Oregon State University
EECS Senior Design

FALCON

An Open Quadrotor Flight System

Joey Tomlinson
Scott Rosenbalm
Torben Rasmussen
Sarah Cooley

Table of Contents

Project Introduction

Sponsors and Mentors

Primary:

Revolution Robotics – Aaron Moore
Revolution Robotics – Ben Tribelhorn

Peer Mentors:

Ben Goska
Ryan Albright
Kevin Kemper

The OSU Robotics Club – Aerial Team
Oregon State University – School of EECS

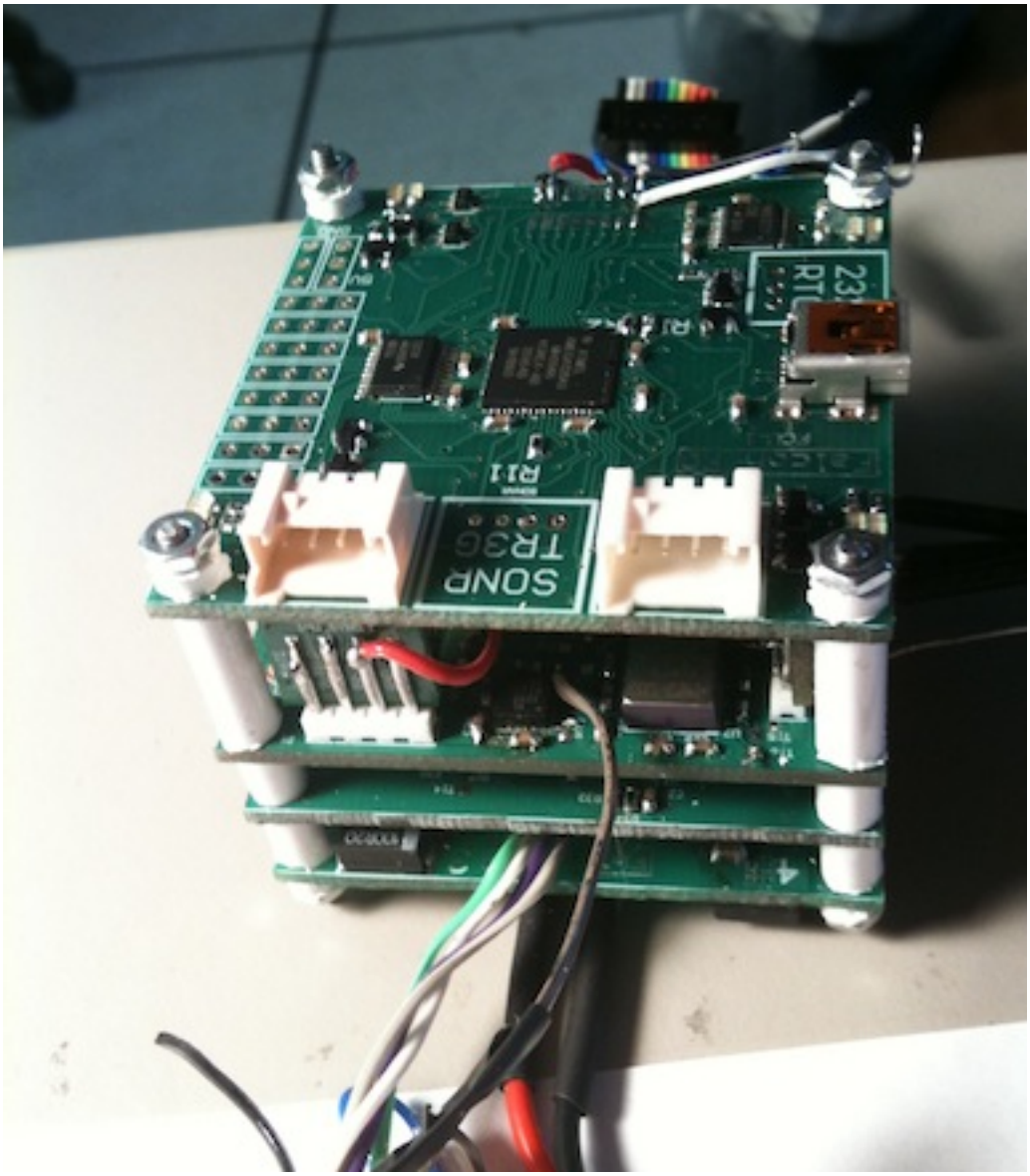


Figure 1: Flight Control System

Overview

We created a generic flight control system for quad-rotor aerial vehicles with capabilities specifically engineered to meet the requirements for the OSU Robotics Club's entry in the 2011 International Aerial Robotics Competition (IARC). IARC requires the vehicle navigate indoors, often in close quarters.
To compete, our flight control system will be designed with performance, stability, and versatility in mind. It will employ high bandwidth, high-resolution MEMS sensors and fast control systems in order to maximize stability performance.

Several multi-disciplinary teams are currently working on the IARC quadrotor. We are building the flight control system while other teams build the quadrotor and write the software for autonomous flight (navigation).

However, while our boards are designed with IARC in mind, they will be generic, open source, and available for other quadrotor vehicles.

We will produce three circuit boards and the associated software to form a complete flight control solution. Our three modules will be the:

The Flight Control Unit (FCU) integrates input from the Inertial Measurement Unit and output to the Motor Controller Unit. It also interfaces with a master controller (already constructed by OSU Aerial). The main stability control loop runs on this processor at a rate of 1kHz. This board will also provide battery monitoring and emergency stop functionality.

The Inertial Measurement Unit (IMU) reads data from MEMS inertial sensors to calculate the angular and lateral motion of the platform. A TI Piccolo processor will process sensor data and provides a state estimate to the FCU every millisecond.

The Motor Controller Unit (MCU) controls the speed of the four brushless DC motors that propel and control the quad-rotor. To maximize platform stability, the motor controllers will have a 1kHz update rate and 10 bit resolution. They will drive motors up to 20A at 12V and will be easily modifiable to support higher current and voltage motors.

Requirements

- The flight control system must interface with the OSU IARC entry
    - Functions normally and continuously between 9 and 12.7 Volts
    - Uses SPI (serial) communication between boards to Communicate with Gumstix Computer
    - Code should be documented and maintainable
- Proper safety precautions should be observed in hardware and software.
    - Remotely activated kill switch.
    - Fail states for unexpected conditions. The controller should decelerate under unanticipated conditions.
- The flight control system should weight less than 300 grams to improve overall efficiency and battery life.
- Our solution must be able to keep the vehicle in flight
    - Powers four flight motors
    - Stable control over the 2kg quadrotor platform
- The vehicle must be aware of its state. Location, battery condition, linear and rotational acceleration, and motor speed data should always be available.
- The vehicle should follow the inputted flight path. Our solution must collect a robust set of environmental and position data.
- Remain within our $1000 budget.

Existing Solutions

Before designing a flight control system from scratch, we researched existing systems and discovered most packages are prohibitively expensive or use slow, inaccurate, sensors. Our design encompasses the inertial measurement unit and the motor controller. Most existing solutions sell each component separately and we would still need to create the flight control unit to join each system.

Specific Product Requirements:

- 20A BLDC motor controllers x 4 motors
- 3 Axis Accelerometers, Gyroscopes, and Magneto Sensors
- 1khz System Update Rate

Existing MCUs

	Current	Voltage	Update Rate	Interface	Weight	Battery Monitor	Over Current	Price
Mikrokopter BLDC	20A	9-14V	500Hz	Serial, I2C, PMW	10g	Yes	Yes	$50
Castle Creations Phoenix-25	25A	9-13V	50Hz	RC PPM	17g	Yes	Yes	$80
Volcano-30	30A	5-17V	50Hz	RC PPM	25g	No	No	$13

Table 1: Commercial MCU Specifications

Mikrokopter BLDC Motor Controller:
This motor controller is open source and specifically designed for fast update brushless DC motors on the mikrokopter quadrotor platform.
Pros: Fast Update Rate, Lightweight, Multiple Interfaces
Cons: Expensive, Poor documentation

Castle Creations Phoenix-25:
These are commercially available motor controllers designed for remote control aircraft applications.
Pros: Lightweight
Cons: Expensive, Slow Update Rate, Only One Interface

Volcano-30:
Commercially available motor controller designed for hobby remote control aircraft applications.
Pros: Inexpensive
Cons: Slow Update Rate, Only One Interface

Existing IMUs

	Acc Range	Acc Noise	Gyro Range	Gyro Noise	Update Rate	Interface	Power	Price
xSens MTi	±5.1g	204Hz	±300 d/s	.05Hz	120Hz	RS232 RS485 RS422	350mW	??
MicroStrain 3DM-GX2	±5.0g	??	±300 d/s	.1Hz	100Hz	RS232	90mA @ 5V	$1700
Sparkfun 6DOF Razor	±3g	150Hz	±300 d/s	.035Hz	50Hz	Analog	??	$90
Sparkfun 6DOF v4	±6g	350Hz	±500 d/s	.4Hz	100Hz	RS232 Bluetooth	150mA @ 5V	$450

Table 2: Commercial IMU Specifications

xSens MTi:
Commercially available IMU meant for industrial and robotics applications.
Pros: Low Power, Low Noise
Cons: Slow Update Rate, Limited Interface Options, Expensive

MicroStrain 3DM-GX2:
This is a commercially available IMU meant to control aircraft attitude and heading reference.
Pros: Low Noise, Low Power
Cons: Slow Update Rate, Expensive

Sparkfun 6DOF Razor IMU:
Hobbyist IMU meant for general-purpose motion sensing applications.
Pros: Inexpensive, Light Weight
Cons: Slow Update Rate, Only One Interface, Small Sensor Ranges

Sparkfun 6DOF v4 IMU:
High quality hobbyist IMU meant for general-purpose motion sensing applications.
Pros: Light Weight, Good Sensor Ranges, Multiple Interfaces
Cons: Expensive, Noisy Sensors

Considering our requirements, none of the currently available IMU/MCU combined solutions would be acceptable solutions. In most cases we would need to purchase 4 MCUs, 1 MCU then build the flight controller main board. Even with a larger budget, the pre-built solutions are too expensive.

Solution

We broke out flight control solution into three major blocks. Each block is on a different circuit board and operates independently of the other blocks. Each communicates through serial with well-defined and documented data packets. We chose this board breakdown since flight controllers are generally sold as an IMU and motor controller.

High Level Design

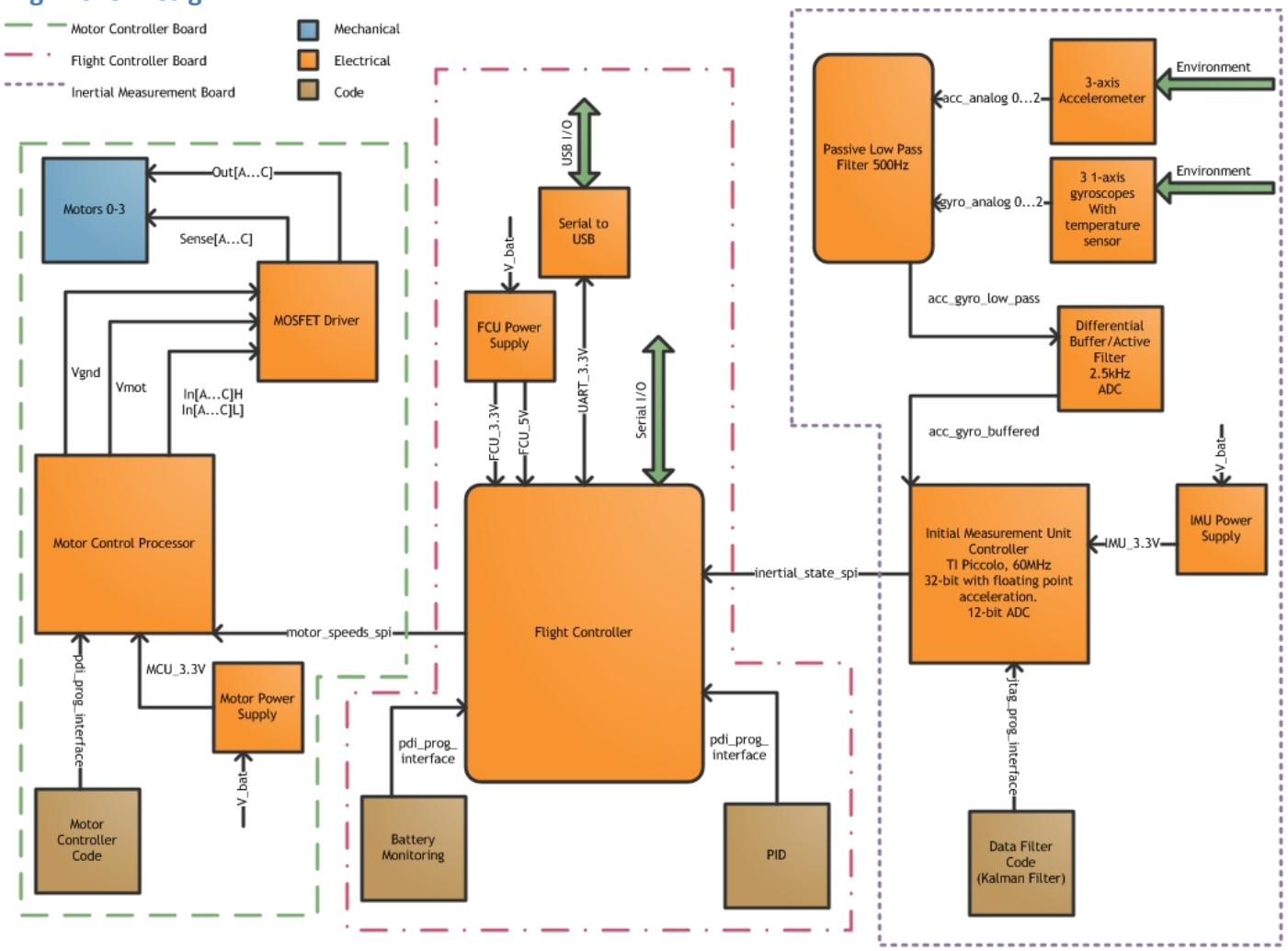


Figure 2: System Block Diagram

Name	Type	Specification
V_3.3V	Electrical	3.3V DC nominal +- .3V
3_phase_ac	Electrical	12V 20A max
pdi_prog_interface	Control	
jtag_prog_interface	Control	
serial	Communication	3.3V 8n1 Serial
gyro_analog	Electrical	300dps 2% Temp drift
acc_analog	Electrical	
Battery	Power	9-12.8V, 100A Li-poly
Power Switch	User Interface	SPDT
Motor Enable Switch	User Interface	SPDT, Must not falsely activate under vibration

Table 3: Top Level Interface Definition

FCU – Flight Control Unit

The FCU, outlined in red in the center of Figure 1, integrates input from the Inertial Measurement Unit and outputs motor speeds to the Motor Controller Unit. It also interfaces with a master controller (already constructed by OSU Aerial).

In software, the main stability control loop runs on this processor at a rate of 1kHz. The board interfaces with a computer using serial over USB and using Xbee wireless serial communication. The FCU provides a console based control mechanism and battery monitoring. From the monitoring computer, a user can control motor speeds and emergency stop functionality.

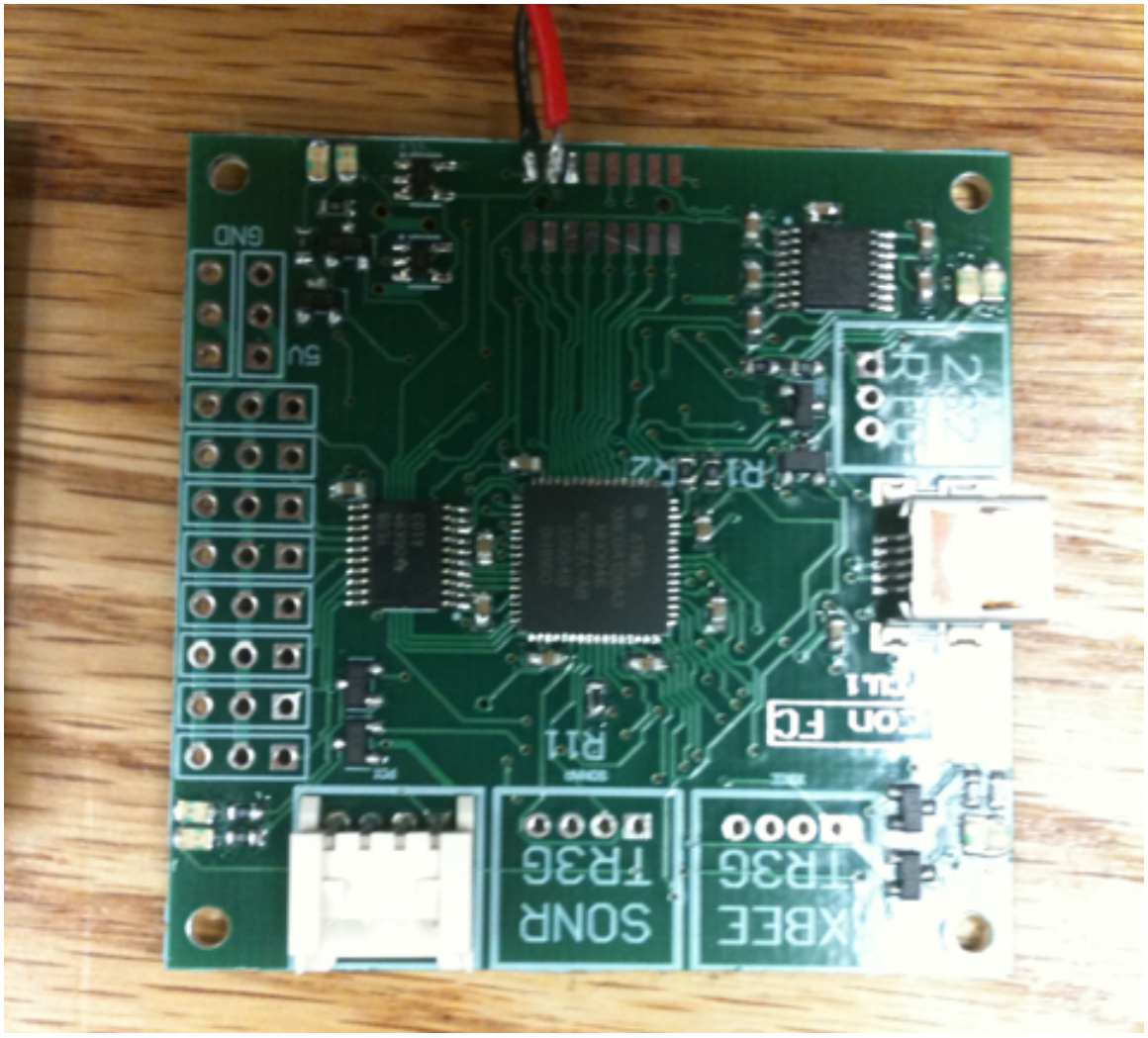


Figure 3: Flight Control Board

Microcontroller

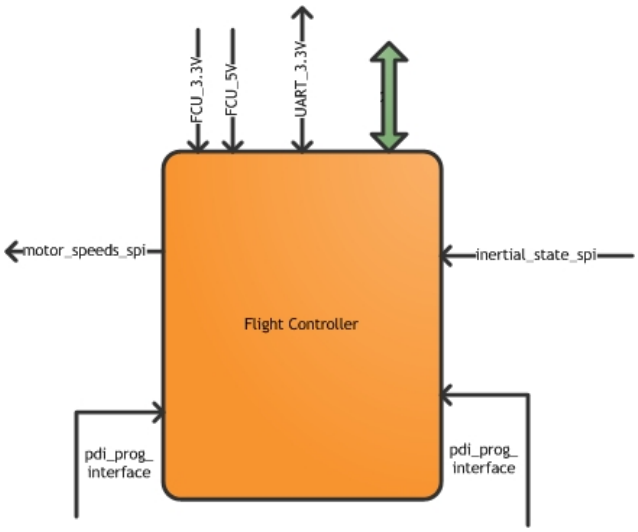


Figure 4: FCU Controller Block

Name	Type	Specification
Vbat	Electrical Input	16.4V (Max)
3.3V	Electrical Output	3.3V, 50A Max
5V	Electrical Output	5V, 50A Max
rc_5_n	Electrical Input	5V PWM
Sonar	Electrical Input/Output	3.3V UART
Xbee	Electrical Input/Output	3.3V UART
rs422	Electrical Input/Output	RS-422 Serial
MISO/MOSI	Electrical Input/Output	3.3V SPI
SCK	Electrical Output	3.3V Clock
MOT_EN	Electrical Output	3.3V Motor Enable signal
SSn	Electrical Output	SPI Chip select signals

Table 4: FCU Controller Interface Definition

The microcontroller is the most important aspect of the FCU since it handles battery monitoring, some basic filtering, and serial behavior controls as well as outputting voltages for the motor controller and interfacing the boards. Specifically, the central flight control unit (fcu) is responsible for communicating between the imu, mcu, and a number of external peripherals. The on-board xmega128a3 reads in filtered digital sensor data from the imu over SPI, and runs a PID control loop to determine the necessary motor speed targets. It then relays this information over SPI to the motor control unit. Also, this processor will be responsible for communicating with external computers (onboard or off) to update PID settings, receive high level commands, or receive control input. This is accomplished with two serial protocols, RS-422, and 3.3V serial to USB.

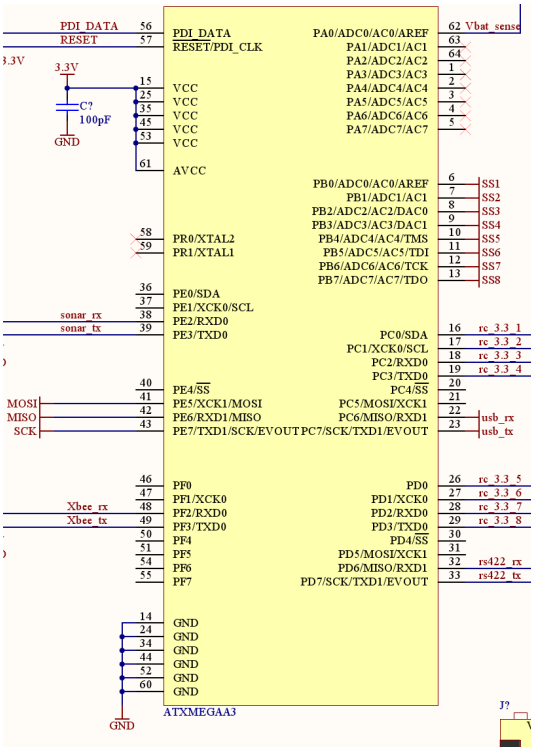


Figure 5: Flight Controller Schematic

The voltage divider used to measure battery voltage must reduce the voltage to below 3.3V for safe operation.
The maximum voltage of a 4-cell Li-Poly battery pack is 4.1V * 4 cells = 16.4V.
V(adc0_max) = 16.4V * (10k / (44.2k + 10k)) = 3.0258V < 3.3V.
Worst case with 1% resistors:
V(adc0_max) = 16.4V * (10.1k / (39.78k + 10.1k)) = 3.3V

All other components are designed to work in low-power situations at either 3.3V or 5V and do not exceed any of the absolute maximum ratings in the Xmega a3 microcontroller datasheet.

Serial to USB

Figure 6: Serial to USB Block

Name	Type	Specification
UART_3.3V	Electrical Input/Output	3.3V Serial
USB I/O	Electrical Input/Output	USB

Table 5: Serial to USB Interface Definition

This device is responsible for converting 3.3V asynchronous serial signals to USB. This will be one of the main ways the central flight controller interfaces with external computers.

The 3.3V serial interface is within tolerances for the chip. The power requirements of the chip are met by the 3.3V power supply on the flight control board.

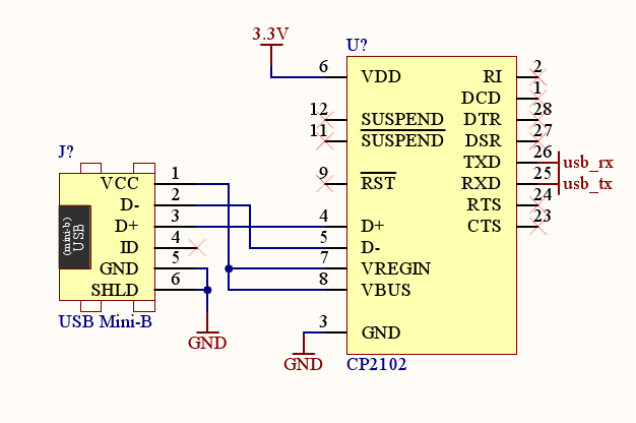


Figure 7: Serial to USB Schematic

Power Supply

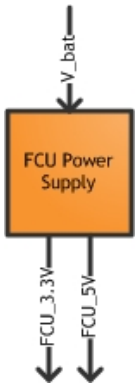


Figure 8: FCU Power Supply Block

Name	Type	Specification
V_bat	Electrical Input	9-12.6V
FCU_3.3V	Electrical Output	3.3V, 50A max
FCU_5V	Electrical Output	5V, 50A max
V_bat	Electrical Input	9-12.6V

Table 6: FCU Power Supply Interface Definition

This block is responsible for providing power to the central flight control unit and its peripherals at both 5 and 3.3V. The linear voltage regulator chips are designed to output 50mA. The total current draw of all parts at each voltage level will not exceed 25mA.

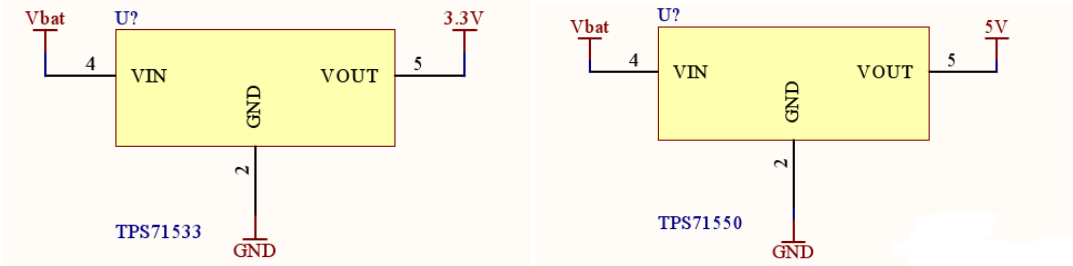


Figure 9: FCU Power Supply Schematic

IMU – Inertial Measurement Unit

Figure 10: IMU Sensor Board

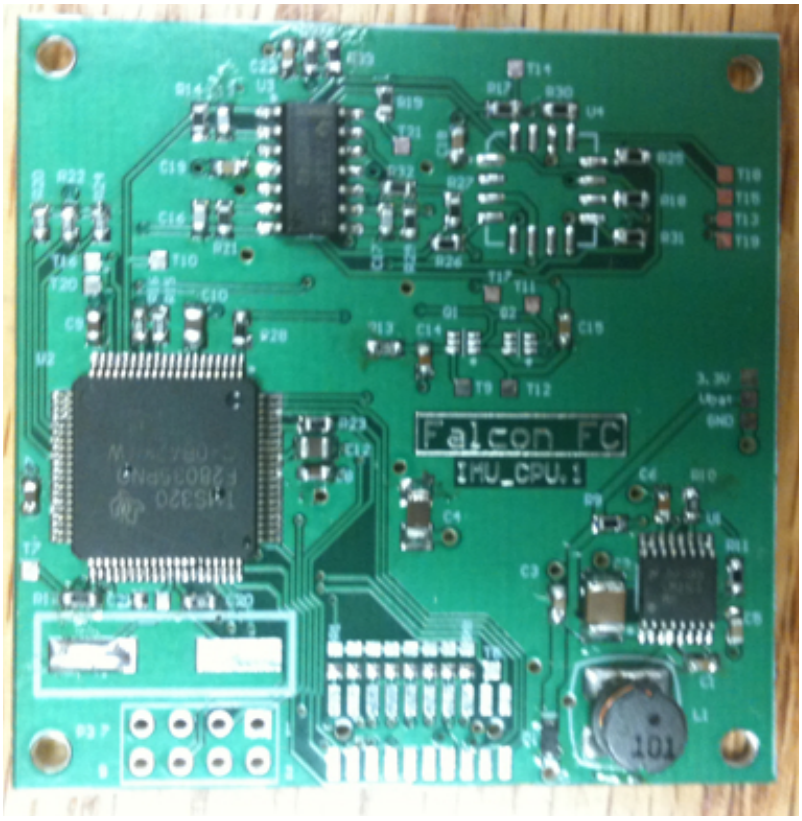


Figure 11: IMU CPU Board

Sensors

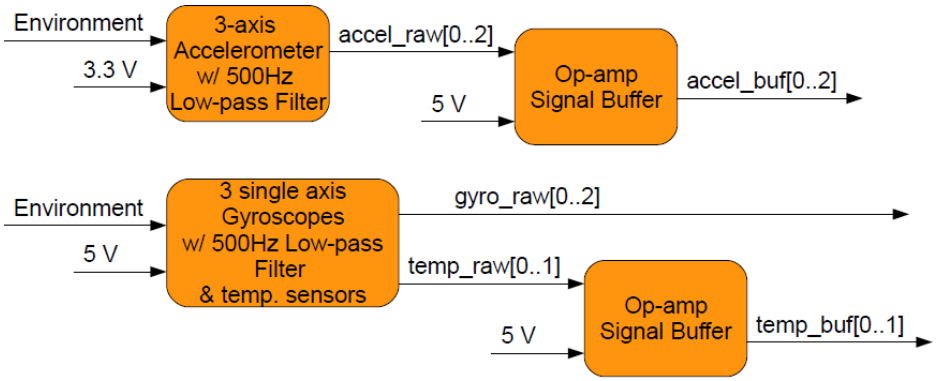


Figure 12: Sensor Block

Name	Type	Specification
5V	Electrical Input	5V power supply.
3.3V	Electrical Input	3.3V power supply.
accel_raw[0...2]	Electrical Signal	Analog outputs from accelerometer. Signal between 0V - 3.3V and has been through passive 500Hz low-pass filter. Un-buffered
accel_buf[0...2]	Electrical Output	Buffered analog outputs from accelerometer. Signal between 0V - 3.3V.
gyro_raw[0...2]	Electrical Output	Analog outputs from gyros. Signal between 0V - 5V and has been through passive 500Hz low-pass filter.
temp_raw[0...1]	Electrical Output	Analog temperature outputs from pitch and yaw gyros, un-buffered. Signal between 0V - 5V.
temp_buf[0...1]	Electrical Output	Buffered analog temperature outputs from pitch and yaw gyros. Signal between 0V - 5V.

Table 7: Sensor Interface Definition

All sensor data is low-pass filtered by applying capacitors to certain pins on the sensor IC. The bandwidth was set to 500 Hz for all sensors to remove aliasing in the signal when sampling at 1kHz, according to Nyquist sampling theory.

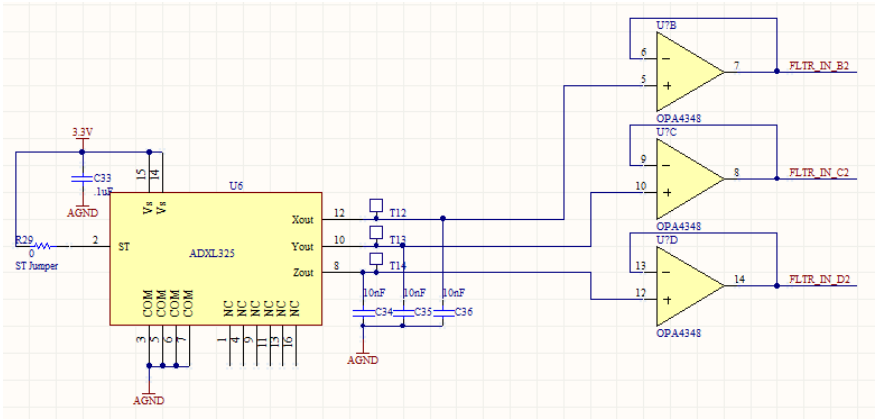


Figure 13: Accelerometer Schematic

For the accelerometers, bandwidth is set by the equation $f = 1/(2\pi(32\ k\Omega) \times C)$, resulting in a capacitance of 10nF for each axis output.
Range: +/-5g max acceleration

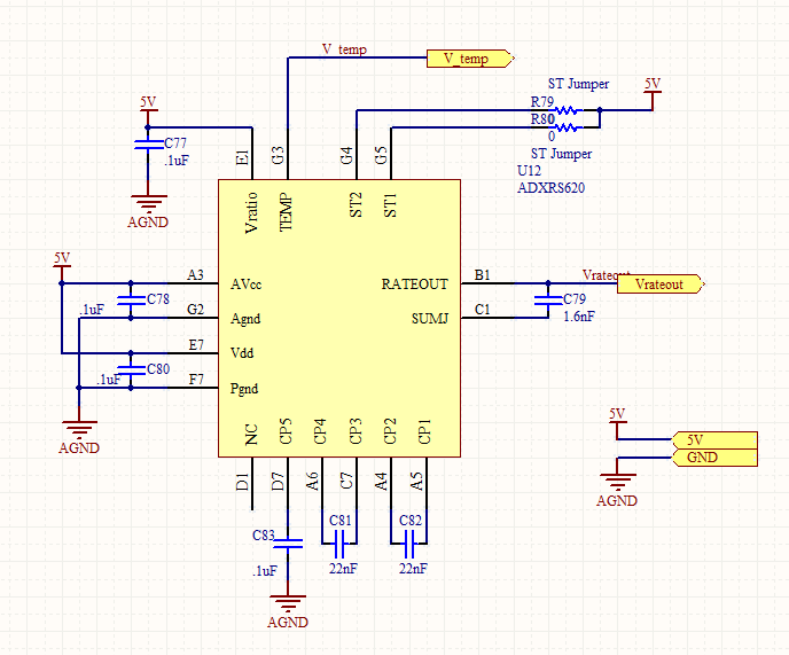


Figure 14: Gyroscope Schematic

For the gyroscopes, bandwidth is set by the equation $f = 1/(2\pi(180\ k\Omega) \times C)$, resulting in a capacitance of 1.6nF for each gyro output.
Range: +/- 250dps

Differential Low Pass Filter and Buffer

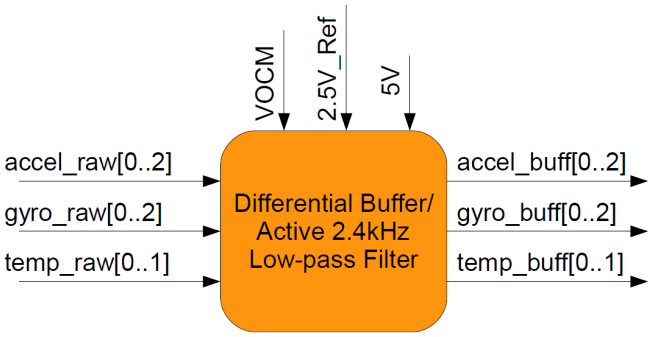


Figure 15: Differential Buffer Block

Name	Type	Specification
accel_raw[0...2]	Electrical Input	Analog outputs from accelerometer. Signal between 0V - 3.3V and has been through passive 500Hz low-pass filter.
gyro_raw[0...2]	Electrical Input	Analog outputs from gyros. Signal between 0V - 5V and has been through passive 500Hz low-pass filter.
temp_raw[0...1]	Electrical Input	Analog temperature outputs from pitch and yaw gyros. Signal between 0V - 5V.
VOCM	Electrical Input	Buffered 2.5V reference.
2.5V_Ref	Electrical Input	2.5V reference voltage from zener diode.
5V	Electrical Input	5V power supply.
accel_buff[0...2]	Electrical Output	Buffered differential pairs of accelerometer data. Signal centered at 2.5V and swings +- 2.5V. Low-pass filtered at 500Hz and 2.4kHz.
gyro_buff[0...2]	Electrical Output	Buffered differential pairs of gyro data. Signal centered at 2.5V and swings +- 2.5V. Low-pass filtered at 500Hz and 2.4kHz.
temp_buff[0...1]	Electrical Output	Buffered differential pairs of temperature data. Signal centered at 2.5V and swings +- 2.5V. Low-pass filtered at 2.4kHz.

Table 8: Differential Buffer Interface Definition

A 500Hz passive low-pass filter is pre-applied to all the analog sensor outputs. To further reduce the noise produced by some of the sensor's resonant frequencies (at around 14kHz), this 2.4kHz active low-pass filter is applied to each signal. The frequency of the active filter is determined by the equation f = 1/(2*pi*R*C), where R and C are in parallel across the output and input of the op-amp. With the chosen values, a 2.4kHz low-pass filter is achieved. Differential op-amps were used because the ADC has differential inputs that should be driven differentially to achieve the best performance. See schematic below.

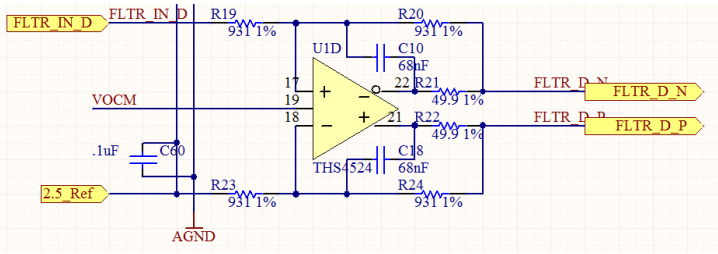


Figure 16: Differential Buffer Schematic

Testing

Apply a 200hz, 1v pk-pk ac signal to the analog inputs. 200Hz is a low enough frequency it should not be filtered. Scope the output signal. The amplitude must be at least 90% of the input amplitude.

Below is an image of an oscilloscope measuring the input and outputs of the buffer/filter block. For this test, a 200Hz 1V pk-pk sine wave (yellow wave, top) was applied to the input of the buffer/filter block. Since the op-amp is differential (and I did not want to short output to ground through the ground of the other probe) I measured the positive (blue wave, middle) and negative (purple wave, bottom) outputs separately, then used the math function to plot positive minus negative (red wave, middle). The image also shows calculations on the right side of the screen. The input amplitude was 0.96V, while the output amplitude was 0.98V. The output amplitude is greater than 90% of the input amplitude therefore the test is passed.

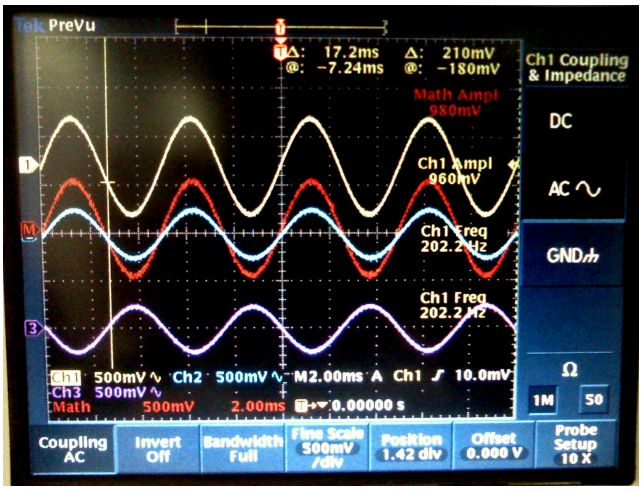


Figure 17: Buffer Passes Low Frequencies Check

Apply a 10kHz, 1v pk-pk ac signal to the analog inputs. 10kHz is a high enough frequency it should be filtered. Scope the output signal. The amplitude out must be less than 50% of the input amplitude.

Below is an image of an oscilloscope measuring the input and outputs of the buffer/filter block. For this test, a 10kHz 1V pk-pk sine wave (large yellow wave in the image) was applied to the input of the buffer/filter block. The positive and negative outputs of the differential op-amp were measured separately, and then subtracted to obtain the full output signal. The output signal is shown in red. The calculations on the right

side of the screen show that the input amplitude was 0.956V and the output amplitude was 0.24V. The output amplitude is less than 50% of the input amplitude therefore the test is passed.

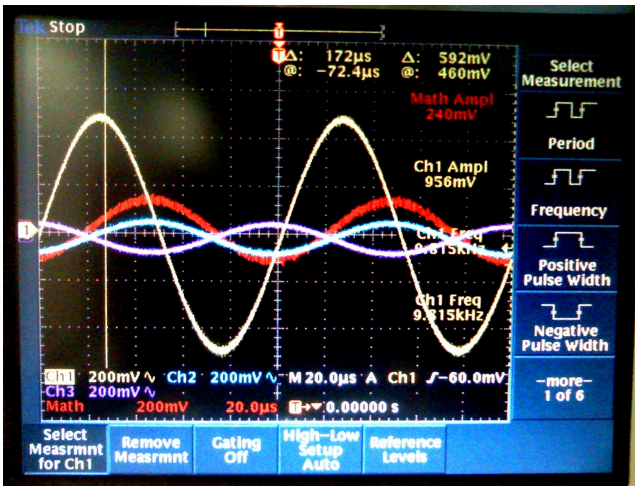


Figure 18: Buffer Filters High Frequencies Check

Power Supply

Figure 19: IMU Power Supply Block

Name	Type	Specification
V_bat	Electrical Input	Power Input From Battery 9V-16V, 12V nominal.
3.3V	Electrical Output	3.3V Power Output for IMU_CPU board. 200mA output

Table 9: IMU Power Interface Definition

This is a simple linear regulator from TI. It has a maximum output current of 500mA, but our system should only need ~100mA.

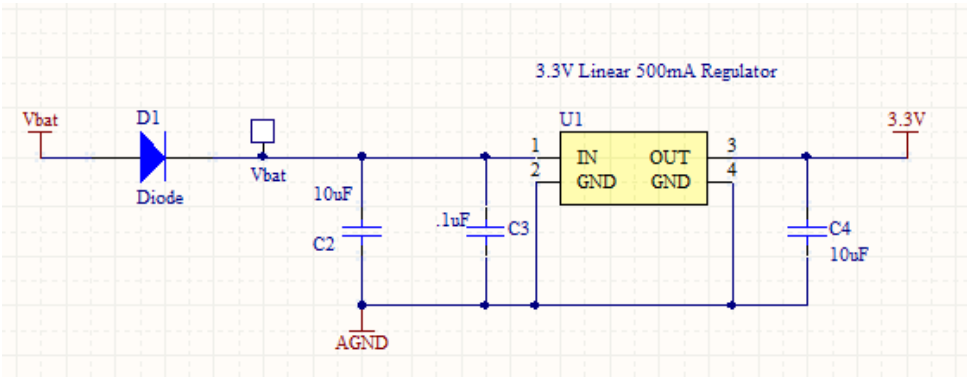


Figure 20: IMU Power Supply Schematic

MCU – Motor Control Unit

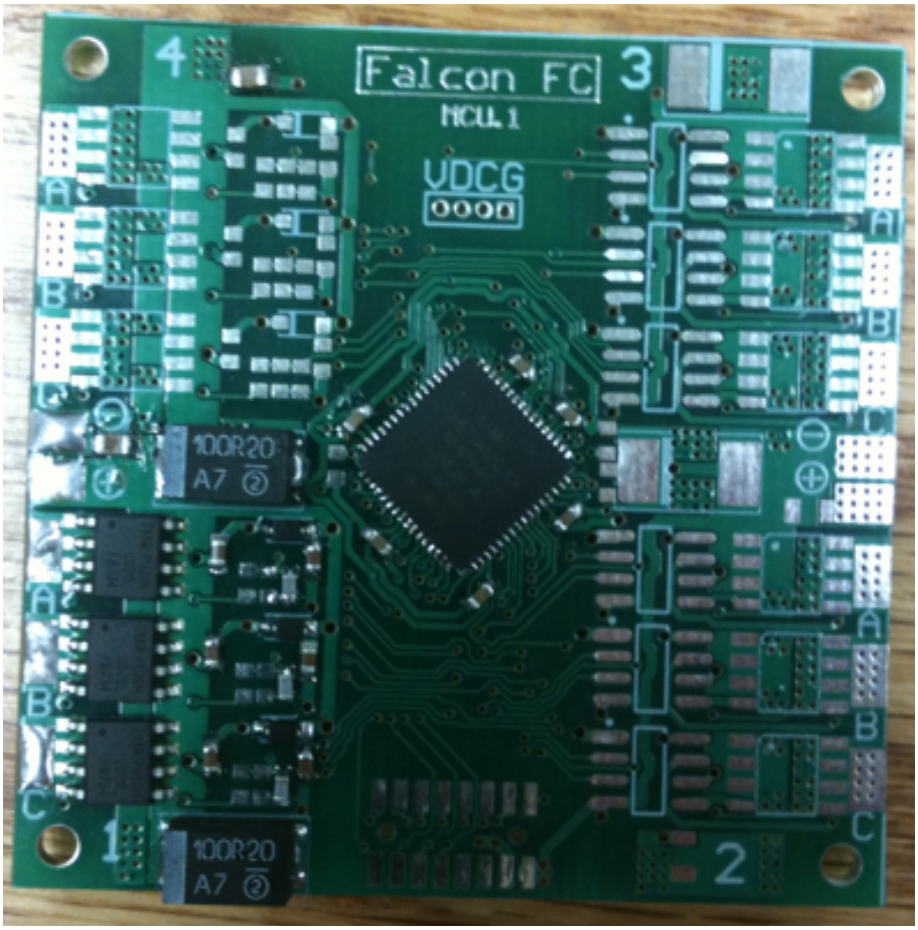


Figure 21: MCU Board

MOSFET Half-Bridge

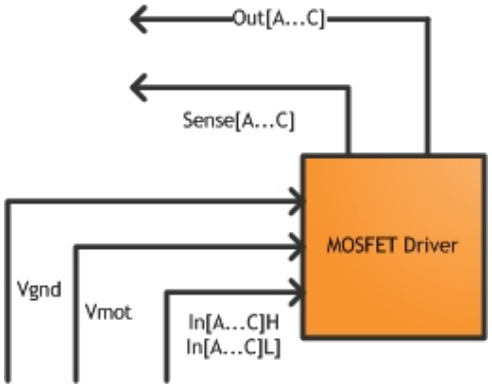


Figure 22: MOSFET Half-Bridge Block

Name	Type	Specification
Vmot	Electrical Input	Motor Power Input
Vgnd	Electrical Input	Motor Power Input
In[A..C]H	Electrical Input	3.3V Pulse Width Modulation, High side FETs
In[A..C]L	Electrical Input	3.3V Pulse Width Modulation, Low side FETs
Out[A..C]	Electrical Output	Power Output To Motors
Sense[A..C]	Electrical Output	0-3.3V Analog Feedback From Motor Coils

Table 10: MOSFET Half-Bridge Interface Definition

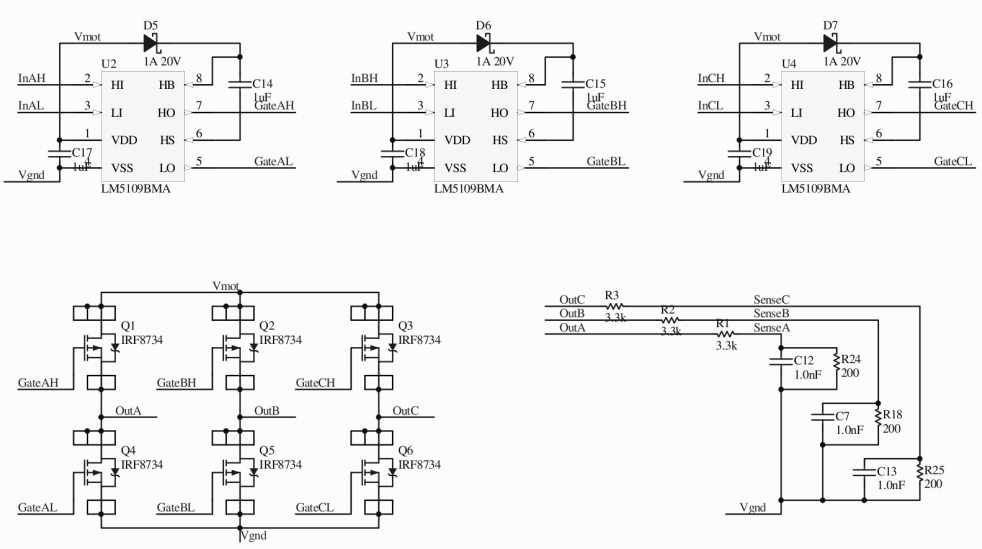


Figure 23: MOSFET Half-Bridge Schematic

MOSFET Validation
Vds(max) = 30V, Id(max) @ 70 Degrees C = 17A, Rds(on) = 3.5mOhms, Pdiss(max) @ 70 Degrees C = 1.6W
At out 15A max motor current, Pdiss = I2*R = 152*(3.5x10-3) = 0.7875W

MOSFET Driver Validation
Imax = 1A, Vdd = 8V to 14V, I/O threshold = 0.8V(low)-2.2V(high)

Sense Voltage Divider Validation
Vout = Vin(R2/Rtotal) = 12.6(200/3.5e3) = 0.72V max Sense Voltage

Microprocessor

Motors

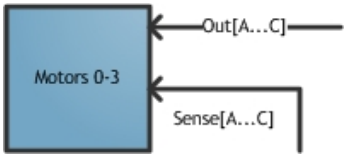


Figure 24: Motors Block

Name	Type	Specification
Sense[A..C]	Electrical Output	0-3.3V Analog Feedback From Motor Coils for Back EMF Sensing
Out[A..C]	Electrical Input	Power Input To Motors

Table 11: Motor Block Interface Definition

Motors were selected in previous years. In order to maintain level flight, the motors had to lift and accelerate the 2kg quadrotor with relative ease. It was determined that 15A at 9V should be more than enough to power the quadrotor. The motors were also selected based on availability; the motors chosen are commonly used by hobbyists and easily replaceable.

Power Supply

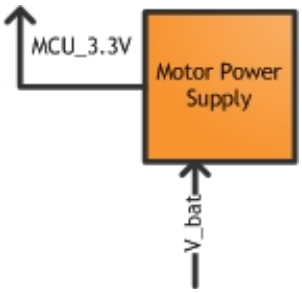


Figure 25: Motor Power Block

Name	Type	Specification
V_bat	Electrical Input	9-12.6V, 20mA max
MCU_3.3V	Electrical Output	3.3V, 20mA max

Table 12: Motor Power Interface Definition

The linear voltage regulator chips are designed to output 50mA. The total current draw of all parts at each voltage level will not exceed 20mA. With this in mind, our voltage regulators should be more than adequate.

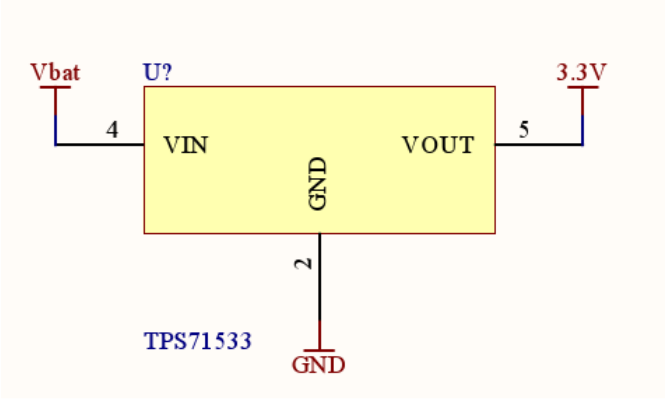


Figure 26: Motor Power Schematic

System Testing and Validation

1khz IMU Update Speed

The IMU needs to provide an inertial state estimate at a rate high enough to allow the vehicle to be as stable as possible. This is important because excess platform vibrations negatively influence the camera and lidar sensors.
The IMU must provide an inertial state estimate once every millisecond.

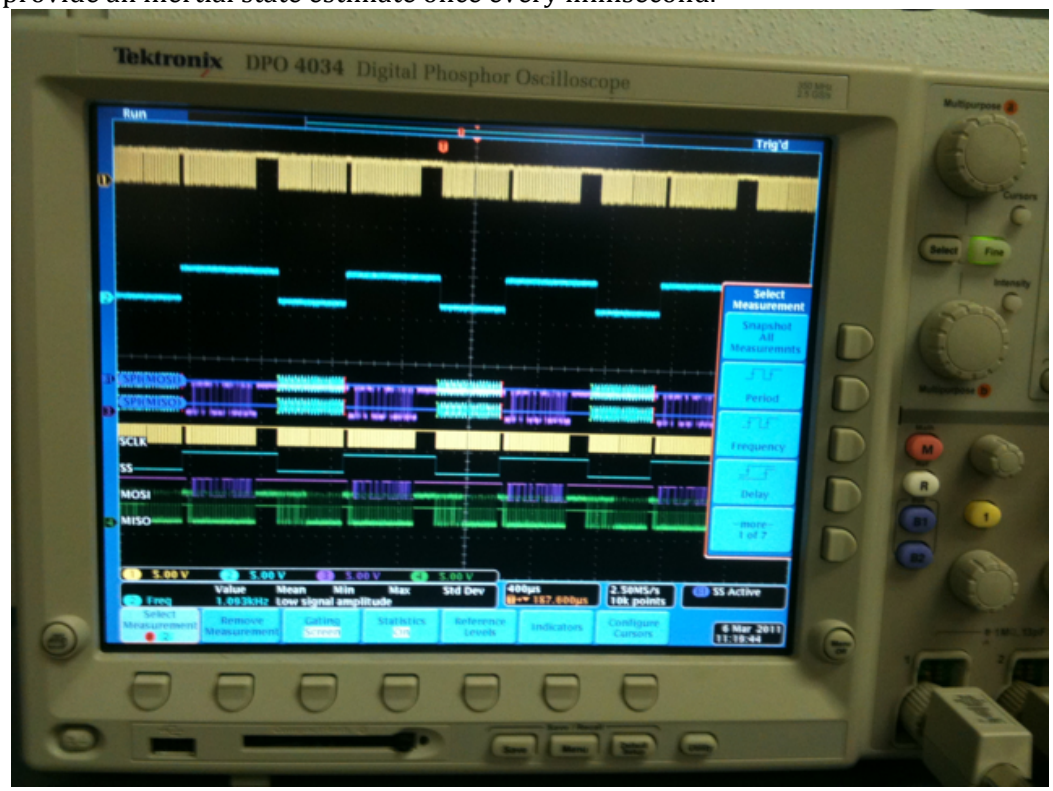


Figure 27: Oscilloscope Data and Refresh Rate

Transfers Data Packets Between Boards

Data must be accurate to maintain a stable platform. FCU receives data packets from the IMU and MCU. IMU transfers roll, pitch, yaw. MCU transfers motor speeds. Data streams to the FCU through spi and values make sense.

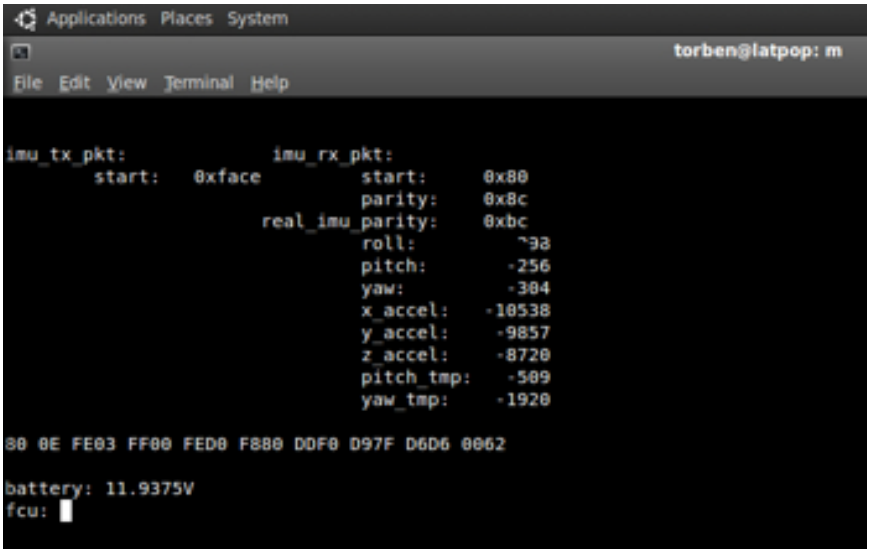


Figure 28: IMU Data with SPI

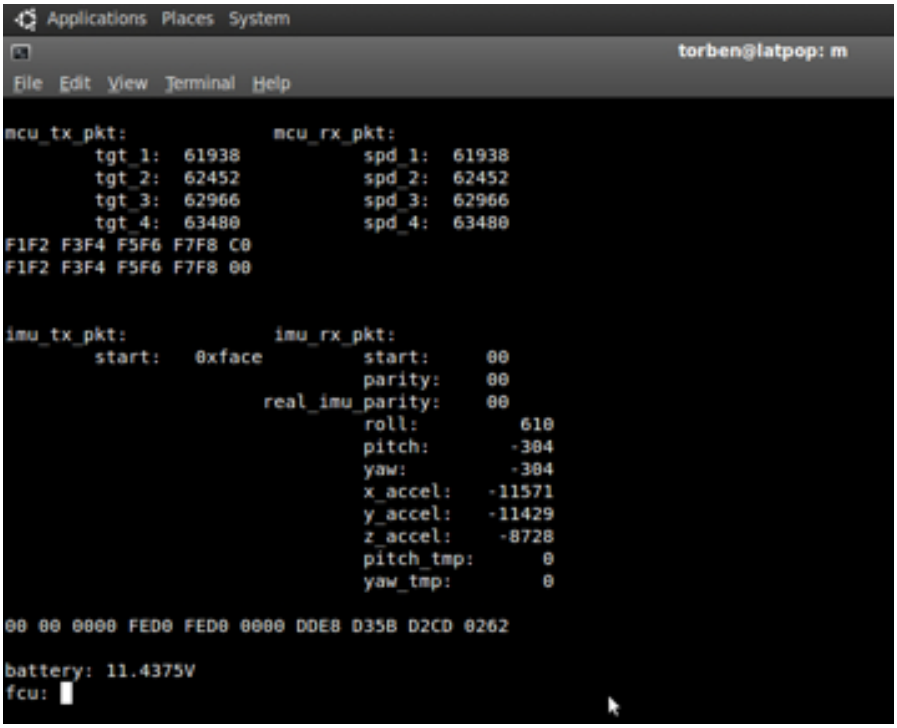


Figure 29: MCU Data with SPI

Battery Powered

The flight control system continues operating smoothly between 9 and 13 volts. Voltage provided by a power supply so we could easily modify the voltage.

See the video:
http://beaversource.oregonstate.edu/projects/44x201007/attachment/wiki/TestEvidence/IMG_0524.mov

External Kill Switch

The flight controller must include a fail-safe kill switch. In our case, the flight controller must accept a wireless signal that causes all motors to be disabled upon activation.

http://web.engr.oregonstate.edu/~cooleys/kill_switch.MOV

Interfaces with the OSU IARC entry

Our flight control system must interface with the vehicle's main computer. The flight control unit (FCU) uses a serial over usb to interface to the master computer on the OSU IARC entry. See Data Packets Transferred test for serial over USB data.

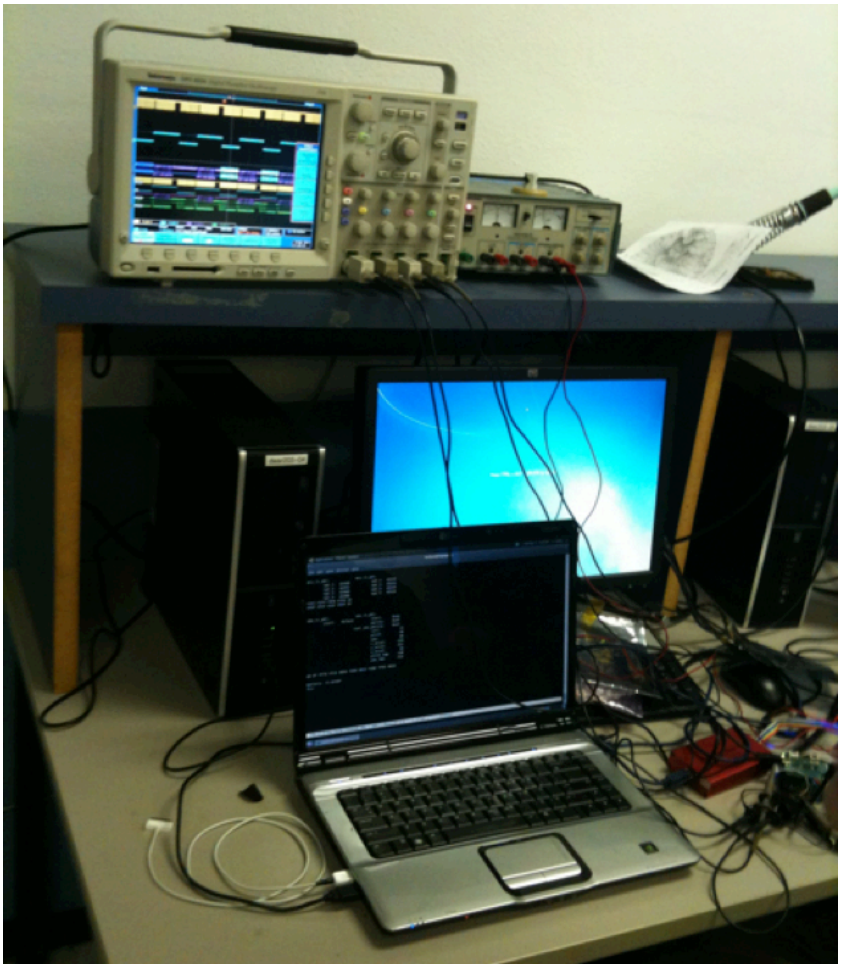


Figure 30: Lab Setup with USB

Light Weight

The OSU IARC team needs the flight control system to be as light-weight as possible in order to keep their platform under 1.5kgs. Indeed, the entire flight control system should weight less than 300 grams.

Figure 31: Weight

Powers the four flight motors

The IARC submission uses four brushless DC motors. To power these motors, our motor control unit (MCU) should be able to supply a minimum of 15A per motor (60A total).

Vibration Resistance

All board-to-board connectors should be secured with screws or other similar secure fasteners. Fasteners and spacers hold boards stable and equally spaced. Solder joints hold through handling and testing.

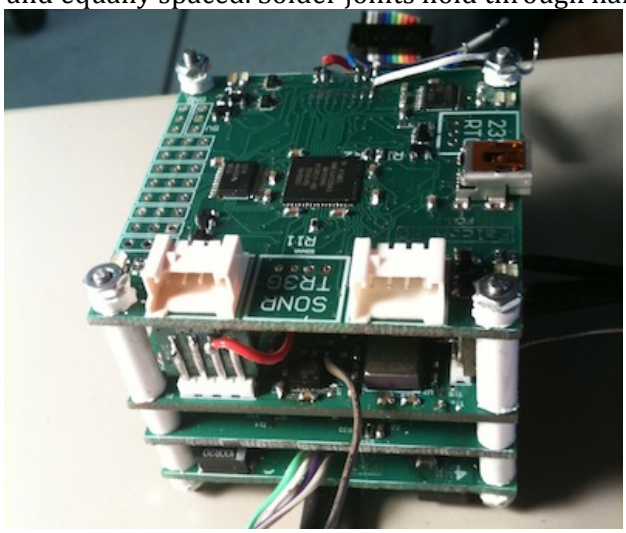


Figure 32: Board with Fasteners

Wireless Serial Communication

FCU receives serial data packets wirelessly through an xbee module. Data is transmitted accurately and in a timely manner.

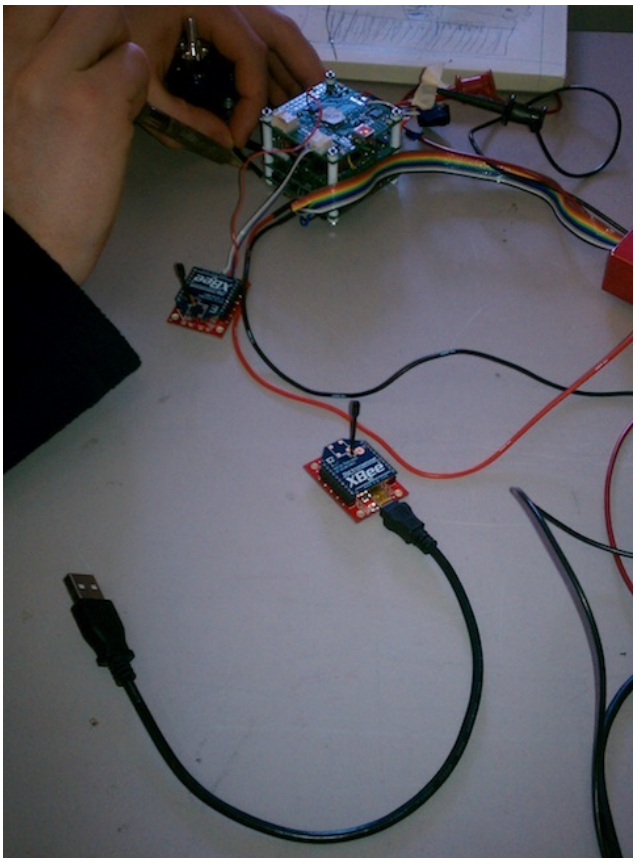


Figure 33: Wireless with Xbee Modules

Realtime Data Graphing

Data is graphed as it's received. Graph updates at least one point per second.

Figure 34: Realtime Data Graphing

TI Parts Used

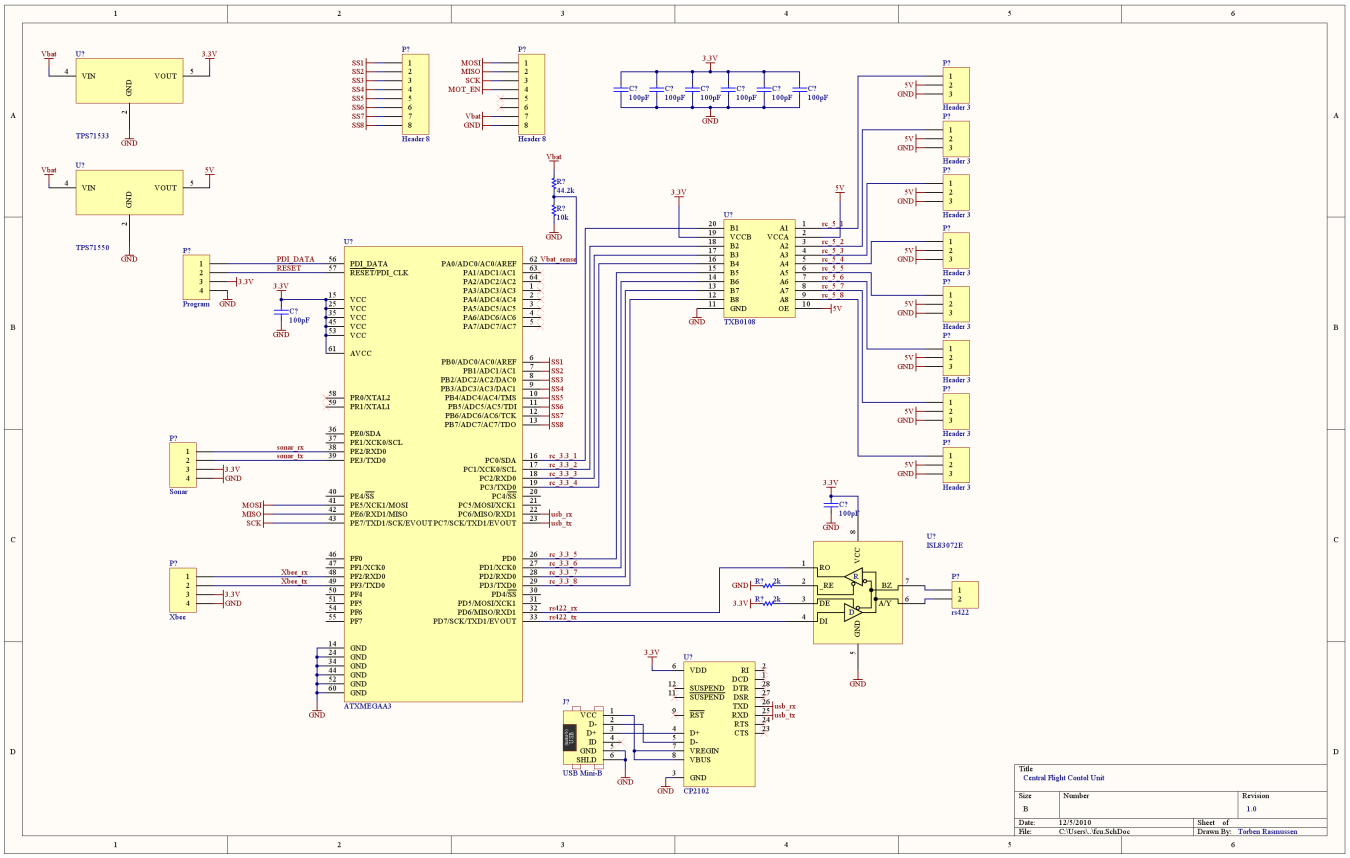
Texas Instruments parts play several essential roles at the core of our inertial measurement unit (IMU) design. All of our sensors are buffered by TI op-amps; the analog inertial sensors are buffered by a rail-to-rail op-amp (OPA4348). Then the signals are fed into a differential op-amp (THS4524). This op-amp setup converts our single-ended sensor signals into differential pairs, centered around 2.5V, as well as providing an active low pass filter. The sensor signals are then fed into an eight-channel delta-sigma analog-to-digital converter (ADS1178). The ADS1178 allows us to simultaneously sample eight channels at rates of up to 52K samples/sec.

High-speed simultaneous sampling is crucial for inertial-based control and platform stability. The ADS1178 was the obvious best choice for our application.

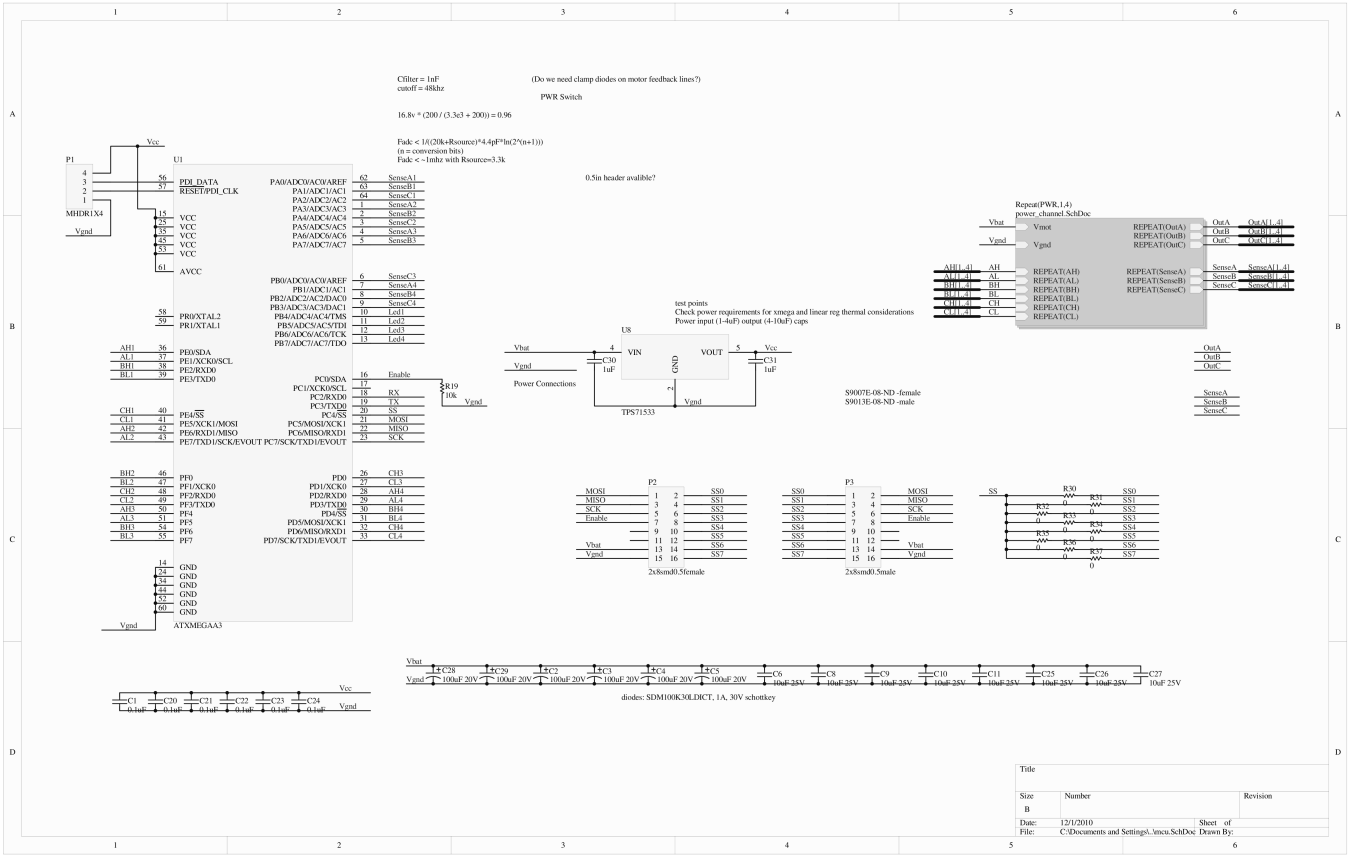
We picked the powerful TMS320F28035 (Piccolo F28035) as our filtering processor to leave a large margin for any computationally complex filtering algorithms we may use. This beast of a processor runs at 60MHz, has 32-bit operations, and has a floating-point co-processor that can do single cycle 32-bit floating point operations while the main processor is busy servicing i/o requests. This revision does not actually use the floating-point coprocessor, but it would be used if we were doing more complex data filtering.

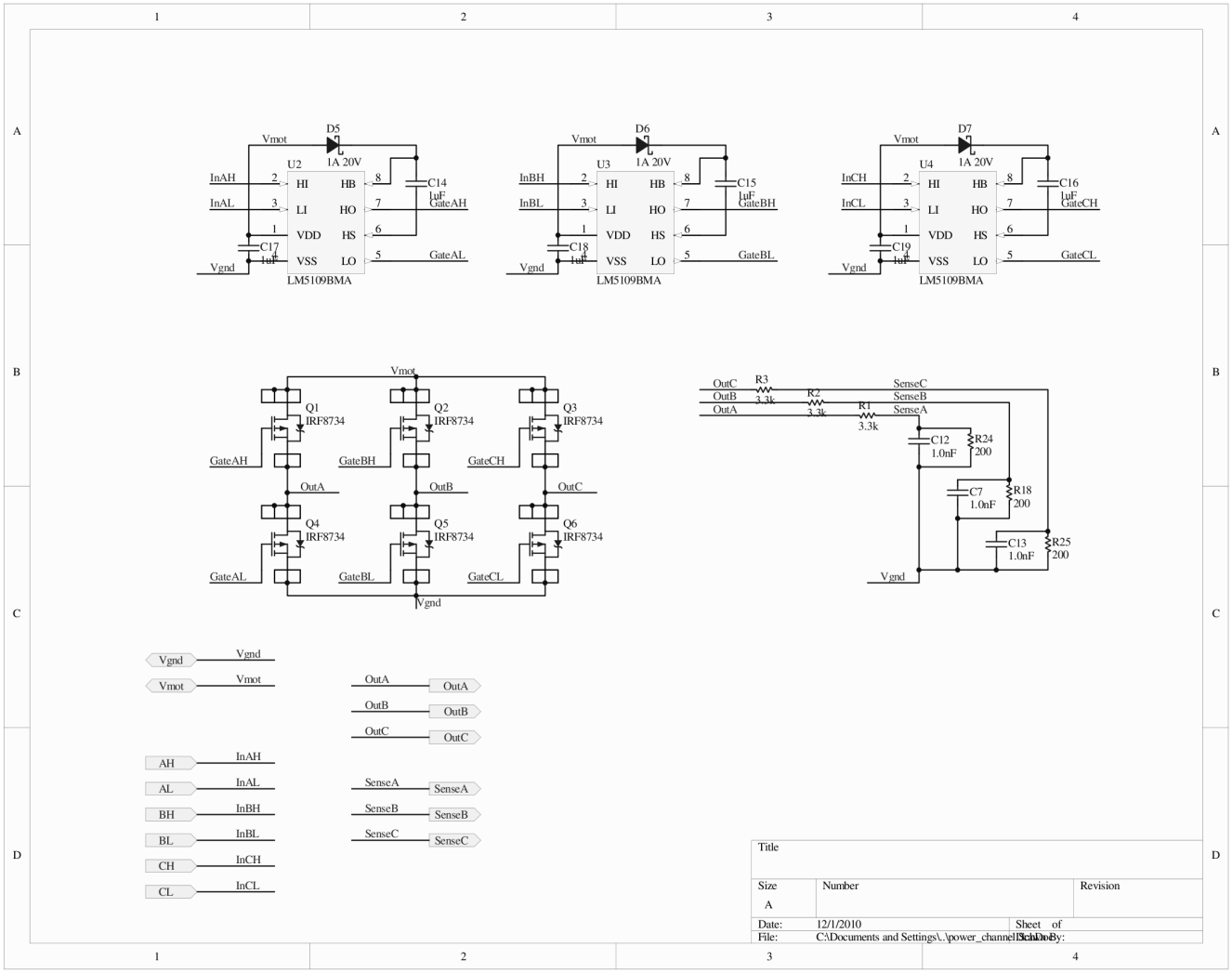
Appendix I: Schematics

FCU Schematic

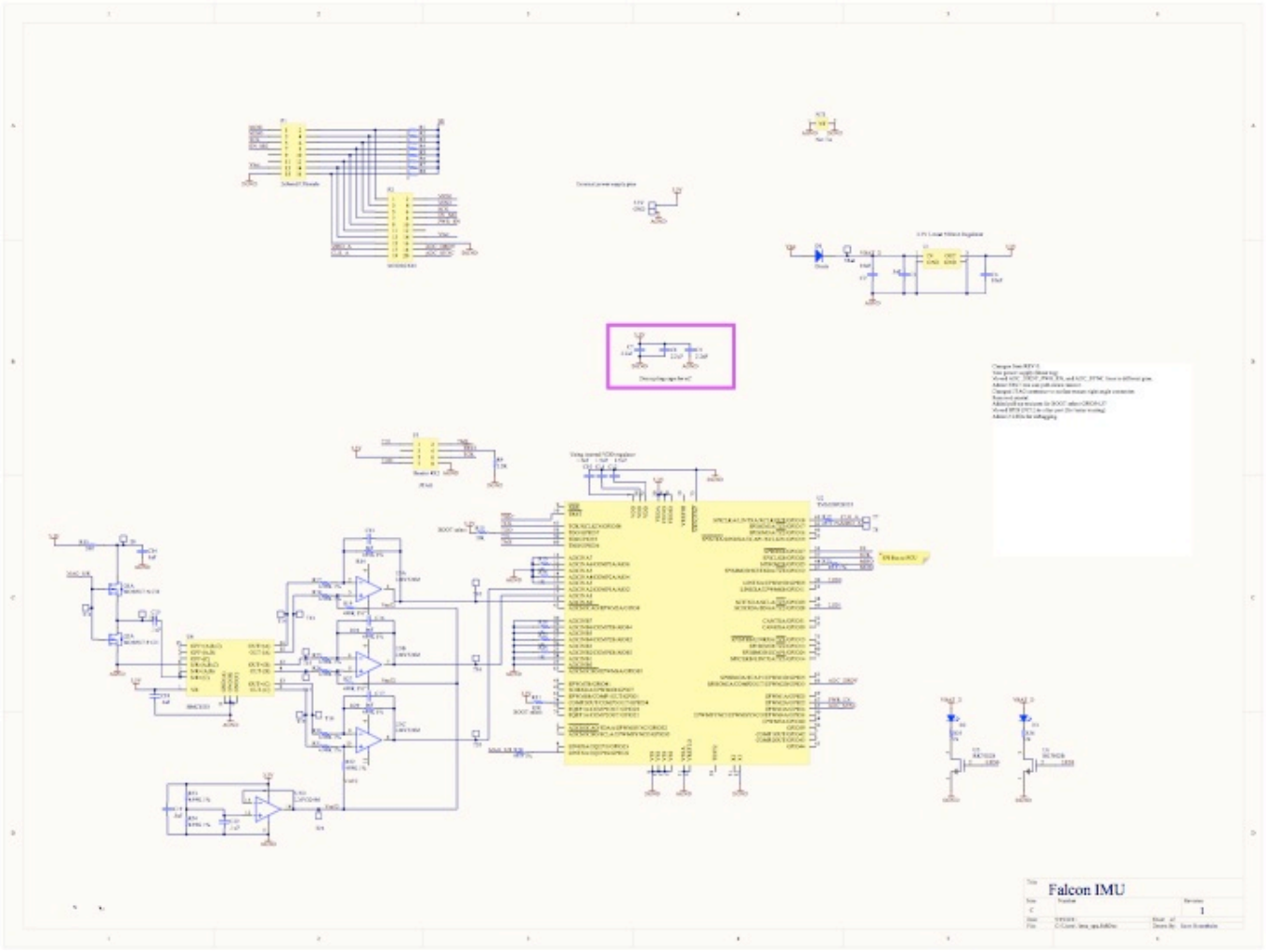


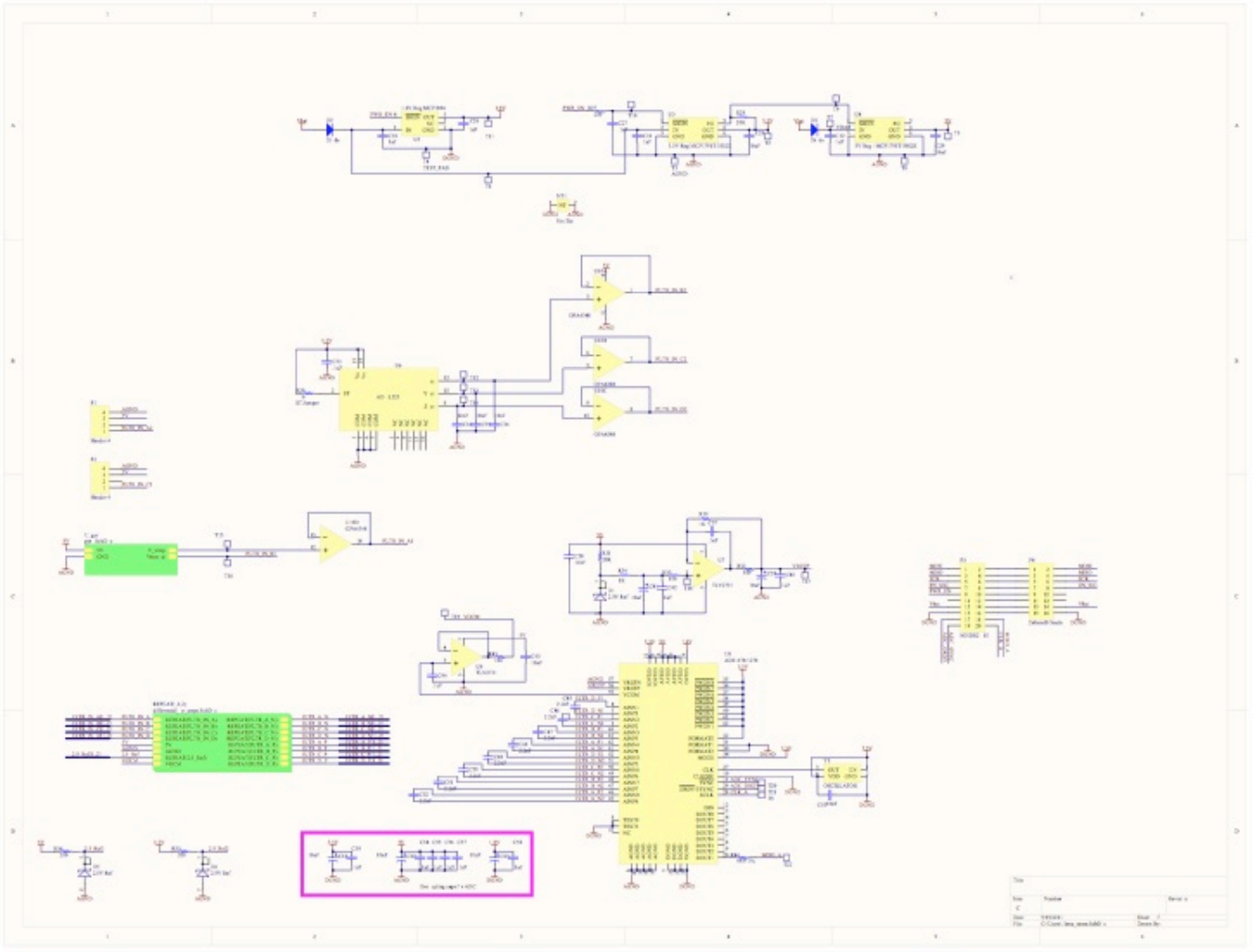
MCU Schematic




Vmot
D5
1A 20V
U2
InAH
InAL
Vgnd
HI
LI
VDD
VSS
HB
HO
HS
LO
C14
1uF
GateAH
C17
1uF
GateAL
LM5109BMA
D6
U3
InBH
InBL
C15
GateBH
C18
GateBL
D7
U4
InCH
InCL
C16
GateCH
C19
GateCL
Q1
IRF8734
Q2
Q3
Q4
Q5
Q6
OutA
OutB
OutC
OutC
OutB
OutA
R3
R2
R1
3.3k
SenseC
SenseB
SenseA
C12
1.0nF
R24
200
C7
R18
C13
R25
Vgnd
Vmot
AH
AL
BH
BL
CH
CL
InAH
InAL
InBH
InBL
InCH
InCL
SenseA
SenseB
SenseC
Title
Size
A
Number
Revision
Date:
12/1/2010
Sheet of
File:
C:\Documents and Settings\..\power_channel3.SchDoc
Drawn By:

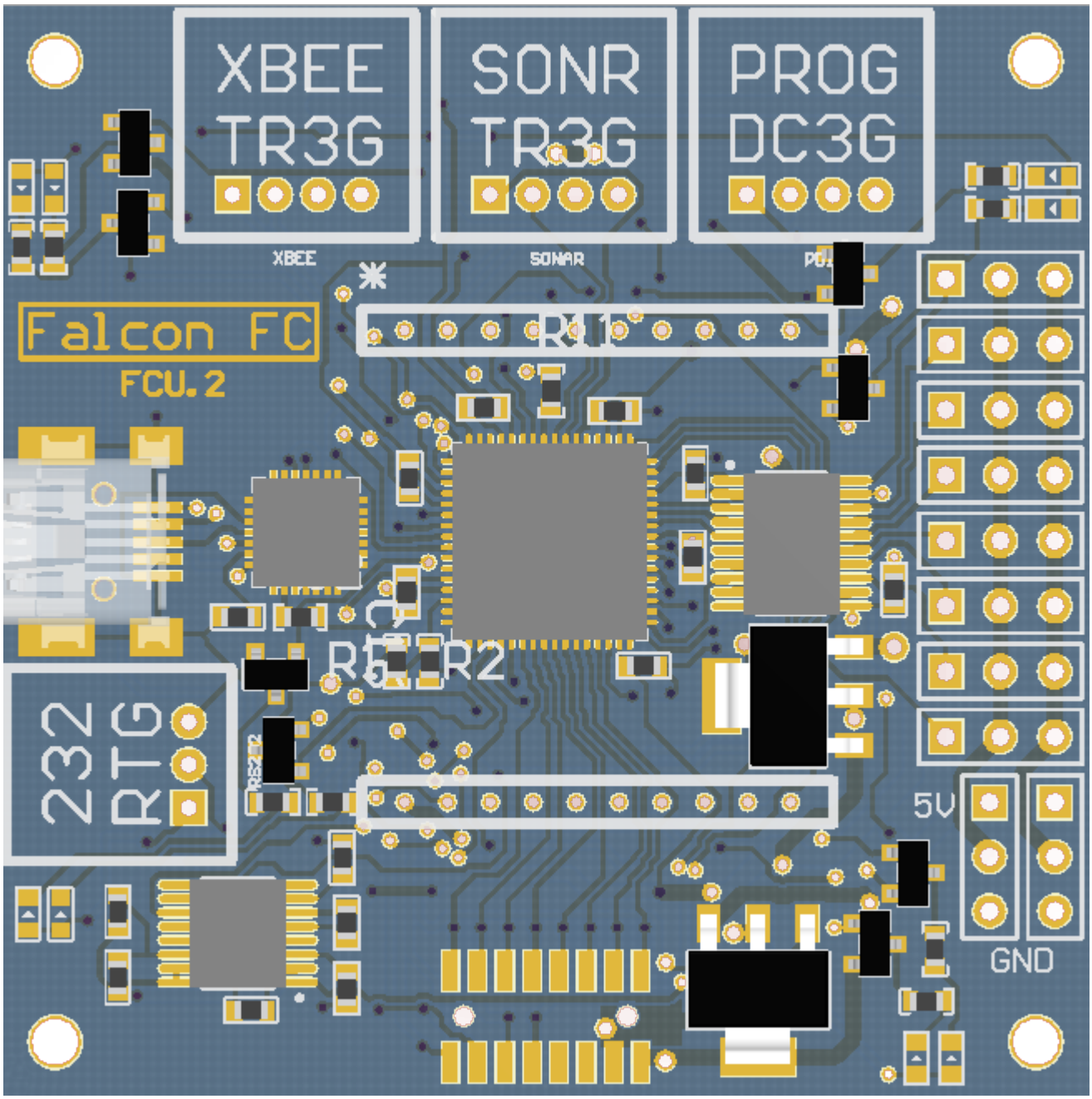
IMU Schematic



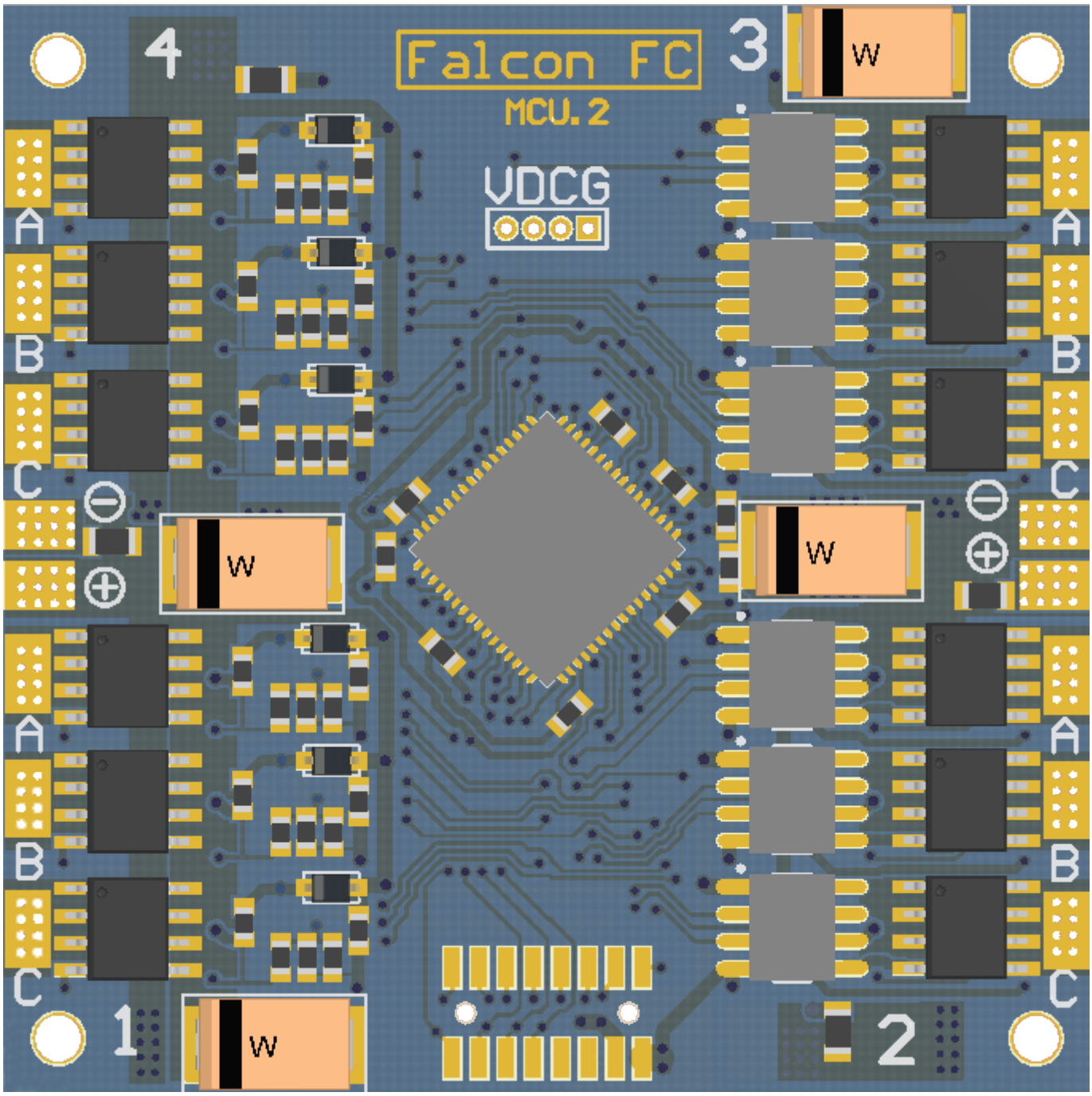


Appendix II: PCB Layout

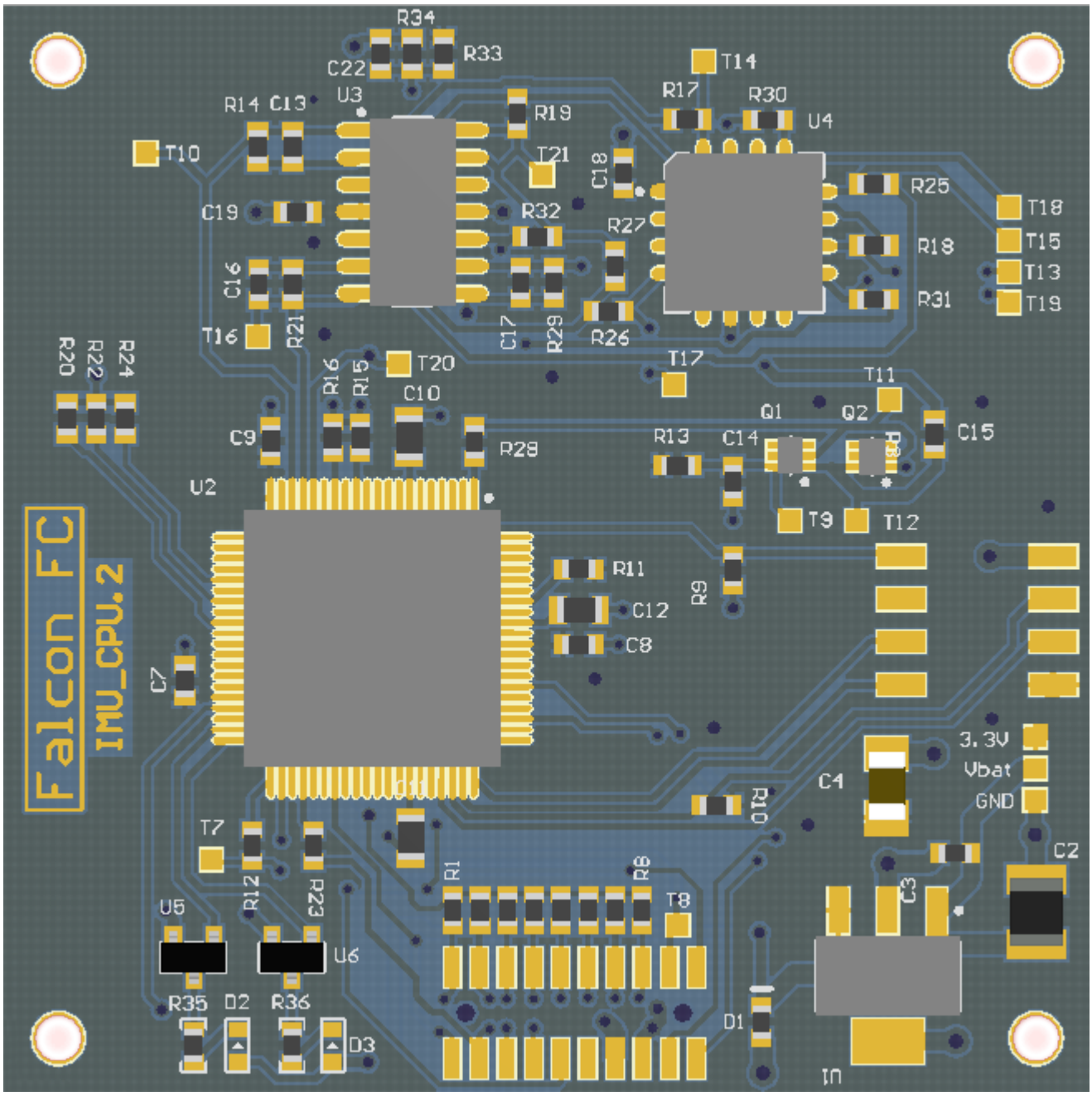
FCU Board

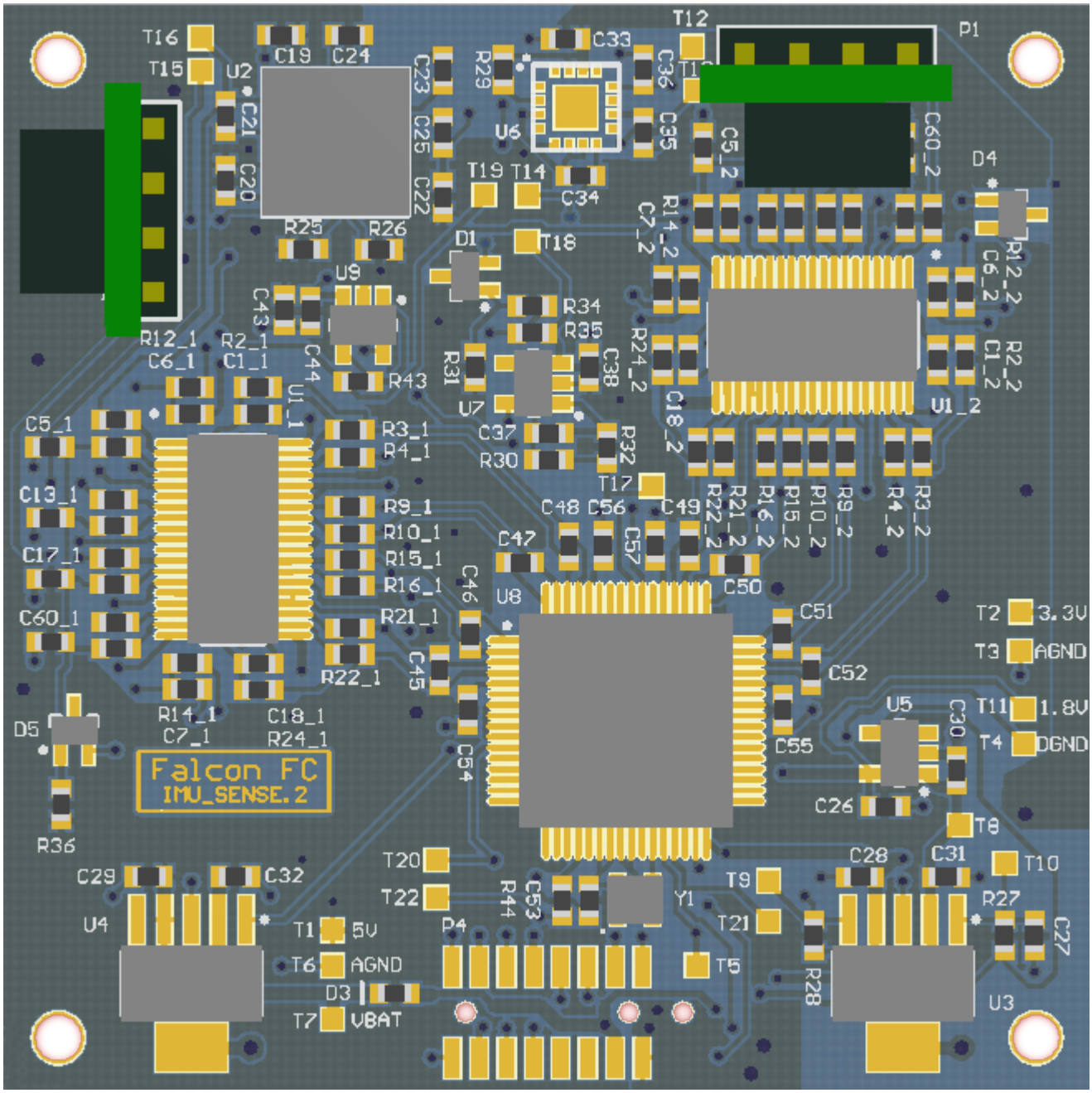


MCU Board



IMU Board


Falcon FC
IMU_CPU.2
R34
R33
C22
U3
R14
C13
T10
R19
T21
C18
T14
R17
R30
U4
R25
R18
R31
T18
T15
T13
T19
C19
R32
R27
C16
T16
R21
C17
R29
R26
T17
T11
T20
R16
R15
C10
Q1
Q2
C15
C9
R28
R13
C14
T9
T12
U2
R11
C12
C8
R9
C7
R10
C4
3.3V
Vbat
GND
C2
C3
T7
R12
R23
R1
R6
T8
U5
U6
R35
D2
R36
D3
D1
U1


Falcon FC
IMU_SENSE.2
T16
T15
U2
C19
C24
C23
C25
C22
C21
C20
R25
R26
U9
C43
C44
R43
R12_1
R2_1
C6_1
C1_1
U1_1
C5_1
C13_1
C17_1
C60_1
R3_1
R4_1
R9_1
R10_1
R15_1
R16_1
R21_1
R22_1
C46
C45
R14_1
C7_1
C18_1
R24_1
D5
R36
C54
C29
C32
U4
T1 5V
T6 AGND
D3
T7 VBAT
T20
T22
P4
R44
C53
Y1
T9
T21
T5
C28
C31
T10
R27
C27
R28
U3
T8
C26
U5
C30
T11 1.8V
T4 DGND
C55
C51
C52
T2 3.3V
T3 AGND
C50
C47
C48
C56
C49
C57
T17
R32
R30
C37
U7
R31
R34
R35
C38
T18
D1
T19
T14
C34
U6
C33
C36
C35
T12
T13
P1
C5_2
C60_2
D4
R14_2
C7_2
R12_2
C6_2
R24_2
C18_2
U1_2
R2_2
C1_2
R22_2
R21_2
R16_2
R15_2
R10_2
R9_2
R4_2
R3_2

Appendix III: Parts List