
Figure B.4: Coriolis term.
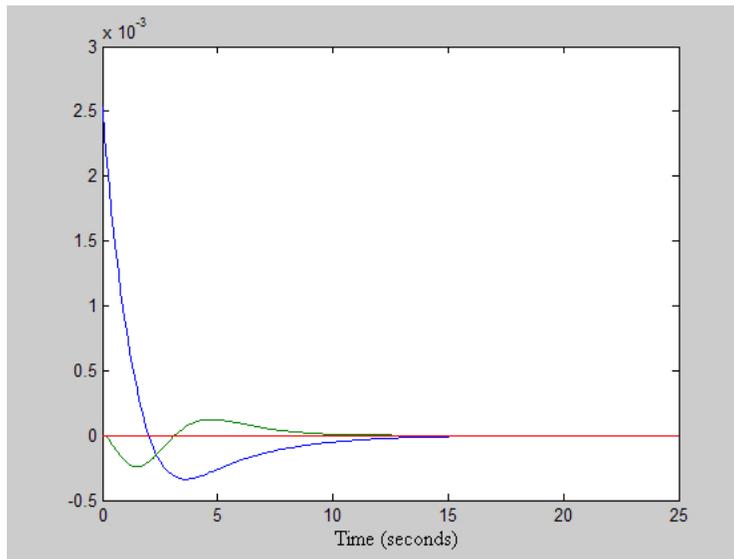
Figure B.5: τ term.
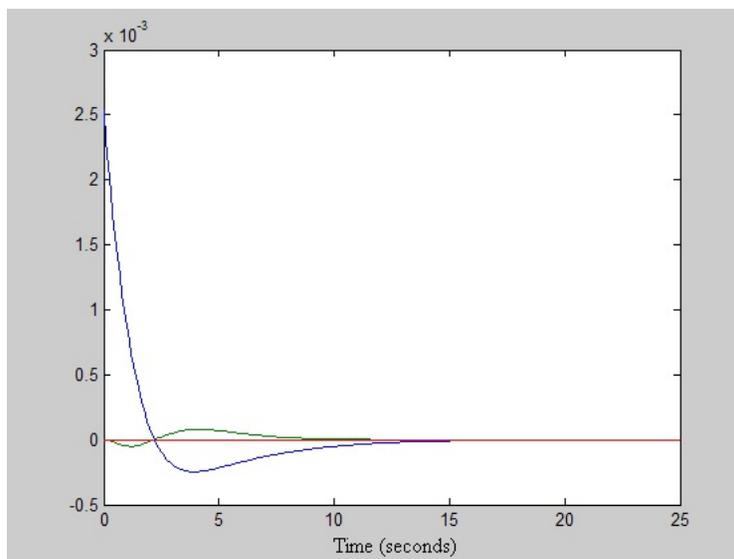
Figure B.6: $J\tilde{\tau}$ term.

B.3 Quad-Rotor Measurements

Table B.1: The weights of individual components of the quad-rotor.

Component	Weights (g)
Aluminum Test Arms	10
Frame Hub	50
Gear Box w/out Gear	12
Gear	1
Electric Speed Controllers (ESCs)	24
Gear Box Mounts	5
Propeller	17
Board	17

Table B.2: The dimensions of individual components of the quad-rotor.

Component	Length (in)	Width (in)	Height (in)
Aluminum Test Arms	48.88	0.31	0.37
Frame Hub	2.6	2.6	0.26
Gear Box w/out Gear	0.98	0.54	0.98
Gear	1	1	0.25
Electric Speed Controllers (ESCs)	1	0.68	0.18
Gear Box Mounts	0.75	0.52	0.93
Propeller	9	0.9	0.45
Board	3	2	0.065

Table B.3: Wire gauges.

Gauge	Diameter (cm)	Max Amps (open air)	Max Amps (enclosed)
22	0.064	7	5
20	0.081	11	7.5
12	0.205	41	23
10	0.259	55	33

Table B.4: Different gearboxes compared for optimal efficiency.

Gearbox Ratios	5.86:1	7:1	8.6:1
Amps	4.98	3.45	2.21
F/T Endurance 900 mA LiPo (mins)	10.8	15.6	24.4
Efficiency	80.12	81.22	78.54
Thrust (g)	436	353	264
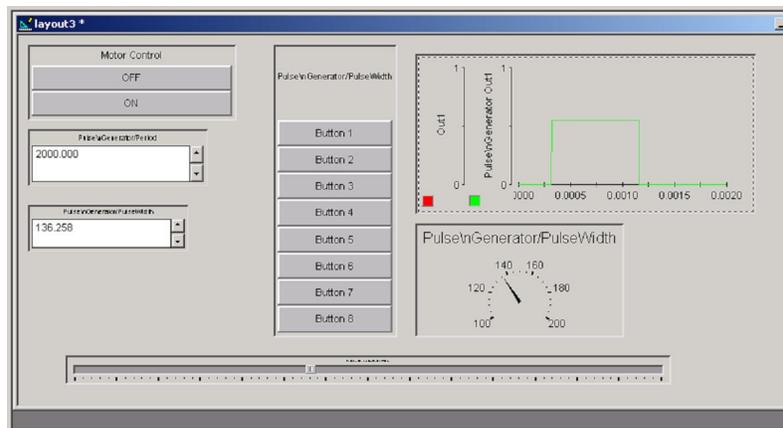
Figure B.7: Brushless motors controlled manually with dSPACE interface.