

Hands-on Lab

PID Implementation



Motorized Tethered Cart: Experiment Platform

The motorized tethered cart is an experimental platform that serves to demonstrate control system design. Cruise control is a familiar feature often found in cars. Speed is monitored and regulated despite highway slope and cargo weight.

This lab implements cruise control on the motorized tethered cart. The pivot's optical encoder is monitored to measure cart velocity. PID control is used to regulate the cart's velocity despite loads.

In a previous lab, voltage was applied and cart velocity was acquired. The resulting first order response revealed the system rise time and steady-state gain. These values were then used to form the open-loop transfer function.

The LabVIEW Control and Simulation Toolkit used this transfer function to simulate open-loop response. Since the plant is linear, one can calculate the voltage needed to achieve a desired velocity. However changes in slope (e.g. mat has hills and valleys) and mass (e.g. loading) will change the plant and the open-loop transfer function is no longer accurate. One has no recourse but to identify the mapping that relates voltage input to cart velocity output. Such open-loop control becomes futile because slope and mass (e.g. consuming fuel changes load weight) are constantly changing. To overcome such futility, closed-loop control is used. Changes in desired speed are monitored to amplify or attenuate voltage input.

Concept 1: Effect of Load on Open-Loop Step Response

Step 1: Repeat open-loop step response (no load) and capture steady-state response

In a previous lab, code for `cartOpenLoopVelocity1_0.vi` was created. The front panel and block diagram are shown in **Figures 1-1A** and **1-1B** for reference.

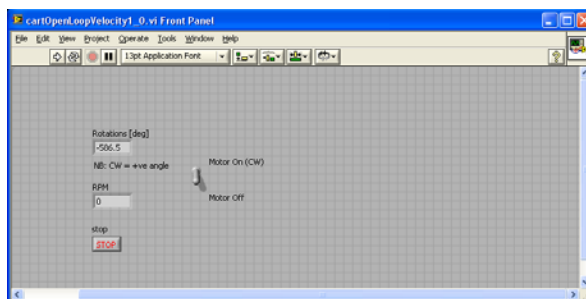


Figure 1-1A: Step Response Front Panel

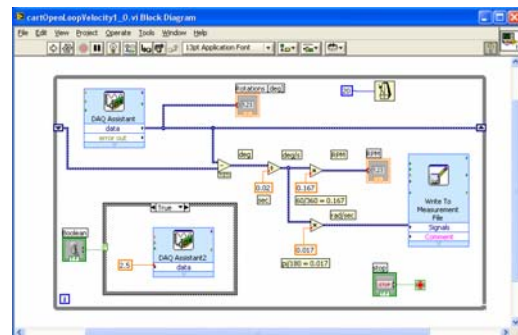


Figure 1-1B: Block Diagram

With a 2.5 Volt step input, cart velocity was captured every 0.020 seconds. The resulting Excel plot is shown in **Figure 1-2**

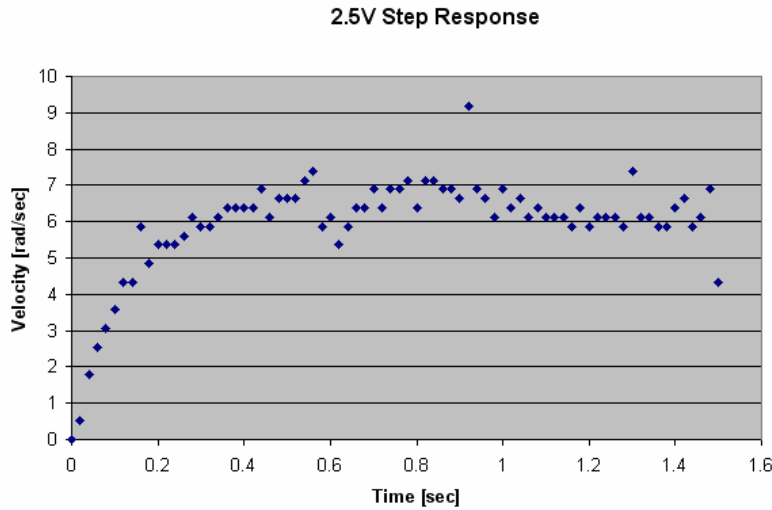


Figure 1-2: Cart's Open-Loop Step Response

From Figure 1-2, one can identify that the system's rise time $t \approx 0.11$ sec and steady-state velocity $Y_{ss} = 6.5$ rad/sec. The resulting open-loop transfer function for a voltage input $F(s)$ and cart velocity output $Y(s)$ is:

$$\frac{Y(s)}{F(s)} = \frac{2.60}{1 + 0.11 \cdot s} = \frac{23.63}{s + 9.09}$$

Step 2: Model the open-loop transfer function and simulate step input response

Again, in a previous lab, the LabVIEW Control and Simulation toolbox was used to create `cartOpenLoopStepResponseSimulation.vi`. Make sure the Step Size is set for 0.020 seconds. This value matches the sampling time used in the experiments. The resulting simulated step response is given in **Figure 2-1**.

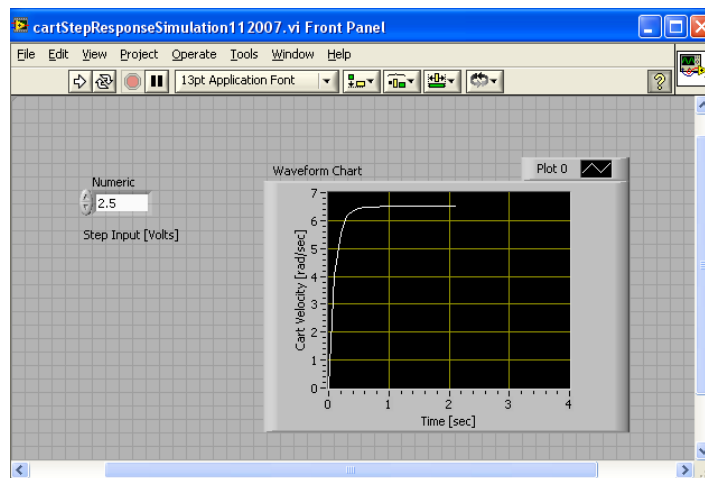


Figure 2-1: Simulation of cart open-loop step response. NB: looks similar to Figure 1-2

Step 3: Load cart with plates and repeat step response experiments

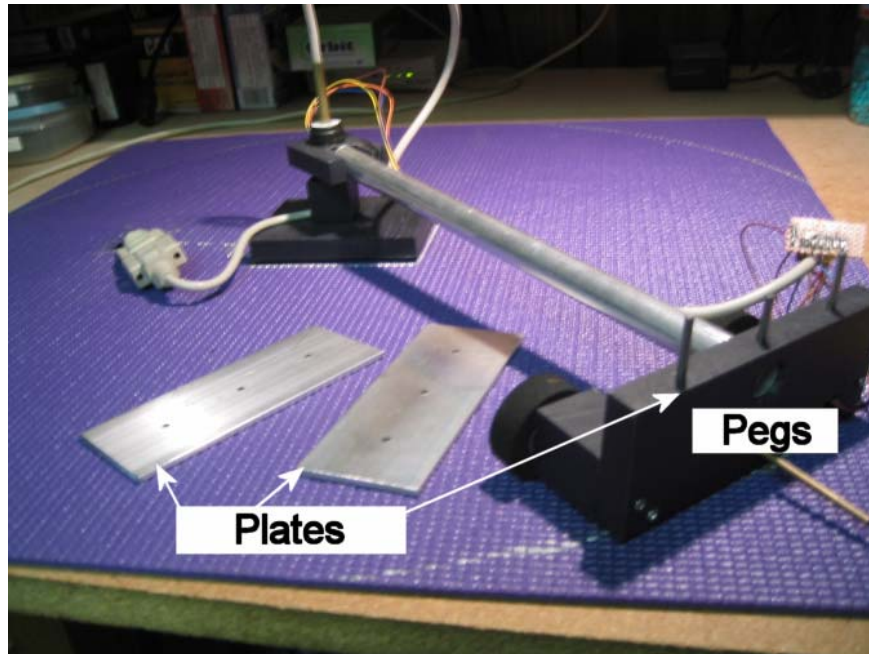


Figure 1-3: Load cart by fitting plates over the pegs found on cart

Executing `cartOpenLoopVelocity1_0.vi` results in step responses shown in **Figure 1-4A** (1 plate loading) and **Figure 1-4B** (2 plate loading). The net effect is that the steady-state velocity changes due to load.

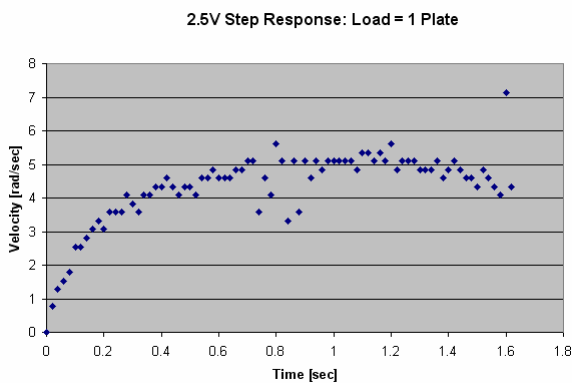


Figure 1-4A: Step response for 1 plate

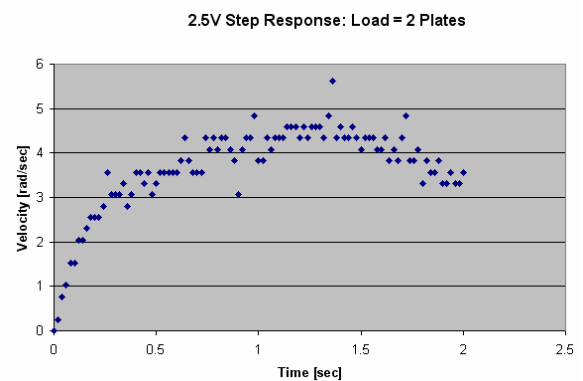


Figure 1-4B: Step response for 2 plates

Exercise 1:

1-1 Plot the step response plots for a loaded cart with 1 plate (see Figure 1-4A) and 2 plates (see Figure 1-4B)

Concept 2: PID Simulation

The LabVIEW Control and Simulation Toolbox can be used to implement and simulate PID.

Step 1: Create the following Front Panel (**Figure 2-1A**) and Block Diagram (**Figure 2-1C**). Note: set the Step Size to 0.020 seconds (**Figure 2-1B**). Save as `cartPidSimulation1_0.vi`

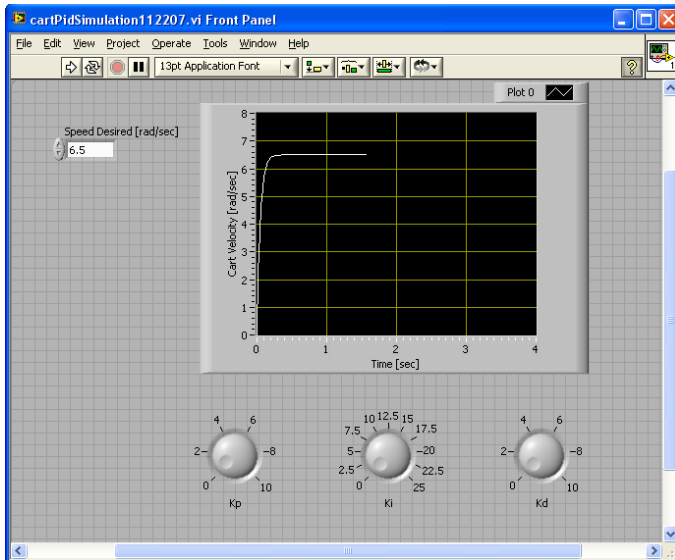


Figure 2-1A: PID Simulation Front Panel

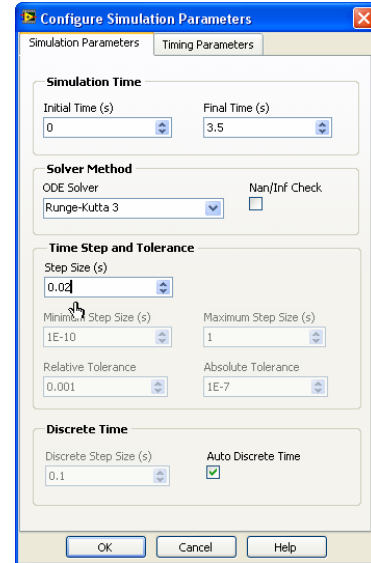


Figure 2-1B: Simulation Parameters

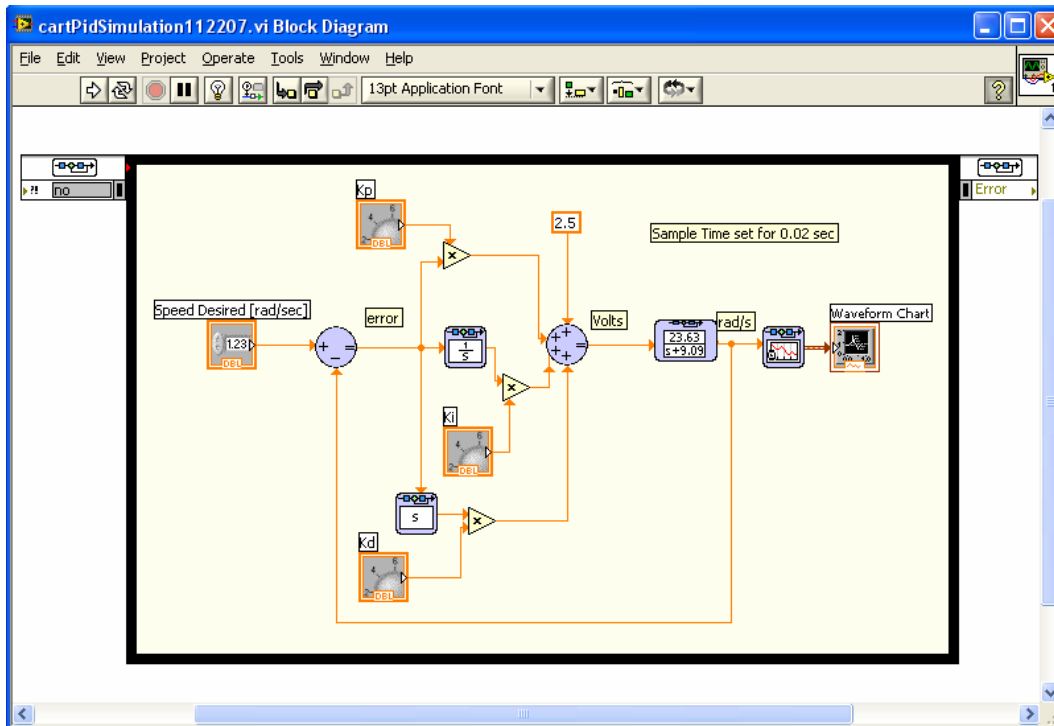


Figure 2-1C: PID Simulation Block Diagram

Step 2: Determine Gain Limits and Deadzoning

Execute the simulation and adjust gains to achieve a desired response and avoid instability. Observe that derivative gain results in a growing oscillatory response. Such instability must be avoided when implementing a real-world PID.

The USB-6211 analog outputs are limited to ± 10 Volts. In simulation, the PID controller may calculate values for applied voltage that are greater than these limits. In practice, the applied voltage will be clipped (also known as **deadzoning**) and the cart will not be getting the voltages that simulations calculate. As a result, one can add an additional waveform graph to display the applied voltage in simulation. This is shown in **Figure 2-2A** and **Figure 2-2B**.

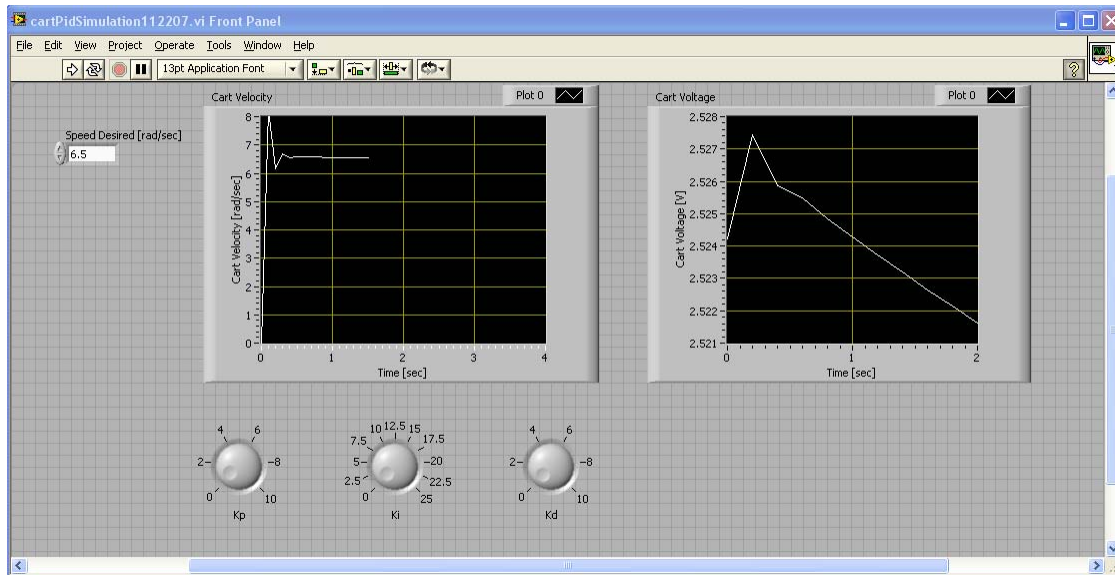


Figure 2-2A: PID Simulation Front Panel. Voltage applied to cart is also displayed

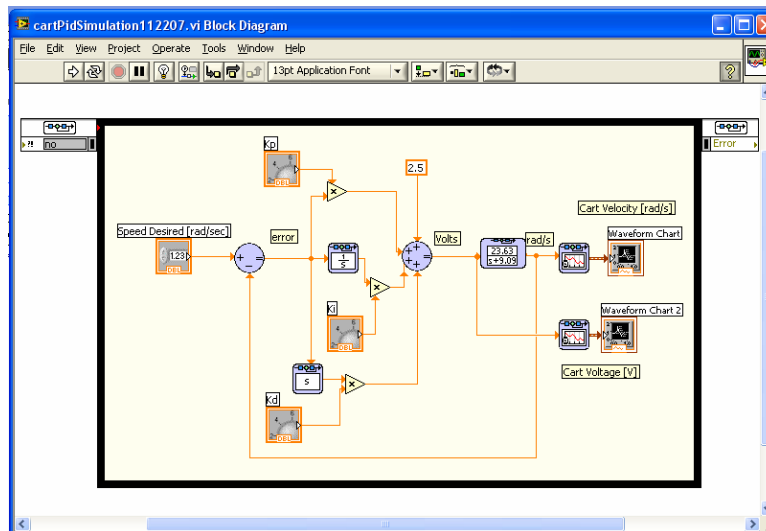


Figure 2-2B: PID Simulation Block Diagram. Voltage applied to cart waveform added.

Exercise 2:

- 2-1 Set the [P, I, D] gains to [0.5, 0, 0]. Simulate and screen-capture the response. Identify the rise-time and steady-state response
- 2-2 Set the [P, I, D] gains to [0.5, 0.2, 0]. Simulate and screen-capture the response. Identify the rise-time and steady-state response

Concept 3: PID Implementation

Basic control elements like adders and shift registers are used to implement PID experimentally.

Step 1: Implement the Front Panel and Block Diagram. Save as `cartPid1_0.vi`

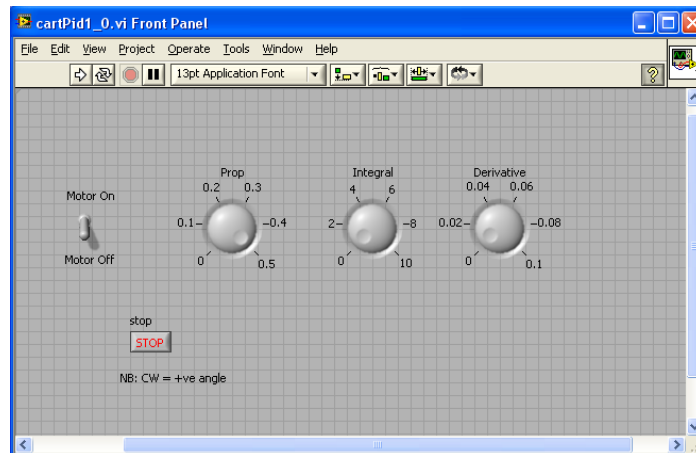


Figure 3-1A: PID Front Panel (file: `cartPid1_0.vi`)

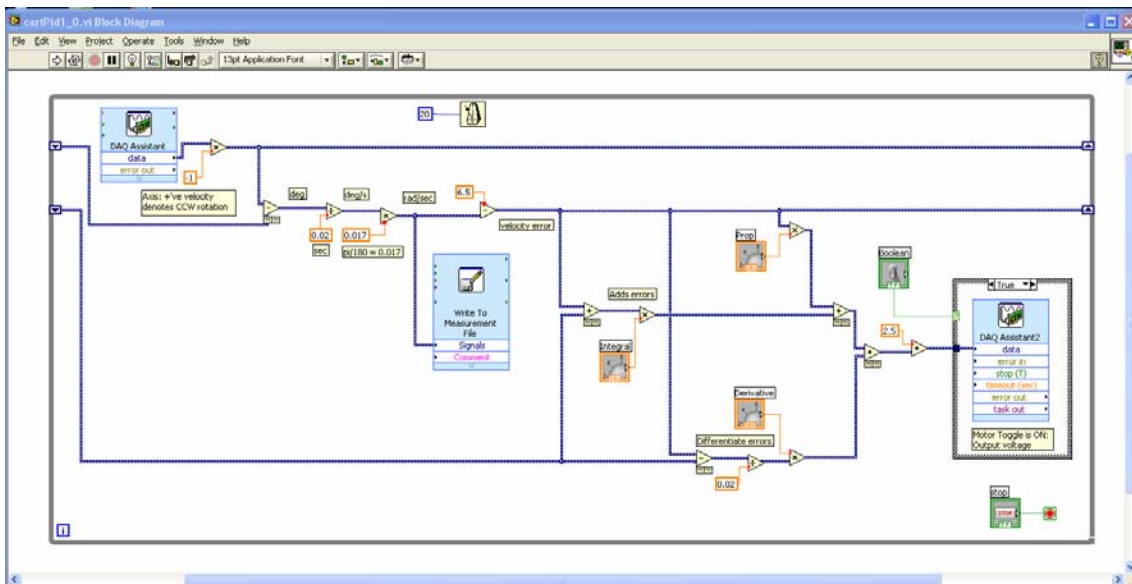


Figure 3-1B: PID Block Diagram (file: `cartPid1_0.vi`)

The case structure in **Figure 3-1B**, uses the toggle switch (see Front Panel in **Figure 3-1A**). When toggled `TRUE`, the `DAQ Assist` applies the calculated output voltage on Analog Channel 0. If toggled `FALSE`, 0 Volts is applied and the cart does not move.

Exercise 3:

- 3-1 Execute the program. Tune the [P, I, D] gains to [0.5, 0, 0]. Toggle the motor switch to **ON** and let the cart rotate 2 to 3 times. Toggle **OFF** and then kill your program. Plot the resulting velocity data and compare to the screen image captured in Exercise 2-1.
- 3-2 Execute the program. Tune the [P, I, D] gains to [0.5, 0.2, 0]. Toggle the motor switch to **ON** and let the cart rotate 2 to 3 times. Toggle **OFF** and then kill your program. Plot the resulting velocity data and compare to the screen image captured in Exercise 2-2.
- 3-3 Explain any discrepancies between the response plots found in simulation and experiments above.
- 3-4 Execute the program. Tune the [P, I, D] gains to [0.5, 0.2, 0.01]. Toggle the motor switch to **ON**. The cart will likely race quickly. Toggle **OFF** and then kill your program. How does the cart response consistent (or not consistent) with simulation?