Autonomous Hoverable Flying Robot

James M. Dunfield and M. Tarbouchi
Royal Military college of Canada, Electrical and Computer Engineering Department, Canada

Abstract

In this paper the design, development and testing of an inexpensive hoverable flying robot to be capable of achieving vertical takeoff and landing (VTOL), and to be able to sustain a specified attitude at a specified height is presented. The ability to be able to autonomously navigate through a predefined path was designated for a future phase.

This project is different from most autonomous flying robots as it focuses on a four-propeller configuration. In addition, this project uses fixed pitch propellers instead of variable pitch rotors resulting in a greatly reduced cost and mechanical complexity. The downside is that this introduces significant additional challenges in the control system, as the response time of a propulsion system is much greater than that possible through a variable pitch rotor system.

I. INTRODUCTION

Autonomous flight is a relatively new research of robotics to an entirely new level [5]. The ability of autonomous flying vehicles to operate far from an observer's view has many applications in the military, civilian rescue, land mine detection and space exploration domains.

A flying vehicle of this configuration is also being researched at the Aircraft Aerodynamics and Design Group at the University of Stanford in collaboration with the National Space Administration (NASA)[1], although the focus is on a different level. In addition, some work in this field was done in [2] and [3] but it was never implemented.

This paper describes the design of the Autonomous Hoverable Flying Robot. The overall architecture will first be explained, followed by an explanation of all hardware systems. The software, including the control system, shall then be described. Finally, some of the engineering challenges which were encountered will be identified as well as the project results to date.

II. CONCEPT DESIGN

The basic concept was to create an inexpensive flying robot which would use, as much as possible, commercially available off-the-shelf parts. Various flying configurations were considered, including three-propeller designs as well as four propeller designs, with various strength propulsion systems, but for purposes of cost and simplicity of control, a four-propeller design was selected which used four equally powered propulsion systems mounted symmetrically around a frame, as seen in Figure 1. With the four-propeller design, two propellers rotate in one direction (left & right) and two rotate in the opposite direction (front & back). The reason for the pair of propellers spinning in opposite directions is to counter the rotational torque that is caused. Therefore, when all the propellers are spinning at the same speed, the torque from the two propellers spinning in one direction cancels out

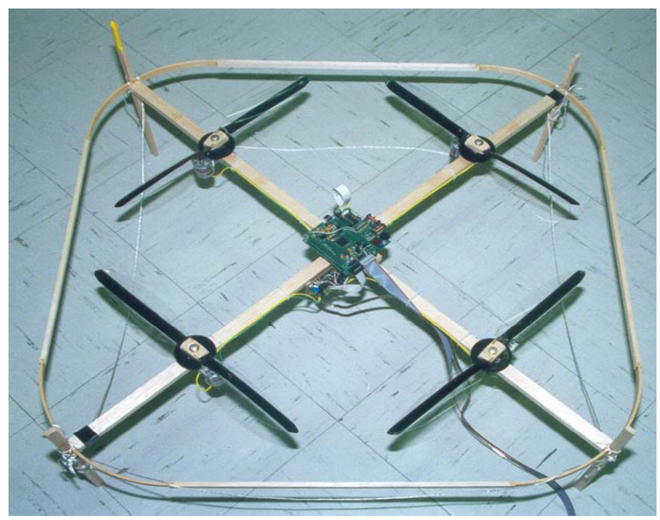

Figure 1. The Autonomous Hoverable Flying Robot

the torque of the other two propellers. The pitch and roll are controlled by adjusting the power to two opposing motors in opposite amounts. The yaw is controlled by increasing the speed to one pair of motors spinning in the one direction, and decreasing the speed to the other pair. Height is controlled by simultaneously increasing or decreasing the speed of all four propellers.

III. SYSTEM ARCHITECTURE

The system architecture diagram for the Autonomous Hoverable Flying Robot is shown in Figure 2 and displays how the various modules are interconnected to the main control module. The frame is the support structure for all other physical modules. The Power Supply module takes the 12V from the battery and supplies all the modules with the required power. The majority of the sensors are incorporated onto a single board. This board is attached as a daughter board via 50 pin connectors directly under the microcontroller board, in order to conserve space and weight, as shown in Figure 3. All external connections from the daughter board are accomplished through the use of three cables: the motor driver cable; the base station/RS232 cable; and the sonar range finder cable. The modules on this board are described later in this paper.

IV. FRAME

The purpose of the frame is to provide a superstructure to which all hardware modules are attached, and to provide a safety shield. Although various materials were considered, balsa wood was used because of its very high strength to weight ratio and ease of use. The frame extends beyond any rotors for the safety of personnel and property. At each propeller and motor site, a hole was drilled through the frame for the propeller axle. Into this hole, a hardwood sleeve was inserted which holds the bearings, as seen in Figure 4. The sleeve was necessary to withstand the vibrations of the bearings. A motor bracket was also specially designed and constructed to fasten the motor to the frame. The purpose of this bracket was to support the motor, provide a method of adjustment for gear fitting and to provide air ventilation for motor cooling. The microcontroller and daughter board are mounted on the top and middle of the frame. The sonar transducer is located underneath the frame, directly under the microcontroller board. Finally, the motor driver board and sonar range finder board are mounted vertically on one of the main cross struts, near the middle.

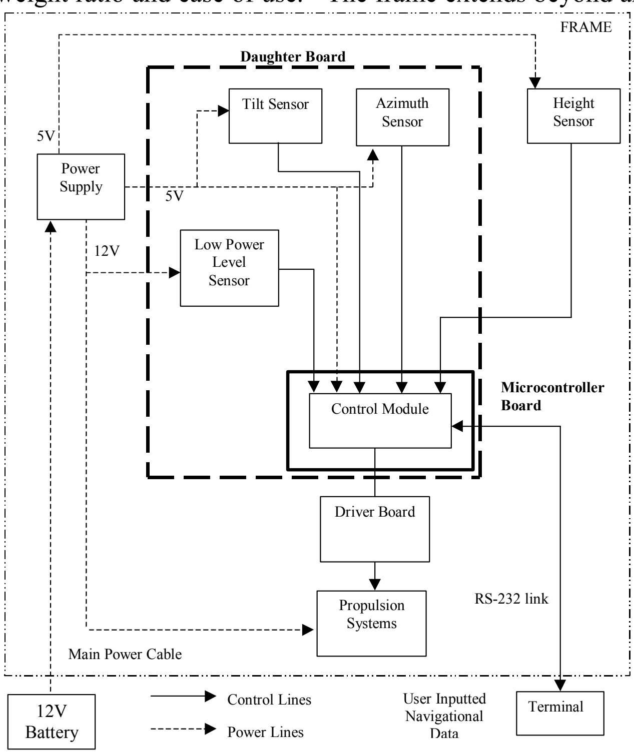

Figure 2. System Architecture

V. PROPULSION SYSTEM

Gas and electric motors were considered, but for purposes of ease of testing and control, electric motors were selected. Furthermore, electric propulsion systems are suitable for inside use, and the noise level produced is about 73dB at 2m which is much better than for gas powered systems.

Designing a propulsion system that is capable of lifting its own weight and that of the battery system and electronic payload requires a very delicate balance between a large set of factors. Although they were much more efficient, brushless motors were not selected due to their extreme cost and increased weight. The performance of triple and quadruple bladed propellers, along with ducted fans, were also analysed, but all produced significantly lower thrust than the two bladed configurations. The cost of the propellers was a significant factor as the larger propellers produced much greater thrust, but were also substantially more expensive.

Assumptions were made for all component weights and that the maximum tilt angle would be 10^{o}. The platform was designed to have a lift capacity of 1500 g, which allowed for 280 g of control electronics.

The most general equation for the power absorbed by the propeller is [4]:

$$Pwr = K_p * D^4 * P * Rpm^3$$

where:
D = the propeller diameter (length) in feet;
P = the pitch of the propeller in feet; and
Kp is the propeller constant, which equates to propeller efficiency.

The software analysis tool, MotoCalc, was used for the design of the propulsion system. It simplified comparisons among various factors by using tables and graphs. With this tool, analysis was possible using the characteristics of actual motors and other components which would be used in the final product. The final configuration, as seen in Figure 4, uses a Graupner Speed 400 7.2V #1794 motor, and a Zinger

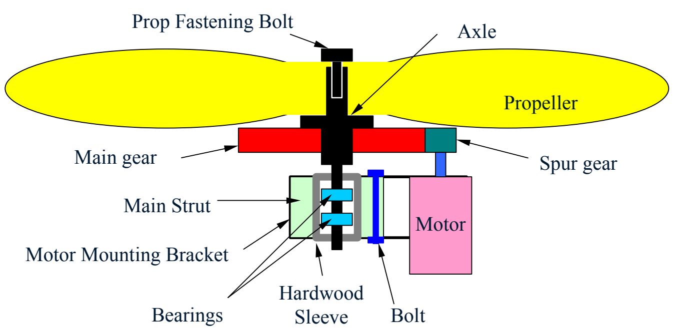

Figure 3. Propulsion System

14 X 8 two-blade propeller with a gear reduction ratio of 6:1.

The control of the motors is accomplished through the use of 5th generation Logic Level Power MOSFETs. These particular MOSFETs were used as they possessed a very small R_{DS} (about 0.006 Ω). This value is important, as the motors are supplied with a very large current(6-11A). The pulse width modulation signals from the microcontroller are connected to optoisolators which isolate the signal and convert it to 0-12V level, for an improved V_{GS}. A pull down resistor is also connected to each gate to avoid the gate floating to its on position.

VI. SOFTWARECONTROLSYSTEM ARCHITECTURE

The control system architecture for the Autonomous Hoverable Flying Robot is composed of three major software modules. The first is the Initialization Module, which sets up the processor and all the sensors for start-up and acquires the flight parameters. The second module is the Behaviour Control Module (BCM) which functions like the captain of a ship deciding what course of action to take, and calling out orders to the crew. This module is continuously running during the operation of the robot. The final module is the

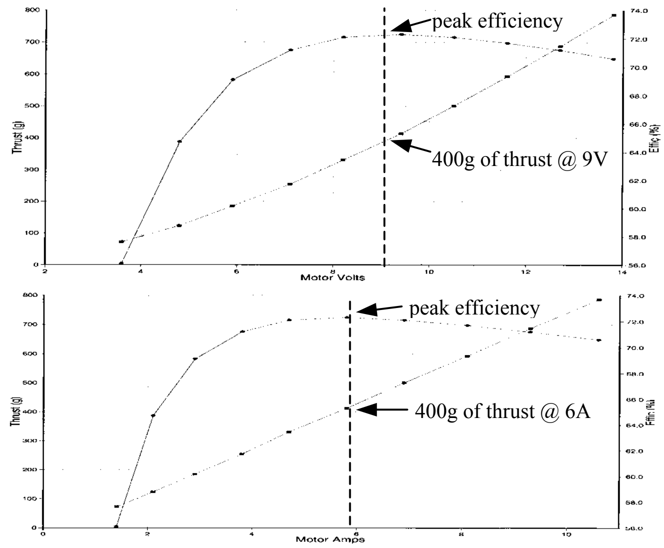

Figure 4. Thrust and Efficiency Versus Voltage and Current

Fuzzy Control Module (FCM). It functions like the ship's crew, carrying out the orders of the Captain. This module incorporates all the feedback and control systems using data from the sensors to provide the stabilization necessary for the robot to maintain stable flight. The FCM functions completely independently of the BCM, although it is turned on and off by the BCM. The only interaction between the two is that the FCM uses four variables which are amended periodically by the BCM. The FCM need not know how or when the other modules work, and vice versa. Therefore both modules can be modified as desired without any effect of the other.

VII. FUZZY CONTROL MODULE (FCM)

The purpose of the FCM is to provide stable flight for the robot. The FCM is an interrupt subroutine executed every 8msec by the Real-Time Interrupt module of the 68HC12B32 processor. As it is the only interrupt being used and its execution time is estimated at 2msec, its execution time is guaranteed. During each operation, it uses whatever values have been set for the requested roll, pitch, azimuth and height, along with the actual values received or calculated for the sensors. The module compares the actual values with the requested values and using the feedback, adjusts the speed of the four motors accordingly. At every activation the roll, pitch, tilt, and yaw are calculated and the motor speeds are updated. Every second iteration (16ms), the rate of change in tilt is calculated. Every 4th iteration (32ms), the height error and the rate of change in height are calculated. Finally, every 200ms, the azimuth data is updated from the compass.

The FCM incorporates four feedback systems, one for each axis, as seen in Figure 7. Initially, all motor values are set to a default value of half speed ($7F). The roll affects the left and right motors while the pitch affects the front and back motors. The requested and actual roll values are used to determine the error and rate of change values. These values are then fed through a Fuzzy Logic controller [8] to derive the final amount of power that must be added or subtracted from each motor. Five member functions (levels) are used for the error and rate of change inputs, during the fuzzification process and five output levels ("singletons") are used, in determining the final offset value in the defuzzification process. The same procedure occurs for the pitch control.

Next is the yaw or azimuth adjustment, which involves a small change in power to all four motors, adding to the left and right, and subtracting to the front and back. This is done proportionally according to the amount of error. The final adjustment is for the height, which uses a fuzzy-PI (Proportional Integral) logic controller. Again, the error and rate of change values are calculated from the requested and actual values. The output of the Fuzzy Logic controller is accumulated into a value used as the power offset. This offset gradually increases depending on the height of the robot, the rate of change, and direction of the change. In effect, the height control section biases the motor values up and down.

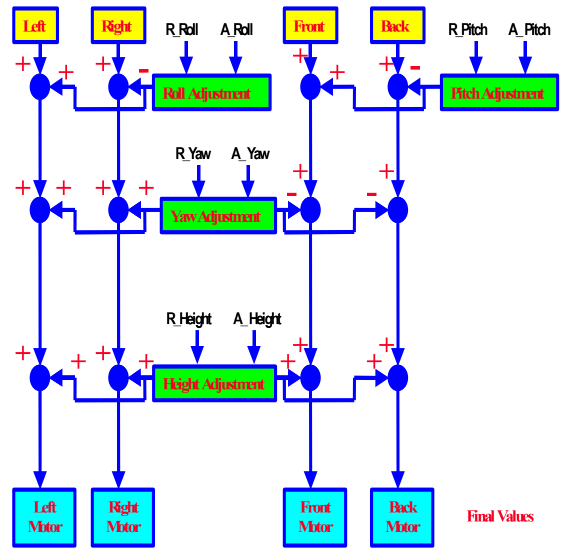

Figure 5. Fuzzy Control Module

VIII. HARDWARE AND IMPLEMENTATION CHALLENGES

The Motorola 68HC12B32 microcontroller was selected for the robot because of its many interface modules including pulse width modulation (PWM) ports, for DC motor control, and timer and analogue-to-digital (ATD) ports used for various sensors. This microcontroller also provides ample RAM(1K), EEPROM(768 bytes), and FLASH EEPROM (32K) for development and running of the necessary programs. Finally, the 68HC12B32 was the first microcontroller to have instructions specifically for performing Fuzzy Logic. A lightweight and compact controller board was selected from Technological Arts.
With the complexity of this work, many challenges were encountered in the implementation of the design. The first significant unexpected problem was that the vibrations due to the four motors, rotating at about 3000 RPM, was greatly affecting the readings by the tilt sensor. Originally the pulse width modulated output from the tilt sensor was being used, so this signal was then fed through a 2nd order active low pass filter, with a cut off frequency of 1/2 Hz. This proved to be effective in significantly reducing the affect of the vibrations.

Once stability testing of one axis was started, it became very obvious how unstable the system was. The first attempt to correct the situation was to improve the fuzzy logic controllers. The complexity of the fuzzy logic computations which were originally designed with three levels, were then increased to five for both inputs and output levels of the roll and pitch fuzzy controllers.The next improvement to the design focused on improving the response time of the control system. It was recognized that there are three main areas where delays occur: program delay; tilt sensor response delay; and propulsion system response delay.

Originally the FCM software was executed as part of the main program which resulted in a non-deterministic response time, and was over 30msec. Implementing a Real-Time Interrupt for the FCM software then solved the program delay. This guaranteed that the control program would be executed every 8mSec. The tilt sensor response delay was addressed by increasing the cutoff frequency of the LP filter. This allows the microcontroller to sense movement more quickly. In order for this filter to still efficiently cut off the higher frequencies, it was increased to a fifth order filter by switching to a fourth order Sallen Key filter and using the analogue output of the tilt sensor which added a passive filter.

The final delay, the response time of the propulsion system, was even more involved. A thorough analysis was conducted of the actual speed of the propellers in response to changes to the voltage applied to the motor. Figure 8 shows that with the original wood propellers, it took about 800mSec to ramp up for a speed increase change of 15%. Knowing that a significant portion of this was caused because of the inertia of the propeller being made of wood weighing about 32 g, an alternative was sought.

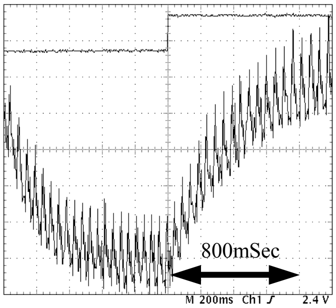

Figure 6. Wood Propeller Response

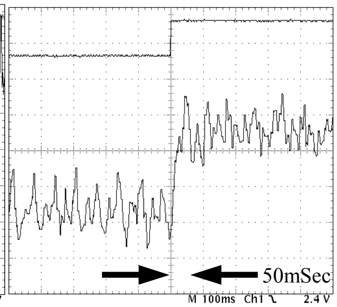

Figure 9. Plastic Propeller Response

CONCLUSION

The goal for the project was to create a robot that could vertically take off, hover at a given altitude facing a given direction, and be able to maintain a given tilt angle in order to travel in a given direction. Relative stability was achieved using fuzzy logic controller in one axis and all the remaining modules were successfully designed and implemented. Stability has not yet been achieved in two axes, although testing is still ongoing.

REFERENCES

[1] H. Bortman, "Whirlybugs", *New Scientist* 5 June 1999, p. 40-43

[2] J. Borenstein, "The HoverBot- An Electrically Powered Flying Robot", University of Michigan Advanced Technologies Lab –White Paper

[3] S. Marti, Concept Paper: "Free Flying Micro Platform and Papa-TV-Bot: Evolving Autonomously Hovering Mobots" MIT: The Media Laboratory

[4] B. Boucher, *The Electric Motor Handbook*, Astro Flight Inc. 1994.

[5] Charles, J., "CMU's autonomous helicopter explores new territory", IEEE Intelligent Systems and Their Applications V 13 No. 5 p. 85-87 Sep/Oct 1998.

[6] B. Boucher, "Understanding the Physics of Low Energy Flight", Astro Flight Inc. 1996

[7] H.R. Everett, *Sensors for Mobile Robots. Theory and Applications*. A.K. Peters, Wellesley, MA; 1995.

[8] Passino K. M., Stephen Yurkovich "*Fuzzy Control*" Addison-Wesley, 1998.