Multi-Disciplinary Senior Design Conference
Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York 14623

Project Number: 09122

MICRO AERIAL VEHICLE – PRELIMINARY FLIGHT CONTROL SYSTEM

Erik Bellandi – ME – Project Manager

Ben Wager – ME – Lead Engineer

Garrett Argenna – ME

Tahar Allag – EE

Michael Pepen – EE

Ramon Campusano – CE

Stephen Nichols – CE

ABSTRACT

The goal of the Micro Aerial Vehicle (MAV) Control System project was to lay the foundation for aircraft autonomy and build a system that will lead to an autonomous flight control system. Dr. Jeffrey Kozak, of the Mechanical Engineering Dept. at Rochester Institute of Technology, advised the project with specific product requirements and team guidance. The primary objective was to develop an adaptable system with preliminary control laws to provide stability augmentation for a given platform. The project had three major categories which were; aircraft simulations and control laws, sensors and electronics, and finally implementation of the control law and sensor calculation codes within the electronics. The final product was a custom PCB electronics board with integrated sensors and a field-programmable gate array (FPGA) with embedded dual-core microcontrollers to implement the sensor calculation and control law codes. A two-axis gimbaled test stand was also developed to accurately test the sensors and system commands.

INTRODUCTION AND BACKGROUND

Micro Aerial Vehicles had been an area of research for many years following DARPA's MAV initiative which lead to the International MAV competition. The focus of the IMAV competition was on creating very small scale vehicles that would be flow by remote and be capable of video surveillance. In 2007 the IMAV competition ended and new competitions like the European MAV competition were created to continue to push research areas for MAV's. With advancements in technology autonomous vehicles has been a large area of research and the EMAV and other competitions are now focused on autonomy as well as size however the general size definition has increased.

Based on DARPA's original definition, a MAV was any aerial vehicle with its largest linear dimension being less then 16cm (6.3in). With the new competitions and the focus more on autonomy, the general size definition has been increased to 80cm (31.5in).

RIT had been involved in MAV research since 2000 and to stay at the forefront of research and technology the MAV projects have followed the shift toward autonomy. Flight autonomy is a new area of research for students at RIT and this project is the first project to progress to the goal of autonomy.

NOMENCLATURE

ADC = Analog to Digital Converter
DARPA= Defense Advanced Research Projects Agency
ECB = Electronics Control Board
FPGA = Field Programmable Gate Array
GCC = GNU Compiler Collection
GPIO = General Purpose Input / Output
GPS = Global Positioning System
IMU = Inertial Measurement Unit
LED = Light Emitting Diode
MAV = Micro Aerial Vehicle

MIPS = Microprocessor without Interlocked Pipeline Stages
PCB = Printed Circuit Board
PID = Proportional Integral Derivative control
PLL = Phase-Locked Loop
PWM = Pulse-Width Modulation
SAS = Stability Augmentation System
UART = Universal Asynchronous Receiver / Transmitter
VHDL = VHSIC hardware description language
VHSIC = Very High Speed Integrated Circuits

CONTROL LAWS AND SIMULATION

One of the most important tasks in designing a control system is accurately modeling the system to be controlled. This is generally referred to as the plant model and it is a mathematical model of the dynamic behavior of the system. Most aircraft can be modeled using a set of generalized differential equations. These models are dependent of non-dimensionalized aerodynamic coefficients, which must be determined using an aircraft's geometry and lift/drag characteristics. The coefficients describe lift, drag and side force dynamics as well as pitching, yawing and rolling moment dynamics. These coefficients are usually determined using wind tunnel testing; however they can be approximated using the Air Force Research Lab's Digital DATCOM program and methods described in most Flight Dynamics textbooks (1). Using the latter two methods, a plant model of the 2009 micro air vehicle being designed by the P09123 senior design team was calculated and implemented in Simulink for control system simulations.

Flying aircraft can be difficult to control due to the handling qualities inherent in the plane. These qualities are rated by pilots and compiled by aircraft class and by the dynamic stability modes for aircraft. After determining the handling ratings for the MAV, the stability was augmented to achieve level one characteristics. This was done using rate feedback to introduce artificial damping to the system.

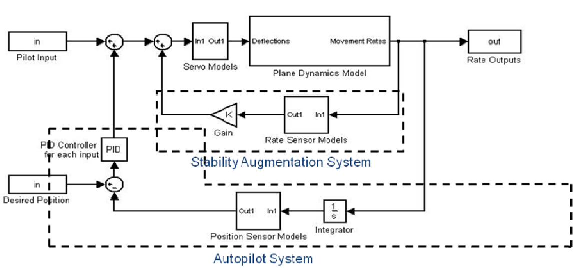

Figure 1: Control System Concept

An aircraft's attitude is its angular orientation while in flight. Using the inertial measurement unit, these angles can be measured and fed back to a controller to maintain a desired attitude. Using three PID control laws and a linearized plant model, the aircraft's pitch, roll and heading angles can be set and maintained. Simulations in Simulink have been conducted to verify the controller's performance. This is useful for waypoint navigation, as it will enable the aircraft to reorient itself after experiencing a wind disturbance.

The altitude controller is similar to the attitude control, in that it maintains a desired flight characteristic and rejects any disturbances. The control law is also a PID controller with altitude feedback to control a linearized plant model and is tuned to maintain a desired altitude. The altitude is fed back by pressure readings from a pitot tube that will be mounted on the aircraft's wing. This control law has also been simulated in Simulink to verify the desired performance characteristics.

ELECTRONICS AND SENOSORS

The EP3C16E144C8N FPGA was chosen for the implementation of the dual core controllers. The FPGA utilized a 16Mbit flash memory device for boot up configuration. To adequately accommodate the desired features of the FPGA, four separate power schemes were developed. These sources were a 1.2V, 2.5V, 3.3V, and a 5V linearly regulated power supply. The variety of power sources allows the FPGA to utilize internal PLL drivers to generate the PWM signals. The PWM signals are internally level shifted from 3.3V to 5V to ensure the FPGA is isolated from external noise and the larger power rail. The signals from the pilot are transmitted to the Radio receiver then driven in to the high power PWM input of the Electronics control board (ECB). The signals are stepped down and received by one of the cores of the FPGA. This core is responsible for capturing pilot inputs and monitoring range of inputs to determine if an override condition has been experienced. The FPGA then generates output to the level shifter which drives the physical servo-motors which deflect the control surfaces of the aircraft.

The Inertial measurement unit (IMU) chosen was the ADIS16350/PCBZ-ND. This device enables us to measure linear and angular accelerations in six degree of freedom. The internal 12bit analog to digital converter (ADC) was utilized to digitize three analog sensors. The sensors required for the control system were a differential pressure sensor, static pressure sensor, and a temperature sensor. The sensors operated in a 5V peak voltage which was filtered and stepped down to accommodate the dynamic range of the ADC within the IMU. The ECB also included ten GPIO for testing and real time monitoring. The system utilized an external jumper cable to handle high power servo actuation, thus reducing the need for high power traces on the PCB traces.

Lastly the system required a 2"x 5" footprint and after several iterations of the design this goal was accomplished utilizing only a four layer board. A dedicated ground plane was used to ensure zero potential point is constant across all components. The noise analysis of the system revealed that due to the bulk capacitance and low potential levels of the system noise on the PWM, Analog sensors, and IMU was well below acceptable margins. In future designs the Layout could be reduced even further through repositioning of the major components. This would allow the size of the host aircraft to be reduced, which is the future goal of the project.

CODE IMPLIMENTATION

The senior design project needed a programming platform to facilitate the integration of the sensor and the control system together. The two main options for the programming platform were a microcontroller or a FPGA. The FPGA was chosen because it provides flexibility to implement any hardware components that could be needed to communicate to the digital sensors. Using an FPGA, an open source embedded microcontroller core would provide the computing power need implement the control system. The microcontroller core used was an open source MIPS processor called Plasma and it was implemented in VHDL. The microcontroller core was found on http://www.opencores.org. The microcontroller core was optimized and adapted to fit the project's need. A problem with the implementing the control system was trying to fully understand all the calculations that system would have to do. A concern the team had was that one core might not be able to gather sensor data and calculate servos outputs fast enough. A second core was added to alleviate the concern.

A modular approach was taken in designing to the project's programming platform. The sensor core was responsible for communication with the sensors and transmitting the sensor data to the control system core via shared registers. The control system core would take the sensor data and calculate the new servo positions. Each core has a small amount of shared memory that can be read by the other core for the purpose of exchanging information back and forth. The algorithm created to synchronize the shared memory goes as follows. Both cores share a global timer so they both start their operating cycles at the same time. The operating cycle is the cycle when either core will run through its main loop thus either calculating for the control system or communication to sensors. Both cores will wait until the global timer resets before starting execution of their code thus providing synchronization. Both cores will read their shared register at the beginning of the operating cycle and then each will write back to the other's core at the end of the operating cycle.

Both microcontroller cores were customized with peripheral modules to optimize certain actions according to their responsibilities. The sensor core was customized with four communication modules. Each module takes of sending data and command then waits to receive and parse the responses. The four communication modules are IMU, GPS, SD card, and UART. The control system core was customized with its own double precision floating point module and PWM module.

The SD card provides two different functions in the overall project. The first function of the SD card is to record in-flight sensor readings, pilot input, and control system calculations. A standard SD card reader can later be used on a computer to read the log files. This allows the flight to be replayed on the ground for the purpose of analyzing the plane and control systems performance. The second function the SD card provides is the ability to boot the programs running on the two microcontroller cores. Separate programs are written in C for each core to implement each microcontroller's function. The code is then compiled using the open source compiler GCC to create a MIPS binary file. The microcontroller runs a boot loader program upon power up to load each program into the corresponding microcontroller's memory. This added feature allows for faster testing cycles of different control systems and communication algorithms.

Another hardware component is the status LEDs on the PCB board. These status LEDs allow for feedback to the operator of the status of the system. One LED states when the boot loader has completed its initialized. The second LED provides feedback when the GPS has gotten a satellite signal. The third LED lights up to indicate that the IMU is functioning properly. If any errors occur during initialization, the LEDs blink an error code to provide information potentially helpful in debugging the problem.

TEST STAND

The designed control system has a suite of sensors used as inputs to guide the plane through flight by altering the deflection of the control surfaces. These sensors require calibration, and checks for necessary accuracy. While input sensors like temperature, pressure, and GPS can easily be checked and correlated to laboratory grade ground truth sensors provided by the RIT, the inertial measurement unit (IMU) cannot be easily checked or quantified by any standard instrumentation. The IMU provides the ability to measure attitude change of the MAV, which are critical inputs for autonomous flight control systems. To fulfill this need of testing the Rotational Test Stand for rotation IMU accuracy has been designed.

The purpose of this test stand is to check the accuracy and functionality of the rotational accelerometers in all 3 rotational degrees of freedom as well as to verify attitude calculations implemented in code on the FPGA. While this could potentially call for a design that has 3 respective degrees of motion, we are able to simplify the design to only rotating in 2 axes. This simplification is valid because the test stand can be mounted on its side and thus capturing motion of the neglected dimension. The limitation of this system is that the test stand will only be able to test two degrees of freedom at the same time. For the purposes of measuring sensor accuracy and calculations this is a reasonable limitation. Furthermore, the separate lateral and longitudinal control systems never have more than 2 rotational inputs at one time. So the test stand may test the control systems response to lateral stability in one test run and the longitudinal in a second test. The final test stand design is shown below in Figure 2.

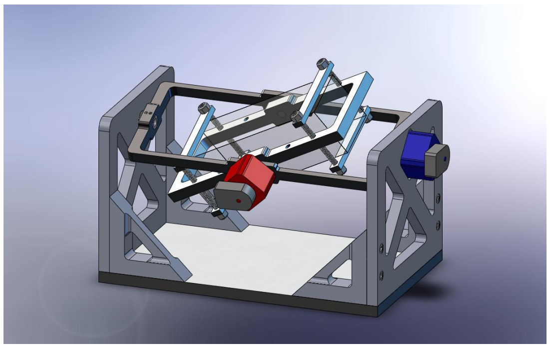

Figure 2: Test Stand Design

Nema_17HX18D stepper motors were chosen as the rotational motors over servo motor because of its precision and light weight. This stepper motor includes a controller that can be programmed by the user to operate "stand alone" or that can receive serial commands from a host PC.

Stepper motors work very differently from the normal dc motor. DC motors work when voltage is applied in its terminal. However, stepper motors have multiple electromagnetic teeth arranged around the center gear called the rotor. These electromagnets are energized by an external control circuit, such as BiStep2A04 controller which was used. In addition to that, hybrid stepper motors have permanent magnet teeth magnetized along the rotor shaft axis. In order to make the rotor shaft turn in a certain direction, power needs to be applied to one of the electromagnetic teeth which makes the gears magnetically attracted to the permanent magnet on the stator. Since the number of teeth on the stator is higher than the teeth in the rotor, there is a slight offset to the next electromagnetic tooth. When the next pulse is applied, the rotor rotates slightly to align with the next tooth. Each of these slight rotations is called a step which is the smallest angle the motor can rotate. The motor chosen has a step angle of 1.8 degrees which is its highest accuracy which for our testing application is more then acceptable.

The BiStep2A04 control board is used to control the stepper motor. The board has two identical connecters used to control two motors X and Y and they can be controlled independently. The controller has three sets of power and common ground. This mostly is done to deliver enough power from the board in case it is needed by the motor and each connector can handle up to 6 amps. Having three motors connected to the source make it possible to supply up to 18 amps. The two windings of the steeper motor are connected to the board and pulse wave forms are sent in the correct pattern.

The driving commands are sent from a computer through a serial connecter to the board. A schematic of the test stand communication is shown below in Figure 3. The controller board gives us three degrees of freedom. We can control the position of the stepper very precisely because of the small step angle. The speed can be controlled over a large range. Also the acceleration over in which the motor achieves a desired speed can be set at very precise ranges.

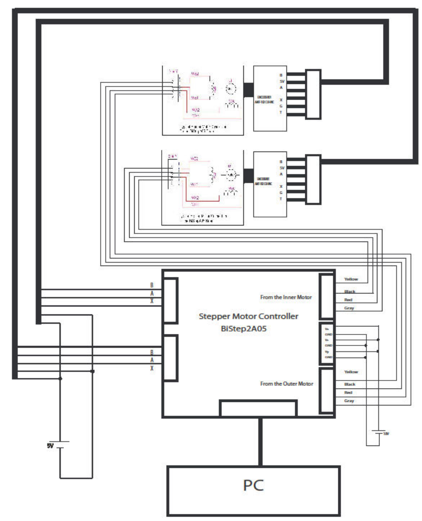

Figure 3: Test Stand Communication

A rotary encoder is needed in our application in order to have feedback of our motor. The AMT 103 encoder was chosen for this application. It is an electromechanical device that converts the angular shaft position into digital data or pulse width modulations PWM. The optical encoder's disc is made of glass with transparent and opaque areas. A light source and photo detector array reads the optical pattern that result from the disc's position at any time. The AMT 102 has two channels A and B. these two channels are known as Quadrature output. The two signals provided by these channels are 90 degrees out phase. By comparing the two signals, the direction of rotation of the motor can be determined. The output data of the encoder are pulses, the frequency and the duty cycle of these pulses indicates the speed and the position of the motor comparing to the initial one. In our application, these data are interpreted by a programmed microprocessor and saved in one of its register. At the end of each application, those data can be converted to degree and seconds.

RESULTS AND DISCUSSION

The test stand was manufactured through the Brinkman Manufacturing Lab at RIT however through uncontrollable circumstances the test stand parts were not completed within the 4 week lead time that was originally specified when the designs were submitted at the beginning of the quarter in MSD II. The test stand was extremely behind schedule and was not complete until after week 10. This delay greatly hindered the team progress and no accurate ground testing was able to be performed.

With no ground testing possible the IMU sensor and attitude calculations were not able to be calibrated and verified. Without accurate attitude measurement the full flight control system was not implemented in the hardware to avoid the risk of malfunction during flight testing. A simplified proportional controller was able to be implemented using basic gravity vector calculations from the IMU and was tested by manually rotating the aircraft.

The primitive autopilot systems were successfully designed and simulated and final implementation would only require the necessary sensor calculations and appropriate choices for the gains which could be determined through flight testing.

The entire electronics board and sensors were fully functional and many ground tests were performed to collect sensor data. The SD card reader and boot loaded was also fully functional as well as the status LED's. The raw sensor data and simple sensor calculations were also successfully stored onto the SD card as well as transmitted wirelessly using the short-range MSP430 wireless communication.

CONCLUSIONS AND RECOMMENDATIONS

Although the full flight control system was not able to be implemented and tested this project has made tremendous progress toward developing and autonomous flight control system. The success of the electronics board, sensor data collection and simple control law implementation is a tremendous gain for the resources and technology for autonomous flight research at RIT.

For future teams we recommend that another EE student works on the PCB which could be further refine and reduced in size and weight. I heavy focus should also be put on implementing and testing of sensor calculations. With the test stand being made at the end of the project future teams can utilize it to test right way.

In control system simulation the plant model is very important and getting the aerodynamics properties is a very larger role. We recommend looking into using Digital DATCOM to do this. Time and resources were not available in the scope of this project so it was not researched enough but would be a very good tool for future teams.

We also recommend researching robust and non-linear control methods to reduce the impact of model uncertainties and allowing the control system to achieve the desired outcome regardless of model uncertainties.

The video camera that was used was blurry at times and has issues with noise. Future teams should invest in a higher quality camera and transmitter to receive higher quality video. The GPS that was purchased only operated at 1 Hz and there are better GPS units available on the market that would allow for more accurate data with a much faster refresh rate.

Additional telemetry would also be very helpful to receive in flight data over longer distances and could also be used to upload control changes or flight requirements.

REFERENCES

(1) Nelson, Robert C. Flight Stability and Automatic Control. 2nd ed. McGraw-Hill. 1998

ACKNOWLEDGMENTS

We would like to acknowledge Impact Technologies LLC, for continuing to fund and support the MAV projects and encouraging the teams to pursue cutting edge technologies. We would also like to thank our faculty guide Dr. Jeffrey Kozak, and all the faculty who have helped us throughout the year, in particular; Dr. Agamemnon Crassidis, Dr. Marcin Lukowiak and Dr. Robert Bowman. We would also like to thank the RIT Aero Design Club for allowing us use of their facilities, resources and knowledge. We would also like to thank Joe Pinzone for all his help in designing, manufacturing and debugging the electronics board