American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS)

ISSN (Print) 2313-4410, ISSN (Online) 2313-4402

© Global Society of Scientific Research and Researchers

http://asrjetsjournal.org/

Modelling of Dynast Simulator to Controlling Speed in a DC Motor using Proportional Integral Derivative Controller

James Agajo[a*], Okoro Patience N.[b], Akande Akinyinka Olukunle[c], Folorunso T. A.[d]

[a,d] Federal University of Technology Dept. of Computer Engineering, P.M.B 65 Minna Nigeria

[c]Federal University of Technology, Dept. of Electrical/Electronic Owerri Ino State Nigeria

[b]Scientific Equipment Development Institute Enugu

[a]Email: agajojul@gmail.com

[c]Email: akandeoluk@gmail.com

[d]Email: funso.taliha@futminna.edu.ng

Abstract

Modelling of DYNAST simulator to controlling speed in a DC motor using proportional integral derivative controller is work aimed at designing, analyzing, and modeling of a DC motor speed control using Proportional Integral Derivatives Controller (PID). It entails how a PID controller can be used to achieve a desired speed response in a dc motor speed control. It presents a detailed simulation of how this PID controller and technology is added to the DC motor speed control with their varying parameters for desired response. Using the Dynast Shell Simulator shows the hardware and it sub system in their simulated platform without the use of complex equations to achieve your aim. This paper relates the place of PID technology in the present control discipline, how it reduces and eliminates errors in the control processes using Dynast Shell Simulator, it came up with a designs of dc motor speed control with and without PID. The work demonstrated that with proportional band at k_p=100, an k_i=200, the controller output is proportional to the error or a change in measurement (depending on the controller).

***Keywords**:* PID; DC Motor; DYNAST SIMULATOR; Speed Control.

----

* Corresponding author.

E-mail address: agajojul@gmail.com.

1. Introduction

In spite of the development of power electronics resources, the direct current machines are becoming more useful as they have found wide application, that is, automobile industry (electric vehicle), weak power using battery system (motor of toy), the electric traction in the multi-machine systems, etc.

The speed of DC motor can be adjusted to a great extent so as to provide easy control and high performance. There are several conventional and numeric controller types intended for controling the DC motor speed at its executing various tasks [3]. PID which simply stand for Proportional Integral Derivative Controller which is used in control processes to eliminate and reduce errors.This errors represents the difference between where you want to go and where you are actually.

Errors is thus defined as the difference between the set point and measurement. PID controllers are widely used in industrial plants because it is simple and robust. Industrial processes are subjected to variation in parameters and parameter perturbations, which when significant makes the system unstable. So the control engineers are on the look out for automatic tuning procedures.

In this work , a scheduling PID tuning parameters strategy for a DC motor speed control is proposed. Motors come in many sizes and types, but their basic function is the same. Motors of all types serve to convert electrical energy into mechanical energy [1].

D.C. motors are motors that run on *Direct Current* from a battery or D.C. power supply. Direct Current is the term used to describe electricity at a constant voltage. When a battery or D.C. power supply is connected between a D.C motor's electrical leads, the motor convert’s electrical energy to mechanical work as the output shaft rotates. PID Controllers are designed to eliminate the need for continuous operator attention. Cruise control in a car and a house thermostat are common examples of how controllers are used to automatically adjust some variable to hold the measurement (or process variable) at the set-point. The set-point is where you would like the measurement to be. Error is defined as the difference between set-point and measurement.

Error = (set-point) - (measurement).

The variable being adjusted is called the manipulated variable which usually is equal to the output of the controller. The output of PID controllers will change in response to a change in measurement or set-point. This work will enhance the reduction of overshoot, reduction of settling time and the steady state trasient response of the control signal. PID controller came to existence because it eliminate these error or reduce them to beareable extent. These error reduction led to the design of DC motor speed control for effective plant control. The design and implementation of DC motor speed control using PID technology is achieved by dynast shell simulator.

2. Section of dynast shell solver

The DYNAST simulation system consists of two separate parts. DYNAST Solver and a DYNAST working environment. DYNAST Solver is composed of several sections sharing common data. The section SYSTEM reads in the system-model description in the form of a set of algebro-differential equations, a block diagram, a multipole diagram, or in a form combining freely these approaches.

DYNAST input-language is composed of statements coded in ASCII characters and both upper and lower case letters may be used in them. DYNAST, however, is not case sensitive and converts all the letters into the upper-case ones. Each of the statements is terminated by the semicolon character ';'. A statement may continue on several lines, and there may be several statements placed in one line. The maximum number of characters in one line including spaces is 80.

3. System design and analysis

In Figure 1 the block diagram gives an overview of a DC motor speed control, also a detailed clue on how it runs[10].

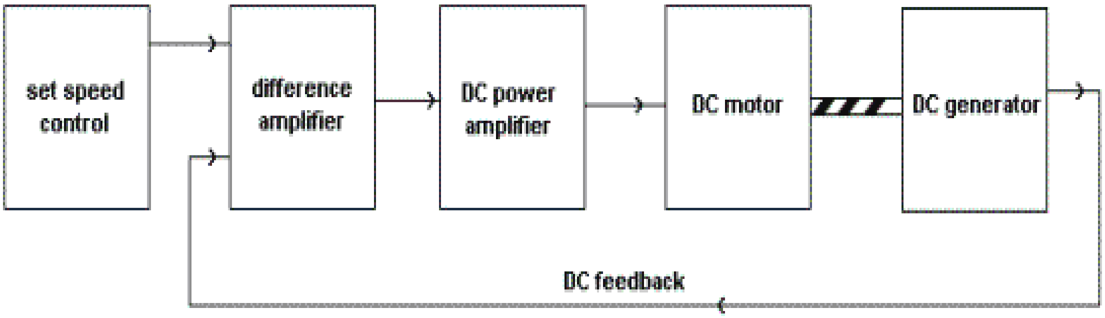


Figure 1: Simple Speed Control Block Diagram

The striped thing between the motor and the dc generator is a shaft which couples them together mechanically. The motor drives the generator (tachometer) via this shaft. The set speed control provides a dc voltage, say 12 volts for maximum speed and zero for stationary. This could be a potentiometer providing any voltage in a range from zero to +12 volts. The differential amplifier will amplify any difference between its two input voltages. If the motor is stationary and the speed control is moved from zero to half speed then, since the tachometer is not rotating and not producing an output voltage, there will be a differential in voltages at the two inputs of the difference amplifier. Therefore there will be an output voltage from the amplifier. Since this voltage is not high enough in value to drive the motor, it is increased in amplitude by the dc amplifier. A dc amplifier is a special type of amplifier which can increase dc voltages [9].

4 DC motor speed modelling

A common actuator in control systems is the DC motor. It directly provides rotary motion and, coupled with wheels or drums and cables, can provide transitional motion. The electric circuit of the armature and the free body diagram of the rotor are shown in figure 2.

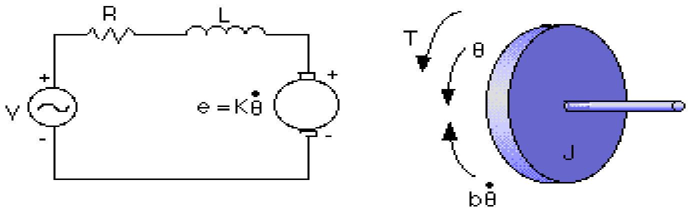


Figure 2: DC MOTOR DESIGN

For this example, we will assume the following values for the physical parameters. These values were derived by experiment from an actual motor in Carnegie Mellon's undergraduate controls lab.

* moment of inertia of the rotor (J) = 0.01 $kg.m^2/s^2$
* damping ratio of the mechanical system (b) = 0.1 Nms
* electromotive force constant (K=Ke=Kt) = 0.01 Nm/Amp
* electric resistance (R) = 1 ohm
* electric inductance (L) = 0.5 H
* input (V): Source Voltage
* output (theta): position of shaft
* The rotor and shaft are assumed to be rigid

The motor torque, T, is related to the armature current, i, by a constant factor Kt. The back emf, e, is related to the rotational velocity by the following equations:

$T=K\tau i$ ....(1)

$T=Ke\theta$ ....(2)

In SI units (which we will use), Kt (armature constant) is equal to Ke (motor constant).

5. Transfer Function

Using Laplace Transforms, the above modeling equations can be expressed in terms of s.

$S(Js+b)\,\theta(s)=KI(s)$ ……………………………………………….(3)

$(Ls+R)I(s)=V-Ks\,\theta(s)$ …………………………………….……….(4)

By eliminating I(s) we can get the following open-loop transfer function, where the rotational speed is the output and the voltage is the input.

$\theta/V = K/(Js+b)(Ls+R)+K^2$ ....(5)

6. State-Space

In the state-space form, the equations above can be expressed by choosing the rotational speed and electric current as the state variables and the voltage as an input. The output is chosen to be the rotational speed.

7. Design requirements

First, our uncompensated motor can only rotate at 0.1 rad/sec with an input voltage of 1 Volt (this will be demonstrated later when the open-loop response is simulated). Since the most basic requirement of a motor is that it should rotate at the desired speed, the steady-state error of the motor speed should be less than 1%. The other performance requirement is that the motor must accelerate to its steady-state speed as soon as it turns on. In this case, we want it to have a settling time of 2 seconds. Since a speed faster than the reference may damage the equipment, we want to have an overshoot of less than 5%.

If we simulate the reference input (r) by an unit step input, then the motor speed output should have:

Settling time less than 2 seconds

- Overshoot less than 5%
- Steady-state error less than 1%

8. PID controller

PID defines th three terms proportionality namely : P -Proportional, I - Integral, D – Derivative as shown in figure 3 .

This error represents the difference between where you want to go (Vset), and where you are actually at (Vsensor). The controller performs the PID mathematical functions on the error and applies the their sum to a process (motor, heater, etc.)

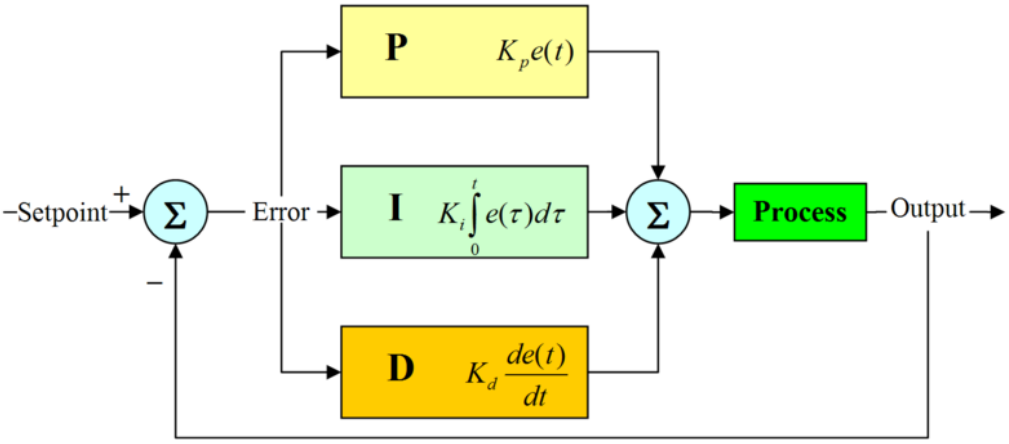


Figure 3: Complete PID Model [3].

9. Tuning PID

Tuning a system means adjusting three multipliers Kp, Ki and Kd adding in various amounts of these functions to get the system to behave the way you want. The table summarizes the PID terms and their effect on a control system[5].

Here is a straight forward approach to get one up quickly.

1. SET KP. Starting with KP=0, KI=0 and KD=0, increase KP until the output starts overshooting and ringing significantly.

2. SET KD. Increase KD until the overshoot is reduced to an acceptable level.

3. SET KI. Increase KI until the final error is equal to zero.

10. The characteristics of P, I, and D controllers

A proportional controller (Kp) will have the effect of reducing the rise time and will reduce ,but never eliminate. An integral control (Ki) will have the effect of eliminating the steady-state error, but it may make the transient response worse. A derivative control (Kd) will have the effect of increasing the stability of the system, reducing the overshoot, and improving the transient response. Effects of each of controllers Kp, Kd, and Ki on a closed-loop system are usually summarized[2]. From the main problem, the dynamic equations and the open-loop transfer function of the DC Motor are:

$S(Js+b)\ \theta(s)=KI(s)$ …………………………………………….(6)

$(Ls+R)I(s)=V - Ks\ \theta(s)$…………………………………… ….(7)

$\theta/V = K/(Js+b)(Ls+R)+K^2$ ....(8)

and the system schematic looks like figure 4:

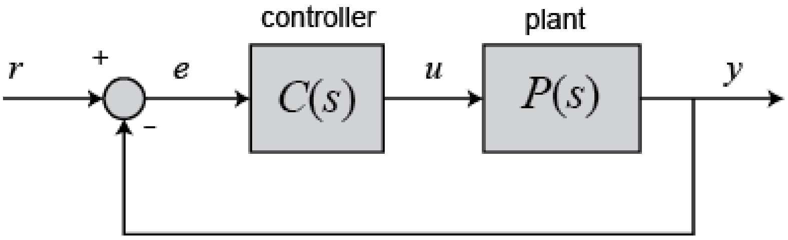


Figure 4: Simple PID Controller block Model

With a 1 rad/sec step input, the design criteria are:

- Settling time less than 2 seconds
- Overshoot less than 5%
- Steady-stage error less than 1%

First, let's take a look at how the PID controller works in a closed-loop system like in figure 4. The general equation is given by

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_p \frac{de}{dt} \quad ....(9)$$

The variable (e) represents the tracking error, the difference between the desired input value (R) and the actual output (Y). This error signal (e) will be sent to the PID controller, and the controller computes both the derivative and the integral of this error signal. The signal (u) just past the controller is now equal to the proportional gain (Kp) times the magnitude of the error plus the integral gain (Ki) times the integral of the error plus the derivative gain (Kd) times the derivative of the error[7] .

$U=ke + Ki\int edt + K_D de/dt$.............(10)

This signal (u) will be sent to the plant, and the new output (Y) will be obtained. This new output (Y) will be sent back to the sensor again to find the new error signal (e). The controller takes this new error signal and computes its derivative and its integral again. This process goes on and on.

11. Running simulation and design using dynast shell

This is the final step as regards the problem set up, it is done after the dc motor has been modelled, after the PID controller has been added to the design.

Also, having inputed the correct parameters of the PID controller to ensure the desired response.

12. Hardware sub system

A Permanent magnet DC machine in figure 5 looks like it

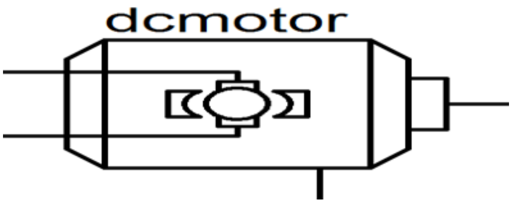


Figure 5: permanent DC machine

- **Interface**

A [V] armature-circuit +terminal

B	[V]	armature-circuit –terminal
S	[rad/s]	machine shaft
H	[rad/s]	machine housing

External Parameters

Kem = 1	[V.s/rad]	machine electro-mechanical constant
La = 0	[H]	armature-circuit inductance
Ra = 0.1	[Ω]	armature-circuit resistance
Jm = 0	[kg.m^2]	armature moment of inertia
Bm = 0	[N.m.s/rad]	armature-to-housing damping factor

12.1 PID controller

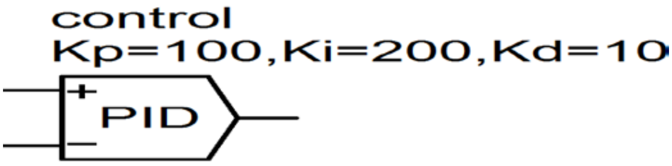


Figure 6: Tunned PID controller

- **Interface**

Up	positive input
Un	negative input
Y	Output

- **External Parameters**

Kp = 1	[–]	proportional gain
Ki = 0	[s^{-1}]	integral gain
Kd = 0	[s]	derivative gain
Tau = 0	[s]	time constant of derivative controller

13. Integrator block

Outputs the integral of scaled input signal

Where t_0 is the starting time of transient analysis.

- **Interface**

In	Input
Out	Output

- **External Parameters**

c = 1	Coefficient

14. Modelling using dynast shell

When using Dynast shell, you need not to deal with any equations at all.You can easily set up the system model in graphical form from a kit of dynamic elements like in figure 7.

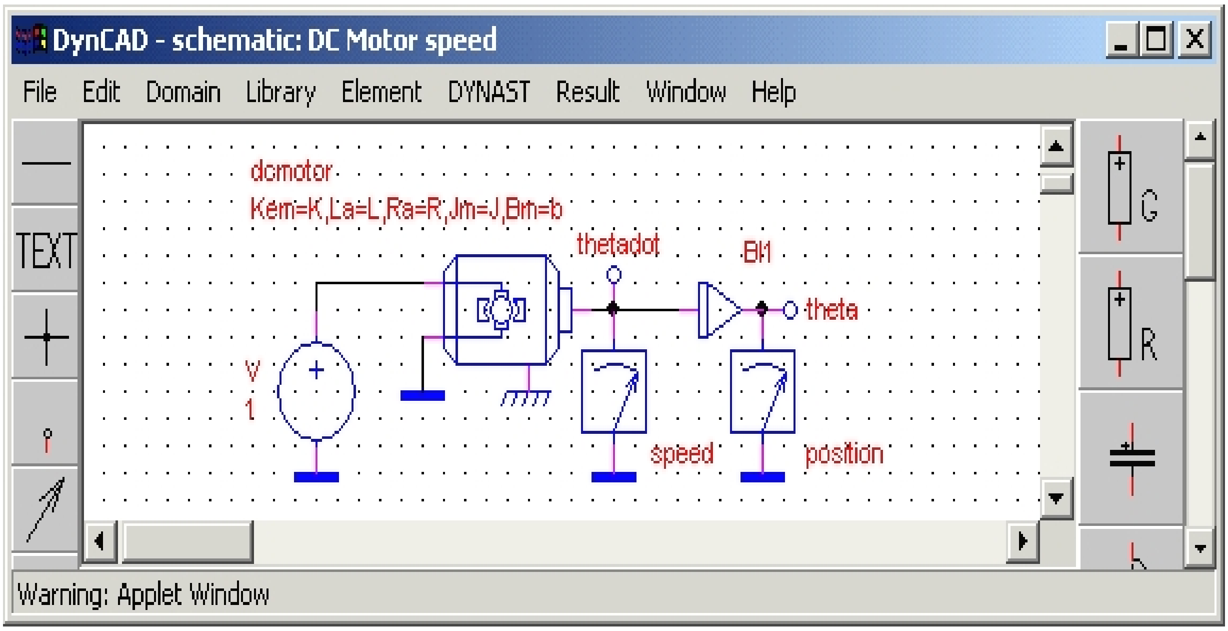


Figure 7: DC motor design without PID controller

15. Step Response Computation

To compute the open-loop step response of the system using DYNCAD

- pull-down the menu DYNAST
- click Compute analysis

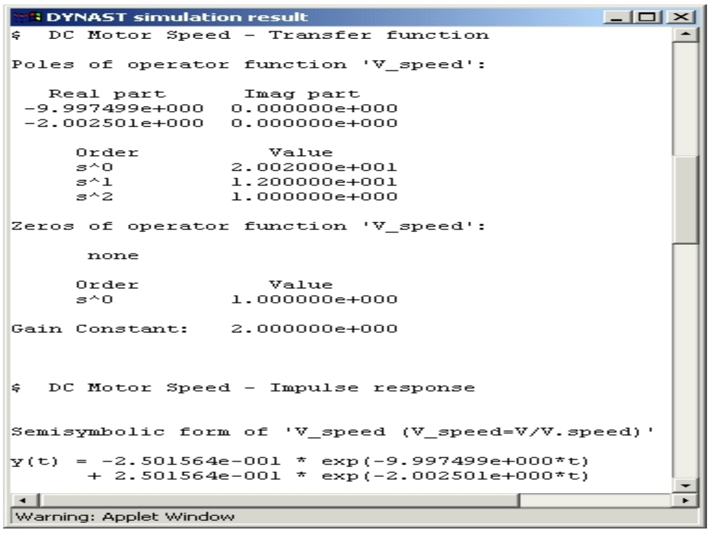
16. Transfer function computation

DYNAST can also provide linearization and semisymbolic analysis of the system model. To see the following list of poles and zeros of the system transfer function(s)

- pull down the menu Results in DYNCAD
- click Textual

You can even find in the listing the following semisymbolic expression for the step response

Figure 8 and figure 9 are simulation result and DC Motor Design With PID Controller



```
$   DC Motor Speed - Transfer function

Poles of operator function 'V_speed':

   Real part        Imag part
 -9.997499e+000   0.000000e+000
 -2.002501e+000   0.000000e+000

     Order           Value
     s^0          2.002000e+001
     s^1          1.200000e+001
     s^2          1.000000e+000

Zeros of operator function 'V_speed':

      none

     Order           Value
     s^0          1.000000e+000

Gain Constant:    2.000000e+000


$   DC Motor Speed - Impulse response


Semisymbolic form of 'V_speed (V_speed=V/V.speed)'

y(t)  = -2.501564e-001  * exp(-9.997499e+000*t)
      + 2.501564e-001  * exp(-2.002501e+000*t)
```

Figure 8: Simulation Result

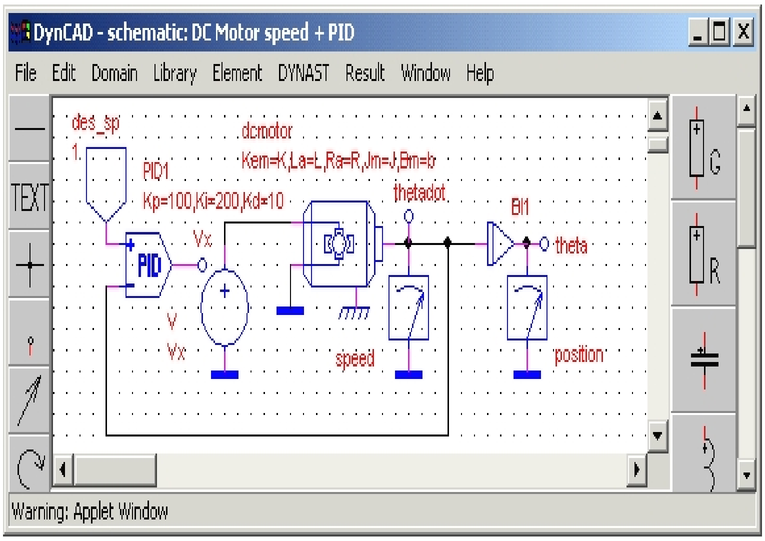


Figure 9: DC Motor Design With PID Controller

17. adding PID controller to main design

Here the PID control loop is added to the main system design, having tunned to the desired parameters to effect the desired response in figure 10 and 11

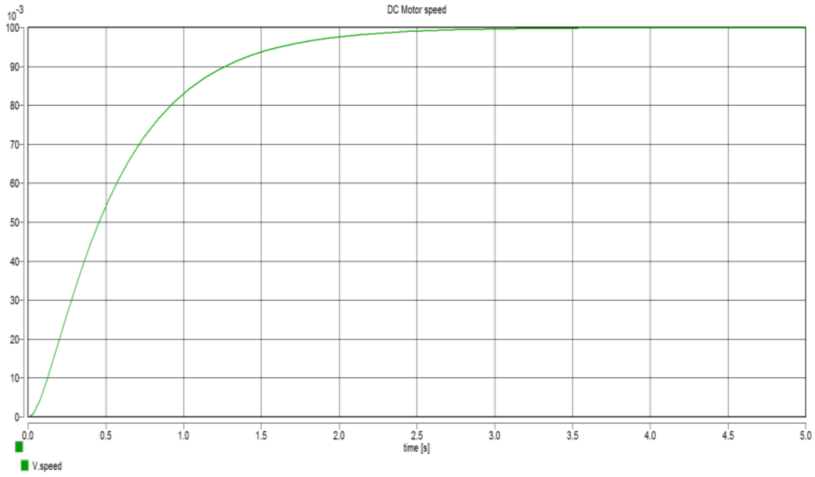


Figure 10: Graph For DC Motor Without PID Showing Velocity Vs Time

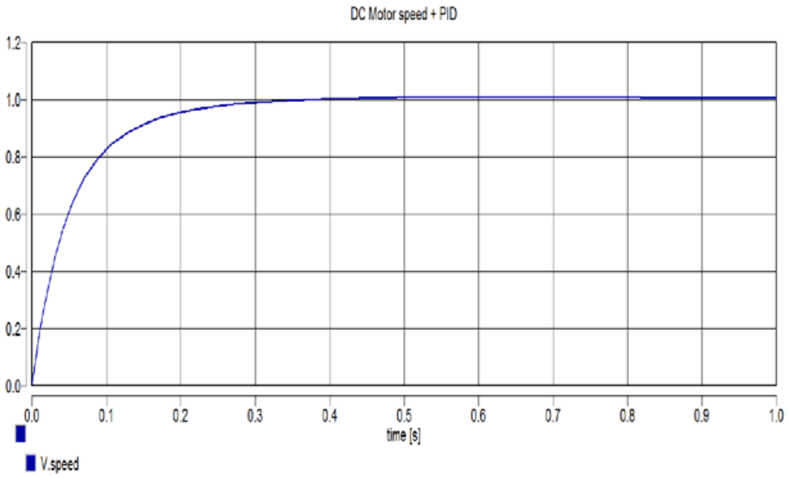


Figure 11: Graph For DC Motor Speed With PID Controller Showing Velocity Vs Time

18. Simulation result

Having set the proportional, integral and derivatives parameter to the desired value, which are;

Kp= 100

With proportional band, the controller output is proportional to the error or a change in measurement (depending on the controller).

(Controller output) = (error)*100/(proportional band)

Ki=200

With integral action, the controller output is proportional to the amount of time the error is present. Integral action eliminates offset.

CONTROLLER OUTPUT = (1/INTEGRAL) (Integral of) e(t) d(t) Therefore the Integral gain reduces the settling time. Integral term will eliminate the steady-state error

Kd=10

With derivative action, the controller output is proportional to the rate of change of the measurement or error. The controller output is calculated by the rate of change of the measurement with time.

Where m is the measurement at time t.

Derivative at this parameter reduces overshoot.

19. Conclusion and recommendation

I will conclude this project having laid a detailed pricedure that shows the design and analysis of a dc motor speed control using PID controller to correct errors.

This simply entails that PID controller proposes that it gives the desire response when added to a dc motor speed control.

This paper demostrated why a PID controller should be used to design a dc motor speed control.Having studied both designs of dc motor speed control with and without PID , it could be concluded that it is the preferred means for controlling the speed of a dc motor control.

With the aid of dynast shell simulator, a comprehensive design of the model was carried out with ease, that is eliminating the need for complex equations and calculations to achieve the aim .

Reference

[1] Boumediène ALLAOUA*, and Brahim MEBARKI Setting Up PID DC Motor Speed Control Alteration Parameters Using Particle. Swarm Optimization Strategy Brahim GASBAOUI, 2007.

[2] Kamaraj Tuning Algorithms for PID Controller Using Soft Computing Techniques by 2001.

[3] lancet Mit Intro to D.C. Motors User conference,Austin texas, Vol 11, 2004.

[4] Rick Bickley Dc Motor Control System For Robot Applications Rick Bickley Publisher, 2003.

[5] K. Samsai Optimal PI Controller Design Simulation, Robotic and Automation Research Unit School of

Electrical Engineering 2007.

[6] Y.P Wang , a Genetic Algorithm to design PI Controller, Int. Conference Power Sytem Technology 2007

[7] Matt Krass, PID Control Theory, Matt Krass 2nd Edition, 2006.

[8] SAFFET AYASUN DC Motor Speed Control Methods Using MATLAB/Simulink Department of Electrical and Electronics Engineering, Nigde University, Nigde 51100, Turkey, 2006.

[9] S. Li and R. Challoo, Restructuring an electric machinery course with an integrative approach and computer-assisted teaching methodology, IEEE Trans Educ 49, 2006.

[10] M.Y. Chow and Y. Tipsuwan, Gain adaptation of networked DC motor controllers based on QOS variations, IEEE Trans Ind Electron 50, 2003.

[11] S. J. Chapman, Electric machinery fundamentals, 3rd edition, WCB/McGraw-Hill, New York, 1999.