

Digital Controller for a Small VTOL UAV



Gregory Kravit
6.111 Fall 2014

iFlight (450mm) ARF

- 4x1000Kv DC motors
- 30A ESCs
- Mass <1.5 kg

The growth and visibility of small drones has been due to the accessibility of tiny, fast microcontrollers
that can perform all the tasks within the refresh rate of the motors



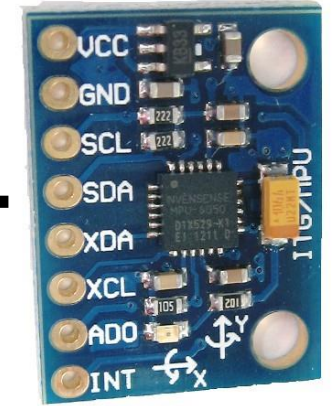
SPEED = SAFE FLIGHT

Exploiting Parallelism in Quadcopter Control

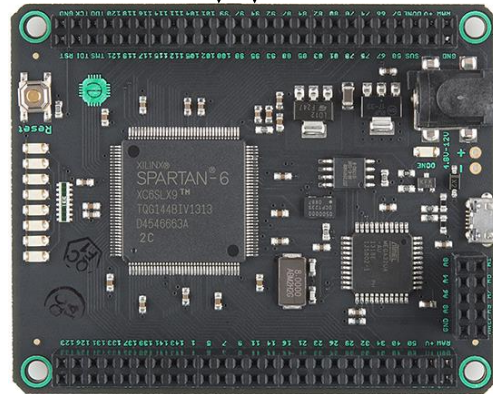
4 Motors + ESCs*



Sensors



FPGA

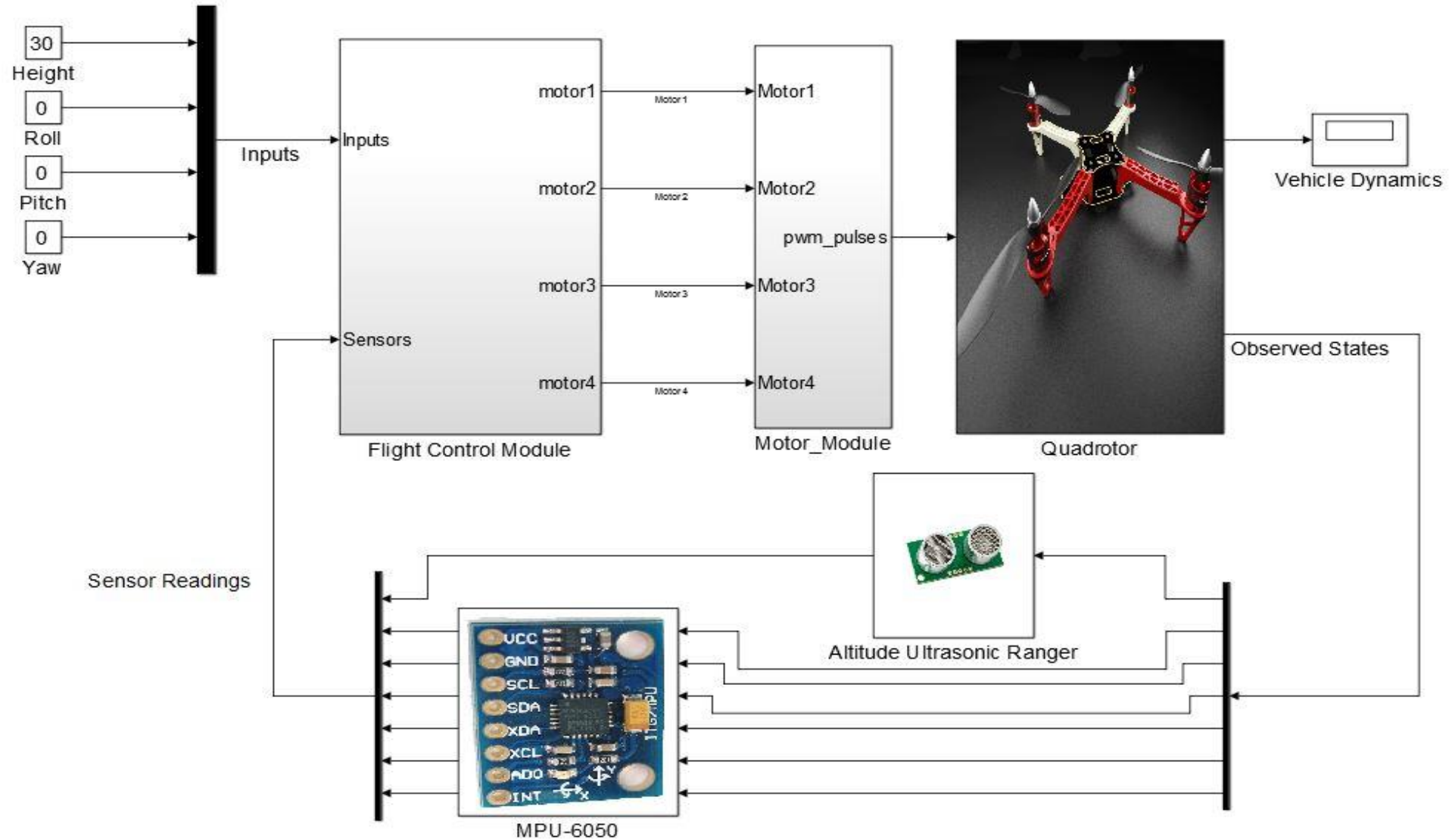


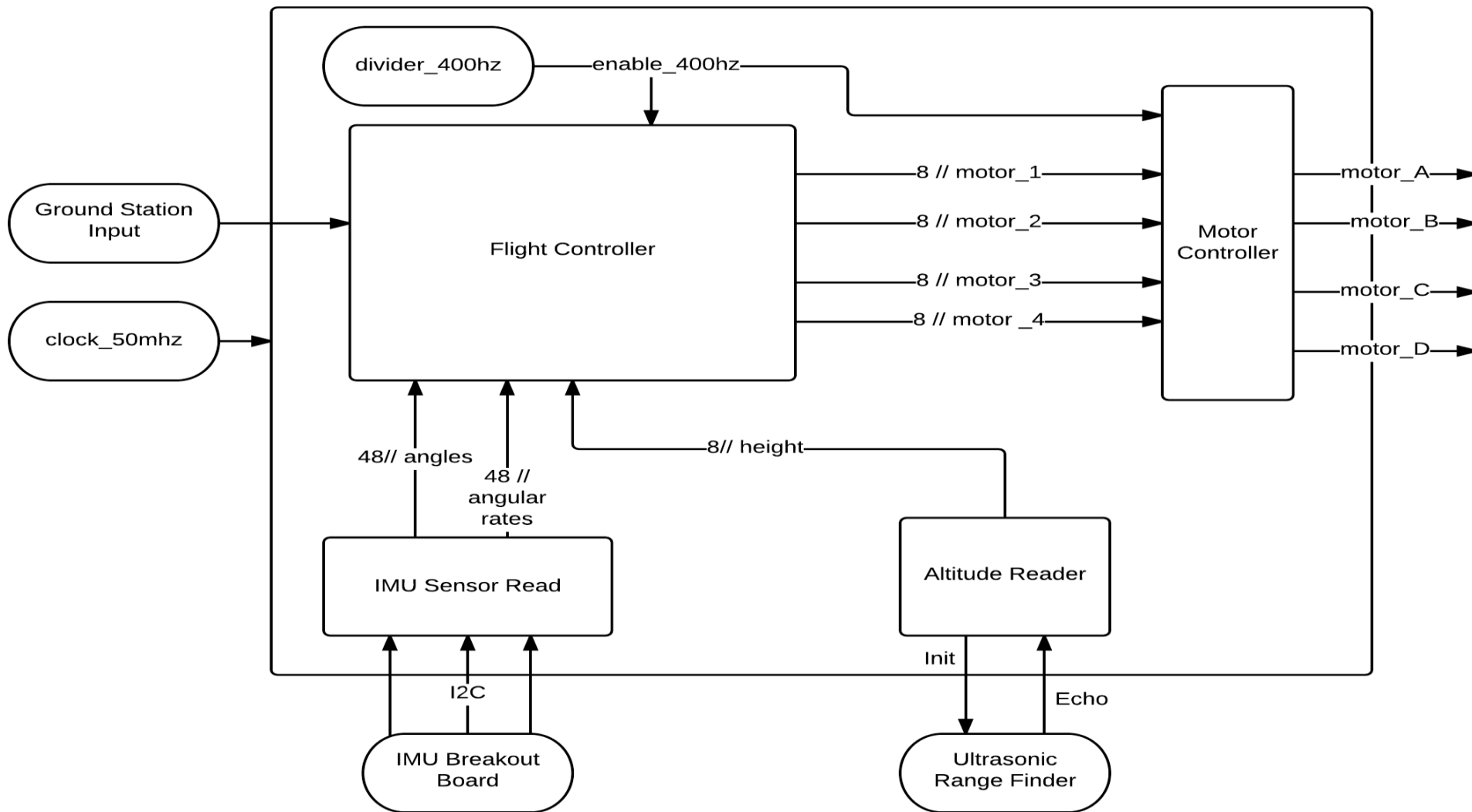
*ESC = Electronic Speed Controller

Project Goals

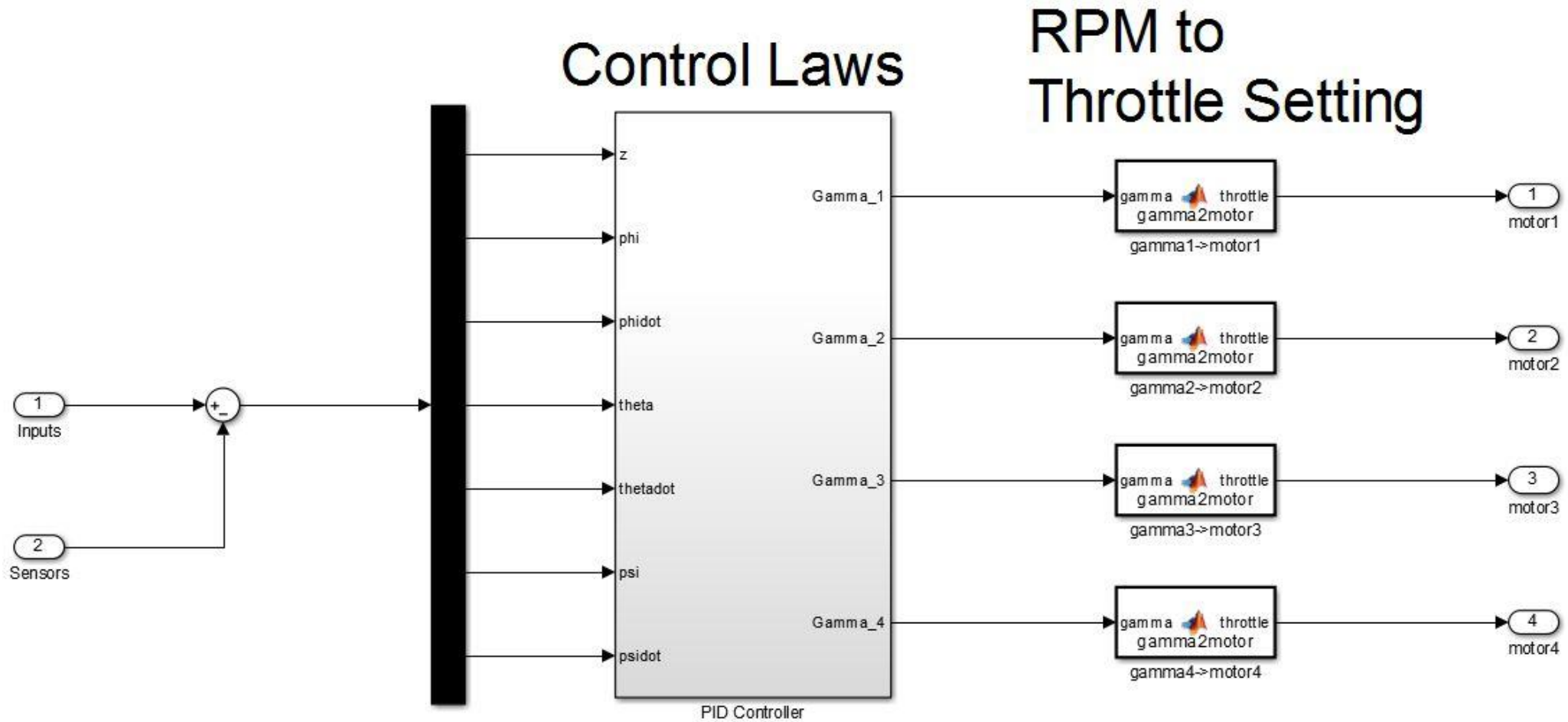
1. Design and implement a quadcopter control system that
 - **Stabilizes the UAV**
 - **Holds a specific altitude**
2. Integrate sensors to read current aircraft states
 - **IMU (Inertial Measurement Unit)-> accelerometer, gyroscope, sensor fusion**
 - **Ultrasonic Sensor**
3. Demonstrate safe, stable flight in the lab environment

Control System Block Diagram





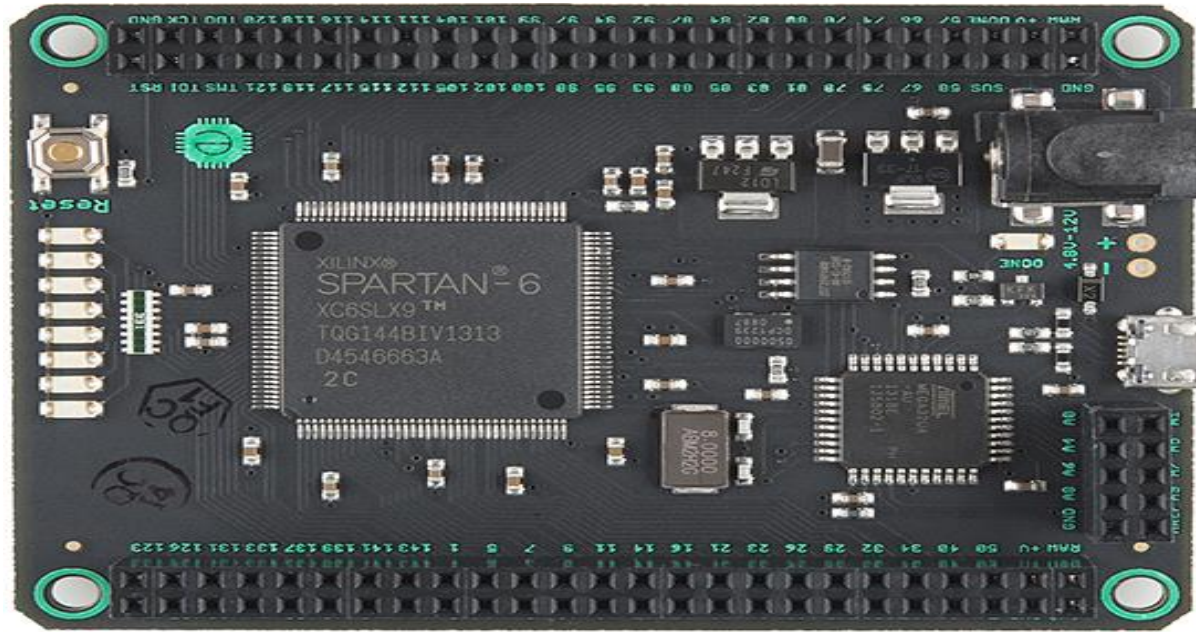
FPGA Module Diagram



$$\text{PID Controller: } e_{\phi} = K_D \dot{\phi} + K_I \int \phi + K_P \phi$$

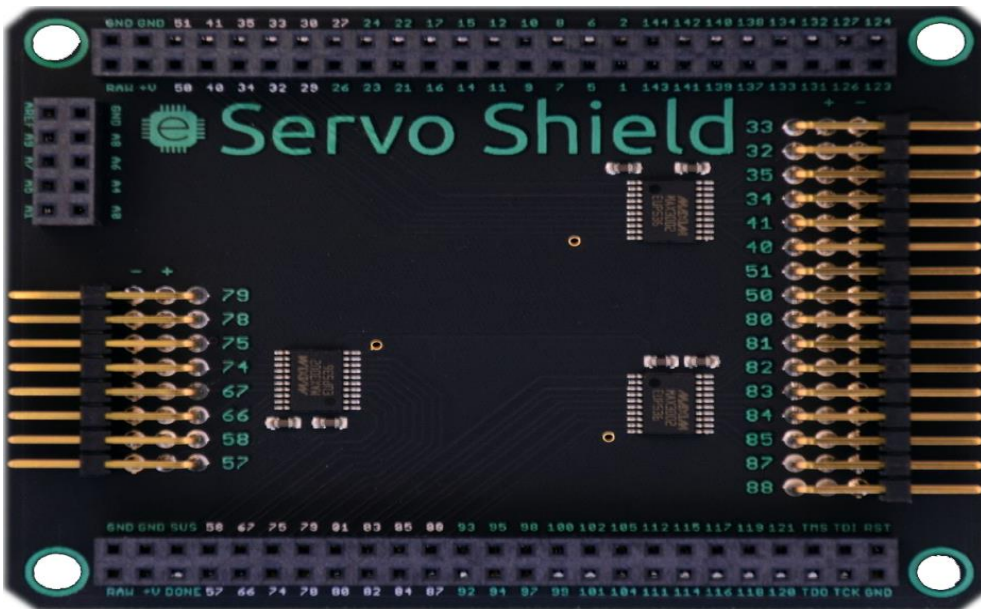
FPGA Module Summary

- Flight Control Module
 - PID arithmetic
 - Map RPM to throttle setting
 - Control Inputs
- Motor Controller
 - Throttle Setting to PWM pulse length
 - PWM pulse out
- IMU Sensor Reader
 - I2C Master
 - Convert quaternions to angles
 - Initialize IMU sensor
- Altitude Range Sensor
 - Send start signal
 - Counter for signal time
 - Receive signal echo



Mojo v3 FPGA *from Embedded Micro*

- 50 Mhz clock
- Spartan xc6-xl9
 - ~9000 logic slices
 - ~1100 LUTs
- Add-on servo shield
- at128Mega microcontroller onboard
- Tutorials available concerning robotics projects on the website



Systems Integration Tasks

- Build UAV from Almost-Ready-to-Fly (ARF) kit
- Tether the power source using speaker cable and PC ATX power supply
- Integrate sensors onto the UAV
- Tune PID parameters for stabilized flight



Expected Issues

- Limited number of LUTs on the FPGA
 - **SOLUTION: Exploit low refresh rate (~400hz) and use limited logic slice arithmetic algorithms as well as onboard DSPs**
- Tuning PID can be difficult
 - **SOLUTION: Model quadrotor and controllers in MATLAB simulation**
- IMU (MPU-6050) is difficult to interface without closed-source developer software
 - **SOLUTION: Translate open source Arduino code to work as FPGA state machines**

Timeline

Week Nov 1-Nov 7 :	<ul style="list-style-type: none">• Order Parts• Test Motors and ESCs• Model quadrotor in MATLAB and determine PID controller equations
Week Nov 8-Nov 14 :	<ul style="list-style-type: none">• Design and implement<ul style="list-style-type: none">○ Motor controller○ Ground Control Station Input• Build quadrotor• Get systems integrated → begin testing
Week Nov 15-Nov 21 :	<ul style="list-style-type: none">• Design and implement<ul style="list-style-type: none">○ Sensor Modules○ Flight Controller• Get systems integrated<ul style="list-style-type: none">○ test all modules○ test working quadrotor• Tune PID controller through testing
Weeks Nov 24-Dec 5 :	<ul style="list-style-type: none">• Finish the system, get everything working• Write final report



Test of the ArduCopter v2.4.2 firmware on the F450 Flame Wheel quadcopter with the APM v2
by Jean-Louis Naudin - March 11, 2012 - <http://diydrones.com/profile/JeanLouisNaudin>

Questions and Comments

Appendix

Control Law Implemented in Simulink

