

---

# **CS-417 INTRODUCTION TO ROBOTICS AND INTELLIGENT SYSTEMS**

**Control**

**Slides by P. Giguere**

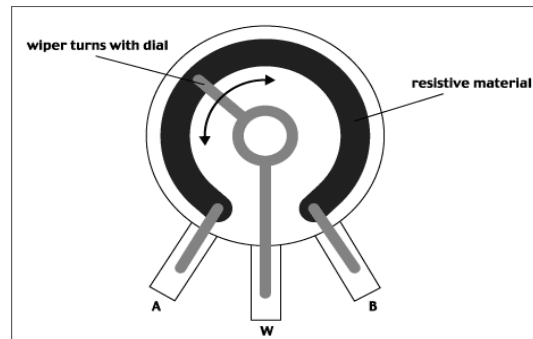
# Actuators + Sensors

- Presented actuators and sensors, e.g.:
  - Electrical motor
  - Angular position encoder
- How to get the best precision/performance?

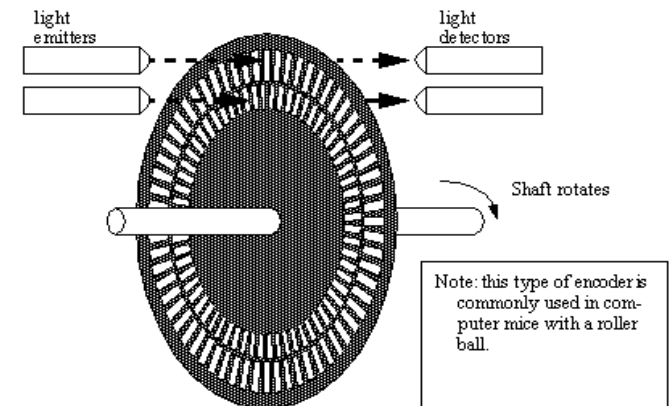
Electric Motor



Potentiometer

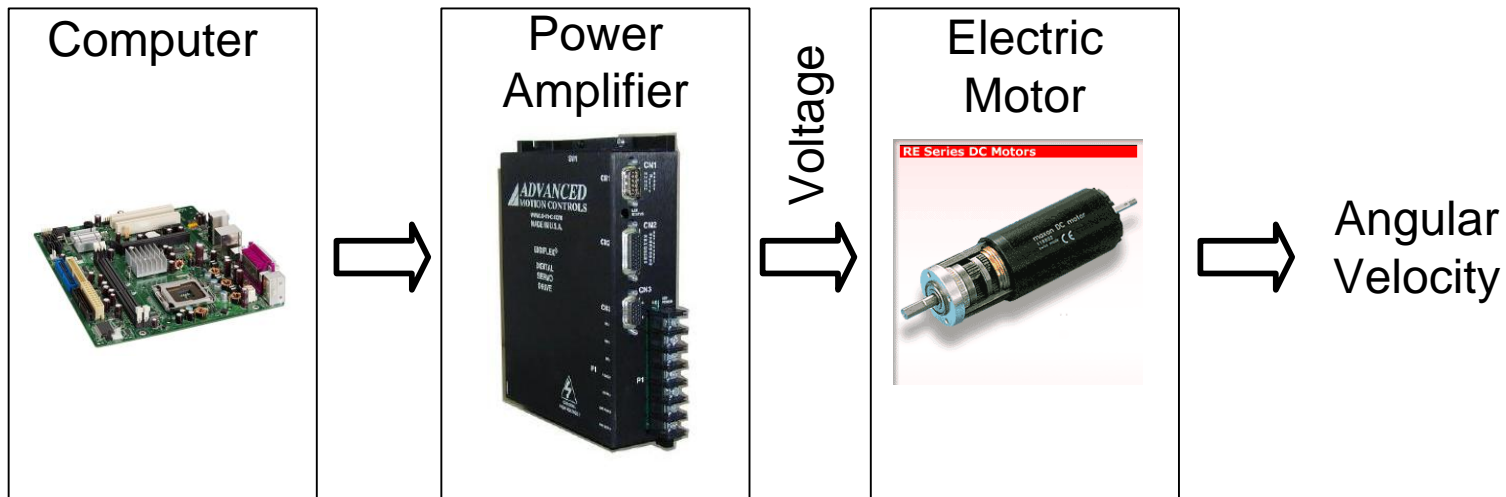


Optical Encoder



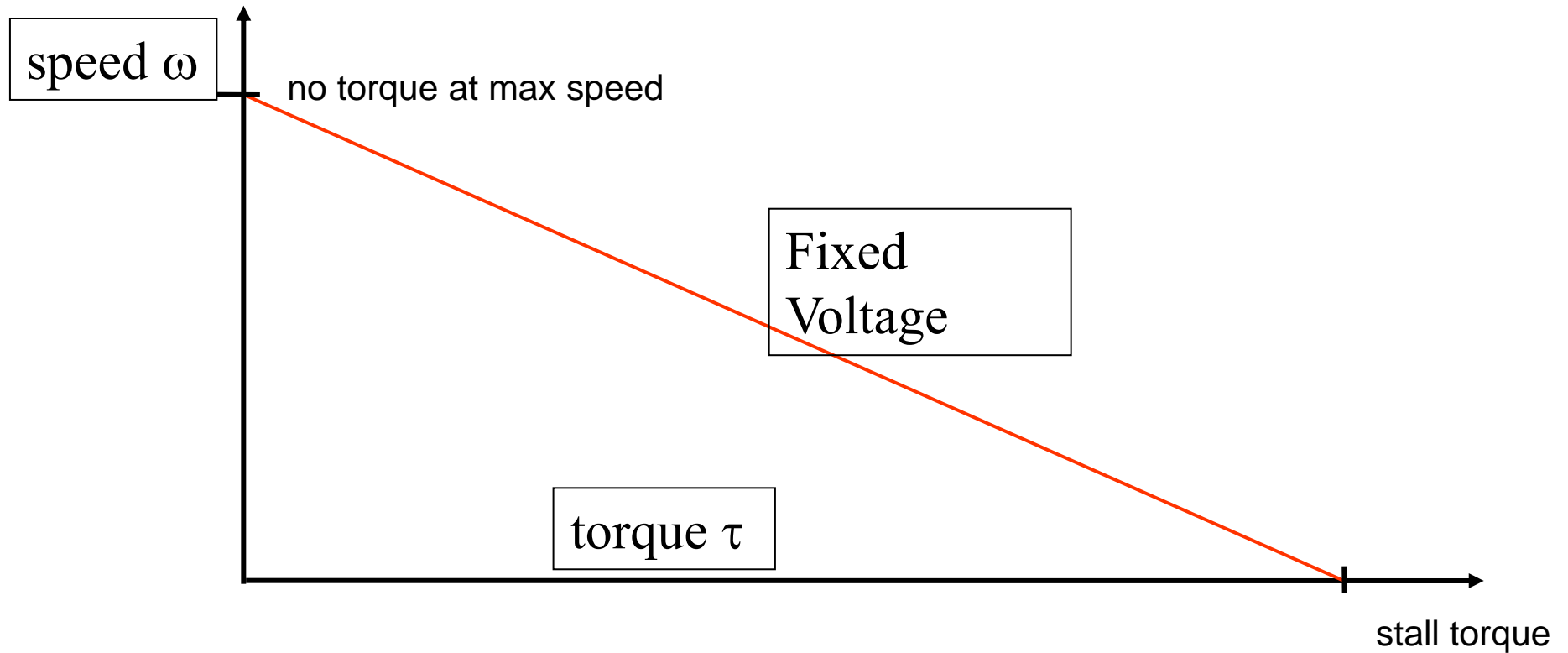
# Open-loop

- Goal: move a differential drive robot
  - spin motors at a given angular velocity
- How: apply a fixed voltage to it,
  - and never check to see if it is rotating properly...



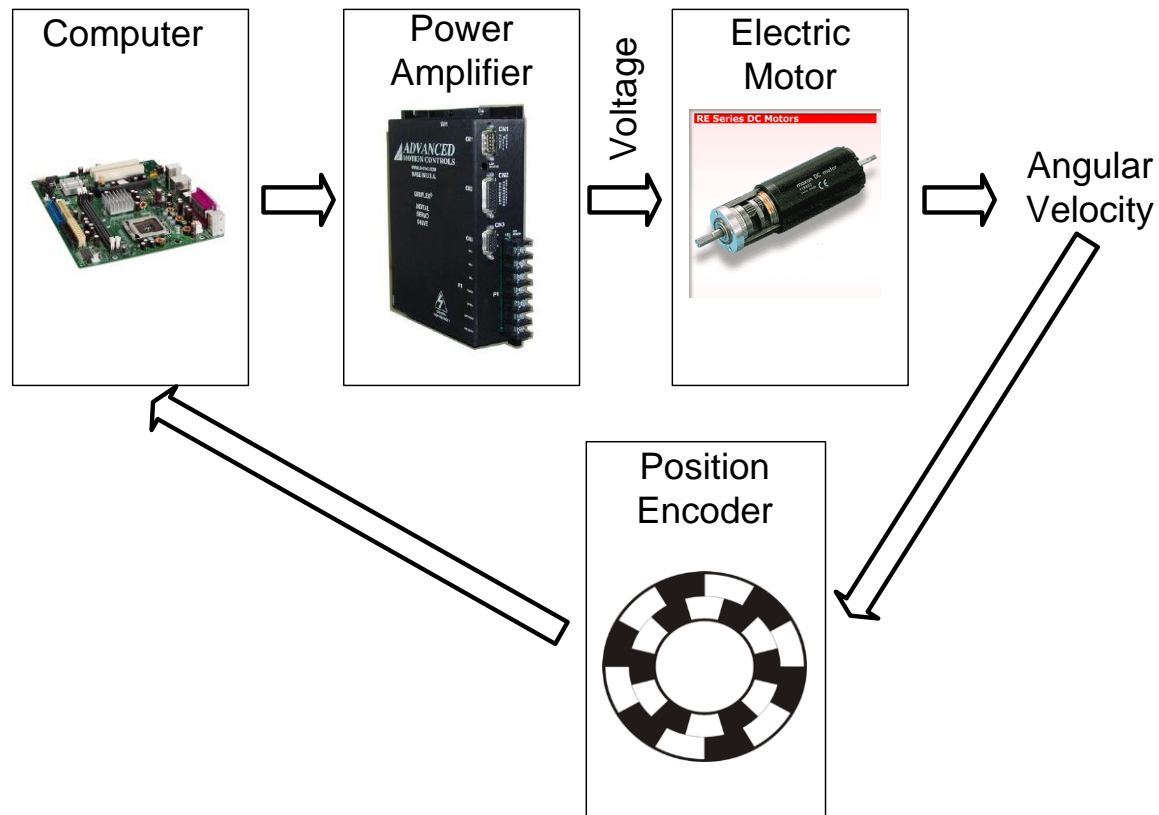
# Open-loop

- Changing load on the motor?
  - Output velocity will change!



# Closing the loop

- Let's measure the actual angular velocities.
- Compensate for changes in load by *feeding back* some information.



# Control Theory

- Roots in another science: *Cybernetics*

Cybernetics is the study of feedback and derived concepts such as communication and control in living organisms, machines and organizations.

- Expression was coined by Norbert Wiener in 1948.

*(my academic great-great-grand-father)*

Norbert Wiener (1913 Harvard University)

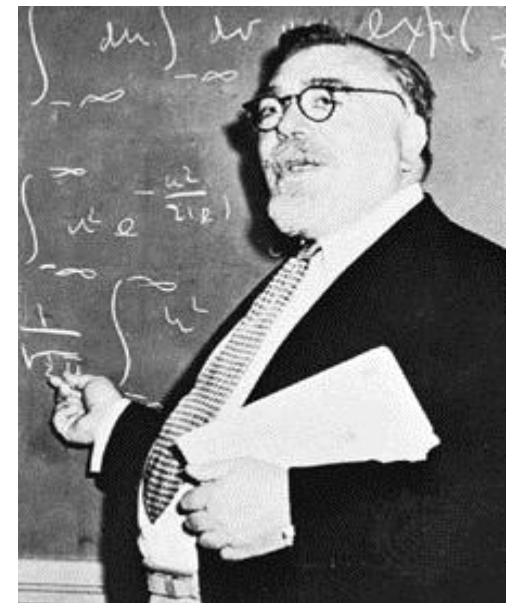
Amar Bose (1956 Massachusetts Institute of Technology)

Alan Oppenheim (1964 Massachusetts Institute of Technology)

Evangelos Milios (1986 Massachusetts Institute of Technology)

Rekleitis, Ioannis (2003 McGill University)

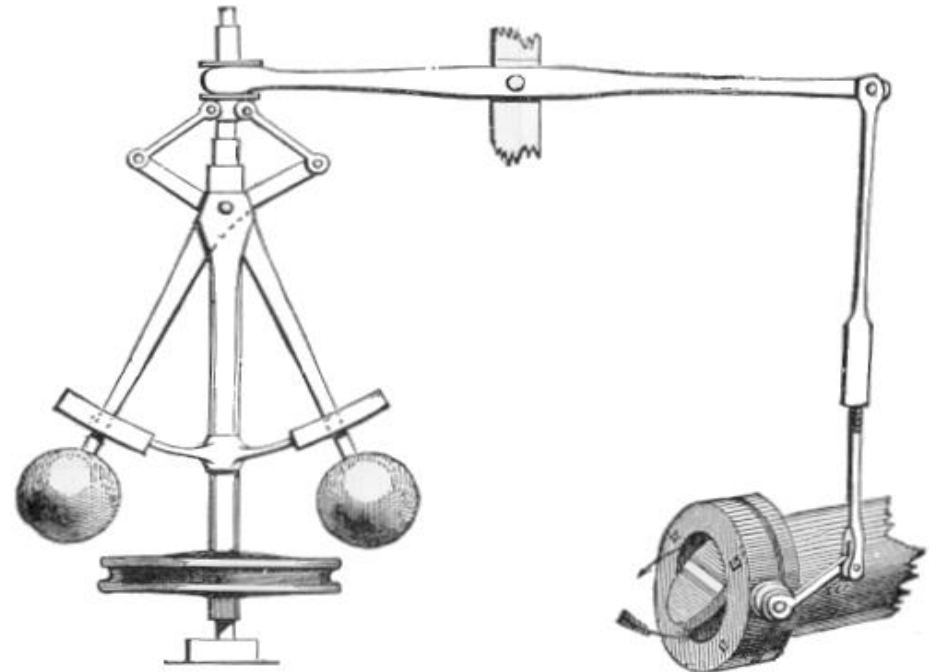
*From <http://genealogy.math.ndsu.nodak.edu/index.php>*



# Early Example of Feedback System

---

- James Watt's "Centrifugal Governor" in 1788.
- Regulates the steam engine speed.



# Other Examples

---

- Body temperature regulation
  - If cold, shiver (muscles produce heat)
  - If hot, sweat (evaporation takes away heat)
- Maintaining social peace
  - If a crime is found (sensor), the guilty party is punished (actuator).
- Cruise control in cars
- Banking industry regulation





# Why Study Control Theory

---

- Used everywhere in robotics/mechatronics
- Systematic approach to analysis and design
- Taxonomy of controller concepts



# Control Systems

---

- Regulation
  - Temperature (thermostat + heating)
- Tracking
  - robot joint rotation
- Process Optimization
  - maintaining best mix of chemicals



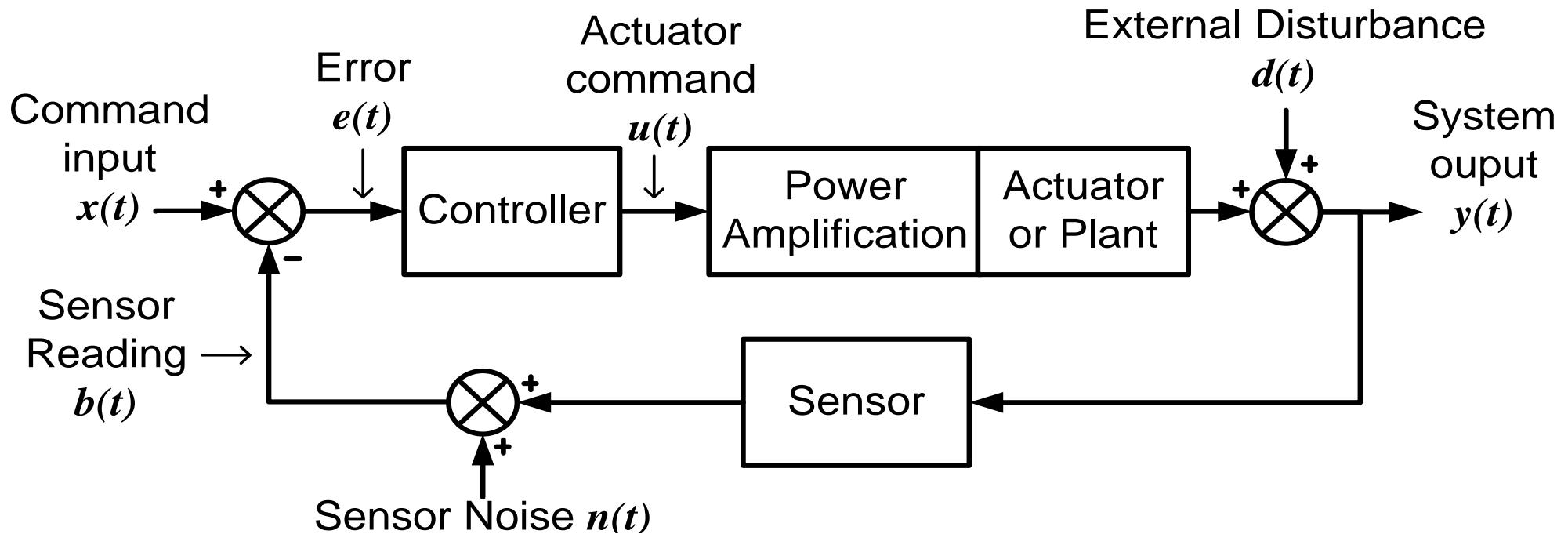
# Components of Feedback Systems

---

- Power amplification
  - Neural signal power ( $\mu\text{W}$ ) vs. muscle power output (tens of  $\text{W}$ )
  - Means it is an active system, as opposed to passive.
- Actuator
- Feedback
  - measurement (sensor)
- Error signal
- Controller



# Classic Feedback Control Diagram



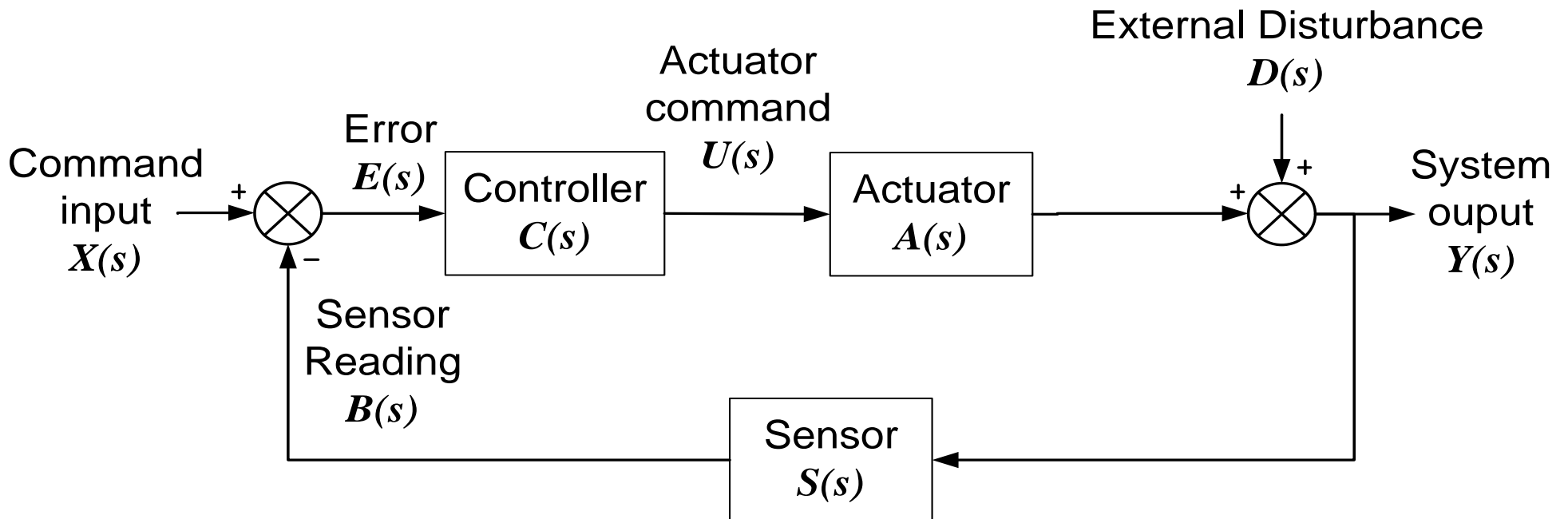
# Transfer Function

---

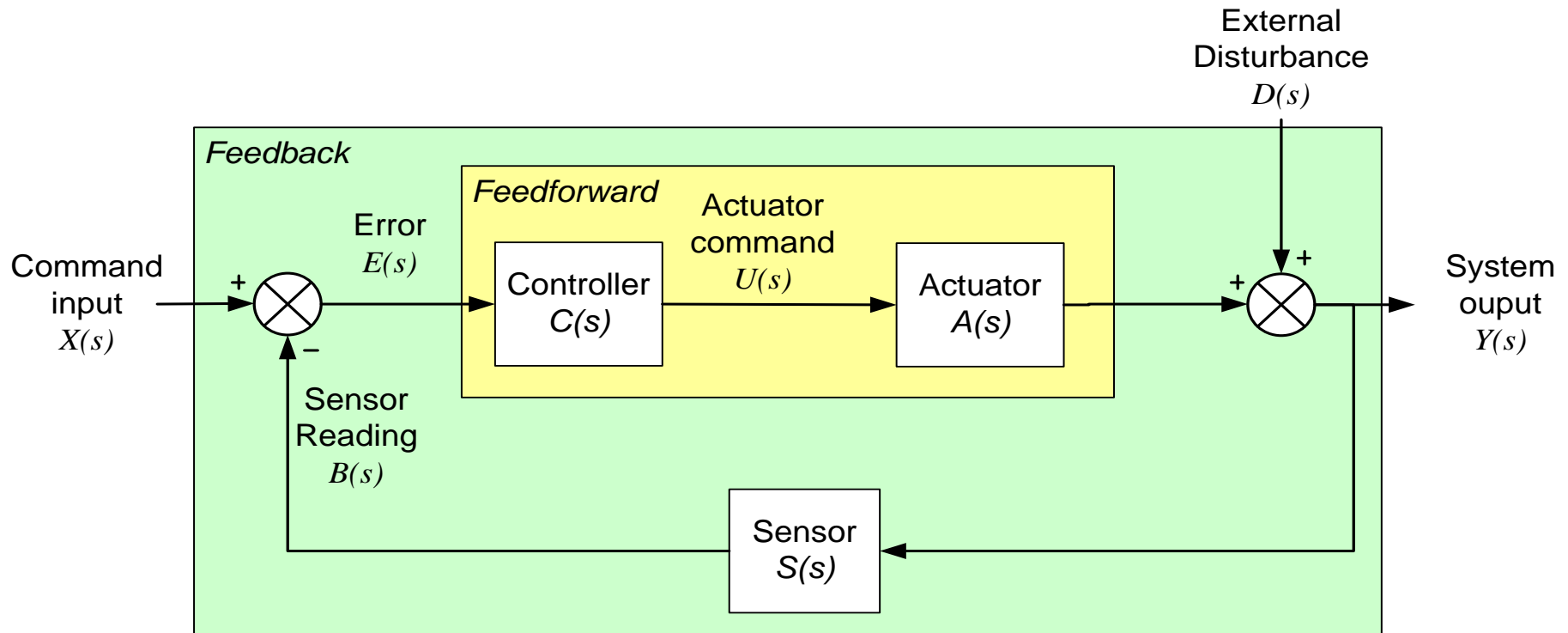
- Definition:  $H(s) = Y(s) / X(s)$
- Relates the output of a linear system to its input.
- Describes how a linear system responds to an impulse... called *impulse response*



# Laplace Transform of Classic Feedback System



# Key Transfer Functions

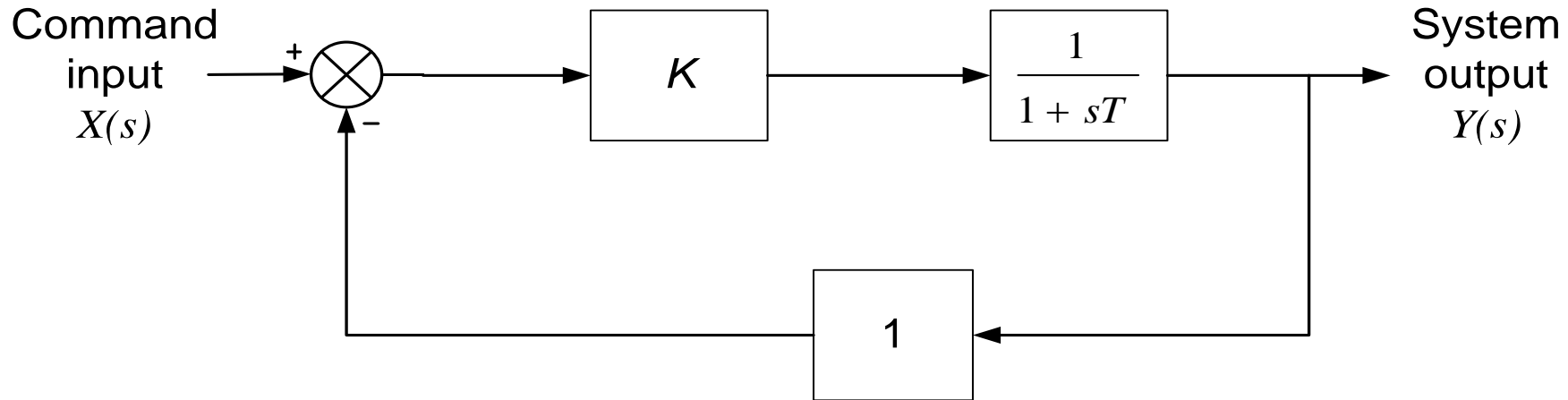


$$\text{Feedforward: } \frac{Y(s)}{E(s)} = C(s)A(s)$$

$$\text{Feedback: } \frac{Y(s)}{X(s)} = \frac{C(s)A(s)}{1 + C(s)A(s)S(s)}$$



# First Order System



$$H(s) = \frac{Y(s)}{X(s)} = \frac{K}{1 + K + sT} \quad u(t) \text{ is step function}$$

Steady state value is given by Final Value Theorem

Steady state error is  $1 - \lim_{s \rightarrow 0} sU(s)H(s)$

$$f(\infty) = \lim_{s \rightarrow 0} sF(s)$$

$$= 1 - \lim_{s \rightarrow 0} s \frac{1}{s} \left\{ \frac{K}{1 + K + sT} \right\} = 1 - \frac{K}{1 + K} = \frac{1}{1 + K} \approx \frac{1}{K}$$

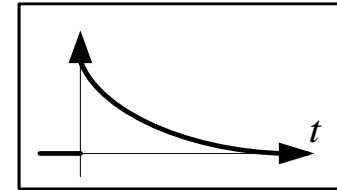




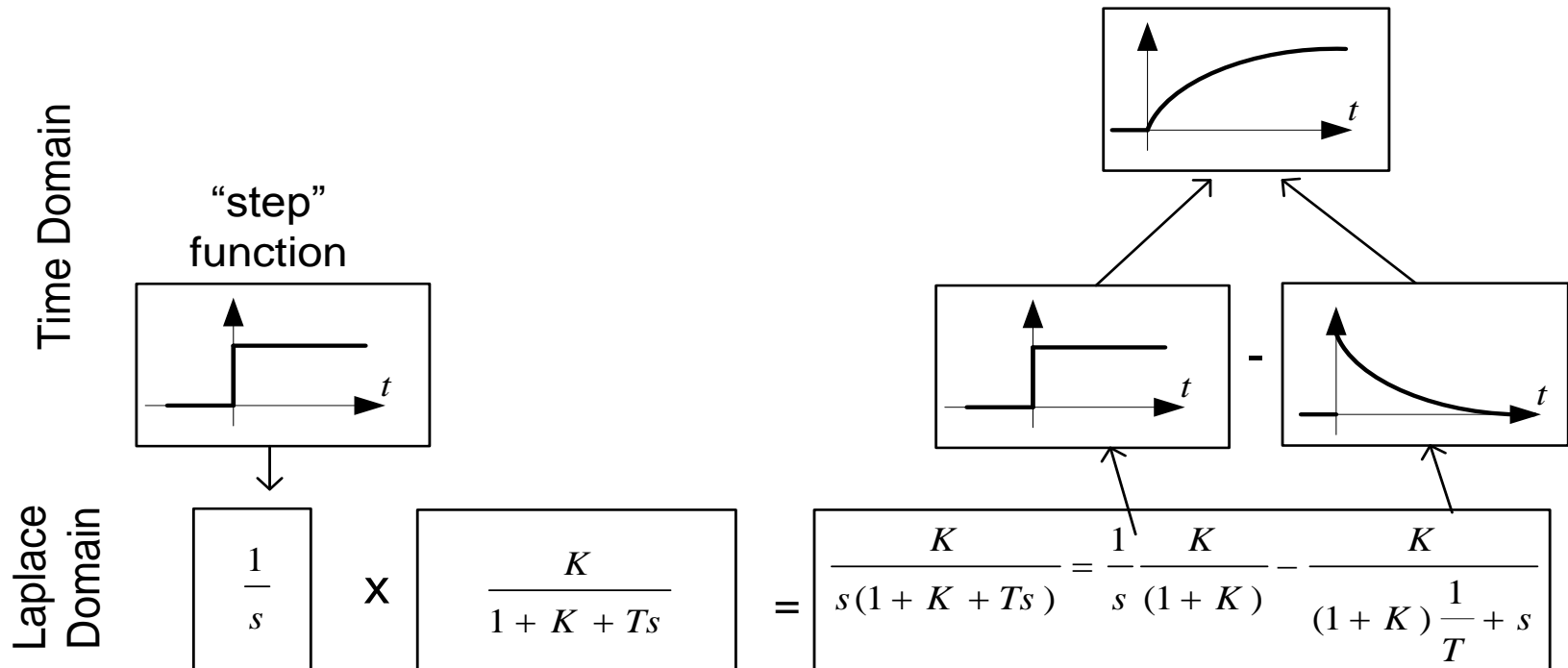
# Response of 1<sup>st</sup> Order System

- Impulse

$$\frac{K}{1 + K + sT}$$



- Step

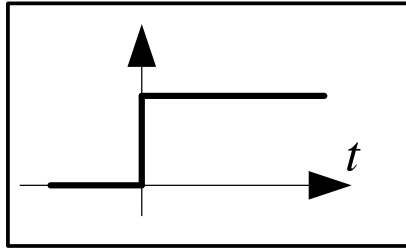


# Steady-State vs. Transient

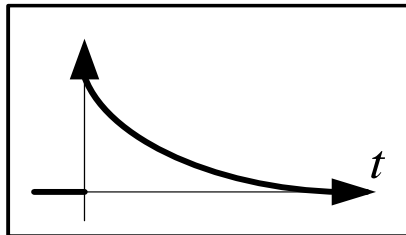
---

- Step Response illustrates how a system response can be decomposed into two components:

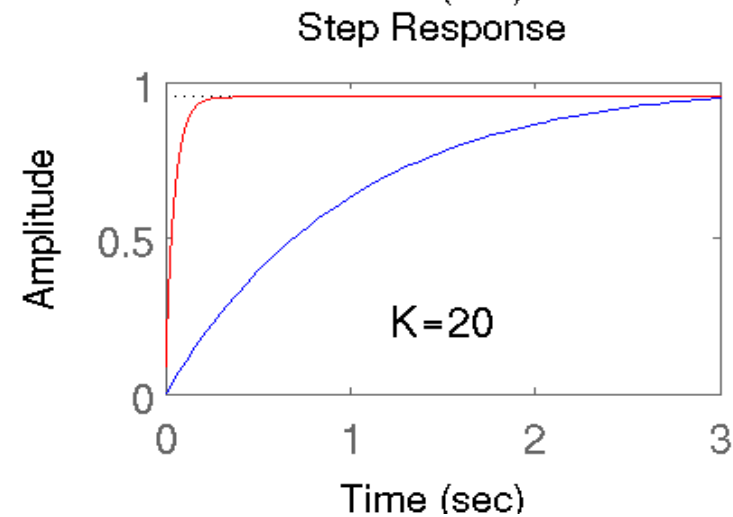
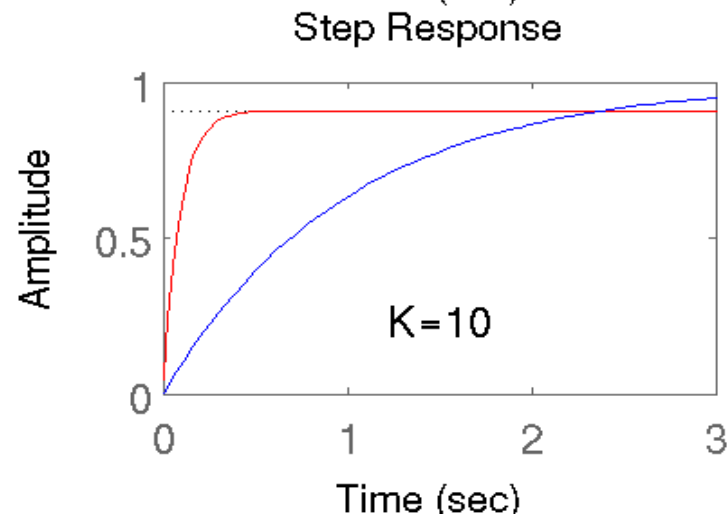
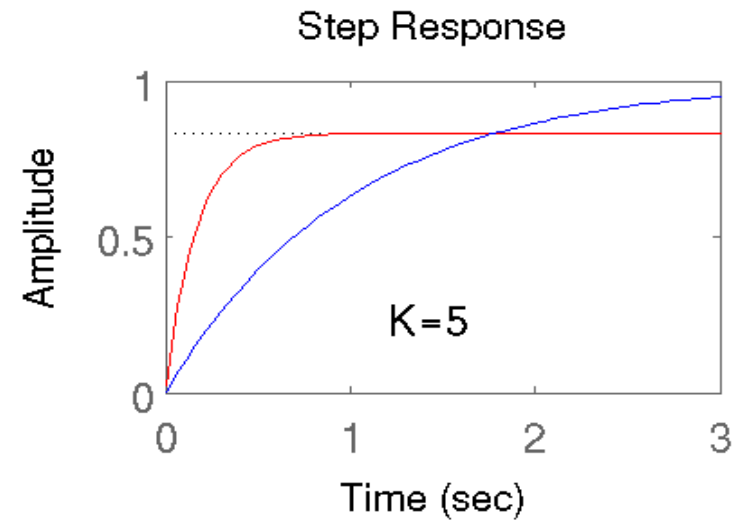
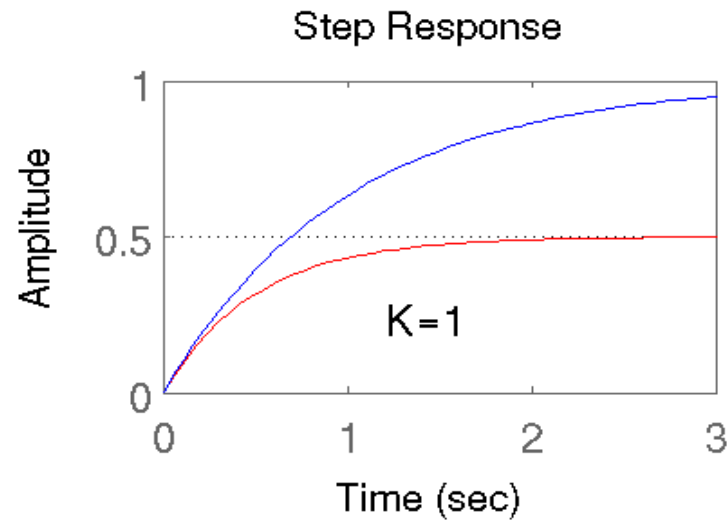
- Steady-state part:



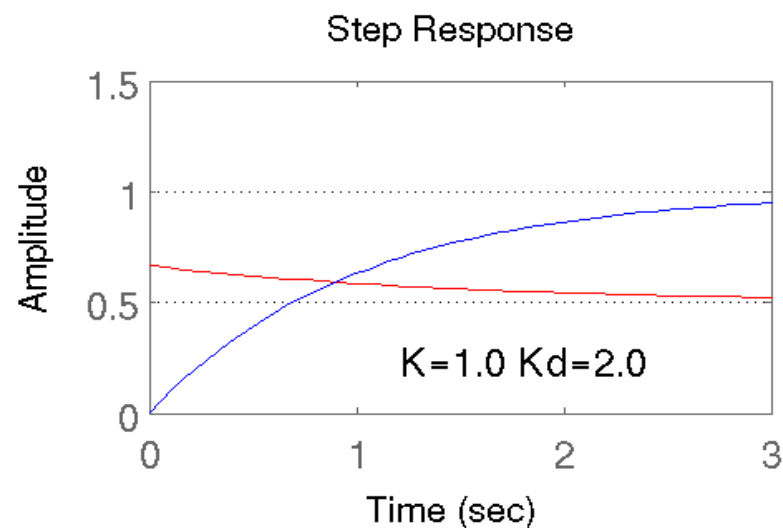
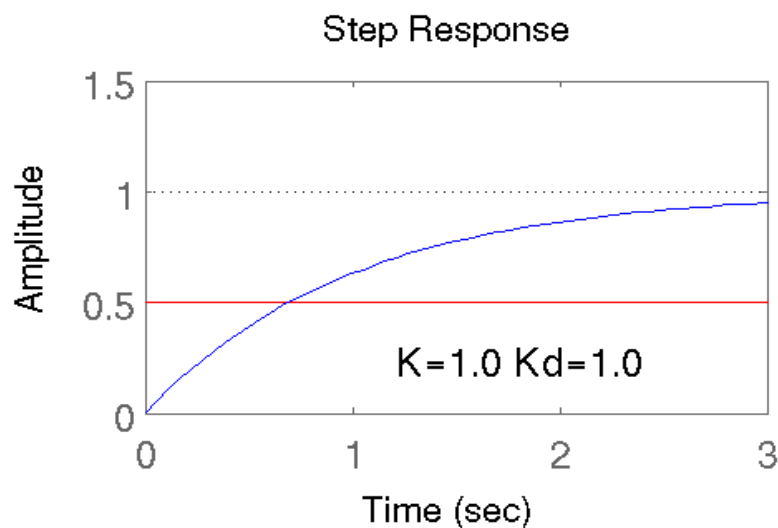
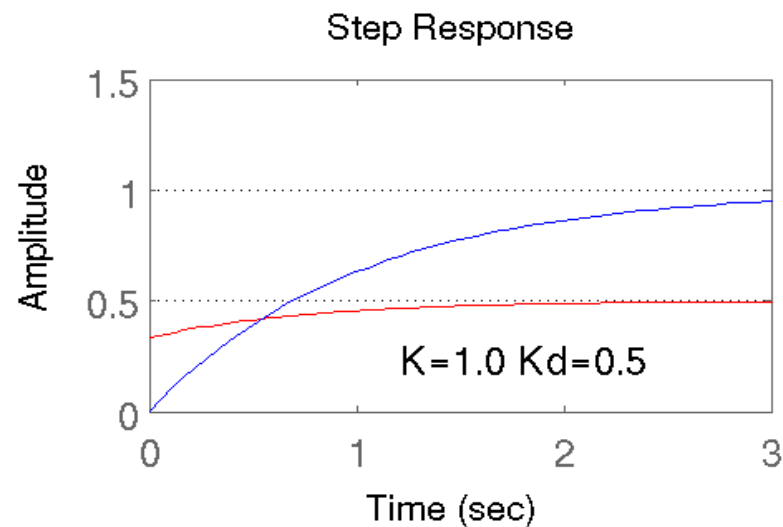
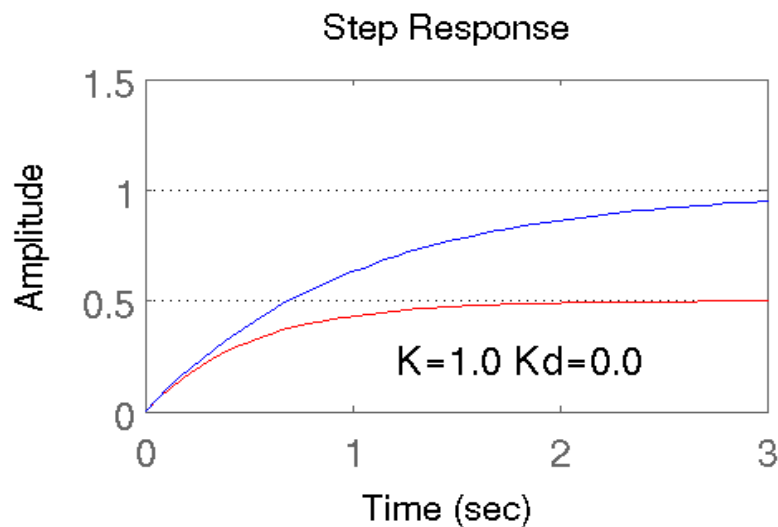
- Transient



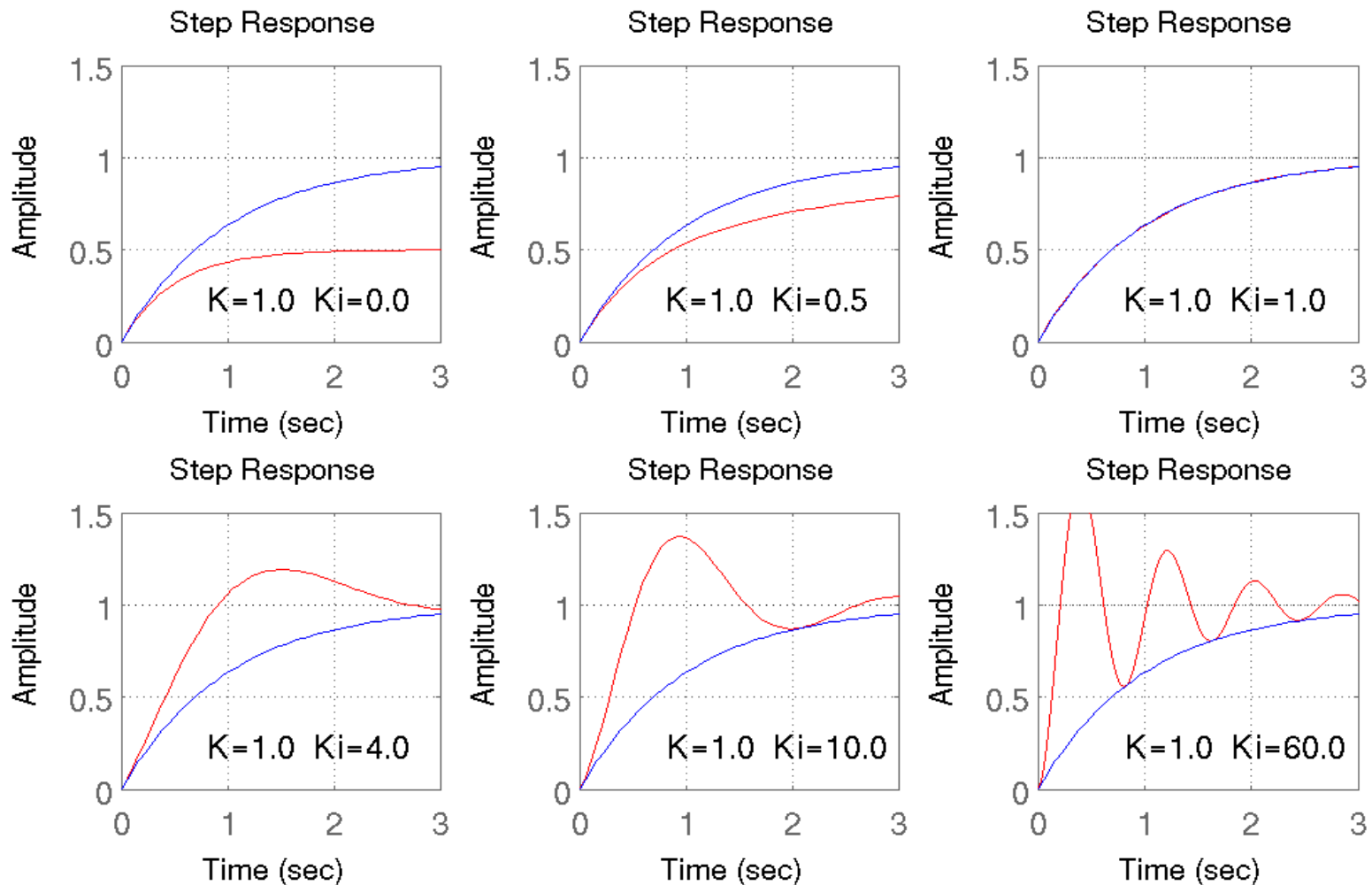
# 1<sup>st</sup> Order System: P Controller



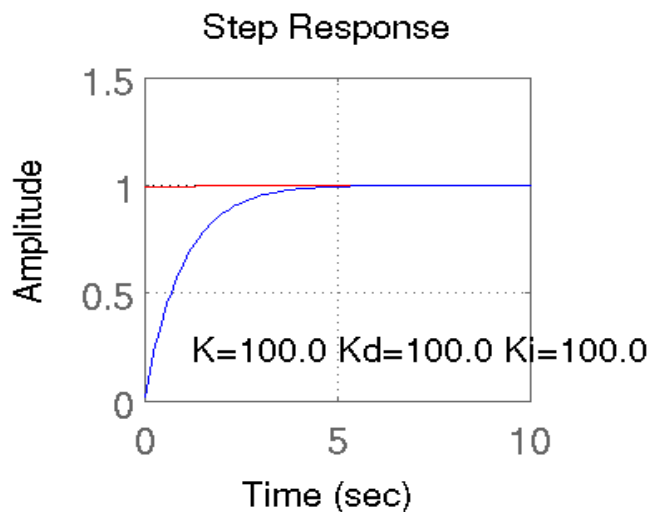
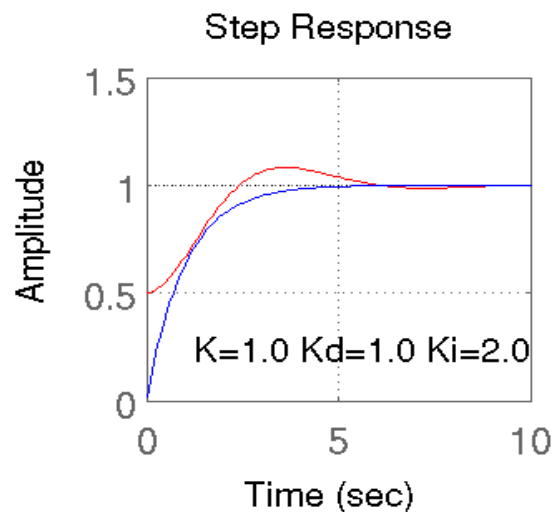
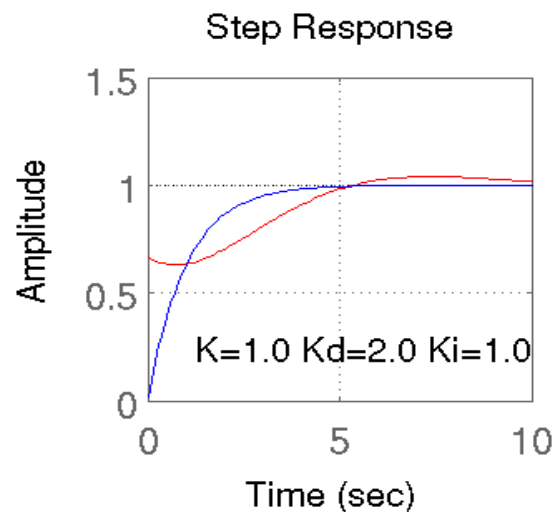
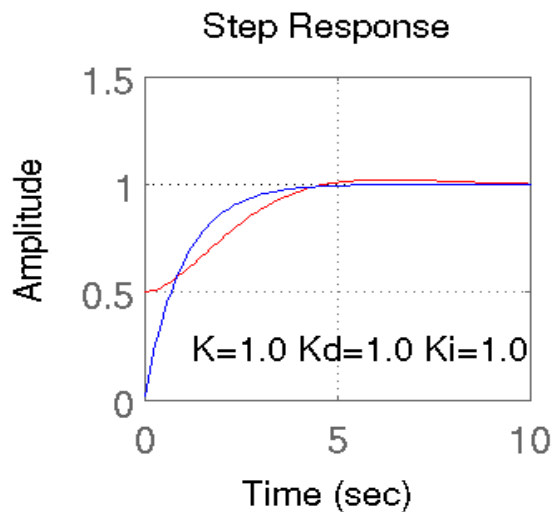
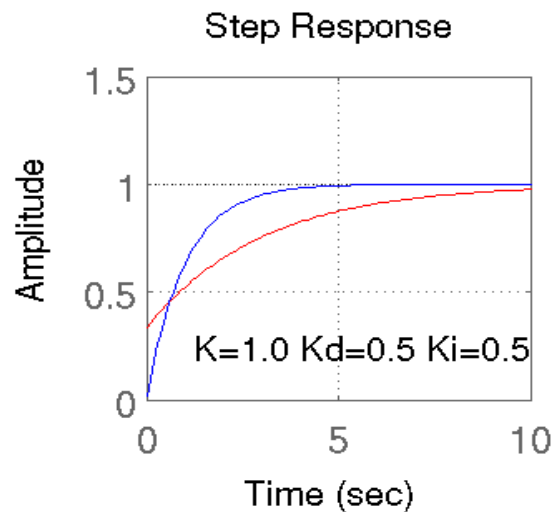
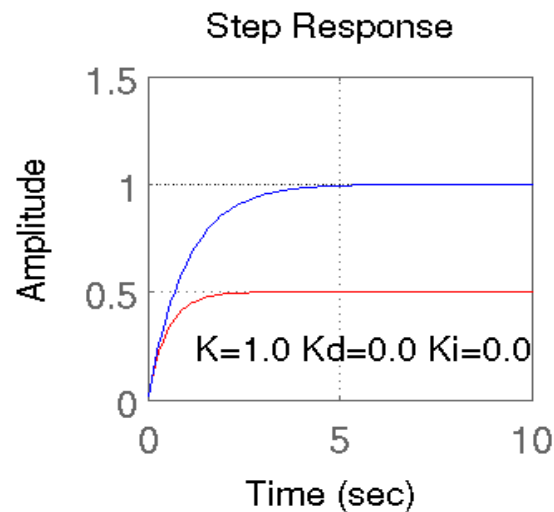
# 1<sup>st</sup> Order System: PD Controller



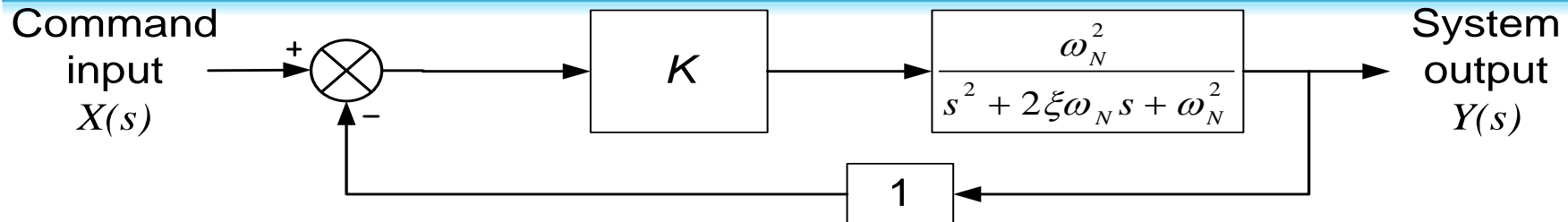
# 1<sup>st</sup> Order System: PI Controller



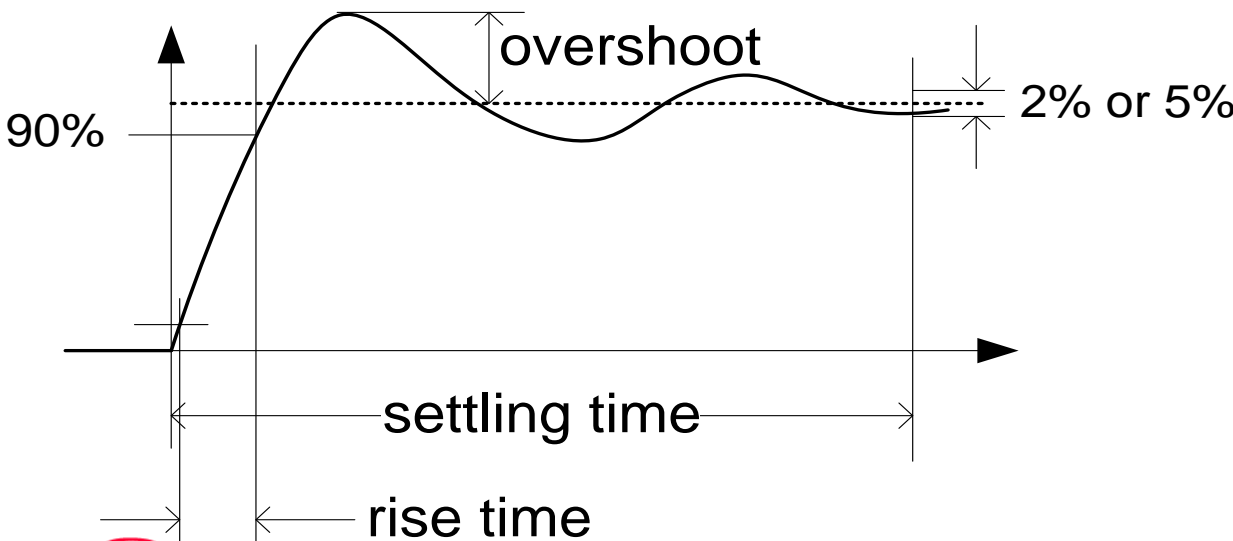
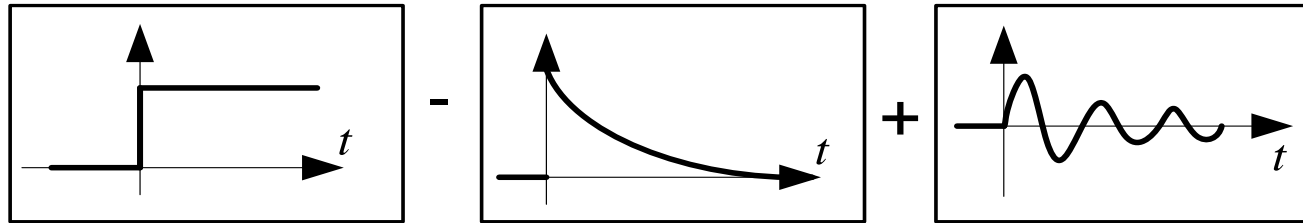
# 1<sup>st</sup> Order System: PID Controller



# Second Order Response



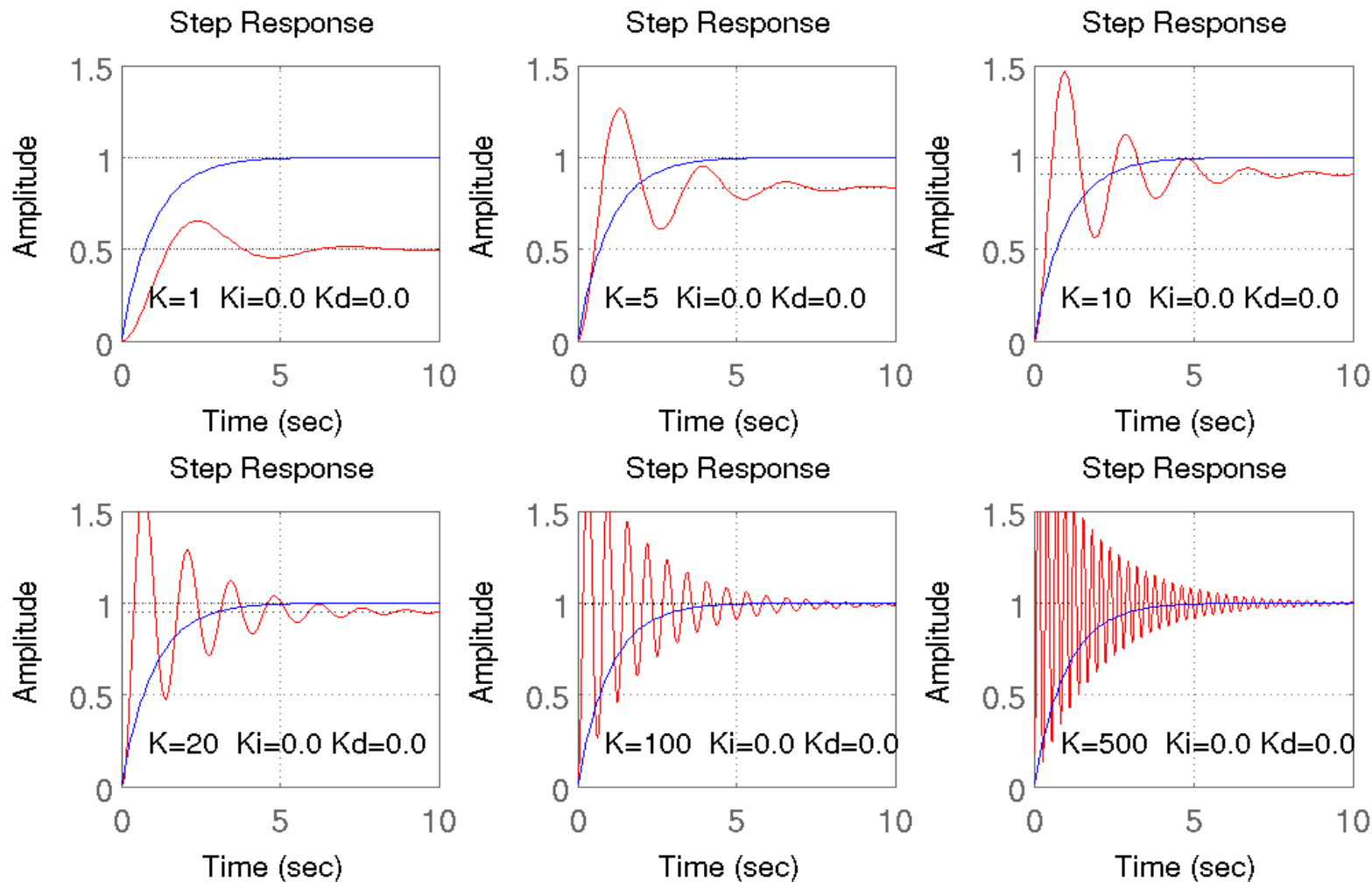
- Typical response to step input is:



**overshoot** -- % of final value exceeded at first oscillation  
**rise time** -- time to span from 10% to 90% of the final value  
**settling time** -- time to reach within 2% or 5% of the final value

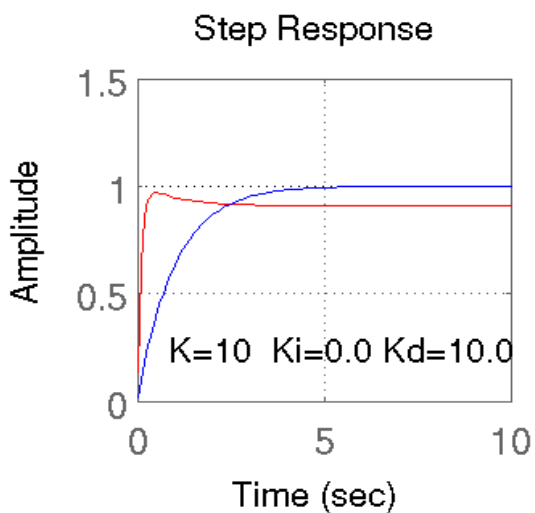
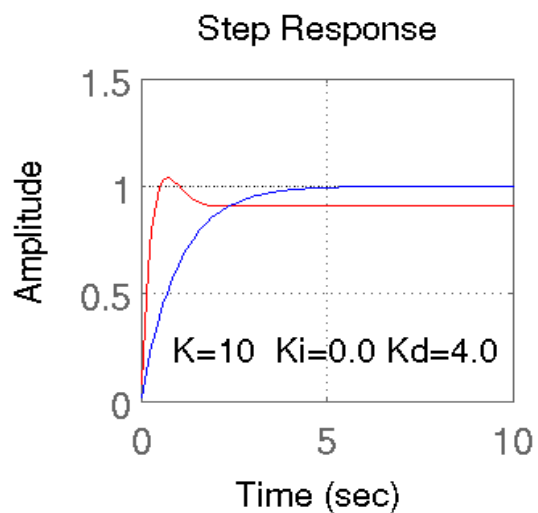
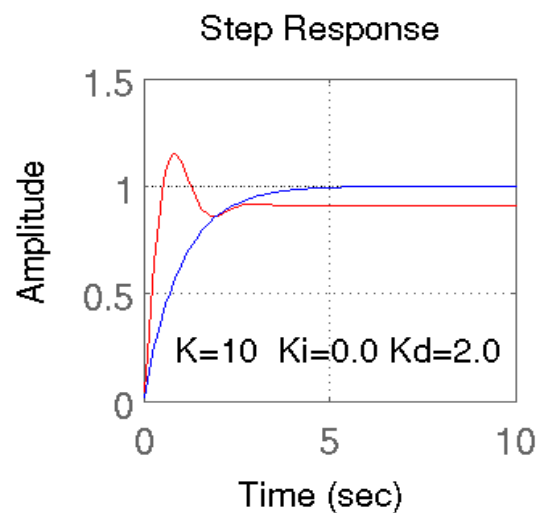
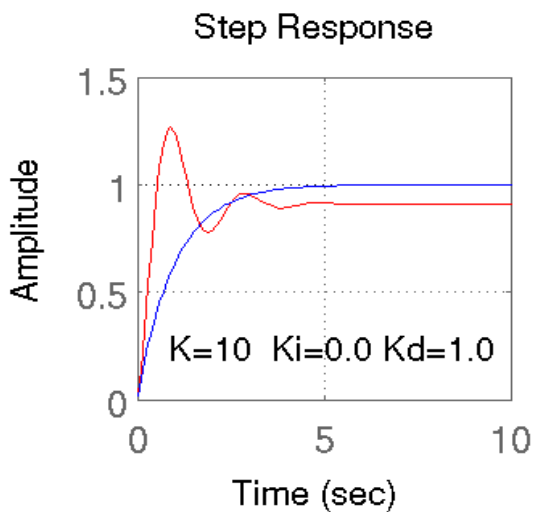
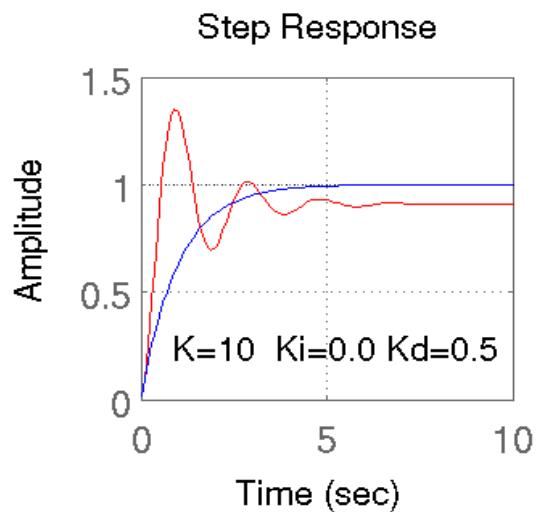
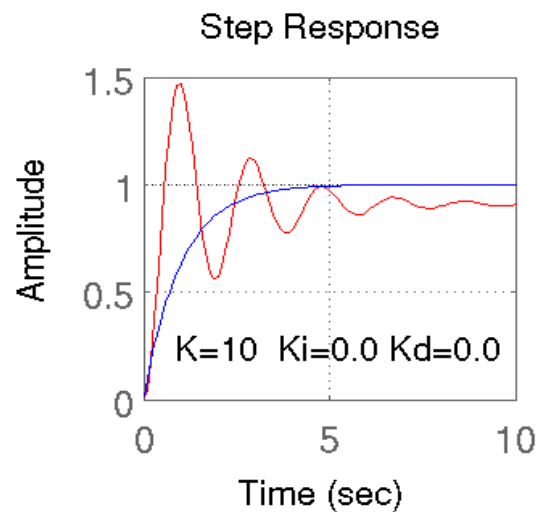


# 2<sup>nd</sup> Order System: P Controller

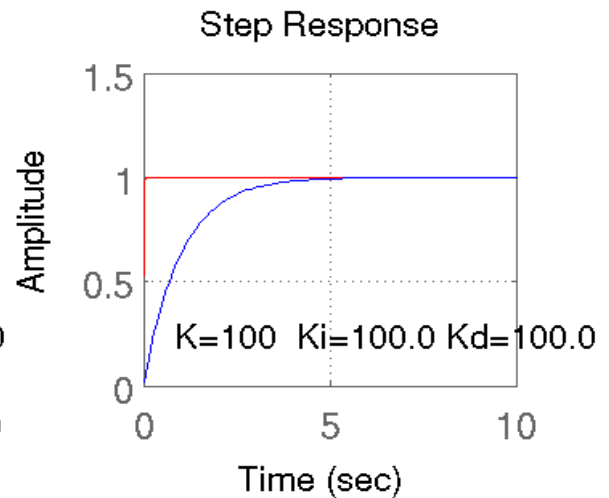
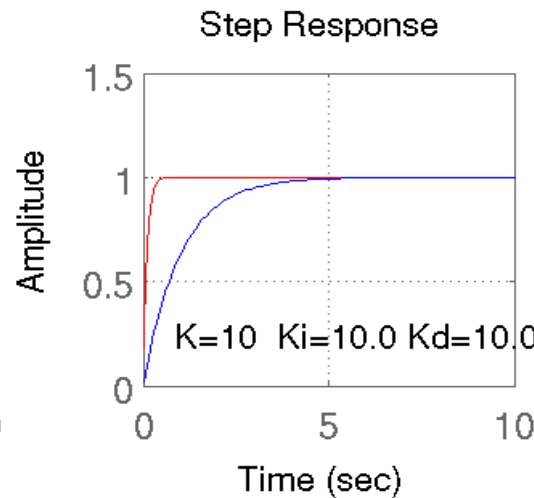
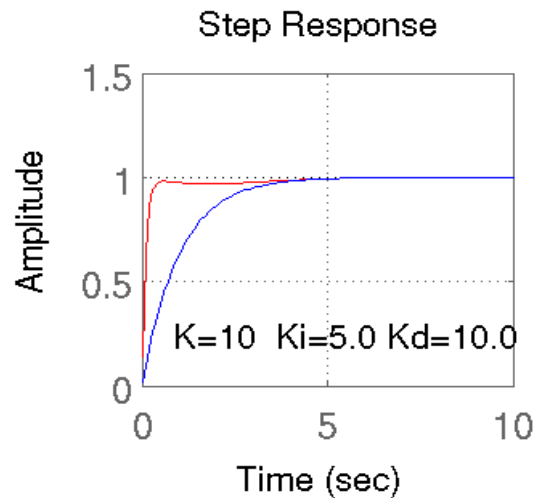
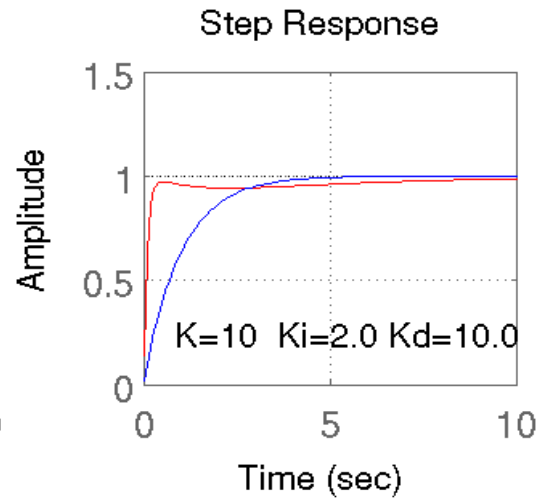
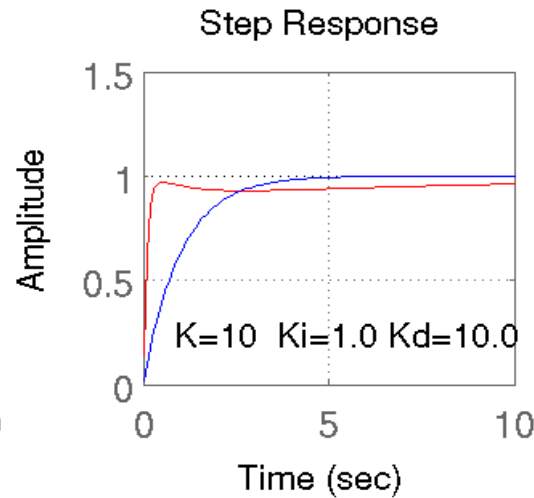
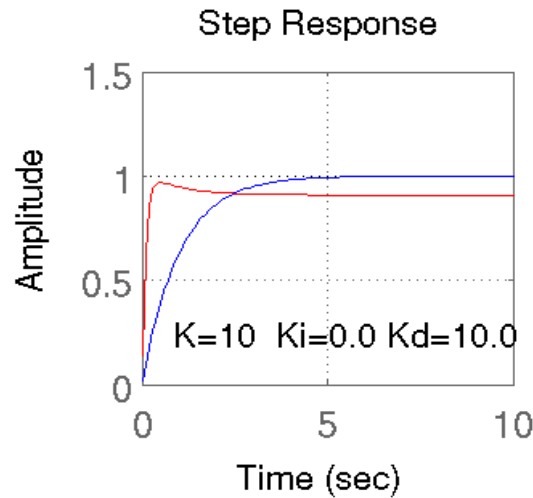




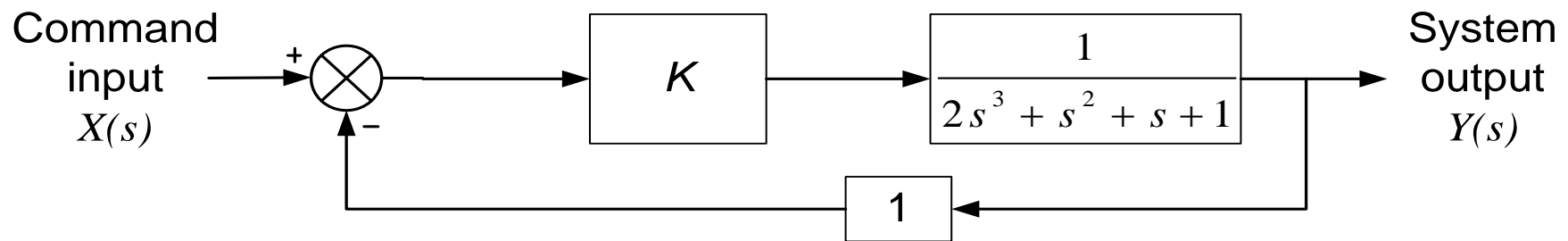
# 2<sup>nd</sup> Order System: PD Controller



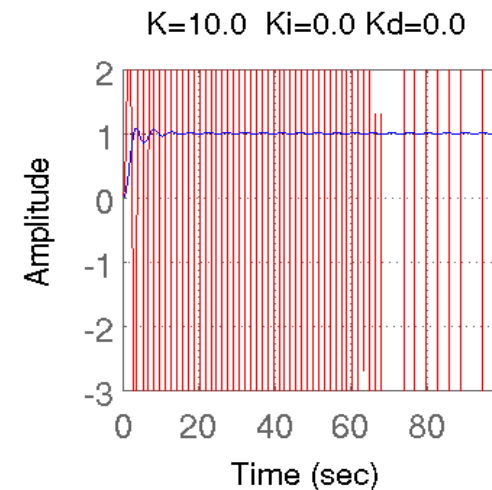
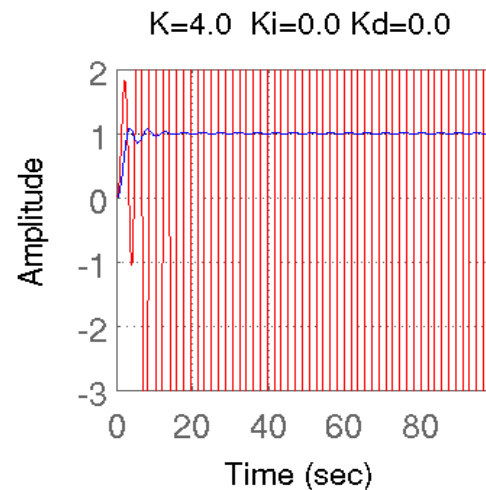
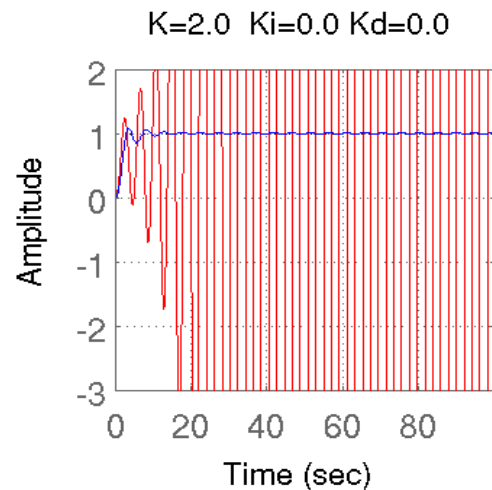
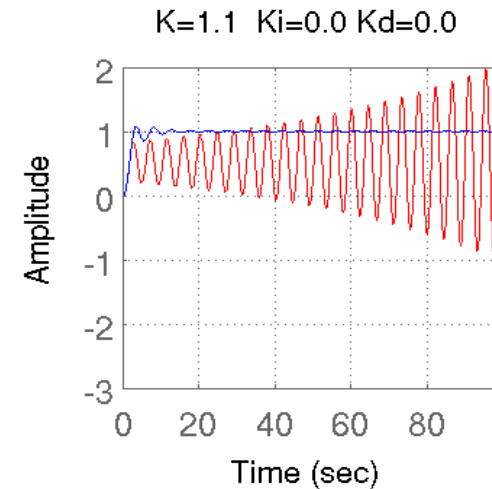
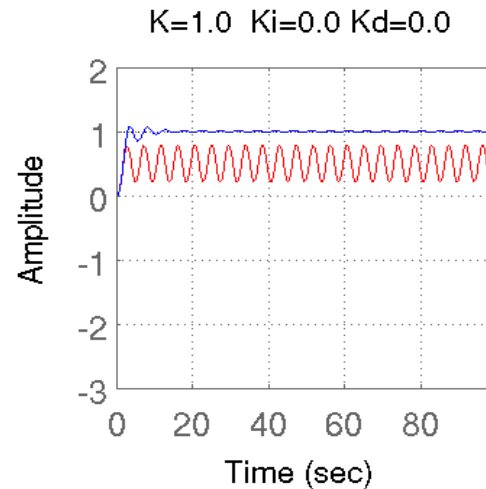
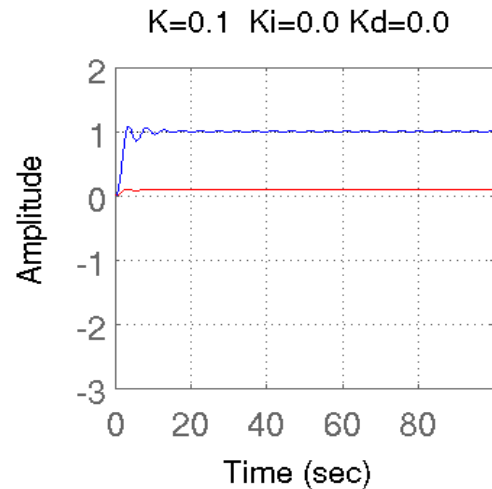
# 2<sup>nd</sup> Order System: PID Controller



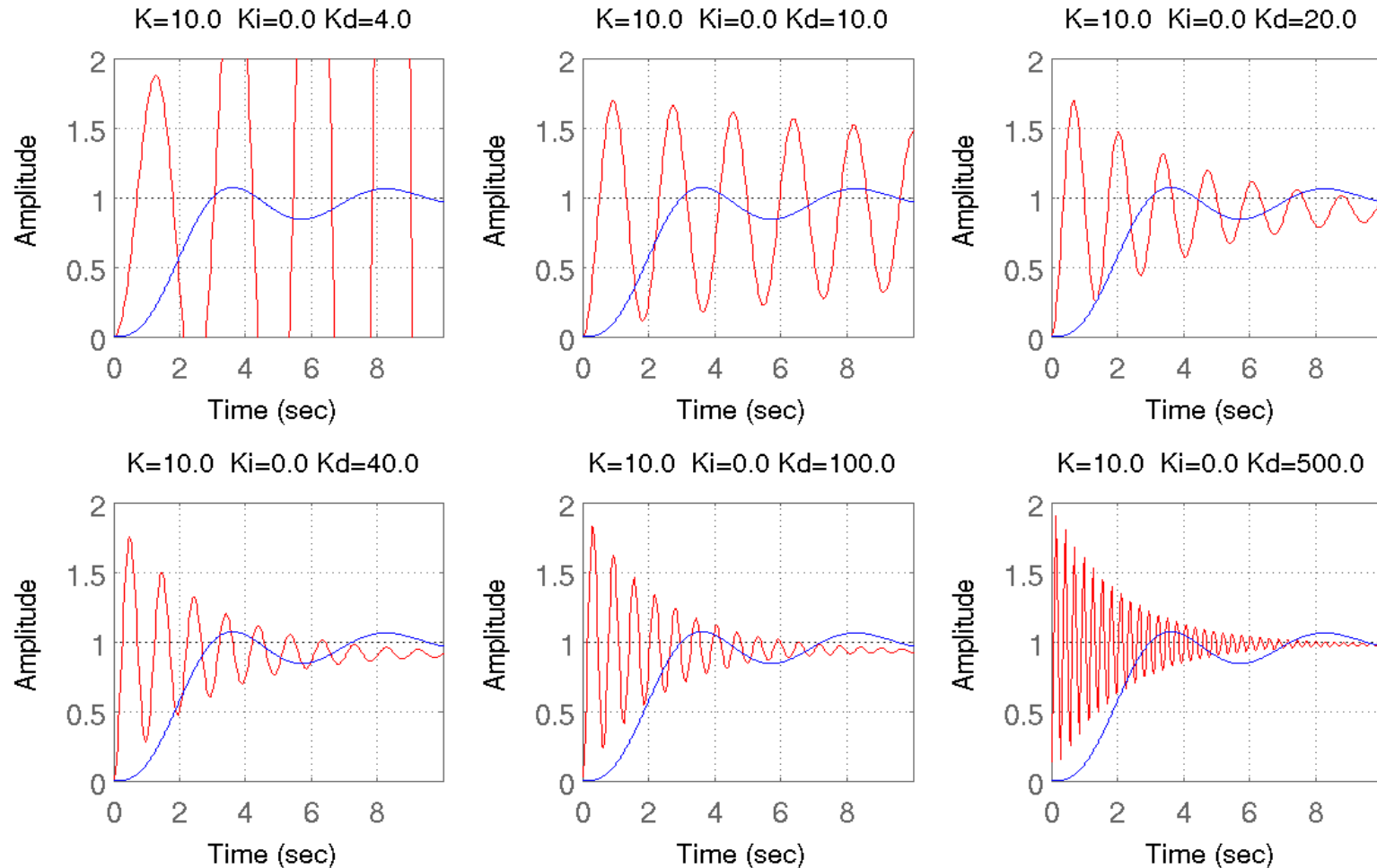
# 3<sup>rd</sup> Order System



# 3<sup>rd</sup> Order System: P Controller

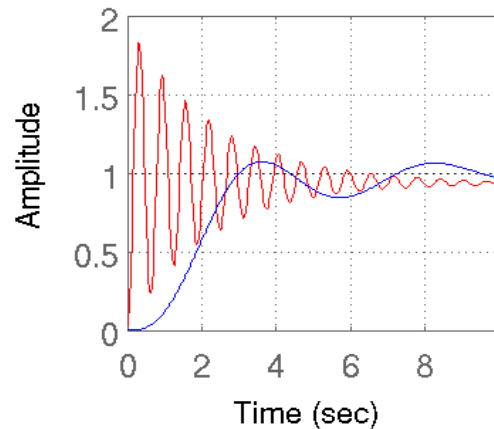


# 3<sup>rd</sup> Order System: PD Controller

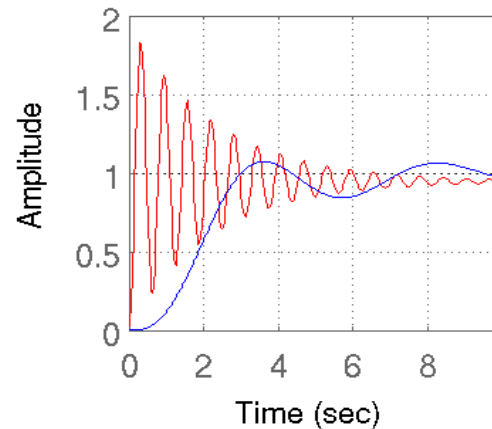


# 3<sup>rd</sup> Order System: PID Controller

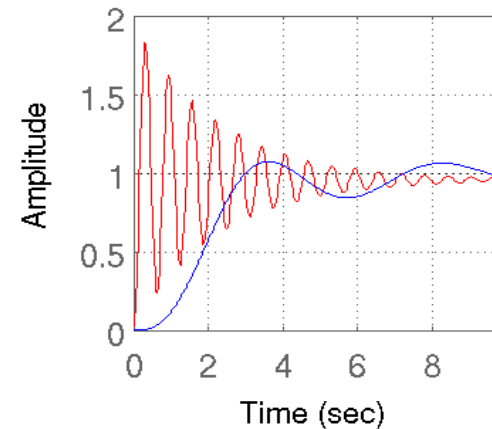
$K=10.0$   $K_i=0.0$   $K_d=100.0$



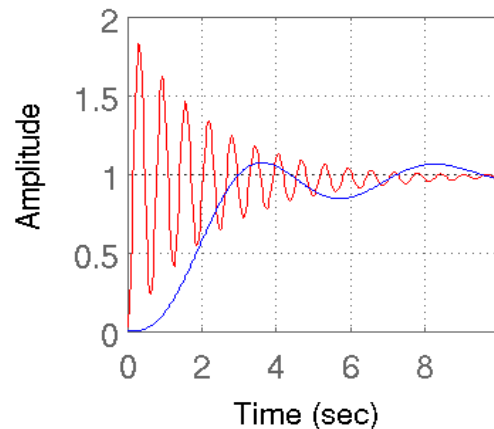
$K=10.0$   $K_i=1.0$   $K_d=100.0$



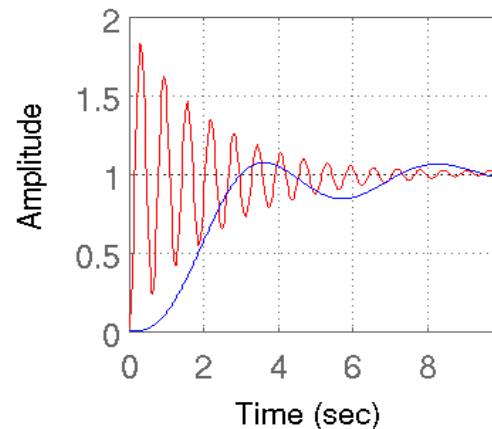
$K=10.0$   $K_i=2.0$   $K_d=100.0$



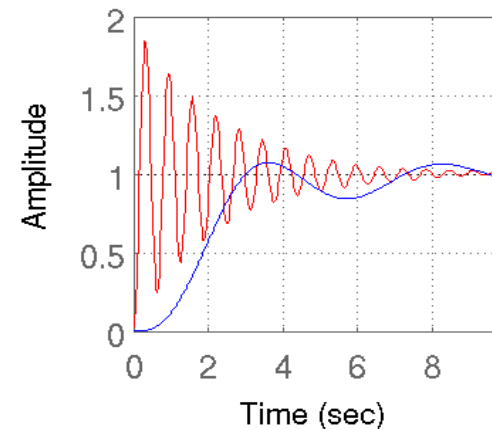
$K=10.0$   $K_i=5.0$   $K_d=100.0$



$K=10.0$   $K_i=10.0$   $K_d=100.0$



$K=10.0$   $K_i=100.0$   $K_d=100.0$



# Basic Controller Functions

---

Proportional control:  $u(t) = K_p e(t)$   $\frac{U(s)}{E(s)} = K_p$

Integral control:  $u(t) = K_i \int_0^t e(t) dt$   $\frac{U(s)}{E(s)} = \frac{K_i}{s}$

Differential control:  $u(t) = K_d \frac{d}{dt} e(t)$   $\frac{U(s)}{E(s)} = K_d s$



# Effect of Controller Functions

---

- Proportional Action
  - Simplest Controller Function
- Integral Action
  - Eliminates steady-state error
  - Can cause oscillations
- Derivative Action (“rate control”)
  - Effective in transient periods
  - Provides faster response (higher sensitivity)
  - Never used alone





# PID Tuning

## How to get the PID parameter values ?

- (1) If we know the transfer function, analytical methods can be used (e.g., root-locus method) to meet the transient and steady-state specs.
- (2) When the system dynamics are not precisely known, we must resort to experimental approaches.

## Ziegler-Nichols Rules for Tuning PID Controller:

Using only Proportional control, turn up the gain until the system oscillates w/o dying down, i.e., is marginally stable. Assume that  $K$  and  $P$  are the resulting gain and oscillation period, respectively.

Then, use

for P control

$$K_p = 0.5 K$$

for PI control

$$K_p = 0.45 K$$
$$K_i = 1.2 / P$$

for PID control

$$K_p = 0.6 K$$
$$K_i = 2.0 / P$$
$$K_d = P / 8.0$$

Ziegler-Nichols Tuning for second or higher order systems



# Advanced Control Topics

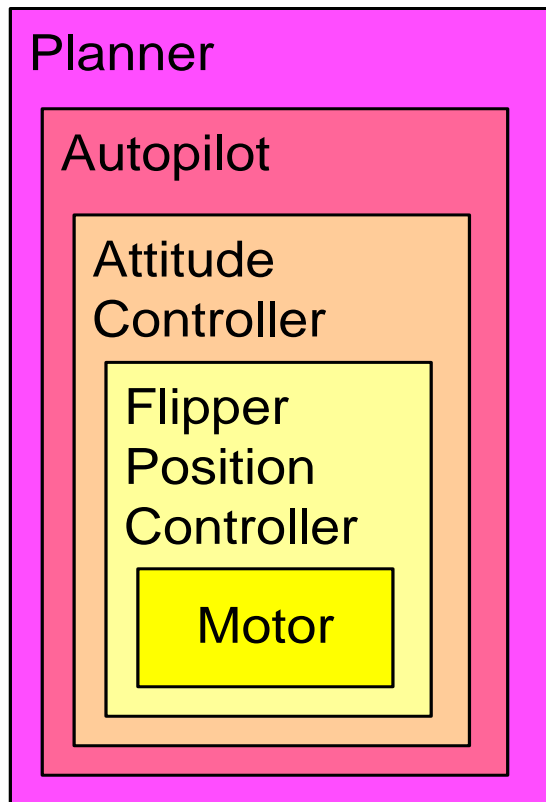
---

- Adaptive Control
  - Controller changes over time (adapts).
- MIMO Control
  - Multiple inputs and/or outputs.
- Predictive Control
  - You measure disturbance and react before measuring change in system output.
- Optimal Control
  - Controller minimizes a cost function of error and control energy.
- Nonlinear systems
  - Neuro-fuzzy control.
  - Challenging to derive analytic results.



# Layered Approach to Control

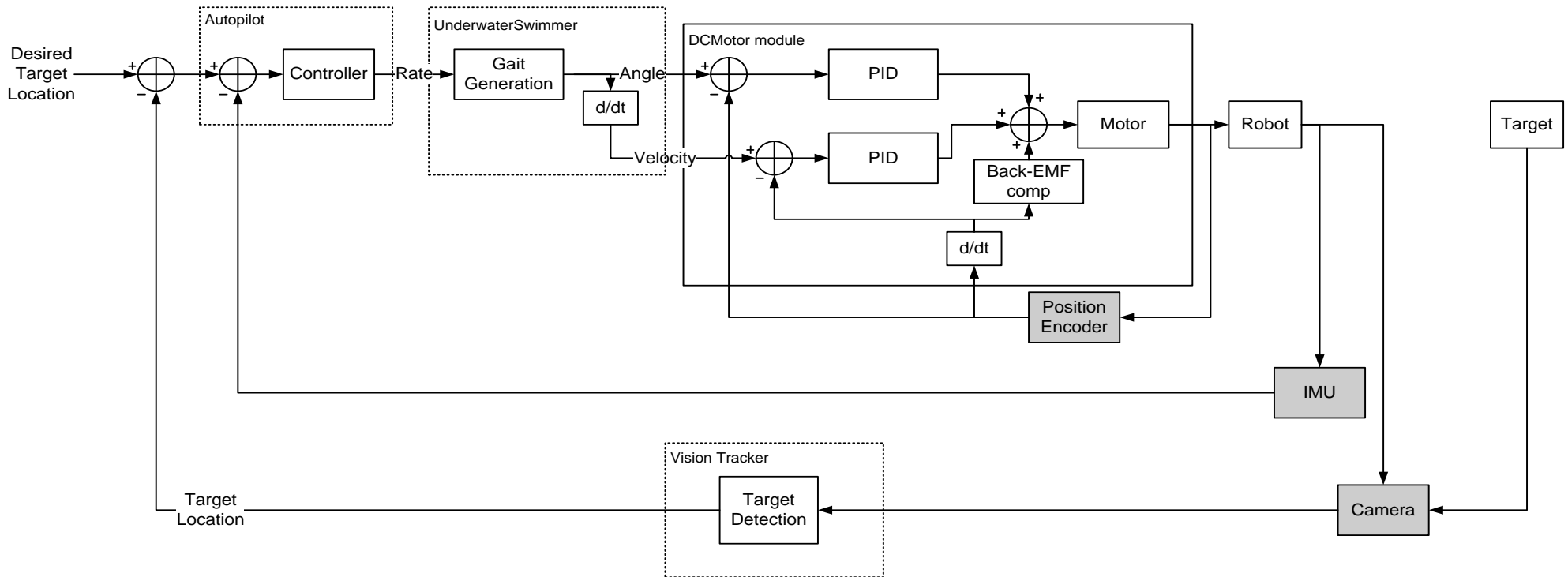
- Robotic systems often have a layered approach to control:



[movie](#)



# Multiple Loops



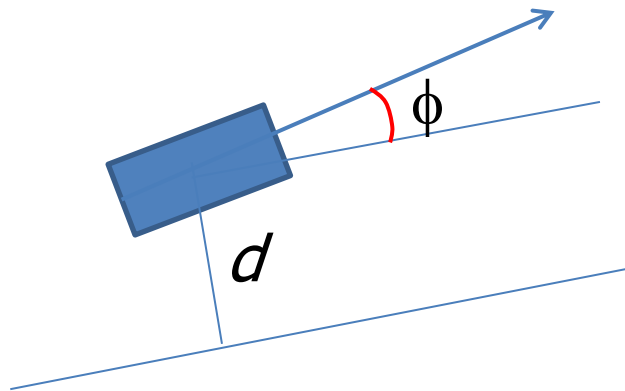
Inner loops are generally “faster” than outer loop.



# Line Following

---

- Consider a 2D robot with pose  $(x, y, \theta)$  following a line. The robot is moving with constant speed  $v$ , and we control the angular velocity  $\omega$  (turnrate).



# Calculating the error

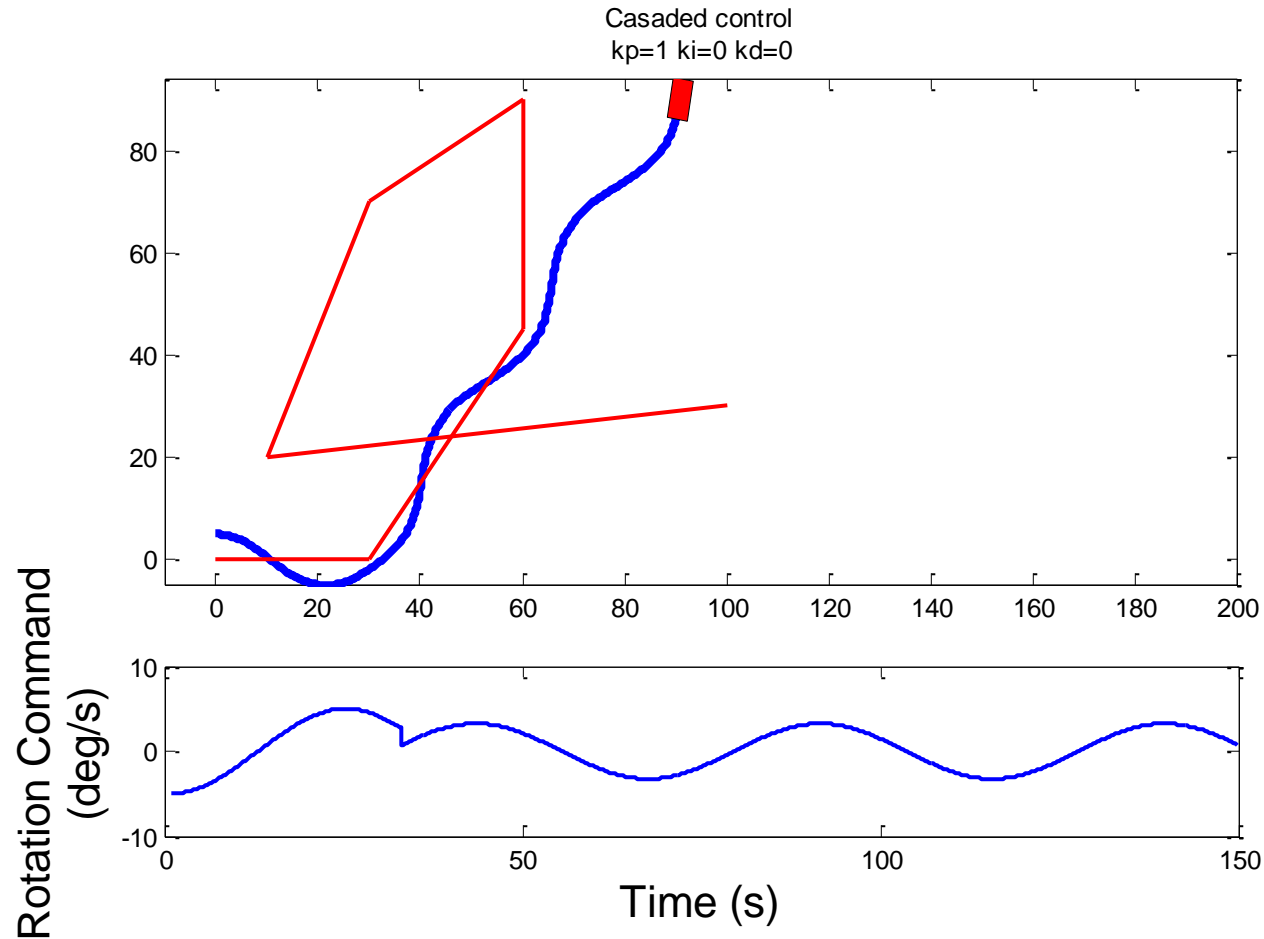
---

- Consider two error measures,
  - $d$
  - $\phi$
- The orientation error is used to estimate when we are heading too much in the wrong direction
  - if  $|\phi| < 90^\circ$   $\omega$  is bounded
- The distance error is what we feed to our controller.



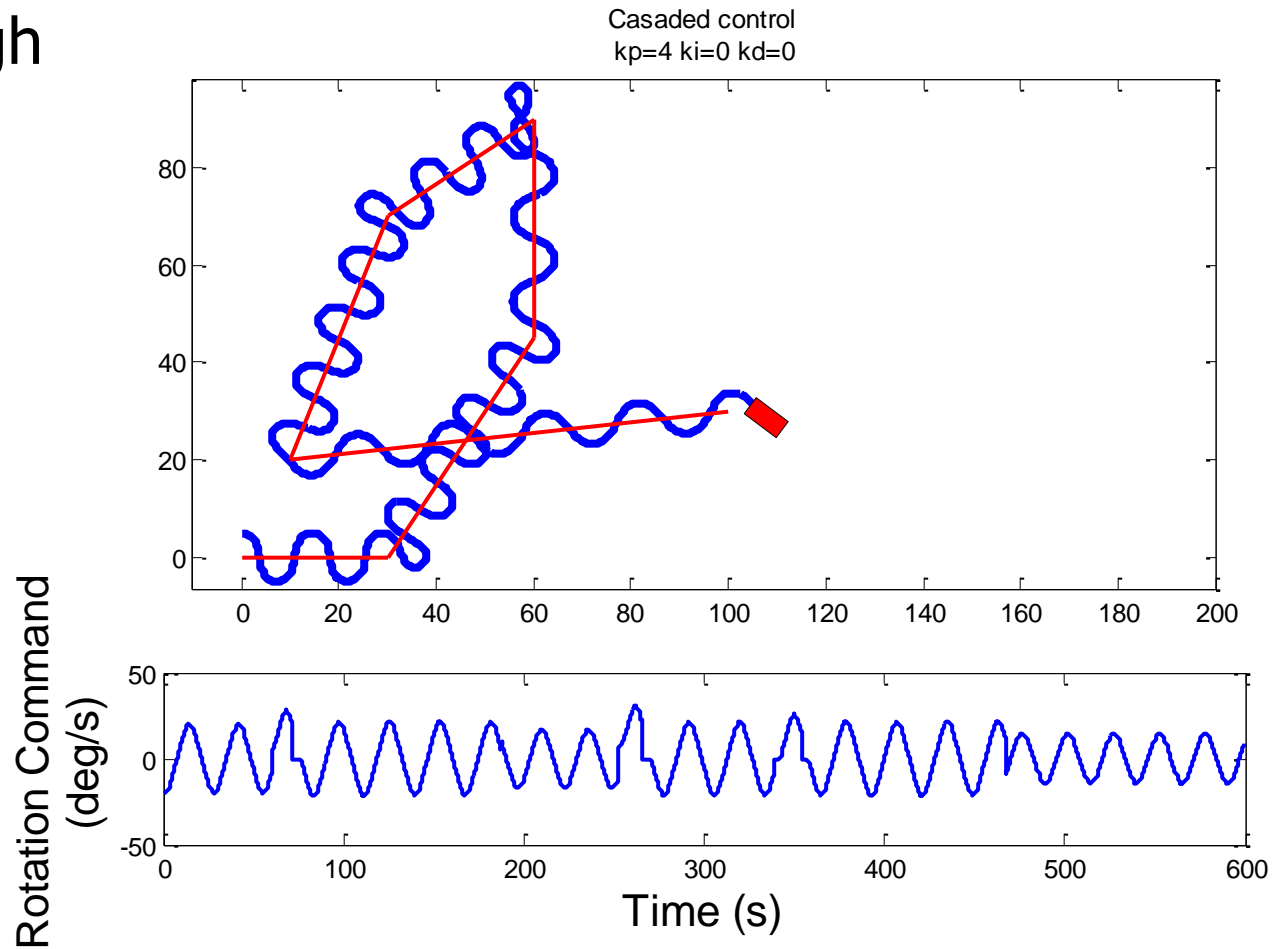
# P-Controller

Gain too low



# P-Controller

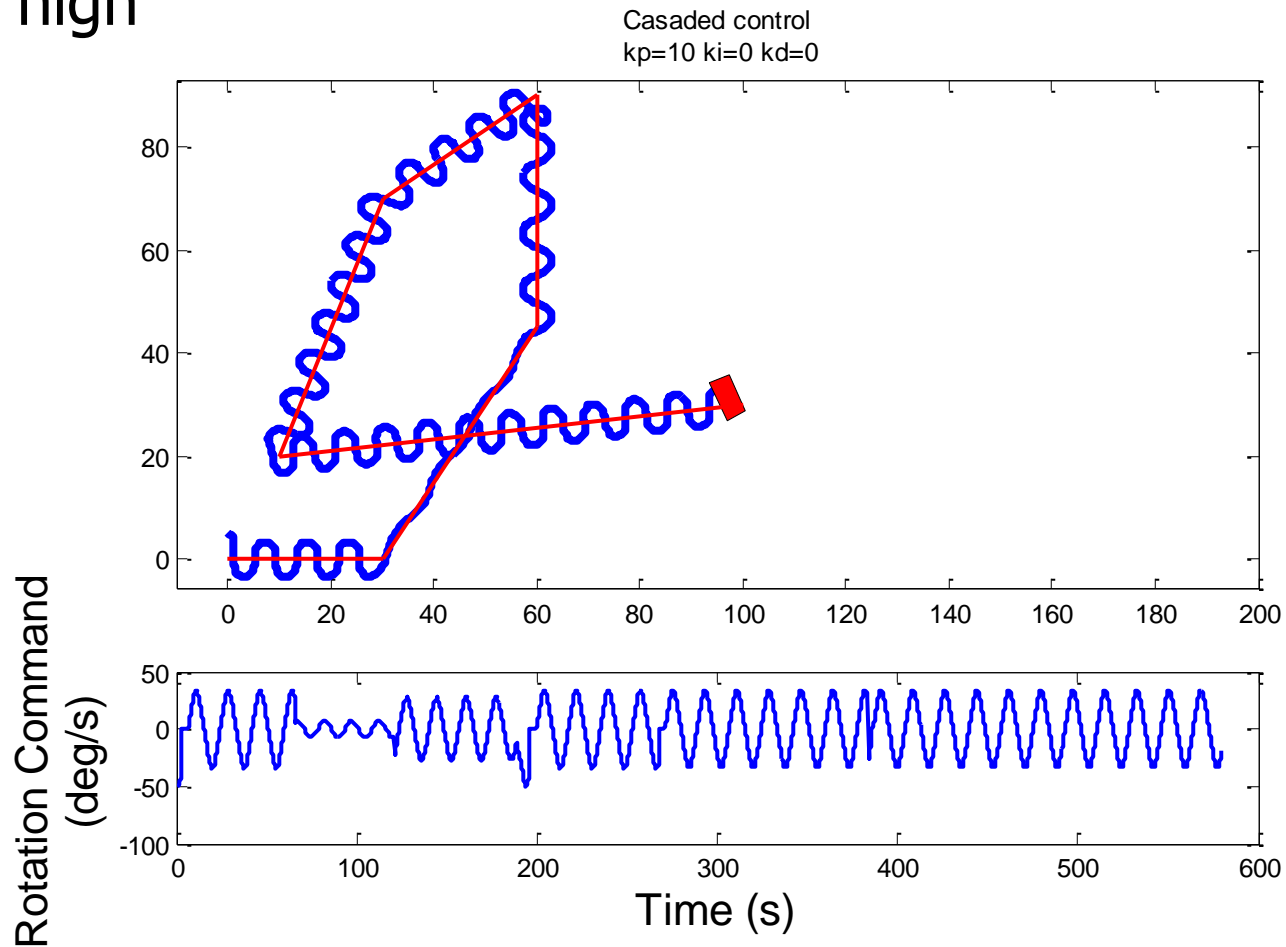
Gain high





# P-Controller

Gain too high



# Pd-Controller

---

- We are using also the derivative of the error:

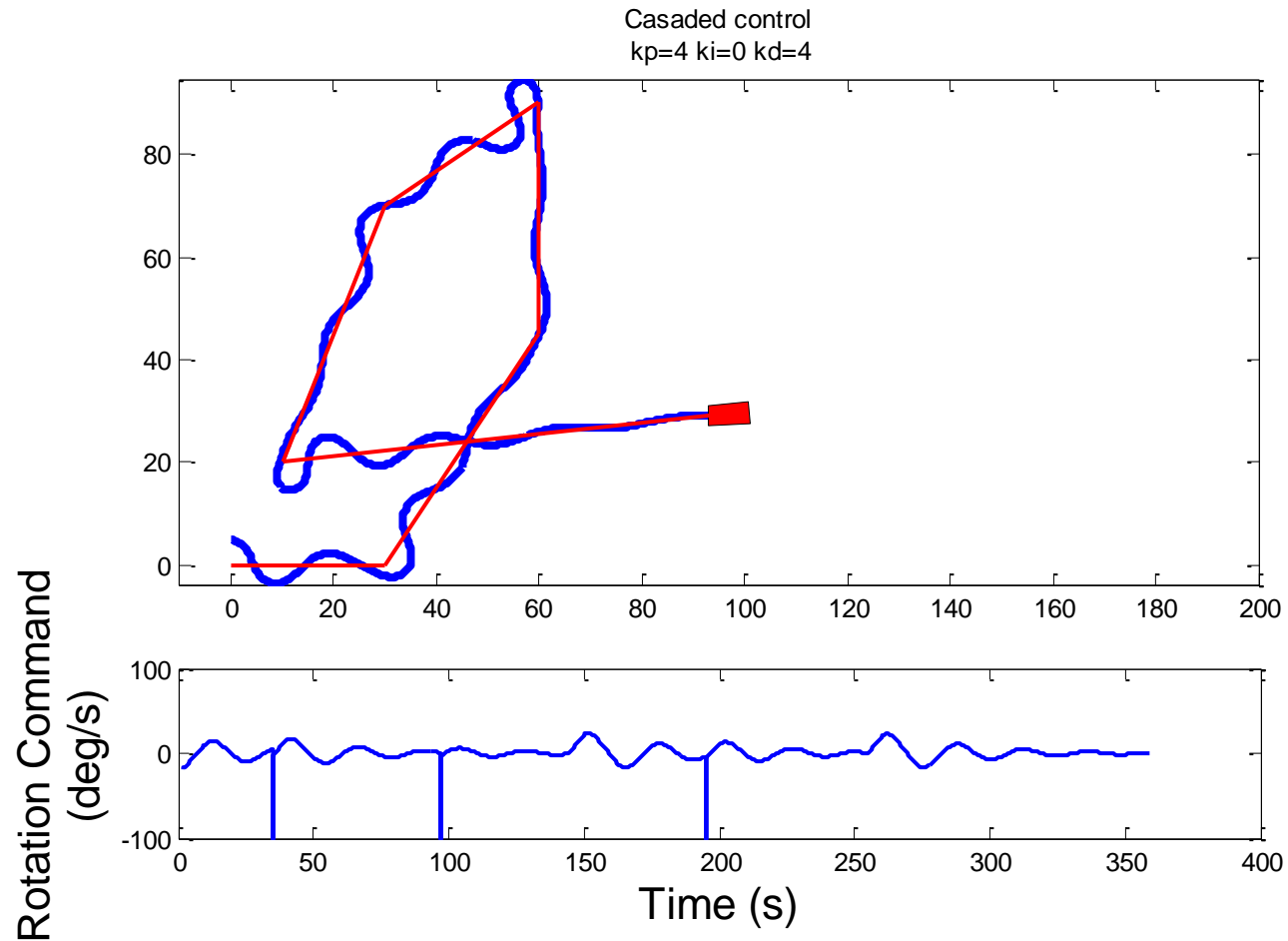
$$- \dot{e} = \frac{d_t - d_{t-1}}{dt}$$

to calculate the turnrate  $\omega$ :

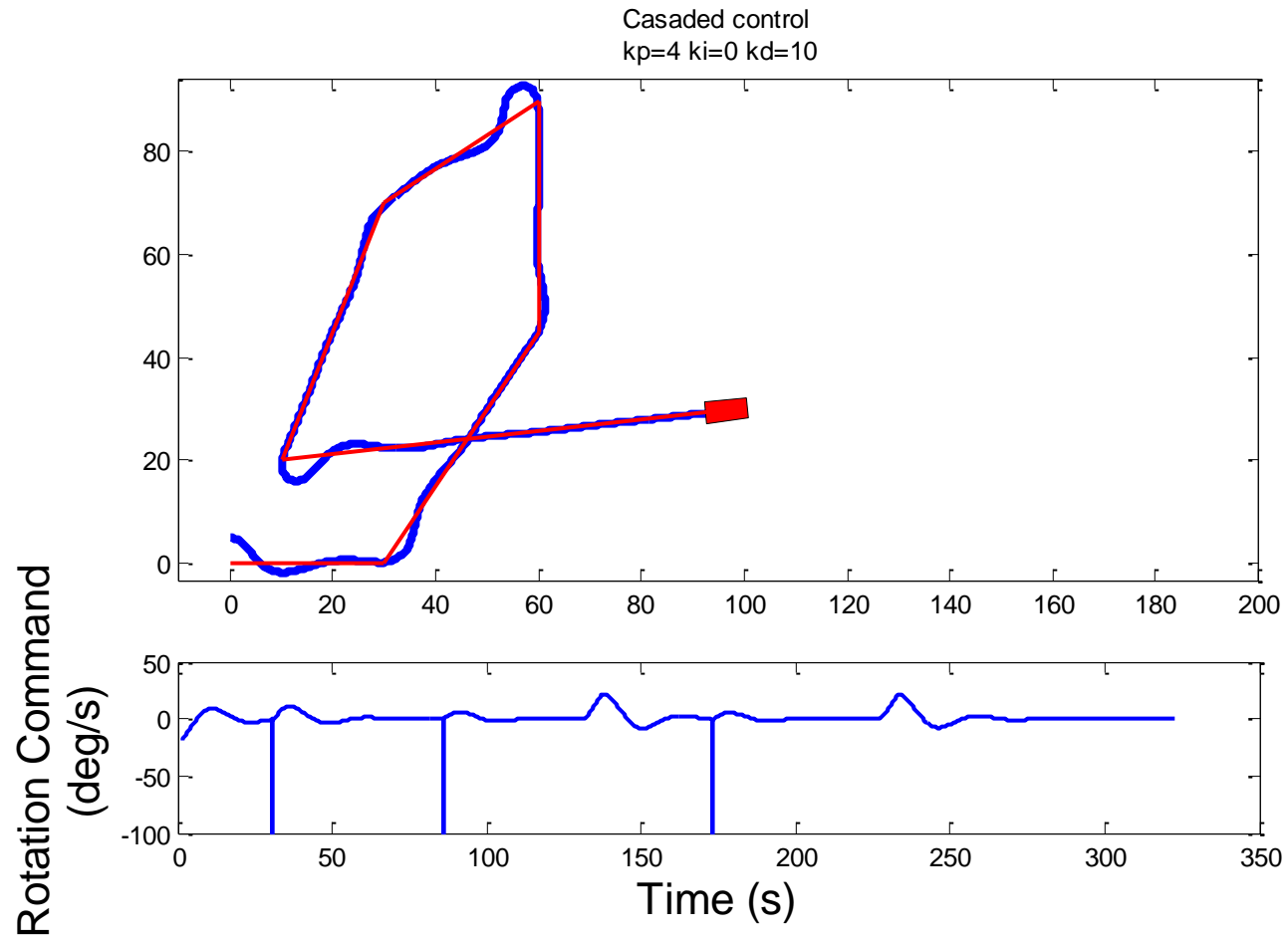
$$\omega = k_p \cdot d_t + k_d \cdot \dot{e}$$



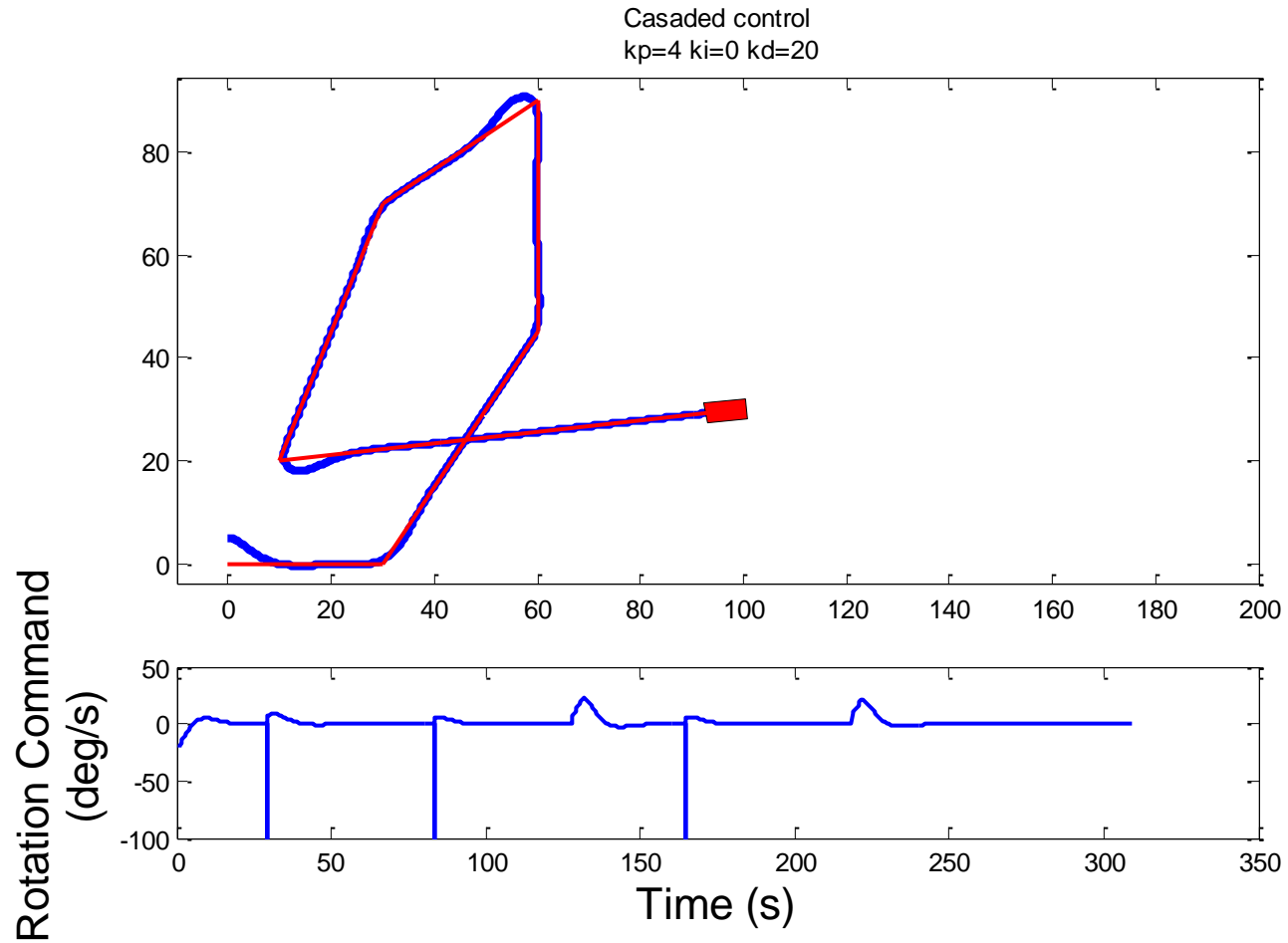
# Pd-Controller



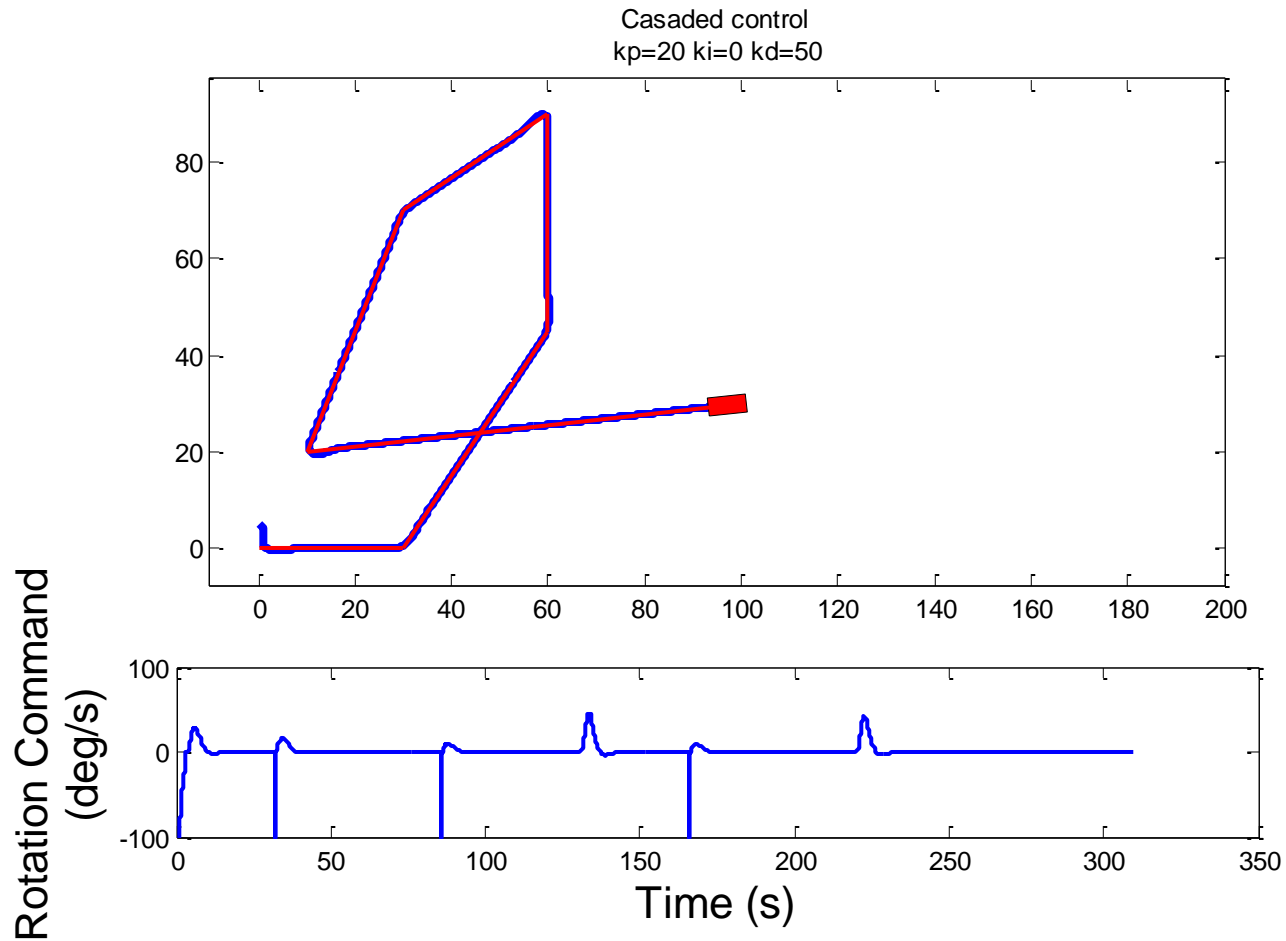
# Pd-Controller



# Pd-Controller



# Pd-Controller



# Questions?

---

