# USB Virtual Reality HID

by

Weston Taylor and Chris Budzynski

Advisor:  Dr. Malinowski

# Agenda

- Project Summary
- Recap
- Schedule
- Completed Work
- Next Steps
- Questions

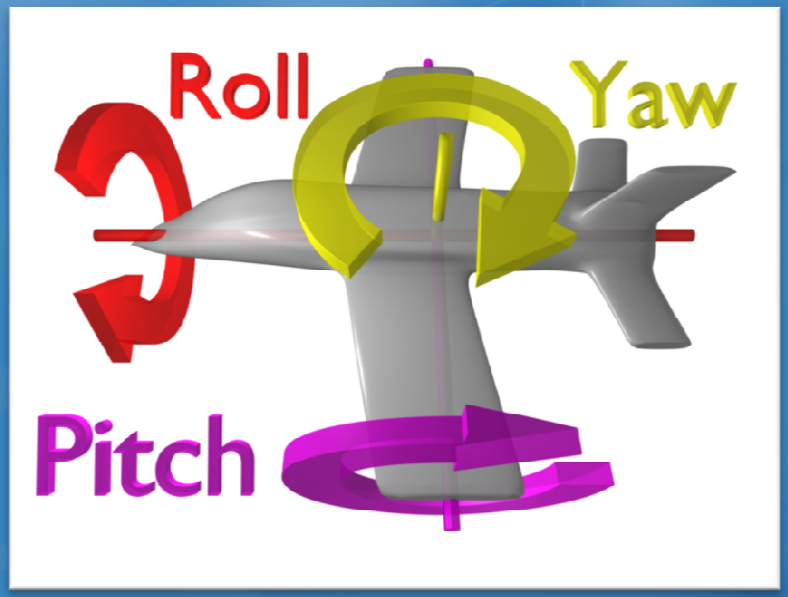
# Project Summary

- **Purpose:** To create a USB (Universal Serial Bus) Virtual Reality HID (Human Interface Device). This USB HID will interface with personal computers and their programs by emulating a USB gamepad.

- **Overall Goal:** To translate user movements into on-screen actions to provide a more realistic interactive platform for PC games and other virtual environments.

# Recap

- Stationary INS (Inertial Navigation System)
  - No Linear Acceleration
  - Only Angular Acceleration (Pitch, Yaw)
  - Gyroscopes measure angular acceleration



# Recap

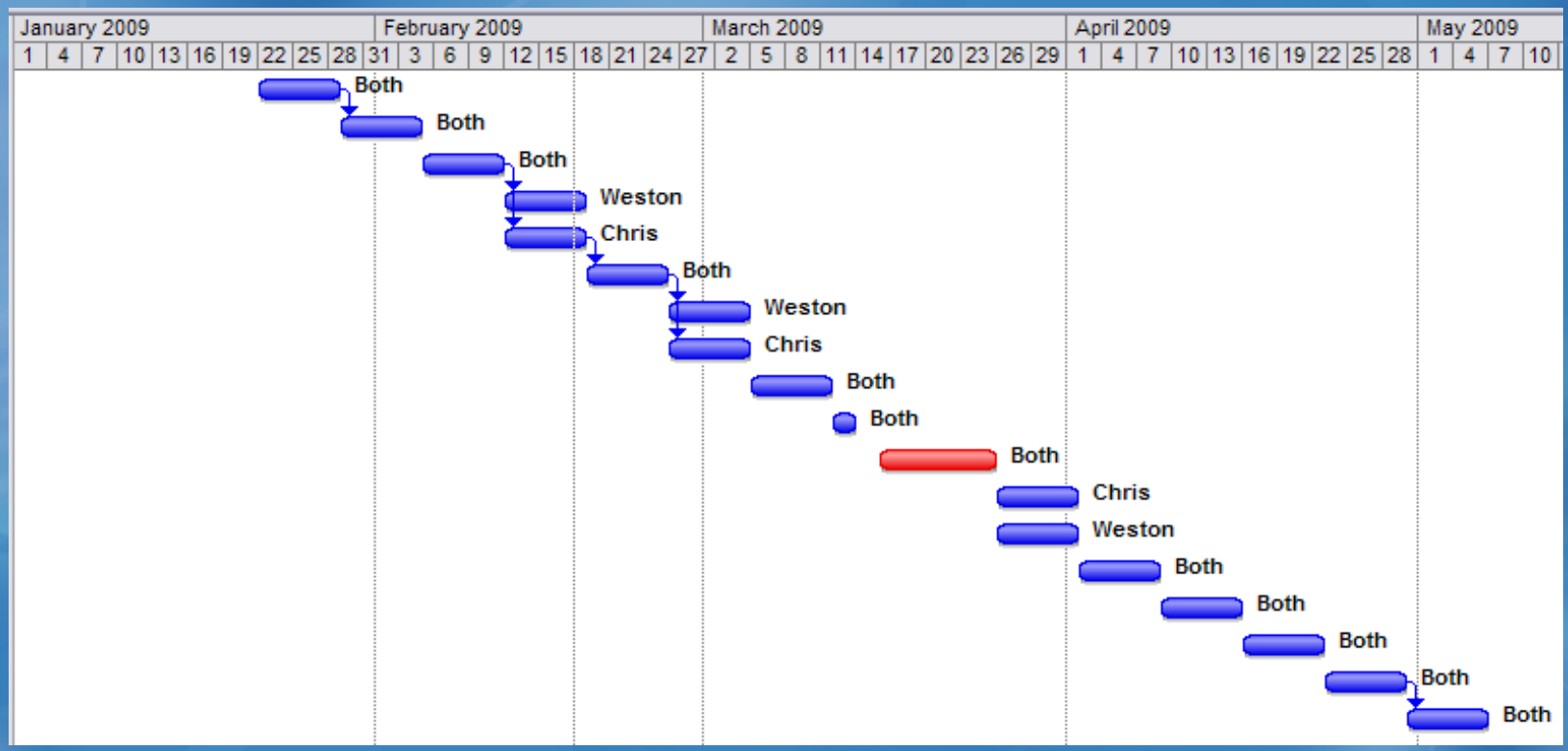
- Gyroscopes
  - Measure °/sec
  - 1 integration to get position (absolute angle)
- Integration Drift
  - Compounded Error
  - Pitch Drift Correction (accelerometer)
  - Yaw Drift Correction (electronic compass)

# Schedule

- Pair Programming

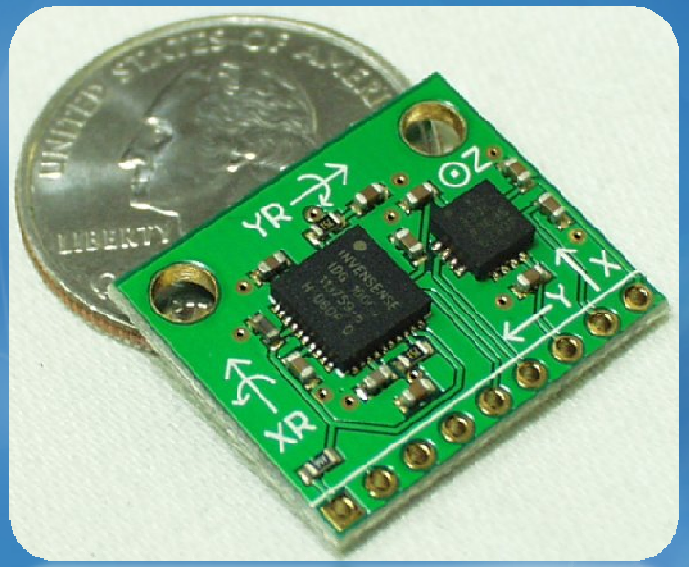
| | Task Name | Duration | Start | Finish | Predecessors | Resourc |
|---|---|---|---|---|---|---|
| 1 | USB Gamepad Conversion | 5 days? | Thu 1/22/09 | Wed 1/28/09 | | Both |
| 2 | USB Gamepad Conversion | 5 days? | Thu 1/29/09 | Wed 2/4/09 | 1 | Both |
| 3 | Sensor Interface / Setup | 5 days? | Thu 2/5/09 | Wed 2/11/09 | | Both |
| 4 | Sensor Coding / Testing | 5 days? | Thu 2/12/09 | Wed 2/18/09 | 3 | Weston |
| 5 | Gyro Drift Testing / Correction | 5 days? | Thu 2/12/09 | Wed 2/18/09 | 3 | Chris |
| 6 | Frame of Reference Equation/Algorithm | 5 days? | Thu 2/19/09 | Wed 2/25/09 | 5 | Both |
| 7 | Frame of Reference Code | 5 days? | Thu 2/26/09 | Wed 3/4/09 | 6 | Weston |
| 8 | Feedback / Drift Correction Algorithm / Code | 5 days? | Thu 2/26/09 | Wed 3/4/09 | 6 | Chris |
| 9 | FPU Co-processor | 5 days | Thu 3/5/09 | Wed 3/11/09 | | Both |
| 10 | USB Controller Coding | 2 days? | Thu 3/12/09 | Fri 3/13/09 | | Both |
| 11 | Spring Break (Catch Up) | 8 days? | Mon 3/16/09 | Wed 3/25/09 | | Both |
| 12 | Step Pad Hardware | 5 days? | Thu 3/26/09 | Wed 4/1/09 | | Chris |
| 13 | Step Pad Software | 5 days? | Thu 3/26/09 | Wed 4/1/09 | | Weston |
| 14 | Integration of Individual Subsystems | 5 days? | Thu 4/2/09 | Wed 4/8/09 | | Both |
| 15 | Catch-up / Implement Wireless | 5 days? | Thu 4/9/09 | Wed 4/15/09 | | Both |
| 16 | Testing | 5 days? | Thu 4/16/09 | Wed 4/22/09 | | Both |
| 17 | Prepare for Presentation and Final Report | 5 days? | Thu 4/23/09 | Wed 4/29/09 | | Both |
| 18 | Presentation / Report Due | 5 days? | Thu 4/30/09 | Wed 5/6/09 | 17 | Both |

# Schedule


January 2009
February 2009
March 2009
April 2009
May 2009
1 4 7 10 13 16 19 22 25 28 31 3 6 9 12 15 18 21 24 27 2 5 8 11 14 17 20 23 26 29 1 4 7 10 13 16 19 22 25 28 1 4 7 10
Both
Both
Both
Weston
Chris
Both
Weston
Chris
Both
Both
Both
Chris
Weston
Both
Both
Both
Both
Both

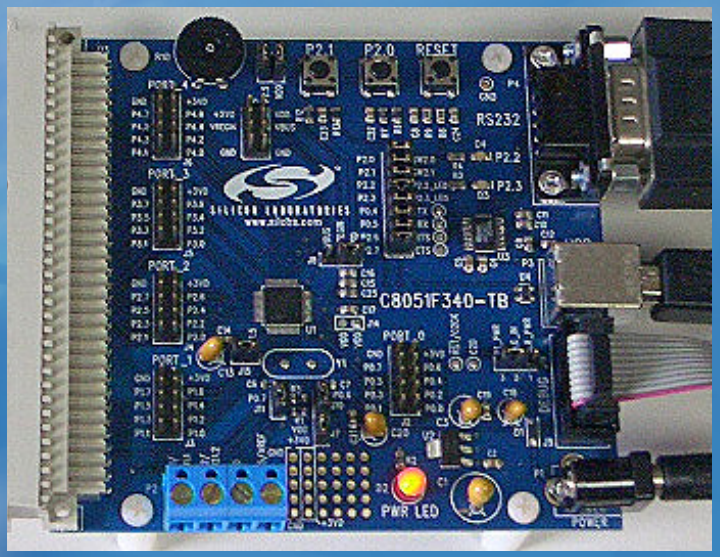
# Completed Work

- Completed or In Progress:
  - USB Gamepad
  - UART Functionality
  - Gyroscope Testing



# USB

- Mouse and Keyboard Working
- Gamepad or Game Controller
  - Error Code 10





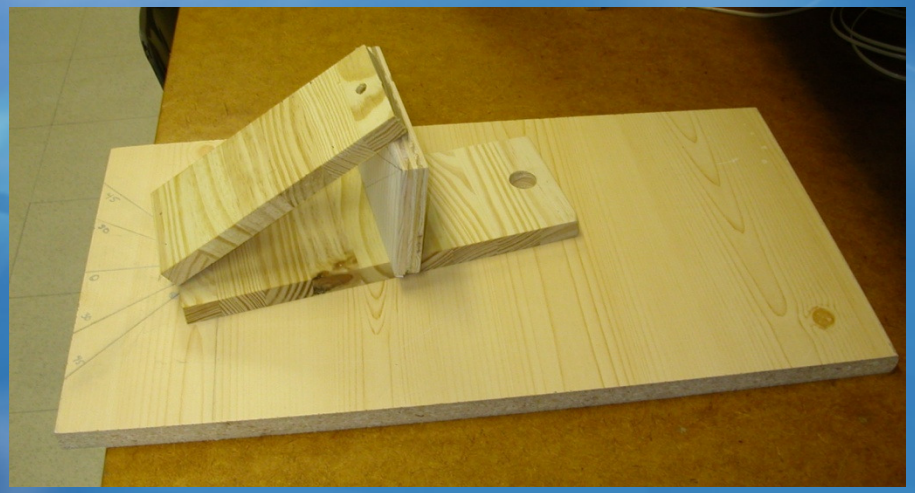
HI-SPEED
CERTIFIED USB

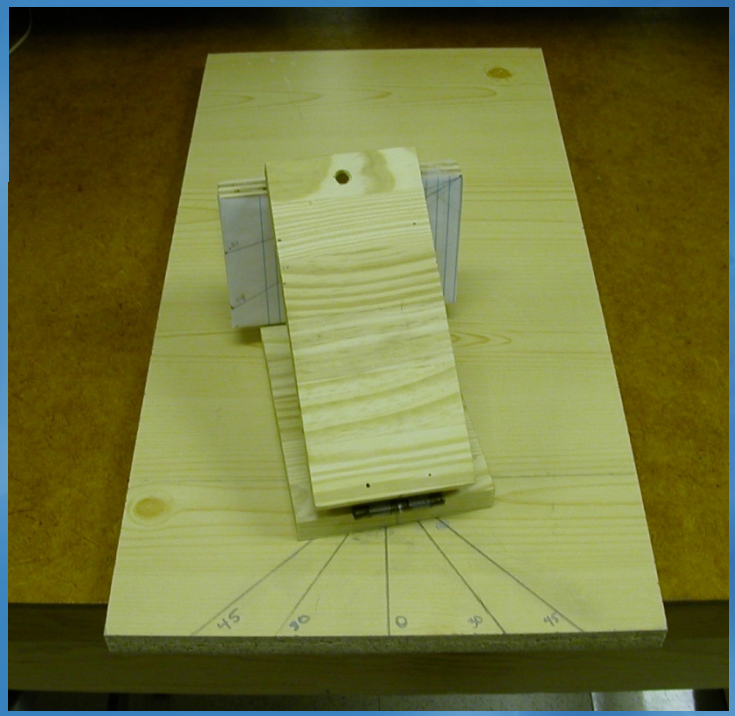
# UART / RS232

- Collect Sensor Data
  - MATLAB
- Debug

# Gyroscope Testing

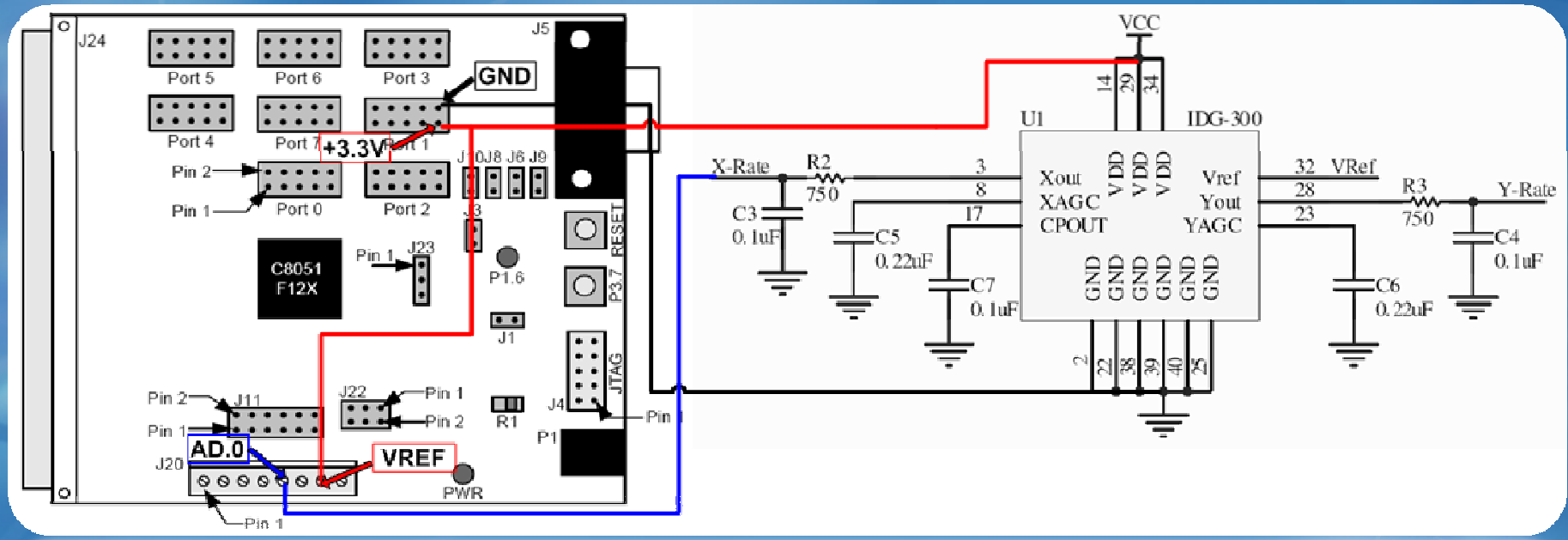
- Test Platform
  - Yaw = 0° – 360° Rotation
  - Pitch = 0° – 90° Rotation





# Gyroscope Testing

- Hardware Schematic



# Gyroscope Testing

- Software – Conversion Factors / Constants
  - Conversion from A/D Value to °/s:

$$\frac{3.3\ \text{V}}{4096\ \text{steps}} \times \frac{1°/\text{s}}{2.0\ \text{mV}} = \frac{1500°/\text{s}}{4096\ \text{steps}}$$

  - Gyro Angular Position = Integration of A/D Data

$$\int \frac{1500°/\text{s}}{4096\ \text{steps}}(\text{ADC})\,dt = \frac{1500°/\text{s}}{4096\ \text{steps}}(\text{ADC})\,t$$

  - Summation As Approximation of Integration

$$\int \frac{1500°/\text{s}}{4096\ \text{steps}}(\text{ADC})\,dt = \frac{1500°/\text{s}}{4096\ \text{steps}}(\text{ADC})\,t \approx \sum \frac{1500°/\text{s}}{4096\ \text{steps}}(\text{ADC})\,\Delta t$$

  - $\Delta T = 1/200\text{Hz} = 0.005$ so Position Approx. is:
    - Angular Position = Previous Position + (Current ADC/546)

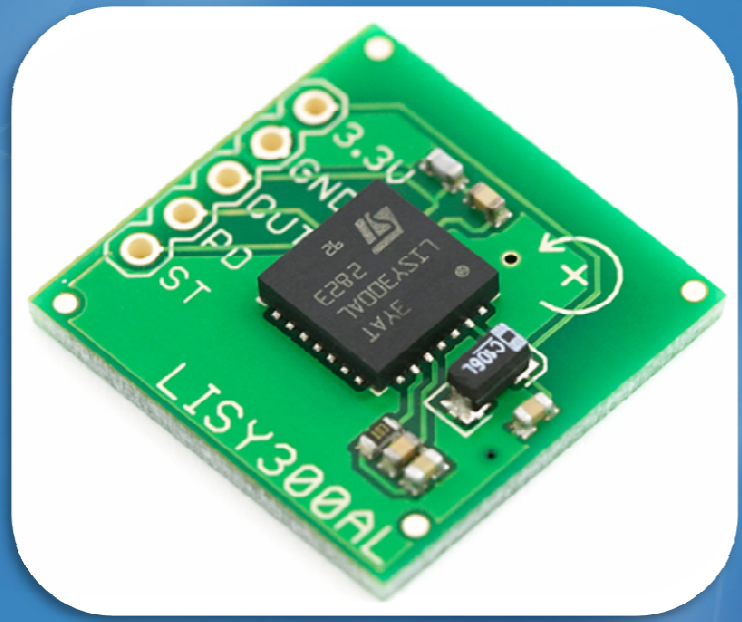
# Gyroscope Testing

- Software
  - A/D Interrupt

```
void ADC0_ISR (void) interrupt 15
{
  // Variables for storage of ADC data
  long yaw = 0;
  static long yaw_integration = 0;  // yaw integration variable
  static int zero_rate_const = 2095;        // Zero Rate Constant
  static int counter_cal = 0;
  static long cal_sum;

    counter++;

    // 12-bit conversion value
    yaw = ADC0H;
    yaw = yaw<<8;
    yaw |= ADC0L;

    // Takes 10 seconds to do this calibration section
    if(counter_cal < 1000) {
        counter_cal++;
        cal_sum += yaw;

    } else if(counter_cal == 1000) {
        yaw_integration = 0;
        counter_cal = 1001;
        zero_rate_const = cal_sum / 1000;    // New zero_rate_const
        cal_sum = 0;     // clear variable
    } else if(counter_cal < 2001) {
        counter_cal++;
        cal_sum += yaw - zero_rate_const;

    } else if(counter_cal == 2001) {
        yaw_integration = 0;
        counter_cal = 2200;
        zero_rate_const += cal_sum / 1000;  // New zero_rate_const
    }

    // Get rid of DC (zero-rate) offset
    //yaw = (yaw - zero_rate_const) / cosf(PI/6);    // use with pitch
    yaw = (yaw - zero_rate_const);  // subtract off Zero-Rate Voltage

    // Window Filter - gets rid of worst noise causing drift
    if(yaw < 8 && yaw > -8) yaw = 0;

    // Integration
    yaw_integration = yaw_integration + yaw;

    // Conversion to degrees
    yaw_degrees = yaw_integration/496;  // Divide by 496 = 2.48 steps/deg * 200 /sec to get to degrees

    // Switch Analog Mux for Axis
    //AMX0SL = 0x00;

    AD0INT = 0;                         // clear ADC conversion complete indicator - must be done manualy
```

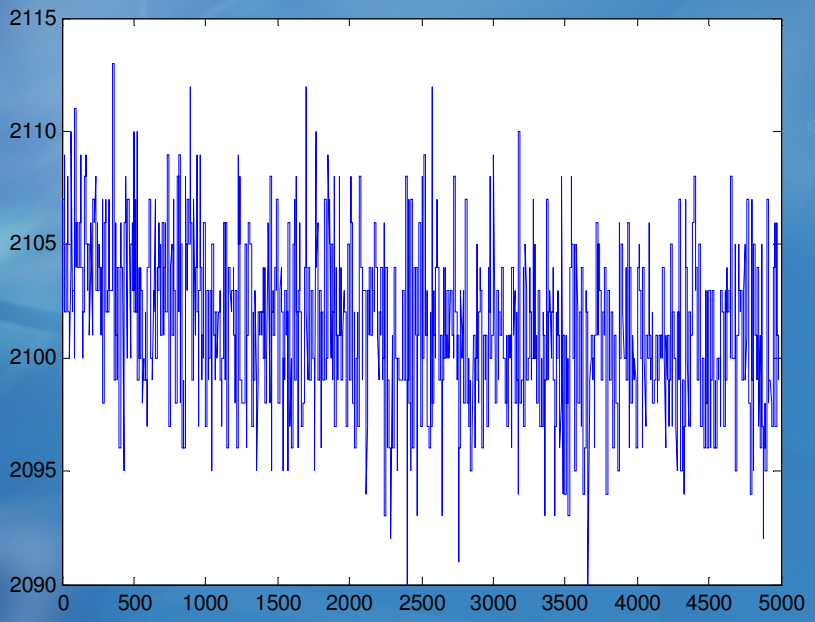
# Gyroscope Testing

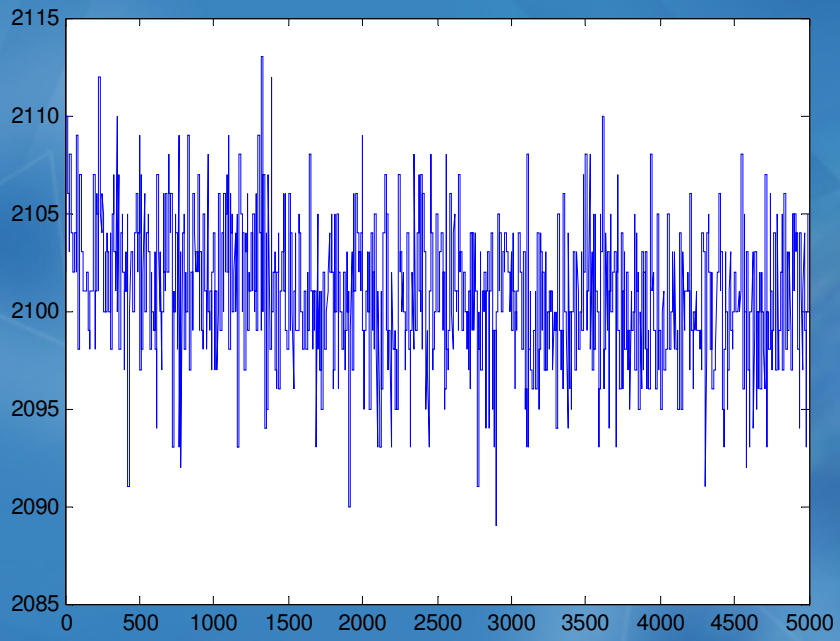
- Started with ST Micro LISY300AL Gyro
  - 5 DOF Board was on backorder
  - 300 °/s gyro (real one is 500 °/s)
  - Yaw Testing
  - Drift



# Gyroscope Testing

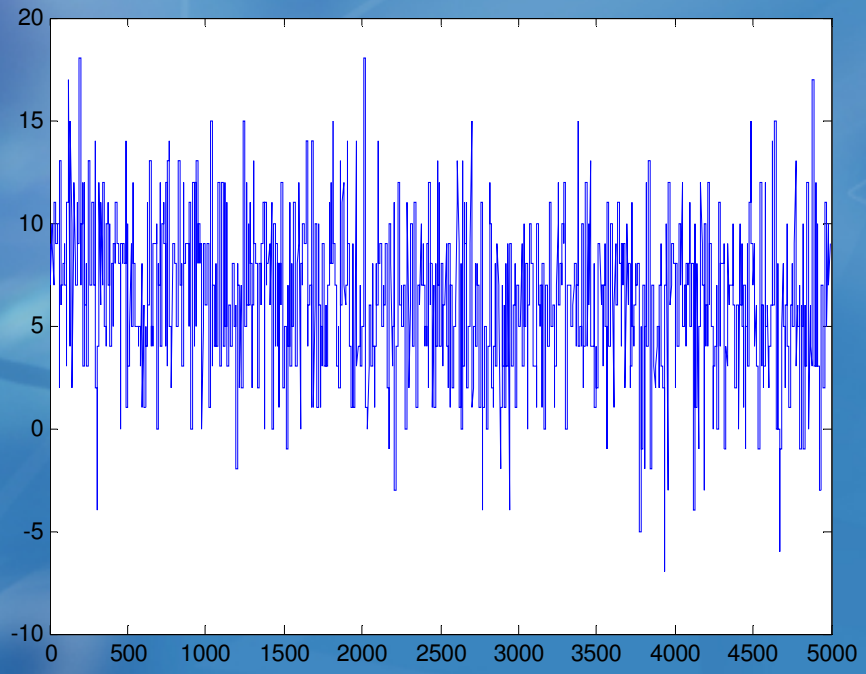
- Drift Analysis
  - HyperTerminal (.txt) → MATLAB
  - Raw A/D Data

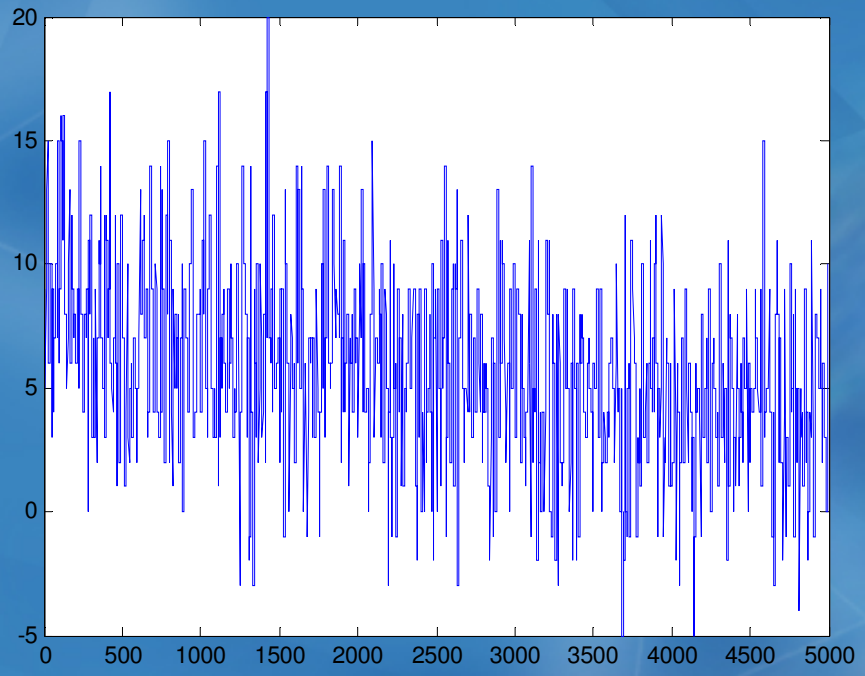


# Gyroscope Testing

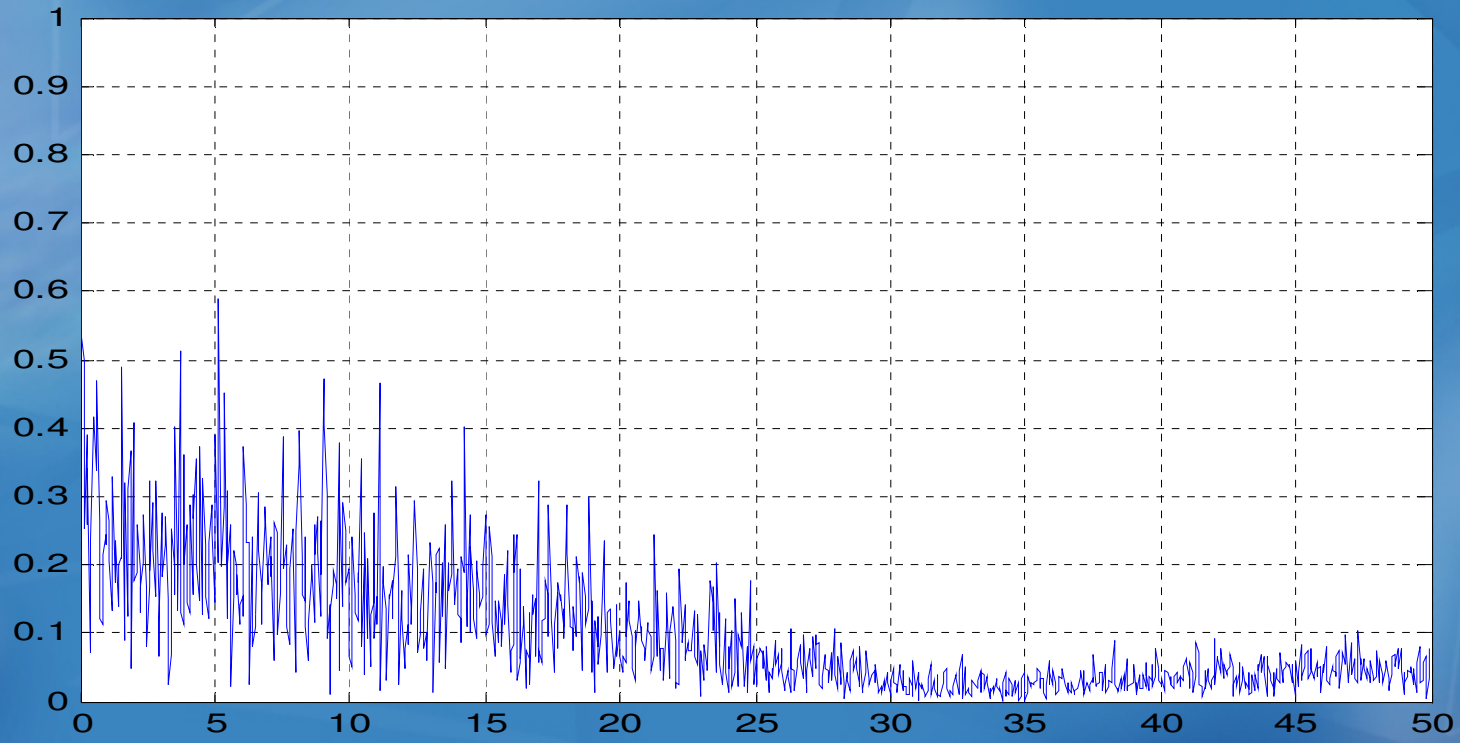
- Drift Analysis
  - After Offset Correction / Calibration (1000 samples)
  - Problem – possibly Zero-Rate Settling Time = 200ms
  - Drifting @ ≈ 1 °/s



# Gyroscope Testing

- Drift Analysis
  - Frequency Spectrum (200 Hz Sampling)
  - Spike at DC = Offset → problem



# Gyroscope Testing

- Our Drift Solutions:
  - Second Calibration

```
// Takes 10 seconds to do this calibration section
if(counter_cal < 1000) {
    counter_cal++;
    cal_sum += yaw;
} else if(counter_cal == 1000) {
    yaw_integration = 0;
    counter_cal = 1001;
    zero_rate_const = cal_sum / 1000;    // New zero_rate_const
    cal_sum = 0;    // clear variable
} else if(counter_cal < 2001) {
    counter_cal++;
    cal_sum += yaw - zero_rate_const;

} else if(counter_cal == 2001) {
    yaw_integration = 0;
    counter_cal = 2200;
    zero_rate_const += cal_sum / 1000;  // New zero_rate_const
}
```

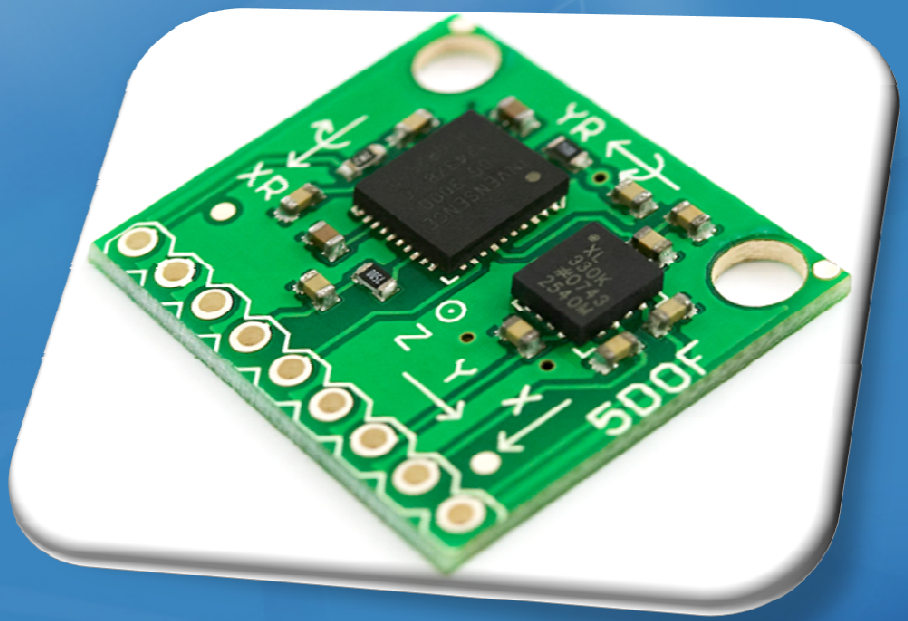
  - Window Filter

```
// Window Filter - gets rid of worst noise causing drift
if(yaw < 8 && yaw > -8) yaw = 0;

// Integration
yaw_integration = yaw_integration + yaw;
```

  - Now Drift ≈ 1° / 15 seconds

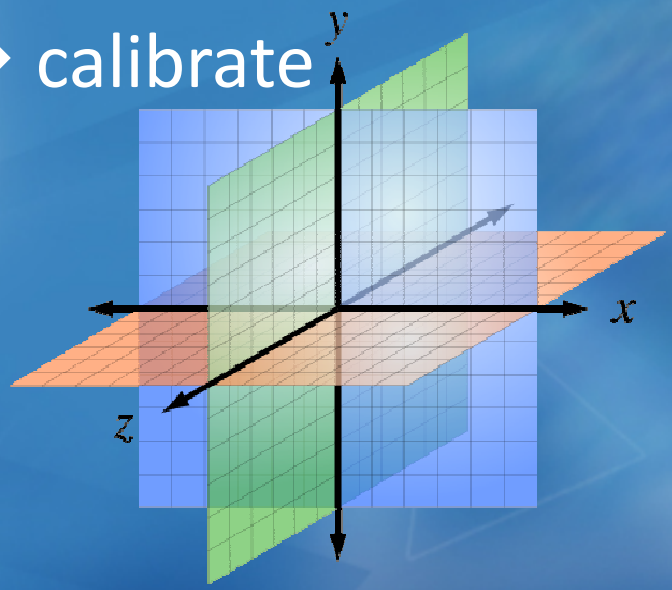
# 5 DOF

- Just Received Last Tuesday (2/10/09)
- Confirmed that code still works
  - With altered constants / conversion factors



# Next Steps

- Frames of Reference Conversions
  - Trig Functions
- Feedback (with still / zero-rate detection)
  - Accelerometers – calculate pitch
  - Electronic Compass – $I^2C$
- Gyro scales different for +/- → calibrate
- Trig Functions Execution Time
  - In interrupt?  Need FPU?



# Questions