Simulation and Implementation of Servo Motor Control with Sliding Mode Control (SMC) using Matlab and LabView

Bondhan Novandy

http://bono02.wordpress.com/

Outline

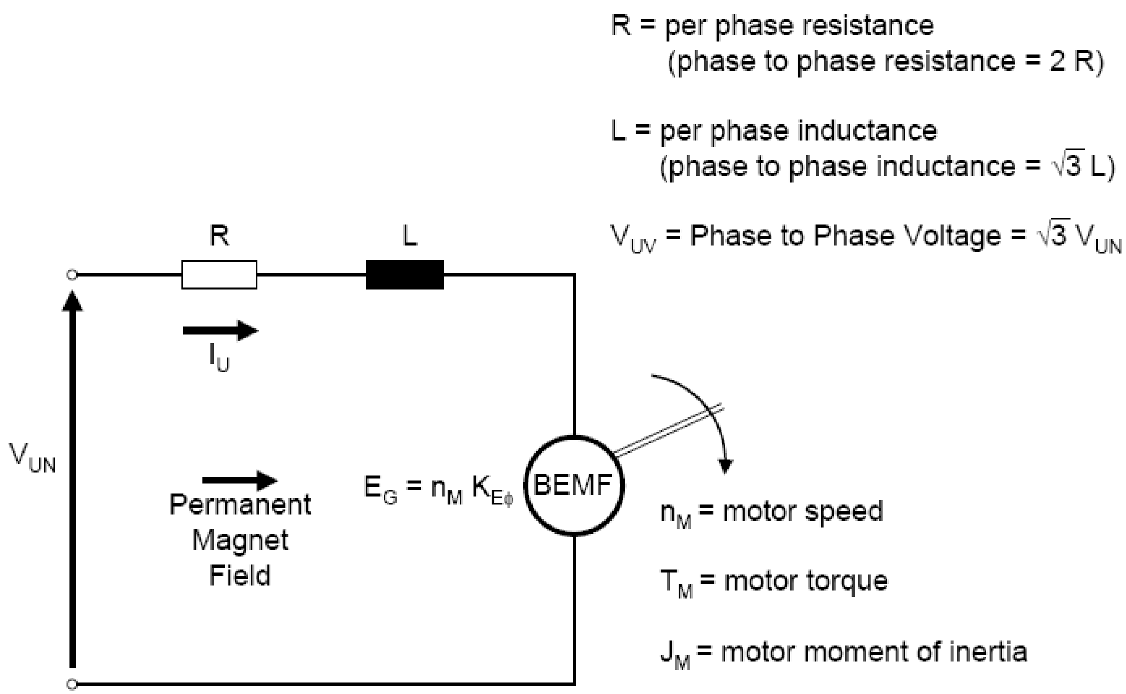
- Background
- Motivation
- AC Servo Motor
- Inverter & Controller
- Hardware Block Diagram
- Mathematical Modeling
- Sliding Mode Controller Design
- Simulation using Matlab
- Simulation & Implementation using LabView
- Several Ideas

Motivation

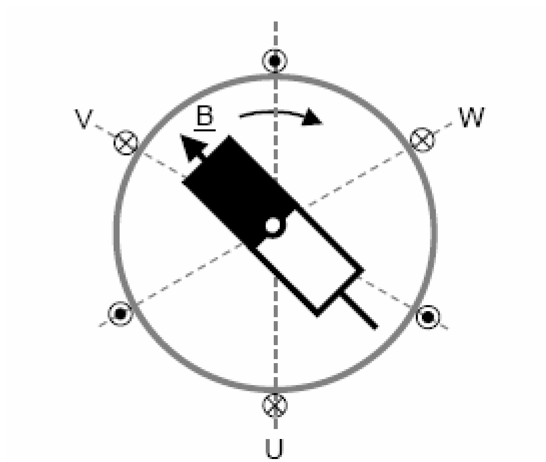
- Sliding Mode Control is a robust control scheme based on the concept of changing the structure of the controller in response to the changing state of the system in order to obtain a desired response
- The biggest advantage of SMC is its insensitivity to variation in system parameters, external disturbances and modeling errors
- This can be achieved by forcing the state trajectory of the plant to the desired surface and maintain the plants state trajectory on this surface for subsequent time
- Because of these factors SMC is chosen as the controller for our device

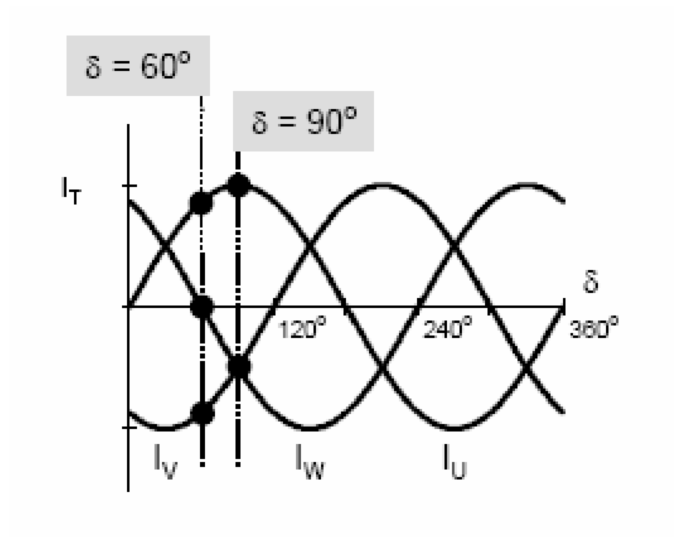
AC Servo Motor

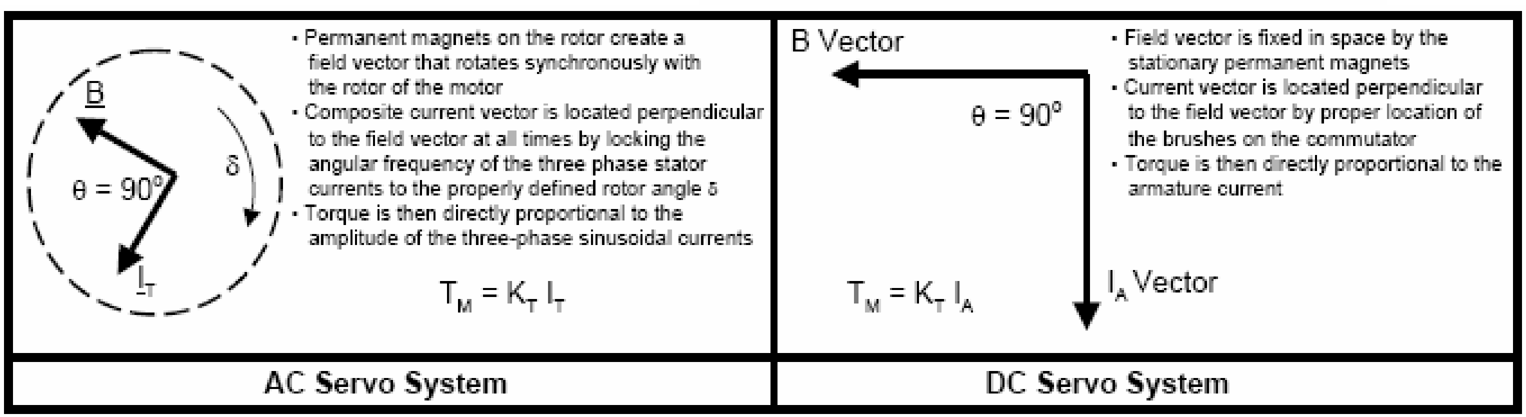
- The difference between AC Servo Motor and DC servo motor is the design of the motor where in AC motor the permanent magnet is on the rotor. The block diagram of an AC servo motor is very similar to the block diagram of DC servo motor:



AC Servo Motor

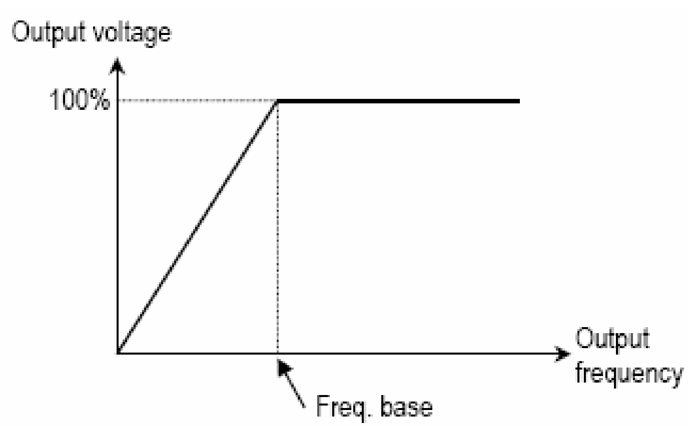






Inverter and Controller

- Inverter is used to transform electricity from 1 single phase into 3 phase
- It works by controlling the rotational speed of an AC motor by controlling the frequency of the electrical power supplied to the motor
- Our inverter the we use the linear V/F mode

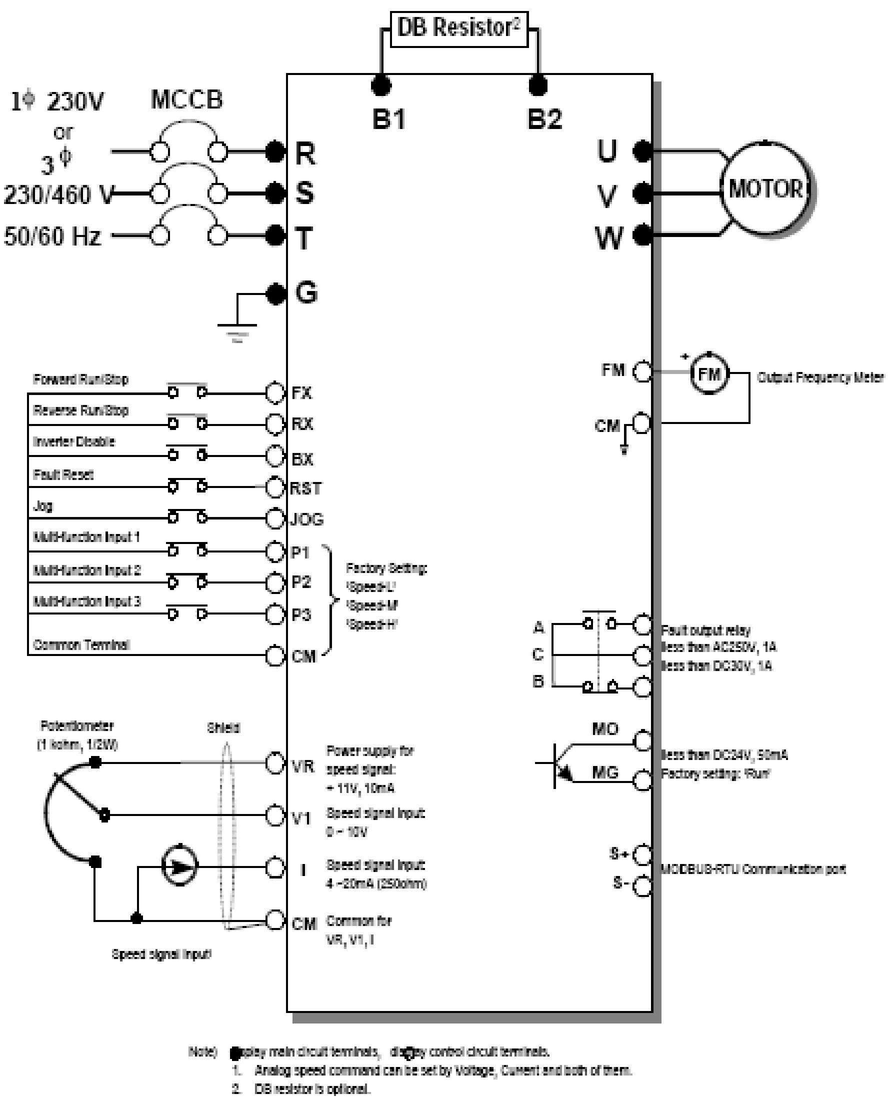


[V/F Pattern: 'Linear']

Output max voltage is equal
to max input voltage = 220 V RMS

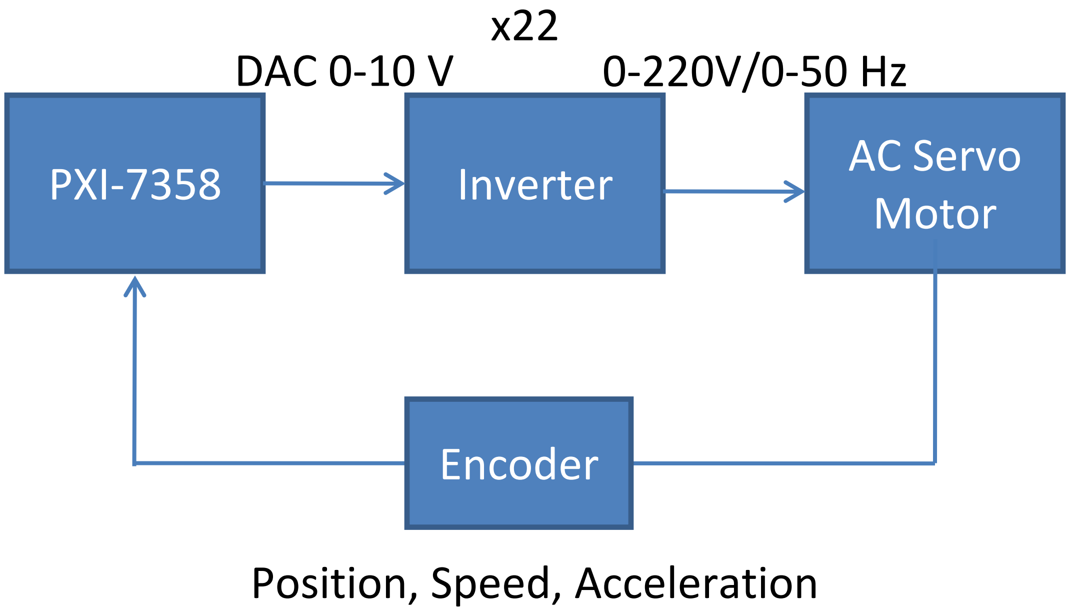
The default frequency is 50 Hz

Inverter and Controller Cont’d

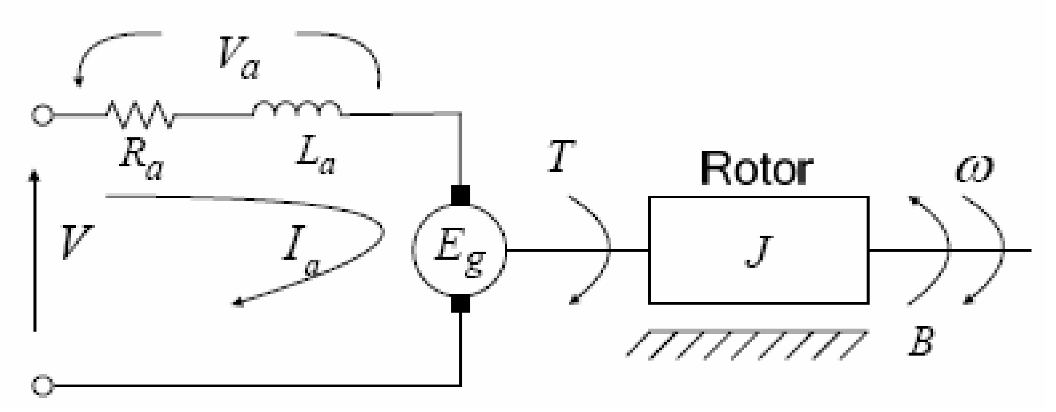


- Analog input voltage 0-10 V is used to change the volt/frequency of the inverter output
- The scale between input : output = 10 : 220 depending on the configuration of V/F
- NI-PXI 7358 is chosen as the controller and it has DAC 0-10 output

Hardware Block Diagram



Mathematical Modeling



Position control:

$$Ia*Ra + La*\frac{dI}{dt} + Kb\,\frac{d\theta}{dt} = V \quad ...(1)$$

$$T - T_{Load} = Kt*Ia - T_{Load} = Tj + Tb = J\,\frac{d^2\theta}{dt^2} + B\,\frac{d\theta}{dt} ...(2)$$

$$Ia = \frac{1}{Kt}\left(J\,\frac{d^2\theta}{dt^2} + B\,\frac{d\theta}{dt} + T_{Load}\right)...(3)$$

Mathematical Modeling Cont'd

$$(3) \rightarrow (1)$$

$$\frac{Ra}{Kt}(J\frac{d^2\theta}{dt^2}+B\frac{d\theta}{dt}+T_{Load})+\frac{La}{Kt}\frac{d}{dt}(J\frac{d^2\theta}{dt^2}+B\frac{d\theta}{dt}+T_{Load})+Kb\frac{d\theta}{dt}=V$$

$$\frac{J*Ra}{Kt}\frac{d^2\theta}{dt^2}+\frac{B*Ra}{Kt}\frac{d\theta}{dt}+\frac{Ra}{Kt}T_{Load}+\frac{J*La}{Kt}\frac{d^3\theta}{dt^3}+\frac{B*La}{Kt}\frac{d^2\theta}{dt^2}+\frac{La}{Kt}T'_{Load}+Kb\frac{d\theta}{dt}=V$$

Finally we get (4):

$$\dot{\theta}_1=\theta_2$$

$$\dot{\theta}_2=\theta_3$$

$$\dot{\theta}_3=\frac{Kt}{J*La}V-\frac{B}{J}\theta_3-\frac{Ra}{La}\theta_3-\frac{Ra*B}{La*J}\theta_2-\frac{Kb*Kt}{La*J}\theta_2-\frac{1}{J}\frac{La}{Kt}T'_{Load}-\frac{Ra}{La*J}T_{Load}$$

Sliding Surface

- We have 3rd order system

$$s = (\frac{d}{dt} + \lambda)^{n-1}\tilde{\theta}, \text{ choosing n=3 and } \tilde{\theta} = \theta - \theta_d \text{ is tracking error}$$

$$= (\frac{d}{dt} + \lambda)^2 \tilde{\theta}$$

$$= (\frac{d^2\tilde{\theta}}{dt^2} + 2\lambda \frac{d\tilde{\theta}}{dt} + \lambda^2\tilde{\theta})$$

$$= \ddot{\tilde{\theta}} + 2\lambda\dot{\tilde{\theta}} + \lambda^2\tilde{\theta}$$

$$= (\ddot{\theta} - \ddot{\theta}_d) + 2\lambda(\dot{\theta} - \dot{\theta}_d) + \lambda^2(\theta - \theta_d)$$

Equivalent Control

- We try to force the state trajectory to slide on our surface so that: $s = 0$ $\dot{s} = 0$

$$\dot{s} = (\dddot{\theta} - \dddot{\theta}_d) + 2\lambda(\ddot{\theta} - \ddot{\theta}_d) + \lambda^2(\dot{\theta} - \dot{\theta}_d)$$

$$0 = \frac{Kt}{J*La}V - \frac{B}{J}\theta_3 - \frac{Ra}{La}\theta_3 - \frac{Ra*B}{La*J}\theta_2 - \frac{Kb*Kt}{La*J}\theta_2 - \frac{1}{J}\frac{La}{Kt}T'_{Load}$$

$$-\frac{Ra}{La*J}T_{Load} - \dddot{\theta}_d + 2\lambda(\theta_3 - \ddot{\theta}_d) + \lambda^2(\theta_2 - \dot{\theta}_d)$$

$$V_{eq} = \frac{B*La}{Kt}\theta_3 + \frac{La}{Kt}T'_{Load} + \frac{Ra*J}{Kt}\theta_3 + \frac{Ra*B}{Kt}\theta_2 + \frac{Ra}{Kt}T_{Load} + Kb*\theta_2 + \frac{La*J}{Kt}\dddot{\theta}_d$$

$$-2\lambda\frac{La*J}{Kt}(\theta_3 - \ddot{\theta}_d) - \lambda^2\frac{La*J}{Kt}(\theta_2 - \dot{\theta}_d)$$

SMC Controller

$$V = V_{eq} + V_{switching}, \textit{where } V_{switching} = -K * sat(s/\phi)$$

and sat is the function:

$$sign(s) \quad if \;\; \mathrm{abs}(s) > \phi$$

$$s/\phi \quad if \;\; \mathrm{abs}(s) < \phi$$

$$V_{eq} = \frac{B*La}{Kt}\theta_3 + \frac{La}{Kt}T'_{Load} + \frac{Ra*J}{Kt}\theta_3 + \frac{Ra*B}{Kt}\theta_2 + \frac{Ra}{Kt}T_{Load} + Kb*\theta_2 + \frac{La*J}{Kt}\dddot{\theta}_d$$

$$-2\lambda\frac{La*J}{Kt}(\theta_3 - \ddot{\theta}_d) - \lambda^2\frac{La*J}{Kt}(\theta_2 - \dot{\theta}_d) - K*sat(s/\phi)$$

Lyapunov Function

If there exist Lyapunov function, so that

1) $V(x) \succ 0 \qquad \forall\, |x| < r$

2) $\dot{V}(x) = \nabla V^T(x)\, f(x) \preceq 0$

for all $|x| < r$

It is stable in the sense of Lyapunov

Stability

We ensure the stability of our system choosing K to be large enough so that stable in the sense of Lyapunov.

Lyapunov candidate:

$$V = \frac{1}{2} s^2 \qquad \succ 0$$

$$\dot{V} = s\dot{s} = s\left(\frac{Kt}{J * La} V - \frac{B}{J}\theta_3 - \frac{Ra}{La}\theta_3 - \frac{Ra * B}{La * J}\theta_2 - \frac{Kb * Kt}{La * J}\theta_2 - \frac{1}{J}\frac{La}{Kt} T'_{Load}\right.$$

$$\left. - \frac{Ra}{La * J} T_{Load} - \dddot{\theta}_d + 2\lambda(\theta_3 - \ddot{\theta}_d) + \lambda^2(\theta_2 - \dot{\theta}_d) - K * sat(s/\phi)\right)$$

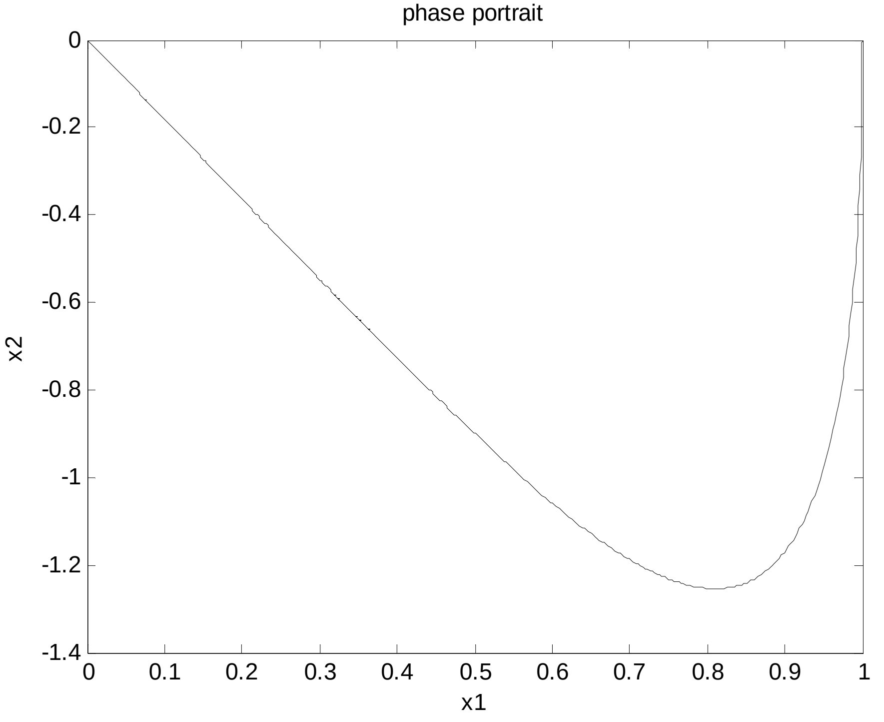
$$\dot{V} = s(f - K * sat(s/\phi))$$

$$= f * s - K|s| \qquad \le 0$$

Simulation Using Matlab

Plant parameter:

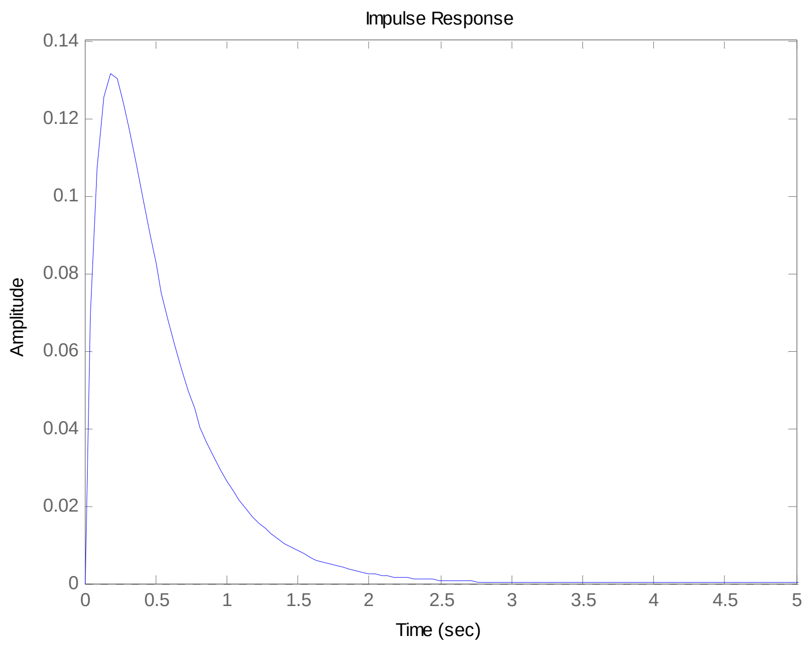
```
j=0.01;
b=0.1;
Kt=0.01;
Kb=0.29098;
Ra=1;
La=0.5;
```

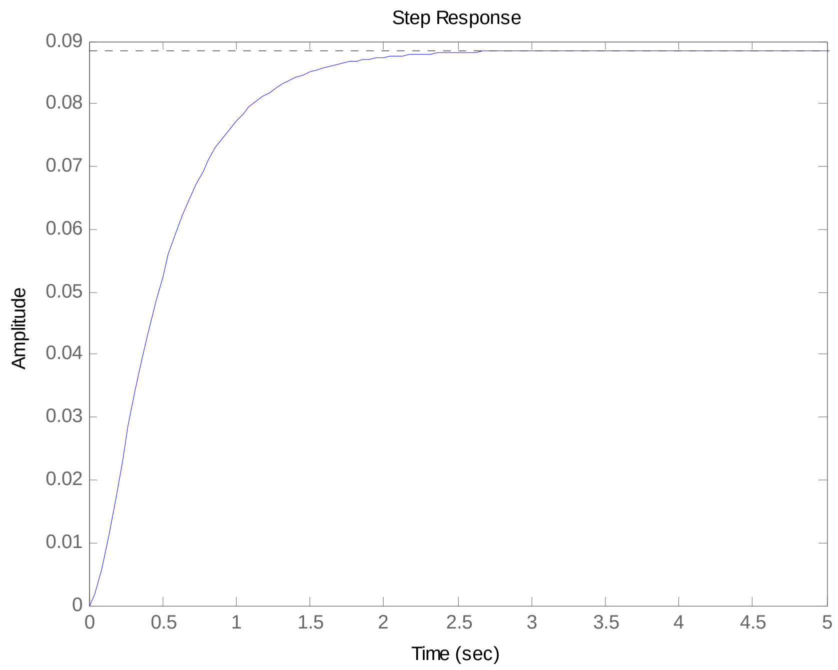



```
The unity feedback
transfer function
is:
       2
-------------------
s^2 + 12 s + 22.58
```

Transient Response Without Controller

```
Stability Analysis (Linear System)
Controllability:
% Number of uncontrollable states
>> unco = length(A)-rank(ctrb(A,B))
unco =
     0
Observability:
% Number of unobservable states
unob = length(A)-rank(Ob)
unob =
     0
Thus the system is stable.
```

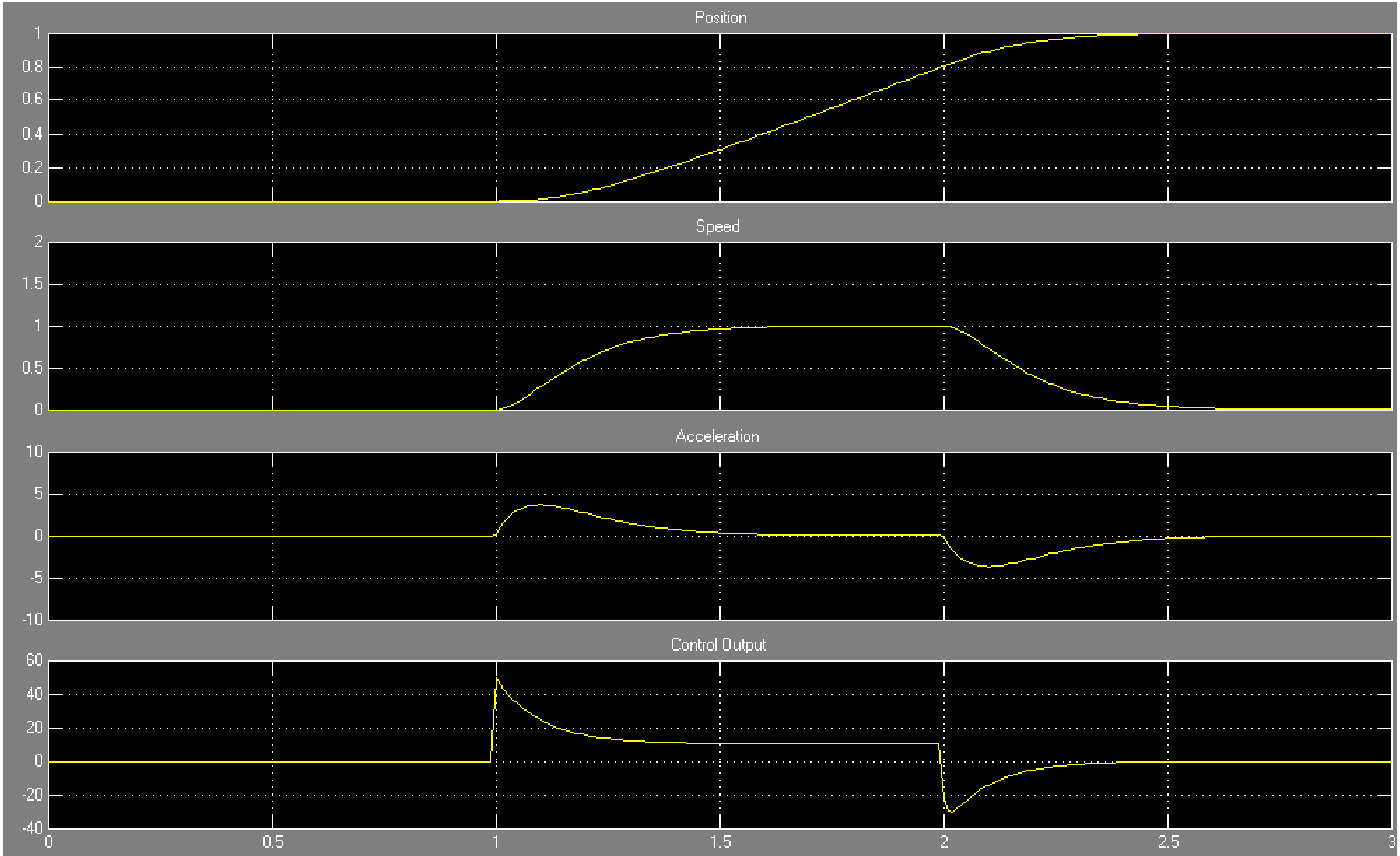




Simulation Result With SMC Controller

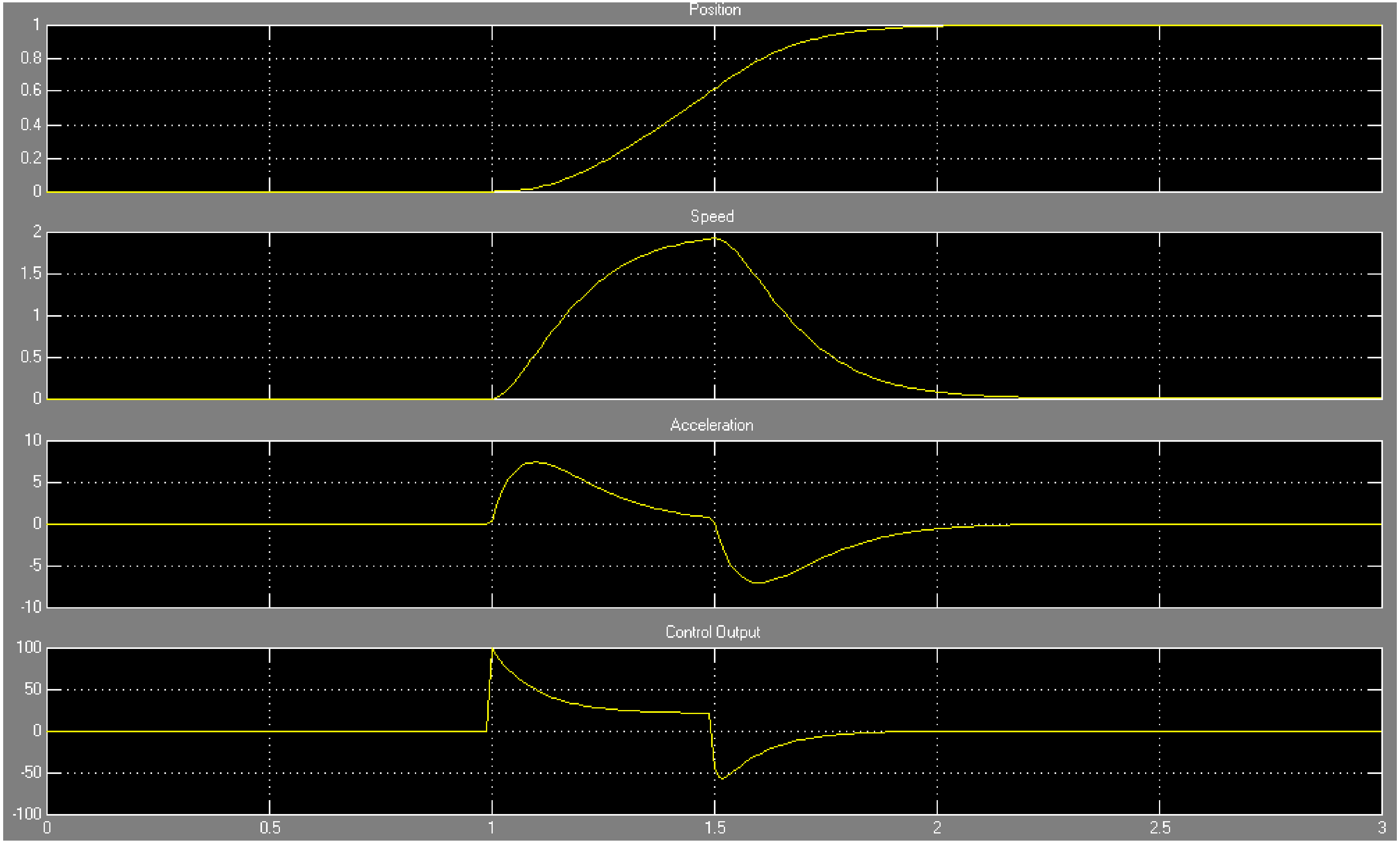
Step Response

K = 50; Lambda = 10; delta = 0.5;



Step Response

K = 100; Lambda = 10; delta = 0.9;



Trajectory Following

The trajectory is defined so that it will not produce shock while moving because of discontinuity.

Quintic polynomial:

$$q0 = a0 + a1*t + a2*t^2 + a3*t^3 + a3*t^4 + a5*t^5$$

$$where:$$

$$q0 = \text{start position}$$

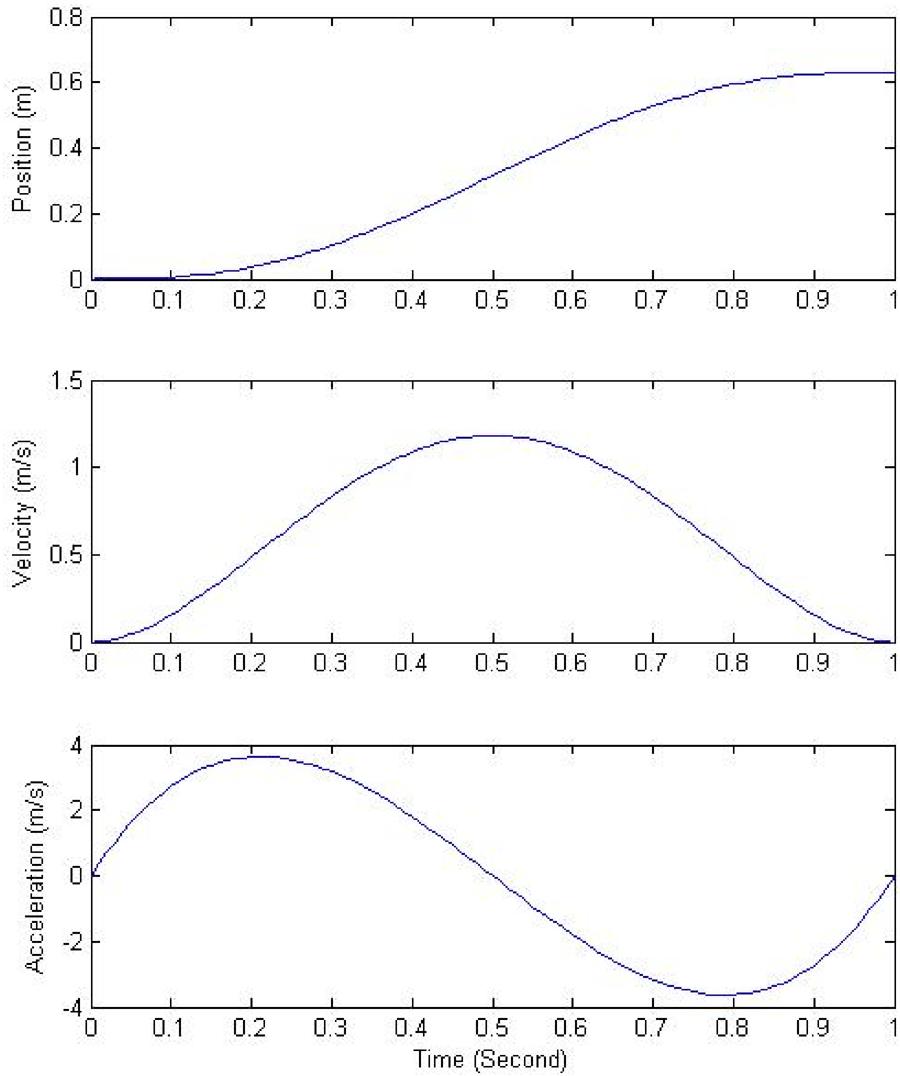
$$qf = \text{final position}$$

$$v0 = \text{start velocity}$$

$$fv = \text{final velociy}$$

$$\alpha 0 = \text{start acceleration}$$

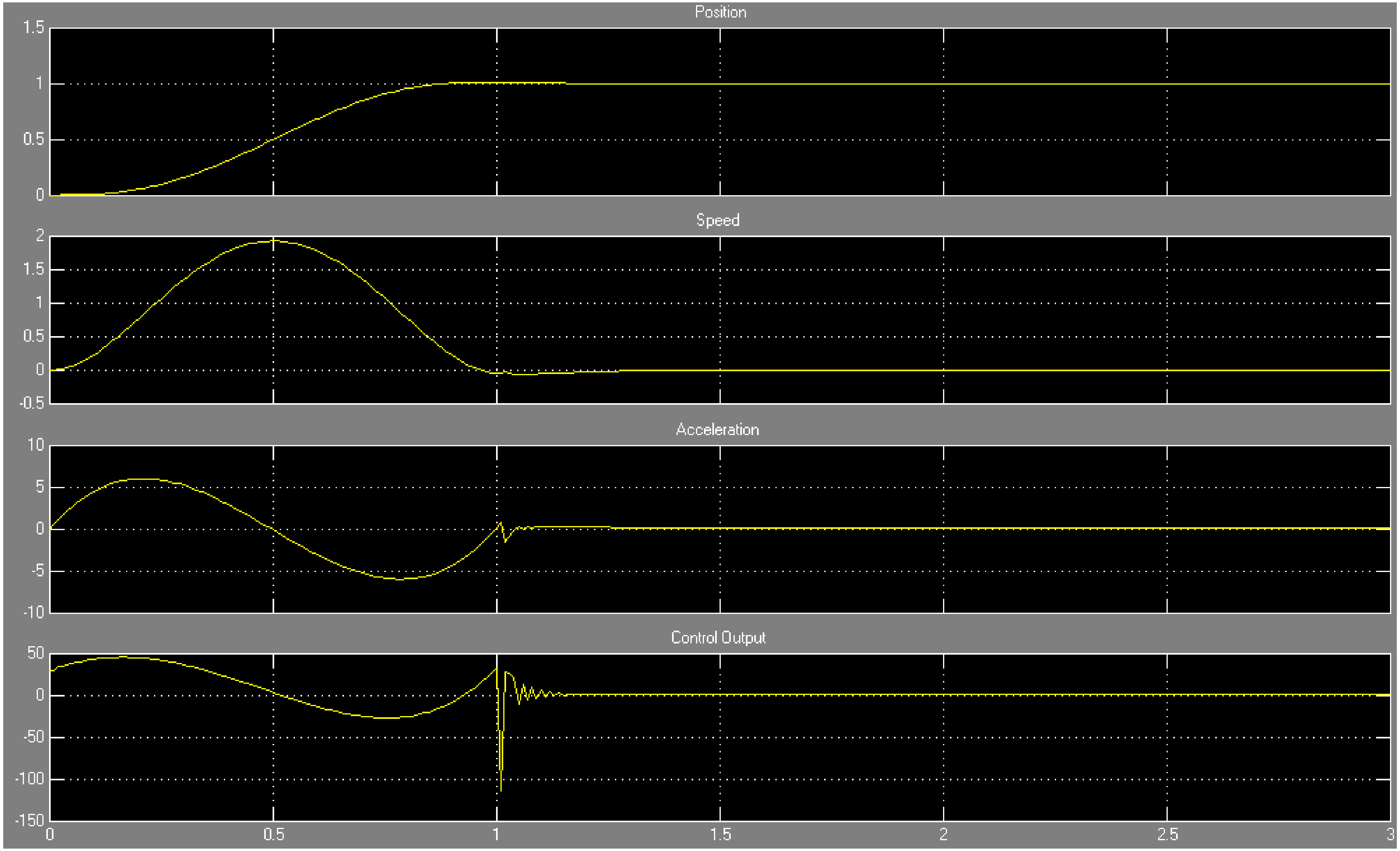
$$\alpha f = \text{final acceleration}$$



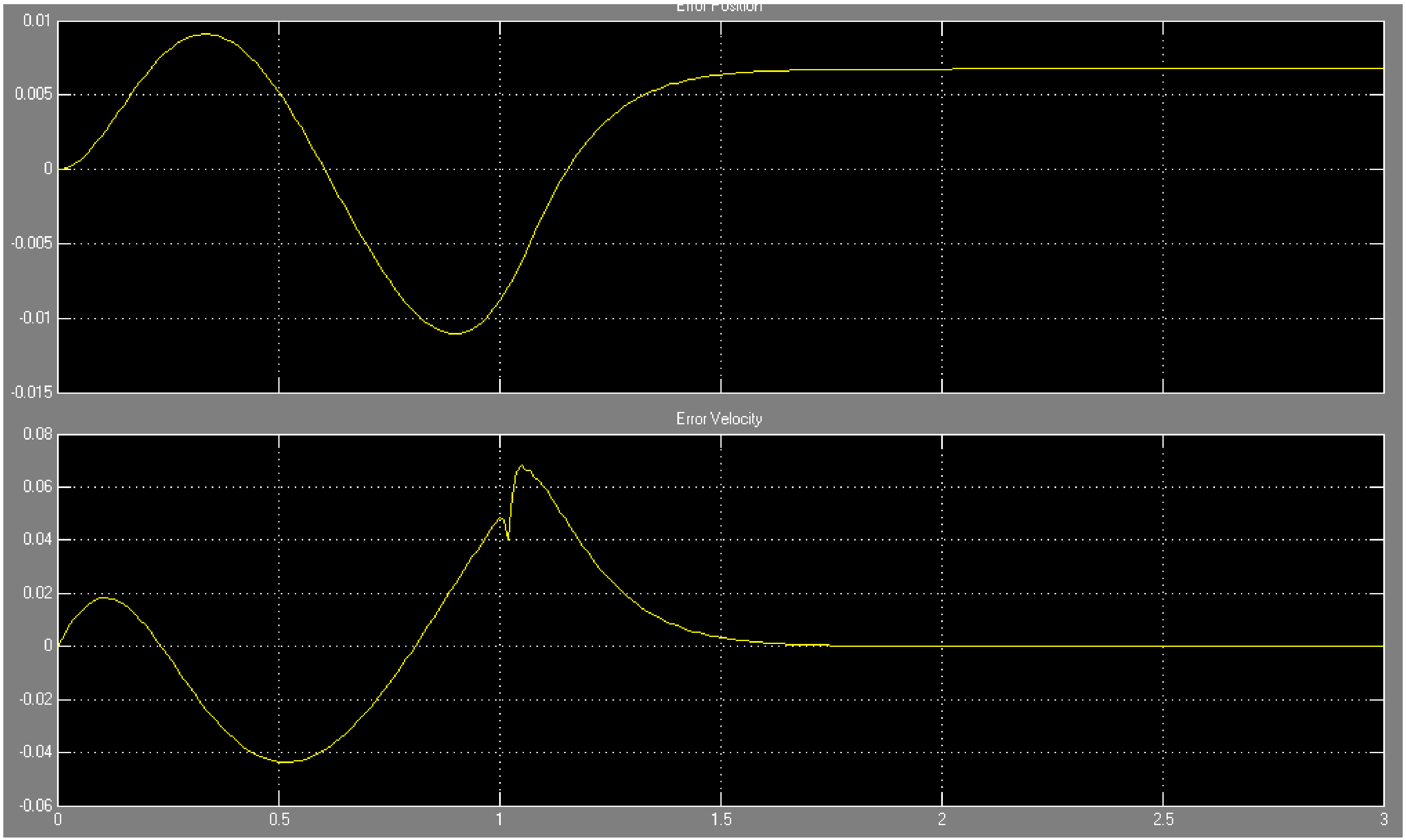
Quintic polynomial trajectory for position, velocity and acceleration

Trajectory Following Response

K = 80;   Lambda = 10;   delta = 0.9;



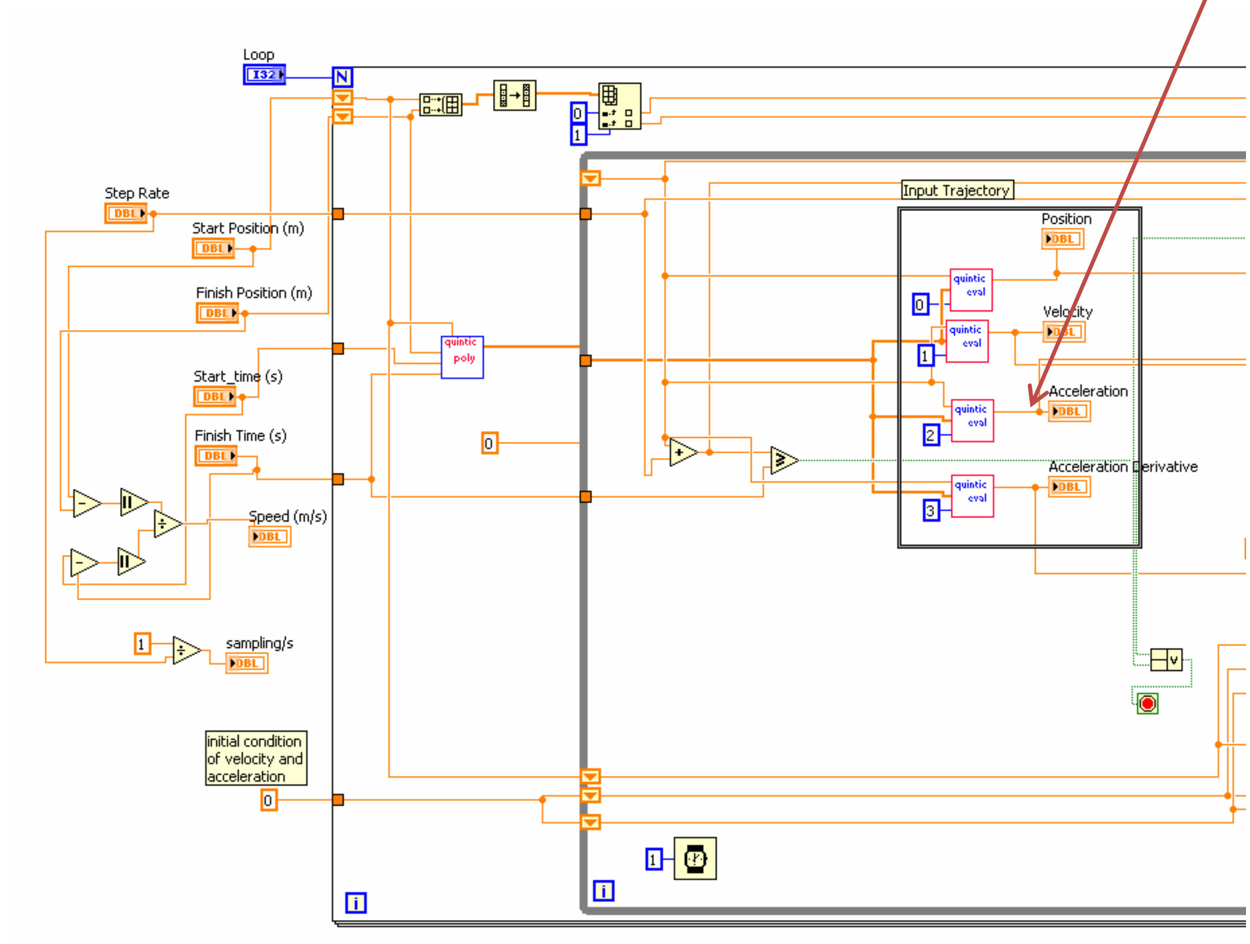
Error



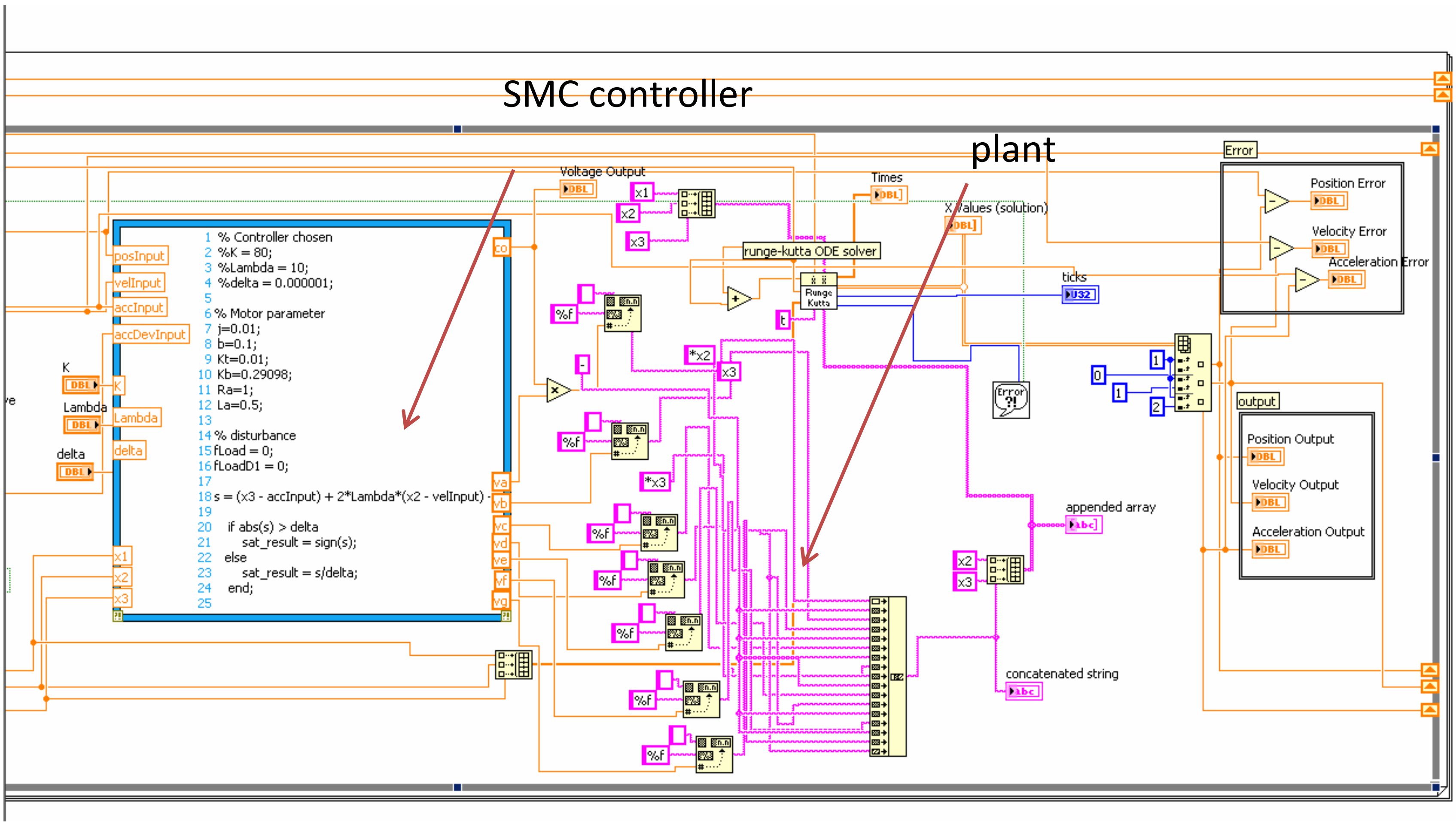
Error Velocity
0.01
0.005
0
-0.005
-0.01
-0.015
0.08
0.06
0.04
0.02
0
-0.02
-0.04
-0.06
0
0.5
1
1.5
2
2.5
3

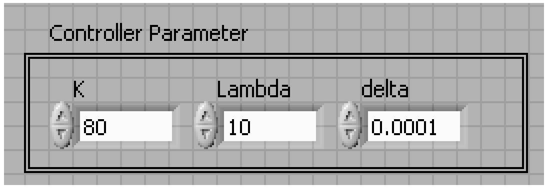
Simulation Using LabView

Trajectory Generator

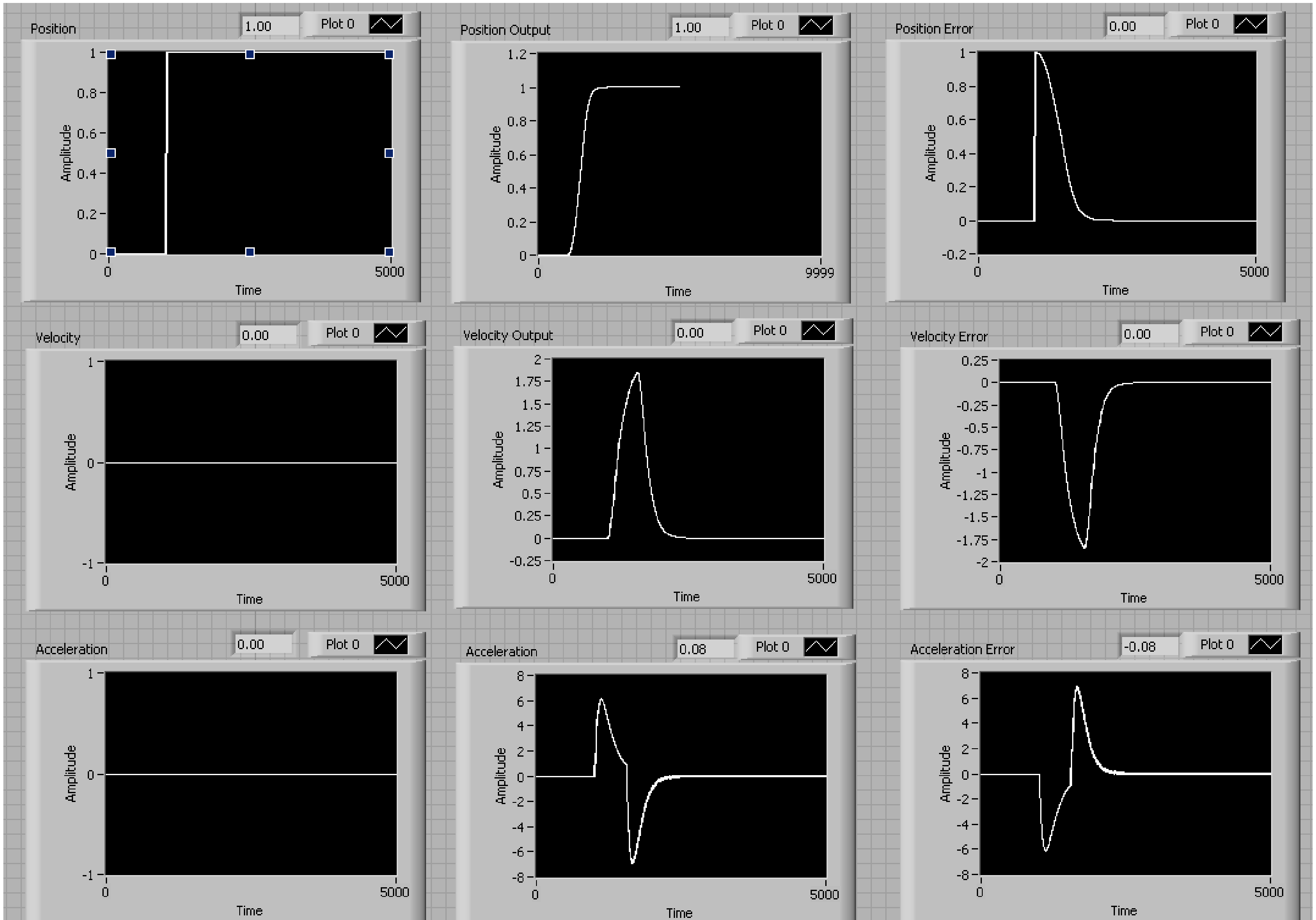


Simulation Using LabView

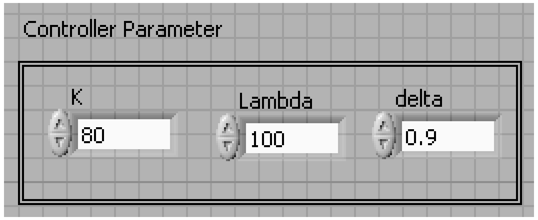


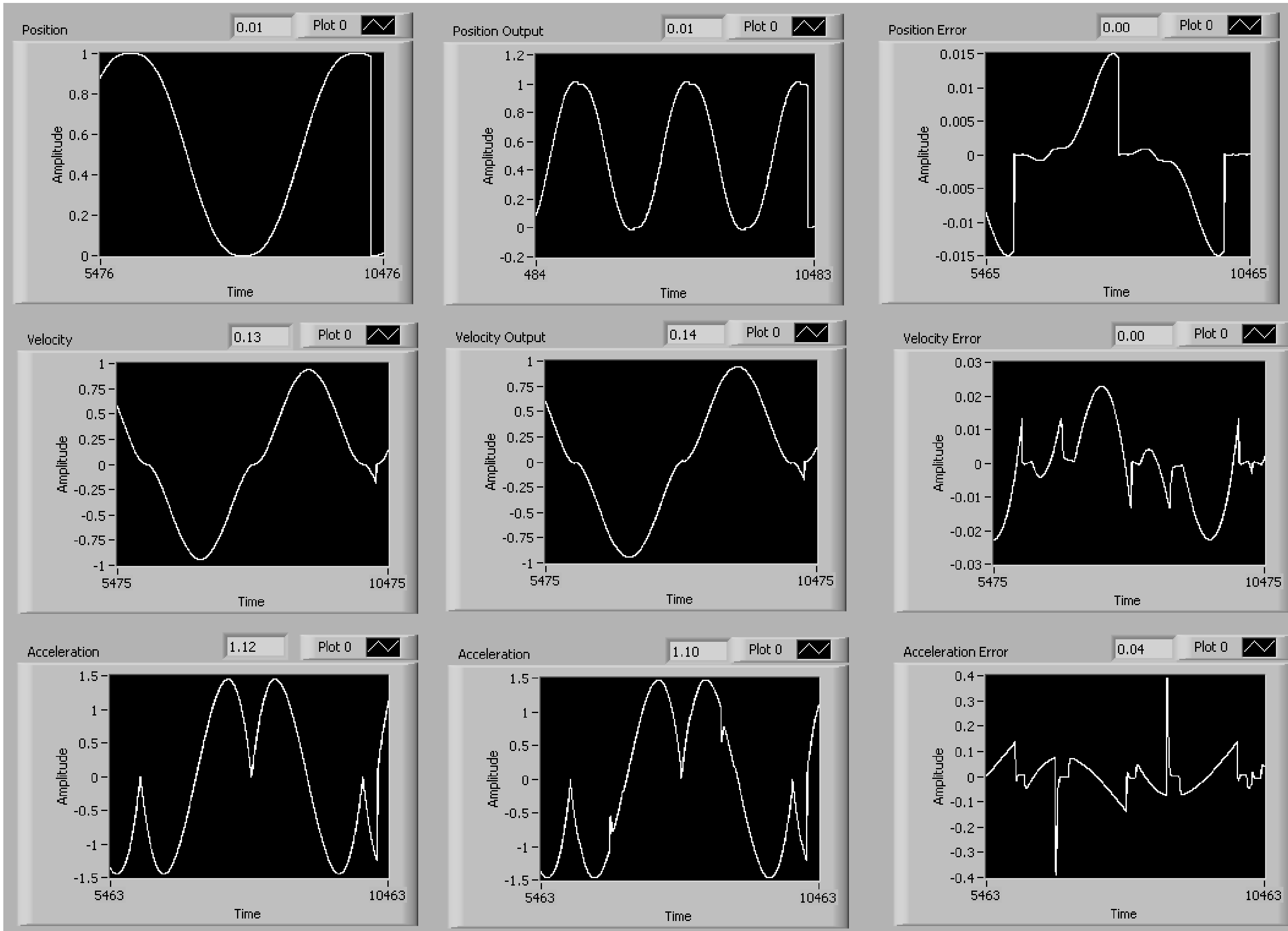


Step Response



Trajectory Response





Implementation

- Implementation in NI Real-Time can be done by replacing the runge-kutta ODE solver and array composition with the real motor which is the SERVO Motor
- The feedback of the motor must be available (encoder)

Problems and Discussion

- Real-time clock generation -> Sampling time
- Motor parameters
- The feedback of the system are:
  - Position
  - Velocity
  - Acceleration

  *) In the simulation when acceleration feedback was defined as 0, the error is still small
- Integral sliding mode control

$$s = \left(\frac{d}{dt} + \lambda\right)^{n-1} \int_0^t \tilde{\theta}$$, choosing n=4 and $\tilde{\theta} = \theta - \theta_d$ is tracking error