Quadrocopter Flight Control

Functional Requirements List and Performance Specifications

By: Eric Backman

Advisor: Dr. Malinowski

Date: 11/5/12

Introduction

Quadrocopters are ideal for small scale flying robots due to their ability to hover like a helicopter without the need to change the rotor blade pitch angle. This simplifies their design and control. Last year, Brad Bergerhouse, Nelson Gaske, and Austin Wenzel worked on an autonomous quadrocopter. They constructed the quadrocopter platform, installed a real time operating system on BeagleBoard, and started sensor implementation.

The primary goal for this year's project is to get the quadrocopter flying and responding to inputs from sensors using microcontroller Linux. The platform is already built but it needs controllers and sensors to function. A joystick will be connected to another Beagleboard or PC that will connect wirelessly to the onboard controller. This will allow a user to control the quadrocopter. Simple object avoidance will be programmed to prevent the quadrocopter from crashing into walls, as well as simple flight patterns to allow for semi-autonomous flight and navigation. Lastly, a camera will be integrated into the system to allow for better navigation.

System Block Diagram

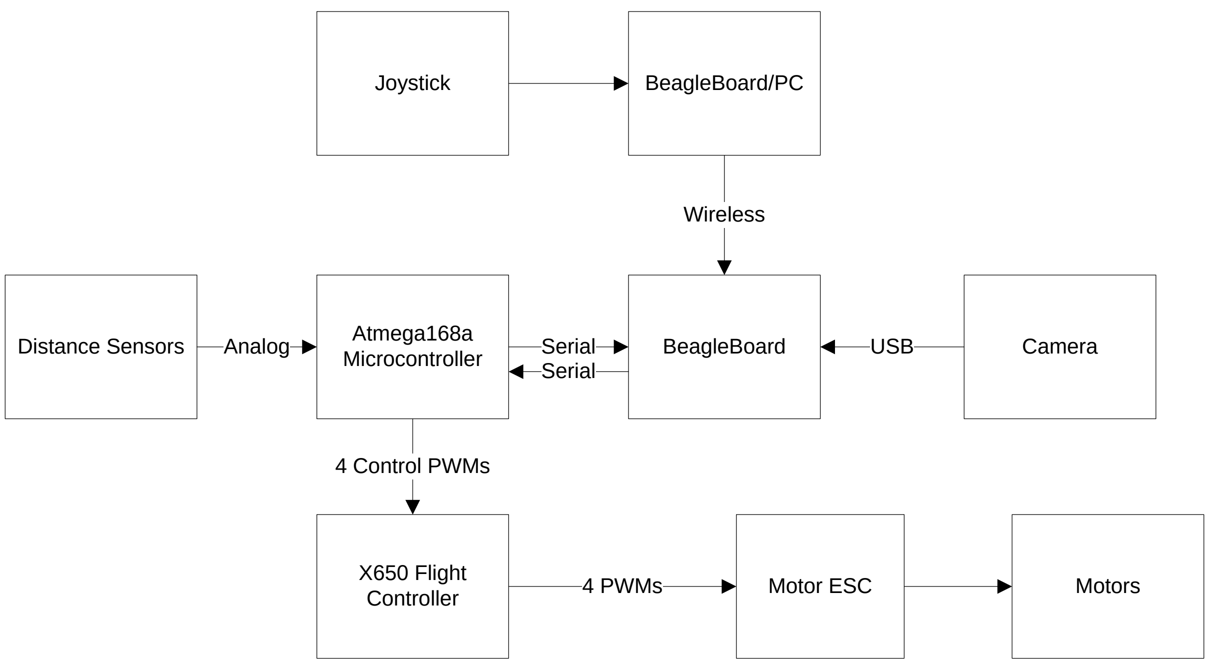

Figure 1. Quadrocopter System Block Diagram

This system will work by taking the inputs from the sensors and sending them to the analog-to-digital converters on the Atmega168a chip. These digital signals will then be sent to the BeagleBoard through a serial port. The BeagleBoard will take these inputs and use them to influence the command coming either from a controller or based on pre-programmed autonomous decisions. It will generate a signal for yaw, pitch, roll and elevation. Since it is tough to generate PWMs from Linux, it will send these signals back to the Atmega168a which will generate the required PWMs and give them to the built-in flight controller. The flight controller will take the commands and use those to run the motors.

Overall System Requirements

- Quadrocopter can take off and land autonomously
- Quadrocopter will be able to fly for at least 10 minutes per charge
- Able to autonomously fly through corridors without crashing into walls
- Able to land on color coded pad using camera

Infrared Sensor Requirements

- Sense distance to all objects within at least 1 meter with accuracy of 1 cm
- Distance sent to microprocessor from 6 sensors at least every millisecond

Atmega168a Microprocessor Requirements

- Read the distance sensor voltage with accuracy equivalent to at least 1 cm accuracy of measured distance
- Send distance sensor data to BeagleBoard at least every 10 ms using serial port
- Create Pulse Width Modulation signal of 50 Hz that has at least 10 us resolution

BeagleBoard Requirements

- Analyze distance sensor data to update command PWM settings every 20 ms
- Filter camera images to determine a certain color coded instruction
- Analyze images acquired from the camera to supply a command that updates PWM settings every 20 ms

References

[1] Brad Bergerhouse, Nelson Gaske, Austin Wenzel. Aerial Collision Avoidance System. Senior Project, Electrical and Computer Engineering Department, Bradley University, May 2012, http://cegt201.bradley.edu/projects/proj2012/quadcptr/

[2] Introduction to Autonomous Mobile Robots, 2/ed., by R. Siegwart, I. R. Nourbakhsh, D. Scaramuzza, MIT Press, 2011, ISBN: 978-0262015356