
Embedded Controller Development for Closed-Loop Process Controls Learning

Athula Kulatunga, Ph.D., CEM, Associate Professor
Ryan Muir, Graduate Research Assistant
IR - Power Electronics Development & Application Lab
ECET Department, Purdue University, West Lafayette, IN, USA

Introduction

The US Department of labor predicts the demand for engineers with the ability to develop more efficient and cost effective industrial processes will grow more than 20% over the next decade [1]. The same report predicts a “less than average” growth for electronics engineers for the same time period except in engineering and design service firms.

Energy efficiency is receiving higher priority in the US industrial sector. Closed loop process control is recognized as one of the areas that can improve the efficiency of all processes [2].

Large corporations such as Black & Veatch and American Electrical Power are actively reaching out to electrical engineering technology programs that teach power and process controls to hire interns and engineers. The abilities of engineering technologists, especially with hands-on experiences, are highly valued at these corporations [3]. Future growth of these industries relies on their engineers’ ability to apply electronics, electrical power, and process controls principles to solve real world problems. Development of most power electronics based systems such as motor drives and switching power supplies requires knowledge related to microcontroller based PID loop implementation.

The following hands-on learning activities have been developed and tested with students to teach microcontrollers and closed loop process controls in an integrated manner. The goal is to provide enough information to readers so that they can duplicate the activity. Additional information can be obtained by contacting the authors.

Activity Development

Prior Knowledge

The first item to consider is the students’ prior knowledge. The emphasis given to electronics circuit development in an ABET (Accreditation Board of Engineering and Technology of American Society of Engineering Education) accredited electrical engineering technology (EET) program may vary from institution to institution. If an EET program has high concentration in analog and digital electronics, then this activity would benefit student learning.

Former students of the process controls class at this institution have requested the knowledge related to microcontroller-based, closed-loop PID (Proportional-Integral-Derivative) controller development experience in addition to analog PID controller circuits that had been a part of the course for many years. The analog closed-loop section includes the development of a PI (proportional and integral) temperature controller that maintains the temperature of a thermoelectric module (or Peltier module).

The activity described in this paper involves a simple method of using a microcontroller to control the speed of a dc motor using standard electronic components. The design consists of deliberately introduced bottlenecks to illustrate potential problems.

It is assumed that students have learned the fundamentals related to obtaining process characteristics, calculating controller tuning constants, and tuning methods such as step response, ultimate cycling, and manual tuning. References [4] and [5] cover these topics.

Block Diagram

Figure 1 depicts the block diagram of the motor controller. The instructional objective is to teach how to implement the block diagram using a microcontroller and power electronics components.

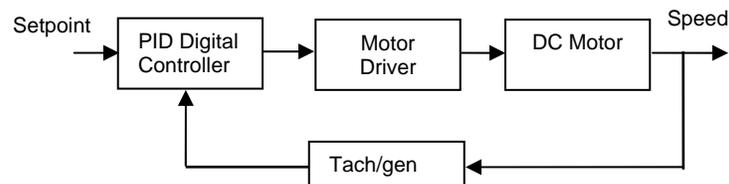


Figure 1 Block diagram of the controller

Features

Certain features are necessary to make the circuit functional as a learning tool. First, motor speed must be varied. The speed of a dc motor is proportional to the average dc voltage. The average of the applied dc is adjusted by varying the duty cycle of the applied voltage, through the use of Pulse Width Modulation (PWM). The setpoint is adjusted manually, via a potentiometer, or by applying an external signal.

Second, the proportional gain value (K_P), integral constant value (K_I), and derivative constant value (K_D), should be able to be entered externally. A set of switches are placed to select a digital equivalent of those tuning constants and another switch to load the value to the controller.

Finally, the behavior of the process variable, the measured variable, and the manipulated variable should be easy to monitor. The process variable is the speed of the motor shaft. A motor-generator pair is shown in Figure 2, where one operates as the motor and the other as the tachometer/generator.

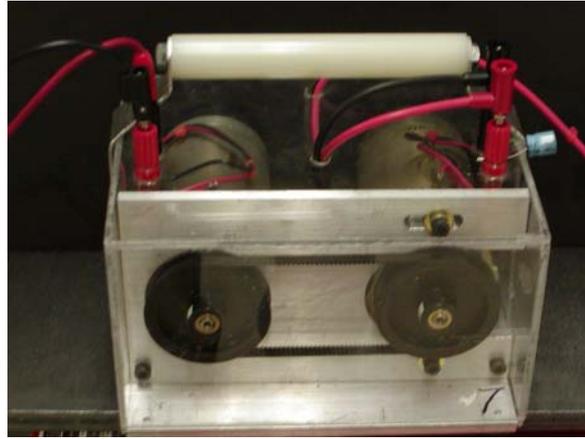


Figure 2 Motor-generator pair

Several test points are spread over the circuit to take electrical measurements easily. Accidental grounding may cause damage to the motor driver, especially with earth grounded oscilloscopes. The best practice is to use a differential probe connected to an oscilloscope to measure voltages.

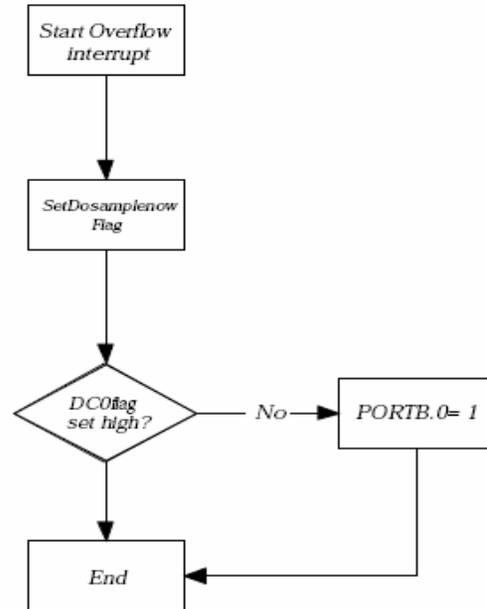
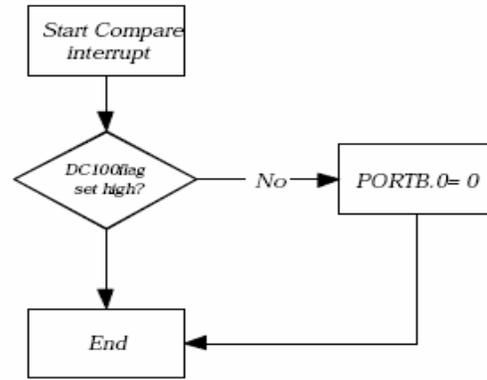
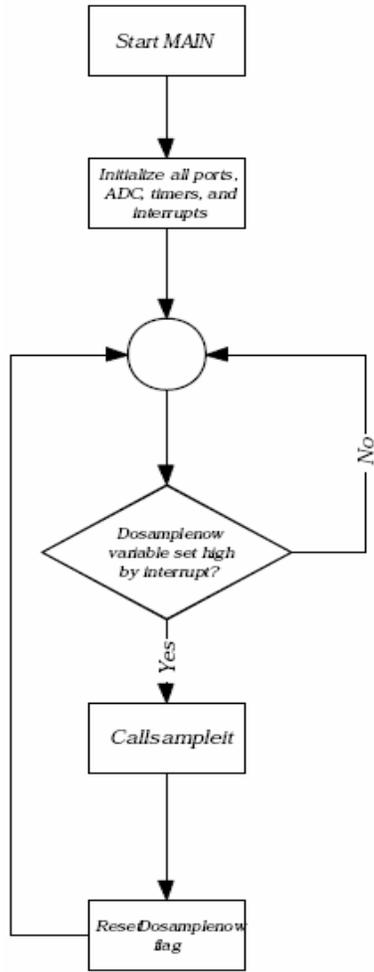
Device Selection

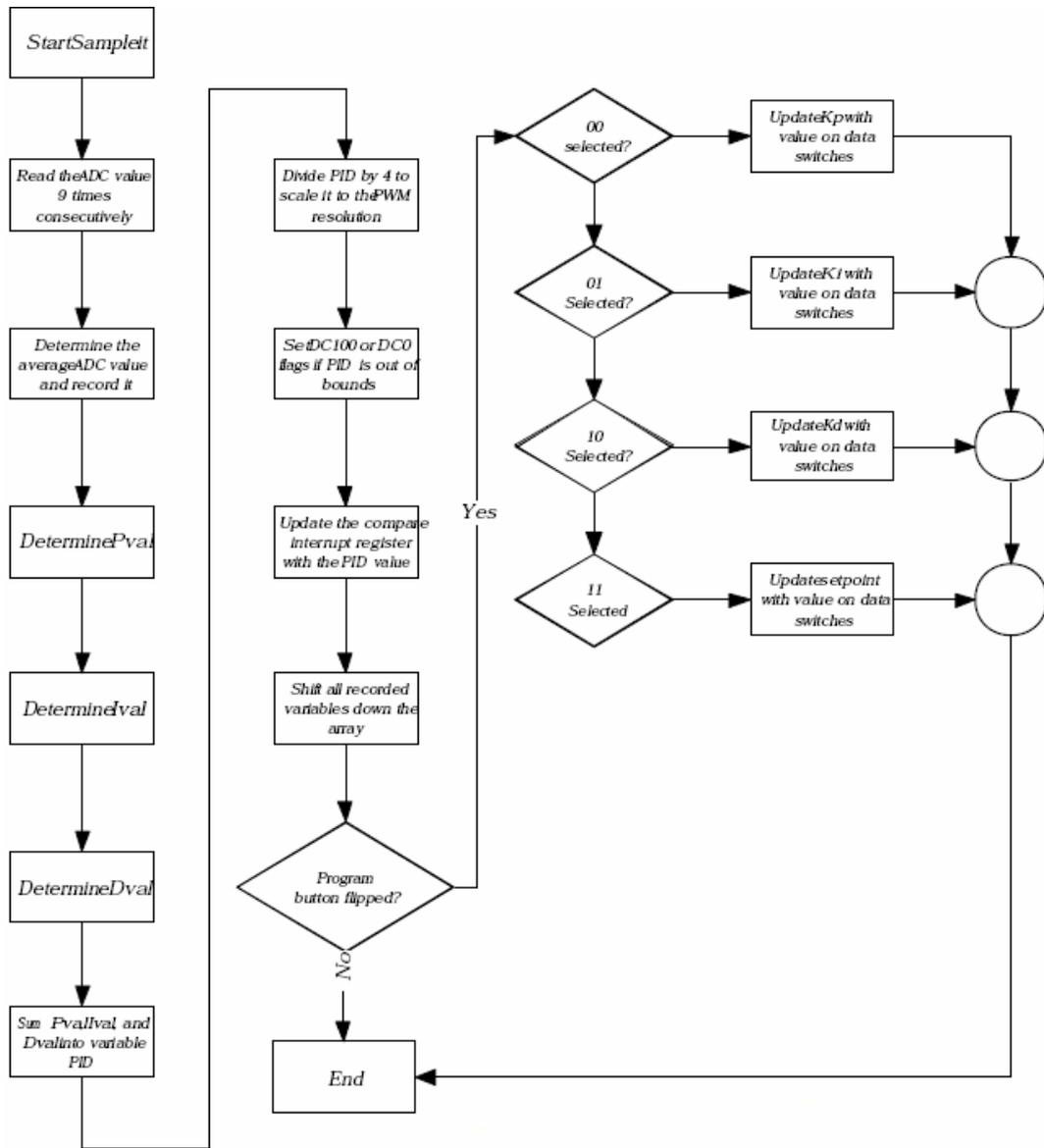
An ATMEL MEGA 16 microcontroller is chosen as the controller because all EET majors at this institute use it in digital electronics courses. The built-in A/D converter has good enough resolution for closed loop control applications.

The motor driver consists of LMD18201, which has a built-in H-bridge driver and four MOSFETs. The device accepts PWM signal and generates a proportional dc average voltage.

Software Description

The following is the top portion program description. The remainder of the code can be obtained from the authors upon request. The entire program was first given to students as a pre-lab activity. The students then identified the purpose of each line before programming the microcontroller. Figure 3 below depicts the flow chart and a portion of the ATMEL code.





/******

This program was produced by the
 CodeWizardAVR V1.24.6 Standard
 Automatic Program Generator
 © Copyright 1998-2005 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfotech.com>
 e-mail:office@hpinfotech.com

Project : Atmel PID Motor Driver
 Version : Muir_1.2
 Date : 1/9/2008
 Author : Ryan Muir
 Company : Purdue University
 Comments:

I/O Setup:

Input: PINC[7..0], PINA[7..4] are numerical input. PINA[3..2] selects the variable to be modified.

-->Selectable variables are P, I, D, and setpoint.

--> PINC is the most significant byte, and PINA[7..4] is the least significant nibble

PINA.1 is activated to program the selected number into the selected variable.

PINA.0 is the 10 bit A/D converter. It is referenced to Aref, but this can be shorted to AVcc, if desired.

PIND[1..0] were used for serial communications during debugging.

Output: PORTB.0 is the PWM output.

User Input Notes: The 12 bit number inputted is scaled such that the input ranges from 0 to 63.98437 in steps of 0.01563. That is, dividing the 12 bit number by 16 shows the number that will be entered into the selected variable.

The setpoint is only a 10 bit number ranging from 0 to 1023. Dividing the 12 bit input number by 4 will give the number to be entered into the setpoint, but all decimals are dropped. So, the 2 least significant bits will be discarded when entering this number.

Feedback Noise Notes: The generator side of the ECET student motor/generator units is very electrically noisy. This noise strongly negatively affects sampling accuracy. The problem is alleviated by sampling multiple times consecutively and then taking the average of the sampled values. In this code, the feedback voltage is sampled 9 times consecutively, and the average feedback voltage from those samples is calculated and used for PID algorithm calculations.

PID Algorithm Notes: The PID algorithm is a direct summing algorithm. P, I, and D calculation are traditional, straight-forward, and fairly intuitive. As stated above, all values used in these algorithms are values averaged from the ADC sampling.

P is found by multiplying Kp by the value of the current error (error is determined by SP - PV).

I is found by "determining the area under the curve." The current sample and past 10 samples are integrated together by summing them all together, and then multiplying the sum by the time period that all 11 samples were taken over.

This product is then finally multiplied by the integral gain variable, K_i . D is found by determining the slope of the current and past variables from the entire sampling history of 11 samples. This is done by finding the average value of the 5 most recent samples, the average value of the 5 oldest samples, and then determining the slope of the line between these two averaged points. Slope is determined by dividing the difference in value between these two averaged points, divided by the averaged difference in time between these two points. Finally, this value is multiplied by the derivative gain term, K_d .

PWM Notes: Timer0, an 8 bit timer, is used to create the pulse width modulation output. Modifications have been made such that a true 100% duty cycle and a true 0% duty cycle can be achieved. It ticks at a rate of 5.859kHz.

```

Chip type      : ATmega16L
Program type   : Application
Clock frequency : 6.000000 MHz
Memory model   : Small
External SRAM size : 0
Data Stack size : 256
*****/

```

Figure 3 Flow chart and a portion of the software program for the controller

PCB Design

Figure 4 depicts the schematic diagram of the controller. User input is controlled by the 2 sets of DIP switches. The lower 3 switches determine which variable inside the ATmega16 microcontroller is to be modified, as well as the “latch” switch, which overwrites the selected variable with the data on the data lines. The upper 12 switches control the data (values for P, I, D, and setpoint variables) lines.

The board is designed to be powered from a single supply, up to 24V. The 24V voltage is directly applied to the motor driver. The voltage is stepped down by the LM7805C 5V voltage regulator to power the microcontroller.

During operation, DC tachometer/generator feedback is supplied into a potentiometer voltage divider. It is intended that the user set the potentiometer so that, at maximum speed, the ADC sees +5V. A zener diode to ground is included to prevent excessive voltage from reaching the ADC. A jumper block is present at the ADC pin in order to facilitate tuning the potentiometer or system.

Several jumpers and test points are included in the design in order to facilitate future expansions to the board. External input to the motor driver for direction, brakes, and

PWM signals can be applied by removing jumpers and asserting a signal onto the appropriate pin. Numerous test points are located at all of the interesting voltage nodes, giving instructive value to students about the operation of the circuit.

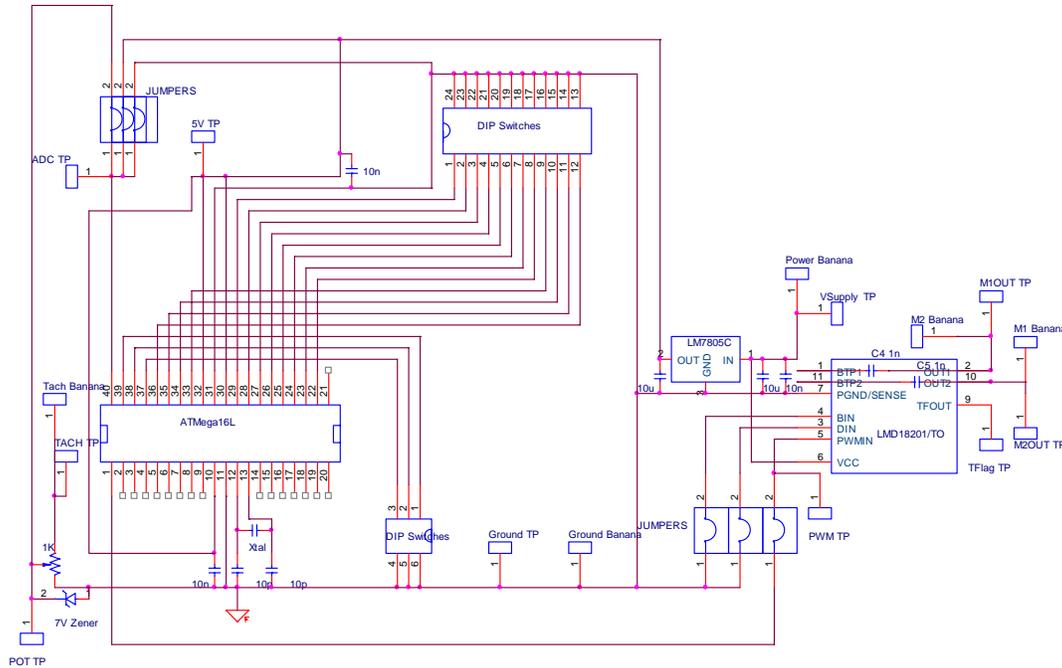


Figure 4 Schematic diagram

Opportunities for Lab Activities

Instructional Objectives and Lab Setup

At the end of these learning activities students will be able to

- 1) design and build a microcontroller based dc motor drive
- 2) obtain a process characteristics curve
- 3) determine the proportional, integral, and derivative gains
- 4) verify the tuning of the controller

The design process and component selection were explained in class. Then students were asked to develop a PCB for the above circuit. Some students needed help selecting correct diameters for holes, sizing of traces for appropriate current, etc. Figure 5 illustrates a sample layout. Table 1 summarizes the function of each switch and jumper.

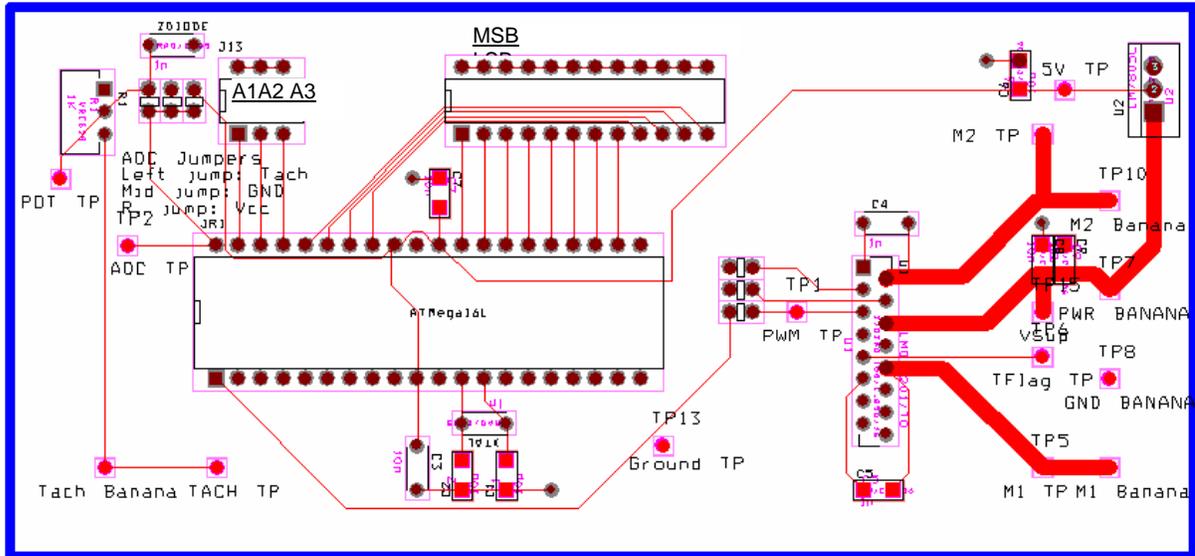


Figure 5 Sample PCB layout

Table 1 Switch and jumper descriptions

A1

UP	1	Switches active
DOWN	0	Switches inactive

A2, A3

A2	A3	Variable selected
1	1	Kp
0	1	Ki
1	0	Kd
0	0	Setpoint

ADC Jumpers

Jumper	When connected
Left	Tach to ADC
Middle	ADC pin grounded (use for to set up feedback voltage to 5V)
Right	ADC at Vcc

Preset parameters

Variable selected	Selectable input range	Calculation of input	Default value
Kp	0-63.98	Switch value/64	0
Ki	0-63.98	Switch value/64	0
Kd	0-63.98	Switch value/64	0
Setpoint	0-1023	Switch value/4	1024

A1

UP	1	Switches active
DOWN	0	Switches inactive

A2, A3

A2	A3	Variable selected
1	1	Kp
0	1	Ki
1	0	Kd
0	0	Setpoint

ADC Jumpers

Jumper	When connected
Left	Tach to ADC
Middle	ADC pin grounded (use for to set up feedback voltage to 5V)
Right	ADC at Vcc

Preset parameters

Variable selected	Selectable input range	Calculation of input	Default value
Kp	0-63.98	Switch value/64	0
Ki	0-63.98	Switch value/64	0
Kd	0-63.98	Switch value/64	0
Setpoint	0-1023	Switch value/4	1024

Initially, students collect data to understand the relationship between applied motor voltage, the motor current, and the motor speed.

The wiring diagram is shown in setup is shown in Figure 6. Power supply is set to the rated voltage of the motor.

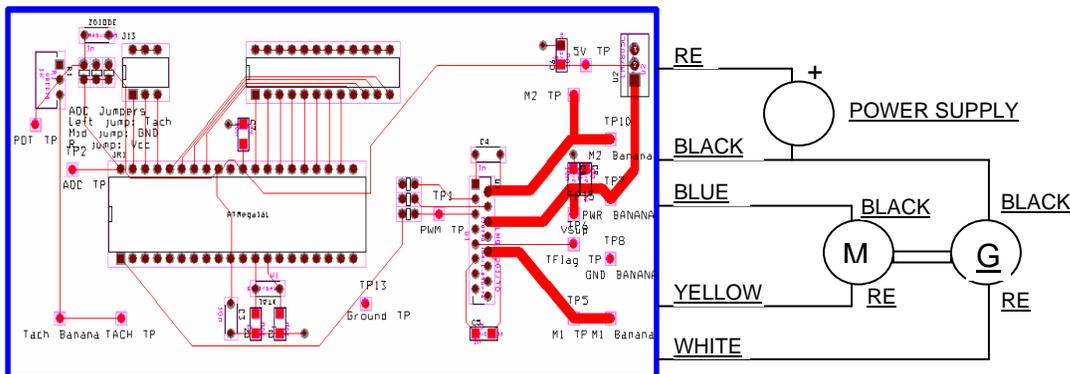


Figure 6 Equipment setup

Figure 7 depicts the entire setup, including the measuring equipment. Two differential probes are used, shown in left, and a dc clamp, shown in right, to capture the microcontroller output, the motor input voltage from the H-Bridge driver, and the motor current.



Figure 7 Equipment setup photo

To obtain the process output curve (motor speed), motor input voltage must be changed in a step fashion and the generator output must be recorded using an oscilloscope or strip chart recorder. This produces a first-order lag response.

Using the process response curve, students will be able to obtain process characteristics such as process gain (K), dead time (t_d), and time constant (τ). These values are helpful in determining appropriate K_P , K_I , and K_D , using the Ziegler-Nichols method. Next, student may try the ultimate tuning method to get the tuning constants, K_P , K_I , and K_D . Finally, students may try manual tuning to get the fastest response possible. The tuning process can be verified by observing the quarter-amplitude-decay behavior of the process variable [4 & 5].

Sample Waveforms

The setup also allows students to experiment with the motor response to various combinations of K_P , K_I , and K_D . For example, Figure 8 shows the PWM output of the microcontroller (top waveform), the motor input voltage (middle waveform), and the motor current (bottom waveform) with low gain values of K_P (left) and very high K_P (right). Too high proportional gain results oscillation. Notice that the duty cycle is changed to maintain the motor speed around the setpoint value. The differential probes are at 20 times attenuation for voltage measurements and no attenuation for the current probe.

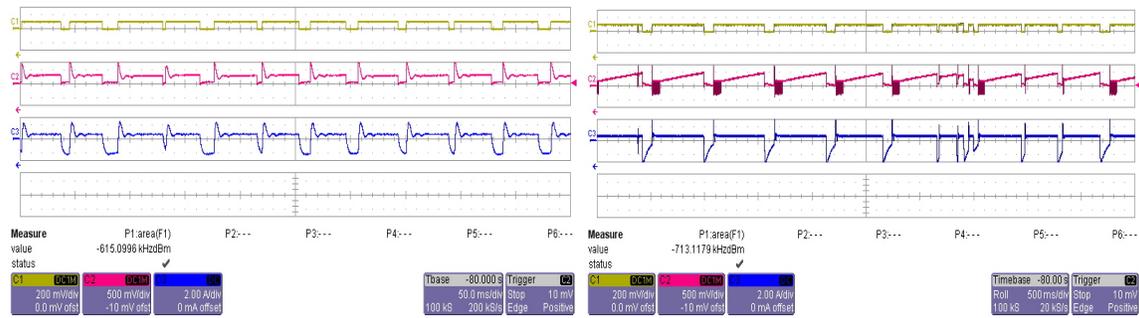


Figure 8 Waveforms for low proportional gain (left) and very high proportional gain (right)

By adding the appropriate K_P , K_I , and K_D the oscillation is minimized. See figure 9.

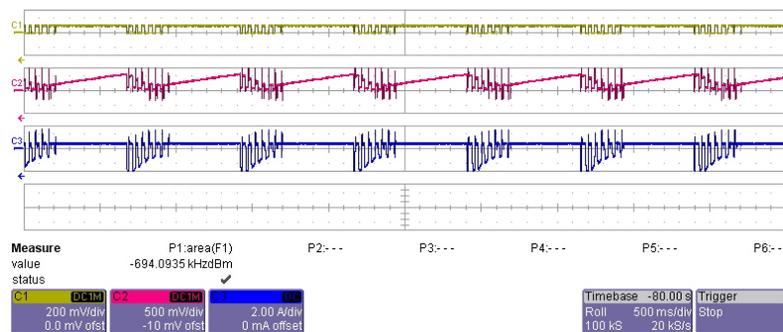


Figure 9 A tuned response is obtained by adding appropriate K_P , K_I , and K_D

Student Feedback

A simple survey was given to obtain feedback from students. A pre-survey indicated that ninety percent of the students enrolled in the course had no prior experience related to closed loop control with a microcontroller other than simple on-off control. Eighty percent mentioned that they did not know how to integrate hardware for motor speed control. A post survey asked students to propose ways to improve the given controller. Students realized the limitations of using a second brushed dc motor as the feedback sensor and the need for correctly setting up the range for the variables. They also found that a capacitor can do much to reduce the ripple in a feedback loop before disturbing the loop action. Most students preferred the manual tuning method compared to the other two methods. Some students said that they would use a similar PID type controller in their capstone projects.

Conclusions

Microcontroller implementation of closed loop process control is essential in power electronic based motor controllers, power supplies, and industrial process controls. This hands-on activity helped students learn the process of a microcontroller based PID controller and its development. The time of the learning activity will depend on the students' prior knowledge. Several students had difficulty with the programming aspect, but were excited to learn how pulse width modulation (PWM) works in a controlled fashion. This project could be improved by using a good tachometer instead of a second dc motor. However, the entire experience was very positive for students. Since the board is not fault proof, it is recommended to have several additional boards made available during the lab activity. As follow-up to this activity, a lab exposing the operation of a commercially available PID speed controller would be beneficial to students as well.

References

1. U.S. Department of labor, Bureau of Labor Statistics, *Occupational Outlook Handbook*, http://www.bls.gov/oco/ocos027.htm#projections_data, December 18, 2007
2. Welander, P., *Energy as a Process variable*, Control Engineering, 1/1/2008, <http://www.controleng.com/article/CA6518156.html>
3. Personnel meeting with American Electric Power upper management at AEP corporate headquarters in Columbus, OH, on 1/23/2008
4. Bateson, R. A. (2002), *Introduction to Controls System technology*, Prentice Hall, ISBN 0-13-0306888-6
5. Guptha, S. (2002), *Elements of Control Systems*, Prentice Hall, ISBN 0-13-011989-X