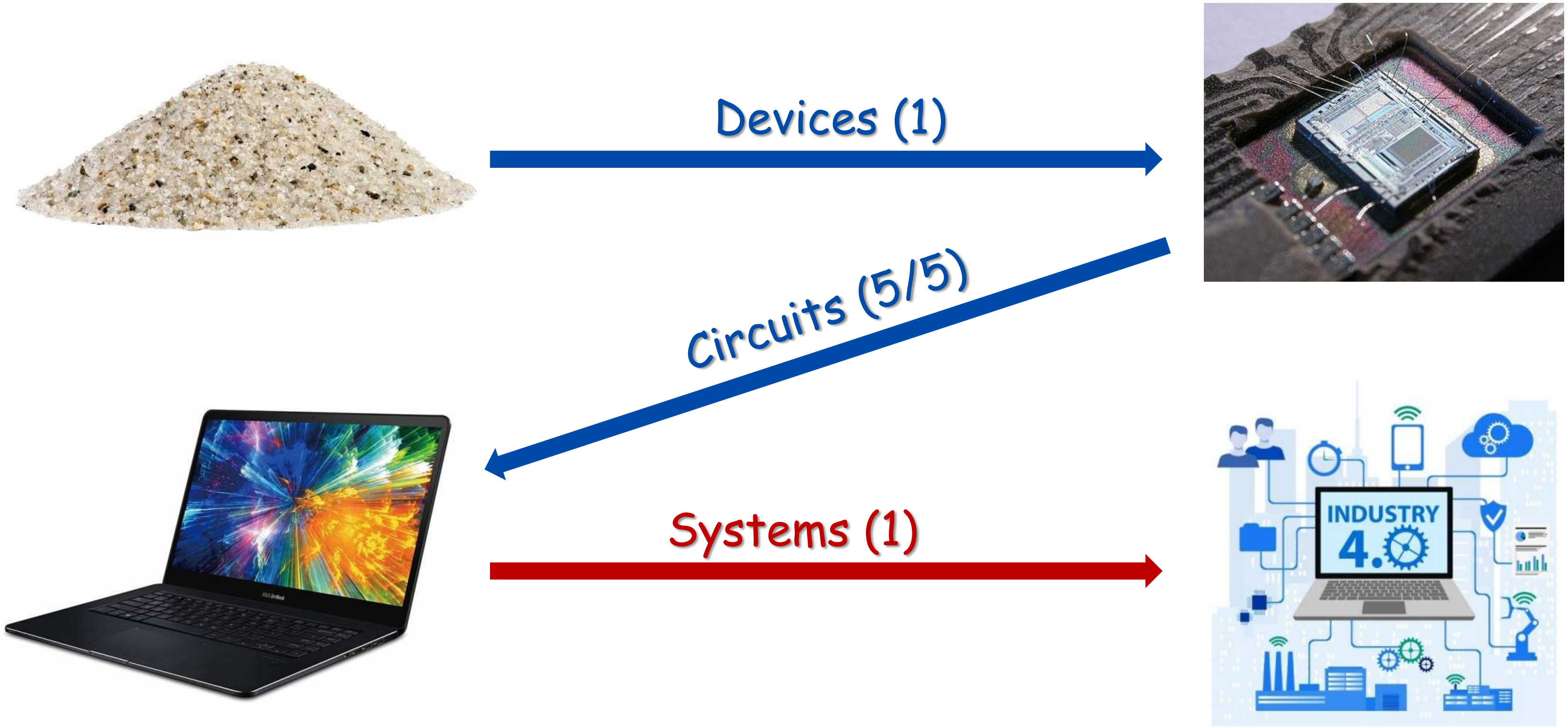
SI100B
Introduction to Information Science and Technology (Part 3: Electrical Engineering)

Lecture #8
Dynamic Systems and Control

Instructor: Haoyu Wang(王浩宇)

Apr 28th, 2023

The Theme Story



(Figures from Internet)

Study Purpose of Lecture #8

- 哲学三问
  - Who are you?
  - Where are you from?
  - Where are you going?

To answer those questions throughout your life

(Figures from Internet)

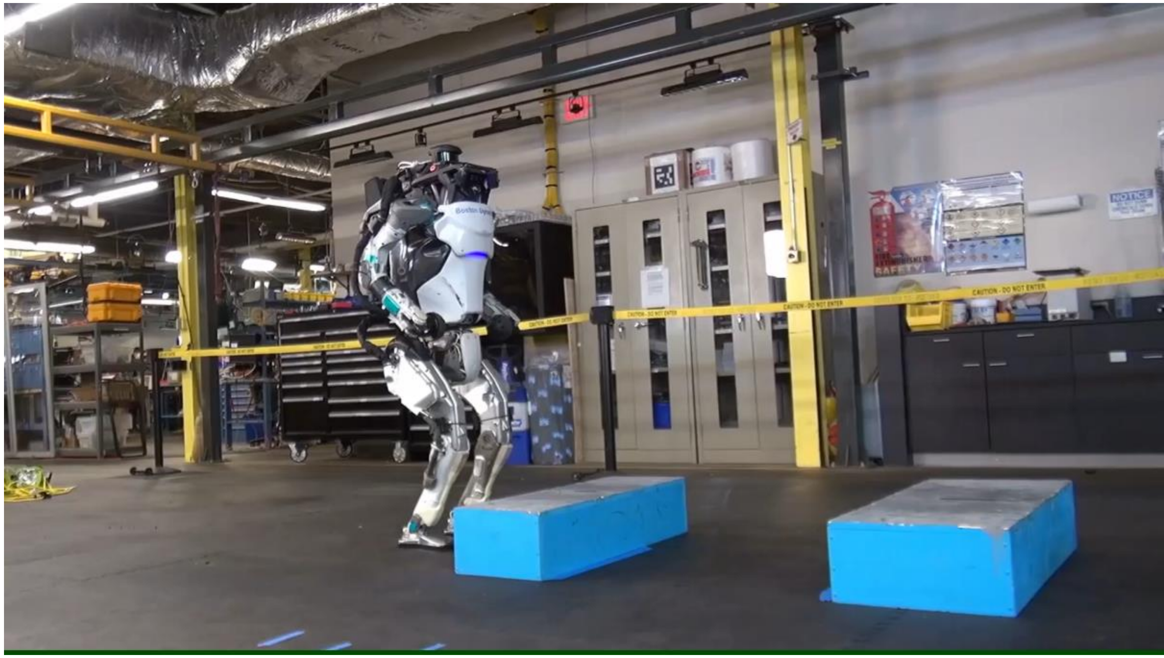
- In this lecture, we ask
  - What is **control** engineering?
  - What is **feedback** control system?
  - How does **PID** controller work?

Lecture Outline

- Control and connectivity towards Industry 4.0
- Mathematical model of a dynamic system
- Feedback control system
  - Block diagram
  - Examples
- Controller design
  - Proportional control
  - Integral control
  - Derivative control

Application of Control Systems

- Autonomous robots
- Autonomous cars
- Quadcopters
- Self-balance robots
- Other more applications



(https://www.youtube.com/watch?v=fRj34o4hN4I)

Application of Control Systems

- Autonomous robots
- Autonomous cars
- Quadcopters
- Self-balance robot
- Other more applications

(https://www.youtube.com/watch?v=tlThdr3O5Qo)

Application of Control Systems

- Autonomous robots
- Autonomous cars
- Quadcopters
- Other more applications

(https://www.youtube.com/watch?v=w2itwFJCgFQ)

Definition of Control Systems

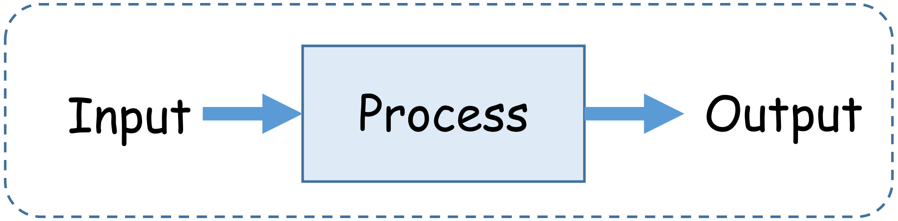
- Other more applications
  - Automatic assembly line
  - Space technology
  - Power systems
  - Smart transportation systems
  - Missile launching systems
  - ...

- What is a control system?

A control system is an interconnection of components forming a **system** configuration that will provide a desired system **response**.
——Richard C. Dorf & Robert H. Bishop, *Modern control systems*

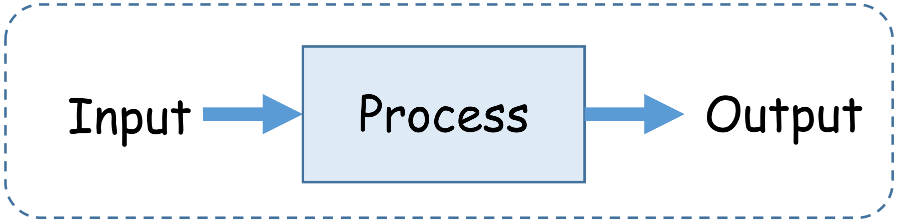
Composition of Control Systems

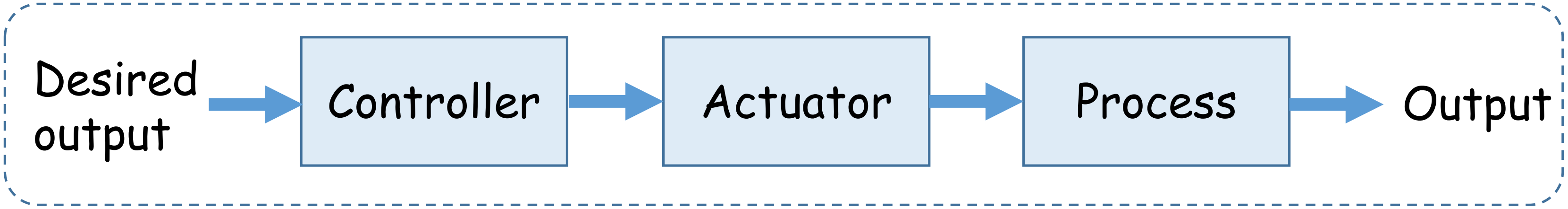
- Linear system
  - Cause-effect relationship for the components
  - Block diagram



Composition of Control Systems

- Linear system
  - Cause-effect relationship for the components
  - Block diagram
- Open-loop control
  - Control the process directly

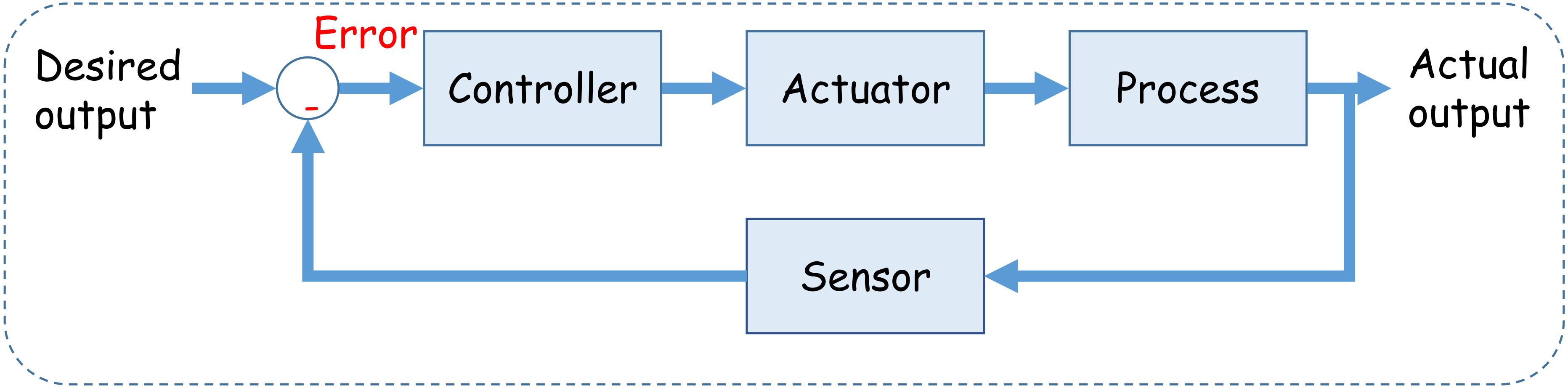




An open-loop control system uses a controller and an actuator to obtain the desired response, without using feedback.

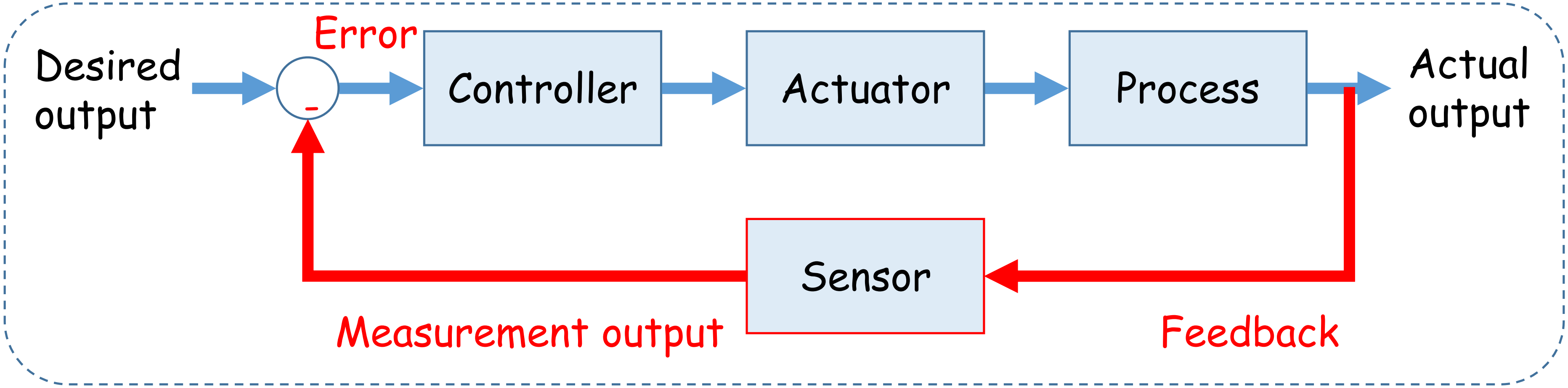
Composition of Control Systems

- Closed-loop system
  - Compare actual output with desire output



Composition of Control Systems

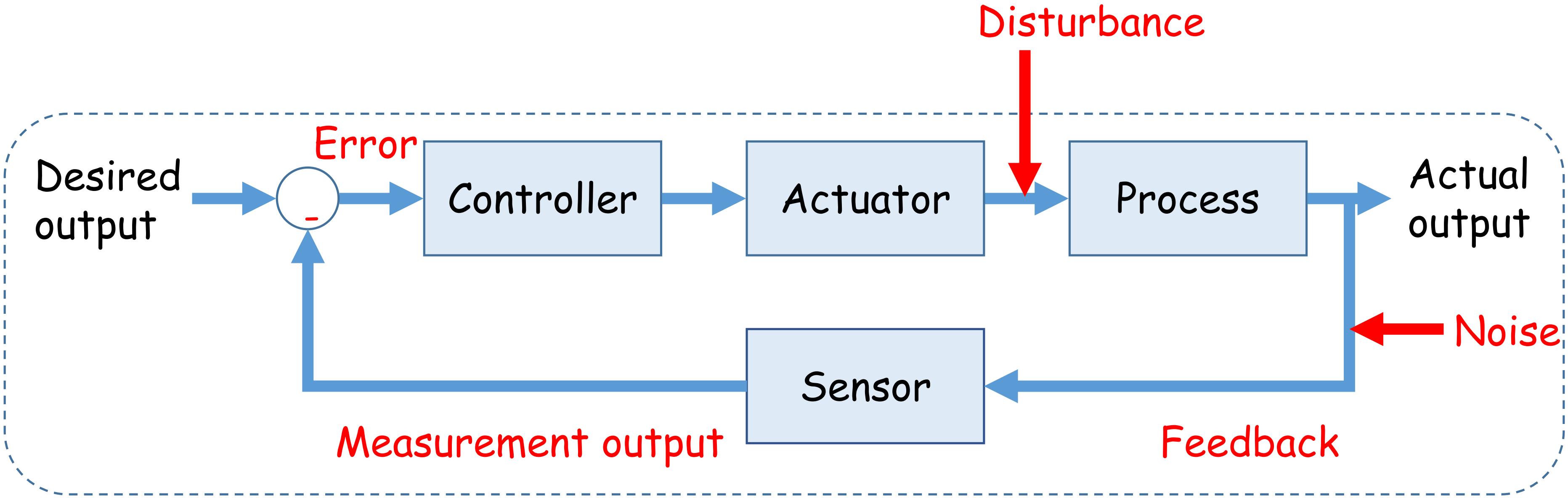
- Closed-loop system
  - Compare actual output with desire output



  - Notice the difference: error based controller

Composition of Control Systems

- Closed-loop system
  - Compare actual output with desire output



  - An actual system also faced with disturbance and noise

Establishment of system model

- Start from a naïve example
  - Consider a autonomous car start up and maintain a constant speed
  - Assuming:

Control variable

1° throttle angle cause a 10km/h change in speed

1° road grade cause a -5km/h change in speed

Disturbance

Establishment of system model

- Start from a naïve example
  - Consider a autonomous car starts up and maintains a constant speed
  - Assuming:

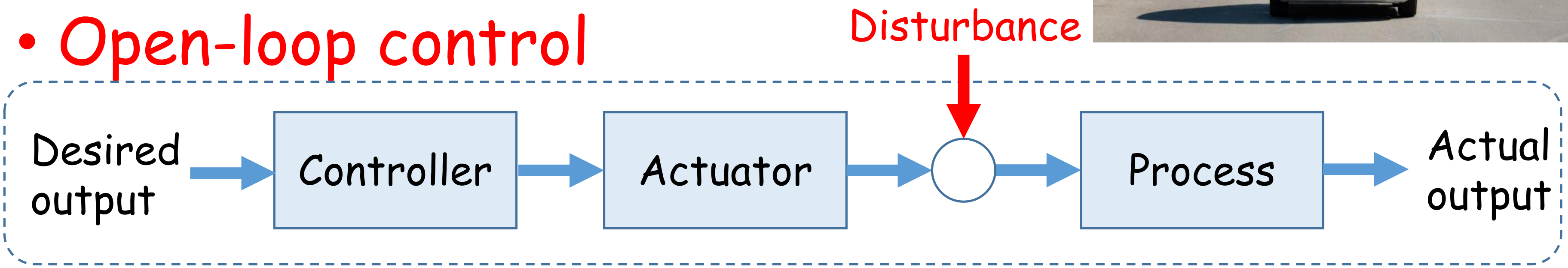
Control variable

1° throttle angle cause a 10km/h change in speed

1° road grade cause a -5km/h change in speed

Disturbance

- Open-loop control



Establishment of system model

- Start from a naïve example
  - Consider a autonomous car start up and maintain a constant speed
  - Assuming:

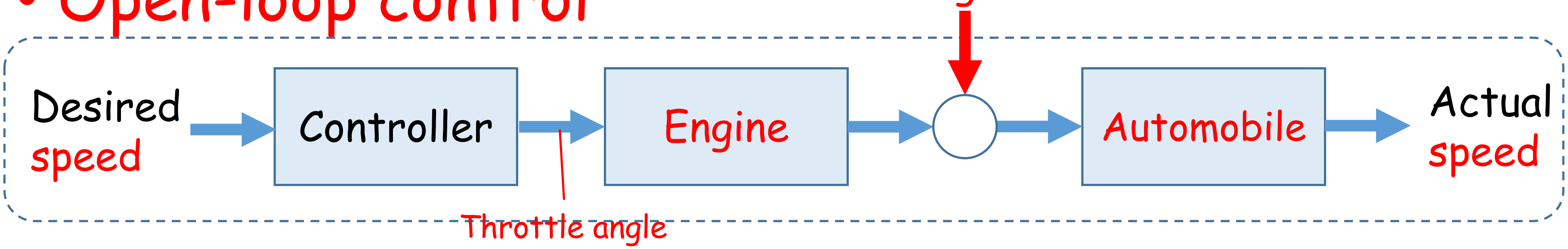
Control variable

1° throttle angle cause a 10km/h change in speed

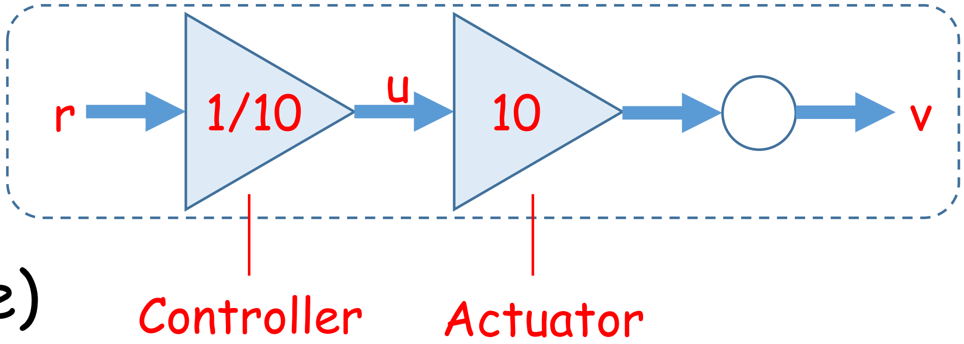
1° road grade cause a -5km/h change in speed

Disturbance

- **Open-loop control**



Establishment of system model



- Open-loop control
  - Parameters definition
  - r: desired speed (reference)
  - u: throttle angle (control variable)
  - w: road grade (disturbance)
  - v: actual speed (output)

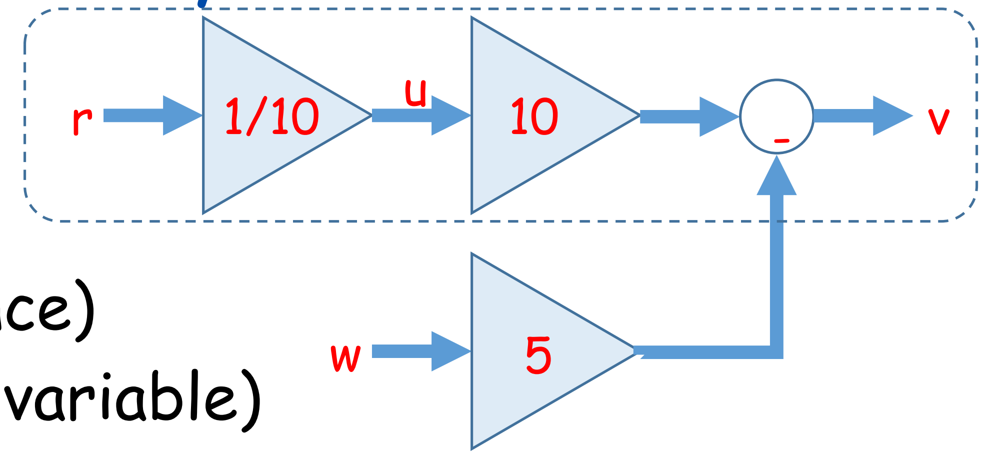
  - If there is no external disturbance (w=0)

$v = 10 * u,\ u = 1/10 * r$ → $v = r$

Assuming desired speed r = 10,

Actual speed v = 10

Establishment of system model



- Open-loop control
  - Parameters definition
  - r: desired speed (reference)
  - u: throttle angle (control variable)
  - w: road grade (disturbance)
  - v: actual speed (output)

  - If there exists external disturbance

$v = 10 * u - 5 * w, \quad u = 1/10 * r \rightarrow v = r - 5 * w$

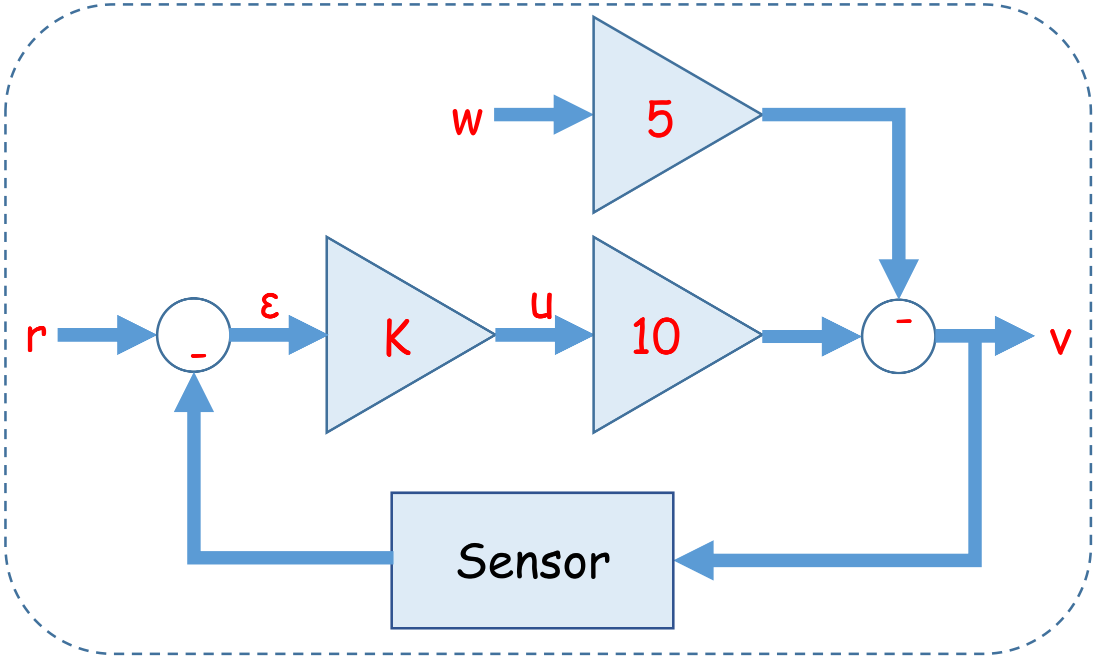
Assuming desired speed r = 10, and small disturbance w = 1

Actual speed v = 5

☹

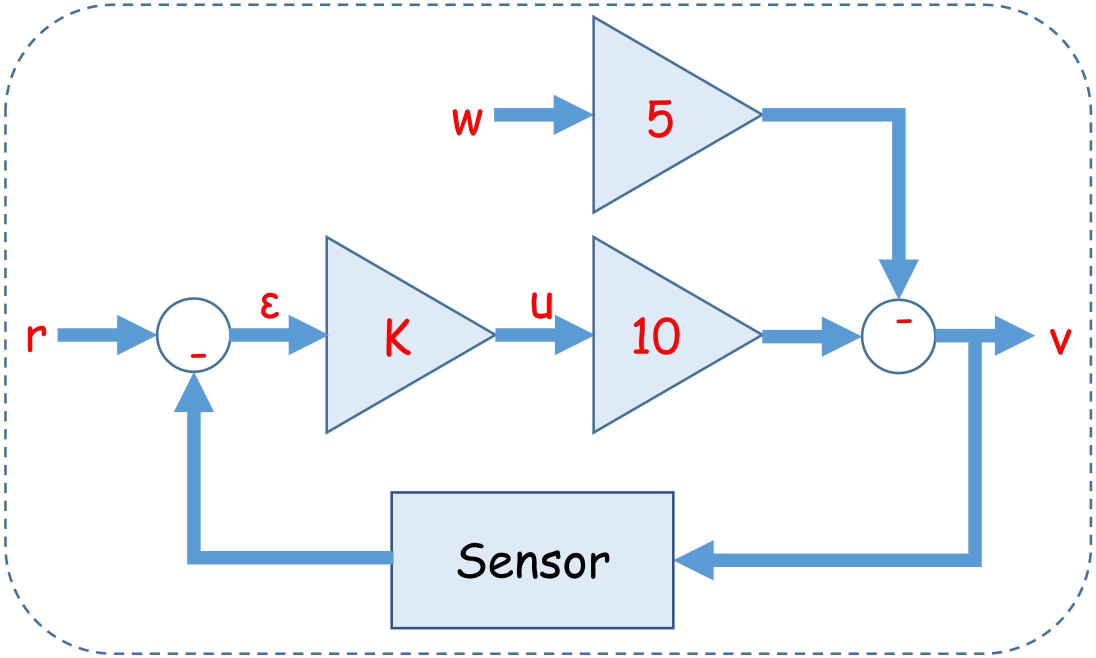
Establishment of system model

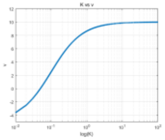
- Closed-loop control
  - ε: difference between actual speed and desired speed (error)
  - K: coefficient (proportional coefficient)



Establishment of system model

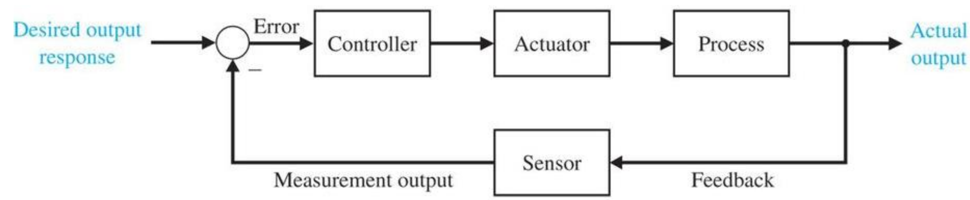
- Closed-loop control
  - ε: difference between actual speed and desired speed (error)
  - K: coefficient (proportional coefficient)

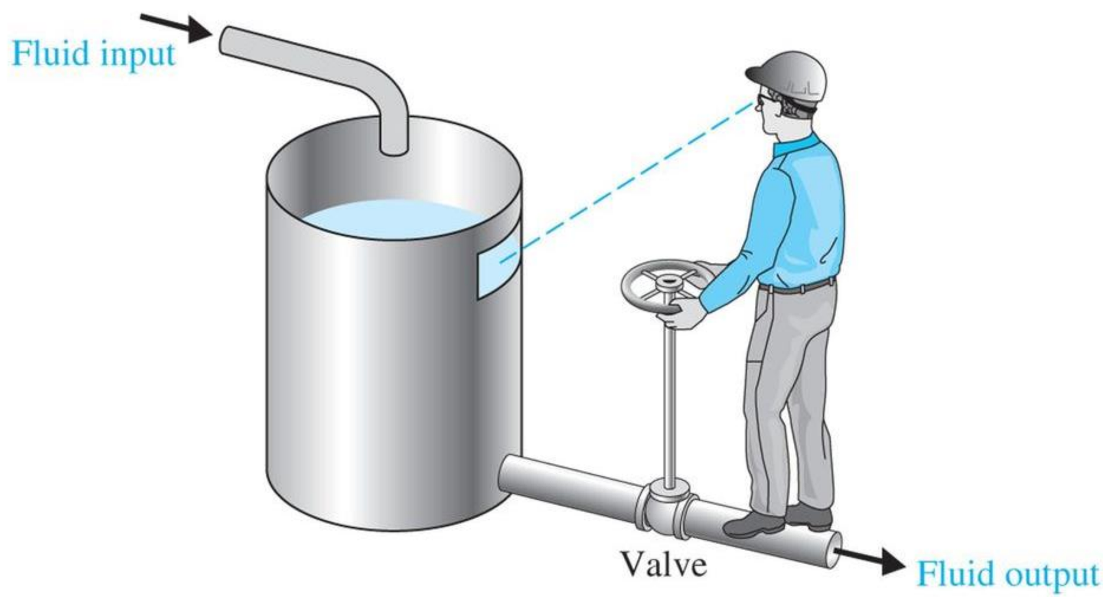




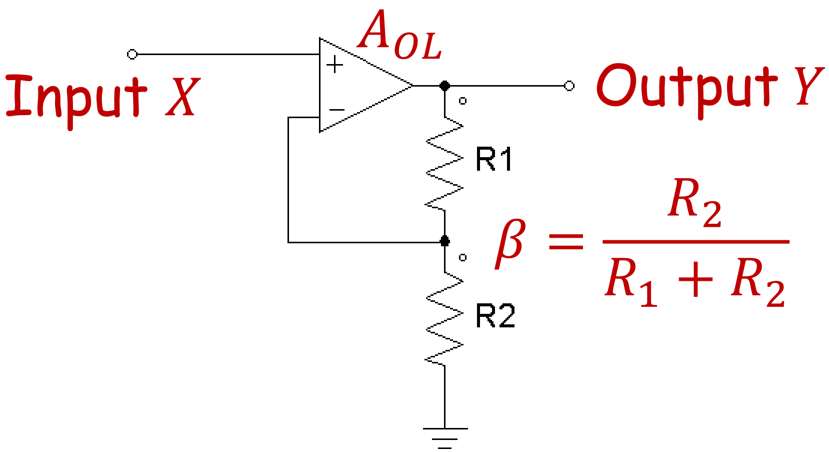
Example 1: manual control system

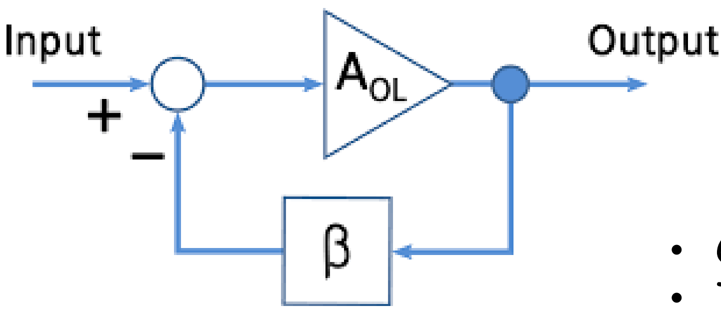
- In this manual control valve system, which one corresponds to the
  - Process
  - Actuator
  - Sensor
  - Controller
  - Desire output
  - Actual output
  - Error





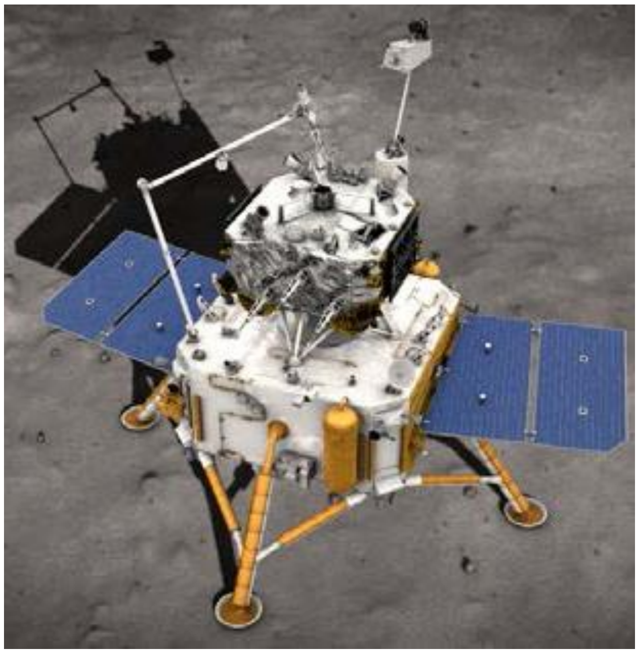
Example 2: Feedback amplifier

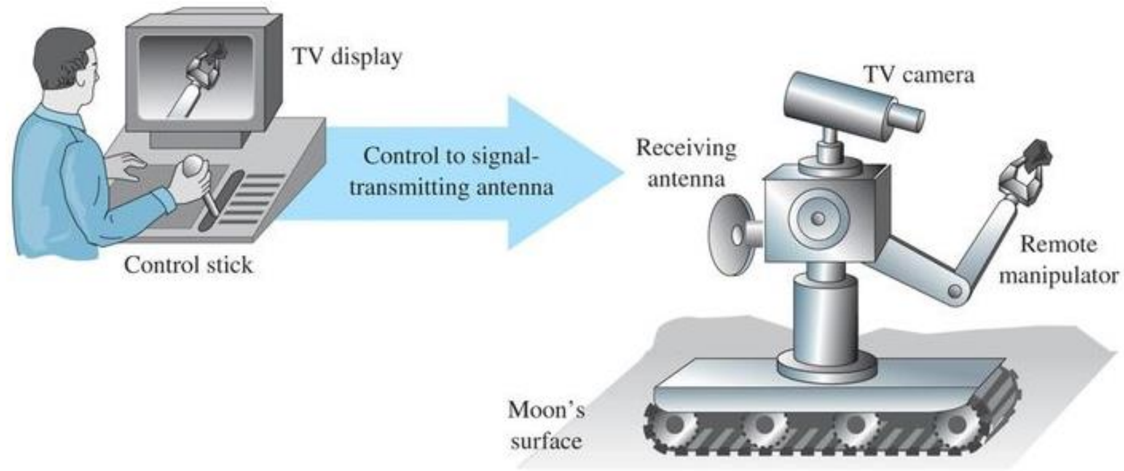




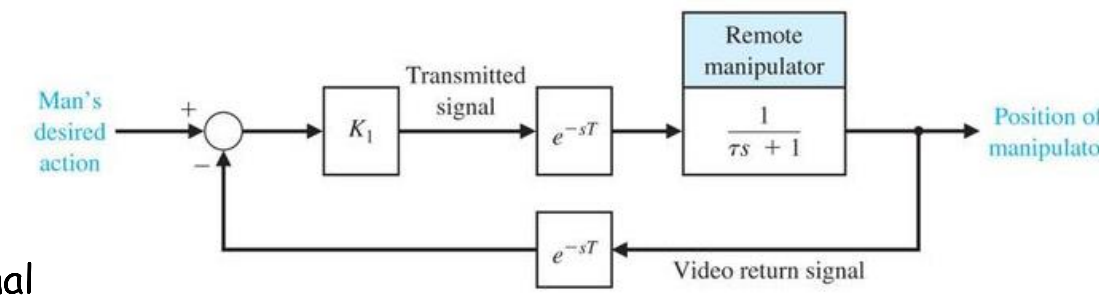
- *Only considered the DC characteristics*
- *The AC characteristics are more complicated*

Example 3: Moon robot





(a)

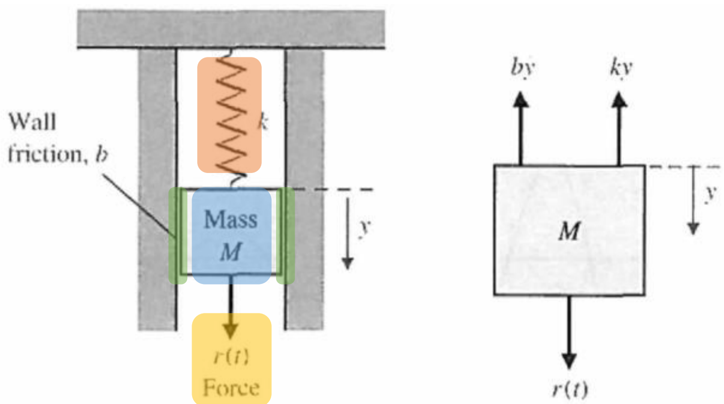


(b)

e^{-sT} models the time delay T in transmission of a communication signal

Differential equations for dynamic modeling

- Governing equation



Mass 质量 | Damping 阻尼 | Spring 弹簧 | Force 力

$$M\frac{d^2y(t)}{dt^2} + b\frac{dy(t)}{dt} + ky(t) = r(t)$$

Table 2.2 Summary of Governing Differential Equations for Ideal Elements

Type of Element	Physical Element	Governing Equation	Energy E or Power $\mathcal{P}$	Symbol
Inductive storage	Electrical inductance	$v_{21} = L\frac{di}{dt}$	$E = \frac{1}{2}Li^2$	
	Translational spring	$v_{21} = \frac{1}{k}\frac{dF}{dt}$	$E = \frac{1}{2}\frac{F^2}{k}$	
	Rotational spring	$\omega_{21} = \frac{1}{k}\frac{dT}{dt}$	$E = \frac{1}{2}\frac{T^2}{k}$	
	Fluid inertia	$P_{21} = I\frac{dQ}{dt}$	$E = \frac{1}{2}IQ^2$	
Capacitive storage	Electrical capacitance	$i = C\frac{dv_{21}}{dt}$	$E = \frac{1}{2}Cv_{21}^2$	
	Translational mass	$F = M\frac{dv_2}{dt}$	$E = \frac{1}{2}Mv_2^2$	
	Rotational mass	$T = J\frac{d\omega_2}{dt}$	$E = \frac{1}{2}J\omega_2^2$	
	Fluid capacitance	$Q = C_f\frac{dP_{21}}{dt}$	$E = \frac{1}{2}C_fP_{21}^2$	
	Thermal capacitance	$q = C_t\frac{d\mathcal{T}_2}{dt}$	$E = C_t\mathcal{T}_2$	
Energy dissipators	Electrical resistance	$i = \frac{1}{R}v_{21}$	$\mathcal{P} = \frac{1}{R}v_{21}^2$	
	Translational damper	$F = bv_{21}$	$\mathcal{P} = bv_{21}^2$	
	Rotational damper	$T = b\omega_{21}$	$\mathcal{P} = b\omega_{21}^2$	
	Fluid resistance	$Q = \frac{1}{R_f}P_{21}$	$\mathcal{P} = \frac{1}{R_f}P_{21}^2$	
	Thermal resistance	$q = \frac{1}{R_t}\mathcal{T}_{21}$	$\mathcal{P} = \frac{1}{R_t}\mathcal{T}_{21}$	

Frequency-domain expressions

$$M\frac{d^2y(t)}{dt^2} + b\frac{dy(t)}{dt} + ky(t) = r(t)$$

- Transfer function
  - Describing the dynamic relationship of the system in the frequency domain

Laplace transform

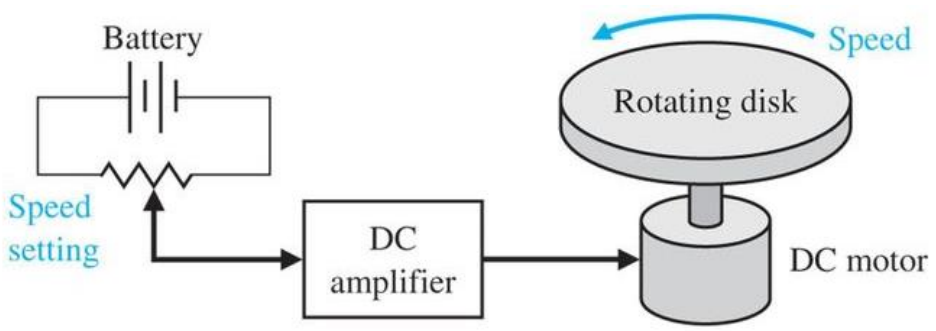
$$Ms^2Y(s) + bsY(s) + kY(s) = R(s)$$

Input/output relation

$$\frac{\text{Output}}{\text{Input}} = \mathrm{G}(s) = \frac{Y(s)}{R(s)} = \frac{1}{Ms^2 + bs + k}$$

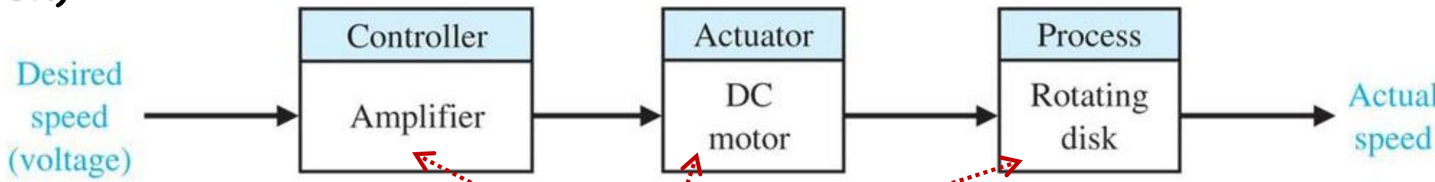
Block diagram

A practical rotational speed control system



(a)

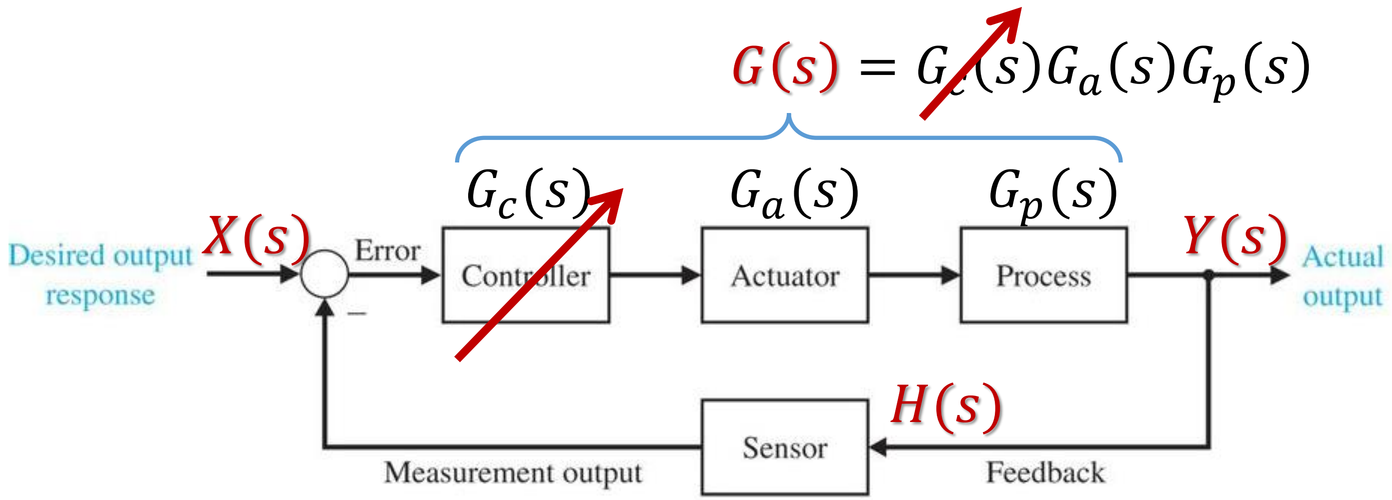
The corresponding block diagram (abstraction)



(b)

Representing a time-domain or frequency-domain (mathematical) model

Mathematical model of feedback control system



Tuning the close-loop system characteristics by changing the controller characteristics

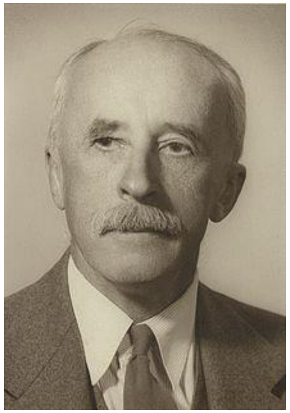
Open-loop gain

Close-loop gain

PID controller

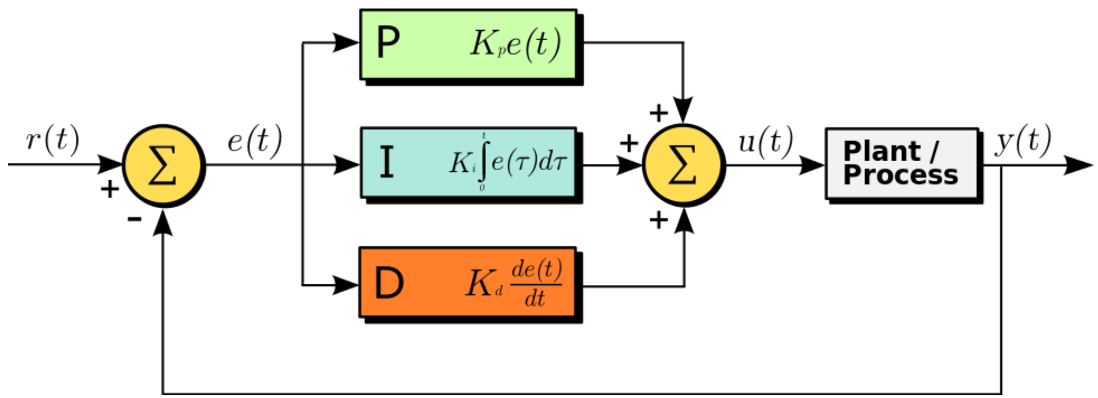
• A Brief History of PID Control

- 1890's, PID (Proportional - Integral – Derivative) Control, originally developed in the form of motor governors, which were manually adjusted
- 1922, the first theory of PID Control was published by Nicolas Minorsky, who was working for the US Navy
- 1940's, the first papers regarding PID tuning appeared
  - there are several hundred different rules for tuning PID controllers (See Dwyer, 2009)
- Nowadays, **97%** of regulatory controllers utilize PID feedback
  - based on a survey of over eleven thousand controllers in the refining, chemicals and pulp and paper industries (see Desborough and Miller, 2002).

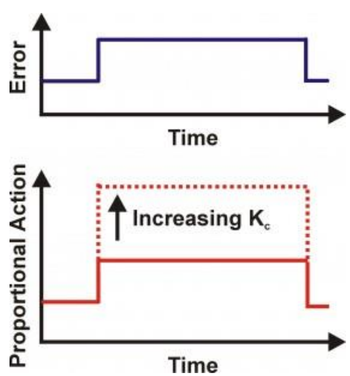


Nicolas Minorsky
(1885- 1970)
a Russian American control theory mathematician, engineer and applied scientist

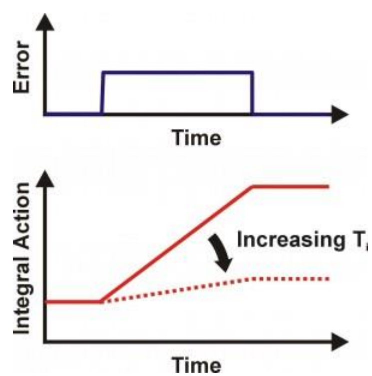
PID controller



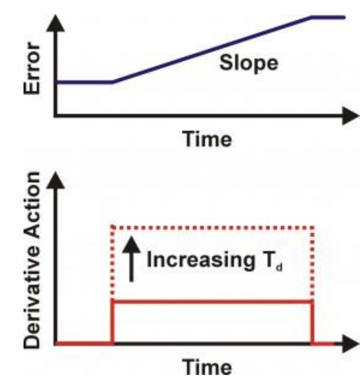
Current error



Past error



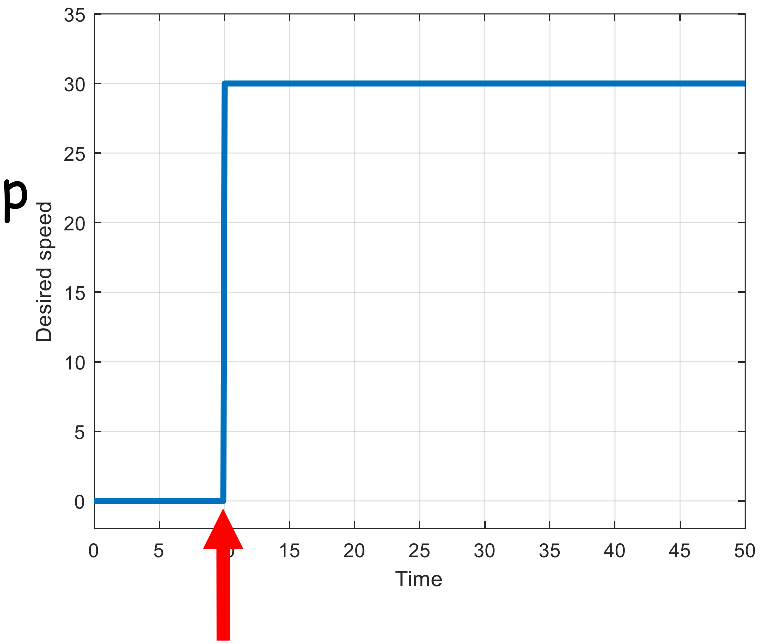
Future error



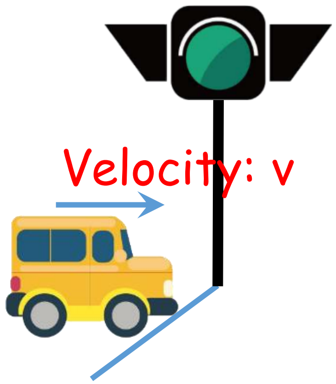
Composition of PID controller

- **Case 1**
  - Autonomous car stops at a red light
  - At t_0, light turns green and car starts up
  - And finally reaches desired speed

  - Desired speed is a step function at t_0



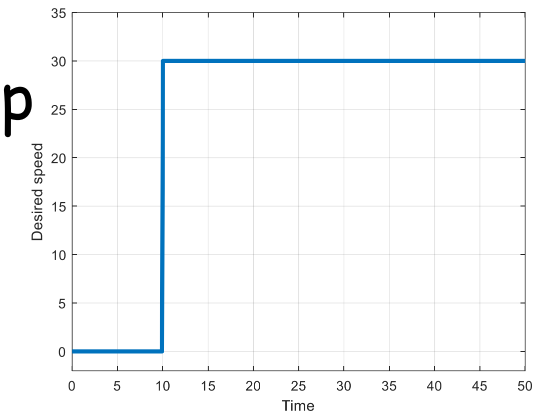
t_0

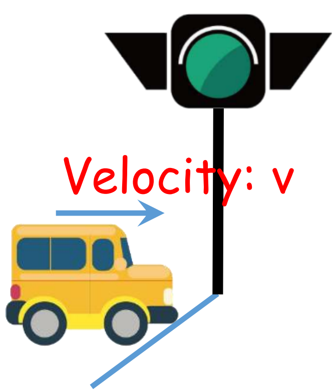


Composition of PID controller

- Case 1
  - Autonomous car stops at a red light
  - At t_0, light turns green and car starts up
  - And finally reaches desired speed

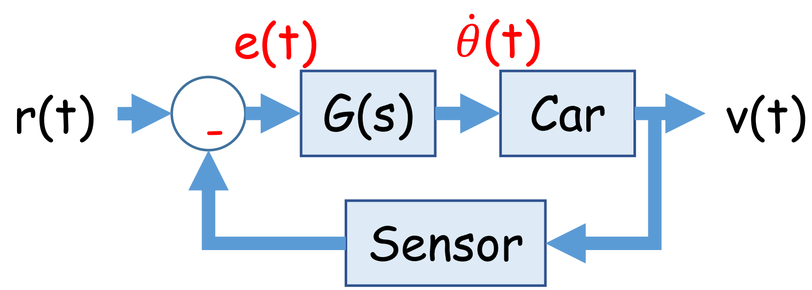
  - Consider a proportional control only





G(s) = Kp, Kp>0

$\dot{\theta}$(t) = Kp*e(t)



θ represents the throttle angle

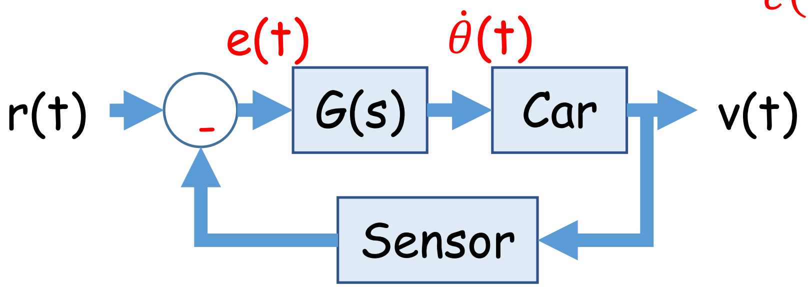
and $\dot{\theta}$ is the derivative, which represents the change in speed

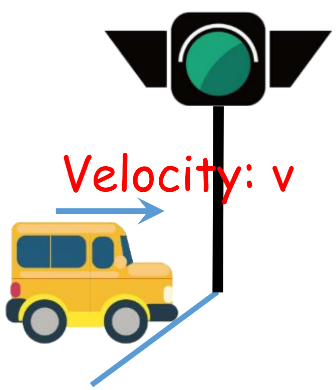
Composition of PID controller

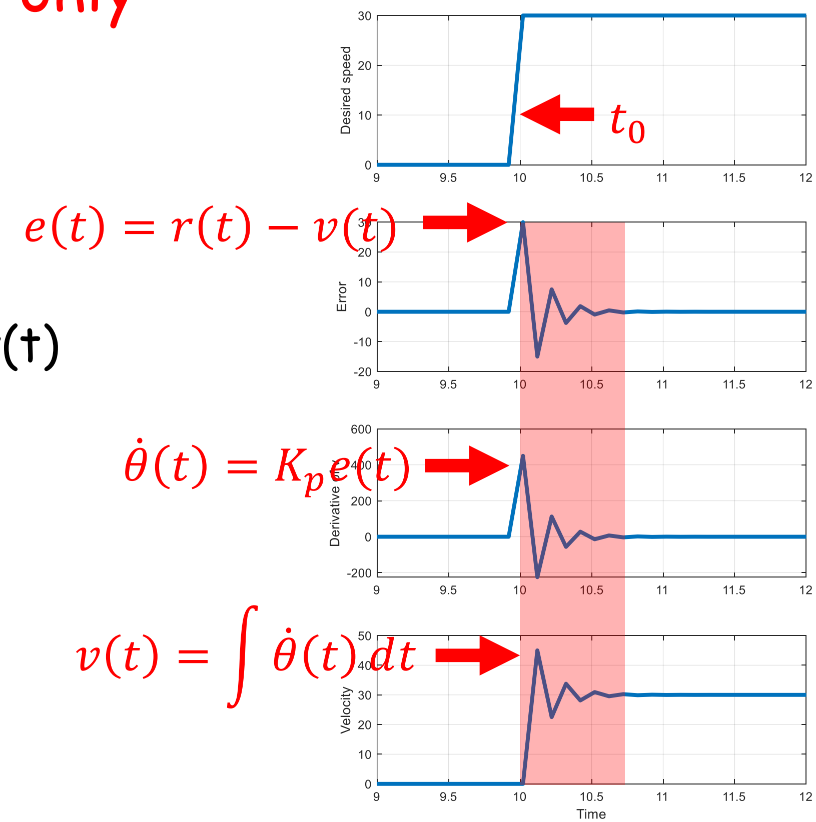
- Case 1
  - Consider a proportional control only

$$G(s) = Kp,\ Kp>0$$

$$\dot{\theta}(t) = Kp*e(t)$$







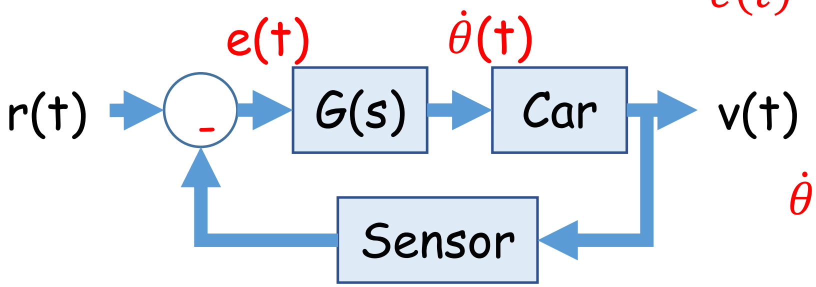
Notice the oscillation in velocity, due to an aggressive Kp

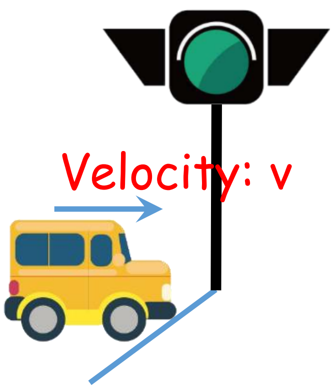
Composition of PID controller

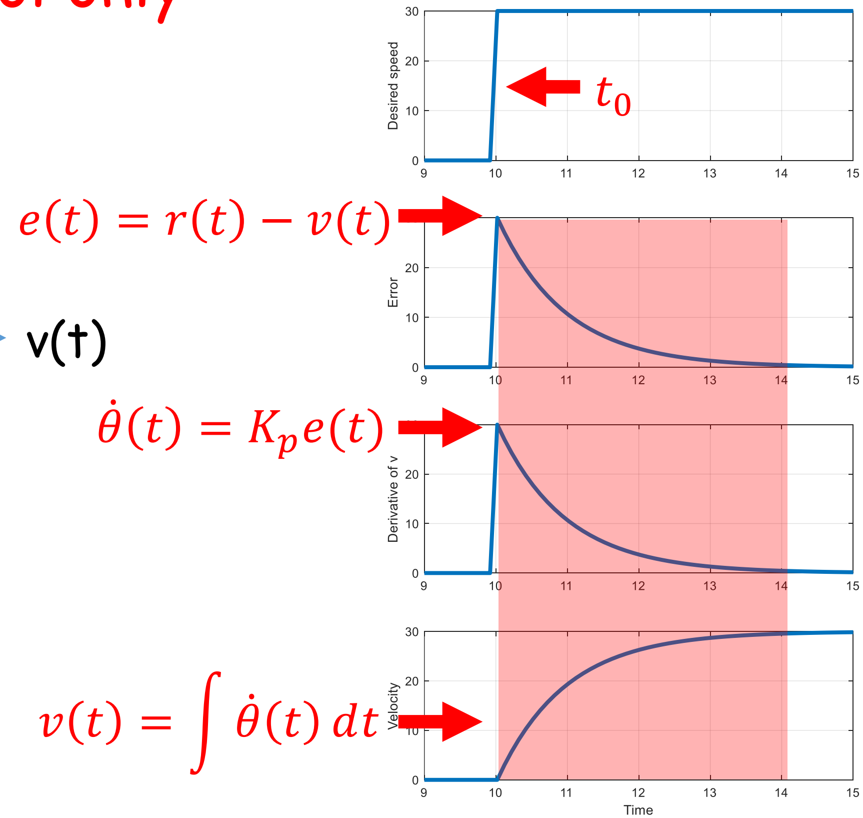
- Case 1
  - Consider a proportional control only

$$G(s) = Kp,\ Kp>0$$

$$\dot{\theta}(t) = Kp*e(t)$$



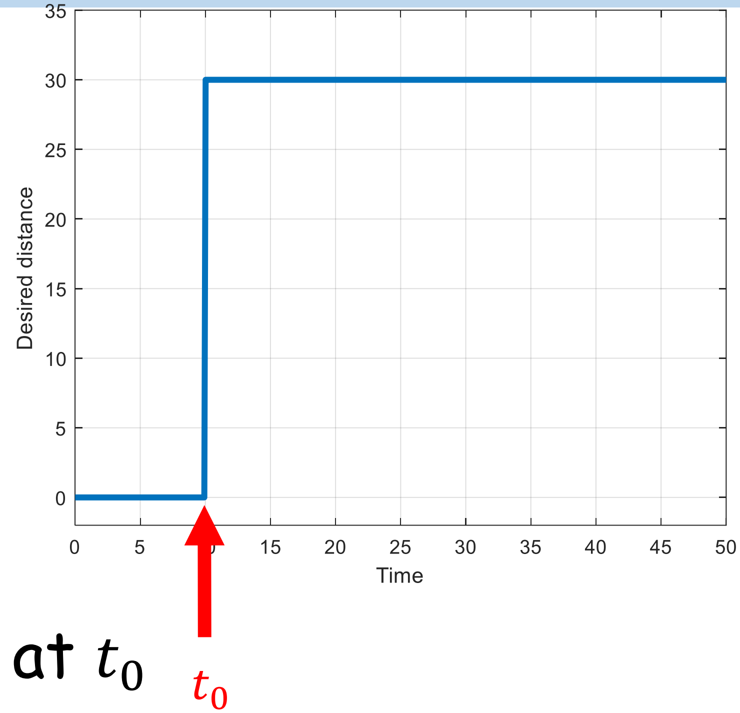




Smaller Kp reduces oscillation, but is more time-consuming

Composition of PID controller

- Case 2
  - Autonomous car stops at a red light
  - Another red light some distance away
  - At t_0, light turns green and car starts up
  - And finally stops at the second light



  - Desired distance is a step function at t_0

Composition of PID controller

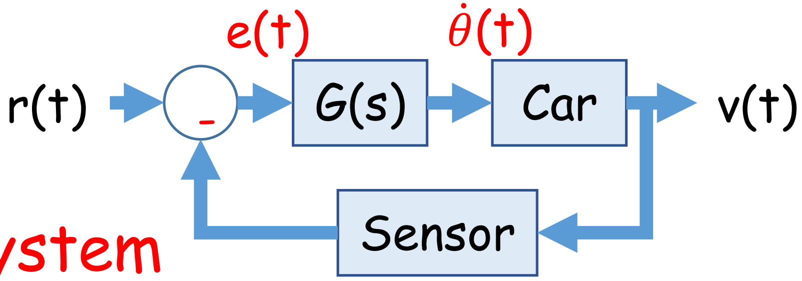
- Case 2
  - If proportional control only

    G(s) = Kp, Kp>0

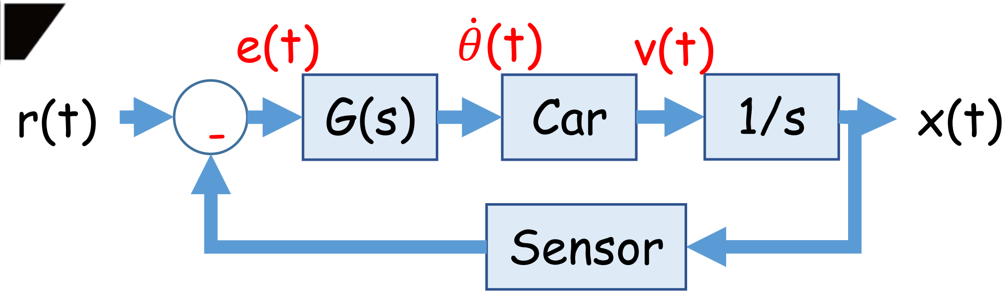
    $\dot{\theta}(t)$ = Kp*e(t)

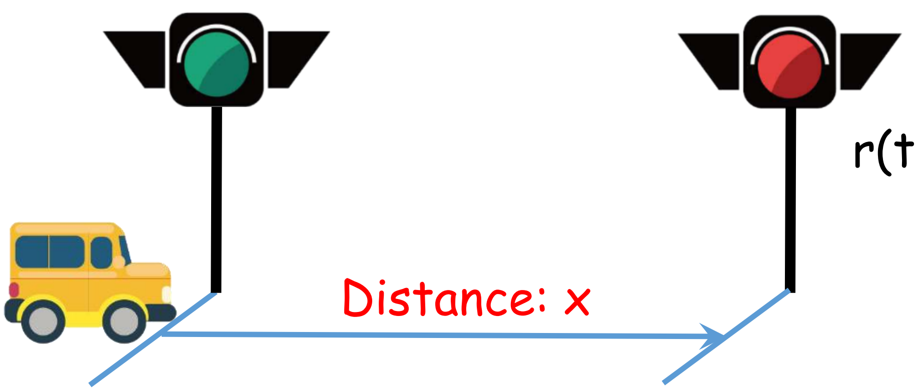
  - Notice the change in control system

Control system of Case 1



Control system of Case 2





Composition of PID controller

- Case 2
  - If proportional control only

    G(s) = Kp, Kp>0

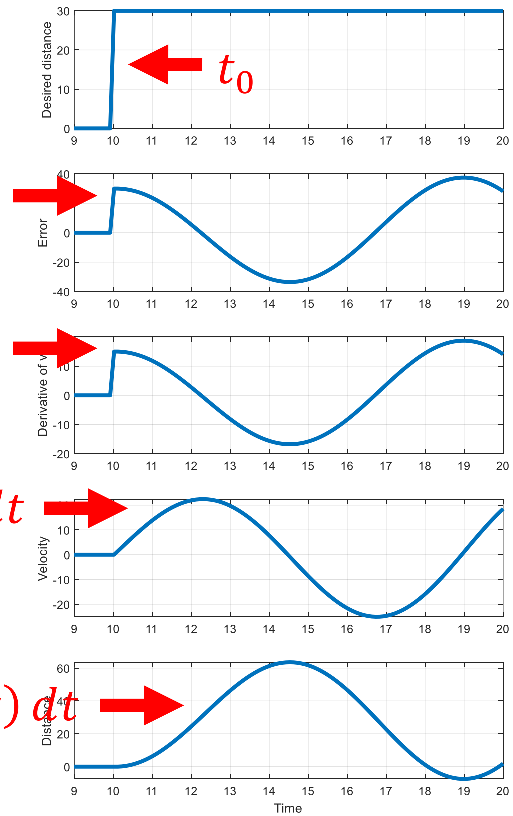
    $\dot{\theta}(t)$ = Kp*e(t)
  - On previous experience, choose small Kp

$$e(t) = r(t) - x(t)$$

$$\dot{\theta}(t) = K_p e(t)$$

$$v(t) = \int \dot{\theta}(t)\, dt$$

$$x(t) = \int v(t)\, dt$$



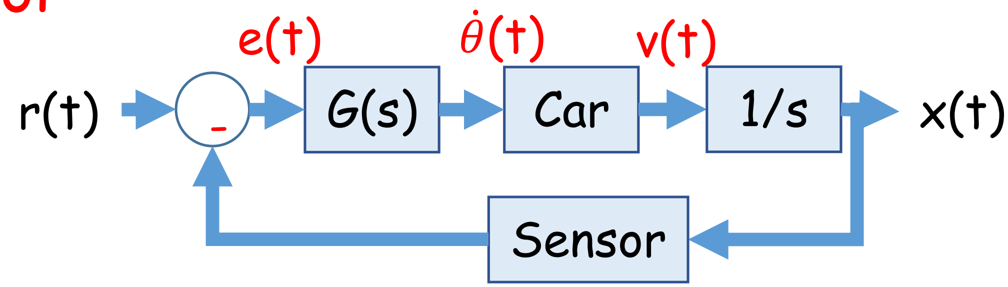
Why is this happening?

Composition of PID controller

- Case 2
  - Introduce derivative control

$G(s) = Kp + Kd*s,\ Kp,Kd>0$

$\dot{\theta}(t) = Kp*e(t) + Kd*\dot{e}(t)$



  - Therefore, proportional-derivative (PD) control

Composition of PID controller

- Case 2
  - Introduce derivative control

    G(s) = Kp + Kd*s, Kp,Kd>0

    $\dot{\theta}(t)$ = Kp*e(t) + Kd*$\dot{e}(t)$

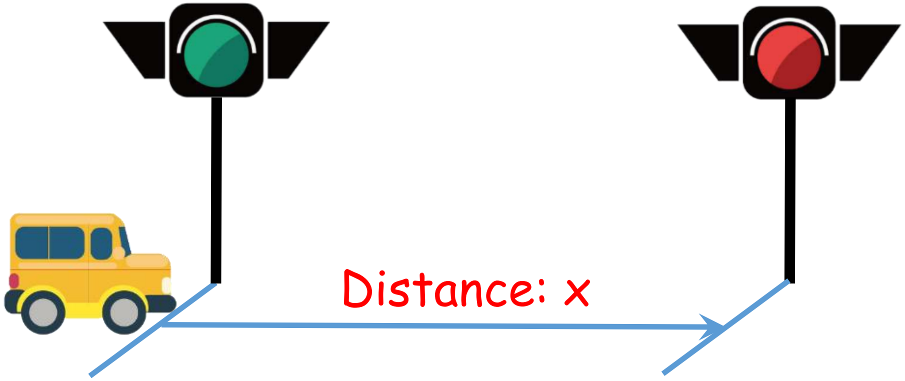
  - Therefore, proportional-derivative (PD) control

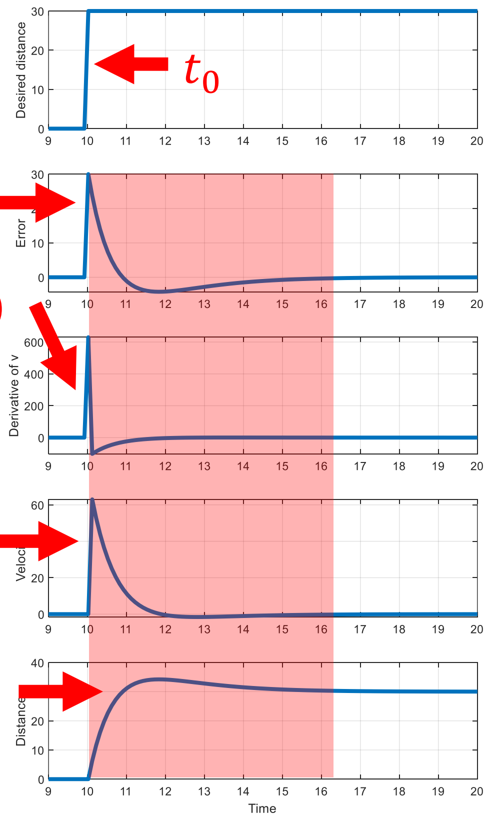
$e(t) = r(t) - x(t)$

$\dot{\theta}(t) = K_p e(t) + K_d \dot{e}(t)$

$v(t) = \int \dot{\theta}(t)\, dt$

$x(t) = \int v(t)\, dt$

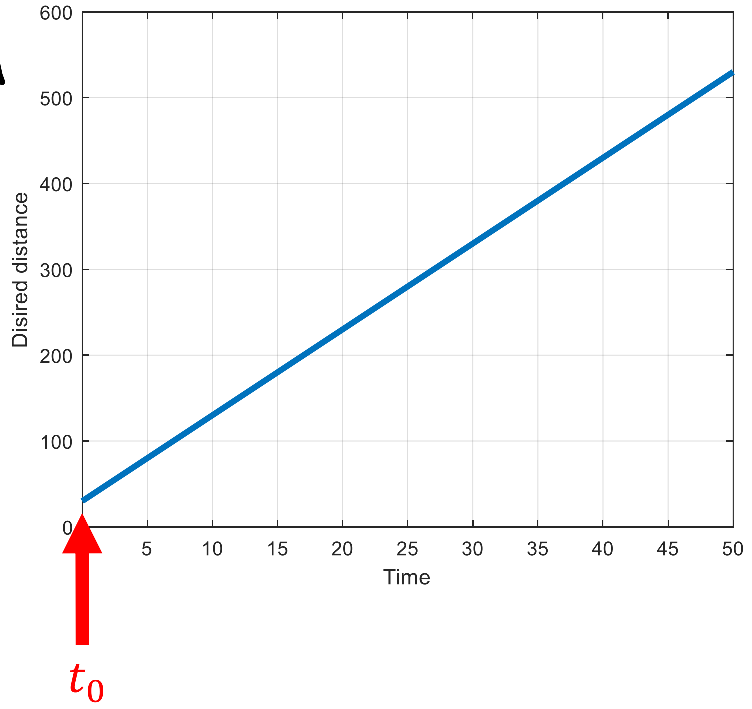


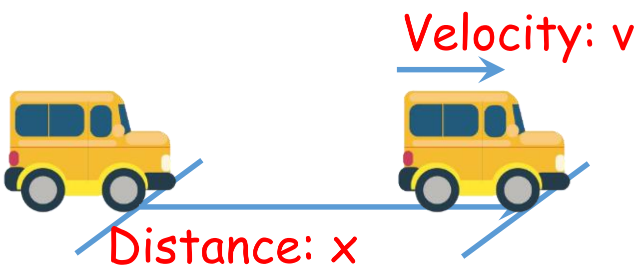


Similarly, oscillation exist (inevitable), but much smaller

Composition of PID controller

- Case 3
  - Car A runs at a constant speed
  - Car B starts up to catch up with A
  - Finally two cars drive side by side
  - Desired distance is a linear function
  - And what is the control variable this time?





Composition of PID controller

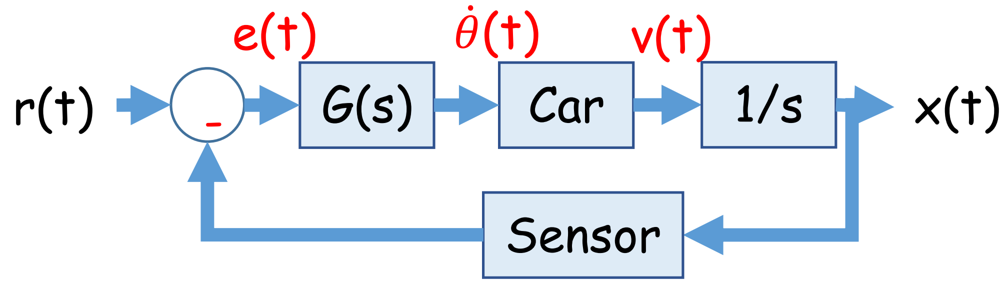
- Case 3
  - If still only consider proportional control

    $G(s) = Kp, Kp>0$

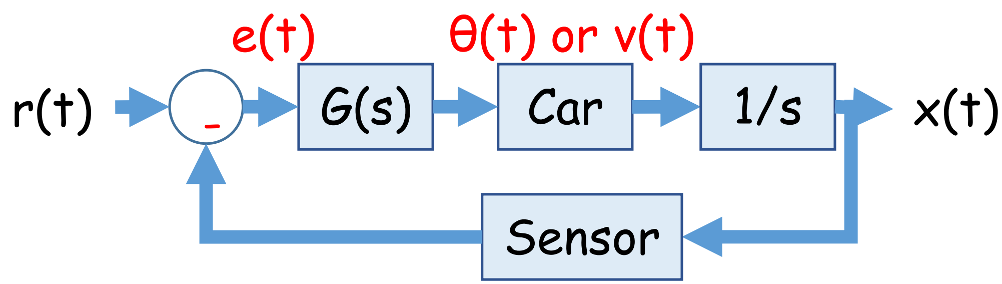
    $\theta(t) = Kp*e(t)$

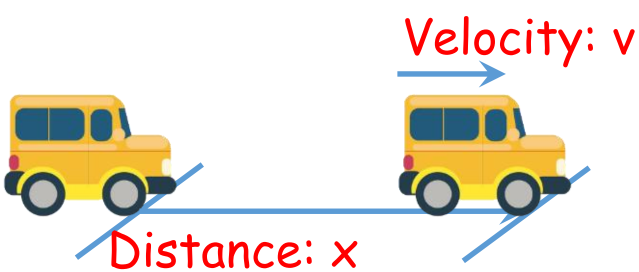
  - Notice this time the control variable is velocity (or throttle angle θ)

Control system of Case 2



Control system of Case 3





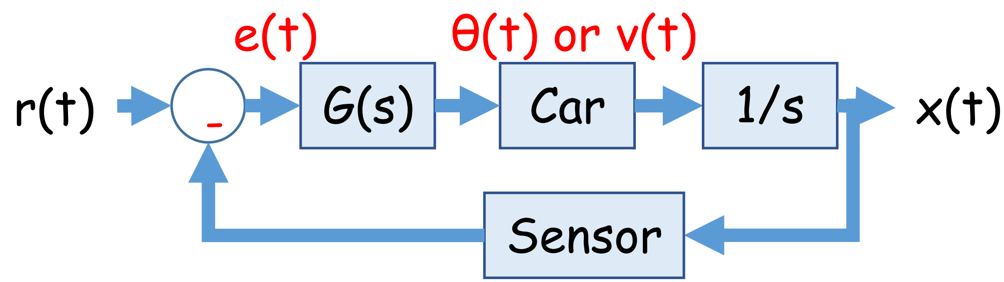
Composition of PID controller

- Case 3
  - If still only consider proportional control

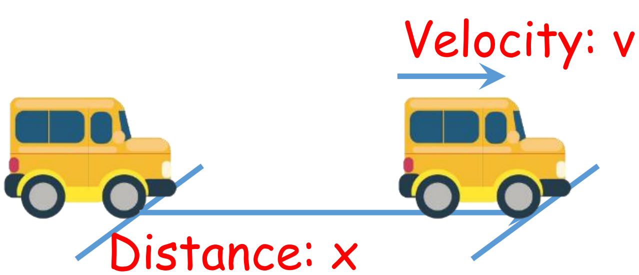
    G(s) = Kp, Kp>0

    v(t) = Kp*e(t)

Control system of Case 3



  - On previous experience, choose small Kp

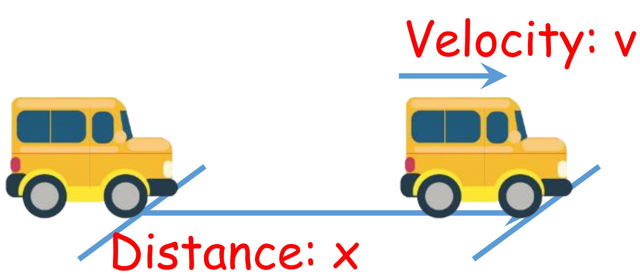


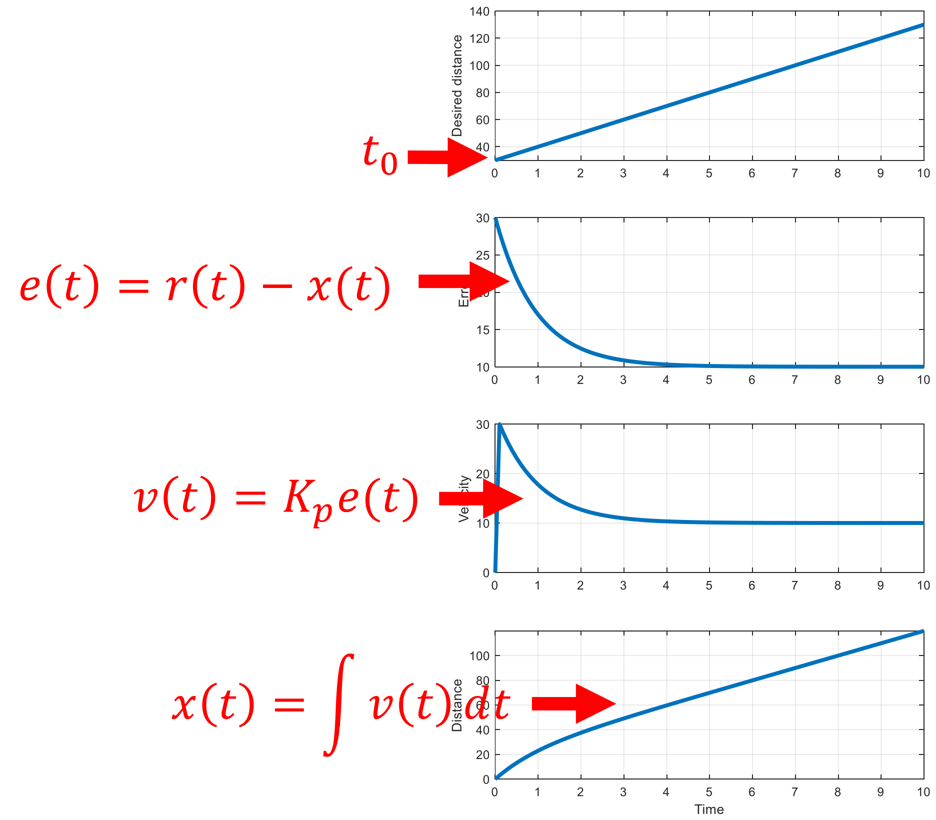
Composition of PID controller

- Case 3
  - If still only consider proportional control

    G(s) = Kp, Kp>0

    v(t) = Kp*e(t)
  - On previous experience, choose small Kp
  - Cannot catch up
  - Final v=Kp*e(t) and e(t) maintains





Composition of PID controller

- Case 3
  - Introduce integral control

    G(s) = Kp + Ki*(1/s), Kp,Ki>0

    v (t) = Kp*e(t) + Ki*$\int e(t)\,dt$

  - Therefore, proportional-integral (PI) control

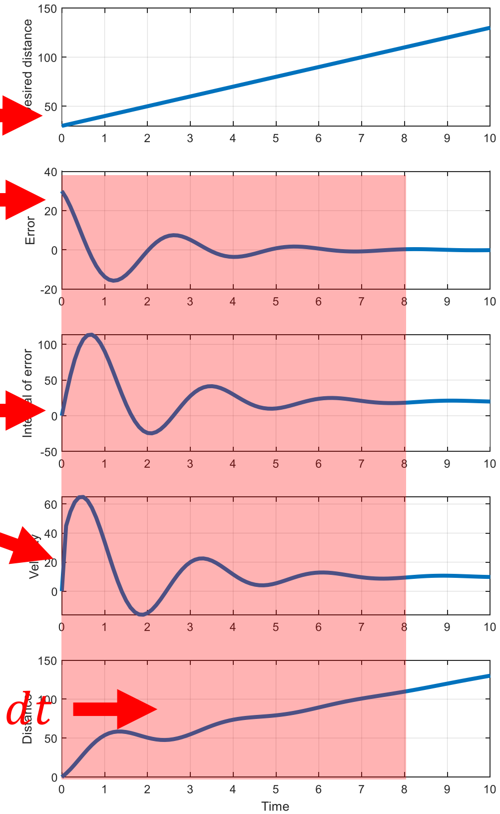
t_0

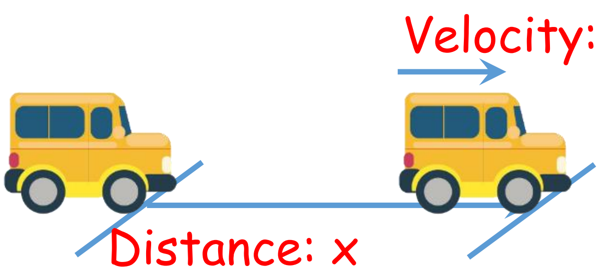
$e(t) = r(t) - x(t)$

$\int e(t)dt$

$v(t) = K_p e(t) + K_i \int e(t)\,dt$

$x(t) = \int v(t)\,dt$

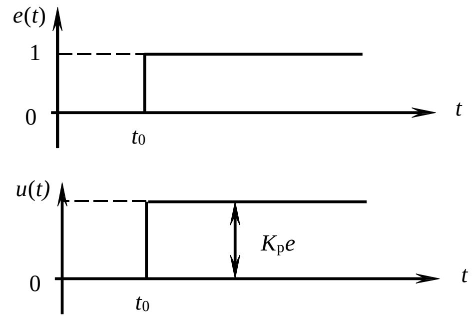




Oscillation inevitable, and integral part increases overshoot

Mathematical analysis of PID controller

- P Control
  - Proportional control (P): accounts for present values of the error
    - U —— control signal
    - K_p—— proportional gain
    - e —— error signal



Step response for P control

- In the Laplace domain

- Pros&Cons
  - Rapid response to track the error signal
  - Steady-state error
  - Prone to be unstable for large K_p

- Proportional control is always present, either by itself, or allied with derivative and/or integral control

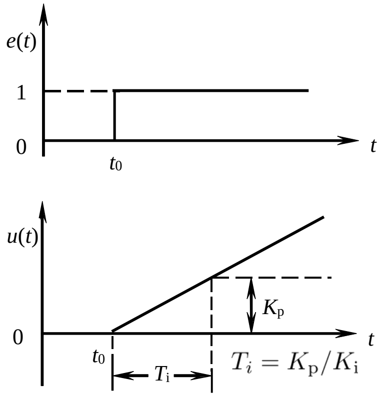
Mathematical analysis of PID controller

- I Control

- Integral control (I): accounts for past values of the error

  - U —— control signal
  - K_i—— integral gain
  - e —— error signal

- In the Laplace domain



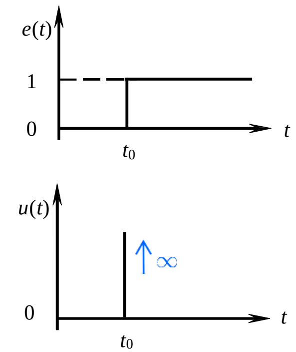
Step response for I control

- Pros&Cons
  - Eliminates the steady-state error that occurs with pure P control
  - Prone to cause the present value to overshoot the setpoint (responds to accumulated errors from the past)

Mathematical analysis of PID controller

- D Control

- Derivative control (D): accounts for possible future trends of the error
  - U —— control signal
  - K_d—— derivative gain
  - e —— error signal



Step response for D control

- In the Laplace domain

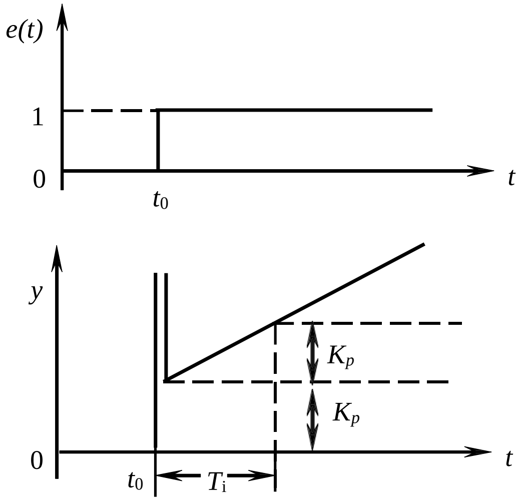
- Pros&Cons
  - Predicts system behavior and thus improves settling time/transient response and stability of the system
  - Helps reduce overshoot, but amplifies noise (derivative kick)
  - Seldom used in practice, 80% of the employed PID controllers have the D part switched-off (see Ang et al., 2005)

Mathematical analysis of PID controller

- PID Control

- Proportional integral derivative control (PID): a combination of P, I and D control

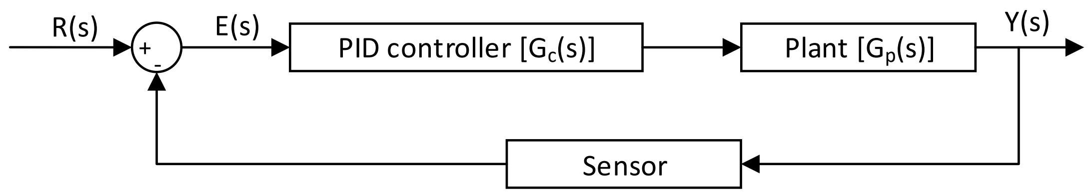
- In the Laplace domain



Step response for PID control

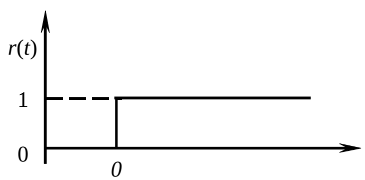
Mathematical analysis of PID controller

- Steady-state error



Input signal: unit step signal

Close-loop gain for PID

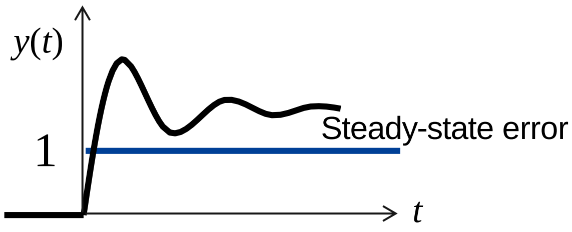


Plant: 2nd order system

Mathematical analysis of PID controller

- Steady-state error

- P control



- Final-value theorem

  - Steady-state error always occurs;
  - Larger K_p makes steady state error goes to zero

Mathematical analysis of PID controller

- Steady-state error

- PD control

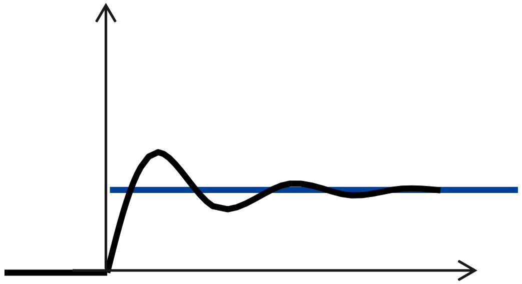
- Final-value theorem

  - Steady-state error remains
  - D control does not track error, only affect the rate of change

Mathematical analysis of PID controller

- Steady-state error

- PI control

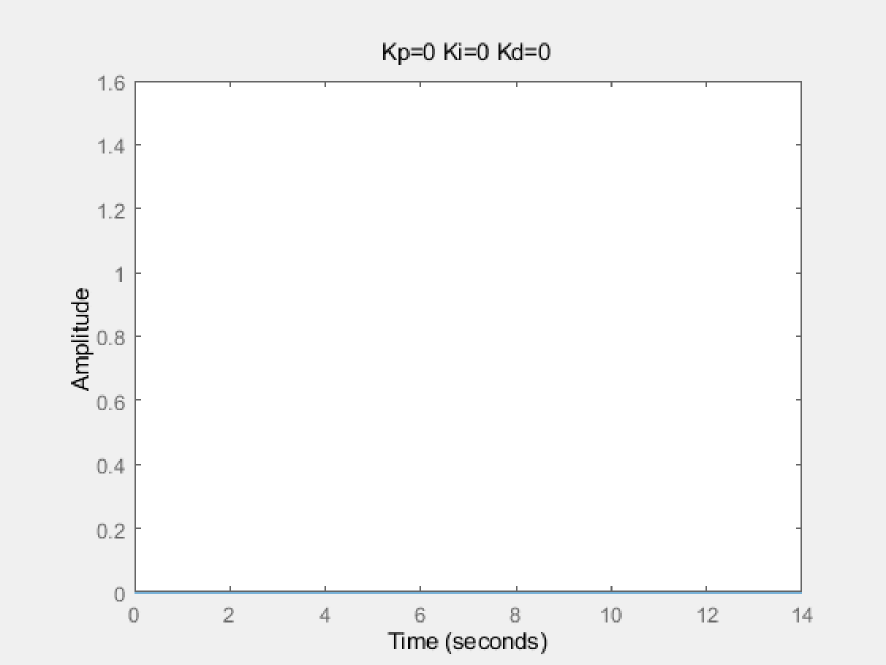


- Final-value theorem

  - Steady-state error is zero for a step reference, even for small K_i (just takes longer to reach steady state).

PID controller

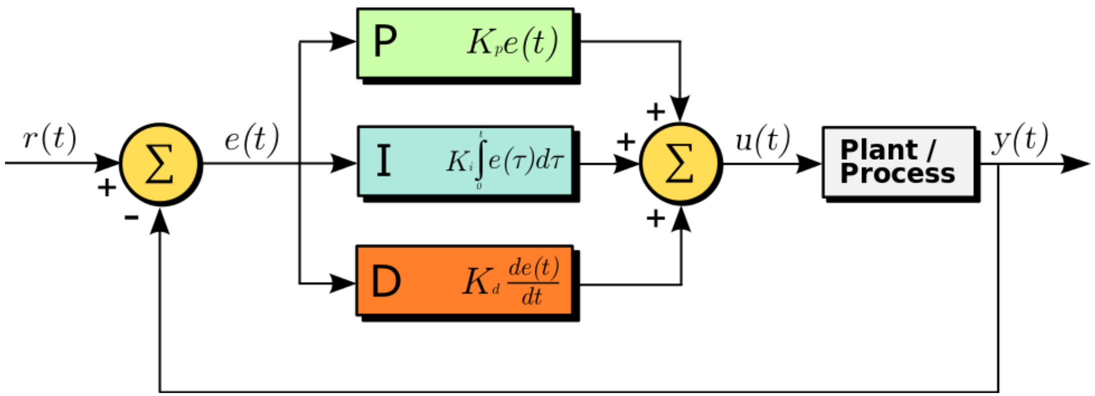
- Steady-state error



- PID control

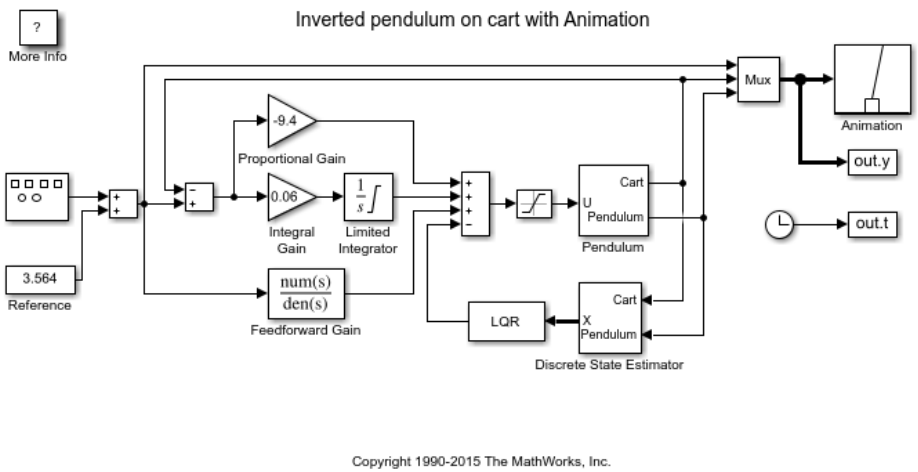
- Final-value theorem

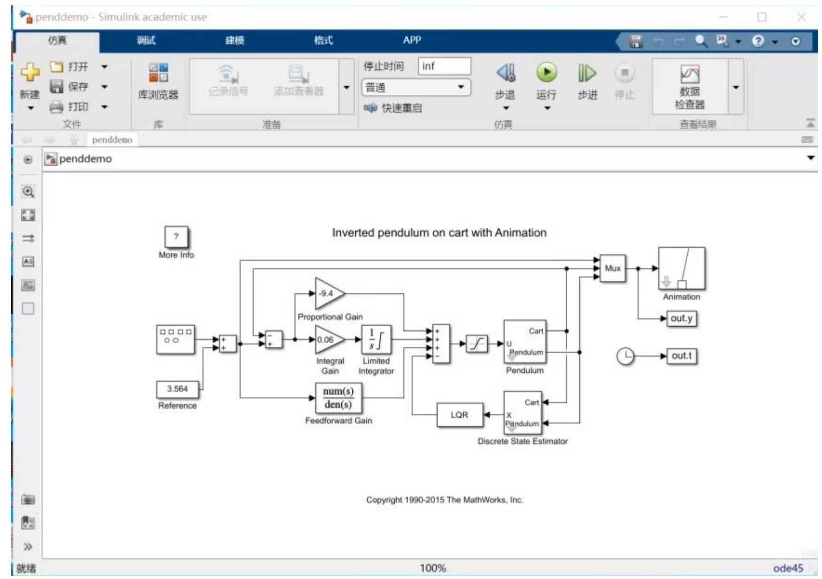
Summary of PID controller



PID gain	Overshoot	Settling-time	Steady-state error
Increasing Kp	Increases	Minimal impact	Decreases
Increasing Ki	Increases	Increases	Zero error
Increasing Kd	Decreases	Decreases	No impact

Inverted pendulum example in Matlab





Key in the command:
>> openExample('simulink_general/penddemoExample')