2010 International Conference on Intelligent Computation Technology and Automation

DESIGN ON A DC MOTOR SPEED CONTROL

Zhu Haishui, Wang Dahu, Zhang Tong, Huang Keming
School of Electrical Engineering & automation, Henan Polytechnic University,Jiaozuo Henan 454000, China
dahuwang@126.com

***Abstract*—In the paper,we designed and built a high efficiency, low cost DC motor controller use with a motorized golf bag. The controller takes user input to control the speed and drives the motor at that speed regardless of load. The speed control is accomplished in an operational amplifier circuit configured for proportional integral control. This provides quick, smooth motor response to the user input speed and keeps the speed regulated as the motor sees an increased load.A buck converter drives the motor, taking power from a battery built into the cart. The buck converter steps down the battery voltage to provide the necessary motor voltage for the desired speed.The controller also contains a protective element for potentially damaging torque loads. The controller allows the motor to be driven at the maximum current load of 50A for five seconds before shutting the motor down and requiring a user reset.**

Keywords-DC motor;controller;amplifier circuite;buck converter;proportional integral control

I. INTRODUCTION

For many casual golf players, carrying clubs around is tedious and degrading to the relaxation of the game. Adding a small motor to the golf bag eases that burden and allows the player to focus on the game and have fun. We selected this project due to our interests in power and motor control. There is high potential for a low-loss, low-cost DC motor speed control to be used in other applications as well.

We designed and built a DC motor controller that operated at very high efficiencies and could be built for a very low price. The controller took power from a 12V battery and used it to drive a 12V DC motor at varying speeds. Speed regulation and torque availability are priorities in most motor applications and were a driving force in our design. The controller we built takes input from the user to control the output speed of the motor and also contains built in safety features in the event that a large load is applied that the motor cannot handle.

The main goals of this project were to achieve high efficiency levels for differing motor loads, provide stable speed regulation, and create a project with low overhead.

12 V, 0 – 50 A application

Operate at 90% efficiency for 50 – 150 Watt loads

Deliver 150 W continuous, 250 W for one minute

Deliver 50 A V for 5 seconds

Automatically shut down after 5 seconds and require hard reset

<10% speed change for load change of 0 – 150 W

II. DESIGN OUTLINE

The design was broken down into different modules to simplify circuit design, construction, debugging, and integration as shown in Figure 1.

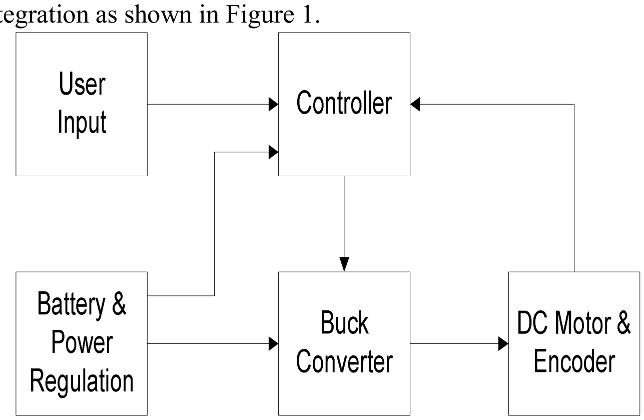


Figure 1 Motor Controller Block Diagram.

A. Battery & Power Regulation

The battery provides all of the necessary power for the motor controller. The battery will be connected to the buck converter to deliver power directly to the motor. The battery will also connect to voltage regulating devices to provide +12V, -12V, and 5V to the control module.

B. User Input

The user input to the controller consists of a potentiometer for adjusting output speed and a reset button to bring the controller back on after an automated shutdown that occurs when the current is too high.

C. Controller

The controller takes speed data from the encoder and compares it to the user input reference speed and adjusts the buck converter's duty cycle. The controller is responsible for reproducing the user input speed in the motor and keeping the speed consistent during disturbances at the output. It also monitors the current into the motor and initiates an automatic shutdown when the current hits 50A for more than 5 sec.

D. Buck Converter

The buck converter takes the DC voltage and steps it down to a lower DC voltage corresponding to the user selected speed. The stepped down voltage is then used to drive the motor. The change in voltage is determined by the buck converter's duty cycle which is managed by the controller block.

978-0-7695-4077-1/10 $26.00 © 2010 IEEE
DOI 10.1109/ICICTA.2010.795

E. DC Motor & Encoder

The DC motor takes the electrical energy from the motor controller and converts it to the mechanical work necessary to drive the golf bag cart. An encoder on the motor's shaft sends speed information back to the controller to calculate the current error and better regulate the output.

III. DESIGN PROCEDURE

A. Battery & Power Regulation Design Decisions

There was only one option for powering the system – a large battery. If our product was to be used in a mobile setting, it was the only way to go. For the circuit to function properly, the battery must be capable of providing an output of 12V and 50A. Since the normal operating range of the circuit remains between 50 and 150 Watts, the battery was sized based on the nominal operation. We would also like to provide power for 3 hours continuous operation between charges.

$$DC\ Battery\ Load\ Amps = 100\ Watts \div 12V = 8.3A$$

$$8.3A \times 3\ hours = 24.9\ Amp-hours$$

To provide power to the rest of the circuits, we used a combination of a linear regulator and an inverter. The linear regulator steps the 12V down to 5V to power the logic elements of the circuit. Since the current used to power the logic ICs is small, the power lost using a linear step-down regulator is negligible. To power the control circuit we needed a dual sided supply of ±12V. This was achieved by using a simple voltage inverter to flip the 12V input of the battery and provide the needed negative supply to the op amps.

B. Control System Design Decisions

The goal of the control system was to track the user input speed and to maintain that speed regardless of the torque applied to the motor. To do this we decided to implement a proportional integral (PI) controller. PI control allows the designer to maintain control of the time response of the system and also gives zero steady state error, which means that the actual motor speed will stabilize to the reference speed input by the user.

The first step of the design process was to obtain a motor and characterize its transfer function. Motors can be modeled as a simple one pole system shown in Eq. (1).

$$P(s) = \frac{K}{\tau s+1} \tag{1}$$

Here K is the tachometer gain and τ is the time constant of the motor. By driving the motor with a 5V step input, these characteristics were measured and then used to model the motor in Matlab. Tachometer gain, K, was found to be 1.01 and the time constant, τ, was measured as 0.0296 seconds. Figure 1 shows the measured and modeled step responses of the motor.

With an accurate motor model, the rest of the PI design could be done. The PI control transfer function is shown in Eq. (2).

$$H(s) = K_P + \frac{K_I}{s} \tag{2}$$

In Eq. (2), K_P is the proportional gain constant and K_I is the integral gain constant. Upon combing the two transfer functions we have and open loop model given by Eq. (3).

$$G(s) = H(s)P(s) = \frac{(K_P s + K_I)(\frac{K}{\tau})}{s^2 + \frac{s}{\tau}} \tag{3}$$

And a closed loop model given by Eq. (4).

$$\frac{G(s)}{1+G(s)} = \frac{\tau^{-1}(KK_P s + KK_I)}{s^2 + \tau^{-1}(1+KK_P)s + \tau^{-1}KK_I} \tag{4}$$

This 2nd order model of the control system allows easy selection of gains for desired motor behavior.

$$\omega^2 = \frac{KK_I}{\tau} \qquad 2\zeta\omega = \frac{KK_P+1}{\tau} \qquad M_p = \exp\left(\frac{-\pi\zeta}{\sqrt{1-\zeta^2}}\right) \qquad t_r = \frac{1.8}{\omega}$$

$$\begin{cases} \omega = natural\ frequency \\ \zeta = damping\ factor \\ M_p = overshoot\ percentage \\ t_r = rise\ time \end{cases}$$

To stay within the design requirement of 10% overshoot to speed, an M_P of 5% was selected to account for error in the real system. A rise time (t_r) was selected to be 15ms to give the motor a quick response. Based on those specifications the gains were calculated. K_I was found to be 422.02 and K_P, 3.93. When this controller was implemented it did not give the desired response. After tweaking the gains and re-simulating in Matlab, new values were found.

$$K = 1.01 \qquad \tau = 0.0296 \qquad K_p = 6 \qquad K_I = 400$$

Later on, it was decided that the 15ms rise time was unnecessary and it was increased to 100ms. This decision was based on the motor's size and the application: when moving a bag of golf clubs, there is no need for that kind of acceleration and puts unneeded pressure on the motor. The rise time could easily be decreased to the order of seconds, if desired, and still provide stable operation. The new rise time required that the gains be decreased. Figure 2 gives the closed loop step response of the system.

$$K_P = 2 \qquad K_I = 200$$

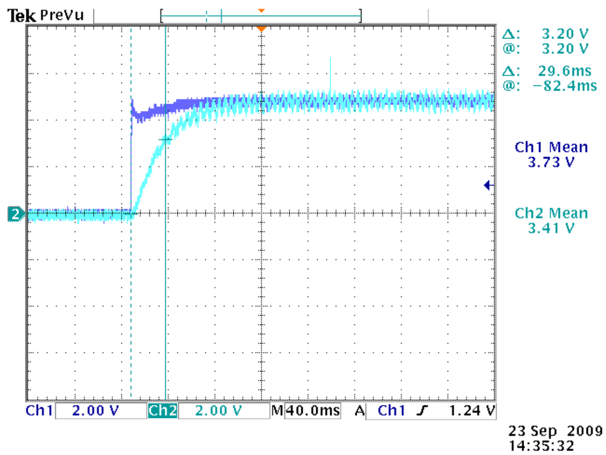
The analog op amp implementation of this controller consists of four blocks – a differential block, an amplifier block, an integrator block, and a summing block. These blocks follow the four operations shown in the Simulink diagram of Fig.3. The op amp circuits and equations are all derived from [1].

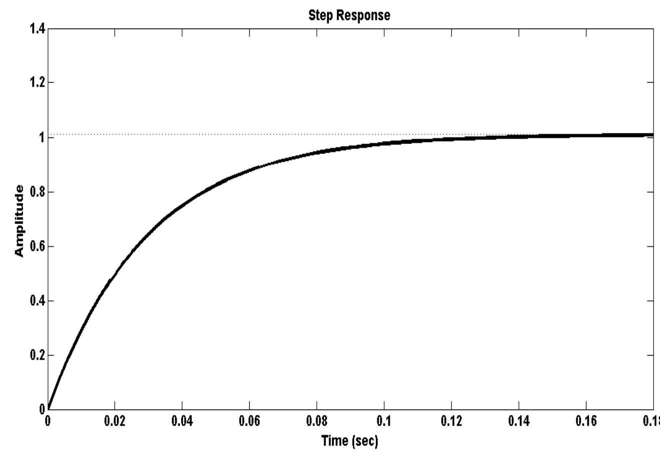
Differential Amp: The differential amplifier circuit is used to subtract the feedback voltage from the user-input speed reference voltage. Its gain properties are shown in Eq. (5).

$$R1 = R2 \qquad R3 = R4$$

$$V_{out} = \frac{R3}{R2}(V_{ref} - V_{fb}) \tag{5}$$

Figure 1 Measured and Modeled Motor Step





Response.

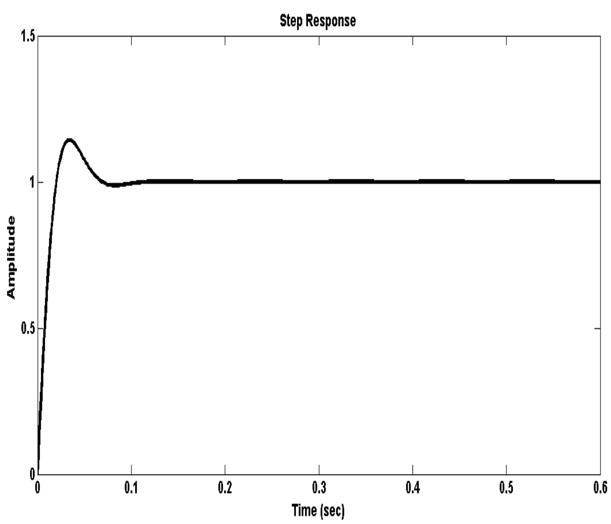


Figure 2 PI Control Step Response.

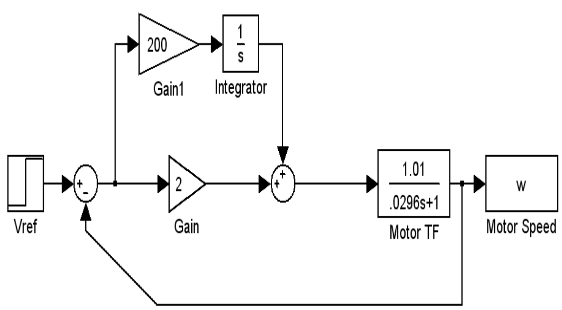


Figure 3 PI Control Simulink Model

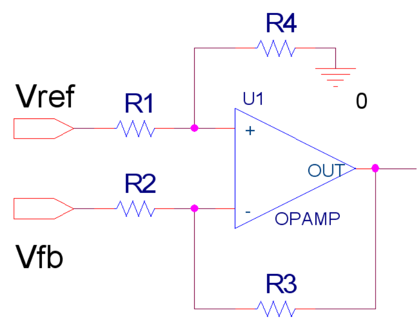


Figure 4 Op Amp Differential Circuit.

Inverting Amp:

The amplifier circuit takes care of the proportional part of the control and provides the error signal (Vref – Vfb) with a constant gain shown in Eq. (6).

$$V_{out} = -\frac{R6}{R5} V_{in} \tag{6}$$

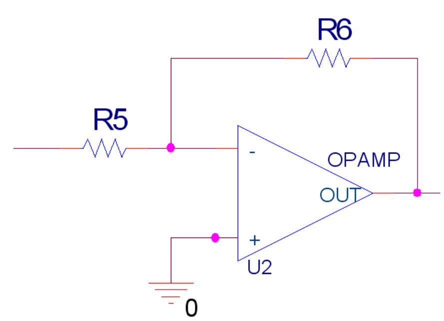


Figure 5 Op Amp Inverting Amplifier Circuit.

Integrator:

The integrator finishes the other branch of the PI control. It integrates the input voltage and also provides the error signal with a gain controlled by Eq. (7).

$$V_{out} = -\frac{1}{RFC1} V_{in}\, dt \qquad (7)$$

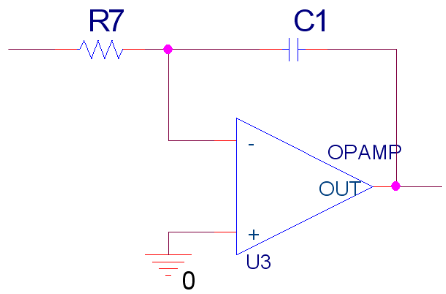


Figure 6 Op Amp Integrator Circuit.

Summer:

The summer adds the proportional control line to the integral control line to give the final output of the circuit. Its properties are given by Eq. (8).

$$R8 = R9$$

$$V_{out} = -\frac{R10}{R9}(V1 + V2) \qquad (8)$$

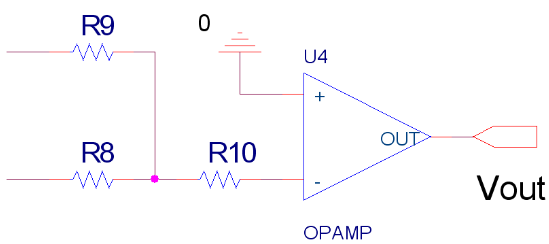


Figure7 Op Amp Summing Circuit.

The output of the op amp control circuit will be a DC voltage. To control the switching MOSFET in the buck converter, we needed to generate a square wave of varying duty cycle. This technique, known as Pulse Width Modulation, was originally going to be accomplished with a PWM generator IC. When trying to actually build the circuit and include a PWM generator, we could not achieve the range of duty cycles needed to drive the motor at the desired speeds. We decided it was in our best interest to construct a PWM circuit using an op amp triangle wave generator and a comparator. The triangle wave generator consists of the inverting amplifier and integrator circuits illustrated above in Figs.5 and 6.

C. Buck Converter Design Decisions

Efficiency was the main design challenge for the buck converter. Initially a basic buck converter schematic, shown in Fig.9, was used. A buck converter is a DC to DC step-down converter. It steps down as shown in Eq. (9) where the duty cycle was determined by the control system.

$$V_{OUT} = V_{IN} * \text{Duty} \qquad (9)$$

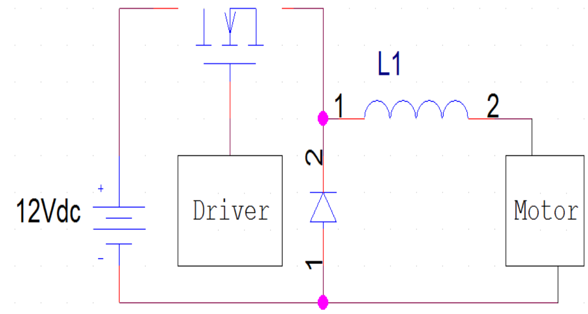


Figure 8 Basic Buck Converter Schematic

However, high current caused diode power losses to be unacceptably high. By switching the diode out for an inversely clocked MOS, power losses went down. This worked very well due to finding a MOS rated at 5mΩ and up to 50A. This required an LT1158 driver to make sure both MOS's were not on at the same time. This increased cost, but it also drastically increased efficiency.

Using a diode, current was limited to one direction. With a MOS, current could flow backwards when switched on. This allowed for regenerative braking. When the motor was decelerating, the MOS that replaced the diode would likely have a significant portion of the duty cycle. Power that would normally be lost as heat could flow back into the ground of the battery. Essentially, the motor acted as a generator. Its large inertia kept the shaft spinning during deceleration periods and pumped current back in to the circuit. With regenerative switching, this current is delivered back to the battery to keep charge longer. It does not have an effect on the circuit's overall efficiency; however, it is a nice feature to have in the device.

The inductance from the DC motor was enough to omit a separate inductor from the buck converter, which helped our design budget. The motor's inductance was almost five times the inductance originally slated for the converter.

For improved efficiency, we attempted to use multiple MOS's in parallel to reduce the on resistance even further. However, using high current MOS's at relatively low current caused additional inefficiencies. With no higher power motor to prove our theory that parallel MOS's would be better, we reverted to using a single MOS for the final design.

D. Over Torque Protection Design Decisions

One major design specification and engineered safety feature of our motor controller was the ability to supply a specified maximum current for five seconds and then shutdown the motor. If this shutdown sequence reached the five second limit, the user was required to give the system a hard reset before the motor would resume operation. Stopping the motor at this maximum current value prevents the motor from burning out attempting to drive a load it cannot handle.

The first step in designing this subproject was to come up with a way to keep track of the motors current. We decided the best way to do this was to add a very small impedance resistor in series with the motor and measure the voltage across the resistor. Because of the motor's high inductance, there was not much current ripple so the voltage across the resistor was almost pure DC. To measure the voltage we decided to come with a specialized differential amplifier that could measure a very small voltage, apply some gain, and output a larger DC voltage that we could compare to a reference voltage corresponding to the maximum current limit. If the motor current reached the maximum value, the voltage from the differential amplifier would be larger than the reference voltage and the comparator would output a logic signal that triggered a counter circuit.

The counter circuit consisted of a clock, a binary counter, a state detector, and some logic gates to control the reset and enable functions on the counter. The original design used a 1MHz oscillator and two 12-bit counters. Upon building this we had issues with the clocking of the second counter and the ICs response to the oscillator. We decided to create our own clock signal using a 555 timer. This allowed us to select the frequency of the clock and reduce to one counter. The generic 555 timer clock is shown in Fig. 9 and Eq. (10) shows the relation of frequency and component values.

$$f = \frac{0.7}{R2C1} \tag{10}$$

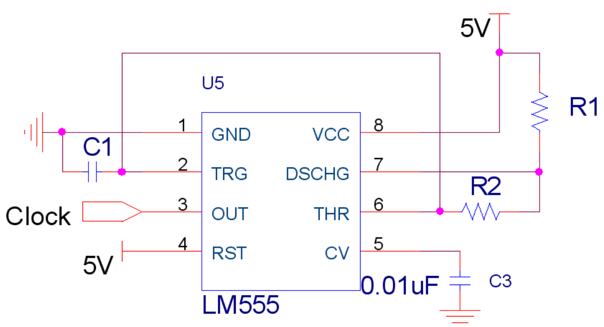


Figure9 LM555 Square Wave Circuit.

The first logic element we knew we needed was a state detector. The state detector was a NAND gate taking inputs from the bits that corresponded to the count we were looking for. Since the counter we used did not have an enable pin, we ANDed the clock signal with the state detector. Lastly, we needed to develop the reset logic. The counter needed to be reset when the hard reset button was pressed by the user and also when the current dropped below the limit before reaching the full five second count to prevent windup in the counter. The logic for the reset is shown in the Karnough map of Table 1.

Table 1 Reset Logic K-Map.

Comparator	*State Detector*	*RST Button*	*Reset Pin*
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$RESET = (Reset\ Button) + (Comparator)(State\ Detector)$$

IV. CONCLUSIONS

We would all consider our project a success. The motor was able to spin and follow the potentiometer input, and maintain constant speed despite motor load. Also, despite the numerous required specifications, our final design managed to meet them all. The control system had the output follow the potentiometer with less than 10% change in speed for loads. The current protection shut the system off after almost exactly 5 seconds of maximum current.

The only part that was not testable for specifications was the buck converter. For the testing ranges we covered, we had higher than 90% efficiency, but we did not have a 50A motor to test the range in the specifications. If we had a motor, we could have made even better conclusions about our efficiency and also the impact of using a second MOS in parallel. The switching components we selected were rated for the 50A load we originally expected. Since we could not operate in this range, the MOSFETs were being driven at the bottom of their load range and therefore were not as efficient as we expected them to be. We believe that driving our system at higher loads would show improved efficiencies.

Moving the entire circuit onto a PCB could have further improved our efficiency as well. A noticeable portion of our efficiency was likely lost to wires. As well as resistance, wires can have significant stray capacitance or inductance. Using a PCB also gives obvious packaging benefits. A PCB would have been necessary for 50A motor testing, as the breadboard we were given had not been tested over 30A.

Another benefit of using a PCB would be the separation of analog, digital, and motor grounds. Lots of digital noise from the counter circuit and PWM was passed into the controller, as well as noise from the motor revolutions. Also, due to regeneration, we had a significant amount of current flowing through ground. A PCB would have allowed us to utilize star grounding techniques to keep from sharing grounds with each subsystem and allowing potential to build up across the grounding plane. Keeping subsystems isolated would have greatly reduced some noise issues we experienced while testing.

REFERENCES

[1] R. Mancini, *Op Amps for Everyone*, Texas Instruments, 2002.

[2] Fairchild Semiconductor, “Data Sheet LM78XX/78XXA 3 Terminal 1 A Positive Voltage Regulator”, [Online Document], March 2008, http://www.fairchildsemi.com/ds/LM/LM7805.pdf.

[3] Intersil, “Data Sheet ICL7660S Super Voltage Converter”, [Online Document], March 2008, http://www.intersil.com/data/fn/fn3179.pdf.

[4] PCB Heaven, “Triangle Wave Generator”, July 2009, http://pcbheaven.com/circuitpages/Triangle_Wave_Generator.

[5] Linear Technology, “Data Sheet LT1158 Half-Bridge N-Channel Power MOSFET Driver”, [Online Document], 1994, http://www.linear.com/pc/downloadDocument.do?navId=H0,C1,C1003,C1142,C1114,P1464,D2064.

[6] Analog Devices, “Data Sheet AD626 Low Cost, Single-Supply, Differential Amplifier”, [Online Document], 2003, http://www.analog.com/static/imported-files/data_sheets/AD626.pdf.