Zachary Pope (zhp76) and Ali Kedwaii (mak3799)

Requirements document

1. Overview

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

The objectives of this project are to design, build and test a brushed DC motor controller. The motor should spin at a constant speed and the operator can specify the desired set point. Educationally, students are learning how to interface a DC motor, how to measure speed using input capture, and how to implement a digital controller running in the background.

1.2. Process: How will the project be developed?

The project will be developed using the LaunchPad. 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. The system may use the on board switches or off-board switches. A hardware/software interface will be designed that allows software to control the DC motor. 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?

EE445L students are the engineers and the TA is the client. Student A will build and test the sensor system. Student B will build the actuator and switch input. Both students will work on the controller.

Zach - motor output and controller

Ali - tachometer input, switch input, LCD output

1.4. Interactions with Existing Systems: How will it fit in?

The system will use the microcontroller board, a solderless breadboard, and the DC motor shown in Figure 10.1. The wiring connector for the DC motor is described in the PCB Artist file `Lab10E_Artist.sch`. It will be powered using the USB cable. **You must use a +5V power from the lab bench, but please do not power the motor with a voltage above +5V. Do not connect this bench supply to Vbus (LaunchPad +5V). However, you must have a common ground.**

1.5. Terminology: Define terms used in the document.

Integral controller - **sums the instantaneous error over time and gives the accumulated offset that should have been corrected previously**

PWM - **Pulse-Width Module - technique used to deliver a variable signal using an on/off signal with a variable percentage of time the signal is on**

Board support package - **set of software routines that abstract the I/O hardware such that the same high-level code can run on multiple computers**

Back emf - **voltage generated by turning off switch, due to the large change in current (dI/dt)**

Torque - **rotational force that is given by the cross product of position and force**

Time constant - **the time to reach 63.2% of the final output after the input is instantaneously increased**

Hysteresis - **dependence of the state of a system on its history**

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, **the PI controller uses the feedback from the encoder to adjust the speed of the motor until it is at the desired speed, and** then the motor should spin at a constant speed. **The feedback loop tries to correct the motor speed once every second.** If switch 1 is pressed and released, the desired speed should increase by 5 rps, up to a maximum of 40 rps. If switch 2 is pressed and released, the desired speed should decrease by 5 rps, down to a minimum of 0 rps. **The feedback loop will correct the motor speed to reflect these changes.** Both the desired and actual speeds **will be displayed numerically** on the color LCD.

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. Details can be found in the lab manual.

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 60 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 60 rps after the set point is changed from 40 to 60 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. The tachometer will be used to measure motor speed. The DC motor will operate under no load conditions.

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

Figure 10.2 shows that 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.

3.2. Audits: How will the clients evaluate progress?

The preparation and demonstration are due at the beginning of the lab periods on the dates listed in the syllabus.

3.3. Outcomes: What are the deliverables? How do we know when it is done?

There are three deliverables: preparation, demonstration, and report.