Ronald Macmaster and Parth Adhia

EID: Rpm953 and Pda375

11/11/2016

EE 445L – Lab 10: DC Motor Control

Requirements Document

1. Overview

1.1. Objectives: Why are we doing this project? What is the purpose?

Our primary objectives for this project are to design, build and test a brushed DC motor controller. We will interface a DC motor through a tachometer, implement background processing with periodic interrupts, and develop an integral controller. The motor will spin at a constant speed and allow the operator to specify the desired set point. We will learn how to interface a DC motor, how to measure speed using input capture, and how to implement a digital controller running in the background.

We also hope to explore and better understand the engineering design life cycle. The high-level steps to this process include Analysis, Design, Development, Testing, and Deployment. These steps will be used to describe our process of analyzing the project through a requirements document, designing the high-level software and hardware diagrams, developing the system's software, testing the full implementation, and repeating the process until a functional product can be demonstrated.

1.2. Process: How will the project be developed?

The project will be developed using the TM4C123 board. There will be two switches that the operator will use to specify the desired speed of the motor. The system will be built on a solderless breadboard and run on the usual USB power. There will be at least five hardware/software modules: **tachometer input, switch input, motor output, LCD output, and the motor controller**. The process will be to design and test each module independently from the other modules. After each module is tested, the system will be built and tested.

1.3. Roles and Responsibilities: Who will do what? Who are the clients?

Ronald Macmaster and Parth Adhia are the engineers and the Dylan Zika (TA) is the client. We have modified this document to clarify the project's requirements and terms & conditions. We can divide responsibilities of the project however we wish, but, at the time of demonstration, both of us will understand all aspects of the design.

Who will do what?

Ronald Macmaster will be responsible for drafting the requirements document and the lab report. He will also create the software design diagrams that will consist of a call graph and a data-flow graph. He will draft up the interface files for the various software modules and create program skeletons for the various driver files. **He will write the software to interface the motor through the PWM, two-switch controller, and tachometer.** Finally, he will supervise and contribute to the software development process and submit all the assignment documentation.

Parth Adhia will be responsible for drafting the hardware design documentation. This will require a schematic diagram of the external system hardware that should be built using the PCB Artist program. He will also be responsible to supervise proper assembly of the external system hardware circuit. **He is responsible for providing the tactile switches, DC Motor, TIP120, 1N914 diode, and OP2350 amplifier**. Finally, he will edit and contribute to the various project documentation and software modules as fit.

Both engineers will contribute equally to the schematic layout. Each engineer will also have a chance to contribute to the project documentation as well as the low-level software drivers. **The entire project will require both engineers to research the datasheets for the transistors, diodes, and DC Motor.**

1.4. Interactions with Existing Systems: How will it fit in? How will it connect to other systems?

The system will run on a TM4C123 protoboard, a solderless breadboard, and a DC motor of our choice. The wiring connector for the DC motor is described in the PCB Artist file **Lab10E_Artist.sch**. The motor output will flow through the two outer pins of the tachometer, and the motor's response data will flow to the controller through the two inner pins of the tachometer. It will be powered using the USB cable. If we choose to power the system through the lab station's +5V power supply, we will proceed with caution.

1.5. Terminology: *We herein define the following key terms*:

Integral Controller:

An **Integral Controller,** is a control-loop feedback mechanism that is often used in industrial systems. The controller will calculate an error value from the measured input and apply a correction based on a specifically determined output equation.

PWM:

Pulse Width Modulation is a technique to deliver a variable signal with a variable percentage of time the signal is high / on. On our TM4C123, the Pulse Width Modulator can generate a periodic square wave of a specific duty cycle and fixed frequency.

Board Support Package:

A set of software routines that abstract the I/O hardware so that the same high-level code can be run on multiple computers. An application level example would be the Java Virtual Machine.

Back EMF:

Back-EMF, or counter-electromotive force, is the induced voltage that resists the current which induced it. This is commonly noticed with inductor components which resist sudden changes in the component current.

Torque:

An object's **Torque,** can be thought of as its tendency to rotate about an axis. It can be calculated by multiplying the rotational force by the moment arm. Applying torque to an object will cause it to experience angular acceleration.

Time Constant:

The **Time Constant** is a special unit of time measurement specifically for analog circuits. Usually, one time constant is the time required for the output to reach 63.2% of the final output after an input is instantaneously changed.

Hysteresis:

Hysteresis is the condition used to describe a system that depends not only on the input, but the previous outputs as well. ex… a transducer may follow a different response curve when it is increasing rather than decreasing.

1.6. Security: How will intellectual property be managed?

The system may include software from TivaWare and from the book. No software written for this project may be transmitted, viewed, or communicated with any other EE445L student past, present, or future (other than the lab partner of course). It is the responsibility of the team to keep its EE445L lab solutions secure.

2. Function Description

2.1. Functionality: What will the system do precisely?

If all buttons are released, then the motor should spin at a constant speed. If switch 1 is pressed and released, the desired speed should increase by 5 rps, up to a maximum of 55rps. If switch 0 is pressed and released, the desired speed should decrease by 5 rps, down to a minimum of 0 rps. **These set points can be slightly modified later to reflect the true performance of our system.** Both the desired and actual speeds should be plotted on the color LCD as a function of time. **The speed plot should specify a clear range for the speed data. That will include the thresholds for minimum and maximum motor speeds.**

2.2. Scope: List the phases and what will be delivered in each phase.

Phase 1 is the preparation; phase 2 is the demonstration; and phase 3 is the lab report.

2.3. Prototypes: How will intermediate progress be demonstrated?

A prototype system running on the LaunchPad and solderless breadboard will be demonstrated. Progress will be judged by the preparation, demonstration and lab report.

2.4. Performance: Define the measures and describe how they will be determined.

The system will be judged by three qualitative measures. First, the software modules must be easy to understand and well-organized. Second, the system must employ an integral controller running in the background. There should be a clear and obvious abstraction, separating the state estimator, user interface, the controller and the actuator output. Backward jumps in the ISR are not allowed. Third, all software will be judged according to style guidelines. Software must follow the style described in Section 3.3 of the book. There are three quantitative measures. **First, the average speed error at a desired speed of 55 rps will be measured. The average error should be less than 5 rps. Second, the step response is the time it takes for the new speed to hit 55 rps after the set point is changed from 40 to 55 rps. Third, you will measure power supply current to run the system.** There is no particular need to minimize controller error, step response, or system current in this system.

2.5. Usability: Describe the interfaces. Be quantitative if possible.

There will be two switch inputs into the system. The left switch (SW0) will decrement the motor's desired rps by 5. The right switch (SW1) will increment the motor's desired rps by 5. The desired rps will never exceed the range limits of the plot.

We will use a tachometer to measure the motor speed. There are two pins on the tachometer that will be used as inputs to our motor feedback loop. **The DC motor will operate under no load conditions.**

2.6. Safety: Explain any safety requirements and how they will be measured.

Under a no load condition, the motor current will be less than 100 mA. However, under heavy friction this current could be 5 to 10 times higher. Therefore, please run the motors unloaded. Connecting or disconnecting wires on the protoboard while power is applied will damage the microcontroller. Operating the circuit without a snubber diode will also damage the microcontroller.

3. Deliverables

3.1. Reports: How will the system be described?

A lab report described below is due by the due date listed in the syllabus. This report includes the final requirements document.

The final lab report is due on Friday, November, 18th at Midnight. This report will include the final requirements document. The outline of the lab report is as follows:

A) Objectives (final requirements document)

B) Hardware Design (PCB Artist file)

DC motor and tachometer interfaces, showing all external components.

LCD and switch interfaces, showing all external components.

C) Software Design (upload your files as instructed by your TA)

Units on all software variables,
clear distinction between debugging and controller variables.

D) Measurement Data (underlined sections of the lab manual)

Motor voltage, current, and resistance measurements.

I_{BE} and I_{CE} while spinning motor.

Two screenshots of the hardware in operation.

Maximum time to execute once instance of the ISR,

Average controller error, approximate response time.

Current required to run the system, with and without the motor spinning.

E) Analysis and Discussion (short 1 or two sentence answers to the analysis questions)

3.2. Audits: How will the clients evaluate progress?
Clients will be presented with a preparation report the week before the final checkout. The lab report will be presented at conclusion of the project.

3.3. Outcomes: What are the deliverables? How do we know when it is done?
There are three deliverables: preparation, demonstration, and report. Most of our work and design will be documented in the final lab report.

2.0: Hardware Design:

Our system will interact with a DCM 408 motor element. The motor will carry no load, and will draw a maximum unloaded current of 100mA. First, period measurements for the motor will be taken and communicated through the hardware tachometer interface. The switches act as external user input to adjust the motor's speed set point. After processing within the software system's PID equation, the response is ready for output. The final output will be exerted at the TIP120 BJT base node. This will cause a current to flow through the motor and change its speed in rotations per second.

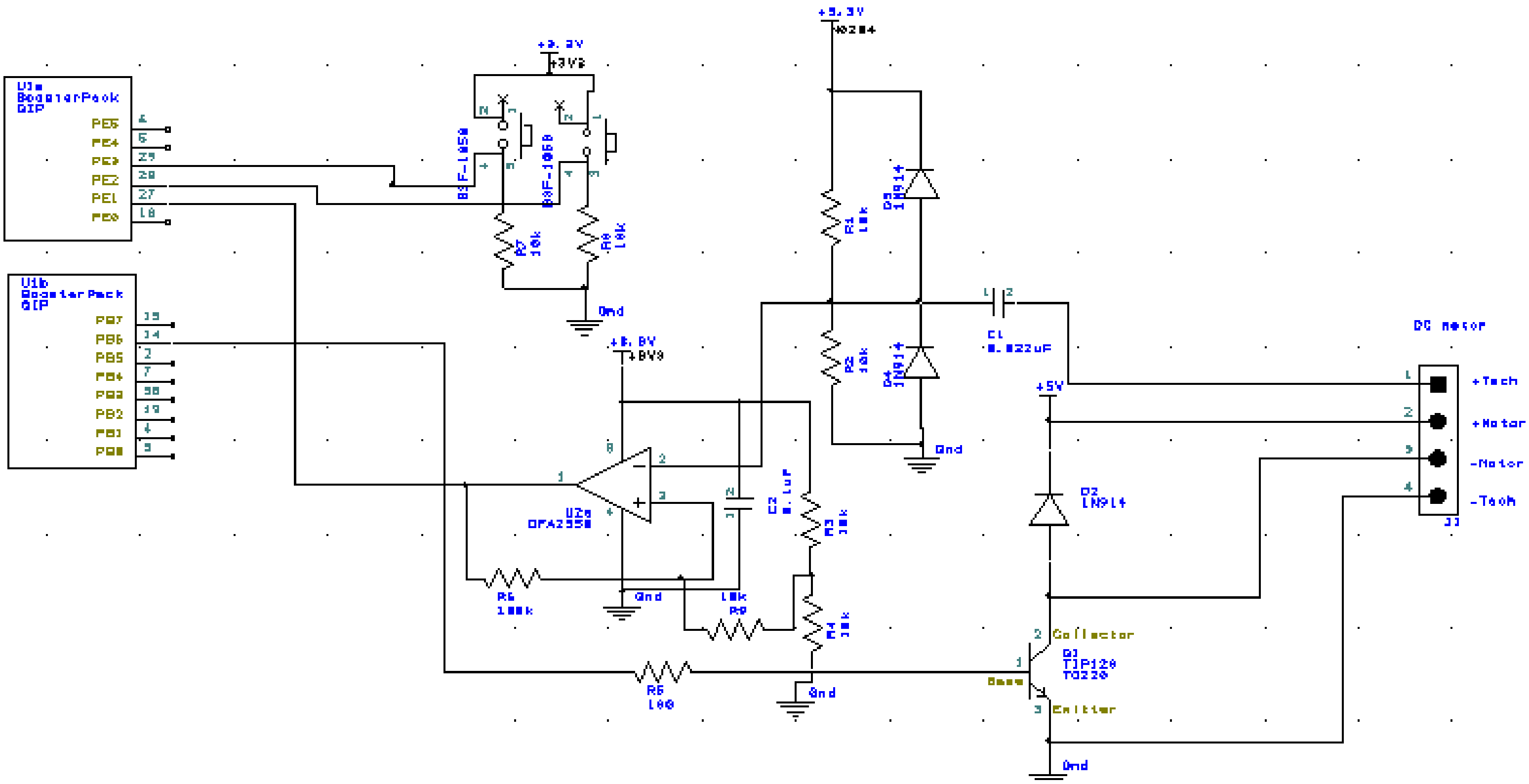


Figure 2.1: *Hardware Schematic for our DC Motor Control System. The switches utilize positive logic to implement the speed controller. The motor's tachometer is interfaced through the TIP120 and PWM. Period measurements are read from the Tachometer into the microcontroller complete the feedback loop.*

3.0: Software Design:

All of our required and additional functionality for our motor controller can be implemented using four basic modules — Switch input, Tachometer input, LCD output, and Motor control. Submitted along with this report are the interface and driver files (.h and .c) for our software modules (Motor, Tachometer, Switch, and LCD).

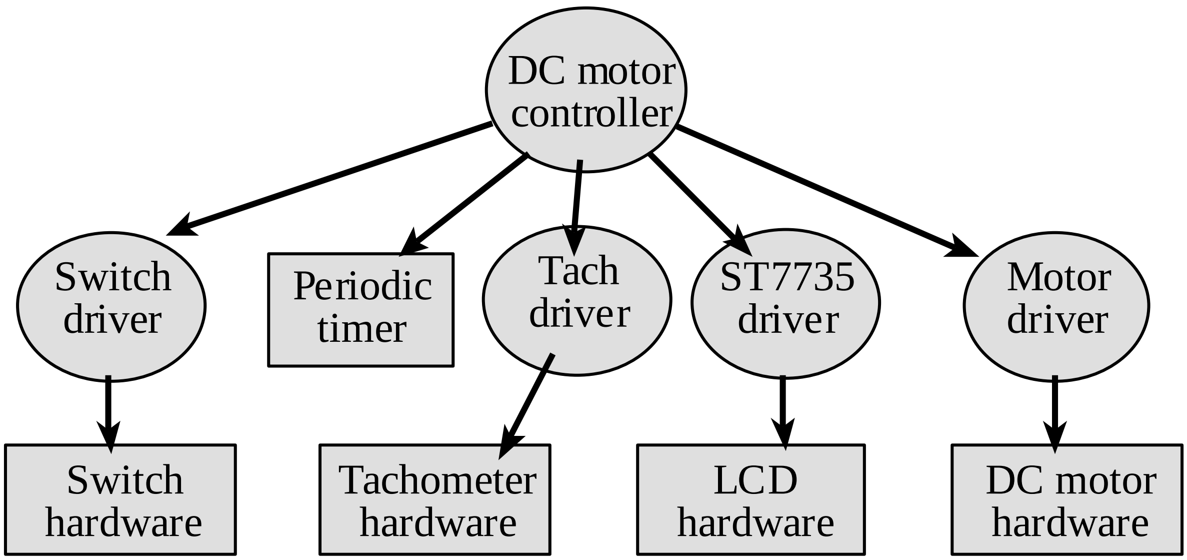


Figure 3.1: *Call graph for our DC Motor Controller. A main driver program manages hardware through secondary software modules.*

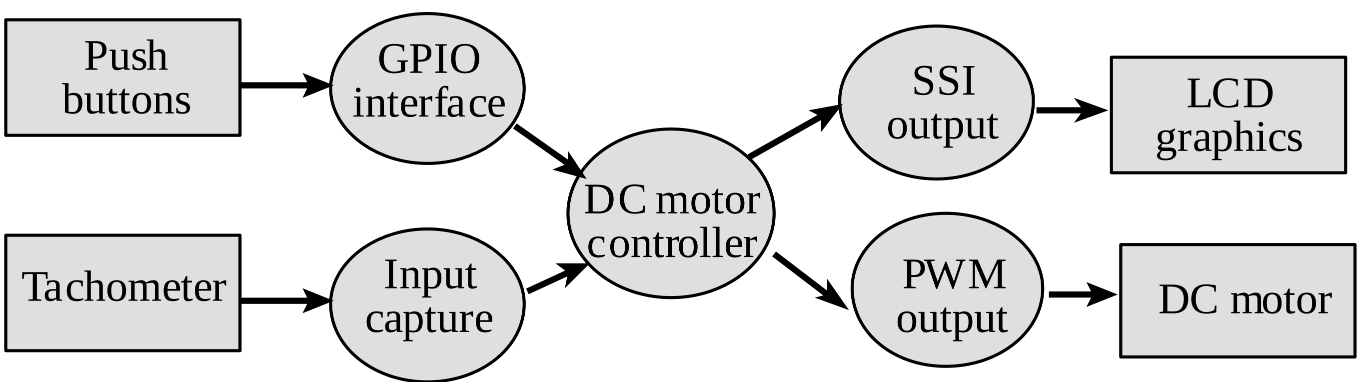


Figure 3.2: *Data flow graph for out DC Motor Controller. The input response is captured through the Tachometer hardware and software driver. The PWM then responds with motor output, creating a feedback loop for the motor interface.*

4.0: Measurement Data:

Measurements were acquired as asked for in the lab manual.

4.1: Motor R, I, and V Measurements

Motor Resistance from Ohm Meter: **45.7 Ohms**

Effective Motor Resistance: 25.8 Ohms

Bench Supply [V]	Voltage [V]	Current [A]
0	0	0.001
1	1	0.063
2	2	0.085
3	3	0.09
4	4	0.098
5	5	0.105

Figure 4.1: Motor current and voltage measurements from the bench supply.

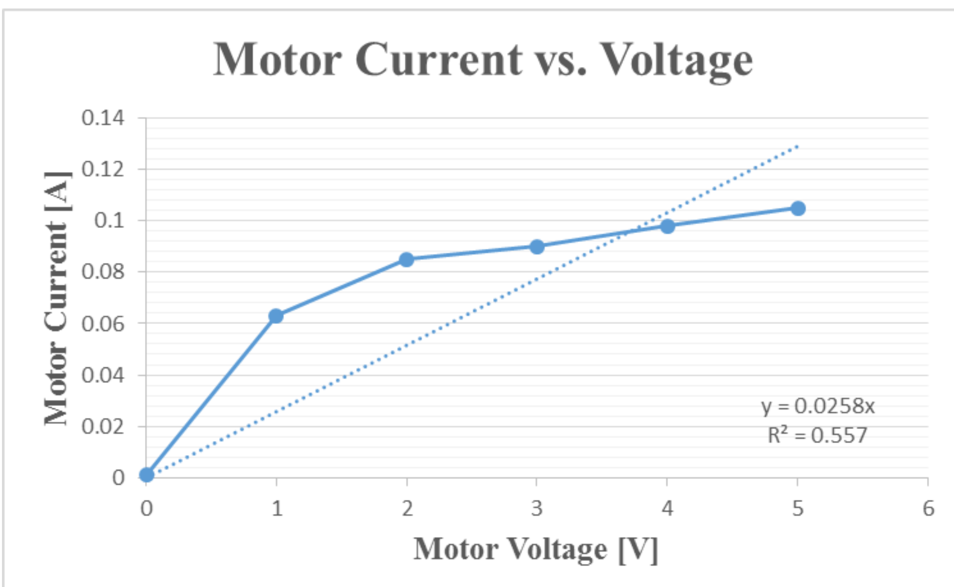


Figure 4.2: Motor current and voltage plot with trendline.

4.2: I_{BE} and I_{CE} while spinning

The motor was driven at full speed with a positive logic switch.

I_{BE} and I_{CE} while spinning: (15.04 mA, 85.86mA).

4.3: Hardware Conversions Visualized

Our Motor Controller requires an external feedback loop circuit in order to properly drive the motor. Here, we demonstrate the conversions that occur from microcontroller PWM output to actual motor voltage. We also demonstrate the conversions that occur from motor tachometer output to microcontroller threshold capture input.

Figure 4.3: Microcontroller PWM output vs. actual motor voltage. The PWM drives a pulse-modulated square wave that in turn drives an alternating current at the motor coil.

Figure 4.4: DC Motor Tachometer output vs. threshold capture microcontroller input. The tachometer wave is an unregulated, noisey and large AC wave. Our controller removes the DC bias, clips the voltages at 0 and 3.3V, and transforms it into a square wave through a threshold detector.

4.4: ISR Execution Time

Our controller operated on three interrupt service routines. The first routine updates the LCD variables and requests a redraw. This routine executes approximately 50 times a second. The second routine is an edge-triggered interrupt that logs and calculates the tachometer period from timer timestamps. The third routine is a feedback response that performs the integral controller calculations and adjusts the duty cycle of the PWM. Out of these three routines, the integral controller take the most time. It takes about **10 us** to execute.

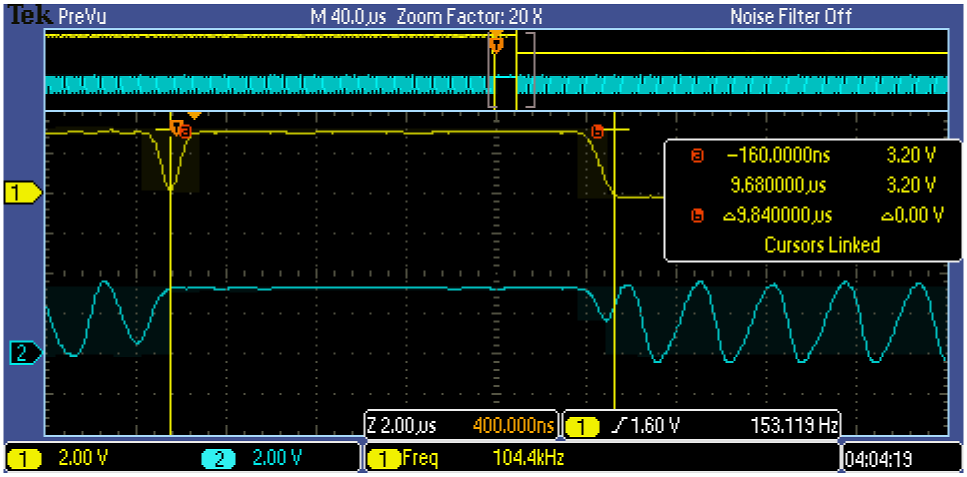


Figure 4.5: ISR Execution of our Integral Controller.

The interrupt service routine takes about 10 us to complete.

4.5: Average Controller Error and Response Time

We calculated the average controller error from 8 samples taken at a desired speed of 50 rps. The average response time (step-up) was estimated when demanding a change from 35rps to 55rps. The average setup time was estimated to be **0.875s**. Below are the tables of measurements we collected.

Motor Sample	Speed (rps)	Error
0	50	0
1	50	0
2	49.8	0.2
3	49.5	0.5
4	50.4	0.4
5	50	0
6	50.1	0.1
7	50.6	0.6

Desired Speed	50
Average Error	**0.225**
Samples	8

Table 4.1: The Average Error was calculated with a desired speed of 50rps from 8 sample measurements. Overall, the average error has a magnitude of 0.225 rps.

4.6: System Power Measurements

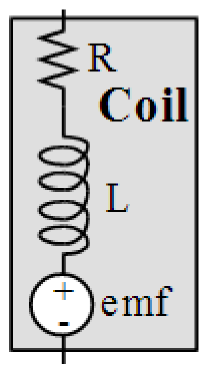
Finally, we measured the total current required to run the system, both when the motor is spinning and when it isn't. The motor controller was powered with 5V from the lab power supply bench. The system requires **83 mA** to operate without the motor. It requires **185mA** to operate with the motor.

5.0: Analysis and Discussion:

1) What is torque? What are its units?

An object's **Torque,** can be thought of as its tendency to rotate about an axis. It can be calculated by multiplying the rotational force by the moment arm. Applying torque to an object will cause it to experience angular acceleration. Its unit of measurement is the newton-meter [N*m].

2) Draw an electrical circuit model for the DC motor coil, and explain the components. Use this circuit model to explain why the current goes up when friction is applied to the shaft



Electrical Circuit model for a DC Motor Coil.

The resistance models the resisitance of the coil wire that current must flow through. The inductance reflects the inductive properties of the motor coil and its tendency to resist current. Finally, the emf reflects the back emf that is generated whenever there is a large shift in the direction of the current through the motor. When a large amount of friction is applied to the motor, the motor needs a greater current to drive the new load. **Applying a large amount of**

friction prevents the motor from overcoming inertia with its emf, so it will attempt to do so with a larger current.

3) Explain what parameters were important for choosing a motor drive interface chip (e.g., TIP120 or 2N2222). How does your circuit satisfy these parameters?

When we chose the TIP120, we considered the parameters of current gain, maximum current, and supply voltage / current. Instead of choosing something like a MOSFET (requires high voltage to activate) we needed something that could be activated with only a modest amount of current. The other choices of BJTs (PN2222 or 2N2222) could not supply enough current to power the motor. We did not want to risk burning out either of these components. The TIP120 is a great current amplifier that can supply up to 5A with a modest current gain.

4) You implemented an integral controller because it is simple and stable. What other controllers could you have used? For one other type of controller how would it have been superior to your integral controller.

An alternative to the integral controller would be an open loop controller. This does not require us to measure feedback from the tachometer. All we must do is output various duty cycle parameters based on the desired speed. It is extremely simple to implement, but it sacrifices most of the system's accuracy.

5) It the motor is spinning at a constant rate, give a definition of electrical power in terms of parameters of this lab? Research the term "mechanical power". Give a definition of mechanical power. Are the electrical power and mechanical power related?

To calculate the electrical power delivered to the motor, we simply use this formula: Power = Voltage * Current * Duty Cycle. Current only flows on the positive edges of the PWM wave, and the voltage * current factor is the traditional value for electrical power. Mechanical power is different from electrical power in how it is calculated. It is the time derivative of work, and it is equal to force * velocity or torque * angular velocity. Specific to our system, **the electrical**

energy delivered to the circuit is the sum of the mechanical energy and the thermal energy dissipated as heat.