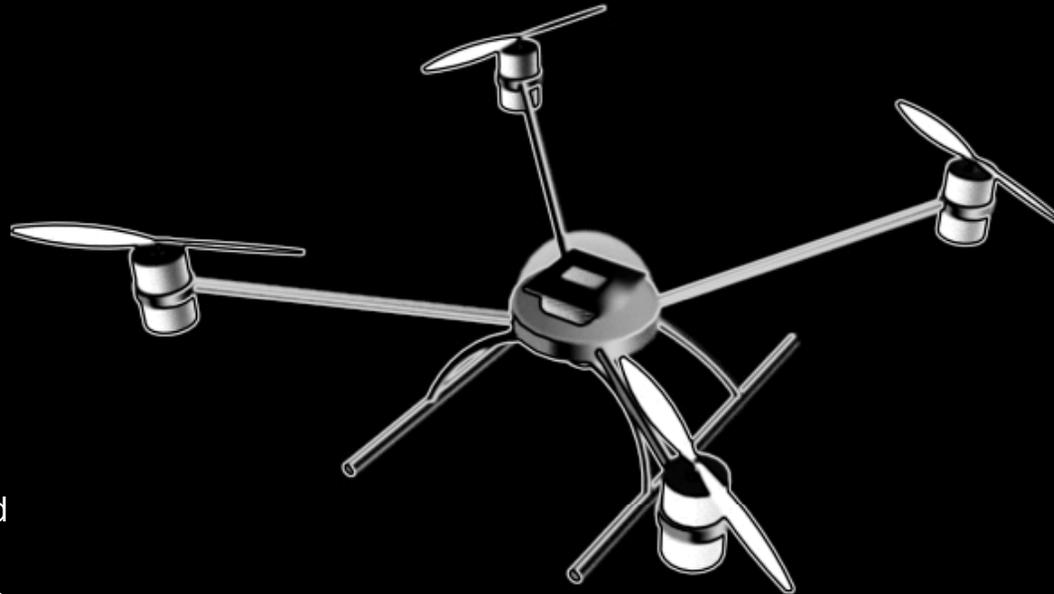


# U.A.R.C.



## Group# 9

Clint Mansfield  
Edwin Giraldo  
Jeremy Brooks

Unmanned Aerial Reconnaissance Copter

Summer 2009

# MOTIVATION

---

- Design a low-cost Unmanned Vehicle that can gather information.
- Military application to protect personnel in unknown terrain.
- Build a vehicle that uses little or no user input.

# GOALS

---

- Fly autonomously without user interaction.
- Stable hover
- Low maintenance
- Lightweight components
- Obstacle avoidance
- Low power consumption

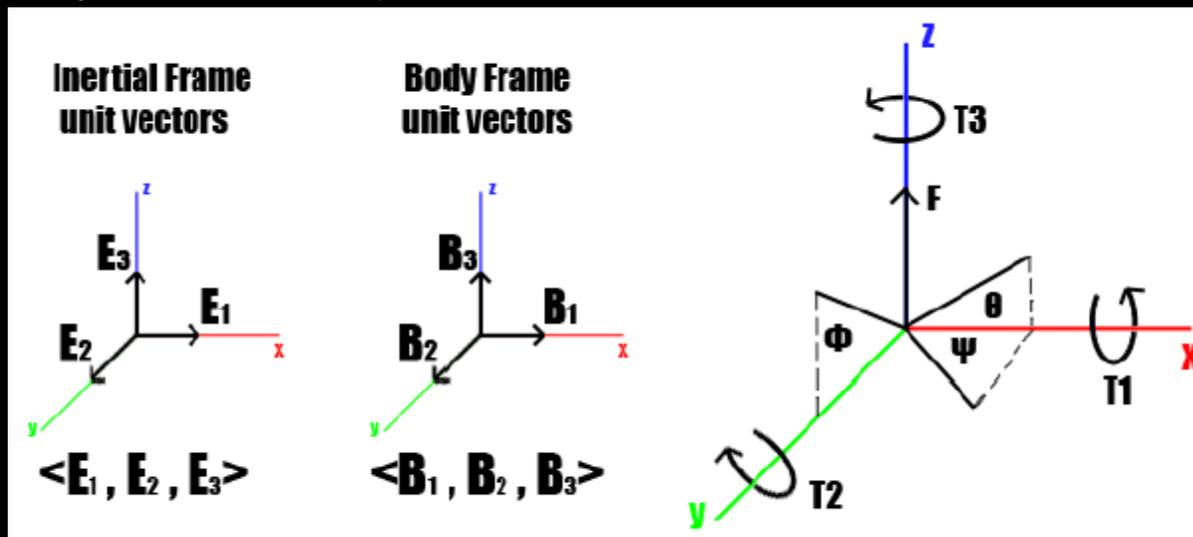
# SPECIFICATIONS

---

- Small in design (portable) ~2ft. X 2 ft. X 7 in.
- Electrically powered by a standard 2-cell 7.4 V rechargeable Li-Po battery.
- 10 – 15 minute flight time operation.
- 6 Degrees of freedom monitored by sensors to maintain stability.
- Throttle is controlled by a 1 – 2 ms PWM signals sent from  $\mu\text{C}$ , one pulse per 20 ms.
- Fail safe system of less than 15 feet away.
- Step down input voltage to 3.3V and 5V for sensors
- Avoid obstacles that range 3 feet or less.

# FLIGHT DYNAMICS

- A good dynamical derivation will allow for realistic simulation design
- UARC is modeled as a symmetrical rigid body
- There are 6 degrees of freedom
- The system is controlled by four inputs being
  - F - total thrust on z-axis
  - T1 - torque about x-axis
  - T2 - torque about y-axis
  - T3 - torque about z-axis
- Since the number of input actuators is less than the DOF, the system is under actuated
- UARC body frame **B** will be represented in inertial frame **E**.



# FLIGHT DYNAMICS (CON'T)

The equations of motion are derived through Newton - Euler formulation

$$F = ma$$

$$\tau = I\alpha$$

These lead to the following

$$\begin{bmatrix} mI & 0 \\ 0 & J \end{bmatrix} \begin{bmatrix} \dot{v}^b \\ \dot{\omega}^b \end{bmatrix} + \begin{bmatrix} \omega^b \times mv^b \\ \omega^b \times J\omega^b \end{bmatrix} = \begin{bmatrix} f^b \\ \tau^b \end{bmatrix}$$

*I is an identity matrix*

*J is an inertial matrix*

*$v^b$  velocity in body frame*

*$\omega^b$  angular velocity*

*$f^b$  is the total of all forces on the rigid body*

*$\tau$  is the sum of all torques on body*

# FLIGHT DYNAMICS (CON'T)

- Ultimately the following equations of translation and rotation can be derived

$$m\ddot{X} = (\cos\Phi \sin\theta \cos\psi + \sin\Phi \sin\psi) u_1 \quad J_x \ddot{\Phi} = \dot{\theta}\dot{\psi} (J_y - J_z) + l u_2$$

$$m\ddot{Y} = (\cos\Phi \sin\theta \sin\psi - \sin\Phi \cos\psi) u_1 \quad J_y \ddot{\theta} = \dot{\Phi}\dot{\psi} (J_z - J_x) + l u_3$$

$$m\ddot{Z} = mg - (\cos\Phi \cos\theta) u_1 \quad J_z \ddot{\psi} = \dot{\Phi}\dot{\theta} (J_x - J_y) + u_4$$

- Below  $u_1, u_2, u_3, \text{ and } u_4$  represent the actuator control inputs.
- These control inputs relate the thrust induced from the individual motors to the square of angular velocity and other aerodynamic coefficients from the props.

$$u_1 = \sum_{i=1}^4 T_i = b (\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \quad \text{Vertical force input}$$

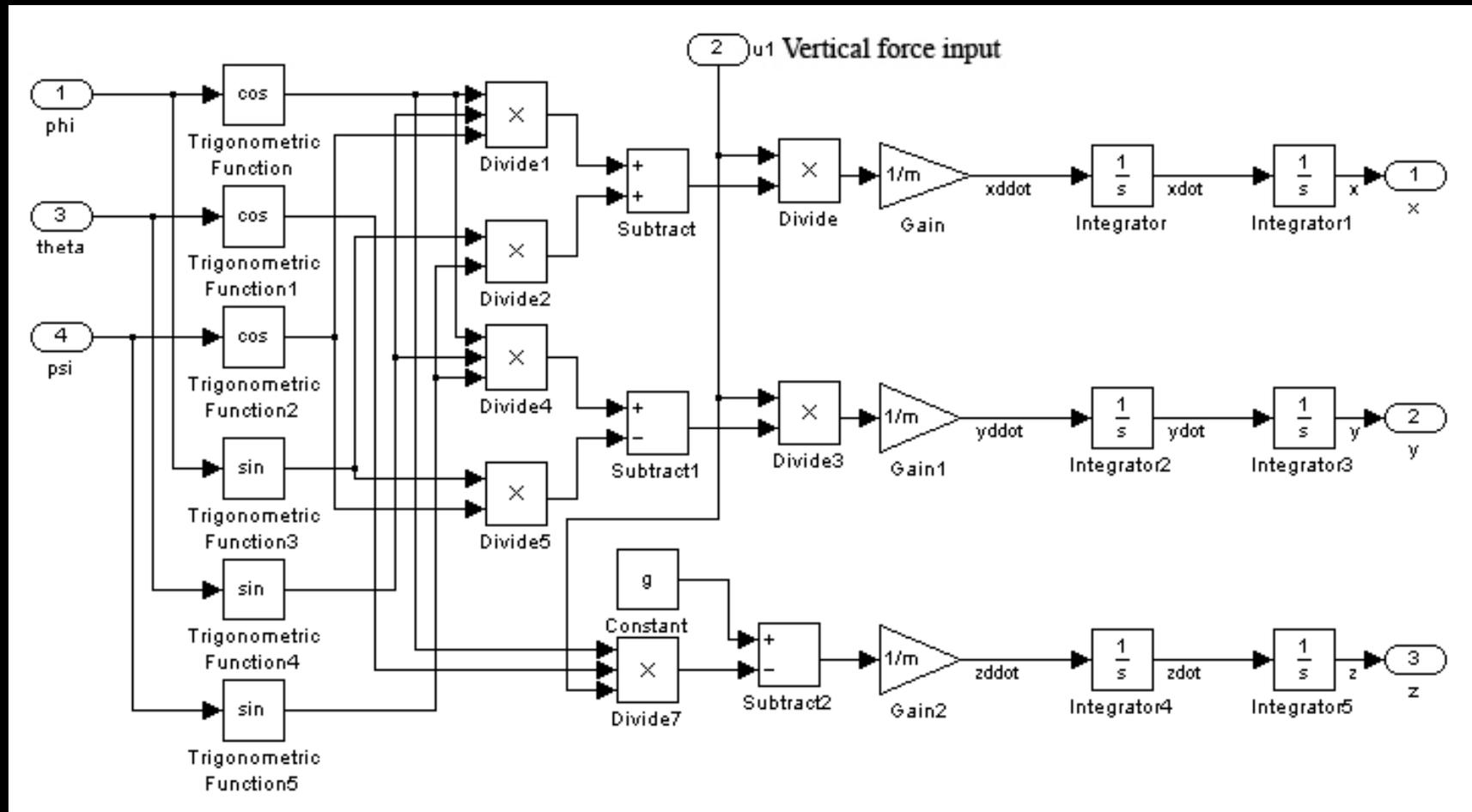
$$u_2 = b (\Omega_4^2 - \Omega_2^2) \quad \text{Roll actuator input}$$

$$u_3 = b (\Omega_3^2 - \Omega_1^2) \quad \text{Pitch actuator input}$$

$$u_4 = b (\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \quad \text{Yaw moment input}$$

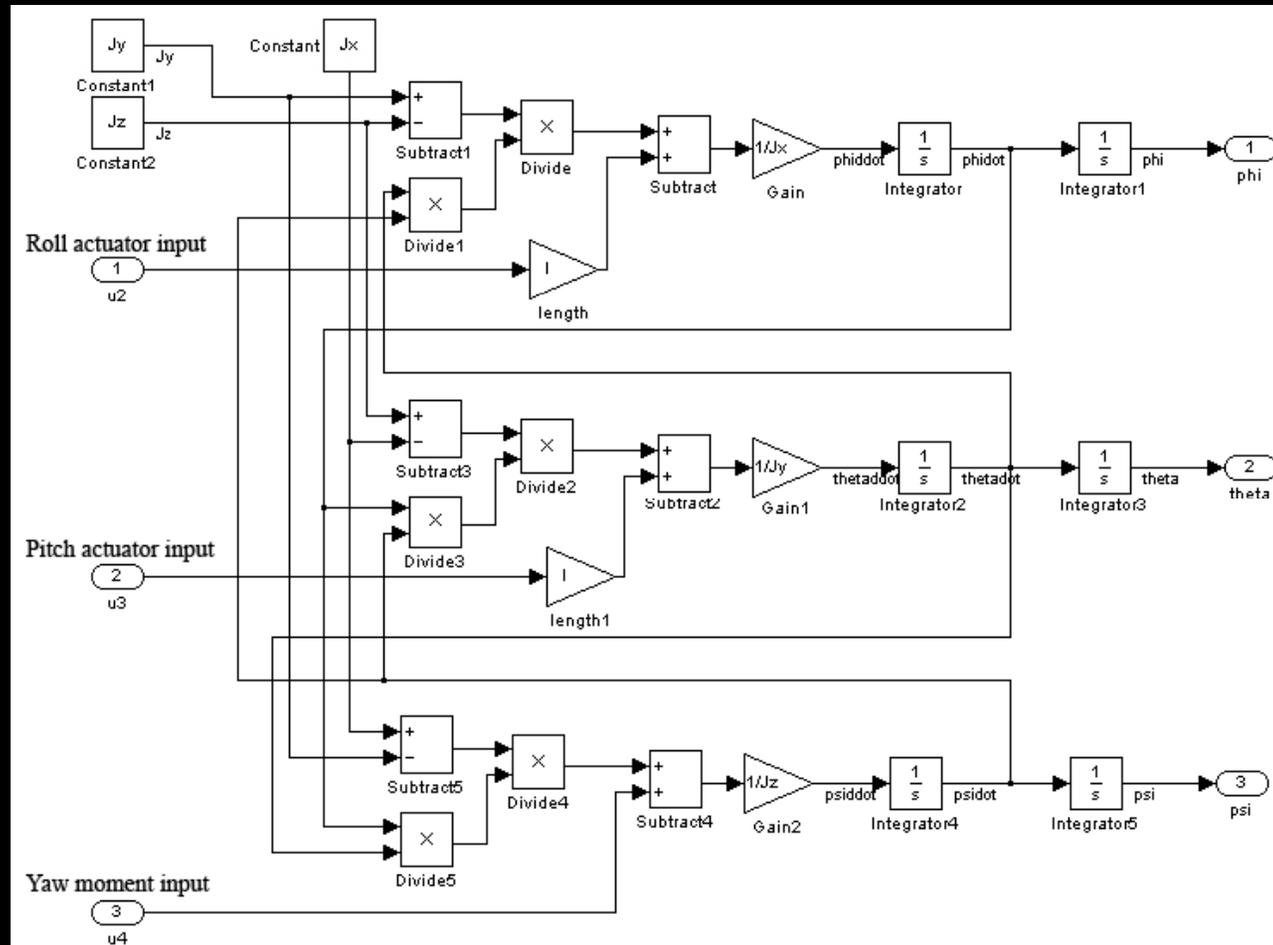
# SIMULATION

## Translational Dynamics



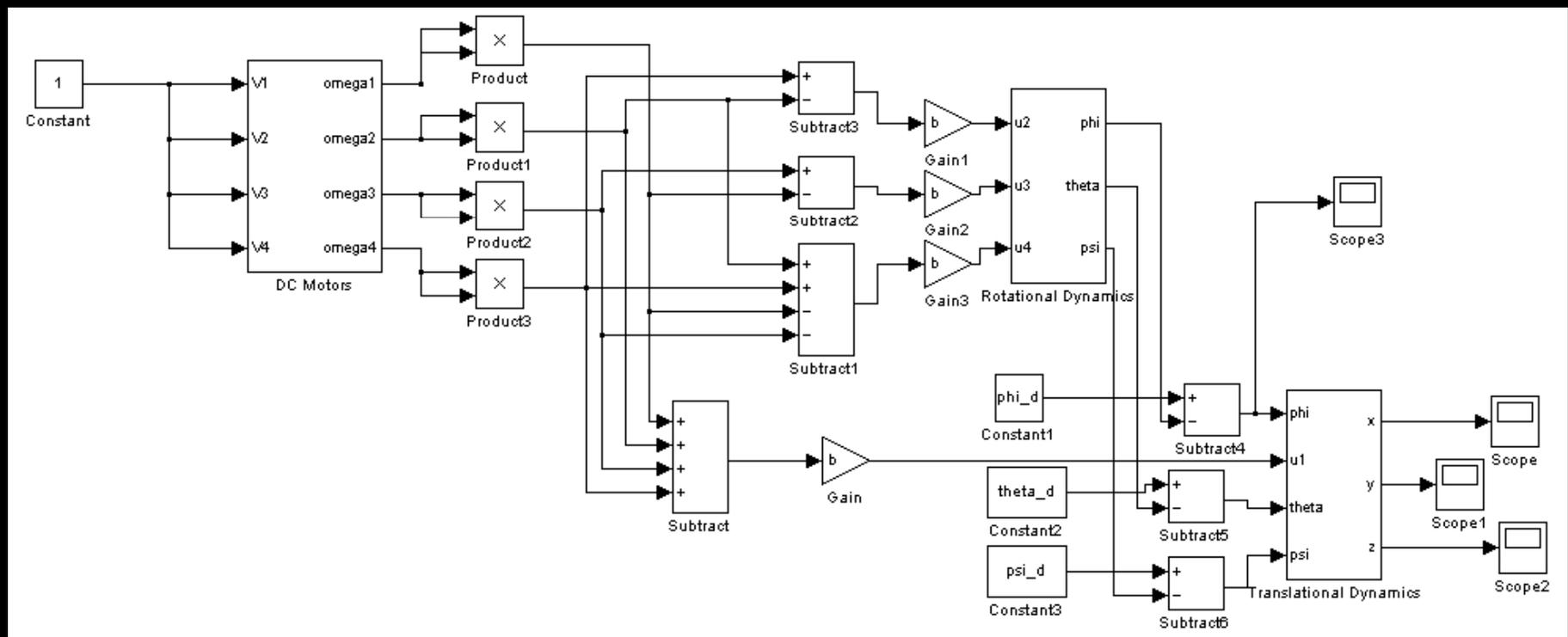
# SIMULATION (CON'T)

## Rotational Dynamics

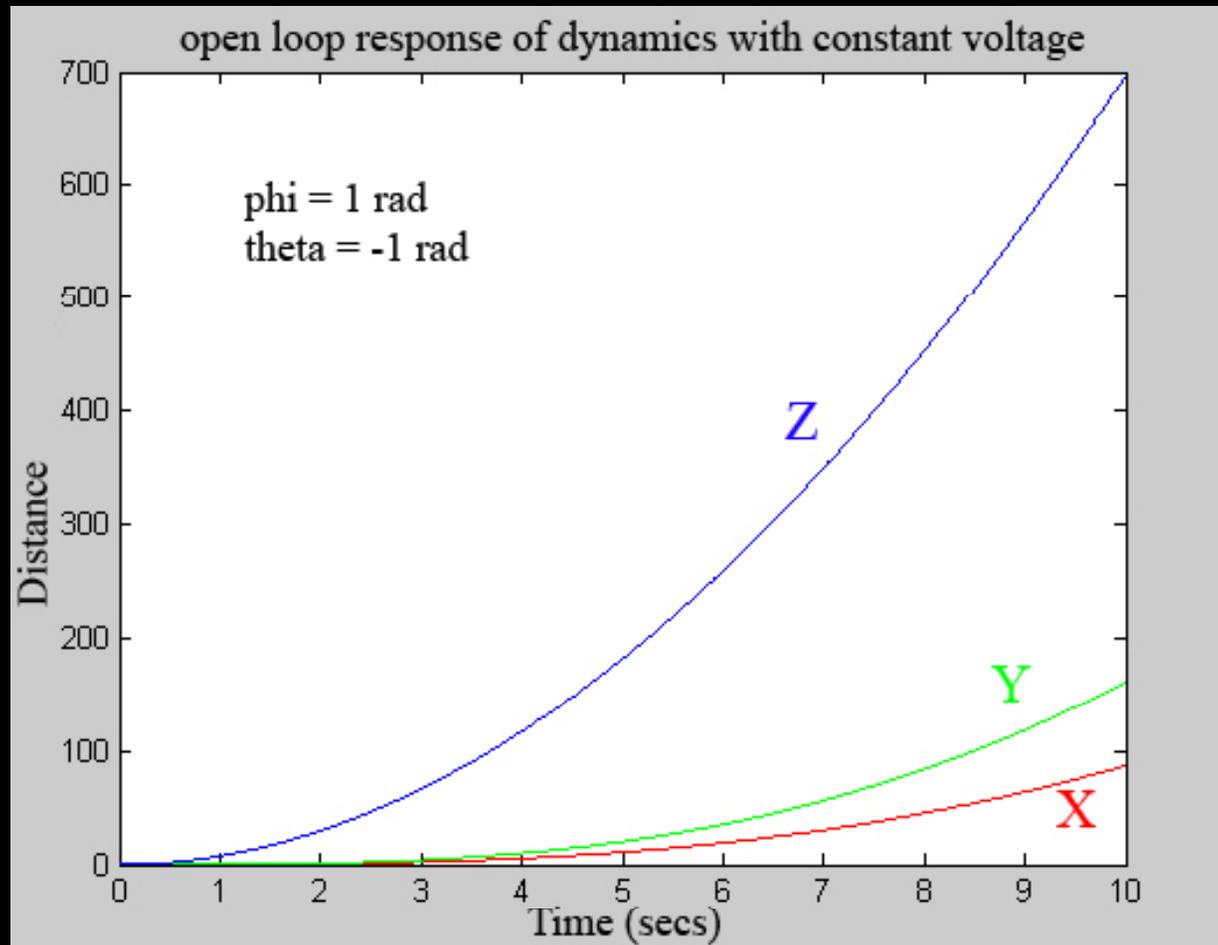


# SIMULATION (CON'T)

- Now the whole system can be compiled
- DC motors are simulated to real as possible
- Desired angles are implemented to induce translation along x and y axis



# OPEN LOOP



# VALUES USED FOR SIMULATION

VARIABLE	VALUE
Jz	0.244 kg*m <sup>2</sup>
Jx	0.122 kg*m <sup>2</sup>
Jy	0.122 kg*m <sup>2</sup>
phi_d	-1 rad/sec
theta_d	1 rad/sec
psi_d	0
length	0.3 m
mass	0.651 kg
b	2
gravity	9.8 m/sec <sup>2</sup>

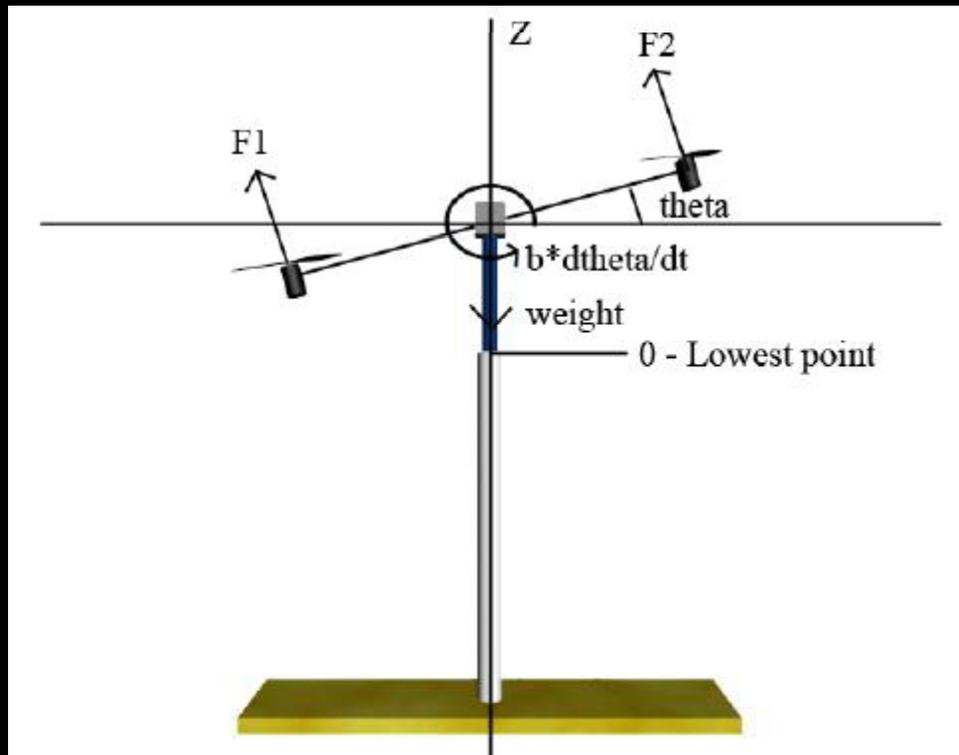
# 2D DESIGN PROTOTYPE



- 2 Degrees of freedom
- Will assist in feedback control implementation
- Uses final design hardware

# SIMULATION

## 2D Design and Dynamics



Translational

$$\sum F_z = m a_z = m \ddot{Z}$$

$$F_1 \cos\theta + F_2 \cos\theta - mg - c \dot{Z} = m \ddot{Z}$$

$$T = (F_2 + F_1) \cos\theta$$

$$m \ddot{Z} = T - c \dot{Z} - mg$$

Rotational

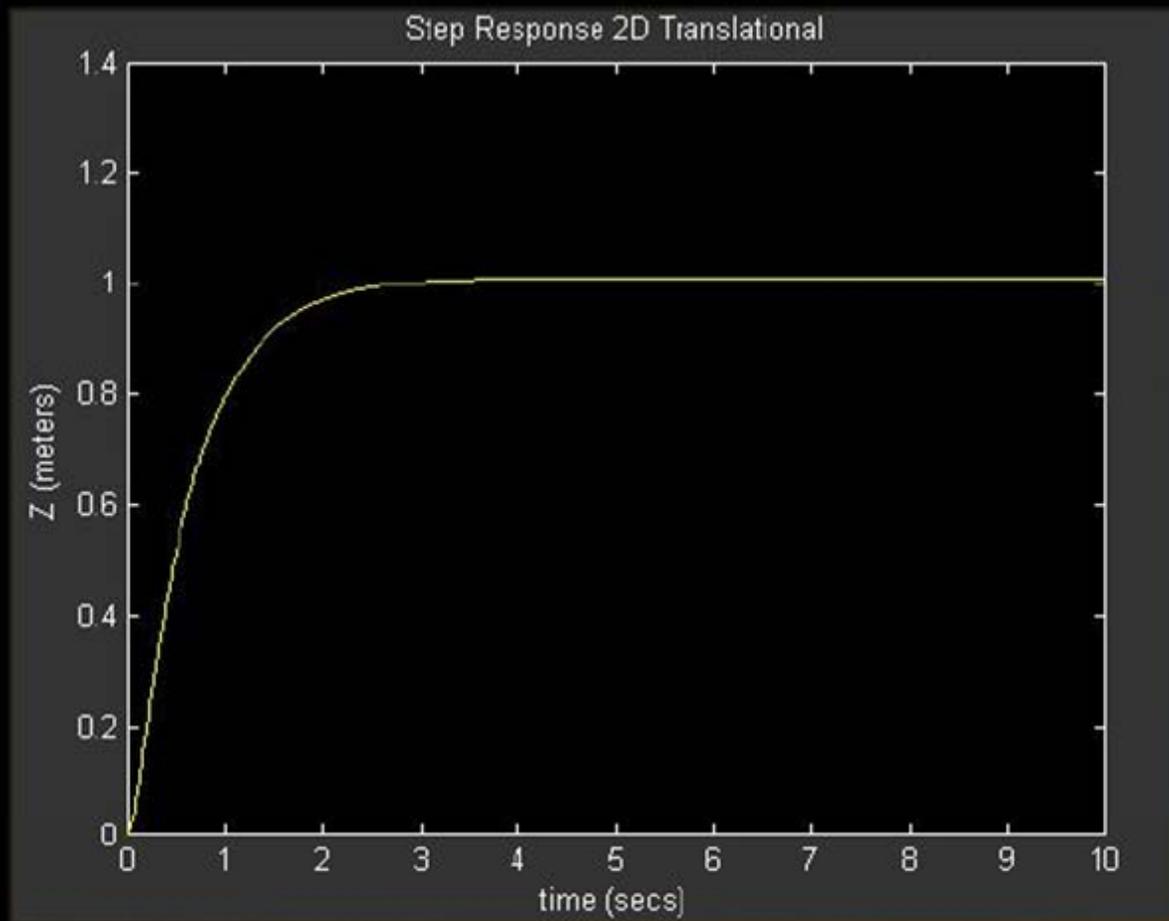
$$\sum \tau = I\alpha = I\ddot{\theta}$$

$$F_2 l - F_1 l - b\dot{\theta} = I\ddot{\theta}$$

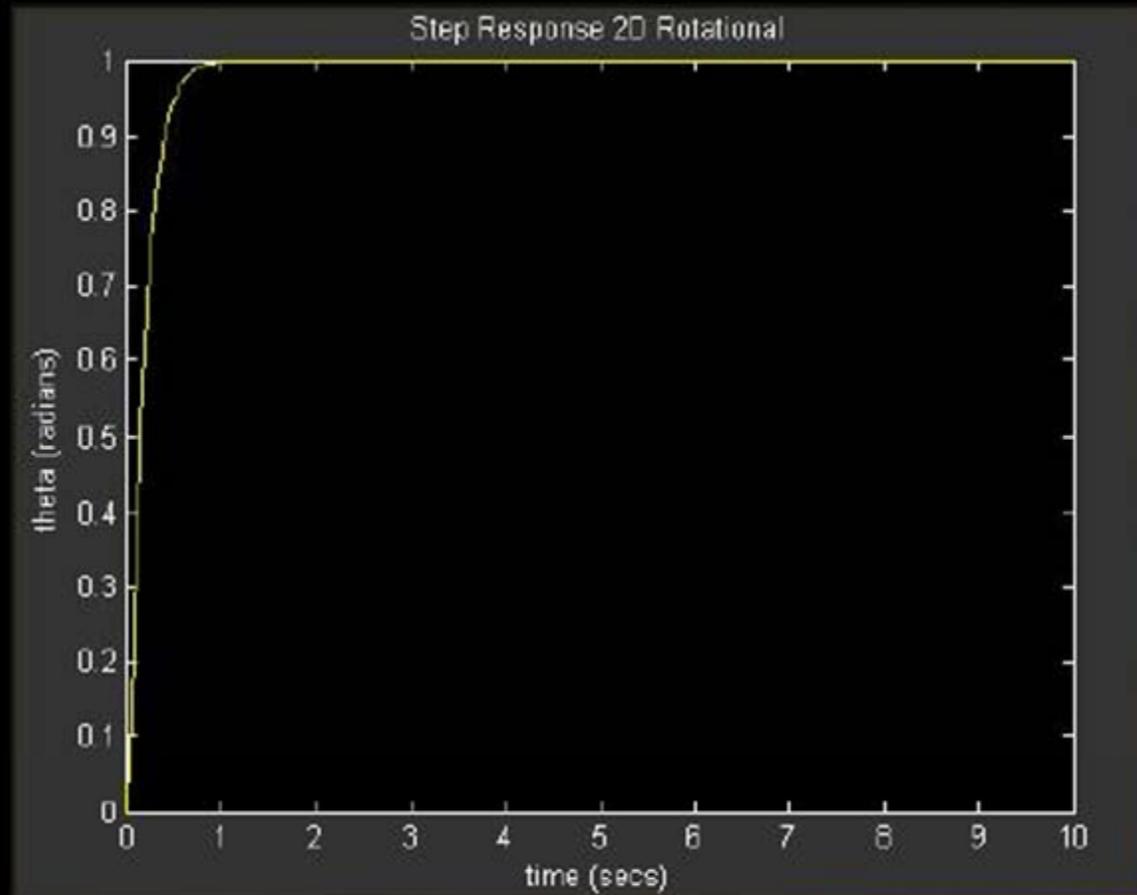
$$R = F_2 - F_1$$

$$I\ddot{\theta} = Rl - b\dot{\theta}$$

# Translational



# Rotational



# COMPONENT SELECTION DECISION

- Microcontroller
  - With the development board; small in design.
  - Enough ports to accept all signals we have.
  - Easy to program.
  - Produce PWM signals

# CONTROLLER OPTIONS

$\mu$ C	Active Power	CPU speed	PWM	RAM	Flash	I/O
TI MSP430	$\sim 220 \mu\text{A}$	16 MHz	1-ch	128 B	2K	10 GPIO 1-ch ADC
Coridium ARMmite	$\sim 50 \text{ mA}$	60 MHz	8-ch	8 K	32K	32 GPIO 8-ch ADC UART
Microchip 18F4550	$\sim 11 \mu\text{A}$	48 MHz	2-ch	1 Kb	16K	35 GPIO 13-ch ADC



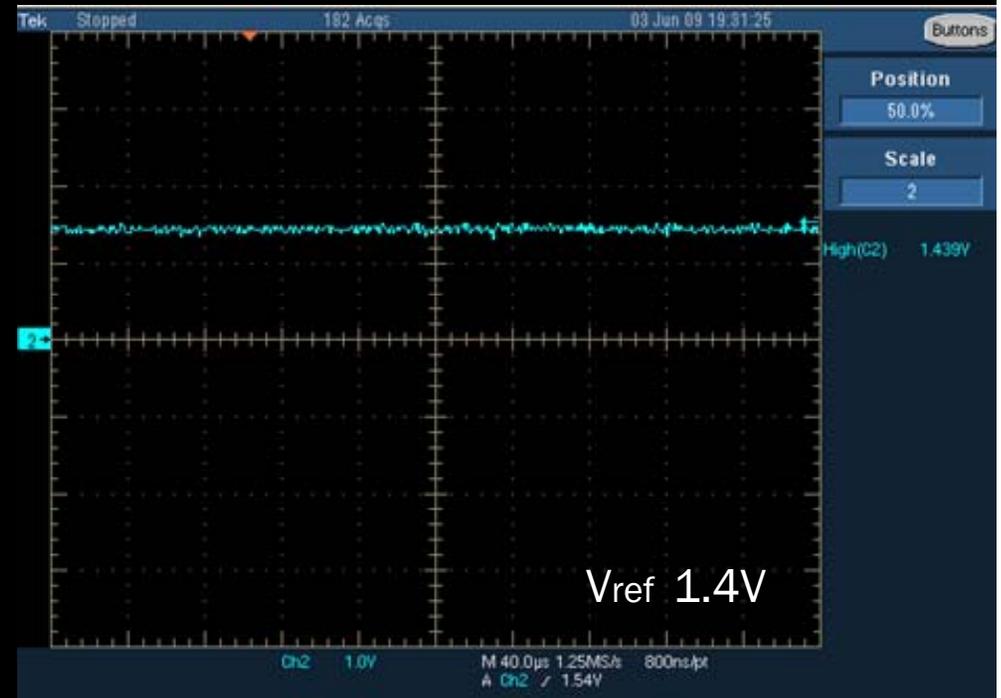
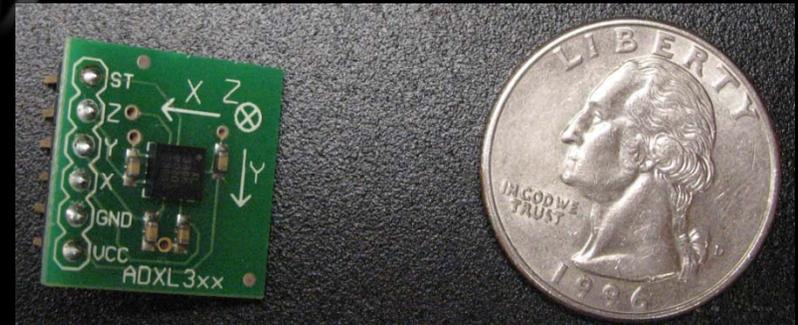
# COMPONENT SELECTION DECISION

- Tilt Sensing
- Low Power Consumption
- Small and Lightweight
- Good Sensitivity

Manufacturer	Part Number	# of Axes	Sensitivity	Current Draw	Price
Analog Devices	ADXL320	2	+/- 5g	480uA	\$29.95
Analog Devices	ADXL330	3	+/- 3g	320uA	\$34.95
STMicroelectronics	LIS3LV02DQ	3	+/-2 or 6	400uA	\$43.95

# COMPONENT SELECTION

- Analog Devices  
ADXL330  
Accelerometer
  - Triple axis
  - 320  $\mu\text{A}$  @ 3.3V  
operation
  - Analog output



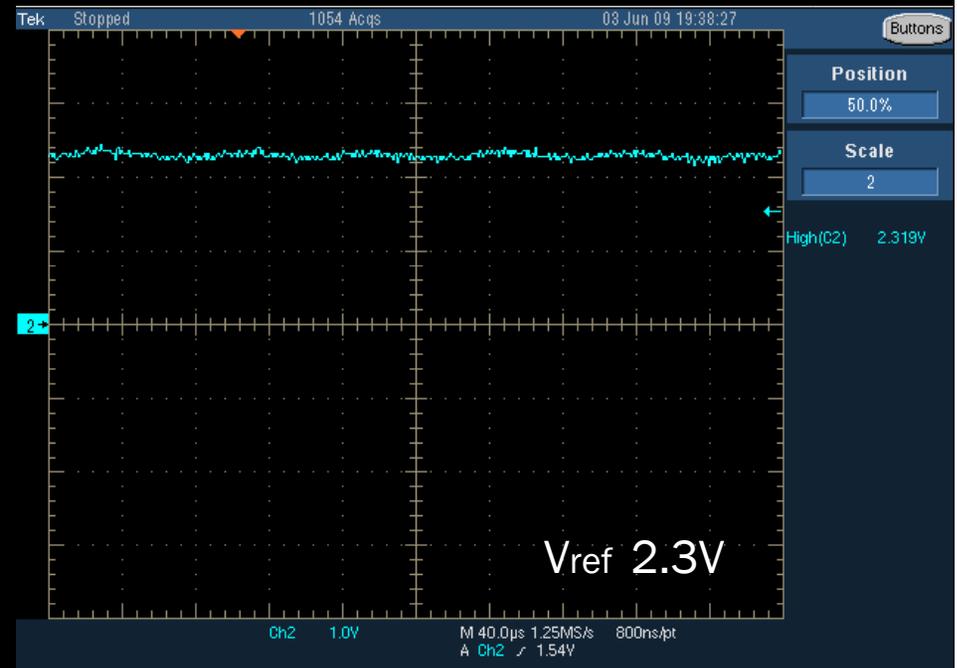
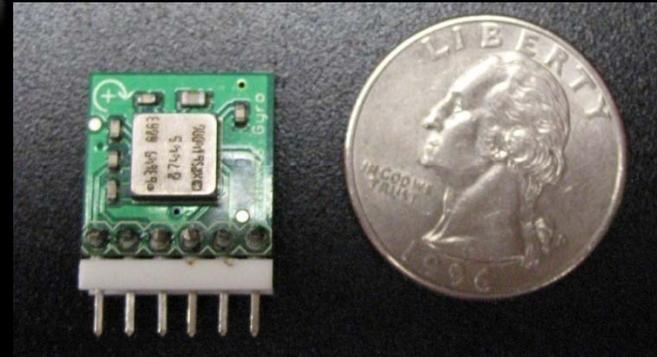
# COMPONENT SELECTION DECISION

- Rotation Sensing
- Low Power Consumption
- Small and Lightweight
- Good Sensitivity

Manufacturer	Part Number	# of Axes	Range	Current Draw	Price
Analog Devices	ADXRS614	1	50° / sec.	5mA	\$64.95
STMicroelectronics	LISY300AL300	1	300° / sec.	4.8mA	\$29.95
Invensense	IDG1215	2	67° / sec	9.5mA	\$74.95

# COMPONENT SELECTION

- Analog Devices  
ADXRS614 Gyroscope
  - 50 °/sec rate sensitivity
  - Single axis
  - 5 mA @ 5V operation
  - Analog output



# COMPONENT SELECTION DECISION

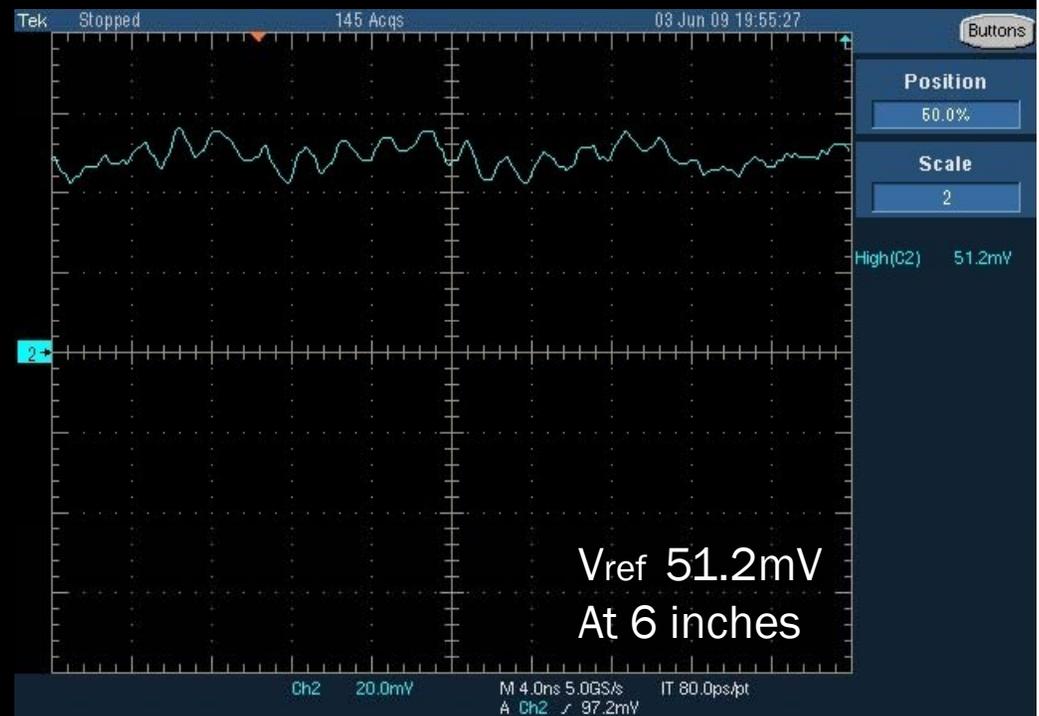
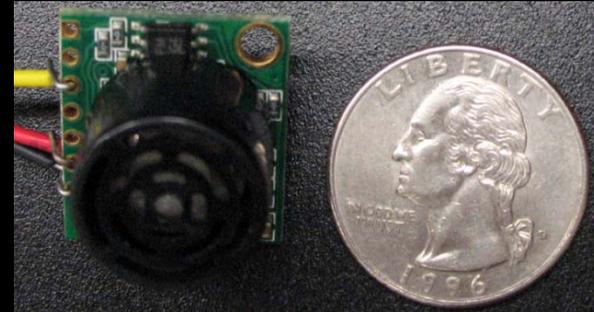
- Height Sensing
- Obstacle Avoidance
- Low Power Consumption
- Small and Lightweight
- Good Sensitivity

Manufacturer	Part Number	Technology	Range	Current Draw	Price
Maxbotix	EZ0	Ultrasonic	0" – 256"	2 mA	\$27.95
Sharp	GP2Y0A02YK0F	Infrared	7" – 60"	50 mA	\$15.95

# COMPONENT SELECTION

- Ultrasonic Range Finder - Maxbotix LV-EZO

- Multiple signal output
- 2 mA @ 5V operation
- 6 – 254in range with 1 inch resolution
- Refresh rate every 49 milliseconds



# FLIGHT HARDWARE

- Frame
  - Carbon Fiber Design
  - Includes:
    - Motor mounts
    - Propellers
    - Main gears
    - Brushed Motors
  - Set of clockwise and counterclockwise blades



# SPEED CONTROLLERS

- Castle Creations Thunderbird - 9
- 9A, 15V Max.
- Weight: 8g
- Auto Motor Cut-off
- Fully Programmable



# MOTORS/PROPS

- Feigao Brushless Motor 1308441S
- 6A, Inrunner
- 2283 RPM/V
- Weight: 43.4g
  
- 12.375" Dia. Propellers
- 2-Piece Black Nylon
- Folding Design

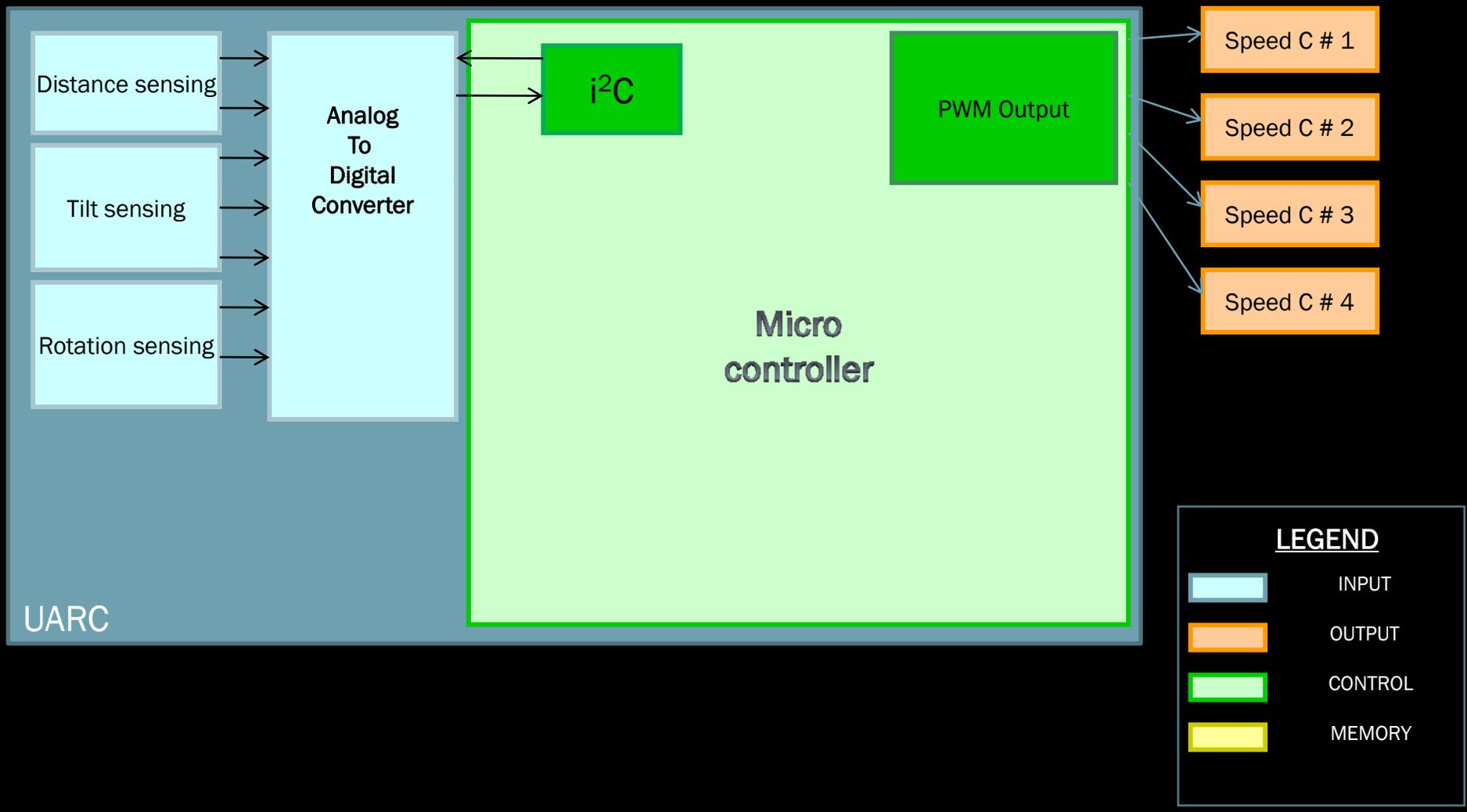


# POWER SUPPLY

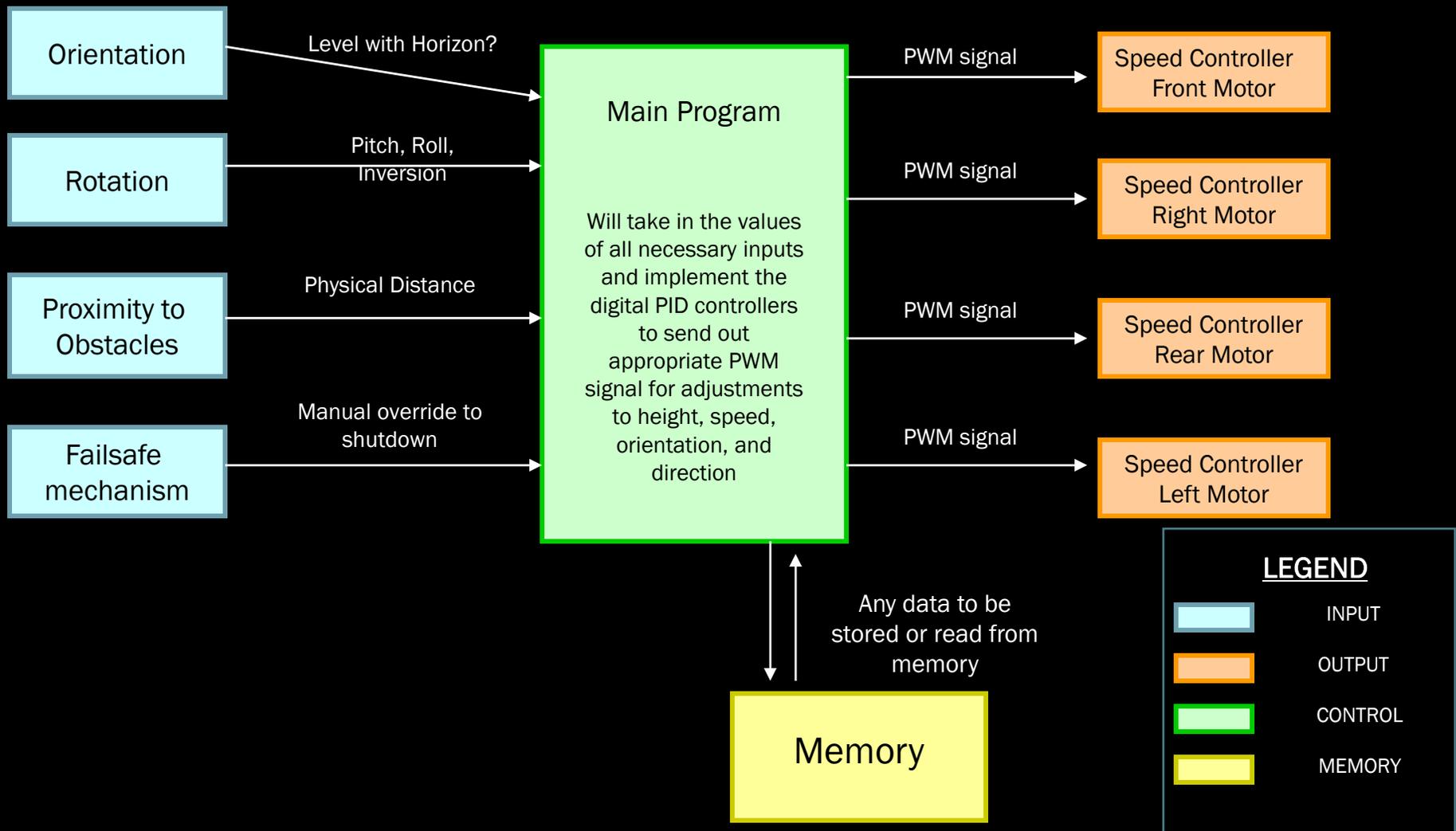
- Thunder Power Lithium Polymer (Li-po)
- 7.4V, 2 cell, 1900mAh
- Max. Continuous Current: 38 A
- Max Burst Current: 76A
- Weight: 95g
- Rechargeable



# HARDWARE BLOCK DIAGRAM

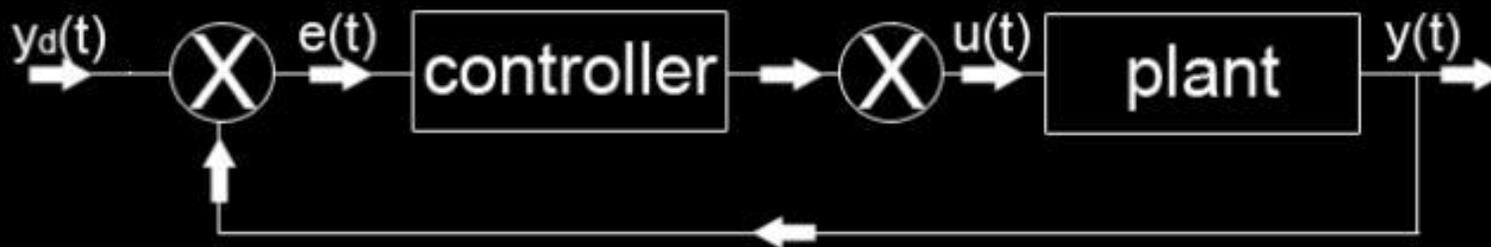


# SOFTWARE BLOCK DIAGRAM

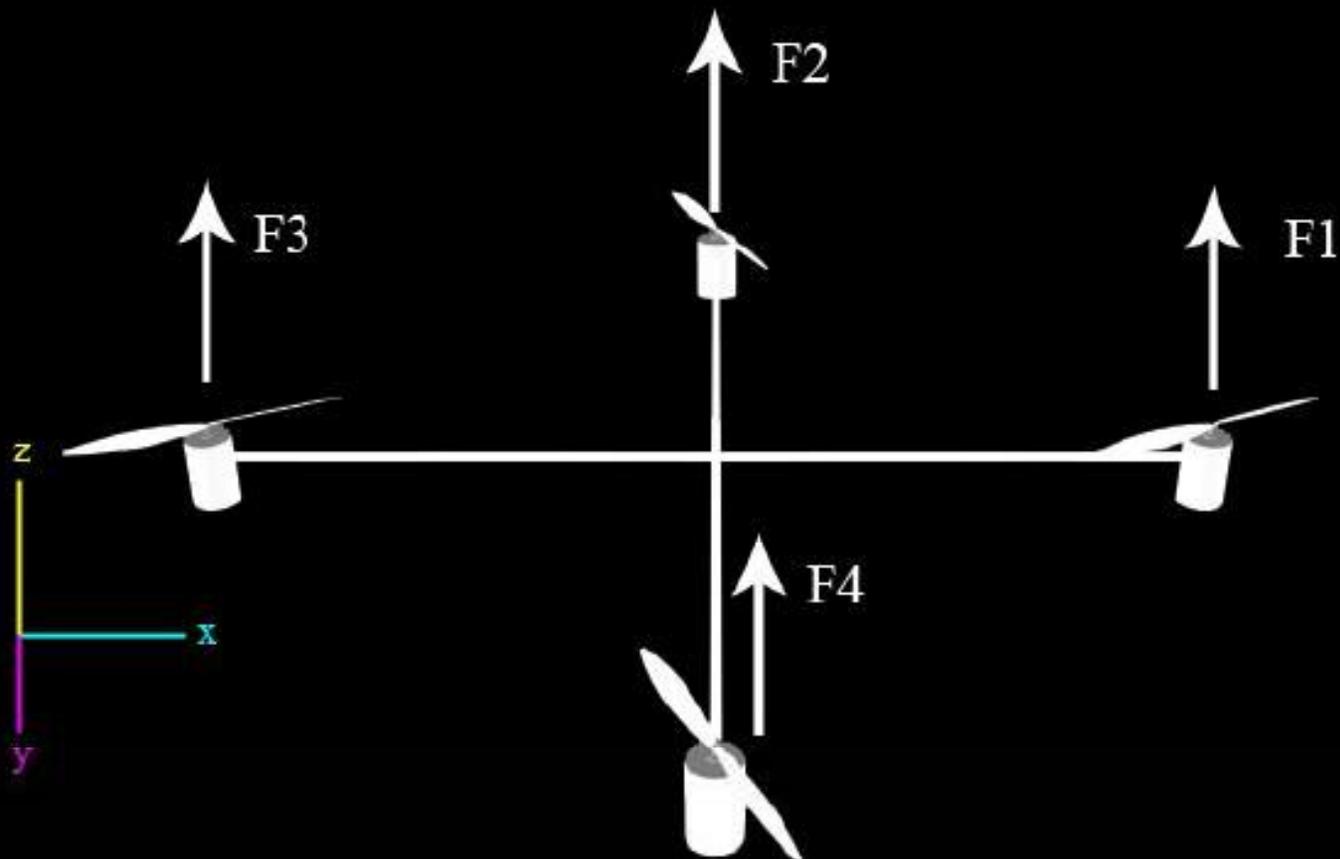


# CONTROL SYSTEM

- There are many theoretical and implemented control techniques for the quad rotor
- A couple methods considered
  - Linear Quadratic Regulator (LQR)
  - Proportional Integral Derivative (PID)
- Advantages
  - LQR is an optimal controller which minimizes error input to the plant, among stability
  - PID control is simple, can be analog or digital, and it's a reliable classic
- Disadvantages
  - LQR requires a precise linear dynamical model of the plant, if not precise system will become unstable
  - A digital PID requires limiting conditions to prevent integral overflow and derivative spikes and must have a constant sampling time
- UARC will implement basic digital PID controllers for stability



# CONTROL SYSTEM

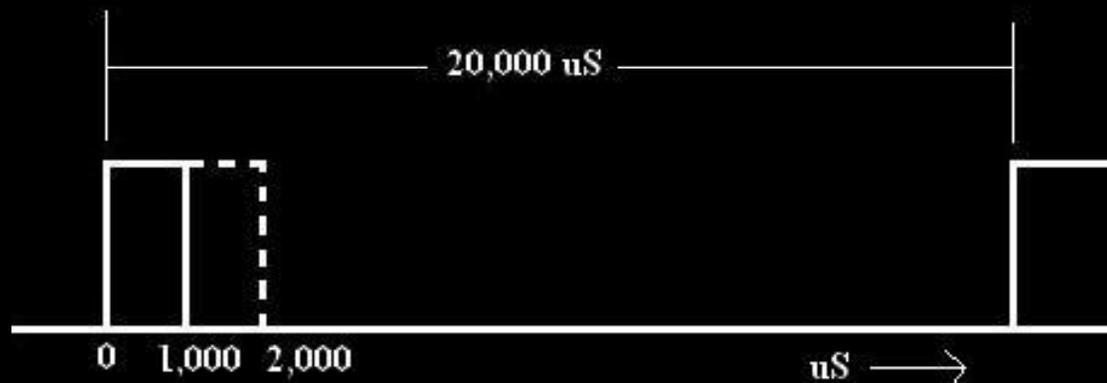


Increase	Decrease	Direction
$F1$	$F3$	$-z$
$F3$	$F1$	$+z$
$F4$	$F2$	$-y$
$F2$	$F4$	$+y$

# CONTROL SPECIFICATIONS

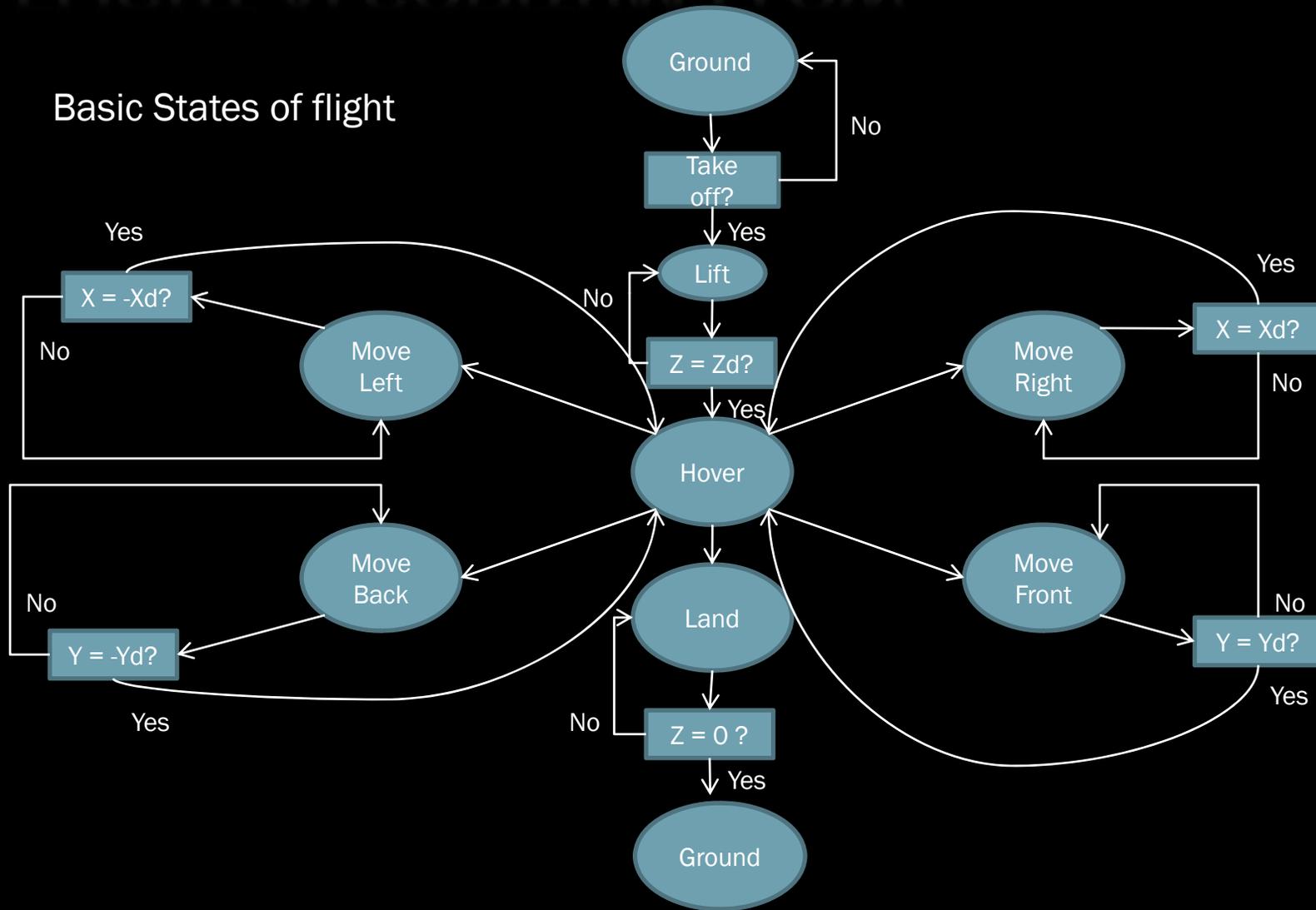
- Digital implementation in C
- Limit error overflow
- Sampling frequency at 50 Hz, which is 20 milliseconds
- Limit PWM duty cycle between 1000 and 2000 uS
- Establish PWM change in duty cycle through pidUpdate
  - $\text{pidOut} = \text{pid\_gyro} + \text{pid\_accel}$
  - $\text{PWM\_M1} = \text{MINDUTY} + \text{DUTYCYCLE} + \text{pidOut}$
  - $\text{PWM\_M2} = \text{MINDUTY} + \text{DUTYCYCLE} - \text{pidOut}$

STATE	DUTYCYCLE (uS)
IDLE	0
LIFT	535
HOVER	500
LAND	450



# FLIGHT ALGORITHM FLOW

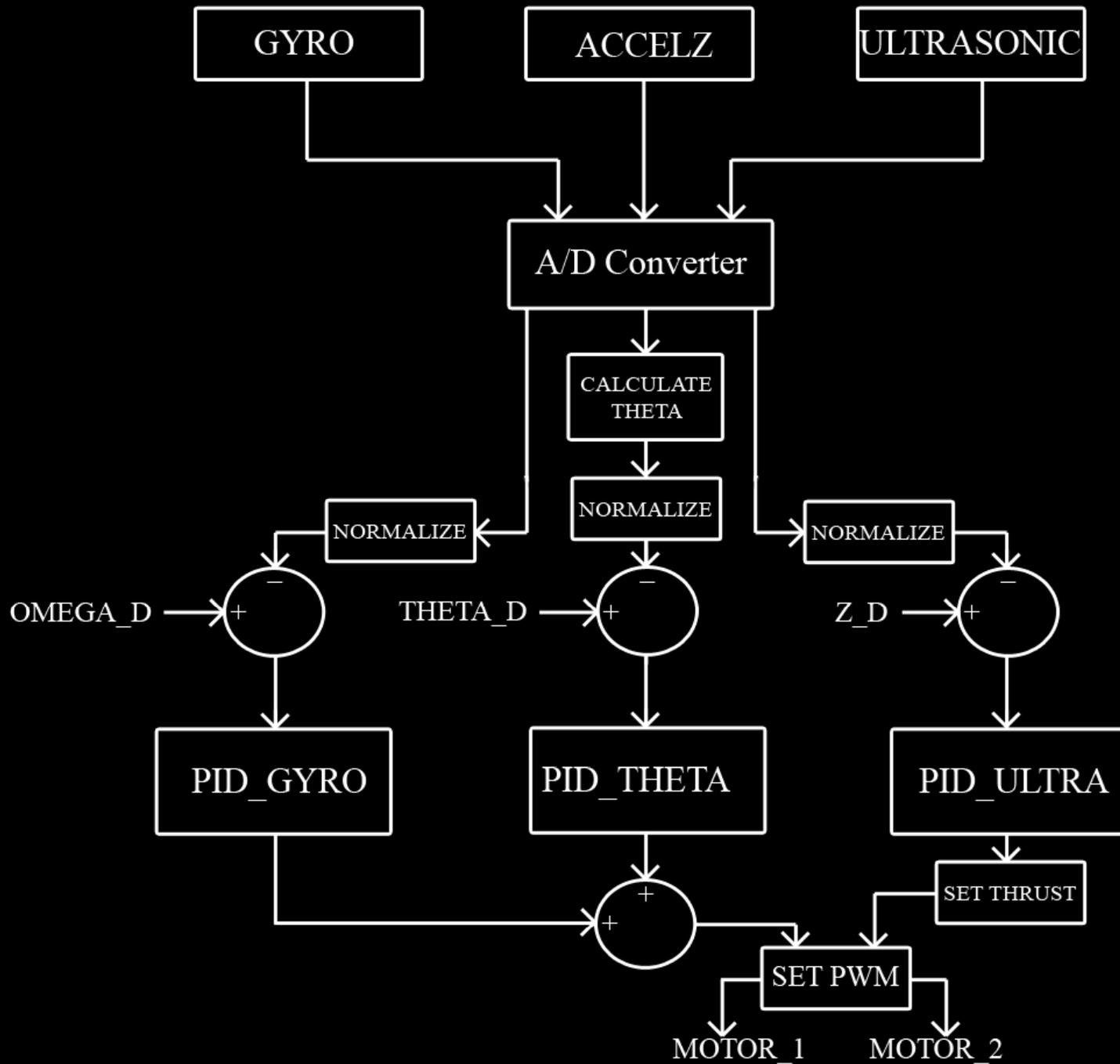
Basic States of flight



# GROUP DUTIES

- Jeremy – Systems implementation, PCB, and testing.
- Clint – Control systems and Simulation
- Edwin – Sensors and software coding





# FUNCTIONS

---

```
init_coridium(); // starts up the coridium microprocessor
```

```
void init_sensors ( ); // initializes sensors
```

```
void init_pid (double p_gain, double i_gain, double d_gain, pidData *pid); // initialize pid gains and overflow limits
```

```
double pid_Controller (double setPoint, double processValue, pidData *pid_st); // update controlled plant input signal
```

```
Void setPWM(int pidOut); // set pwm for each individual motor depending on pid output
```

```
Int getGyro_x ( ); // get corresponding sensors signals from max127 ADC
```

```
Int getGyro_y ( );
```

```
Int getAccel_x ( ); // get accelerometer values to determine phi and x acceleration
```

```
Int getAccel_y ( ); // get accelerometer values to determine theta and y acceleration
```

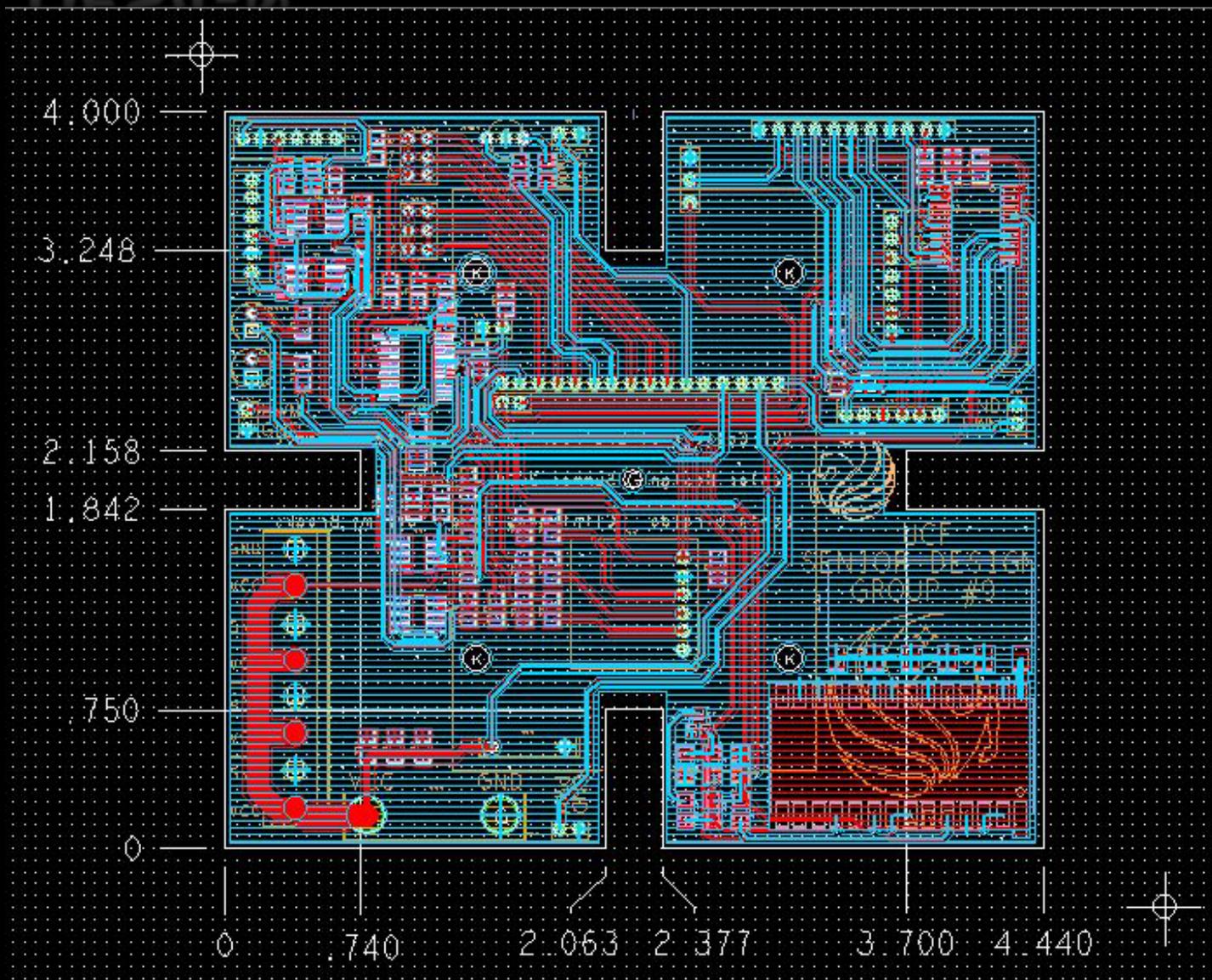
```
Int getAccel_z ( ); // get accelerometer values to determine z acceleration
```

```
Int getUltraTop ( ); // get top ultrasonic sensor reading
```

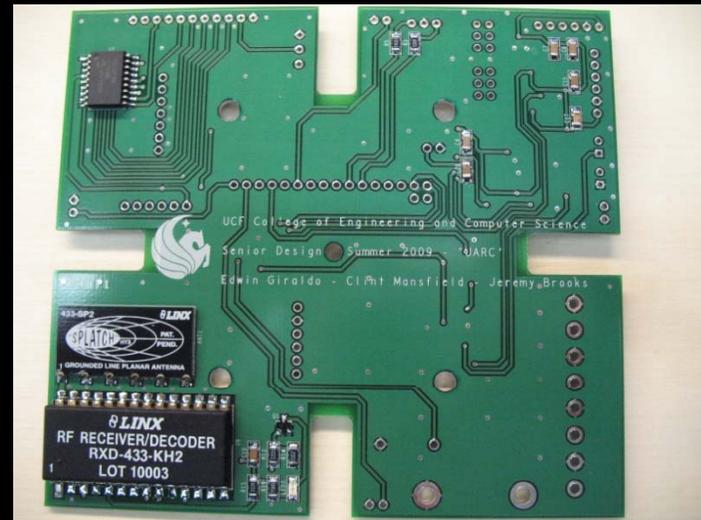
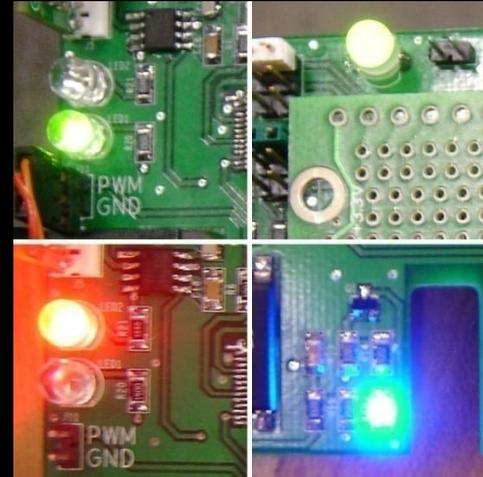
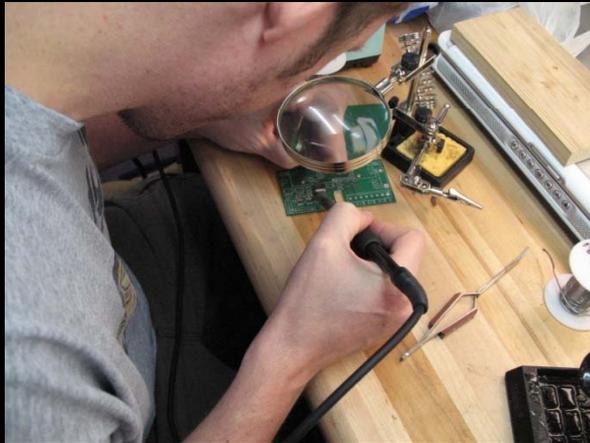
```
Int getUltraBot ( ); // get bottom ultrasonic sensor reading
```



# PCB DESIGN

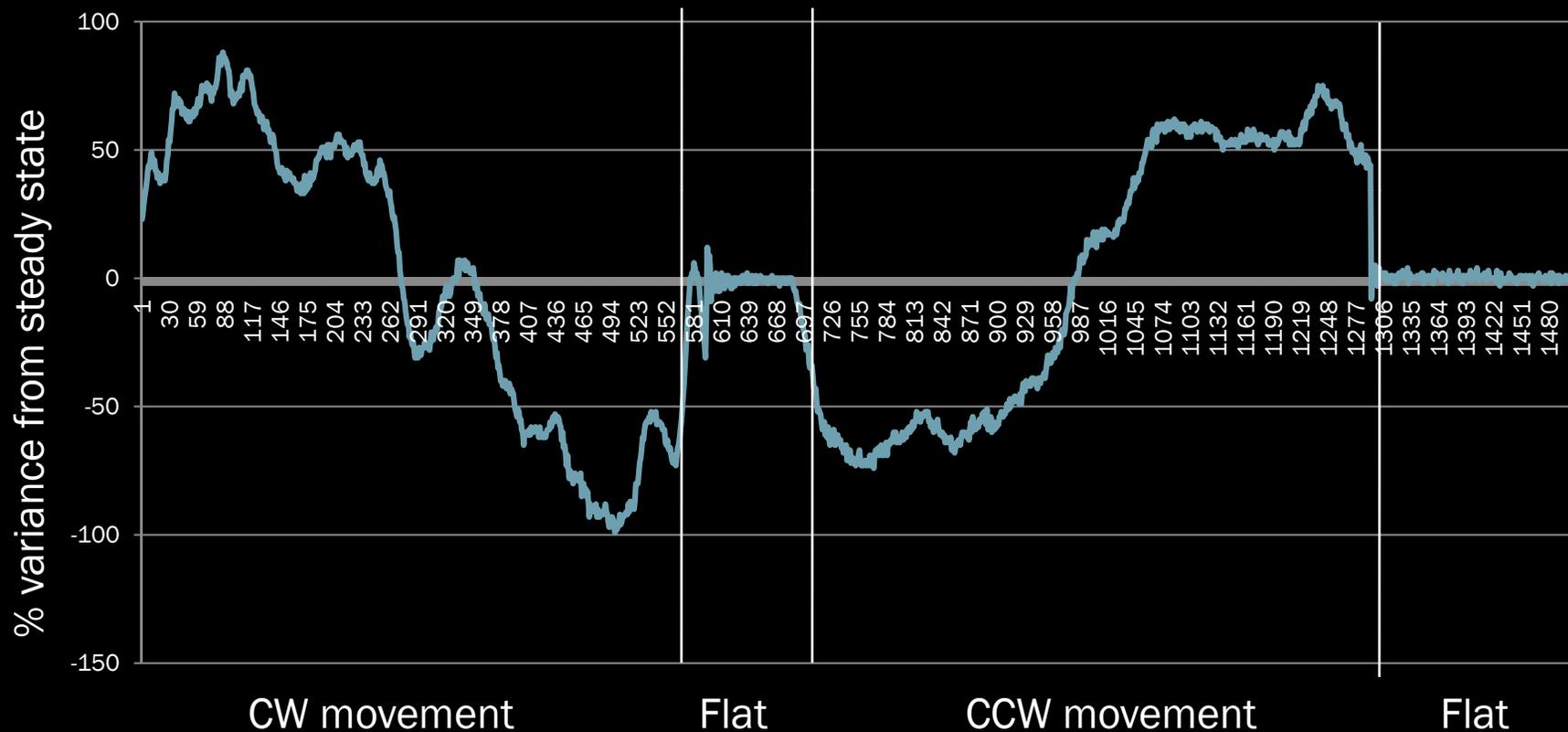


# PCB FUNCTIONALITY



# TESTING

Digital output from normalized Gyro sensor



Note: Samples @ 250 Hz through a first order low-pass filter with cutoff at 1KHz.

# TESTING

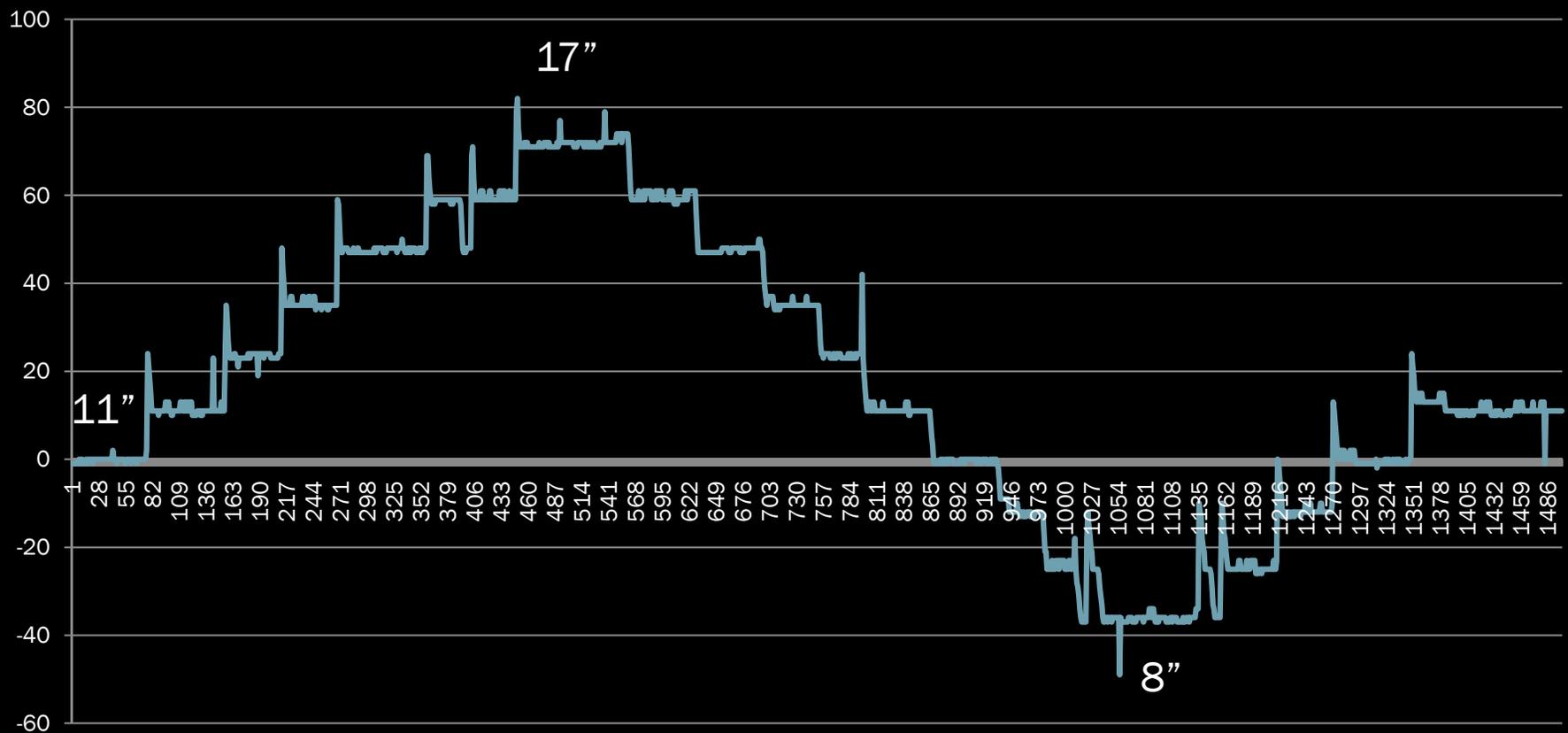
Digital output from normalized accelerometer sensor



Note: Samples @ 250 Hz through a first order low-pass filter with cutoff at 1KHz.

# TESTING

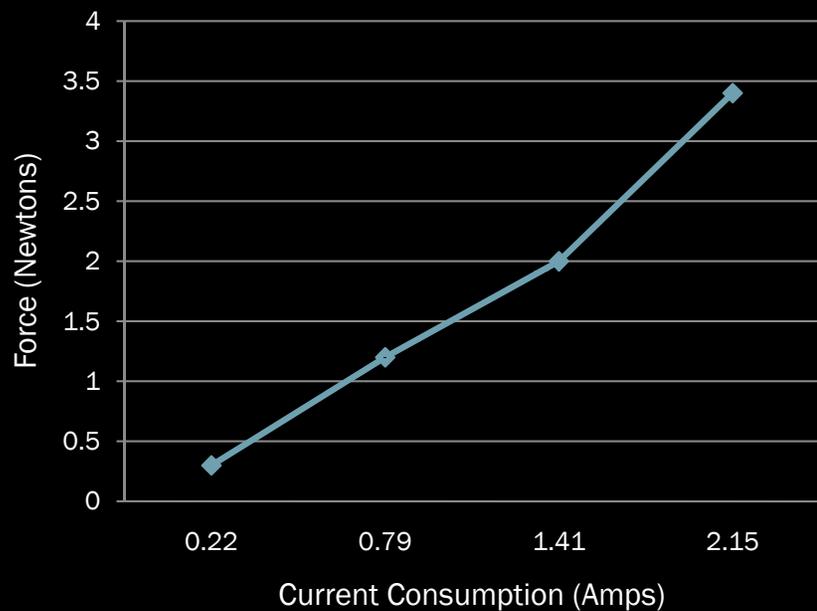
Ultrasonic normalized distance sensor



Note: Samples @ 250 Hz through a first order low-pass filter with cutoff at 1KHz.

# TESTING

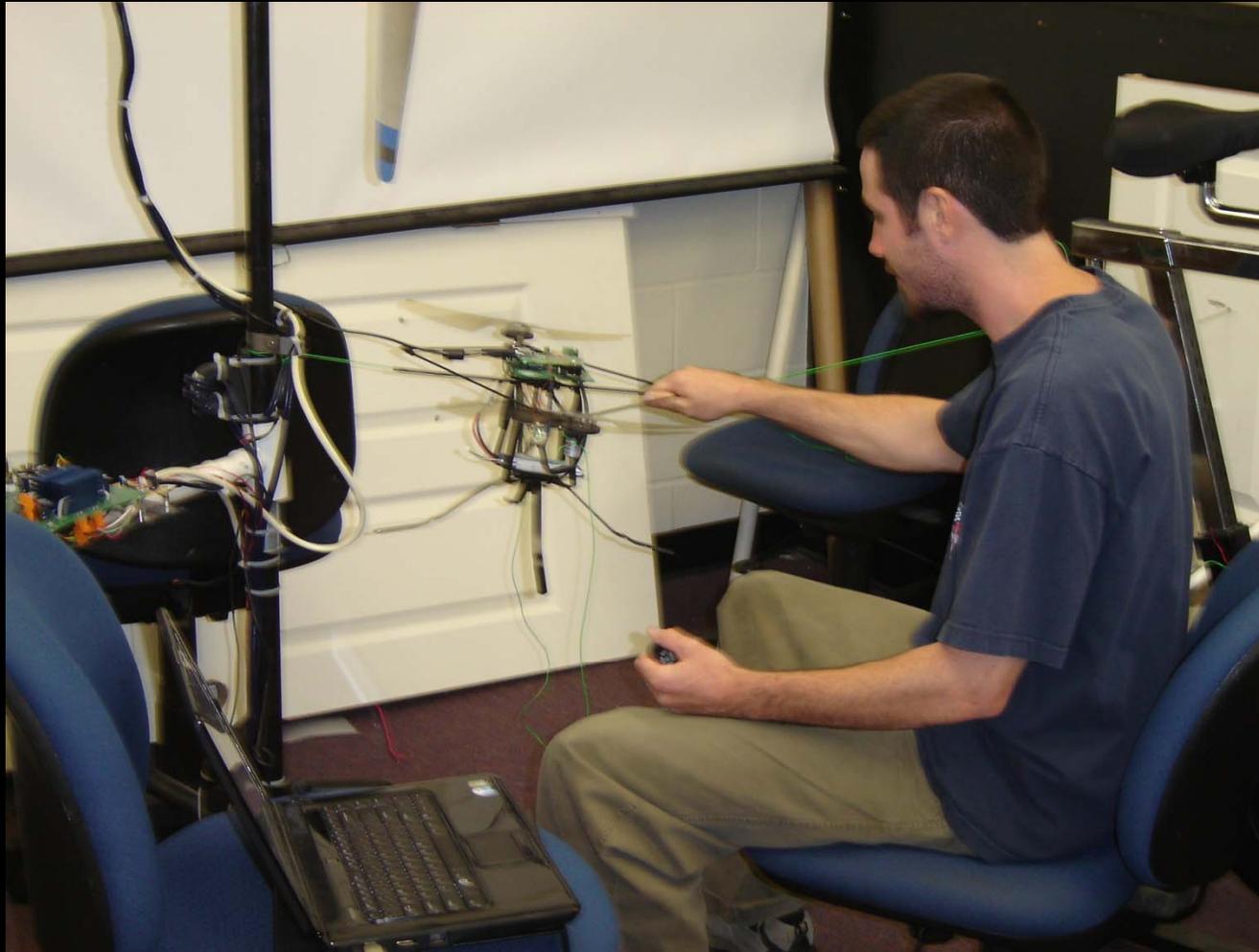
## Thrust Vs. Amperage



Amps	Duty Cycle	Force
0.22 A	4.23%	0.067 lbs
0.79 A	4.73%	0.269 lbs
1.41 A	5.27%	0.449 lbs
<b>2.15 A</b>	<b>5.65%</b>	<b>0.764 lbs</b>



# 2D TESTING



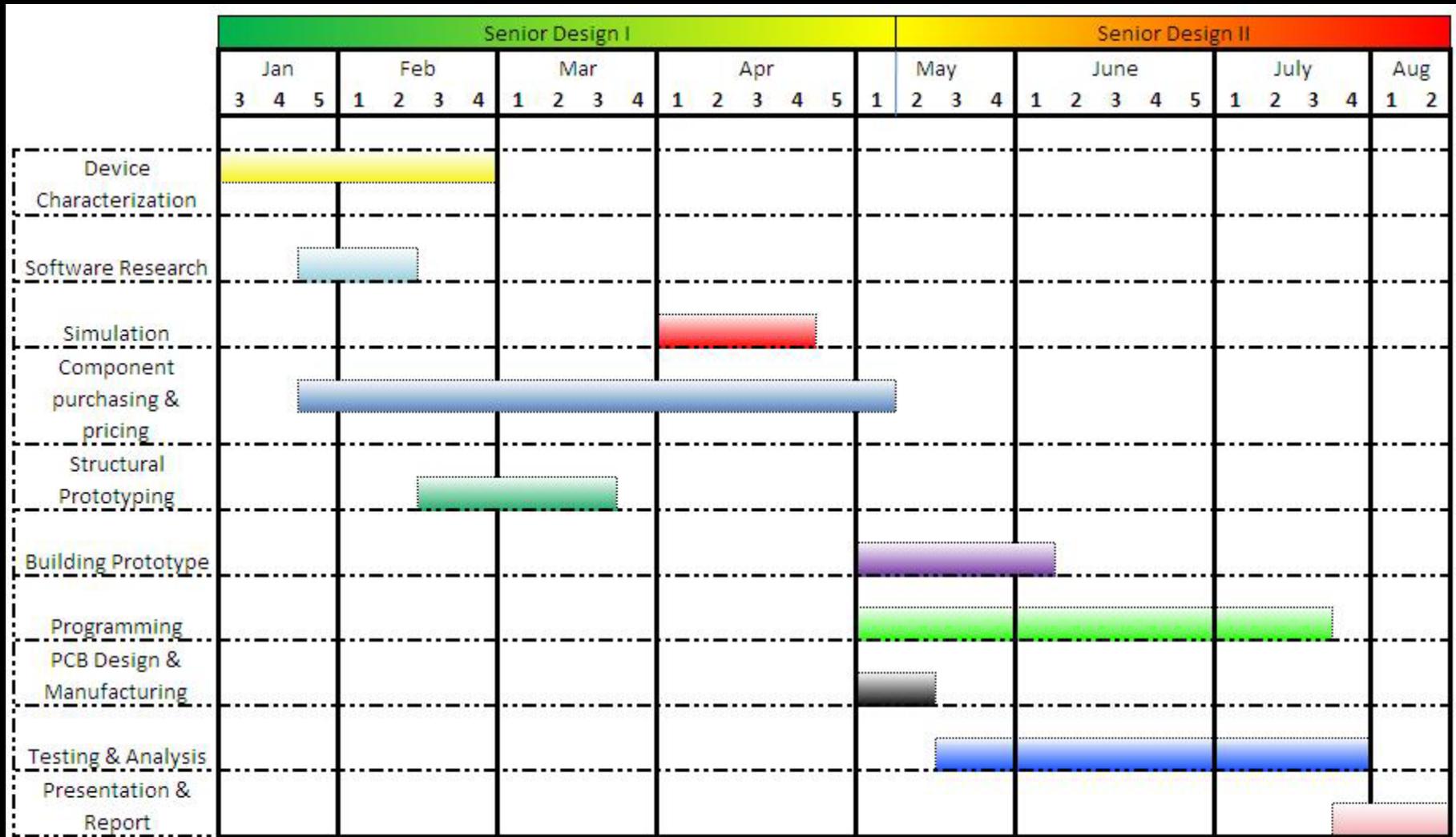
# 3D ASSEMBLY



# PROJECT BUDGET

Item	Vendor	Cost	Spent
Draganflyer airframe	eBay	\$199.95	\$85.90
TI MSP430 Microcontroller	Digi-key	\$59.06	\$24.53
Brushless Motors (X5); pinion gears; adapter ring (X4); bracing kit	BP Hobbies	\$217.10	\$217.10
Speed Controllers (X4)	Graves RC	\$117.11	\$117.11
Gryo Breakout Board (X2)	Sparkfun	\$129.90	\$129.90
Triple Axis Accelerometer	Sparkfun	\$34.95	\$34.95
Coridium ARMmite Microcontroller	Sparkfun	\$59.08	\$59.08
Ultrasonic Range Finder - Maxbotix LV-EZO (X2)	SuperDroid	\$61.60	\$61.60
2-D Assembly	Lowes	\$6.32	\$6.32
Max127 A/D converter I <sup>2</sup> C compliant (X4)	Maxim IC	\$121.74	\$0.00
Over/Under Dual Voltage detector (X2)	Intersil	\$7.20	\$0.00
Lipo Battery Charger	On hand	\$21.95	\$0.00
7.4V Rechargeable Battery 1900 mAH (X2)	eBay	\$74.36	\$74.36
PCB	Advanced Circuits	\$62.99	\$62.99
Passive Components	Various	\$176.39	\$141.12
Servo Testers	E Sky	\$23.12	\$23.12
		\$1,372.82	\$1,038.08

# PROJECT MILESTONE



# OBSTACLES ENCOUNTERED

---

- Burnt ports
- Faulty motor
- Updated software (vista compatibility)
- PWM port not sequential on I/O pins.
- RF I/O port development led.
- Blades limit thrust.
- Jumping of I/O ports.

# LESSONS LEARNED

---

- Voltage supply crucial for performance
  - More consistent sensor output values.
  - More thrust.
- 5V will burn an led.
- Test all equipment promptly after purchase.
- Blades can hurt.