ZJU-UIUC Institute

Zhejiang University / University of Illinois at Urbana-Champaign Institute

ECE 470: Introduction to Robotics

Lecture 16

Liangjing Yang

Assistant Professor, ZJU-UIUC Institute

liangjingyang@intl.zju.edu.cn

Wechat ID: Liangjing_Yang

Last Lecture

- We looked at
  - An overview of robot control
  - Feedback control
  - Joint control and the various components
  - 2nd Order Dynamics System (mechanical mass-spring-damper system)

Recall: Designing Control

- Design control system to achieve specific system behaviors
  - Steady state error
  - Rise time
  - Overshoot
  - Settling time
- Consider Stability

Recall: Designing Robot Control System

- The central question in designing a control system is the performance specification
  - System response
  - Stability analysis

Control of 2nd Order System

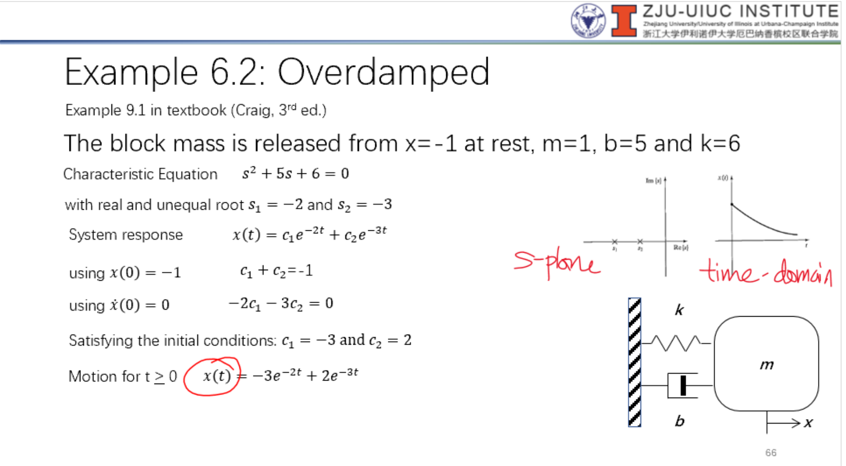
ECE 470: Introduction to Robotics

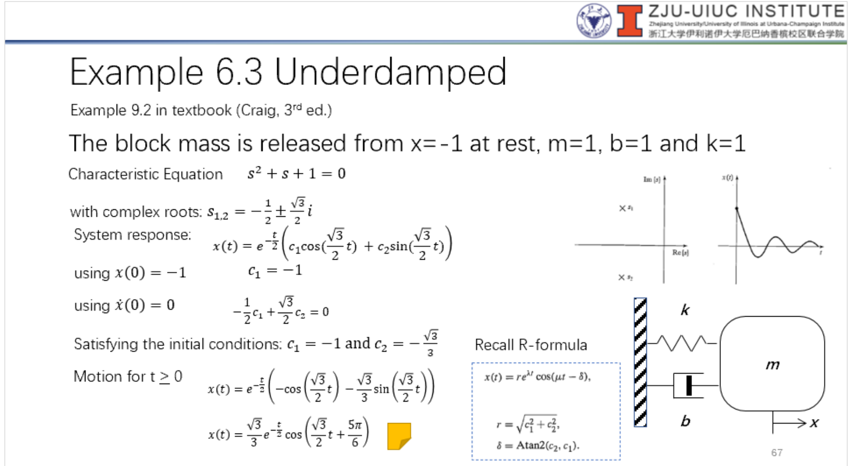
Second-Order System

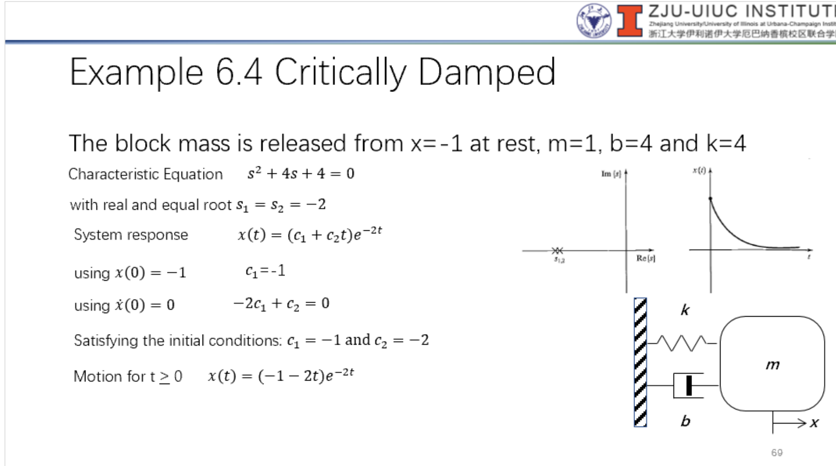
From Ex 6.2-4, the systems are all **stable** where *m, b and k* >0

In free oscillation, the system has natural response

In control applications, the system has forced response







Recall: Second-Order System

The Dynamics of the Mass-Spring-Damper System

$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = 0$$

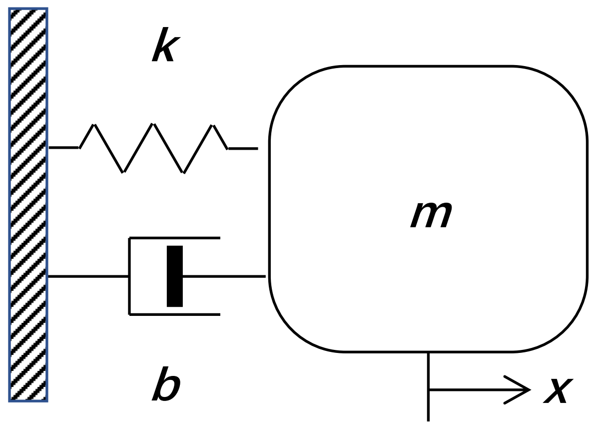
can be described as an oscillatory 2nd order system

$$\ddot{x} + 2\zeta\omega_n\dot{x} + {\omega_n}^2 x = 0$$

characterized by

Natural frequency: $\omega_n = \sqrt{\frac{K}{m}}$

Damping Ratio: $\zeta = \frac{1}{2}\sqrt{\frac{b^2}{mk}}$



Recall: Second-Order System

Oscillatory 2nd order system with free response

Homogenous ODE $\ddot{x} + 2\zeta\omega_n\dot{x} + {\omega_n}^2 x = 0$

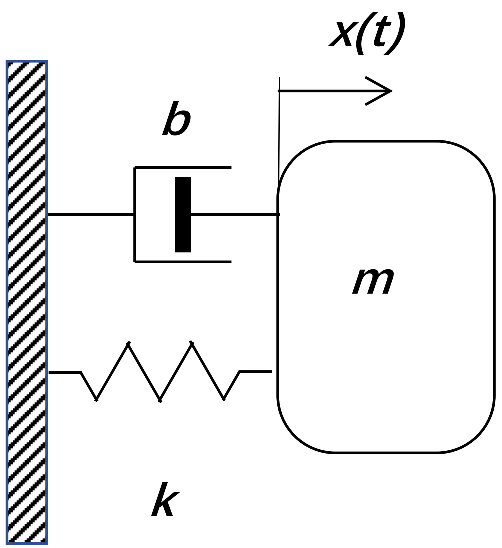
Free response

Overdamped $\zeta > 1$ Real and unequal roots: $s_1 = s_2$

Critically damped $\zeta = 1$ Real and Equal roots $s_1 \neq s_2$

Underdamped $0< \zeta <1$ Complex roots: $s_{1,2} = \lambda \pm \mu i$

No damping $\zeta = 0$ Imaginary roots: $s_{1,2} = \pm \mu i$



Forced Second-Order System

Newton's 2nd Law $$\sum F = m\ddot{x}$$

Net total Force= Drag force+ Spring force+ **External Force**

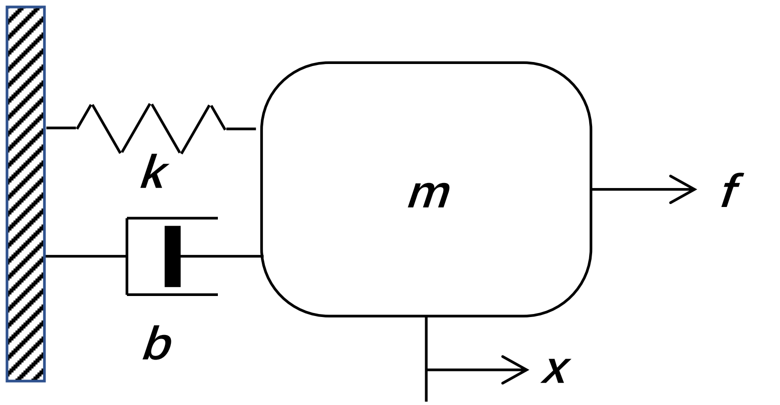
$F_{drag} = -b\dot{x}$

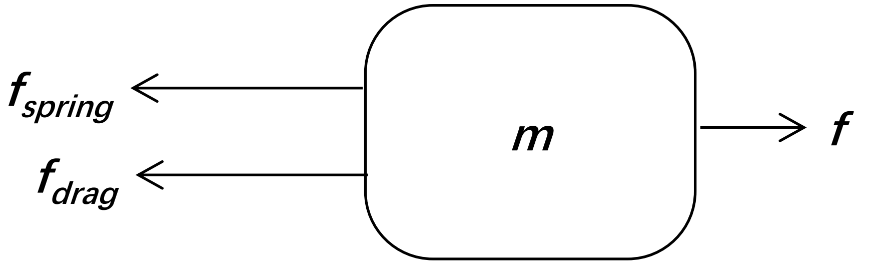
Stokes's law

$F_{Spring} = -kx$

Hooke's law

$$\boxed{m\ddot{x} = -b\dot{x} - kx + f(t)}$$





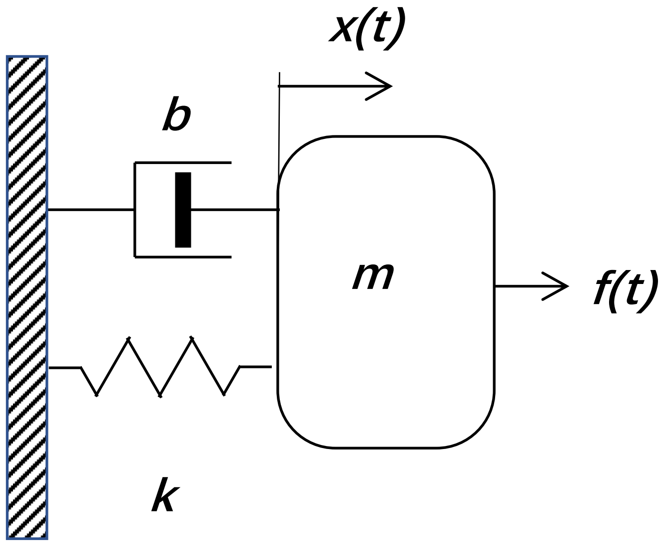
Free Body Diagram

Forced Second-Order System

Forced Oscillatory 2nd order system

Non-homogenous ODE $m\ddot{x}(t) + b\dot{x}(t) + kx(t) = \boxed{}$

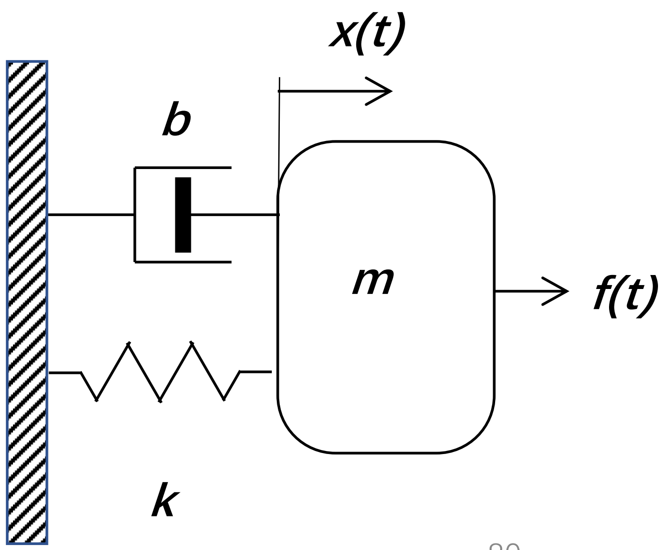
Forced response



Forced: Block Diagram

Block Diagram of the forced system

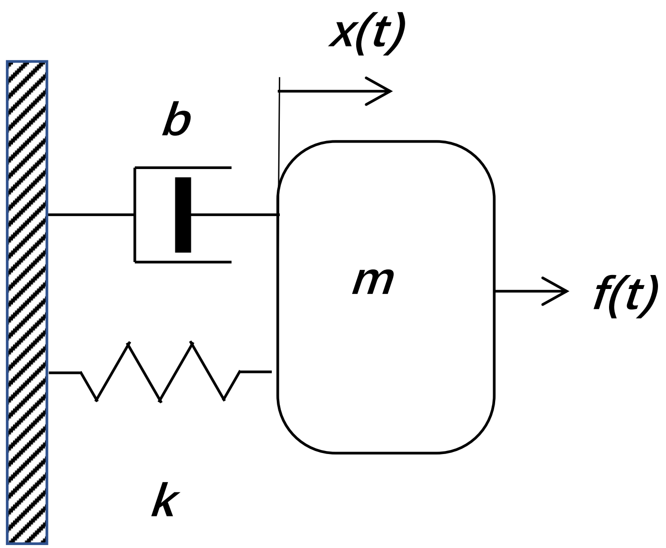
$$m\ddot{x}(t) + b\dot{x}(t) + kx(t) = f(t)$$



Block diagram in s-domain

Second order System can be represented as a double zero-initial-value integrator with negative feedback

$$ms^2X(s) + bsX(s) + kX(s) = F(s)$$



Control of a Second-Order System

Recap Open-loop example:

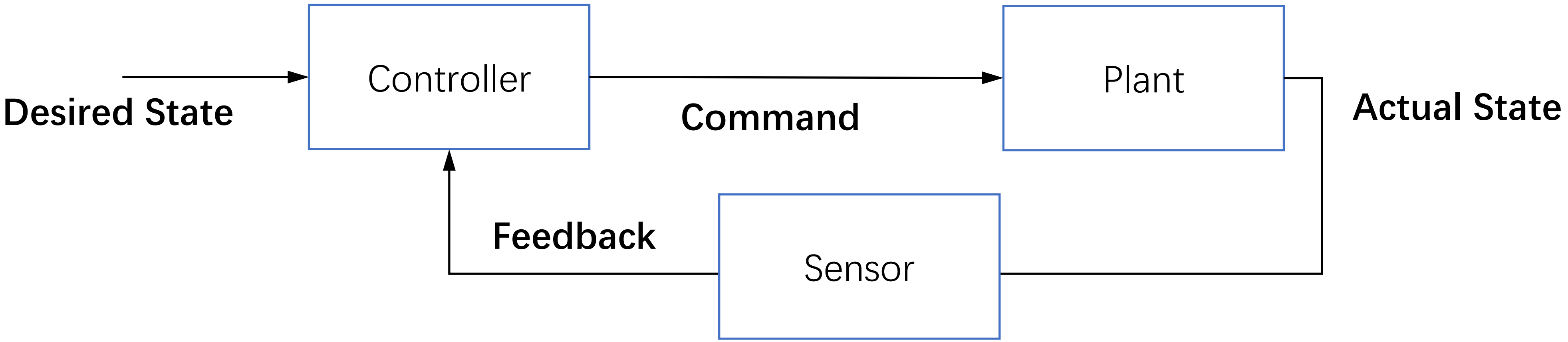
Control the robot to achieve a desired state

Through preestablished model, generate the command that will achieve the desired state. This is known as an **open-loop system**.

Control of a Second-Order System

Recap Closed-loop example:

Control the robot to maintain a desired state



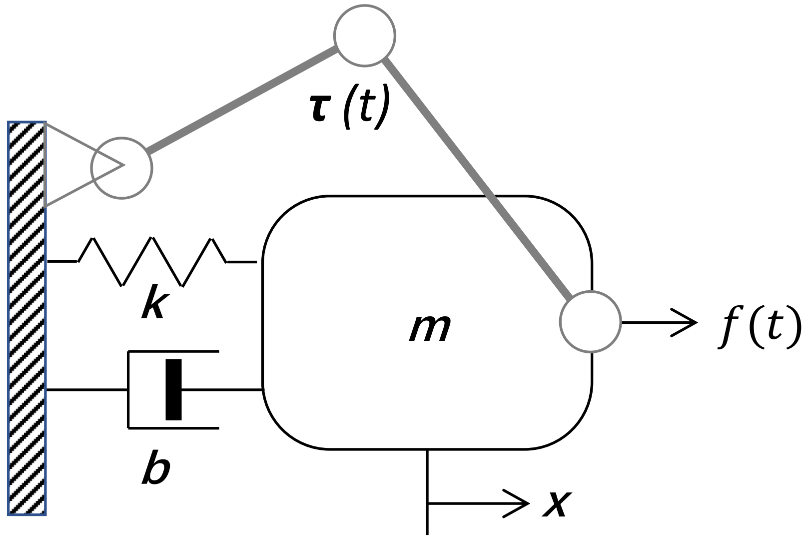
Through sensors, we are able to **feedback** the measurement to produce the command that will minimize the error between desired and actual targeted profile. This is known as a **closed-loop system**.

Control of a Second-Order System

In order to modify the system's behavior as desired, we can use closed-loop control, by using sensors, actuators and control system.

For example, if the position and velocity of the block can be measured, it is possible to apply a force f to the block as shown.

$$m\ddot{x} + b\dot{x} + kx = f(t)$$



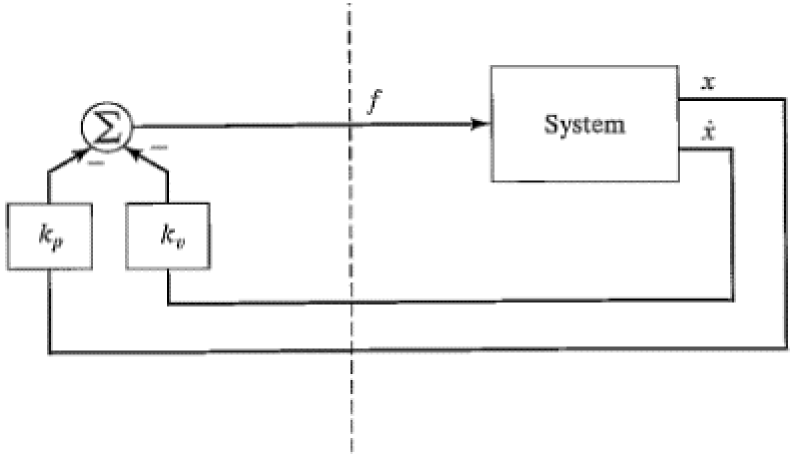
Control of a Second-Order System

Assume that we are able to sense $\dot{x}$ and x and feedback to the controller. We can propose a control law that determine the force that the actuator should apply as a function of the feedback as shown

$$f(t) = -k_p x - k_v \dot{x}$$

$$m\ddot{x} + b\dot{x} + kx = -k_p x - k_v \dot{x},$$

Control Gains



Closed-loop Stiffness

Example 6.5: Closed loop gain

Recall example 6.3: The block mass is released from x=-1 at rest, m=1, b=1 and k=1

Determine the gains K_p and K_v such that the system is being critically damped with a closed loop stiffness of 16.0

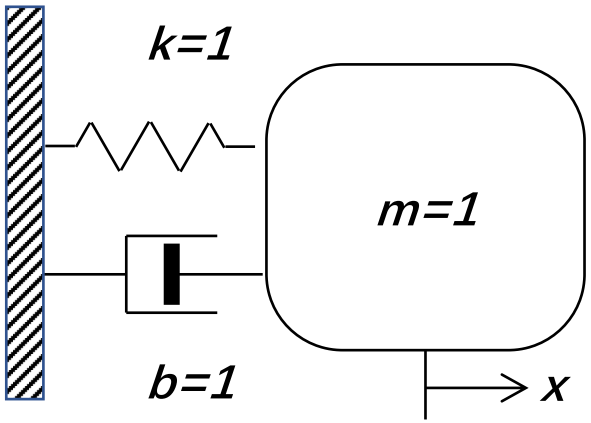
Control Law:

Closed Loop System:

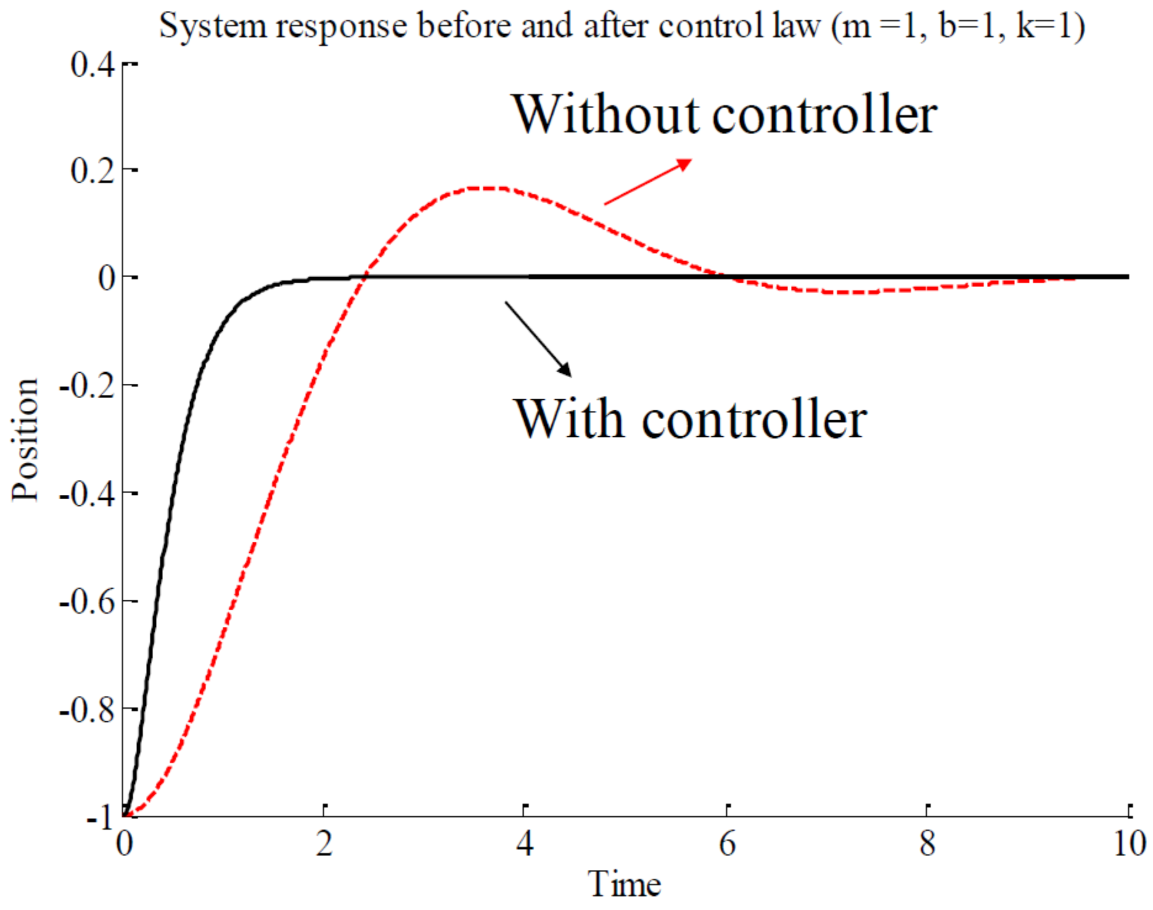
CL Stiffness, Damping:

Since $k' =$ _____ $= 16$, $k_p =$ _____
For Critically damped response, $b' =$ ______
Hence, $k_v =$ _________

Control Gains: $[k_p, k_v] = [\_\_, \_\_]$



Response


System response before and after control law (m =1, b=1, k=1)
Without controller
With controller
Position
Time
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
0
2
4
6
8
10

Control-Law Partitioning

The control system consist of a Model-based portion and a Servo portion

Open loop equation: $m\ddot{x} + b\dot{x} + kx = f$

Model-based portion is a control law in the form

$$f = \alpha f' + \beta$$

Hence, the system equation is written as

$$m\ddot{x} + b\dot{x} + kx = \alpha f' + \beta$$

To make the system unit mass,

$$\alpha = m, \qquad \beta = b\dot{x} + kx$$

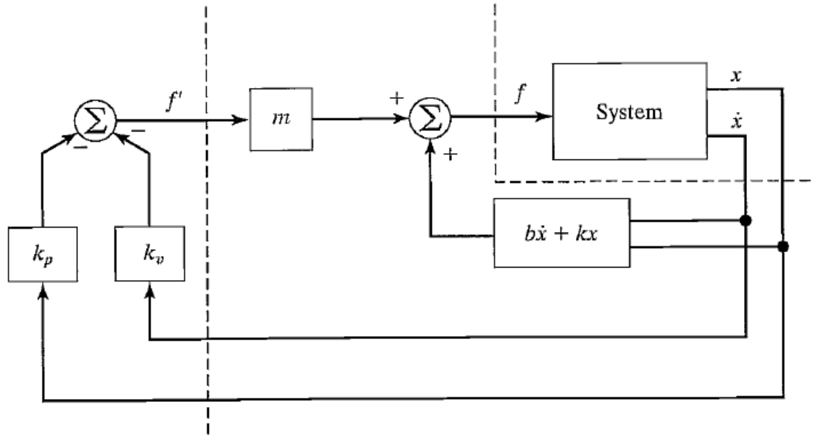
Therefore, $\ddot{x} = f'$

Proceed with servo portion of control law

$$f' = -kpx - kv\,\dot{x}$$

Since the model-based portion make $\ddot{x} = f'$,

$$\ddot{x} + k_p x + k_v\,\dot{x} = 0$$



Example 6.6: Control-Law Partitioning

Recall example 6.3: The block mass is released from x=-1 at rest, m=1, b=1 and k=1
Using control law partitioning, determine the α, β and gains k_p and k_v such that the system is being critically damped with a closed loop stiffness of 16.0

$$\alpha = m = 1$$
$$\beta = \dot{x} + x$$

Effective system equation

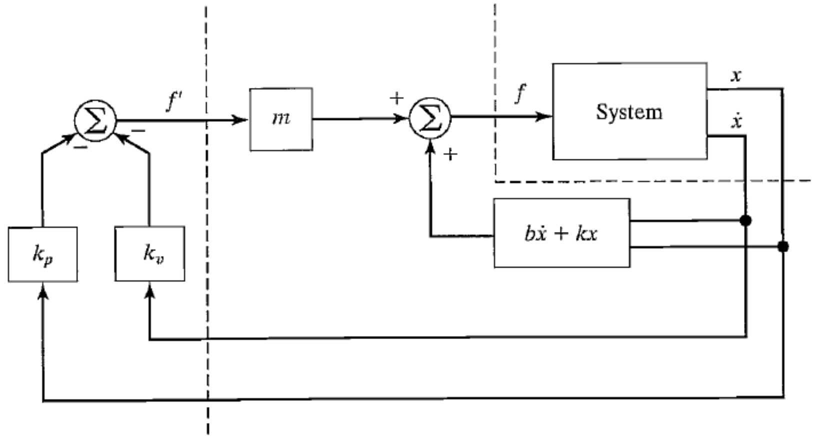
$$\ddot{x} + kv\,\dot{x} + kpx = 0$$

Hence

$$k_p = \_\_$$

For Critically damped response,

$$k_v = \_\_\_\_\_\_$$



Trajectory-Following

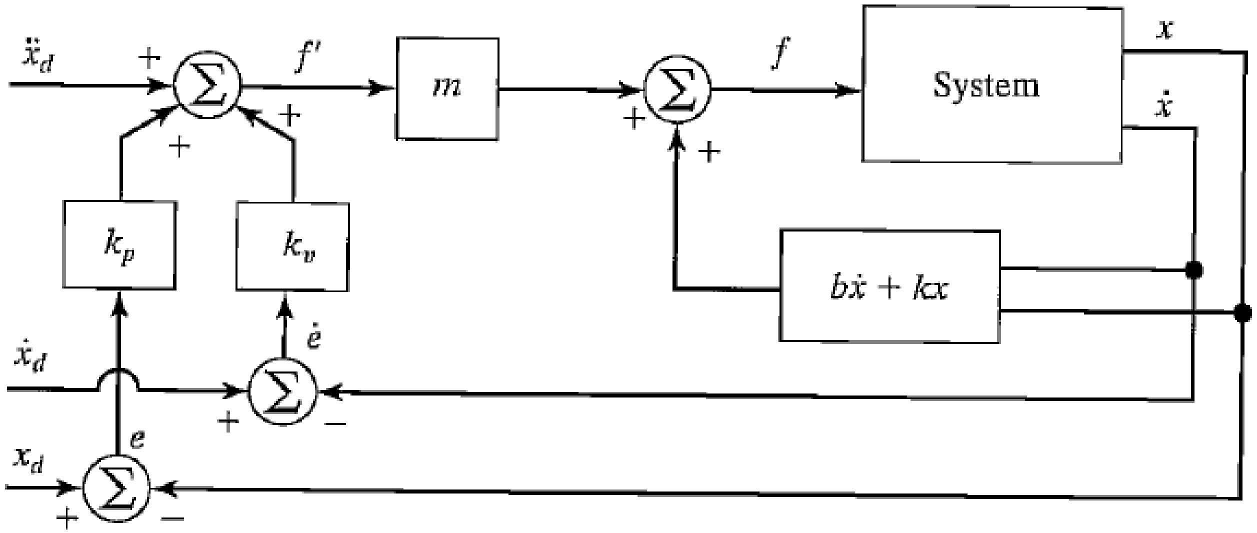
- In Example 6.5, we establish a control law that restore the mass to equilibrium (with forced response) according to our specified response behavior
- This is a form of position-regulation where we maintain the block at the desired position (equilibrium; x=0)
- More generally, we can specify desired motion trajectories to be followed

Trajectory-Following

Trajectory realized by a controller that minimizes **servo error**, $e = xd - x$ with a **servo-control law**, $f' = \ddot{x}_d + kv\,\dot{e} + kpe$

i.e. $\ddot{x} = \ddot{x}_d + kv\,\dot{e} + kpe$

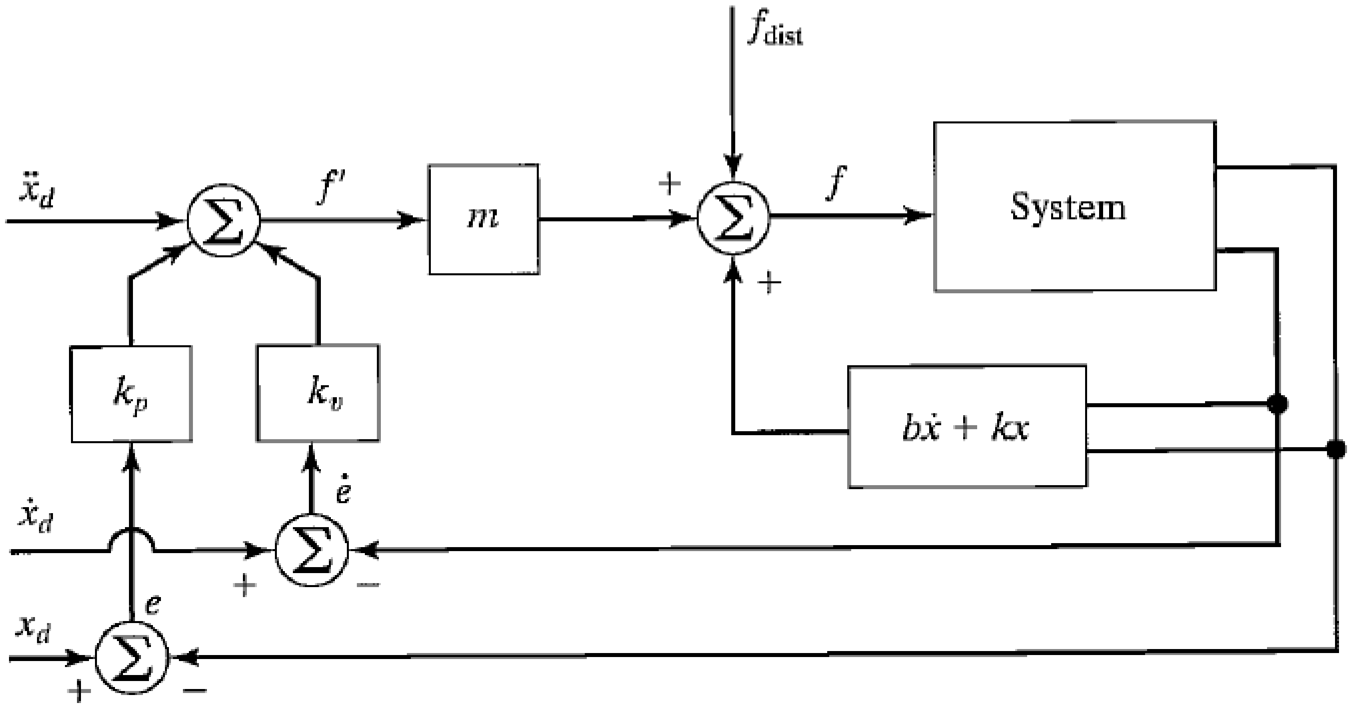
or $\ddot{e} + kv\,\dot{e} + kpe = 0$



Disturbance Rejection

There could be disturbance f_{dist} from the environment making the system equation to be

$$\ddot{e} + kv\,\dot{e} + kpe = f_{\text{dist}}$$

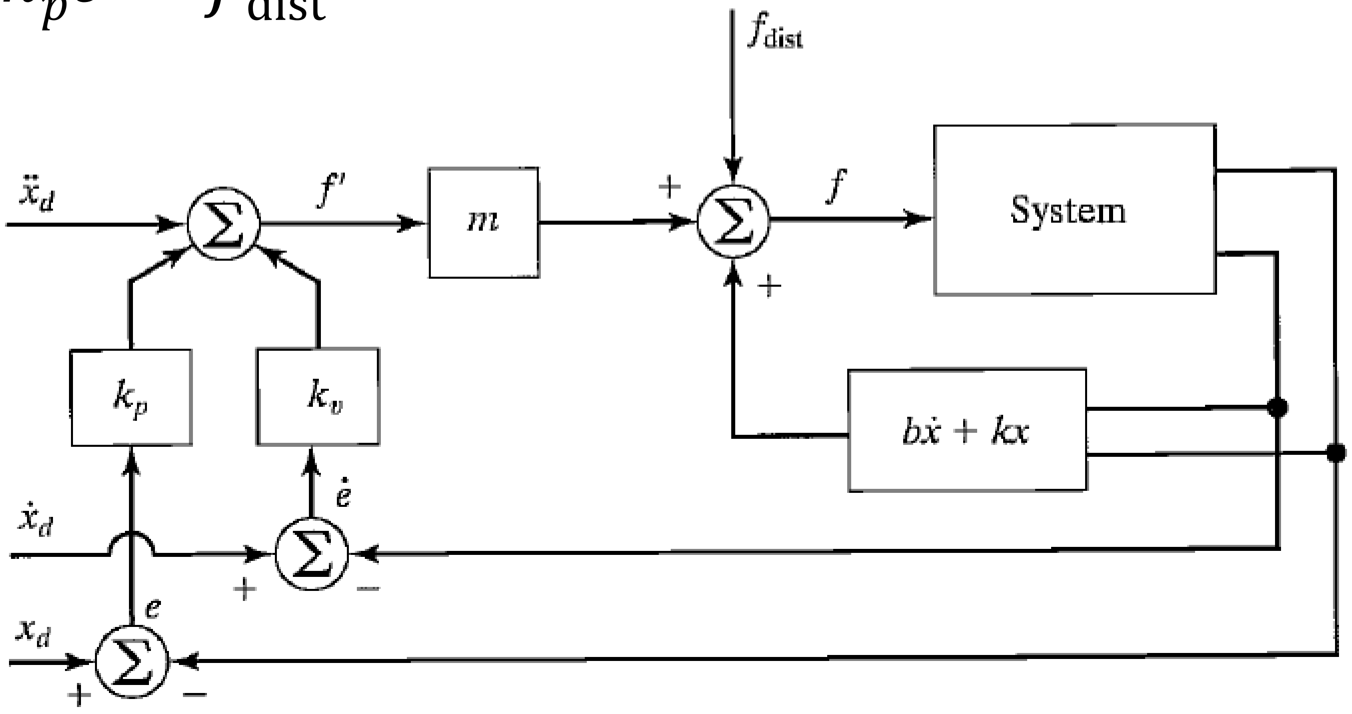


Disturbance Rejection

In case where f_{dist} is a constant, performing steady state analysis at rest, setting derivatives to be zero

$$\ddot{e} + kv\,\dot{e} + kpe = f_{\text{dist}}$$

$$k_p e = f_{\text{dist}}$$



Disturbance Rejection

In case where f_{dist} is a constant, performing steady state analysis at rest, setting derivatives to be zero

$$k_p e_{ss} = f_{\mathrm{dist}} \Rightarrow e_{ss} = f_{\mathrm{dist}}/k_p$$

Setting large k_p may reduce but cannot eliminate e_{ss}

Proportional-Integral-Derivative Control

To rectify the steady state error, an integral term can be added

Control law: $f' = \ddot{x}_d + kv\,\dot{e} + kpe + k_i \int edt$

Error equation: $\ddot{e} + kv\,\dot{e} + kpe + k_i \int e\,dt = f_{dist}$

Take time derivative, $\dddot{e} + kv\ddot{e} + kp\dot{e} + k_i e = \dot{f}_{dist}$

For constant disturbance, $\dot{f}_{dist} = 0$

$k_i e = 0$ so $e = 0$

Independent Joint Control

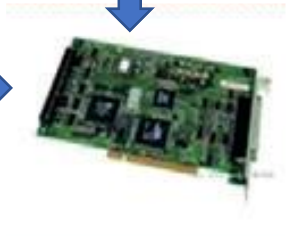
ECE 470 Introduction to Robotics

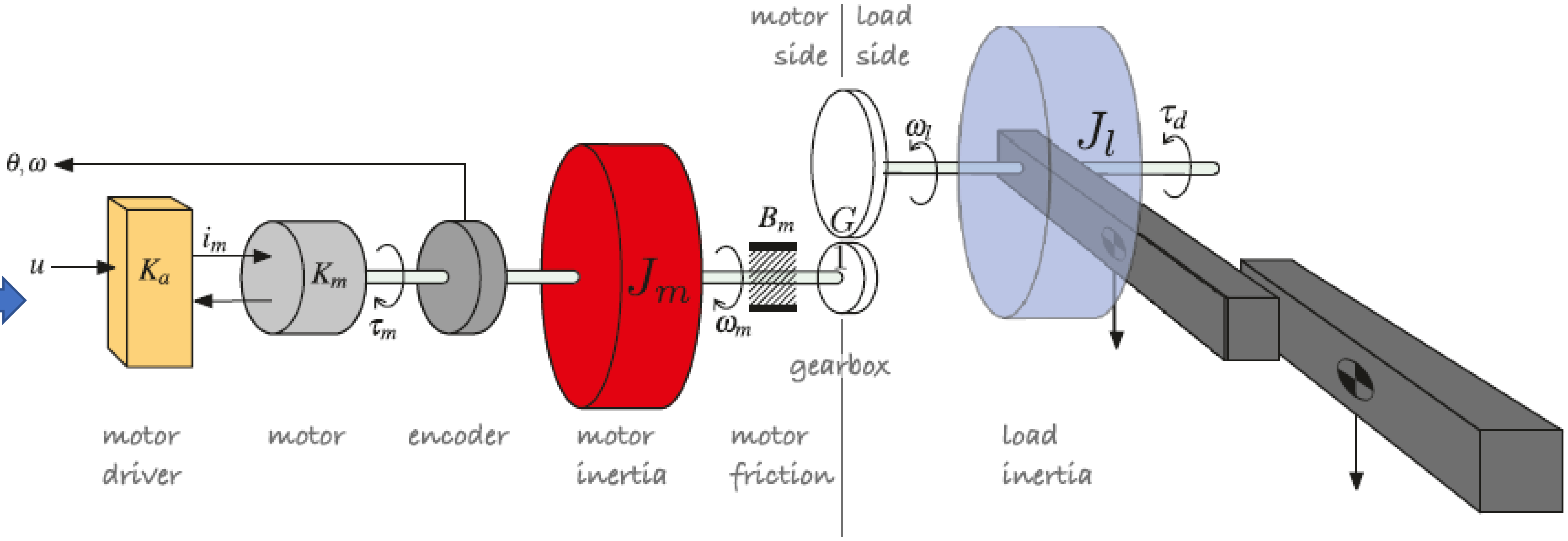
Independent Joint Control

- The 2nd order system dealt with so far has single DOF
- Ultimately, we are interested in multibody robotic systems that involve Multi-Input, Multi-Output (MIMO) control systems
- We shall first adopt an **independent joint control** approach with <u>N independent Single-Input Single-Output (SISO)</u> control systems

Joint Control in Robotic System

Trajectory Planning



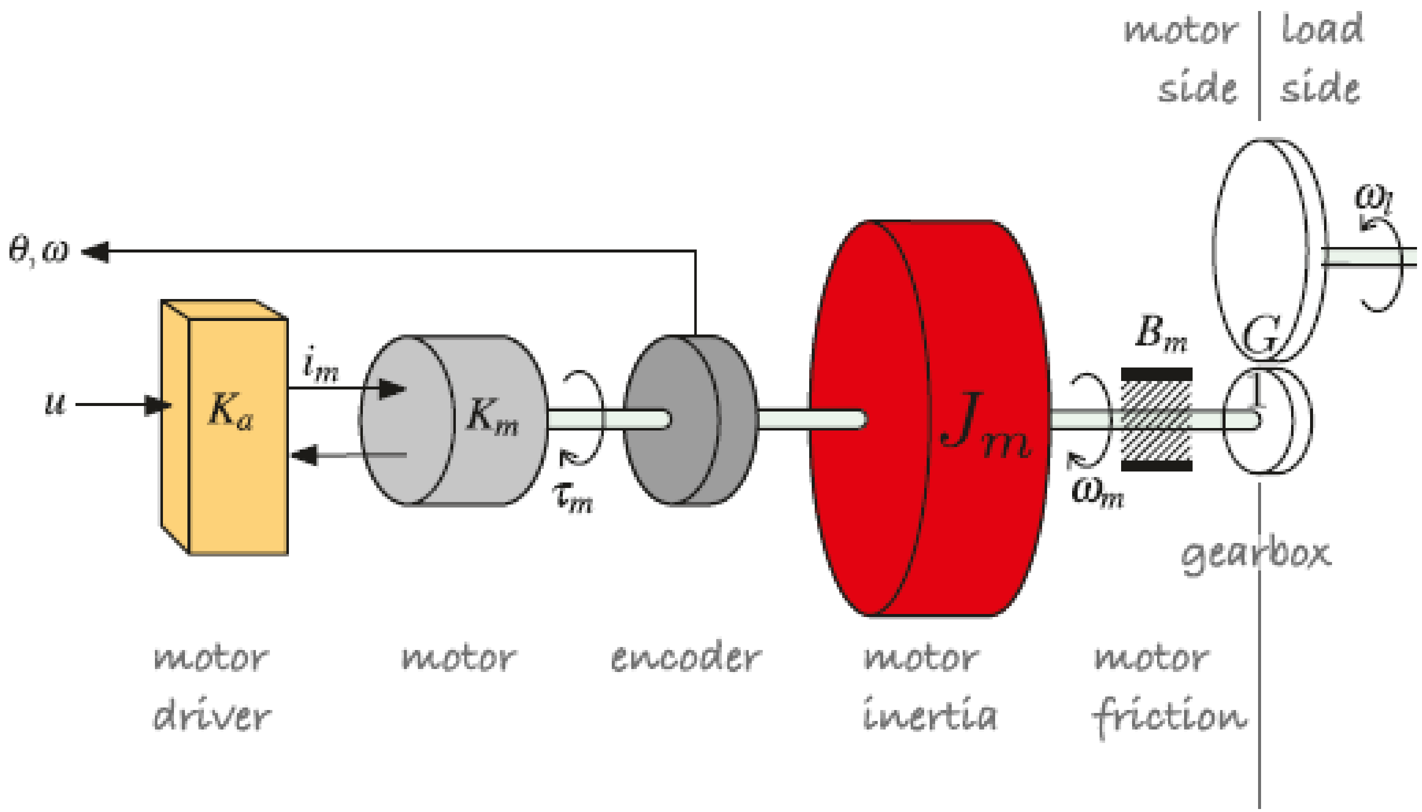


Human-Machine Interface

Controller

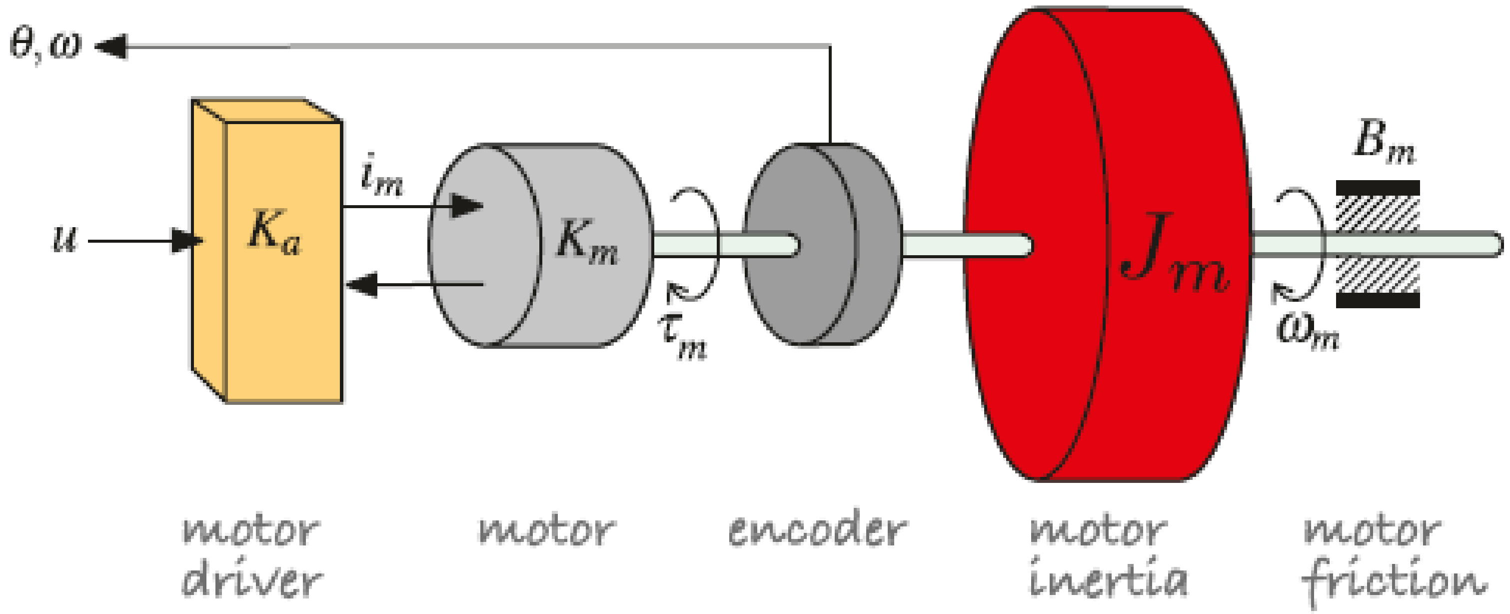
Modeling Single Joint Control

Integrated assembly



Modeling Single Joint Control

A demand voltage u controls the current i_m flowing into the **motor (Actuator)** which generates a torque τ_m that accelerates the rotational inertia J_m and is opposed by friction $B_m \omega_m$. The **encoder (Sensor)** measures rotational speed and angle



Modeling Single Joint Control

- How to model the resultant system as a 2nd order linear system?
    - Combining the Dynamics of the electromechanical system

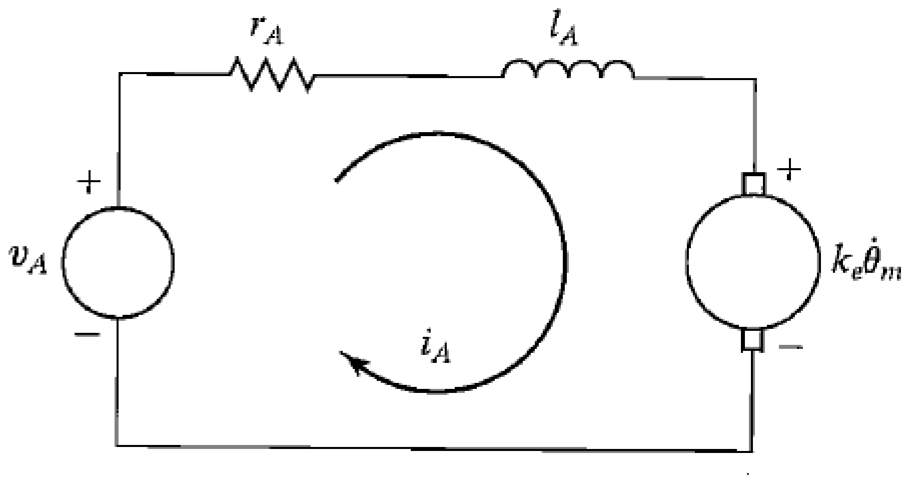
Modeling Single Joint Control

- How model the resultant system as a 2nd order linear system?

Physics Law:
Lenz Force, $F = qV \times B$

Motor Torque, $\tau_m = k_m i_a$

Back emf, $v = k_e \dot{\theta}_m$



Modeling Single Joint Control

- Model the Motor

Current

Physics Law:
Lenz Force, $F = \boxed{qV} \times \boxed{B}$

Magnetic Field

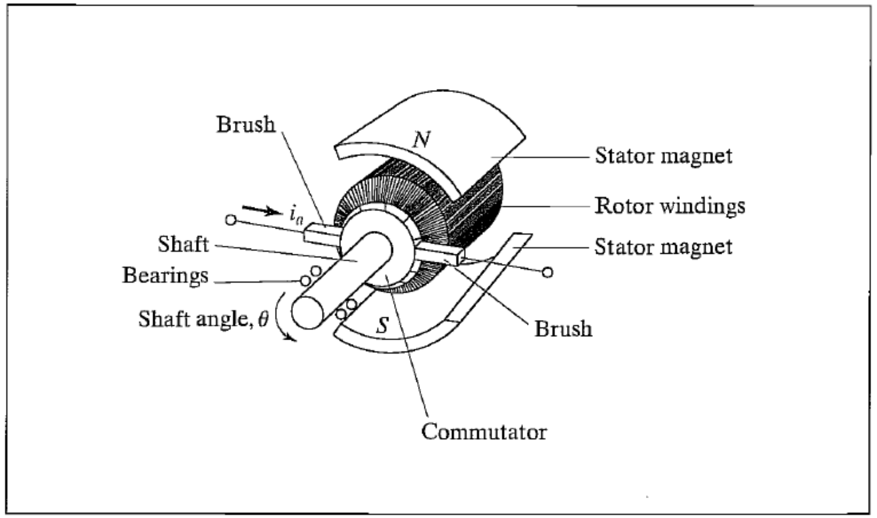
Motor Torque, $\tau_m = k_m i_A$
Back emf, $v = k_e \dot{\theta}_m$

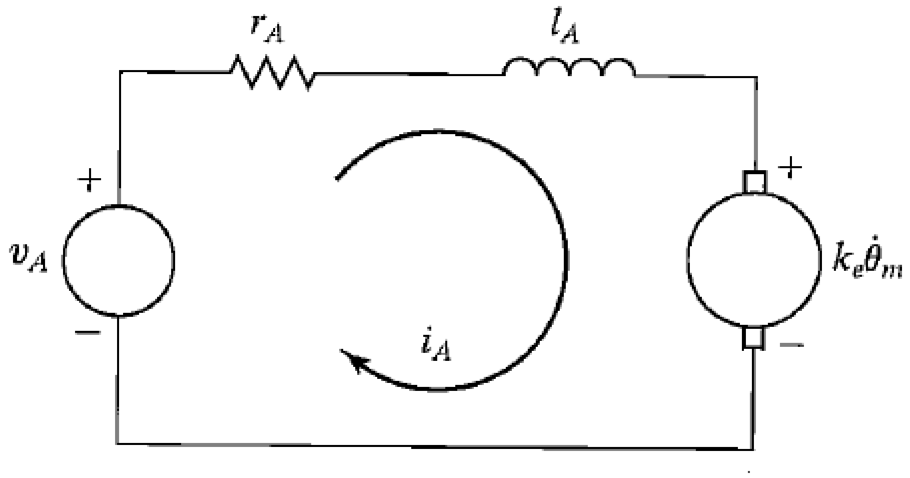
Torque Constant

Torque Constant

Kirchoff Law:

$$v_A = l_A \dot{i}_A + r_A i_A + k_e \dot{\theta}_m$$
$$v_A - k_e \dot{\theta}_m = l_A \dot{i}_A + r_A i_A$$





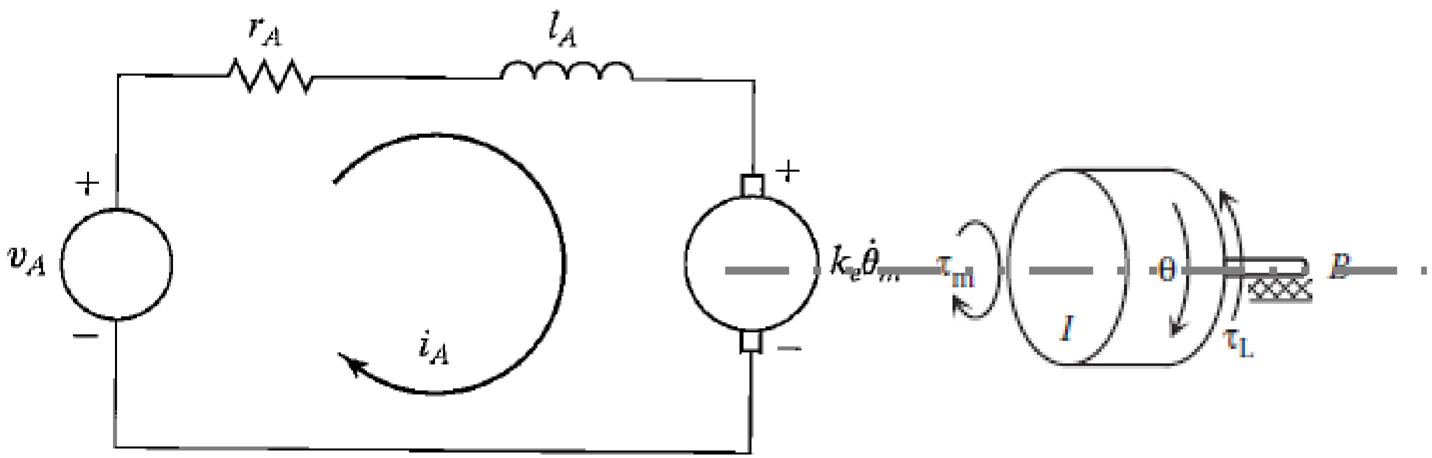
Modeling Single Joint Control

- Model the Motor

Motor Torque, $\tau_m = k_m i_A$
Back emf, $v = k_e \dot{\theta}_m$

Kirchoff Law:

$$v_A = l_A \dot{i}_A + r_A i_A + k_e \dot{\theta}_m$$
$$v_A - k_e \dot{\theta}_m = l_A \dot{i}_A + r_A i_A$$



$$\tau_m - \tau_L - b\dot{\theta} = I\ddot{\theta}$$

$$k_m i_A - b\dot{\theta}_m - I\ddot{\theta}_m = \tau_L$$

Modeling Single Joint Control

- Model the Motor-Gearing-Load

$$\tau_m = I_m\ddot{\theta}_m + bm\dot{\theta}_m + \left({}^{1}/_{\eta}\right)(I\ddot{\theta} + b\dot{\theta})$$

Gear Ratio, η:

$$\tau = \eta\tau_m,$$

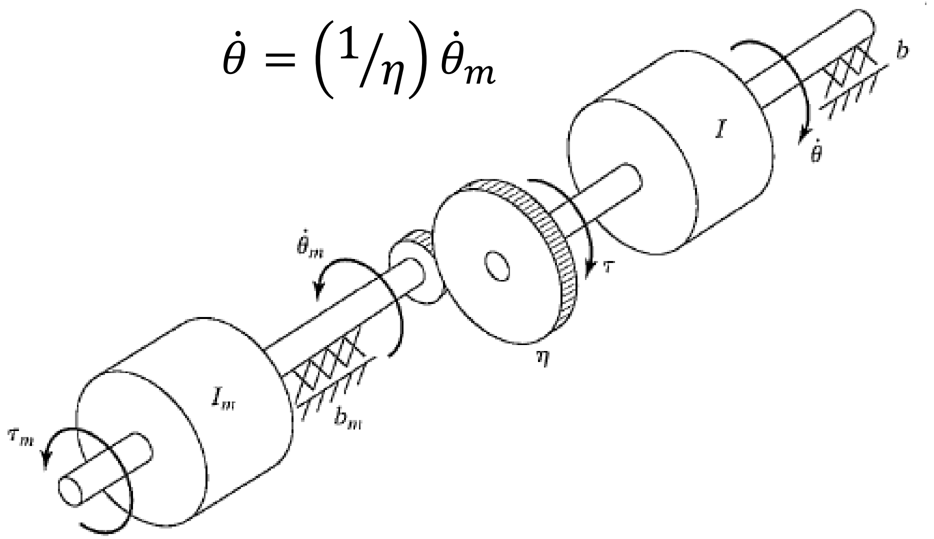
$$\dot{\theta} = \left({}^{1}/_{\eta}\right)\dot{\theta}_m$$

$$\tau_m = \left((Im + {}^{I}/_{\eta^2})\right)\ddot{\theta}_m + \left(b_m + \left({}^{b}/_{\eta^2}\right)\right)\dot{\theta}_m$$

$$\tau = \boxed{\left((\eta^2 I_m + I)\right)}\ddot{\theta} + \boxed{\left(\eta^2 b_m + b\right)}\ \dot{\theta}$$

Effective inertia

Effective damping

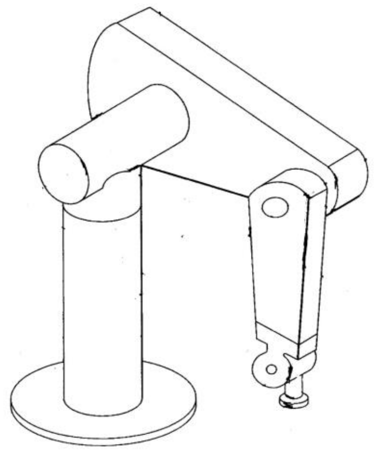


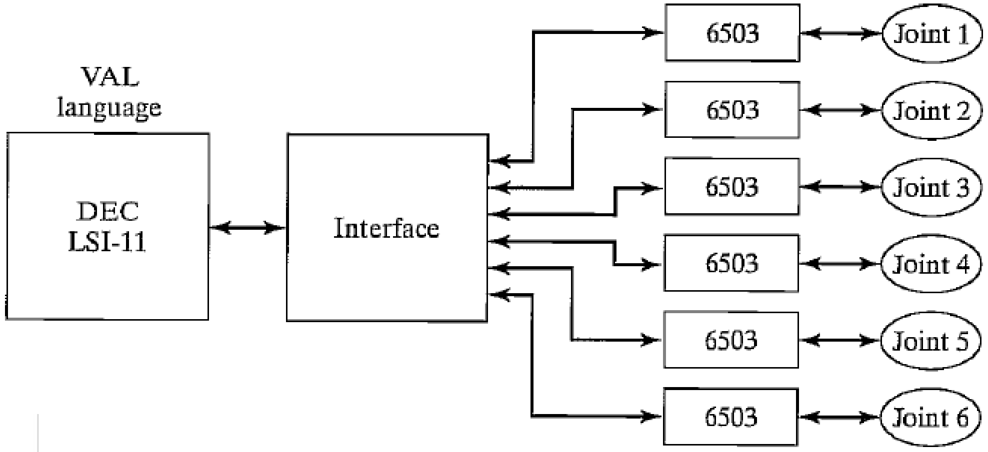
Robot Control Scheme

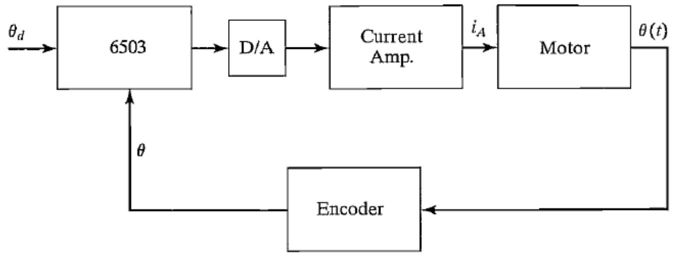
ECE 470 Introduction to Robotics

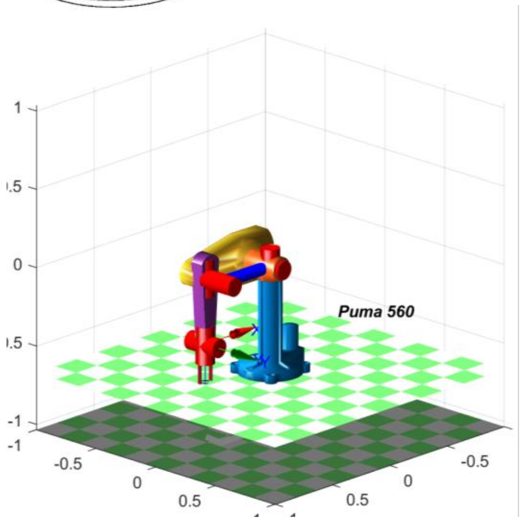
Joint Control for Robot

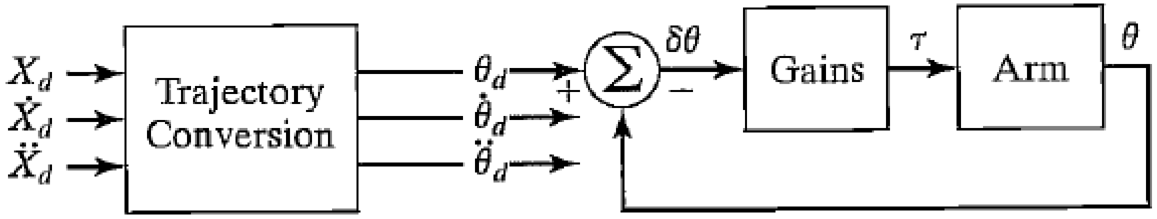
Example of industrial Robotic Arm:
PUMA 560











Joint based vs. Cartesian based

Textbok Chapter 10.8 (Craig 3rd Ed, 2005)

Computation before the control loop

Joint-based-based Control Scheme

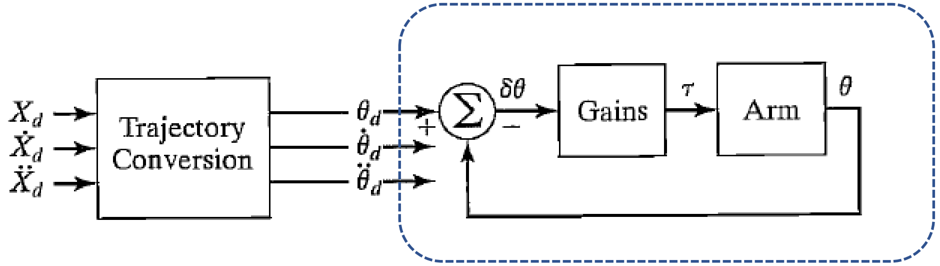


FIGURE 10.10: A joint-based control scheme with Cartesian-path input.

Computation within the control loop

Cartesian-based Control Scheme

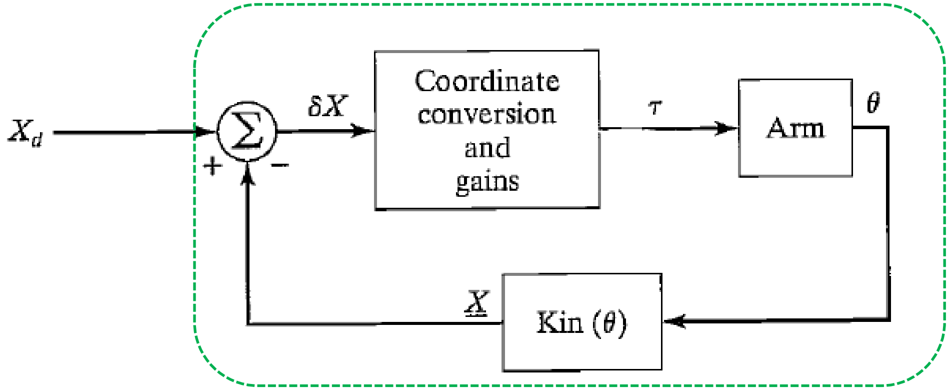


FIGURE 10.11: The concept of a Cartesian-based control scheme.

Joint based vs. Cartesian based

Question: As an engineering designing the control scheme, which ones will you choose for the following robots?

Multi-joint Snake Robot

https://www.gizbot.com/news/indian-scientists-developing-snake-robot-027613.html

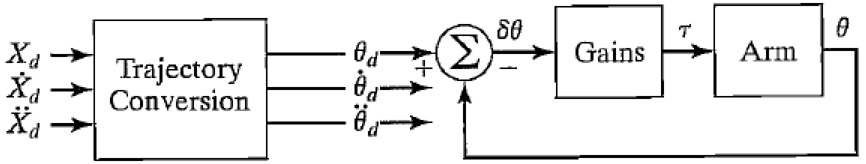


FIGURE 10.10: A joint-based control scheme with Cartesian-path input.

Serial Arm with 2 long links

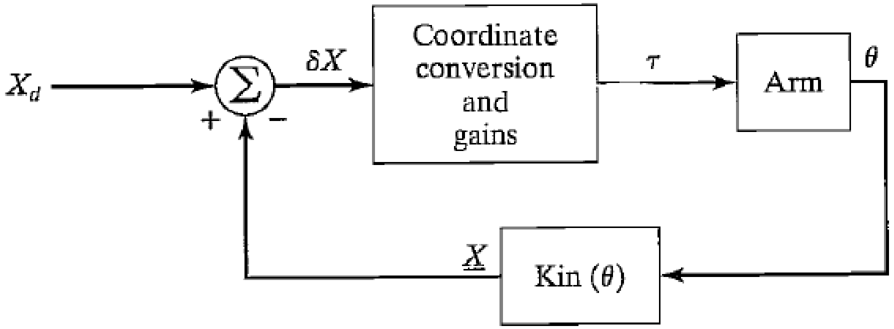


FIGURE 10.11: The concept of a Cartesian-based control scheme.

Cartesian Control Scheme

Textbok Chapter 10.8 (Craig 3rd Ed, 2005)

Both are doing coordinate conversation. Which is doing in force domain?

Coordinate Conversion in Spatial Domain

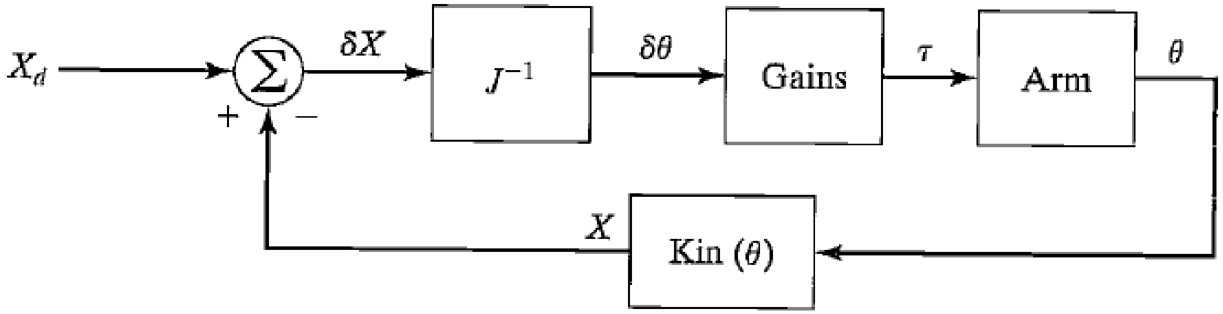


FIGURE 10.12: The inverse-Jacobian Cartesian-control scheme.

Coordinate Conversion in Force Domain

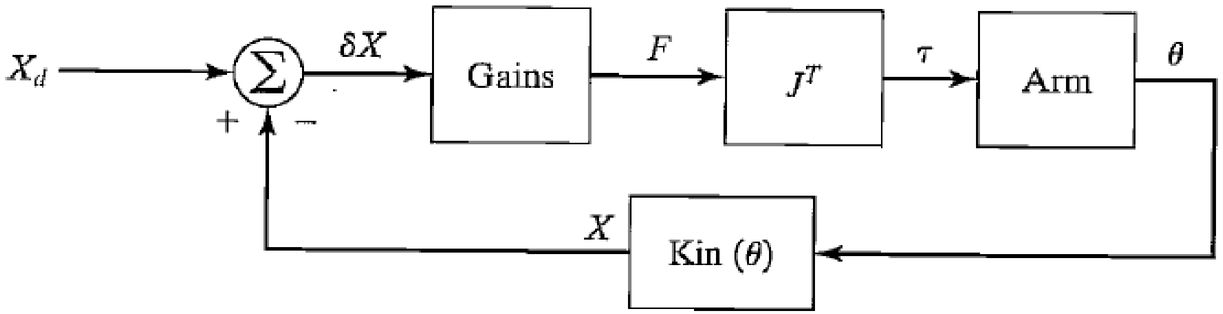


FIGURE 10.13: The transpose-Jacobian Cartesian-control scheme.