

# PID Motor Control

EE 554

Henry Godman  
Tommy Hall  
Matthew Paiz

November 5, 2010

## Introduction

The purpose of this project was to implement an embedded control system using an Altera DE0 FPGA. The control system would be responsible for controlling the rotor speed of a DC motor. The motor would be adjusted to a reference speed provided by the user of the the control system. The motor was to be driven using a PWM implemented on the FPGA. The input capture of the NIOS II soft-core would be responsible for collecting feedback from the motor and reporting any deviations from the reference value.

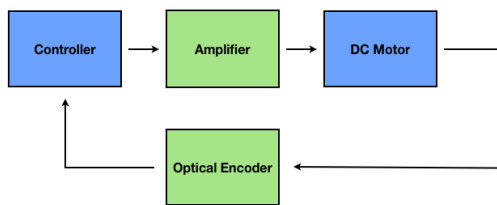


Figure 1: Control System Diagram

## Background

An important characteristic of a DC motor is its time response. It is vital to understand how quickly a motor responds to a given input and how long it takes to reach a steady state. To

model this behavior, it is best to use a generic equation of a first-order system.

$$\frac{G}{s\tau + 1}$$

The value for  $\tau$  can be found through experimentation. After it has been discovered, it can be used to solve for the other following variables when  $\zeta = 0.7$  and  $K_d = 10$ .

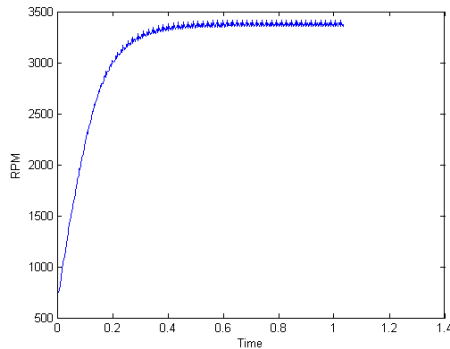
$$\begin{aligned}\omega_n &= \frac{1}{\tau\zeta} \\ K_p &= 2K_d\zeta\omega_n \\ K_i &= \omega_n^2 K_d \\ T_i &= \frac{K_p}{K_i} \\ T_d &= \frac{K_d}{K_p}\end{aligned}$$

After calculating these values, it is possible to implement a Proportional Integral Derivative Controller (PID) using the following equation.

$$\begin{aligned}u(t_k) &= K_p [ \dots \\ &e(t_k) \left( 1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t} \right) + \\ &e(t_{k-1}) \left( -1 - \frac{2T_d}{\Delta t} \right) + \\ &e(t_{k-2}) \left( \frac{T_d}{\Delta t} \right) ] + \\ &u(t_{k-1})\end{aligned}$$

## Motor Characterization

The time constant  $\tau$  was found by inputting a positive reference voltage into the motor for a short amount of time. The output of the optical sensor can be connected to the input capture of the DE0. By measuring the time difference between each successive rising edge of the square wave, the RPM measurements were found. This data was printed to the J-Tag interface and paired with a time-stamp. After recovering the RPM-timestamp pairs, the data was plotted in MATLAB (Figure 2). By analyzing the resulting curve and finding the  $\frac{2}{3}$  rise time, the value for  $\tau$  was found to be 1.40 ms. This experiment was repeated while adjusting the duration of the positive voltage input and averaging the resulting  $\tau$  values together. After calculating



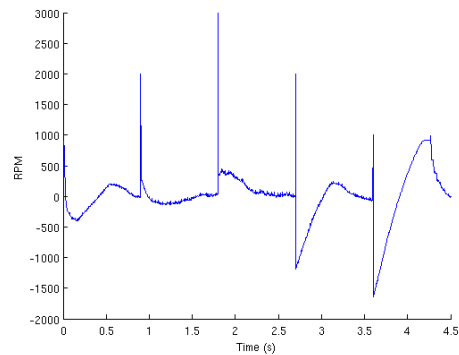
**Figure 2:** DC Motor Characterization

$\tau$ , it was possible to calculate the other following values.

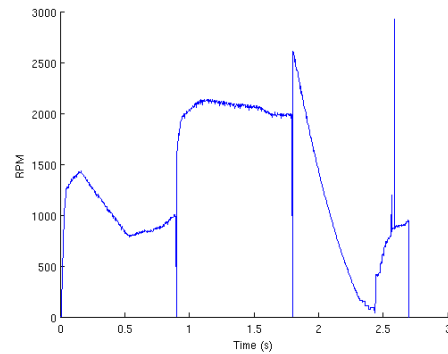
$$\begin{aligned}\omega_n &= 1020.41 \\ K_p &= 1020.41 \\ K_i &= 14285.71 \\ T_i &= 14.0 \\ T_d &= 0.0098\end{aligned}$$

## PID Implementation

Using the equation for the PID and the calculated constants, it was possible to implement the PID equation in the C programming language. The equation would keep track of both error (Figure 3), and output RPM (Figure 4). The plots below show the response of the control system while adjusting the value of the reference RPM. It can be seen from the plots that over time, the total error in the system would decrease to a value close to zero. Consequently, the output RPM would settle at the desired reference value.



**Figure 3:** Error Signal



**Figure 4:** RPM Response