Laboratory: Introduction to Mechatronics

Instructor TA: Edgar Martinez Soberanes
(eem370@mail.usask.ca)
2017-03-30

Lab 6. Closed Loop DC Motor Position Control.
Lab Sessions

- Lab 1. Introduction to the equipment and tools to be used in the lab, which include the development board (PICDEM 2 Plus), the microcontroller (PIC 16F1937), and the USB Oscilloscope (Analog Discovery).


- Lab 3. Experiment 7: LED Control and Interruptions.

- Lab 4. Experiment 10: Stepper Motor Motion Control.

- Lab 5. Experiment 11: DC Motor Speed Control Using PWM.

Lab 1. Outline

- Objective.

- Lecture
  - Open Loop Speed Control.
  - Closed Loop control (PID).
  - Encoder.

- Time for the experiments.
Objective

- Objectives
  - Implement a closed loop position and speed control of a DC motor using a position feedback sensor, PWM output signals, and the PIC Microcontroller.
The open loop system does not have feedback, therefore if the load changes on the motor, the motor speed will change. The control unit cannot command the driver to increase or decrease the power to the motor, as it has no knowledge of the speed change induced in the motor, by the change in load.

PWM Speed control without feedback is an open loop control system.
Pulse Width Modulation (PWM)

Duty Cycle is the percentage time that the signal is in the ON state versus the OFF state.

\[
Duty\ Cycle\ % = \left( \frac{\text{Pulse\ Width}}{\text{Period}} \right) \times 100
\]
The close loop system has feedback. So if the motors speed decreases due to an increase in load, the control unit could command the driver to increase the power to the motor, keeping a constant speed.
Proportional Integral and Derivative Control (PID)

- A PID control system uses 3 gain constants, which are multiplied with the Error Value. The proportional value gain constant is referred to as Kp, the integral value gain constant as Ki and the derivative value gain constant as Kd.
The Kp constant value determines how large the gain response will be to the current error value. A high proportional gain constant will induce a large change in the output, often overshooting the set point level.

The integral control is used to bring long term precision to the control loop. It is proportional to both the magnitude and duration of the error. Using Integration allows for the average error to be calculated, as the sum of the instantaneous error over time provides an accumulated offset. The accumulated error is then multiplied by the Ki gain constant.

Differentiation is used to determine the rate of change, by using the instantaneous error value and the previous error value. The derivative term can be used to predict system behavior, improving settling time.
The control unit will have some reference set point or threshold levels, that it is aiming to keep the motor speed at or between. If a reference set point is used the new feedback signal could be used to generate an error value in the following way:

\[
\text{Error Value} = \text{Set Point} - \text{New Feedback Value}.\]

The ultimate goal for any control system is to respond as quickly as possible, without excessive overshoot and oscillations around the set point.
Incremental Encoder

- An encoder is an electrical mechanical device that converts linear or rotary displacement into digital or pulse signals. The most popular type of encoder is the optical encoder.
PID in the PIC Microcontroller

- For Speed control we need:
  - One PWM output signal
  - One input signal from the encoder

- For Position control we need:
  - Two PWM output signals
  - Two input signals from the encoder
// Assume
// xd - the desired position variable, programmed.
// vd - the desired speed variable, programmed.
// x - the measured position
// v - calculated speed based on measured position
// Kp - proportional gain of the PD control algorithm: a constant.
// Kd - derivative gain of the PD control algorithm: a constant
// Ki - integral gain control
// ui - integral control

// The PD closed loop control position control algorithm

PWM_Out = Kp * (xd - x) + Kd * (v - vd)

// If the objective is only to control speed, but not the
// position, then the PWM output calculation should not be
// function of position information, i.e. a proportional
// closed loop speed control algorithm.

PWM_Out = Kd * (v - vd)

// PID algorithm for position control loop
// Initialize integrator term on startup;
// Set "first_call = true;" on startup.

if (first_call == true)
    ui = 0.0;
    first_call = false;
endif

ui = ui + Ki * (xd - x)

PWM_Out = Kp * (xd - x) + Kd * (v - vd) + ui;
Circuit
Thanks