DC Motor Cruise Controller Design Project

BY MASON NIXON & JACOB BOSARGE

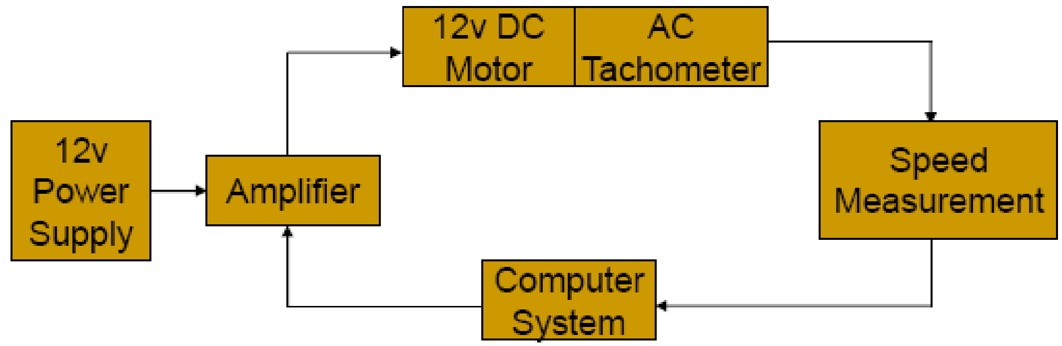
AUBURN UNIVERSITY

SAMUEL GINN
COLLEGE OF ENGINEERING

Overview

- **Project Assignment**
- **Primary Debugging Tools**
- **Cost**
- **Hardware Design**
- **Software Design**
- **Implementation – Experimentation and Control**
- **Problems**
- **Final Design Results**
- **What we learned**
- **Applications of what we learned**

Project Assignment



PID Motor Speed Controller

1. Faster rise time than open loop
2. No steady-state error
3. < 5% overshoot for any speed
4. Fast settling time

Motor Speed Indicator and Selector

1. 3 different speeds using two toggle switches
2. RPM Speed displayed in scientific notation
3. Update display every 250 ms

Primary Debugging Tools

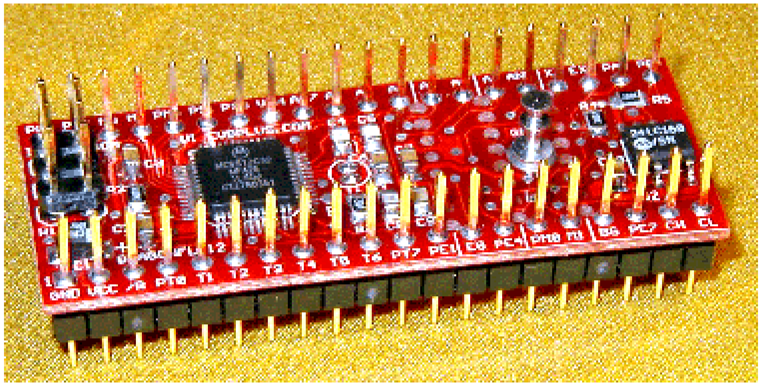
- National Instruments NI 5112 data acquisition board and SCOPE-SFP
- Virtual Bench Logic-Analyzer
- Digital Multimeter
- PSpice

Hardware

- 12V Buehler DC Motor with tachometer
- Freescale MC9S12C32 (HCS12 family) microcontroller mounted in the Wytec Dragonfly 12 DIP-40 evaluation module
- LM311N comparator
- 2N2222 NPN BJT transistor
- 1N4001 rectifier diode

Buehler 12V DC Motor w/ tach

Wytec Dragonfly 12 DIP-40



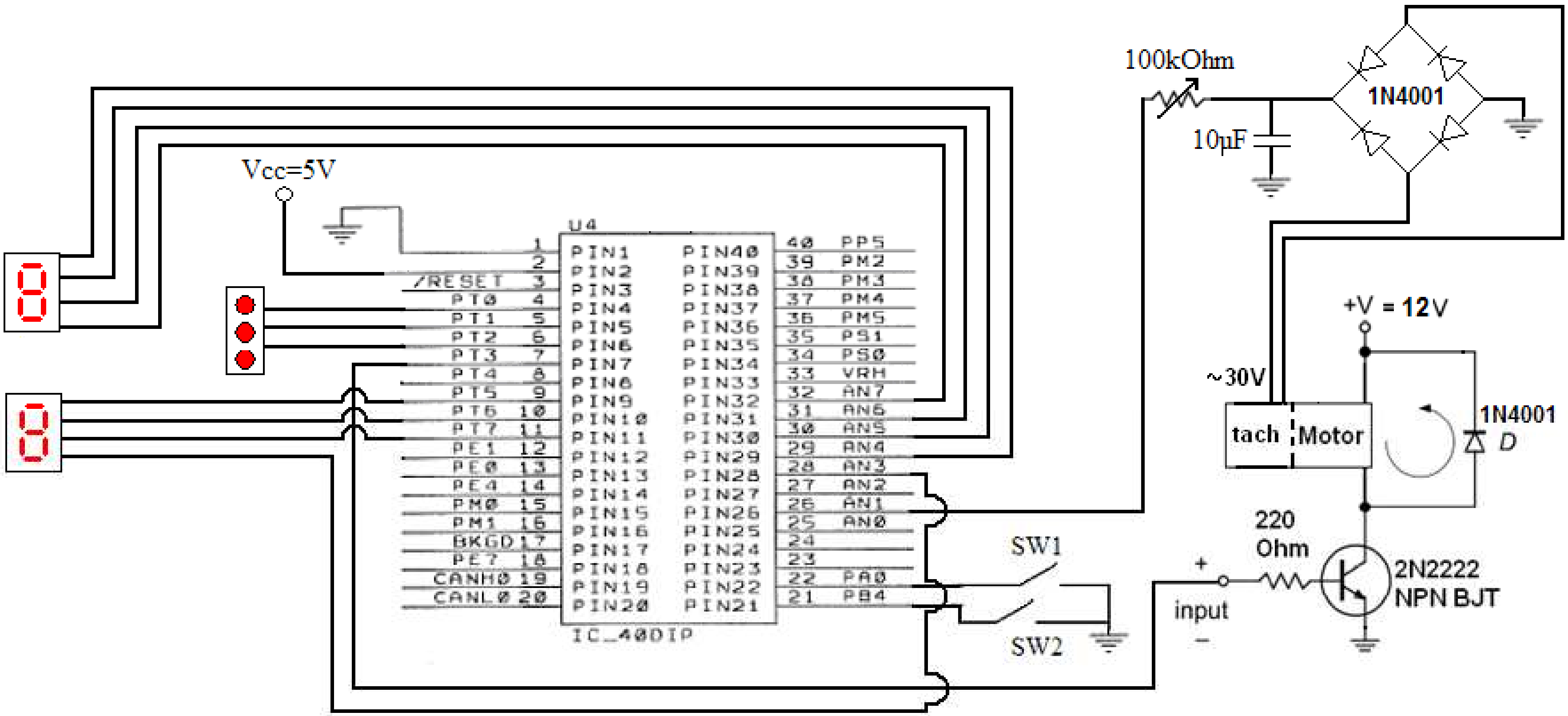
Hardware Cost

6

Hardware	Description	Quantity	Unit Cost
Dragonfly	Microcontroller Development Board	1	$18
Breadboard	Jameco BreadBoard	1	$5.37
Potentiometer	100KOhm, 1/2W	1	$0.83
Diode	1N4001, 50V, 1A	5	$0.10
Capacitor	10microF, 50V	1	$0.21
Resistor	220Ohm, 5%, 1/4W	1	$0.05
Transistor	2N2222A NPN BJT	1	$0.41
Total Cost			**$25.37**

Note: All prices obtained from the AU Chemical Supply Store, except the Dragonfly, which can be found here: http://www.evbplus.com/c32_modules/DIP40.html

Hardware Design – circuit schematic

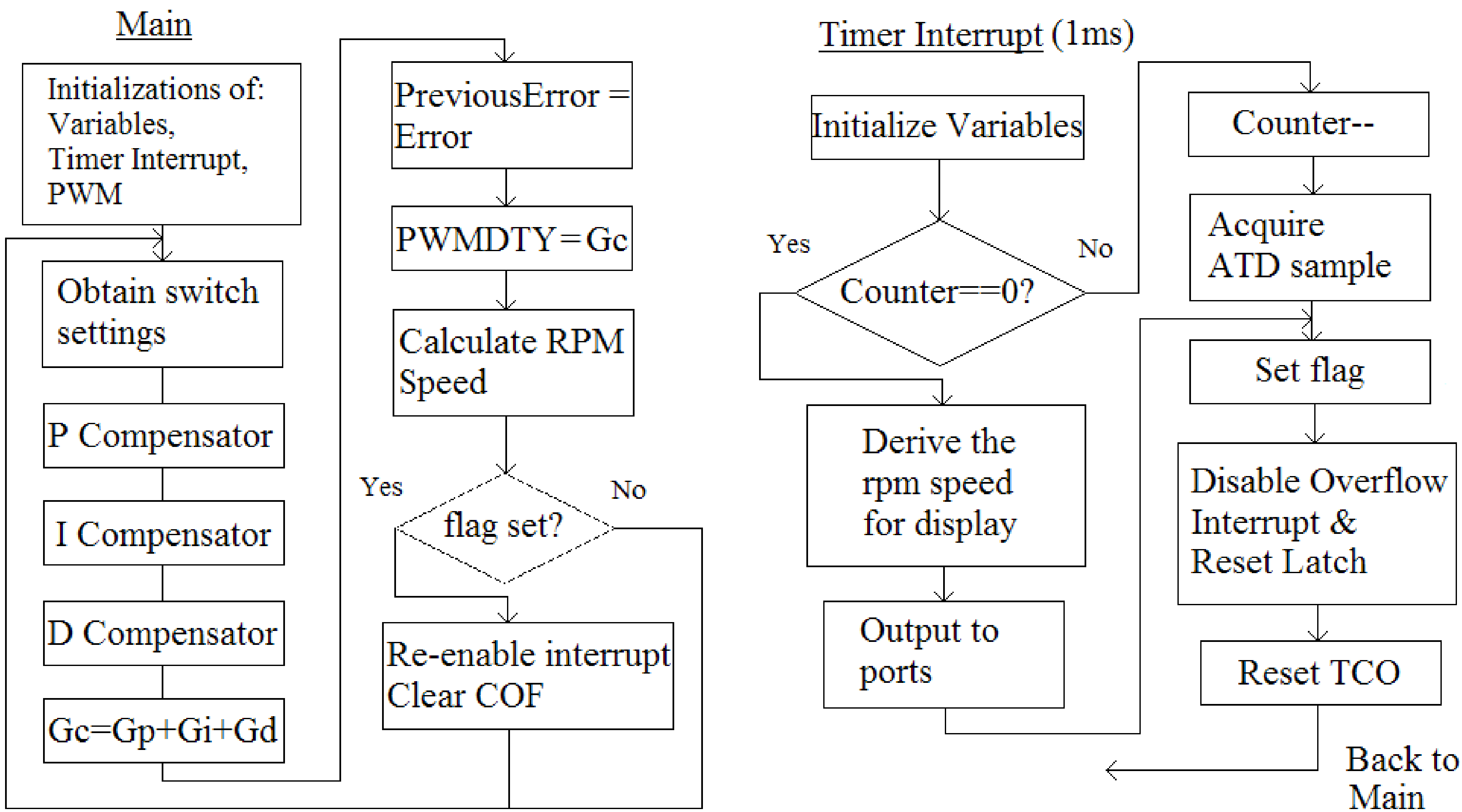


RPM Speed Display

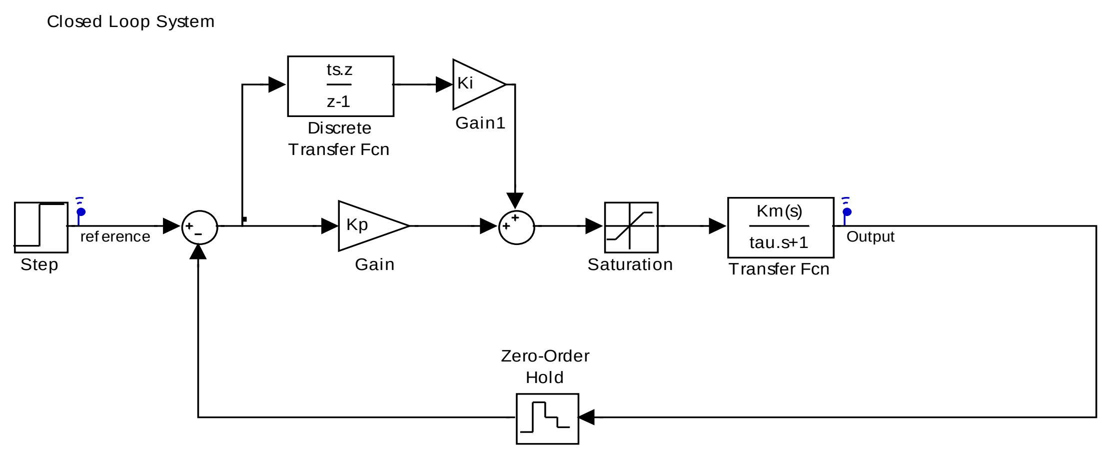
HC12 Microcontroller

Motor Drive System

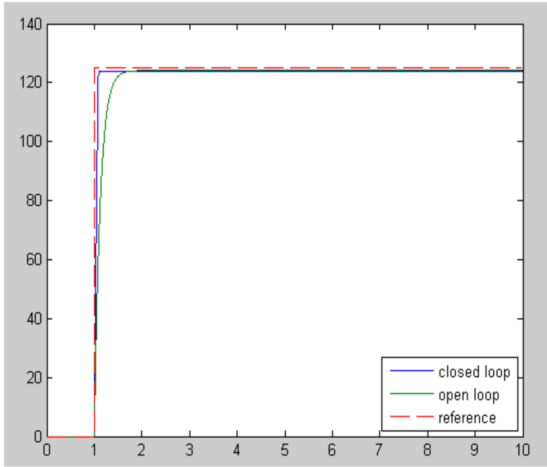
Software Design – Flow Chart

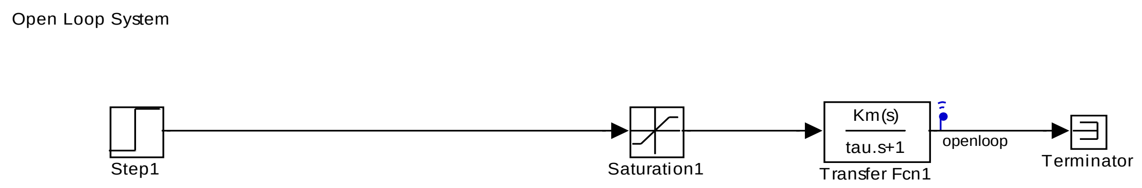


Designing our controller: Simulink



Simulation results





$$C(s) = \frac{0.99287}{(0.12186s + 1)}$$

Design Decisions

ADC vs Input capture

Onboard analog to digital conversion is an option on the HC12.
It is designed to take analog inputs, therefore it is simpler to implement.

RPM speed

$$rpmspeed = \frac{rotations}{minute} = frequency\left(\frac{cycles}{second}\right)\left(\frac{60\ seconds}{minute}\right)\left(\frac{rotation}{8\ cycles}\right)$$

$$rpmspeed = frequency\ x\ 7.5$$

By dividing the rpmspeed by our ADC value for each switch setting and then taking an average of these slopes we get 2.41, then we are able to get a constant scalar of the ADC value of 2.41*7.5=18.1.

$$rpmspeed \approx ADC\ x\ 18.1$$

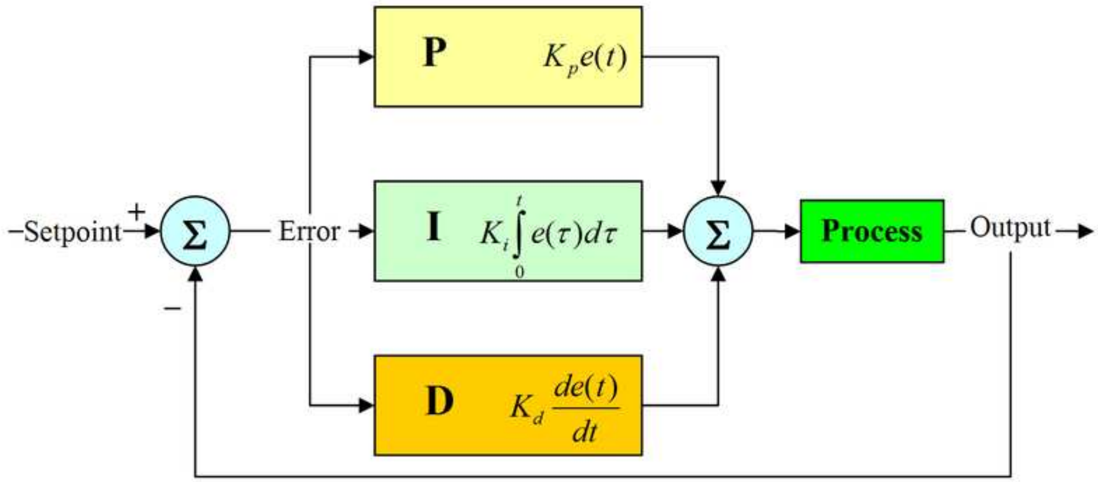
Design Decisions

PI vs. PID

PI
- increases system type by one
- improves steady state response
- reduces stability

PID
- reduces overshoot
- increases stability over PI

Implementation of the PID controller



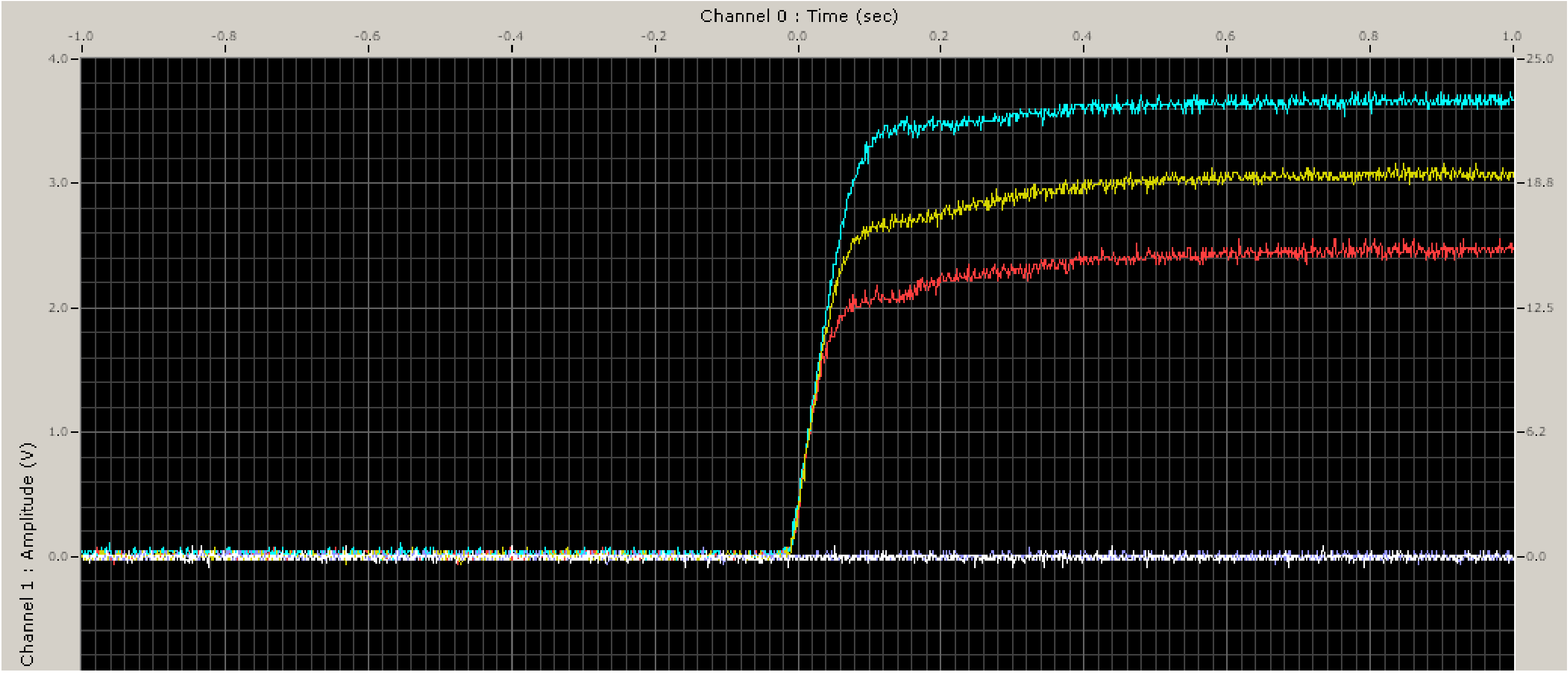

```
error=refvalues[swstate]
-sample;
Gp=Kp*error;
Ierror+=error;
temp=Ki*Ierror;
Gi=temp/(1000);
Derror=(preverror-
error)/2;
temp=Kd*Derror;
Gd=temp/(1000);
Gc=Gp+Gi+Gd;
preverror=error;
temp=0;
PWMDTY3=Gc;
```

Term	Effect on Control System
P Proportional	Typically the main drive in a control loop, KP reduces a large part of the overall error.
I Integral	Reduces the final error in a system. Summing even a small error over time produces a drive signal large enough to move the system toward a smaller error.
D Derivative	Counteracts the KP and KI terms when the output changes quickly. This helps reduce overshoot and ringing. It has no effect on final error.

Problems and Experimentation

- Saturation block
- Sample time
- Set compensator to interrupt
- Hard-coding zero-case instead of calculating it
- Correct the derivative error
- Change the gains (Kp, Ki, and Kd)
- Plot data in Excel
- Time constraints

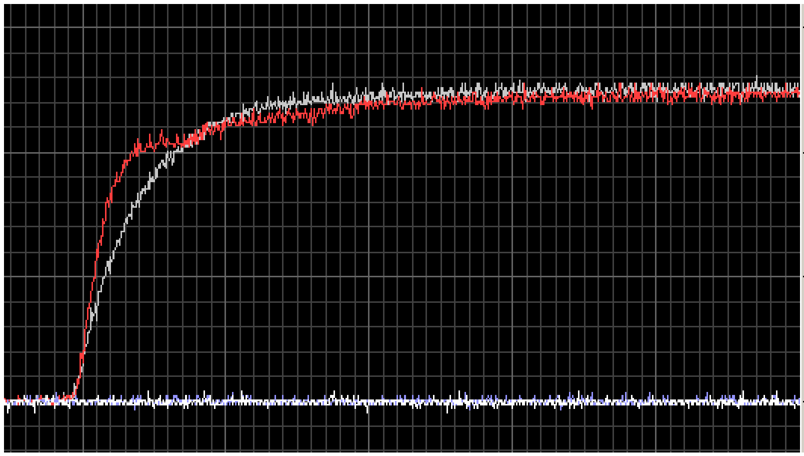
Results



Speed	Overshoot
Low	0.2941 %
Medium	0.2703 %
High	1.8595 %

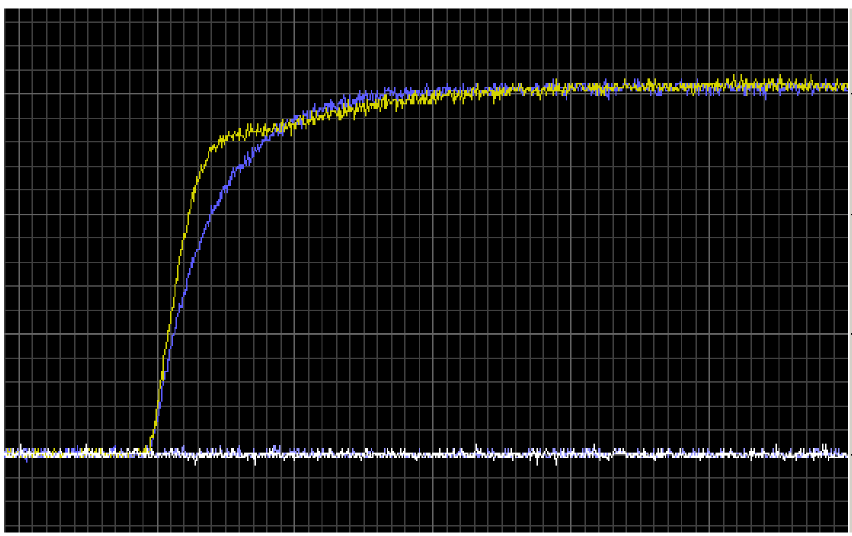
Results

50%: Closed vs Open

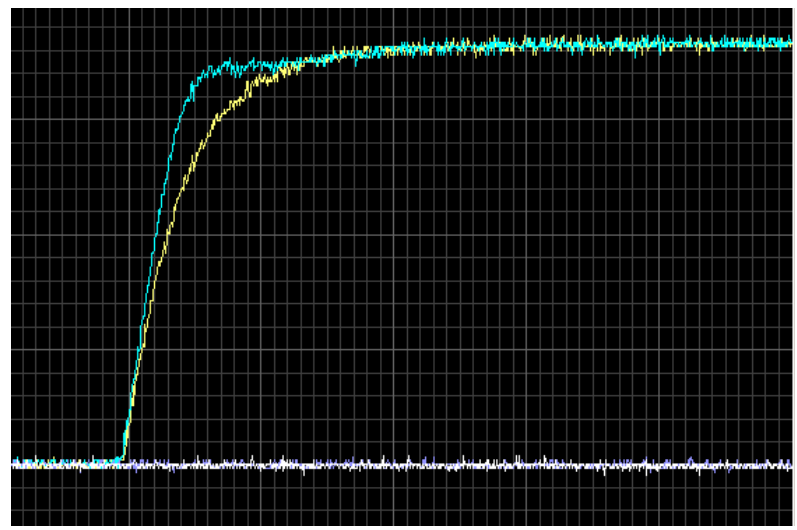


The closed loop response is faster than the open loop.

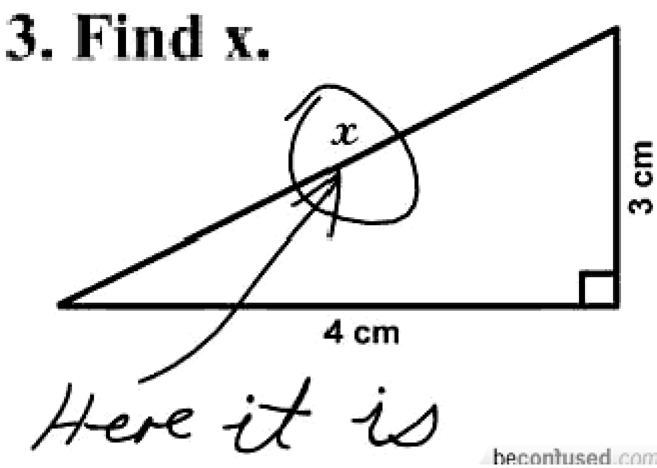
62.5%: Closed vs Open



75%: Closed vs Open



What We Learned



- Check variable types
- Does the math work out?
- Avoid assumptions

- How a PID controller works
- In depth understanding of MATLAB and Excel
- Better programming techniques
- Get things done early

Applications

Mine detecting robots

Magnetic Signature Management

Cruise Control

Questions?