Gregory Kravit
Draft Proposal for 6.111 Final Project

# FPGA Implementation of a Digital Controller for a Small VTOL UAV

## OVERVIEW

For the final project this term, I intend to design and implement a digital controller on a FPGA for a small quadcopter vertical takeoff and landing (VTOL) unmanned aerial vehicle (UAV). The digital controller will be designed using a Mojo V3 FPGA prototype board that will control the four motors on the aircraft and integrate sensor inputs for automatic control. The aircraft will most likely be tethered to a ground station for simplicity and safety reasons in the lab environment. The initial goal of the project will be to initially have the quadcopter be able to be in a stable hover at a selected altitude. Sensor inputs will be gathered from an inertial measuring unit (IMU) for angular rates and accelerations, and from an ultrasonic range finder for height readings.

For the project to be successful, both systems integration and digital design will be important. While much of this project has been done previously by others in one form or another, getting everything to work together safely and effortlessly will require the largest amount of effort. A number of systems have large support bases but often only for microcontrollers. Getting systems to work properly with the FPGA by designing different modules and interfaces to read in sensor data and operate the motors will also require the bulk of the effort.

To make things safer and easier in the lab environment, the quadcopter will be tethered to a power source to avoid the use of Lithium Polymer (LiPo) batteries. LiPo batteries can pose a fire hazard and can be a pain to recharge safely due to the specialize equipment needed. The tether will also keep the quadcopter from getting away and leaving the test area.

Motivations for this project are several fold. Primarily, I felt this final project would be a great opportunity to use the controls knowledge I've learned in my other classes in the Aeronautics and Astronautics department and then apply that to implementing a complete digital aircraft control system. Another consideration is the novelty of designing a FPGA based flight controller. The commercial market for drones (mainly multicopters) has exploded out of tiny radio-controlled (RC) aircraft community in the last few years so there is a large hobbyist and enthusiast community out there. There have been numerous commercial and hobbyist autopilots and flight controllers released into the market, but they all rely on microcontroller processors. Given that microcontrollers have gotten fast enough and cheap enough to provide enough performance to fly tiny aircraft, they don't really have the ability to do anything else. The FPGA provides the opportunity to do the controls and even more, such as image processing, in parallel on a tiny compact platform.

Gregory Kravit
Draft Proposal for 6.111 Final Project

## DESIGN DECISIONS

The Mojo V3 FPGA prototype board ($75) from Embedded Micro has been chosen for this project due to its form factor, its price and that it can easily be connected to a servo shield ($40) operating the motors. The Mojo contains a Spartan 6 XC6SLX9 FPGA and also uses an ATmega23U4 for the boot-loader and for the analog-digital converters (ADC). The Mojo has onboard a 50 MHz (megahertz) crystal which is more than ample for this project. An additional consideration for choosing this platform is that the company offers numerous tutorials for how to program different functionality for the board such as pulse-width-modulation (PWM) control signals. The company also published a demo project of a blob-tracking hexabot that from a systems perspective contains several similarities to this quadcopter project which to me offers a way forward to dealing with several of the issues that I am likely to encounter.

The FPGA will control all systems on the quadcopter through several different modules including a **flight control** (**FC**) module, a **motor control** (MC) module, and **sensor** control modules. The flight control module will contain all the arithmetic functions needed to provide proper inputs for the motor controls. The flight control module will implement a digital PID controller to provide stability and control. A digital PID is common for this application when using the simplified quadrotor dynamics that I will assume to be accurate when I design the controller. The flight control module will take in sensor readings from the sensor modules and make corrections accordingly. The flight control will be designed to work on a 400 Hz (hertz) refresh rate, which maximum refresh rate of the electronic speed controllers (ESCs). The flight control module will output a throttle setting from 0-127 for each motor that the motor control module will then convert into a signal for the ESCs.

The **motor control** module will provide the pulse width modulation signals for the electronic speed controllers (ESCs). The MC will take a throttle setting input from the FC module for each motor and convert that to a PWM signal to the ESCs. The correct PWM signal will be given in a range from 1060 microseconds [us] (idle throttle) to 1860 us (full throttle). The MC will output PWM signals to the pins on the servo shield at 400 Hz.

The sensor modules will be used to read two sensor breakout boards: an ultrasonic range finder and an IMU board. The ultrasonic range finder (MaxSonar EZ2 $27.95) returns a signal either through PWM or analog. PWM will be used because digital inputs will be easy to integrate into the Mojo. A PWM receiver state machine will need to be implemented to read this sensor properly. The ultrasonic range finder will return a signal from 0-225 inches which in this case will be used to describe the height above the ground.

The IMU board ($10) is a breakout board containing the Invensenses MPU6050 6DOF chip. The MPU6050 contains both an accelerometer and gyroscope on chip which it can then calibrate and output desired angular rates and accelerations using on chip sensor fusion algorithms. Communication with the board is performed through I2C. I plan to use the I2C module will need to be implemented. This board was chosen because it can provide accurate state readings in a small form factor and at a low price for the flight control module. Additionally because of the onboard sensor fusion algorithms, a Kalman filtering of the angular rates and accelerations will not be needed.

The quadcopter platform chosen for this project is an iFlight 450B 450mm quadcopter ($150). The quadcopter is a cheaper Chinese clone of the Djii 450 quadcopter. The quadcopter came as a kit with all the necessary parts included: electronic speed controllers (ESCs), specially tuned brushless motors and the frame. This quadcopter was chosen based on price, its relatively small form factor, the DJii's (and its various clones') proven performance in the hobbyist community, and that it can lift up to 1.5 kg (which is more than what will be needed). The kit was also chosen because the ESCs are preloaded with the SimonK ESC firmware. The SimonK firmware is open source and heavily documented which will allow me to understand how to interface with the motors through a custom motor controller implemented in the FPGA.

Gregory Kravit
Draft Proposal for 6.111 Final Project

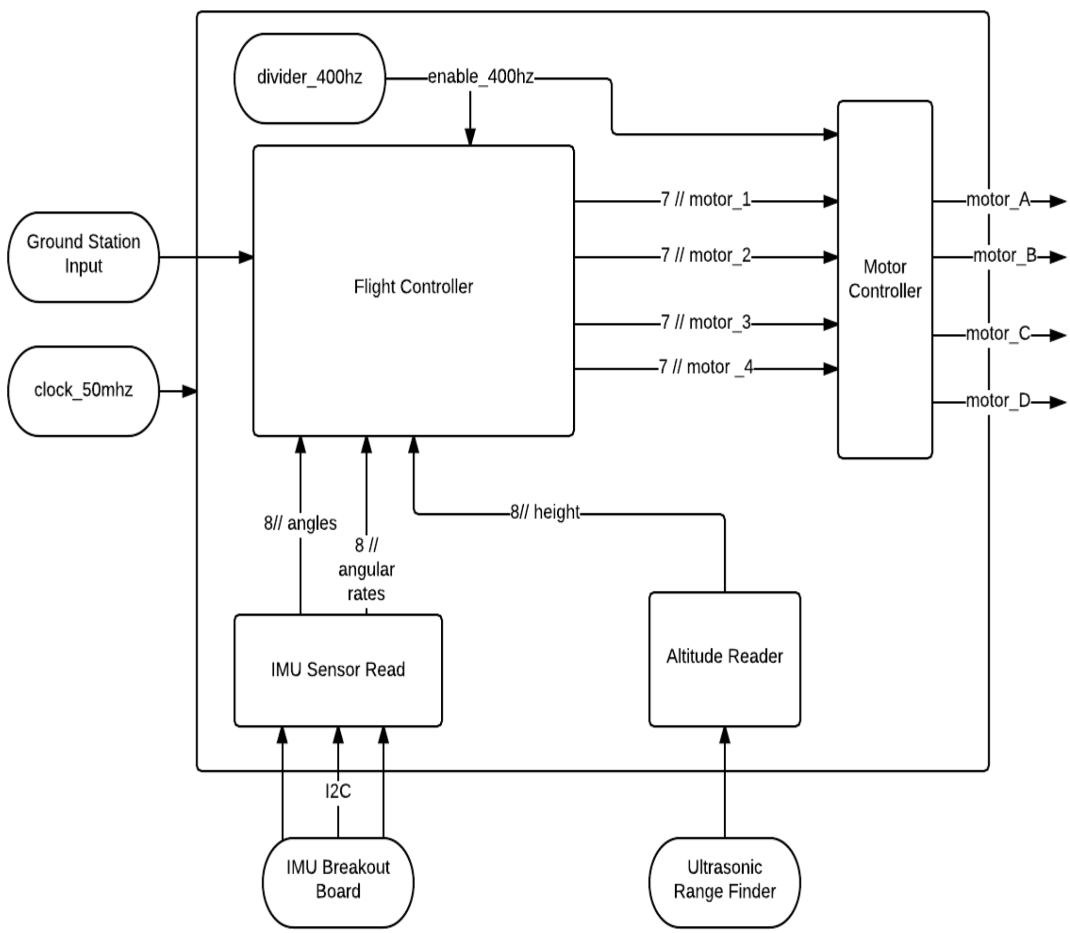
As mentioned in the overview, the quadcopter will be tethered for power. Power will be provided at 12 V and up to 30 amps directly to a power distribution board which is connected to the ESCs. Powering the Mojo FPGA and sensors will either be via the 5v 3A battery eliminator circuit (BEC) on the ESCs or via a 5 volt voltage regulator step down to the DC power input on the Mojo.

Gregory Kravit
Draft Proposal for 6.111 Final Project

# IMPLEMENTATION

Implementation is mostly explained in detailed description. More will be added in revised proposal.

## FPGA Level Diagram:



Gregory Kravit
Draft Proposal for 6.111 Final Project

## TIMELINE

**Week Nov 1-Nov 7 :**

- **Order Parts**
- **Test Motors and ESCs**
- **Model quadrotor in MATLAB and determine PID controller equations**
- **Design and implement**
  - **Motor controller**
  - **Ground Control Station Input**

**Week Nov 8-Nov 14 :**

- Design and implement
  - Sensor Modules
  - Flight Controller
- Build quadrotor
- Get systems integrated→ begin testing

**Week Nov 15-Nov 21 :**

- Get systems integrated
  - test all modules
  - test working quadrotor
- Tune PID controller through testing

**Weeks Nov 24-Dec 5 :**

- Finish the system, get everything working
- Write the final report

## TESTING

Expected Testing:

- Modelsim Workbench Simulations (Individual Modules)
- MATLAB Simulink Hardware In The Loop Simulation
- System Integration Testing
- PID Controller Tuning

## RESOURCES

Lab Materials Needed:

- DC Power Source supply greater than 20 amps current
- Soldering supplies: shrink wrap, soldering iron, solder with flux
- Labkit or breadboard with buttons for ground control inputs
- Mini breadboards for mounting and connecting on board sensors.