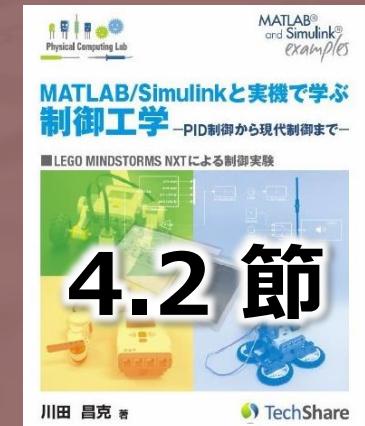
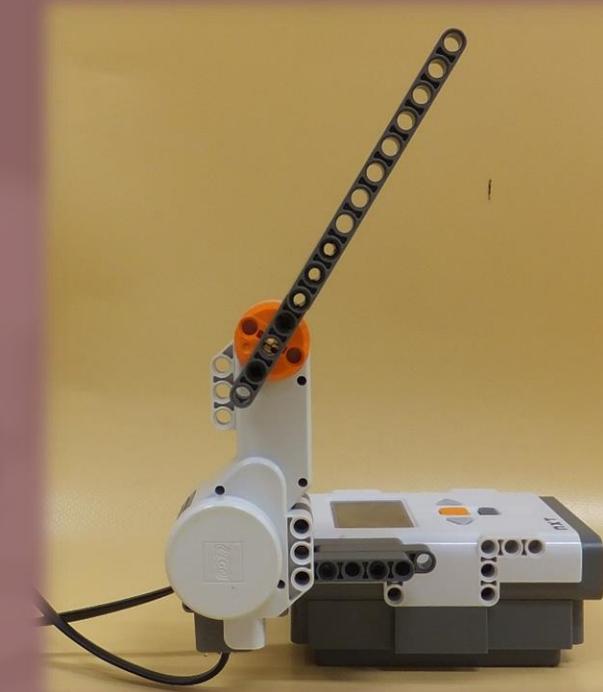
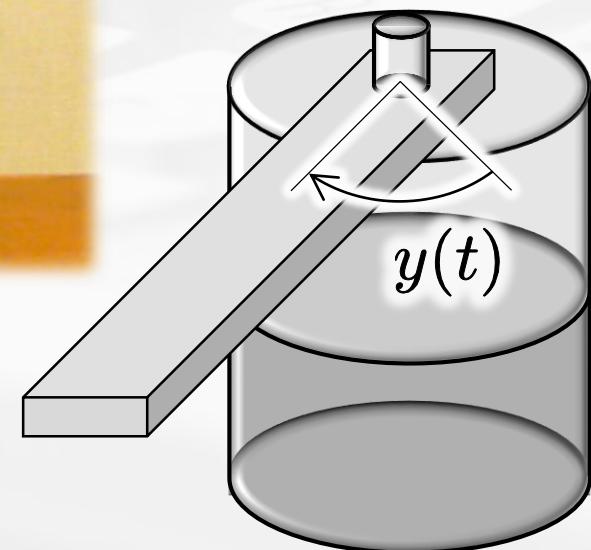
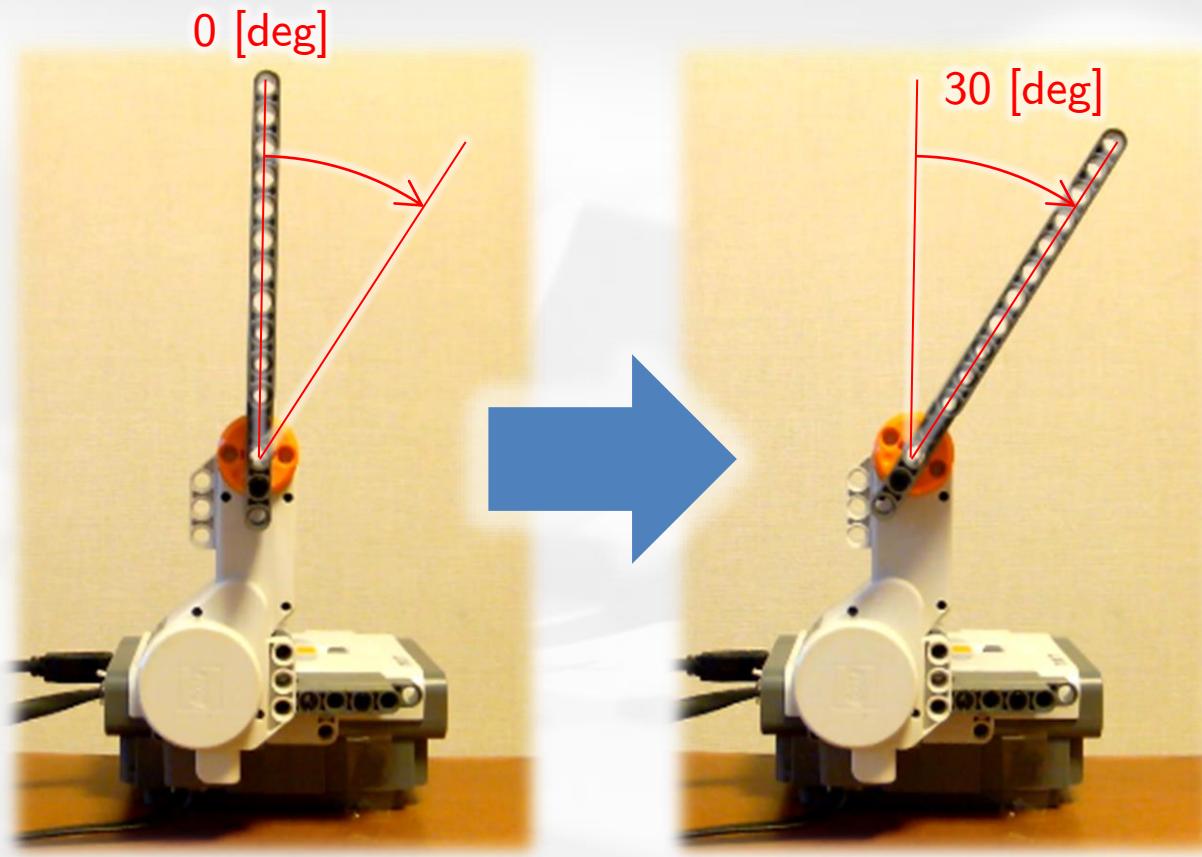


# PID 制御の基礎

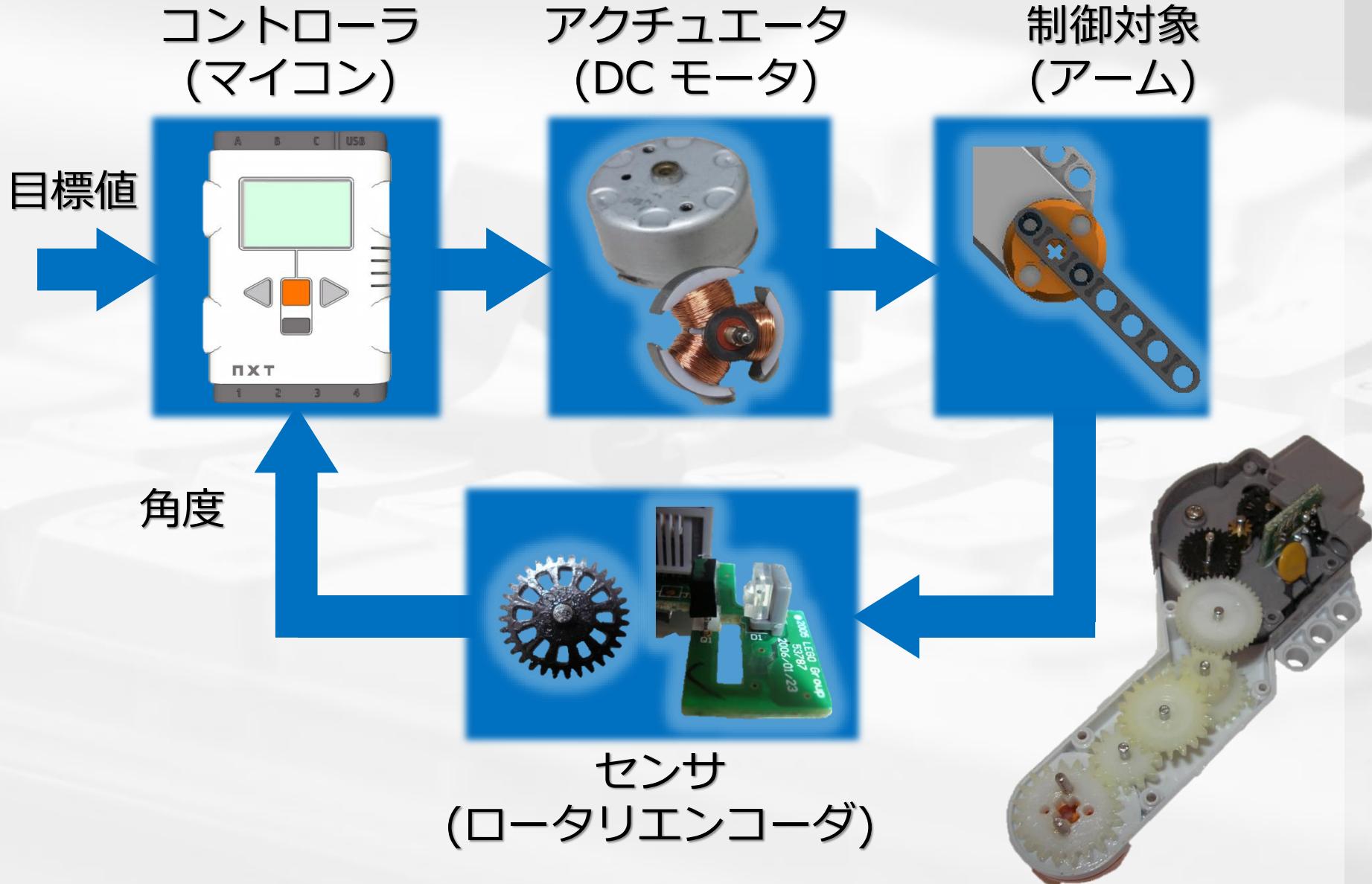
- ON/OFF 制御
- PID 制御
  - P 制御
  - 過渡特性を改善しよう
    - PD 制御と P-D 制御
  - 定常特性を改善しよう
    - PI-D 制御
  - I-PD 制御



# 角度制御実験装置

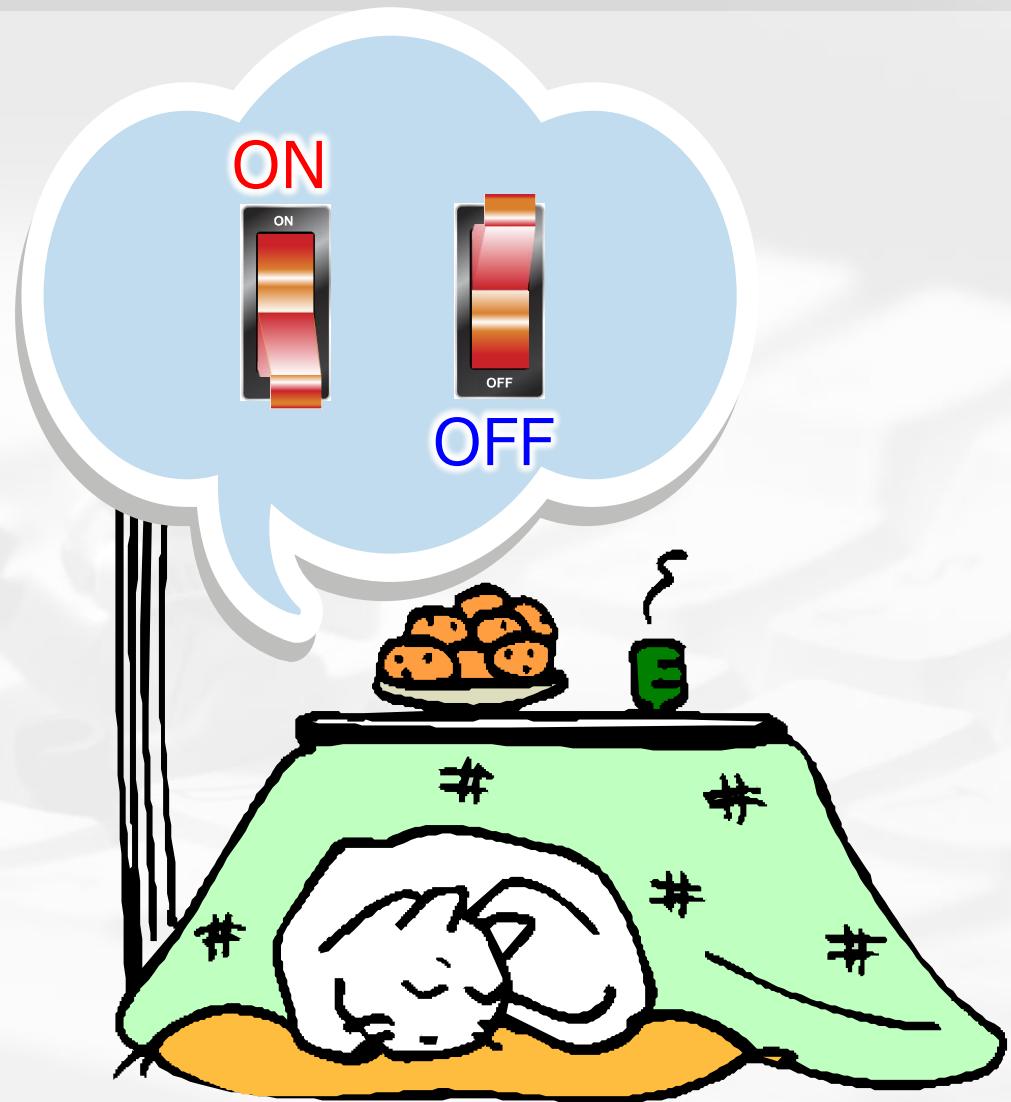


# 角度制御実験装置



# ON/OFF 制御

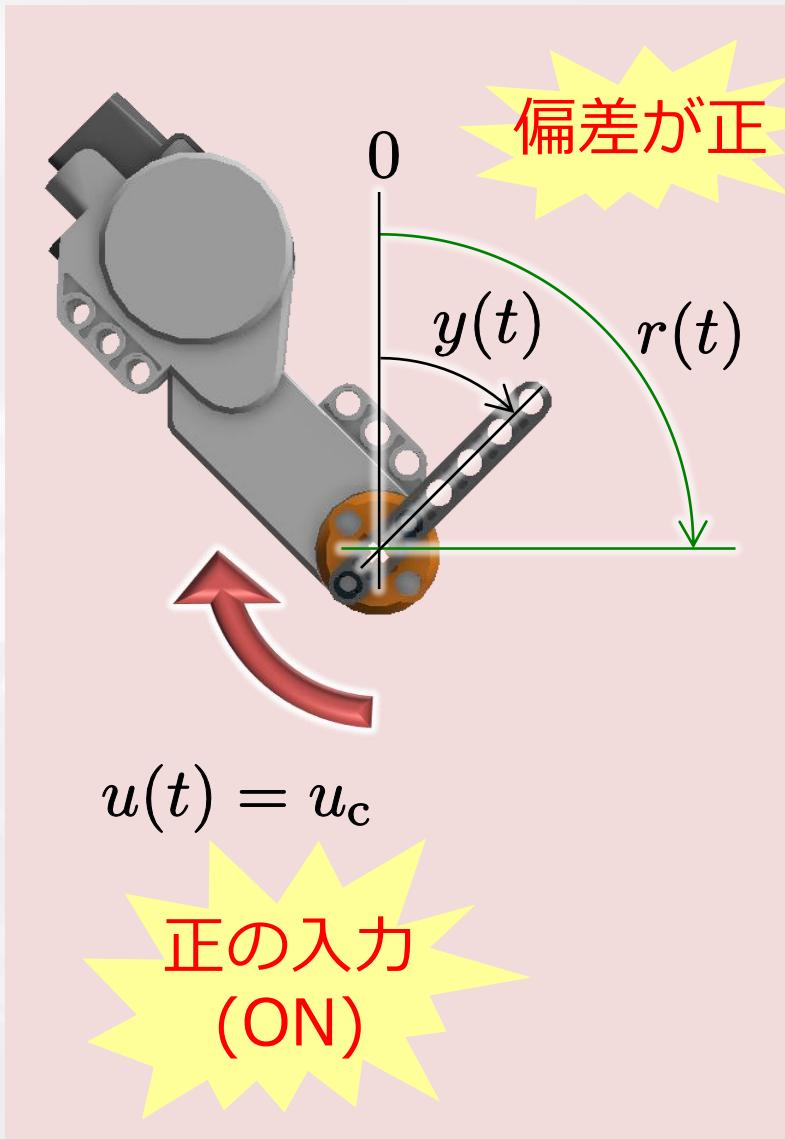
- サーモmostat  
(温度スイッチ) で  
ON/OFF 制御



サーモmostat：「温度とか熱を表す Thermo」と  
「一定にすると言う意味の Stat」の合成語

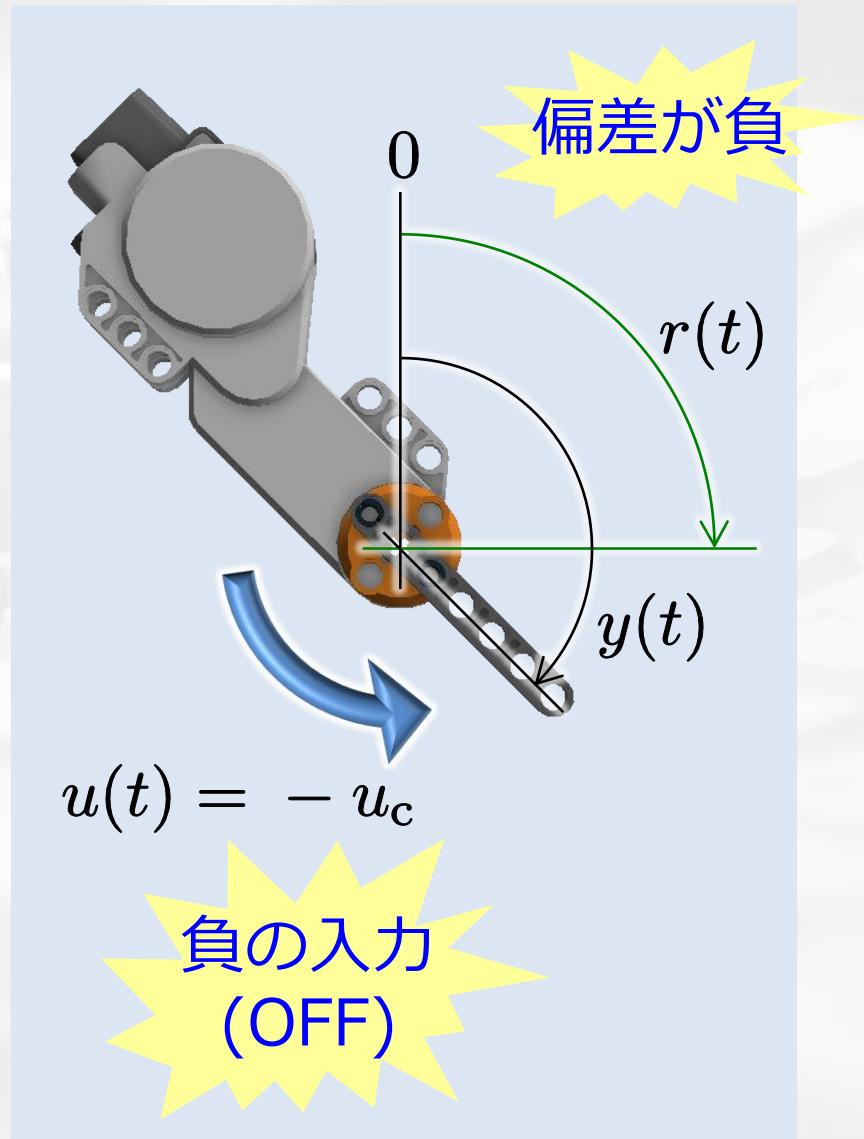
# ON/OFF 制御

$$\text{偏差 } e(t) = r(t) - y(t)$$



$$u(t) = u_c$$

偏差が負



$$u(t) = -u_c$$

# ON/OFF 制御

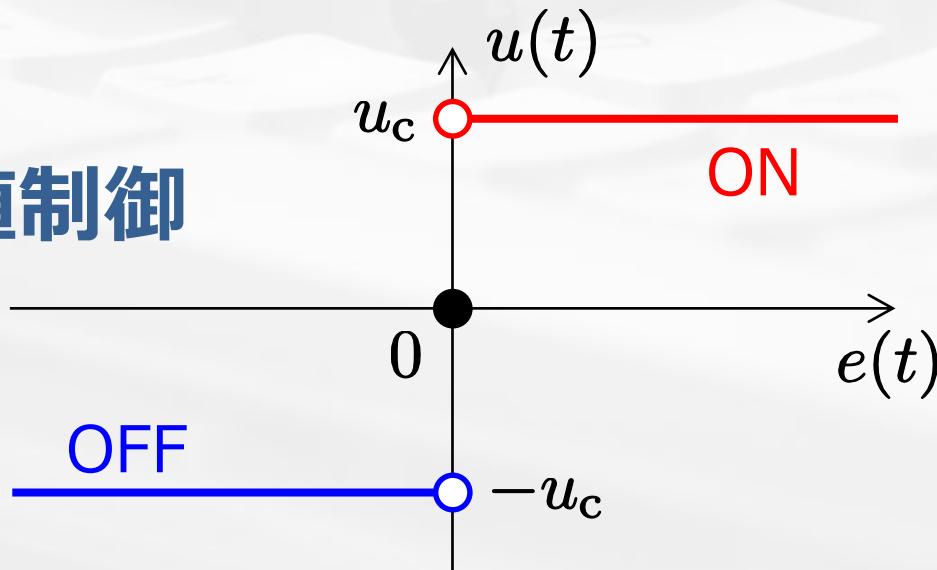
## ON/OFF コントローラ

$$u(t) = u_c \operatorname{sgn} e(t)$$

$$= \begin{cases} u_c & (e(t) > 0) \\ 0 & (e(t) = 0) \\ -u_c & (e(t) < 0) \end{cases} \cdots \cdots \text{ON 動作}$$
$$= \begin{cases} u_c & (e(t) > 0) \\ 0 & (e(t) = 0) \\ -u_c & (e(t) < 0) \end{cases} \cdots \cdots \text{OFF 動作}$$

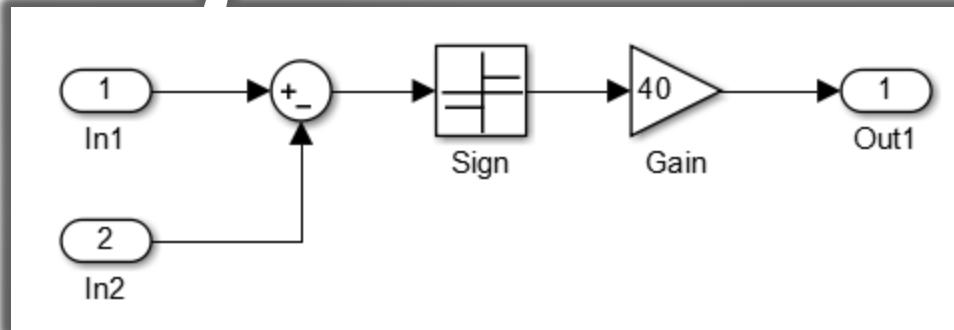
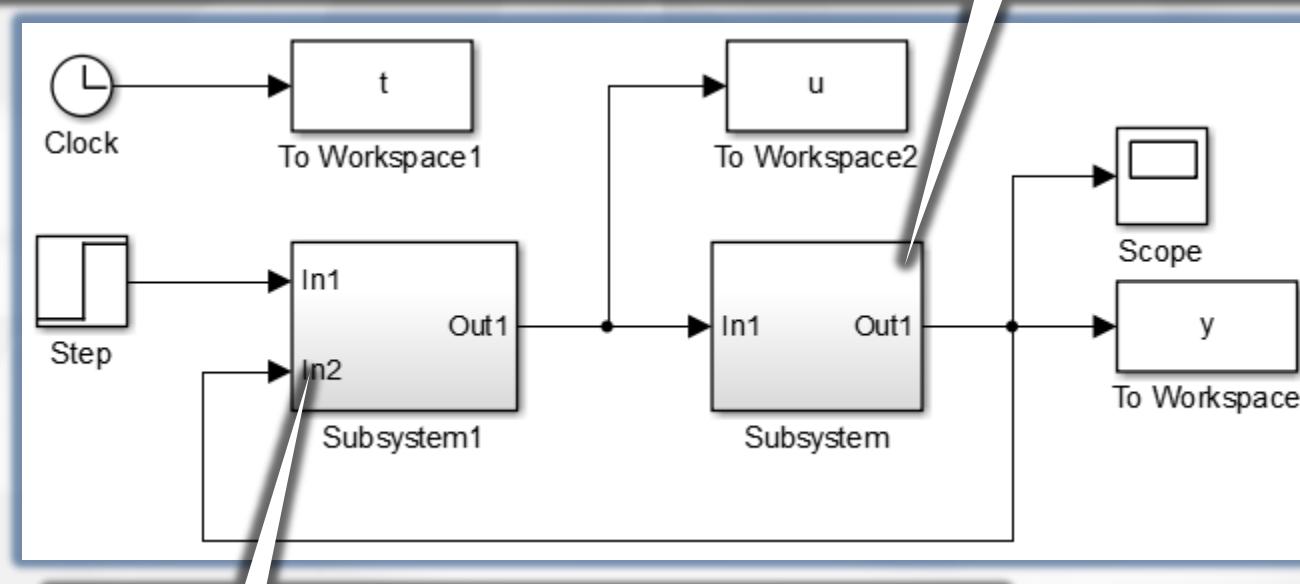
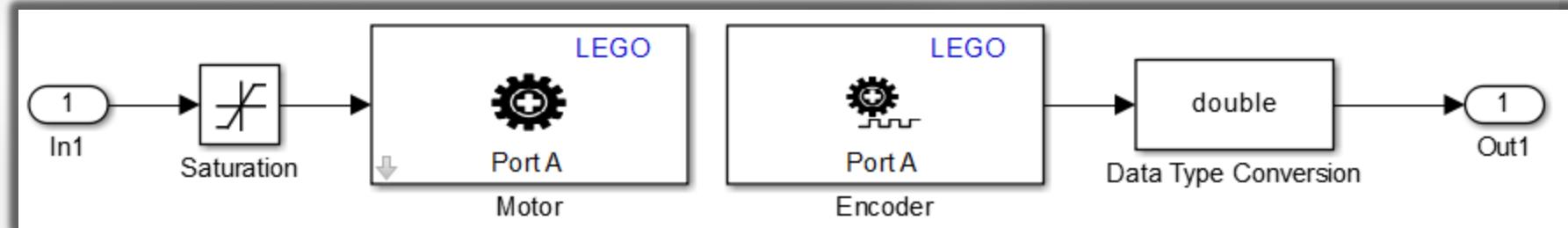
## 2 値制御

$$\operatorname{sgn} x = \begin{cases} 1 & (x > 0) \\ 0 & (x = 0) \\ -1 & (x < 0) \end{cases}$$



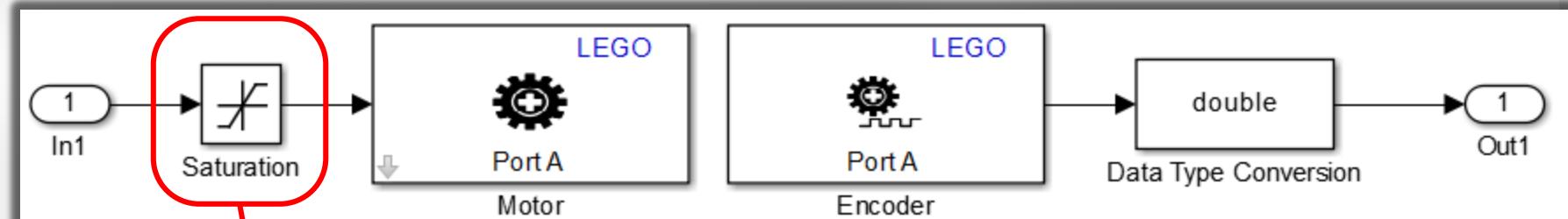
# ON/OFF 制御

>> h = 0.01;



on\_off\_c.slx

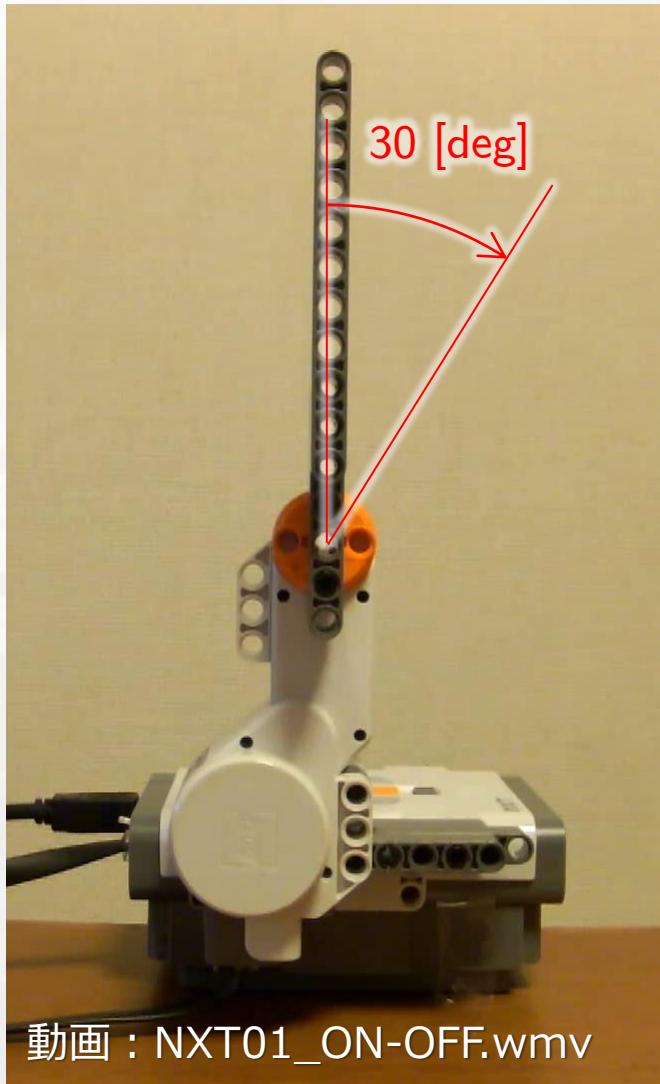
# ON/OFF 制御



入力制限

$$u(t) = \begin{cases} 100 & (u(t) > 100) \\ u(t) & (|u(t)| \leq 100) \\ -100 & (u(t) < -100) \end{cases}$$

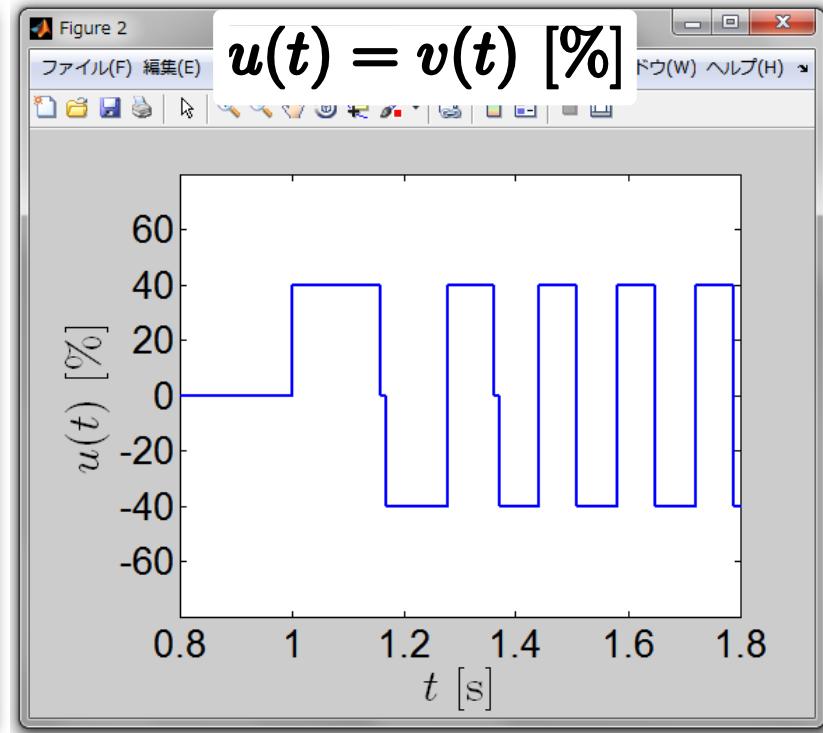
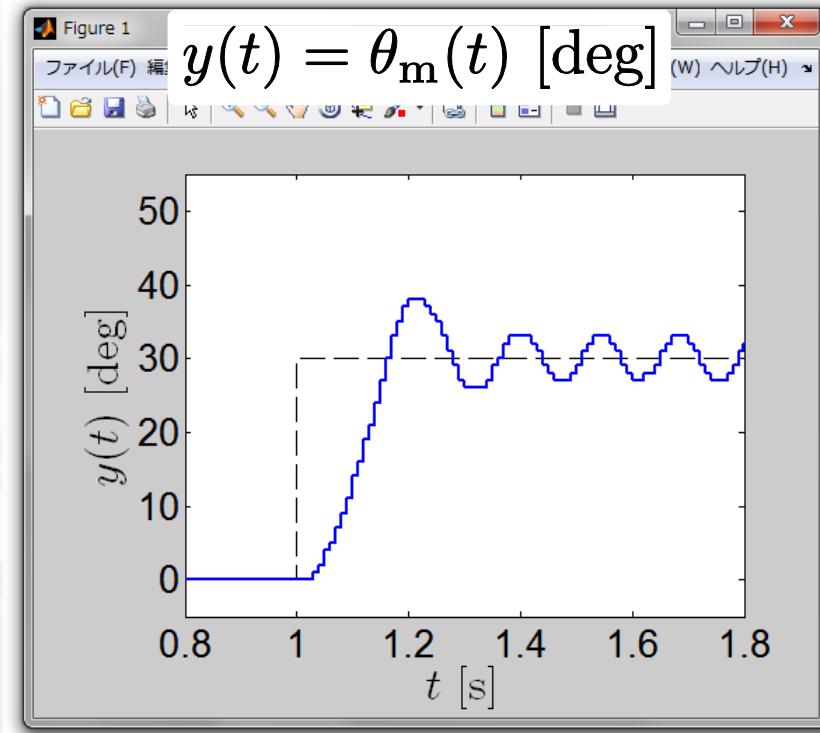
# ON/OFF 制御



ハンチング (脈動) を生じる

# ON/OFF 制御

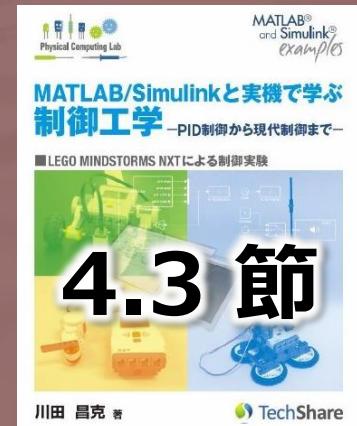
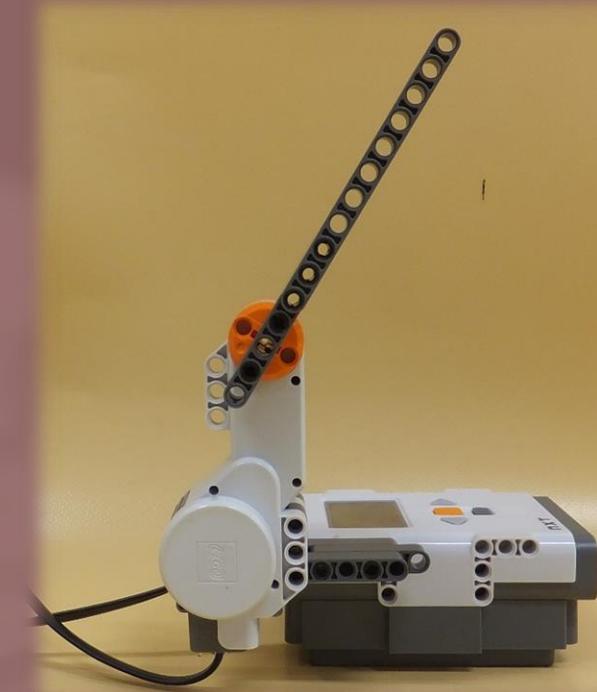
```
>> figure(1); stairs(t,y)  
>> figure(2); stairs(t,u)
```



ハンチング (脈動) を生じる

# PID 制御の基礎

- ON/OFF 制御
- PID 制御
  - P 制御
  - 過渡特性を改善しよう
    - PD 制御と P-D 制御
  - 定常特性を改善しよう
    - PI-D 制御
  - I-PD 制御



# P 制御

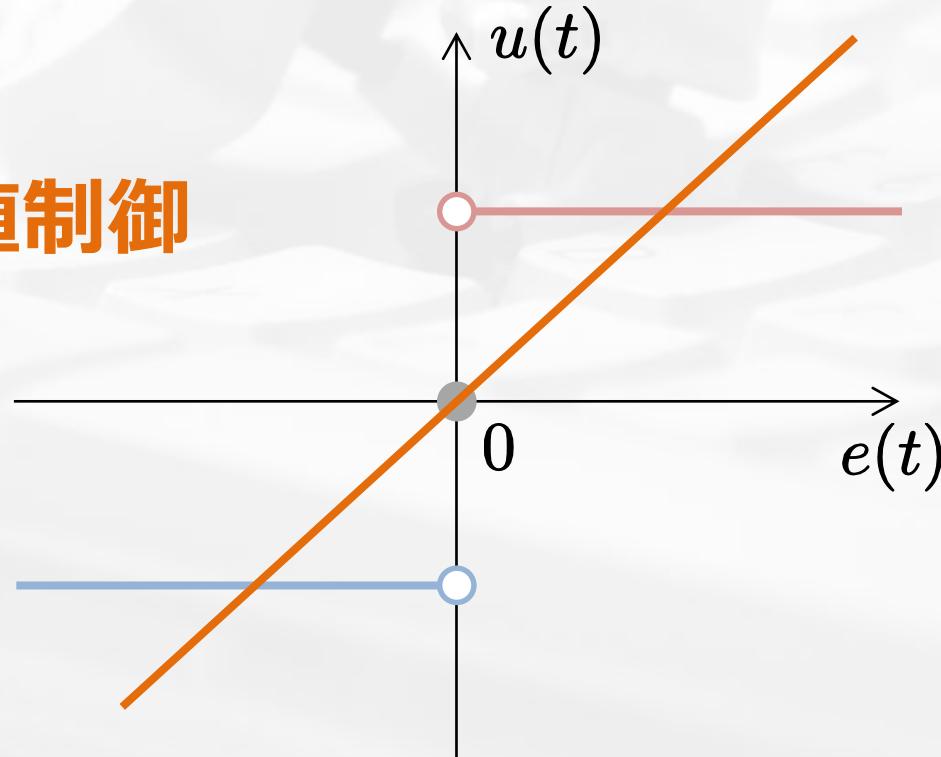
Proportional: 比例

## P コントローラ

$$u(t) = k_P e(t)$$

偏差  $e(t)$  の大きさに応じて  
連続的に変化 (多值制御)

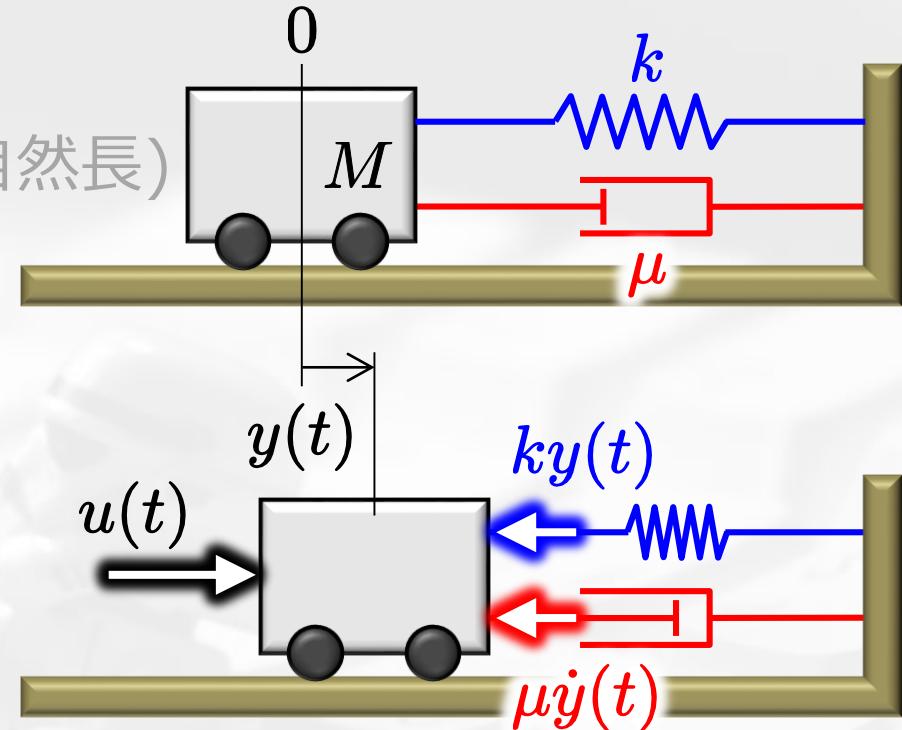
多值制御



# P 制御

マス・ばね・ダンパ系

(自然長)



ニュートンの運動方程式

$$F(t) = Ma(t)$$

$$\overbrace{u(t) - ky(t) - \mu\dot{y}(t)} = \overbrace{M\ddot{y}(t)}$$

$$\rightarrow M\ddot{y}(t) + \mu\dot{y}(t) + ky(t) = u(t)$$

# P 制御

マス・ばね・ダンパ系

$$M\ddot{y}(t) + \mu\dot{y}(t) + ky(t) = u(t)$$

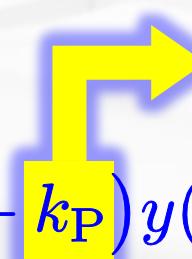


P コントローラ

$$u(t) = k_P(r(t) - y(t))$$



$$M\ddot{y}(t) + \mu\dot{y}(t) + (k + k_P)y(t) = k_P r(t)$$



人為的に “ばね”  
を強くする

# P 制御

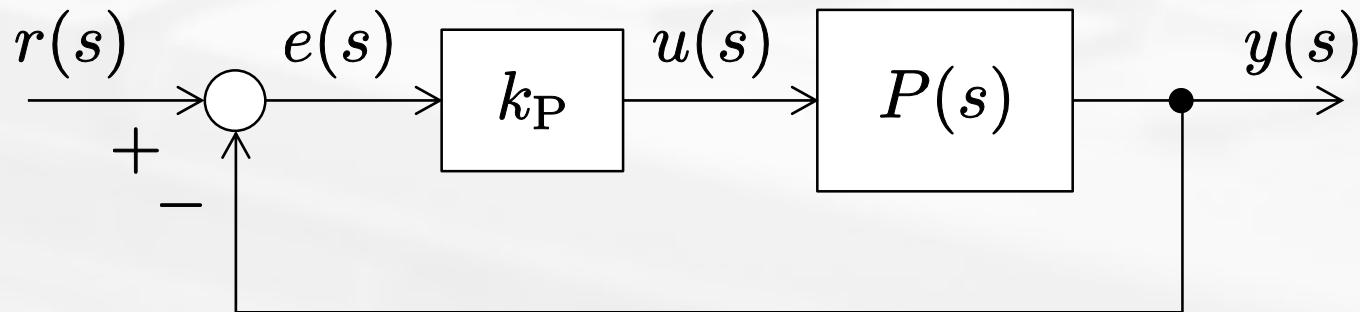
## P コントローラ

$$u(t) = k_P e(t) \quad \longleftrightarrow \quad u(s) = k_P e(s)$$

## 制御対象 (アーム駆動系)

$$\ddot{y}(t) + a\dot{y}(t) = bu(t)$$

$$\longleftrightarrow \quad y(s) = P(s)u(s), \quad P(s) = \frac{b}{s(s+a)}$$



# P 制御

## P コントローラ

$$u(t) = k_P e(t)$$

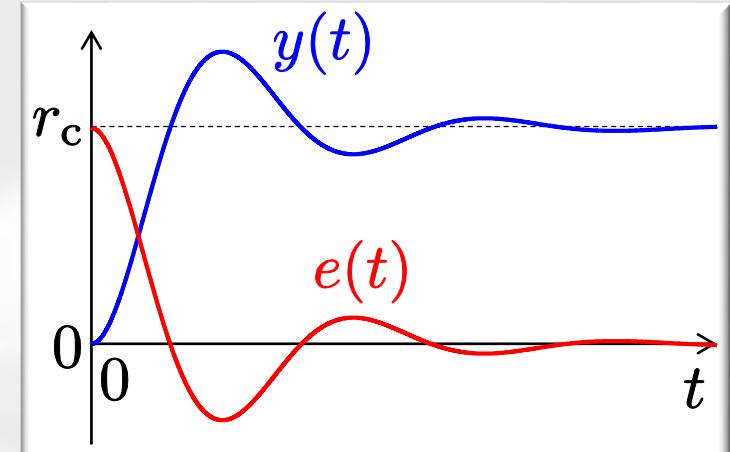
## ゲインの大きさ

$$\begin{cases} |u(t)| \leq 100 & \cdots \text{入力の制限値} \\ r(t) = r_c \ (t \geq 0) & \cdots \text{目標値} \end{cases}$$

$$\rightarrow k_P \leq \frac{100}{\max |e(t)|} = \frac{100}{r_c}$$

$\rightarrow r_c = 90 \ [\text{deg}]$  のとき

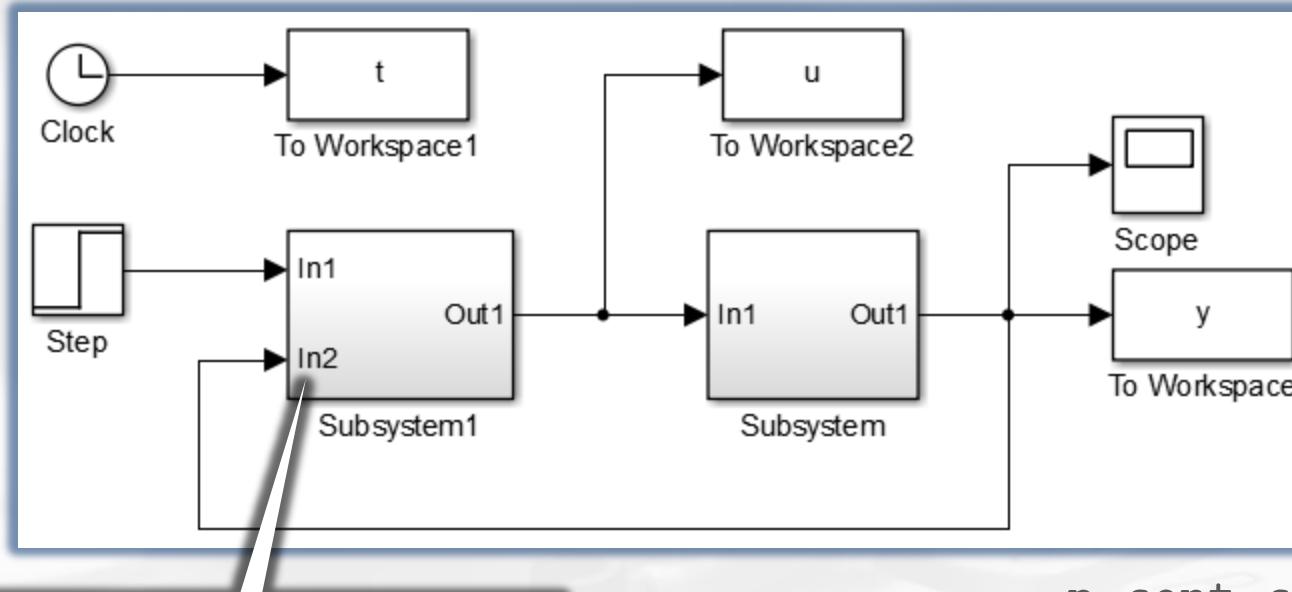
$$k_P \leq \frac{100}{90} = 1.11\dots$$



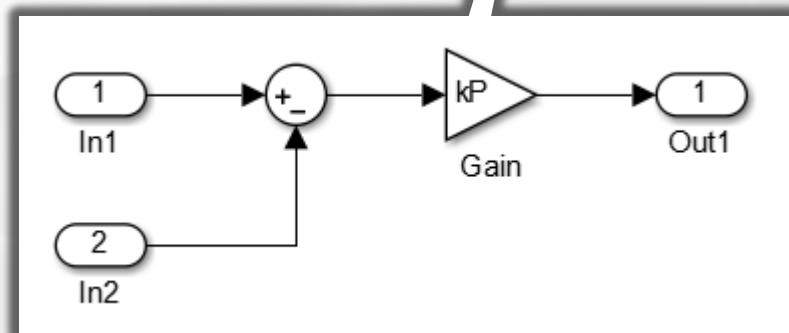
$r_c = 30 \ [\text{deg}]$  のとき

$$k_P \leq \frac{100}{30} = 3.33\dots$$

# P 制御



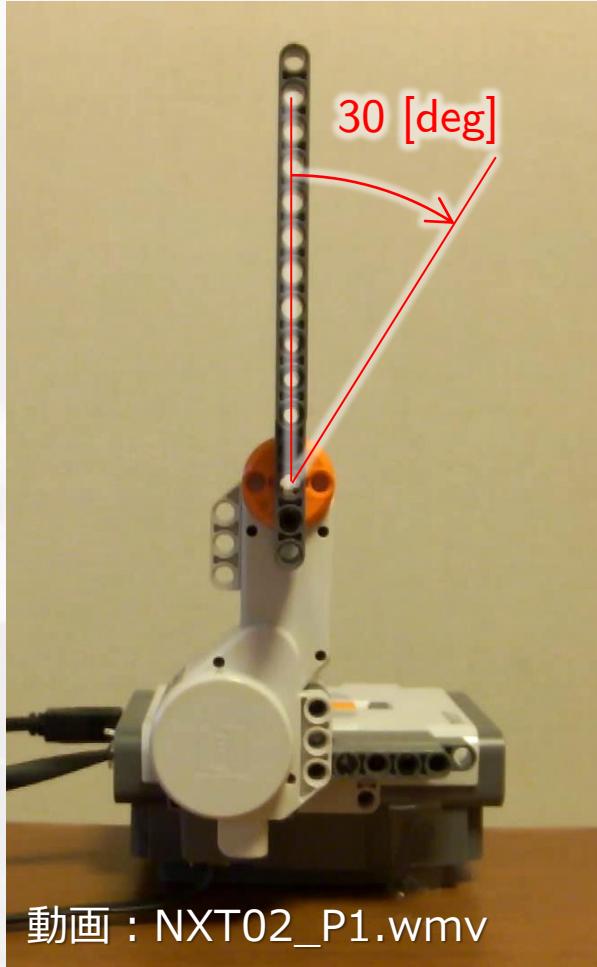
p\_cont\_c.slx



```
>> h = 0.01;  
>> kP = 1;
```

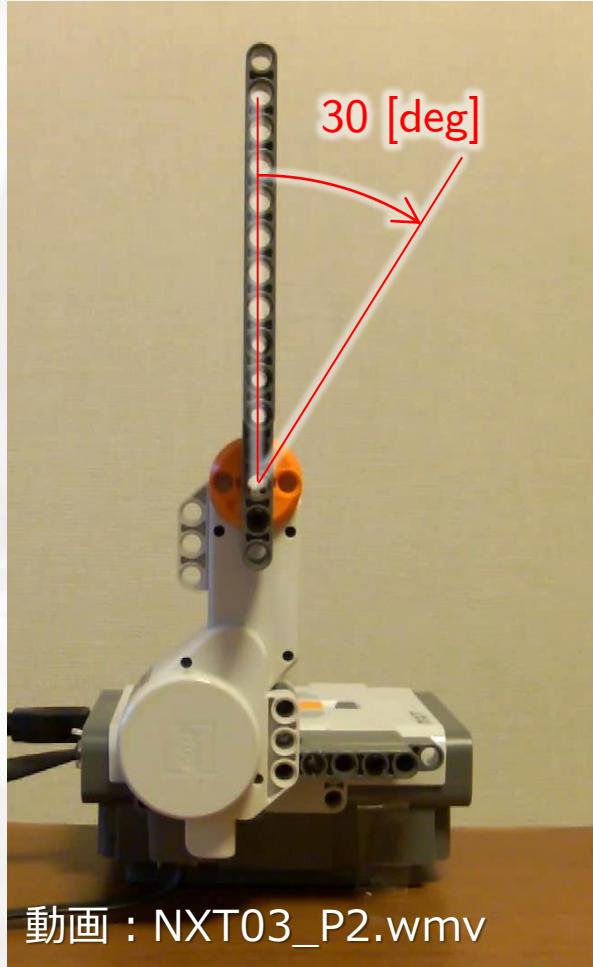
# P 制御

$$k_P = 1$$



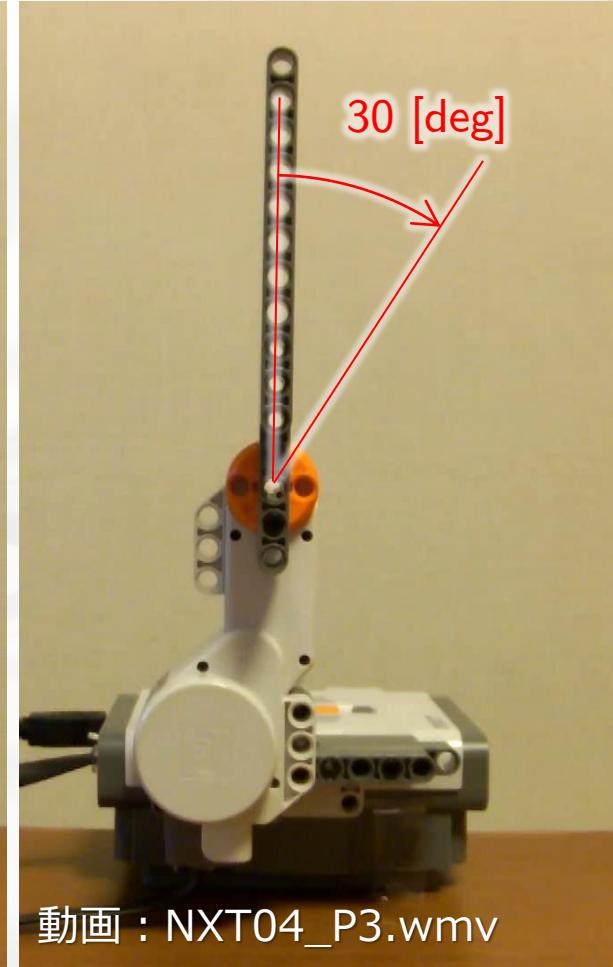
動画：NXT02\_P1.wmv

$$k_P = 2$$



動画：NXT03\_P2.wmv

$$k_P = 3$$

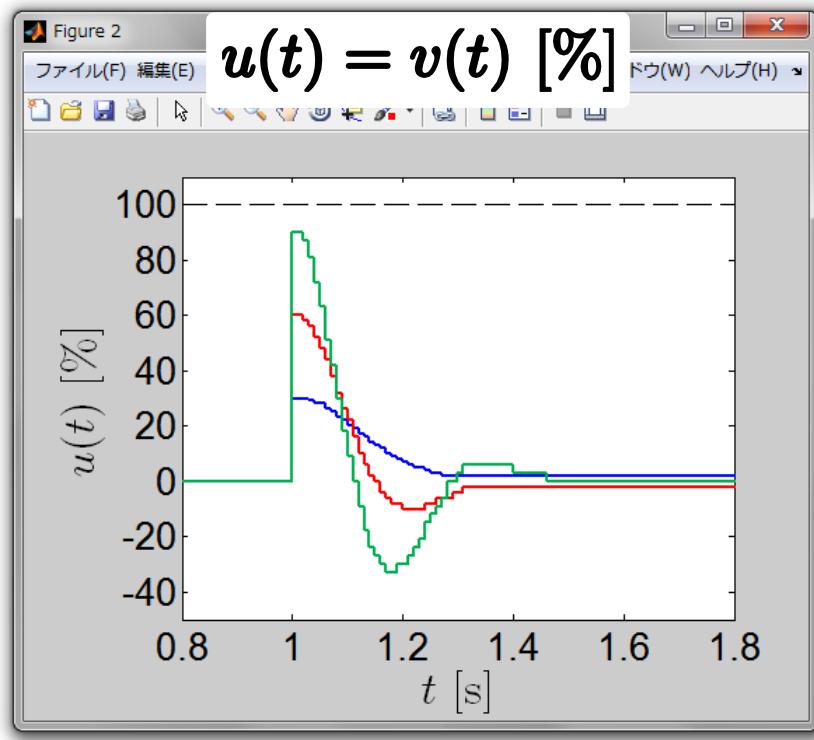
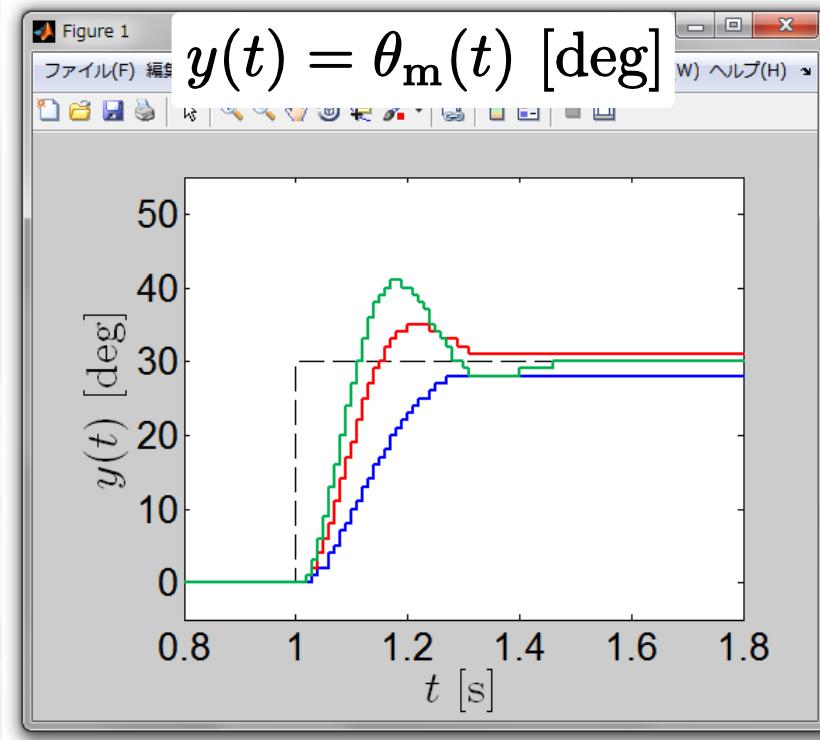


動画：NXT04\_P3.wmv

反応が速くなる／オーバーシュートが大きくなる

# P 制御

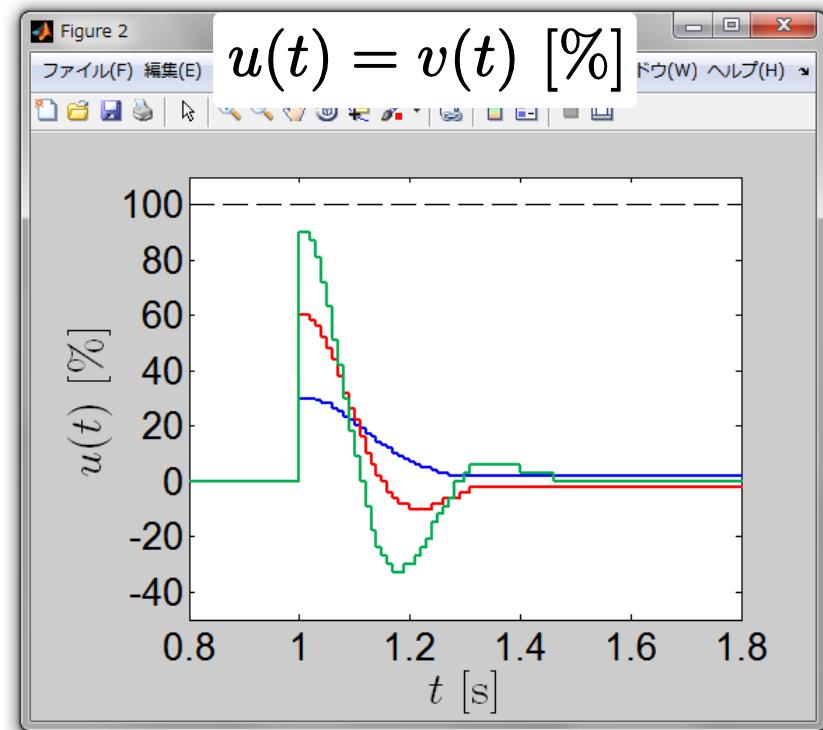
—  $k_P = 1$  —  $k_P = 2$  —  $k_P = 3$



“ $k_P \rightarrow \text{大}$ ” とすると .....

# P 制御

—  $k_P = 1$  —  $k_P = 2$  —  $k_P = 3$

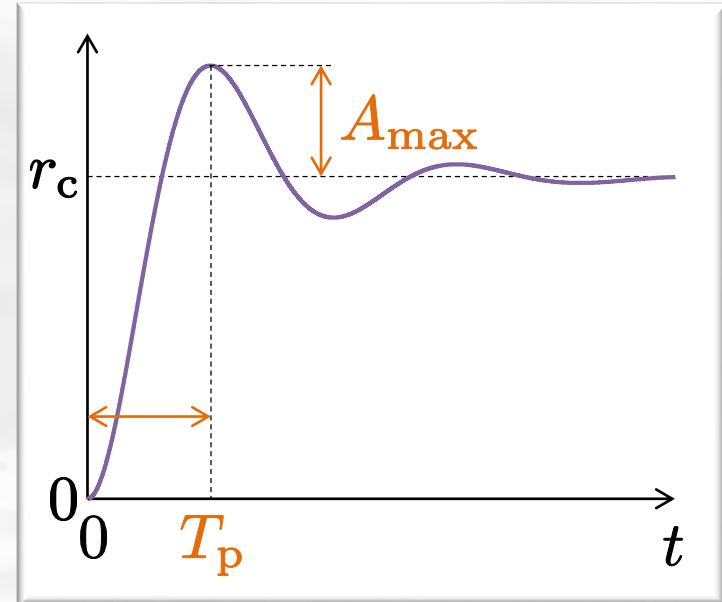
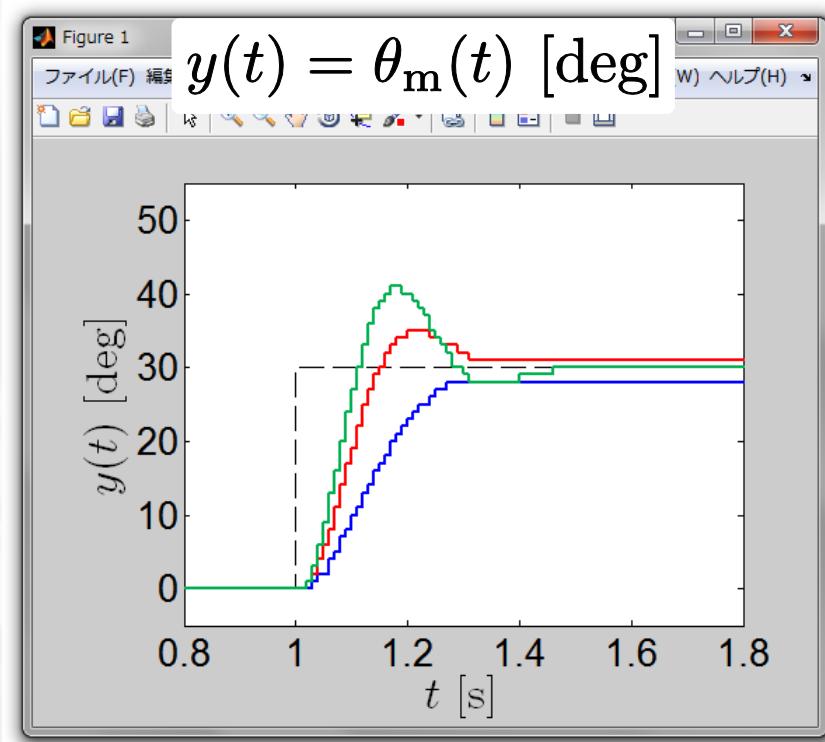


“ $k_P \rightarrow \text{大}$ ” とすると、入力は .....

- 入力  $u(t) \rightarrow \text{大}$

# P 制御

—  $k_P = 1$  —  $k_P = 2$  —  $k_P = 3$

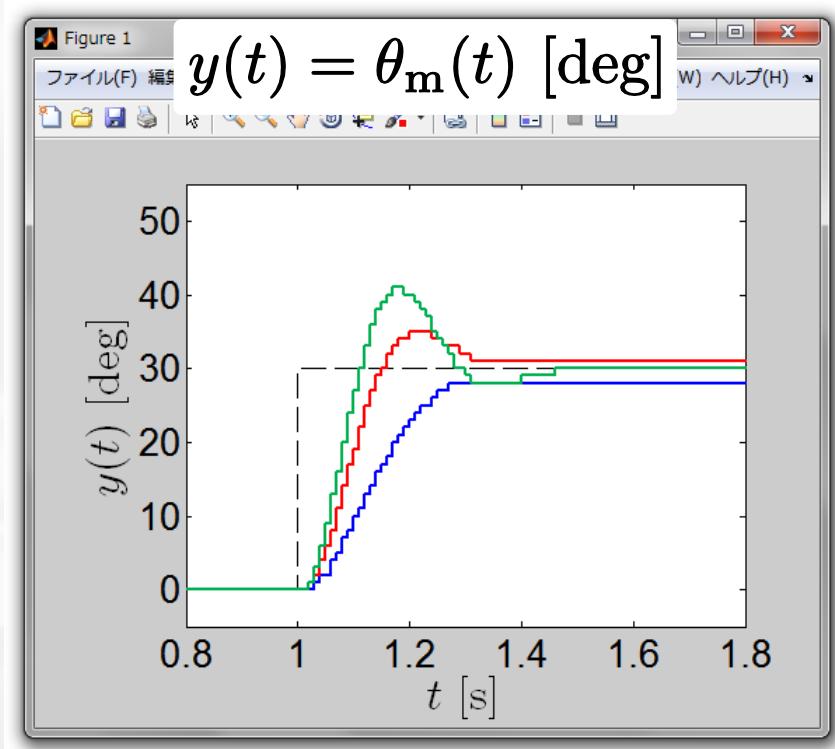


“ $k_P \rightarrow 大$ ” とすると、過渡特性は .....

- 立ち上がりの速さ  $T_p \rightarrow 小$
- オーバーシュート  $A_{max} \rightarrow 大$

# P 制御

—  $k_P = 1$  —  $k_P = 2$  —  $k_P = 3$



“ $k_P \rightarrow \text{大}$ ” とすると, 定常特性は .....

- 定常偏差  $|e(t)| \rightarrow \text{小}$

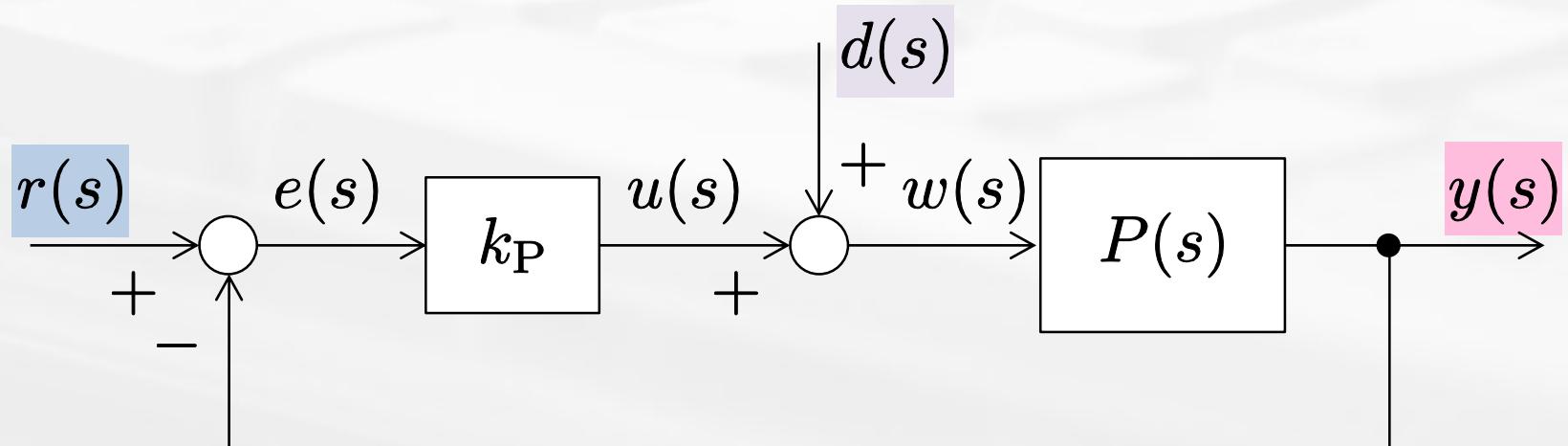
# 理論的な解析

# P 制御

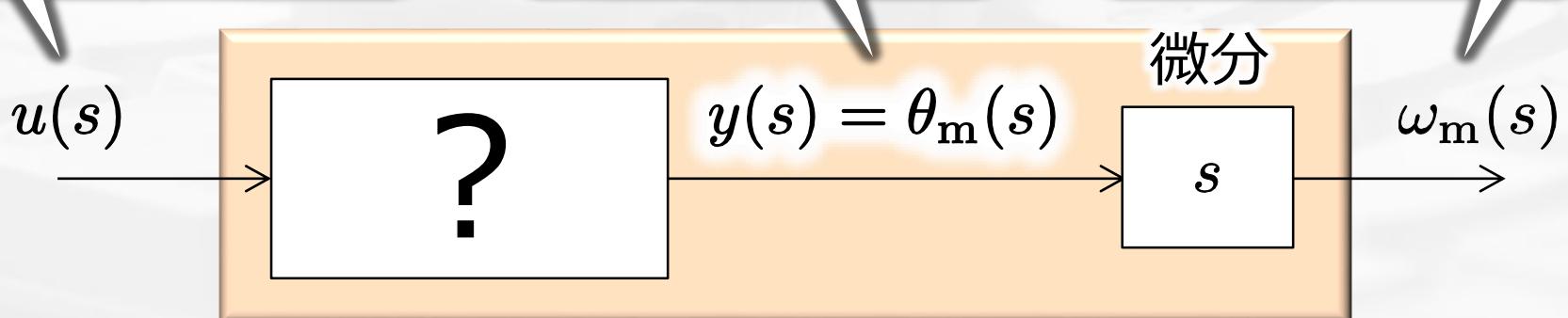
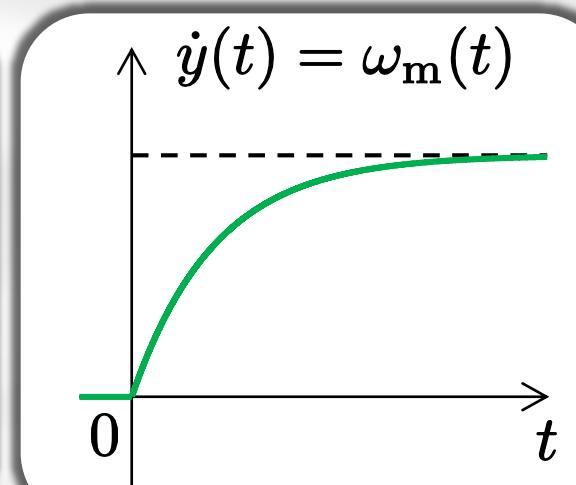
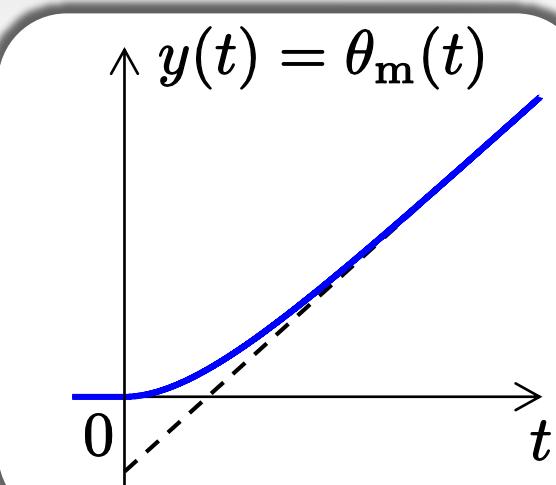
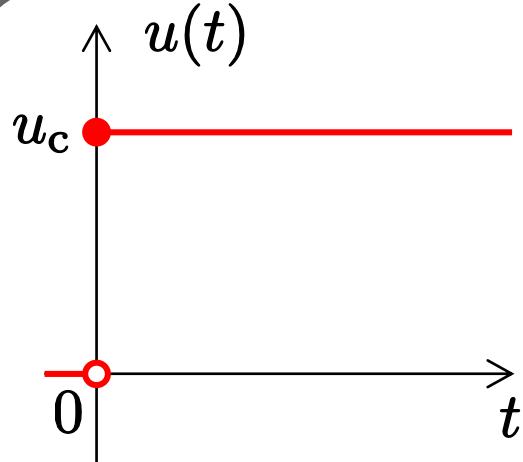
$$\begin{aligned}y(s) &= P(s)w(s) \\&= P(s)(u(s) + d(s)) \\&= P(s)(k_P e(s) + d(s)) \\&= P(s)\{k_P(r(s) - y(s)) + d(s)\}\end{aligned}$$

$$\rightarrow (1 + P(s)k_P)y(s) = P(s)k_P r(s) + P(s)d(s)$$

$$\rightarrow y(s) = \frac{P(s)k_P}{1 + P(s)k_P}r(s) + \frac{P(s)}{1 + P(s)k_P}d(s)$$

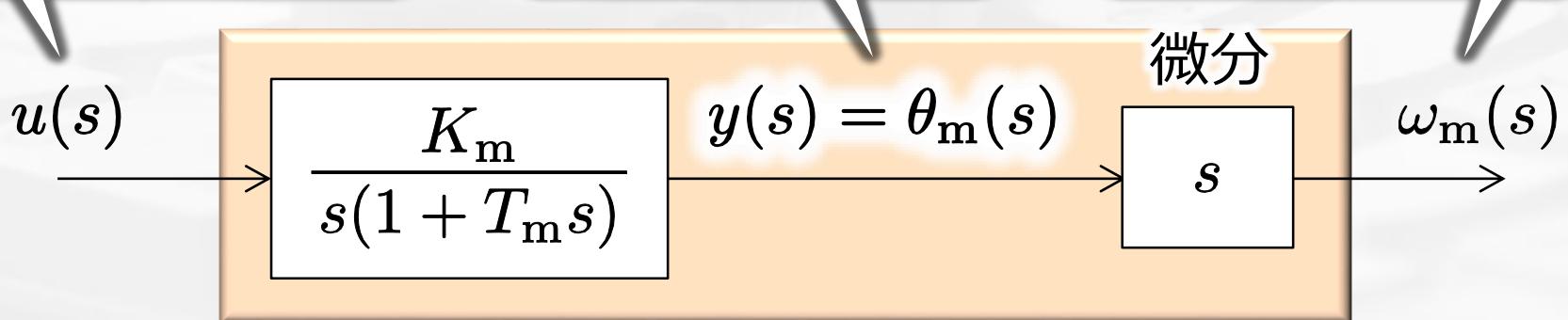
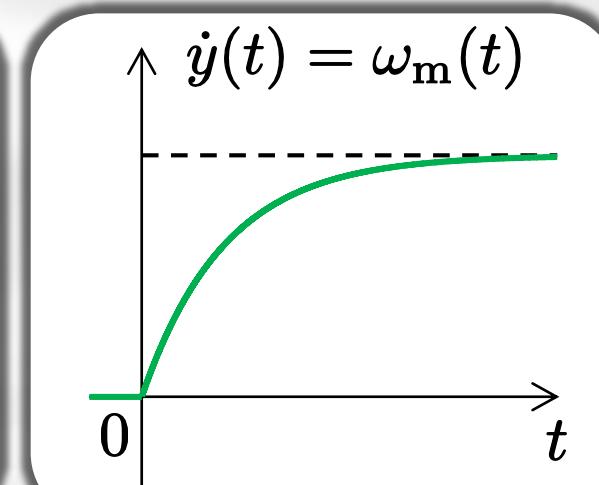
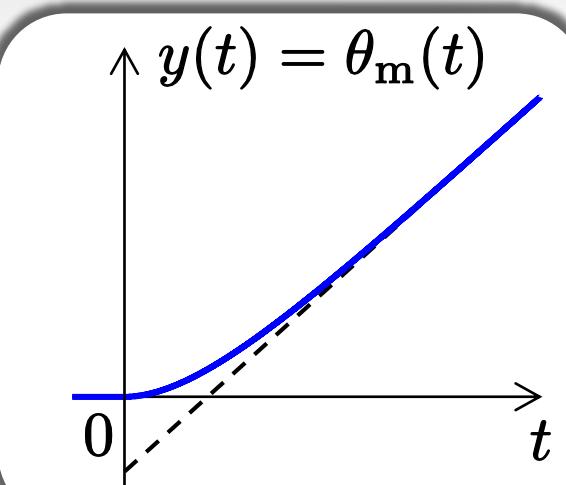
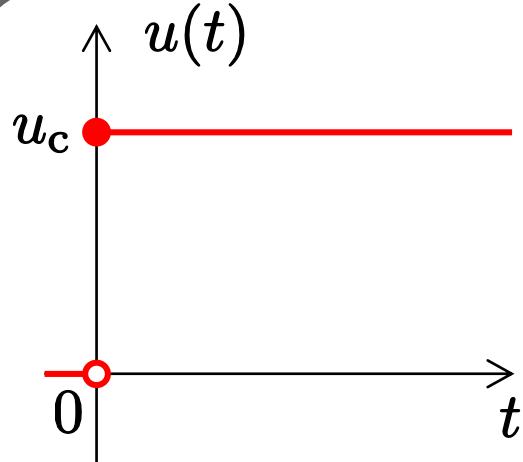


# モータ駆動系のモデル



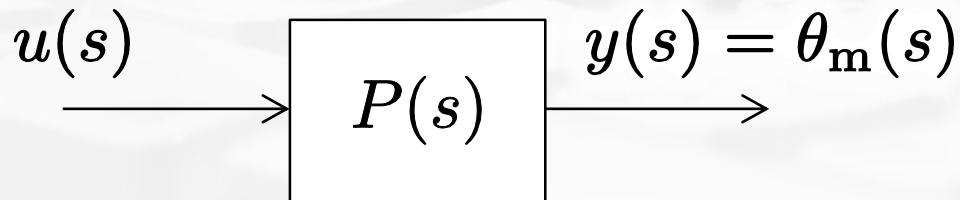
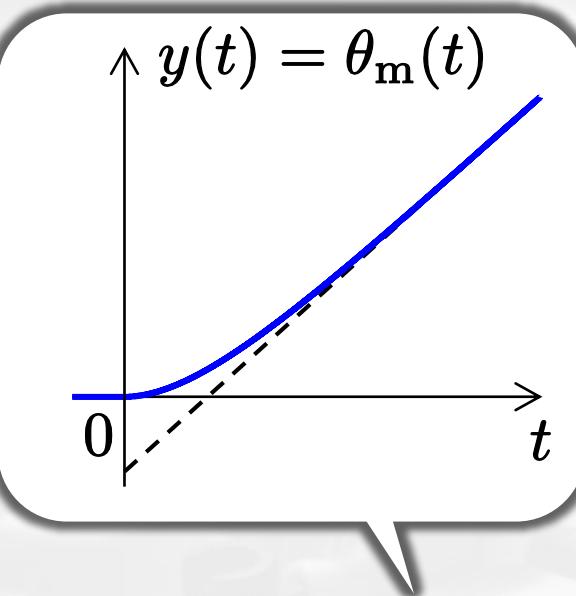
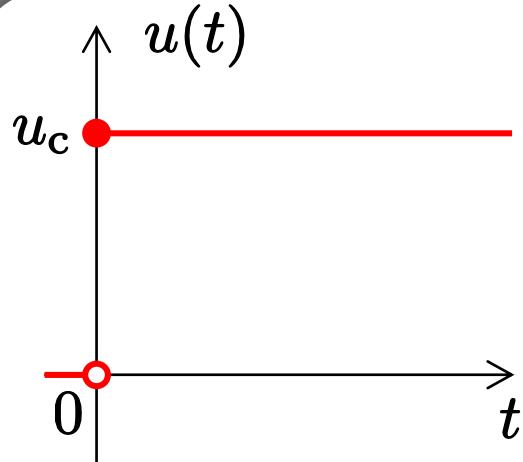
$$\text{1 次遅れ要素 } \frac{K_m}{1 + T_m s}$$

# モータ駆動系のモデル



1 次遅れ要素  $\frac{K_m}{1 + T_m s}$

# モータ駆動系のモデル



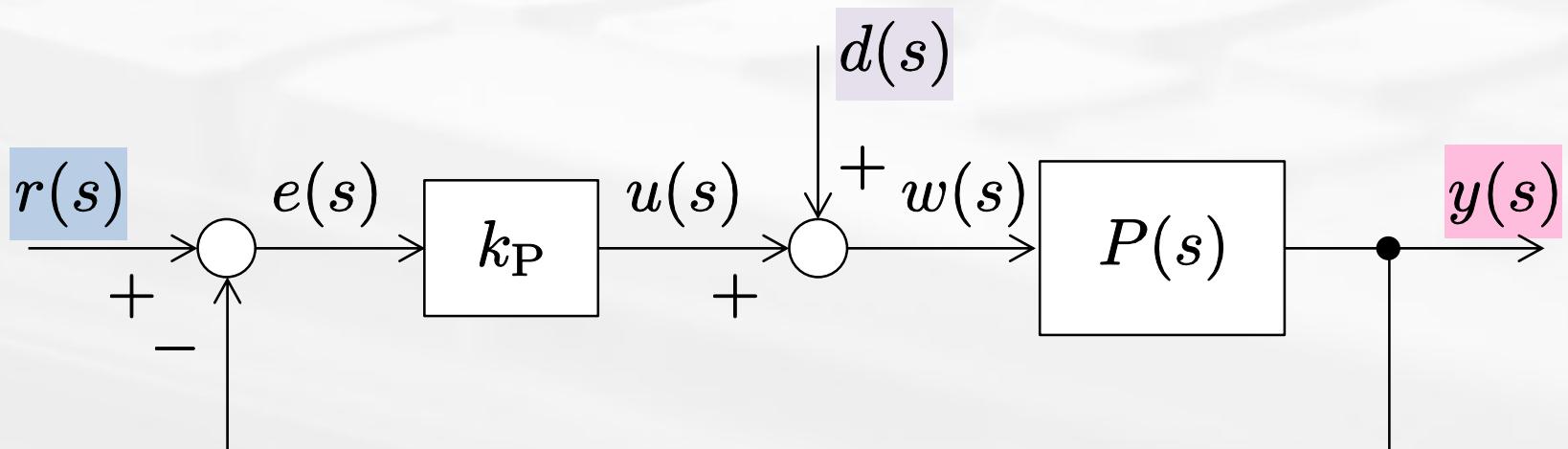
$$P(s) = \frac{K_m}{s(1 + T_m s)} = \frac{b}{s(s + a)}, \quad \begin{cases} a = 1/T_m \\ b = K_m/T_m \end{cases}$$

# P 制御

$$P(s) = \frac{b}{s(s+a)}$$

$$\begin{aligned}
 y(s) &= \frac{P(s)k_P}{1+P(s)k_P} r(s) + \frac{P(s)}{1+P(s)k_P} d(s) \\
 &= \frac{bk_P}{s^2 + as + bk_P} r(s) + \frac{b}{s^2 + as + bk_P} d(s) \\
 &= \underbrace{\frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}}_{G_{yr}(s)} r(s) + \underbrace{\frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}}_{G_{yd}(s)} d(s)
 \end{aligned}$$

2 次遅れ  
要素



## 2 次遅れ要素

$$\left\{ \begin{array}{l} G_{yr}(s) = \frac{b k_P}{s^2 + as + b k_P} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \\ G_{yd}(s) = \frac{b}{s^2 + as + b k_P} = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \end{array} \right.$$

$$\left\{ \begin{array}{l} \text{固有角周波数} \quad \omega_n = \sqrt{b k_P} \\ \text{減衰係数} \quad \zeta = \frac{a}{2\omega_n} = \frac{a}{2\sqrt{b k_P}} \\ \text{ゲイン} \quad K = \frac{1}{k_P} \end{array} \right.$$

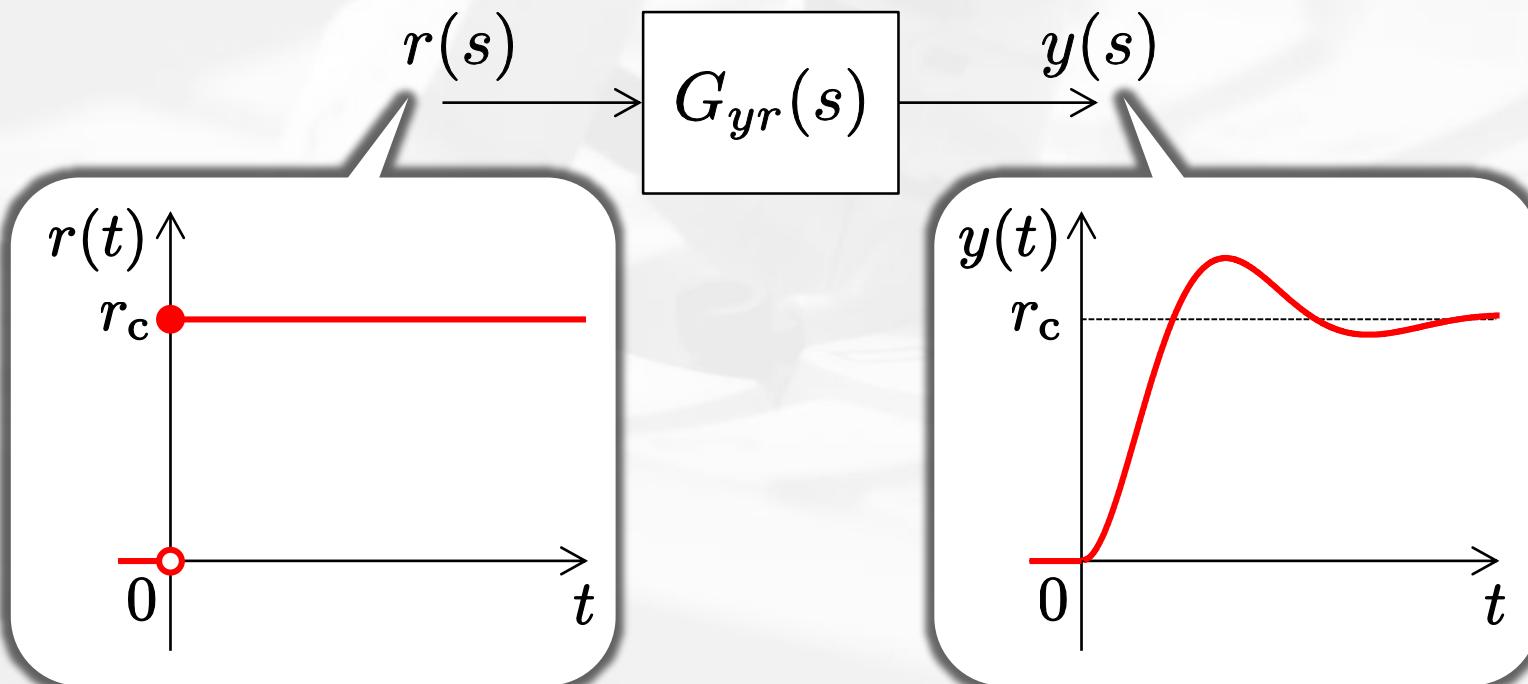
→  $s^2 + 2\zeta\omega_n s + \omega_n^2 = 0$  の解

→ 極  $s = (-\zeta \pm \sqrt{\zeta^2 - 1})\omega_n$

# P 制御

$$d(t) = 0$$

$$y(s) = G_{yr}(s)r(s), \quad G_{yr}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$



# ラプラス変換：最終値の定理

## 最終値の定理

$$y_\infty = \lim_{t \rightarrow \infty} y(t) = \lim_{s \rightarrow 0} s y(s)$$

$r(t) = r_c$  ( $t \geq 0$ ) のとき

$$r(s) = \mathcal{L}[r(t)] = r_c \mathcal{L}[1] = r_c \times \frac{1}{s}$$

$$\rightarrow y(s) = G_{yr}(s)r(s) = G_{yr}(s) \frac{r_c}{s}$$

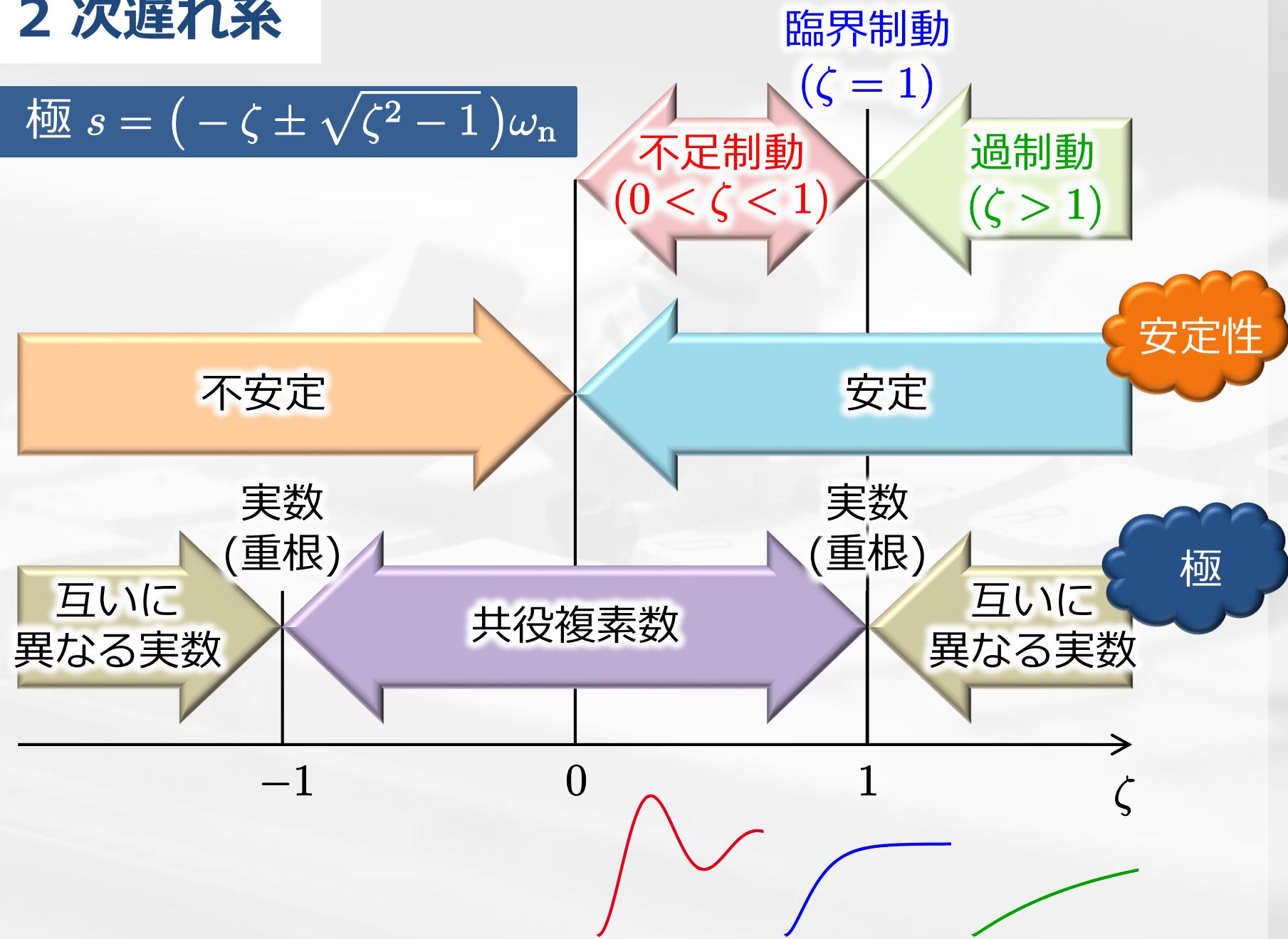
$$\begin{aligned}\rightarrow y_\infty &= \lim_{s \rightarrow 0} s y(s) = \lim_{s \rightarrow 0} G_{yr}(s)r_c = G_{yr}(0)r_c \\ &= r_c\end{aligned}$$

外乱が加わらなければ  
定常偏差は 0

$$G_{yr}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

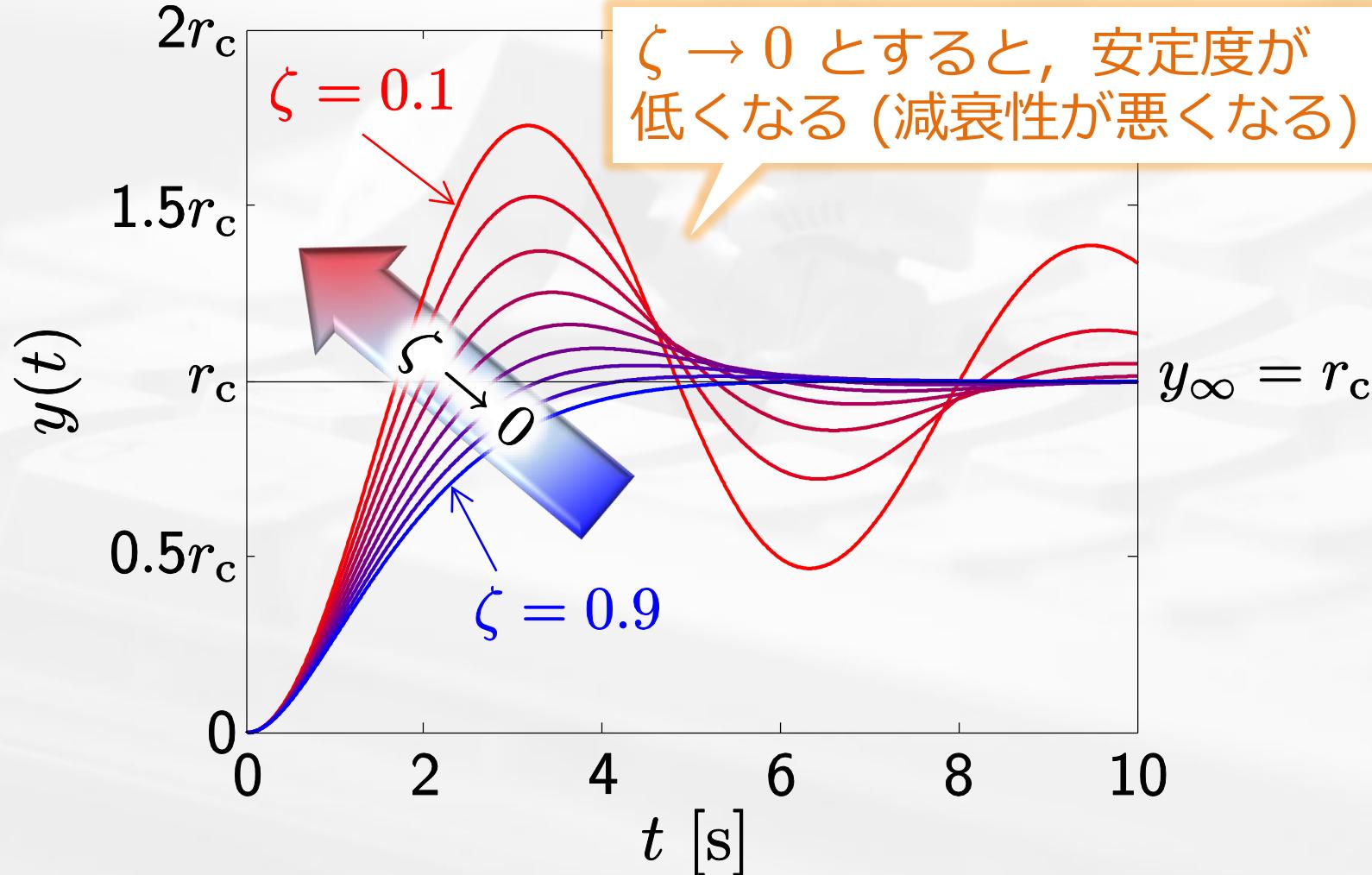
# 2次遅れ系

$$\text{極 } s = (-\zeta \pm \sqrt{\zeta^2 - 1})\omega_n$$



# 2次遅れ系のステップ応答

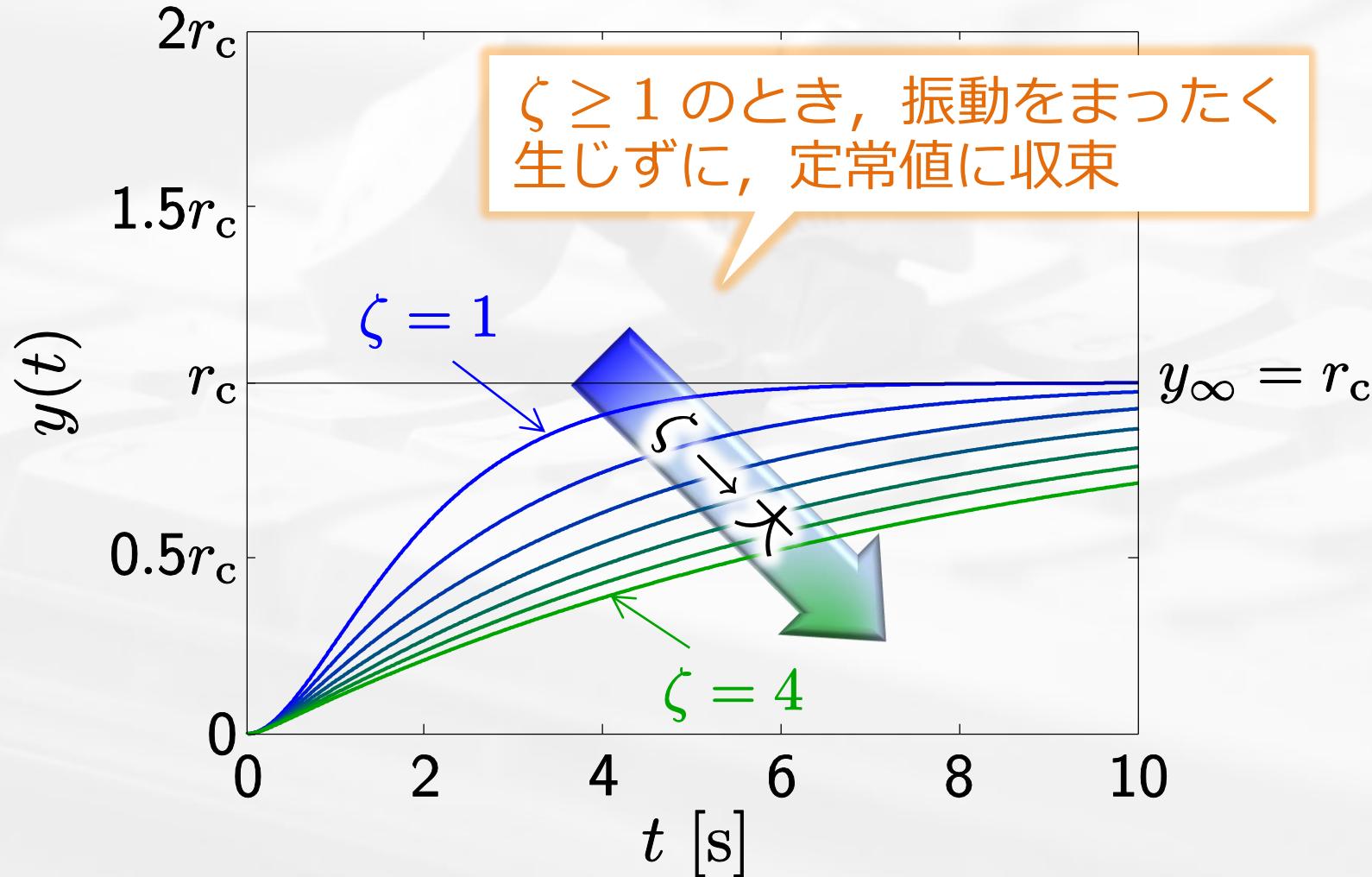
不足制動  $0 < \zeta < 1$



# 2次遅れ系のステップ応答

臨界制動  $\zeta = 1$

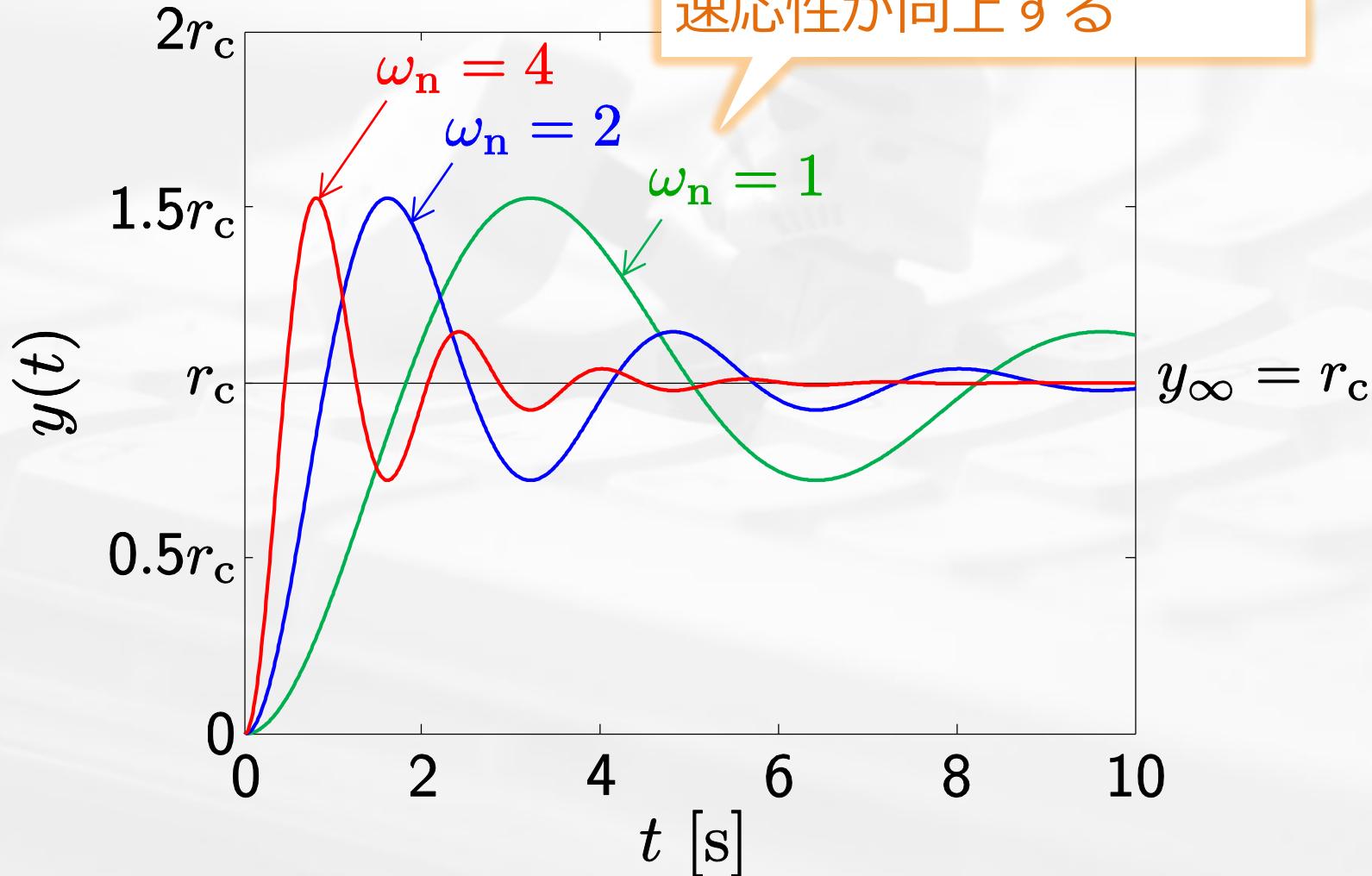
過制動  $\zeta > 1$



## 2次遅れ系のステップ応答

不足制動  $0 < \zeta < 1$

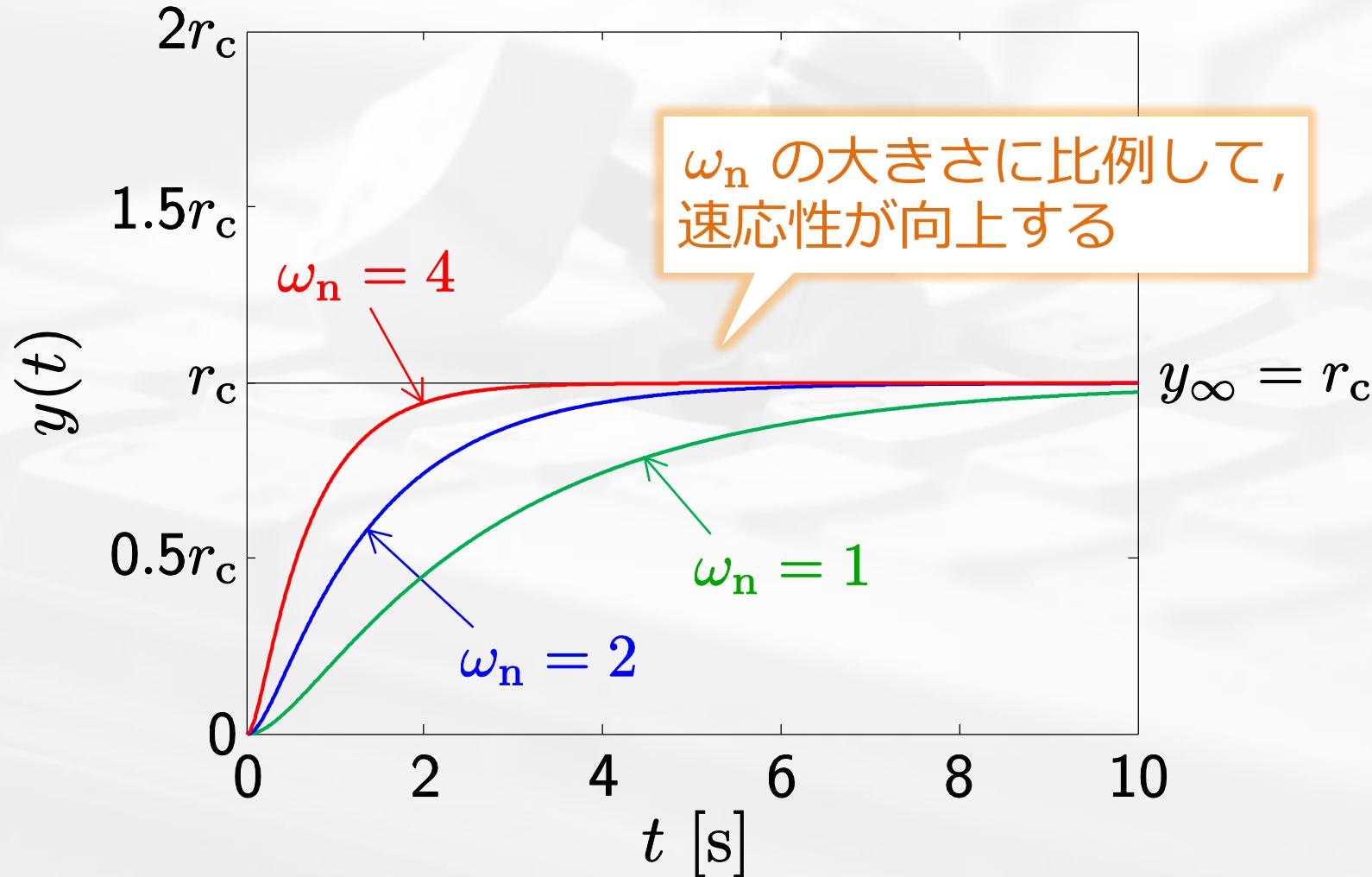
$\omega_n$  の大きさに比例して、  
速応性が向上する



# 2次遅れ系のステップ応答

臨界制動  $\zeta = 1$

過制動  $\zeta > 1$



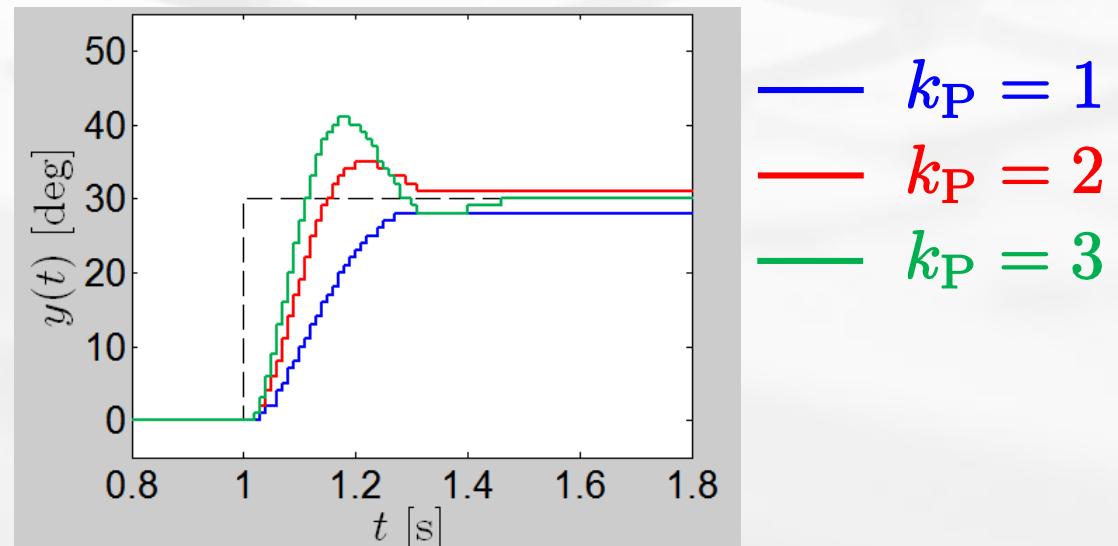
# P 制御

$$d(t) = 0$$

$$y(s) = G_{yr}(s)r(s), \quad G_{yr}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

“ $k_P \rightarrow$  大” とすると、過渡特性は .....

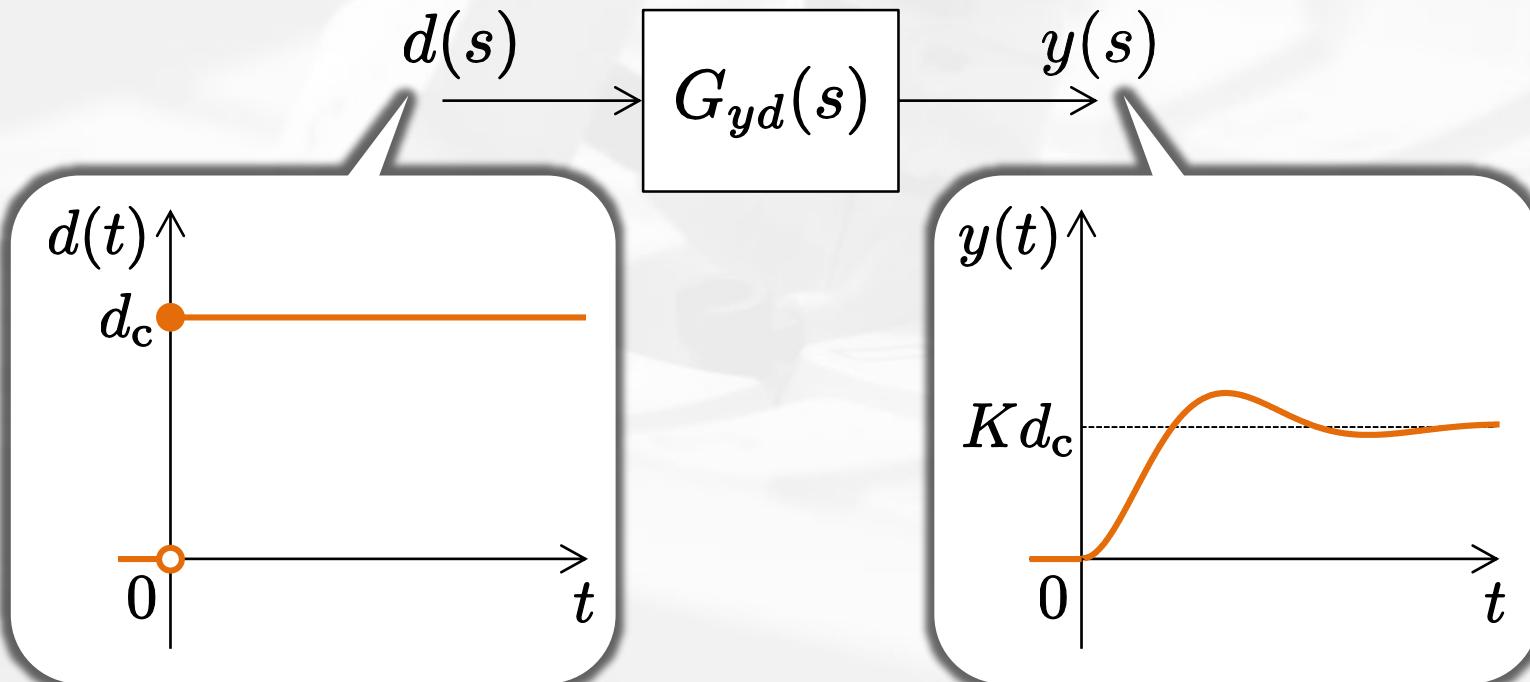
- 固有角周波数  $\omega_n = \sqrt{bk_P} \rightarrow$  大  $\rightarrow$  速応性が向上
- 減衰係数  $\zeta = \frac{a}{2\sqrt{bk_P}} \rightarrow 0 \rightarrow$  安定度が悪化



# P 制御

$$r(t) = 0$$

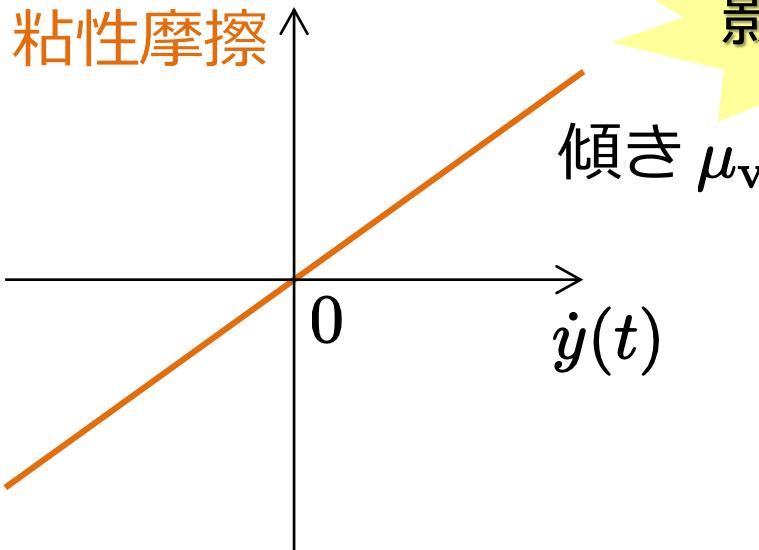
$$y(s) = G_{yd}(s)d(s), \quad G_{yd}(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$



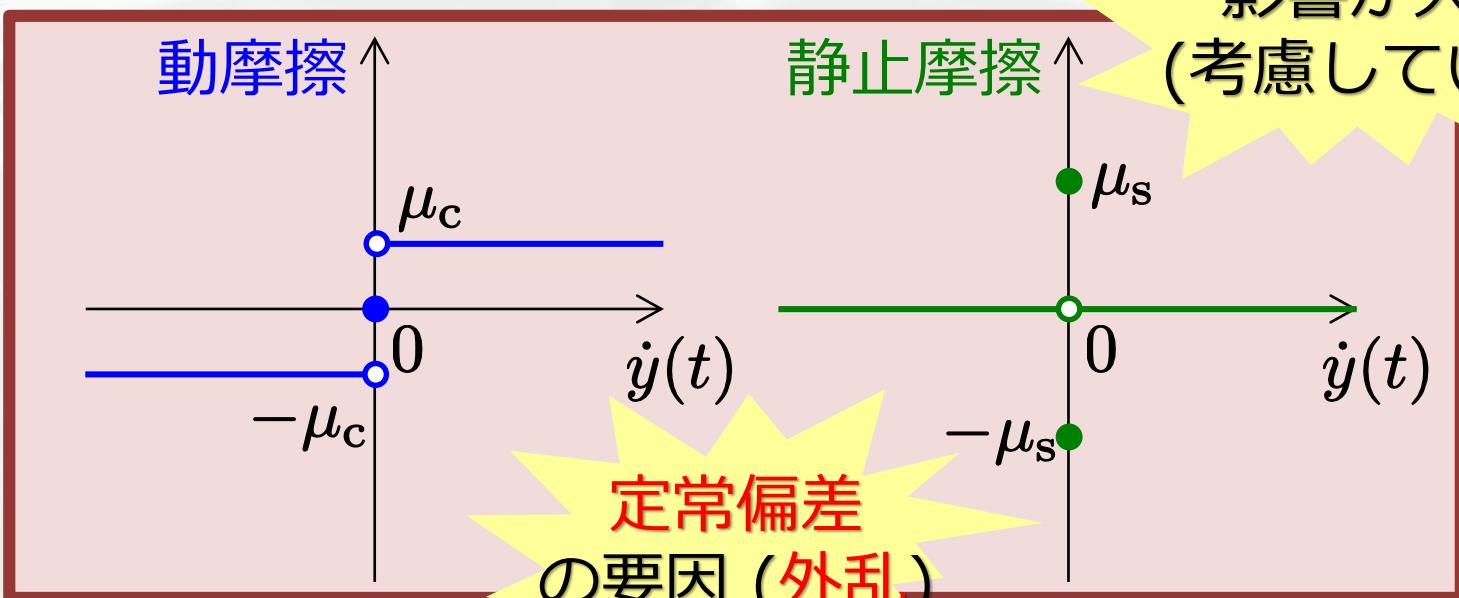
$$y_\infty = \lim_{s \rightarrow 0} sy(s) = Kd_c = \frac{d_c}{k_P} \neq 0$$

定值外乱が加わると、  
定常偏差が残る

# 摩擦の種類



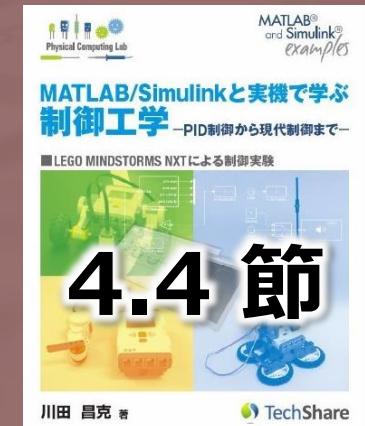
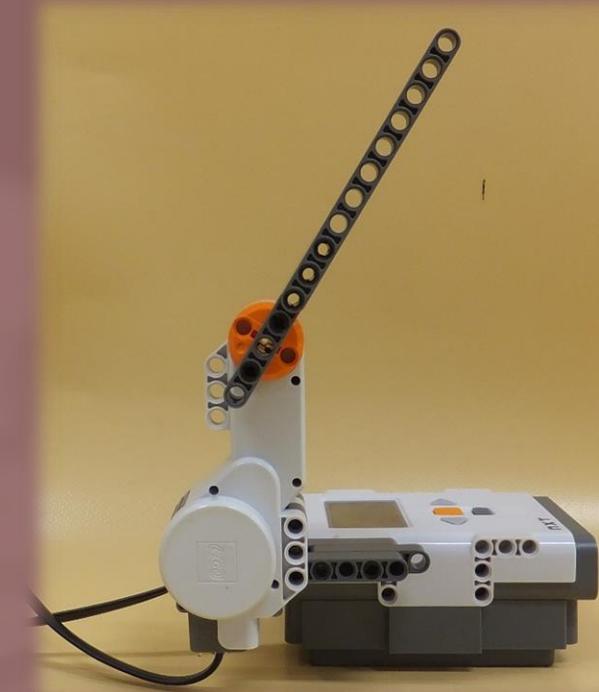
高速時に  
影響が大きい



低速時に  
影響が大きい  
(考慮していない)

# PID 制御の基礎

- ON/OFF 制御
- PID 制御
  - P 制御
  - 過渡特性を改善しよう
    - PD 制御と P-D 制御
  - 定常特性を改善しよう
    - PI-D 制御
  - I-PD 制御



# PD 制御

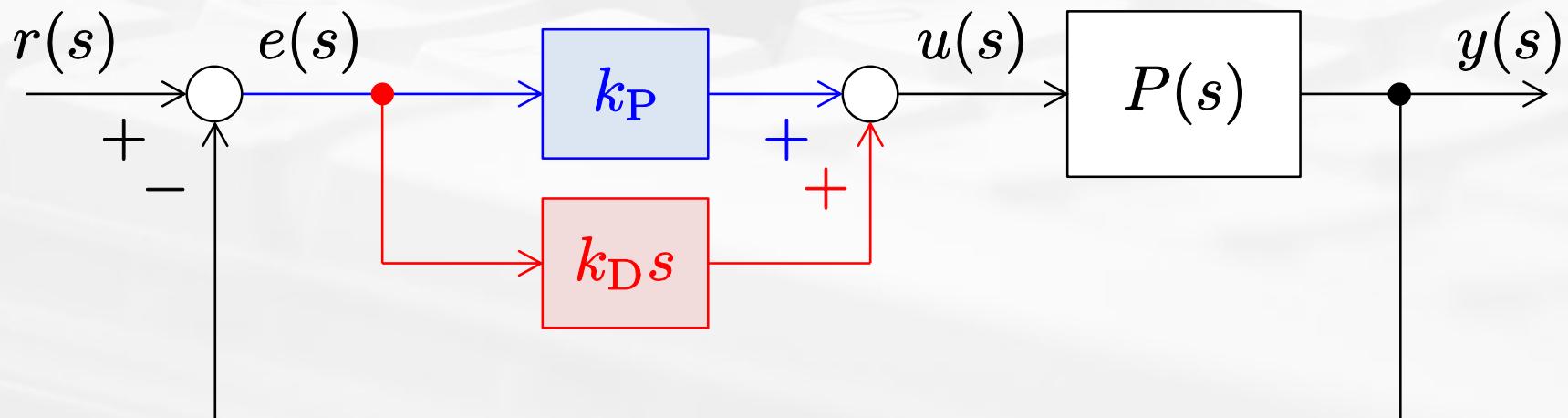
..... 振動を抑制

Derivative: 微分

## PD コントローラ

$$u(t) = k_P e(t) + k_D \dot{e}(t)$$

$$\longleftrightarrow u(s) = \underbrace{k_P e(s)}_{\text{P 動作} \\ (\text{現在})} + \underbrace{k_D s e(s)}_{\text{D 動作} \\ (\text{未来})}$$

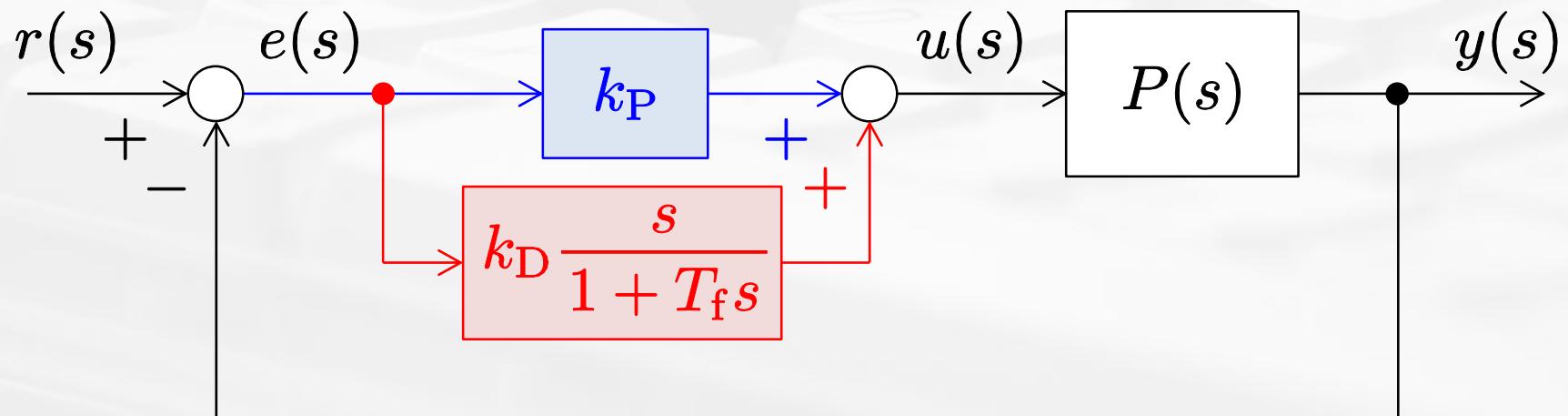


# PD 制御

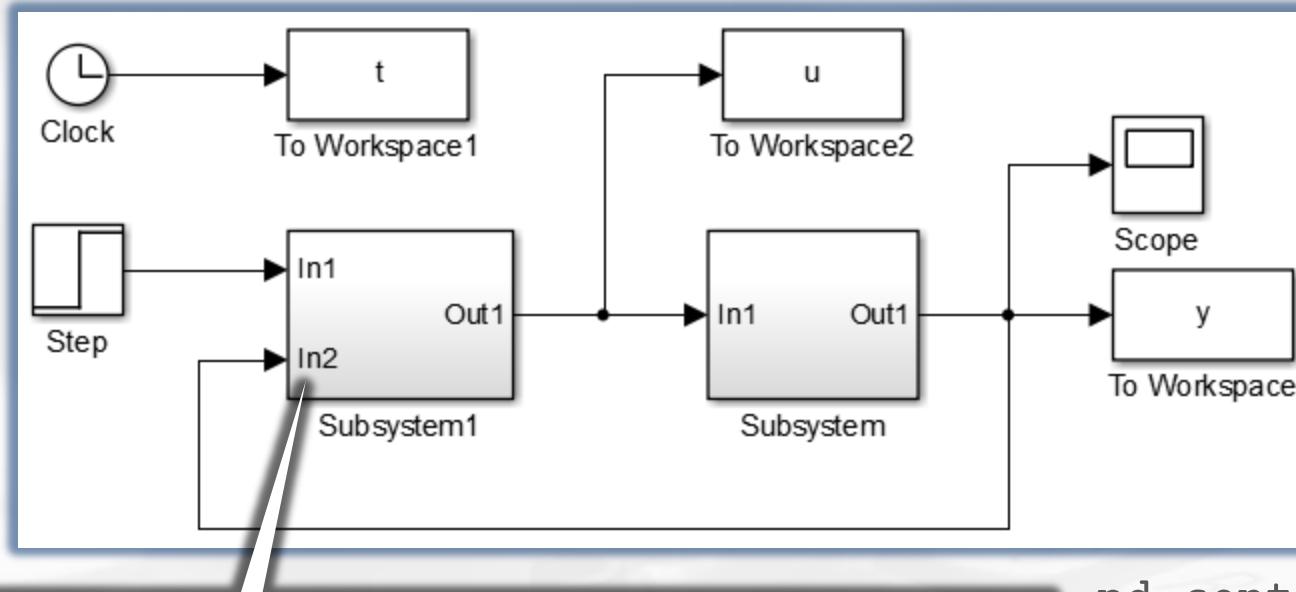
## PD コントローラ

不完全微分

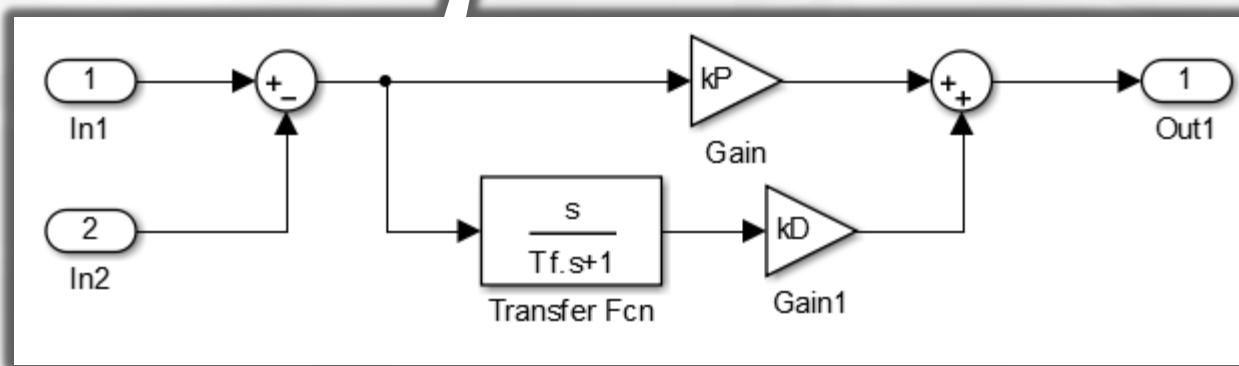
$$u(s) = k_P e(s) + k_D \frac{s}{1 + T_f s} e(s)$$



# PD 制御



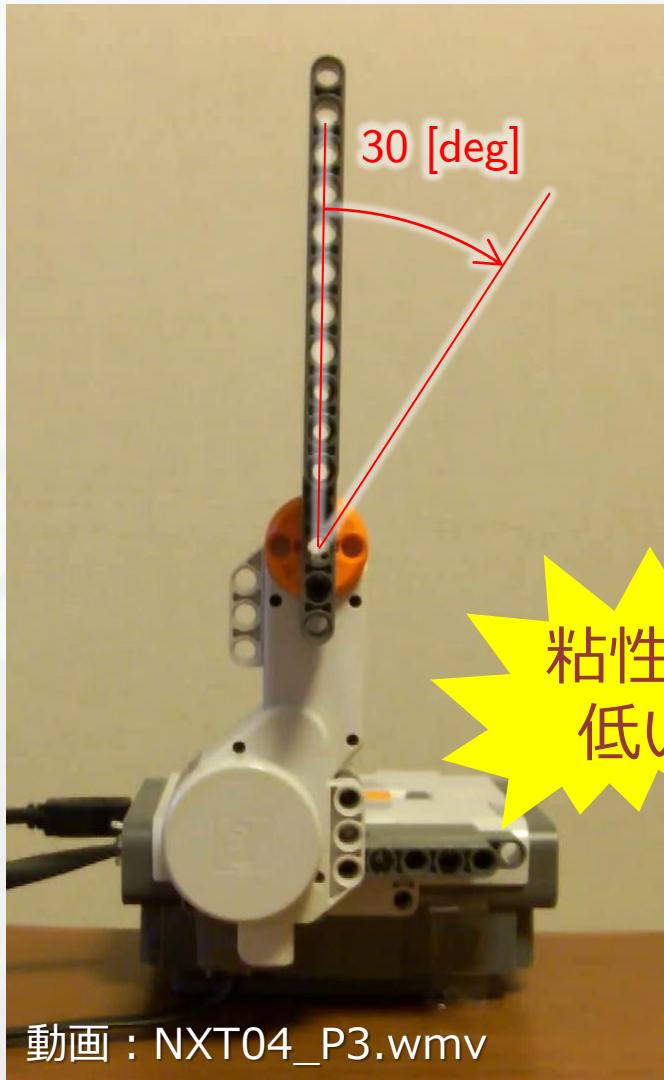
pd\_cont\_c.slx



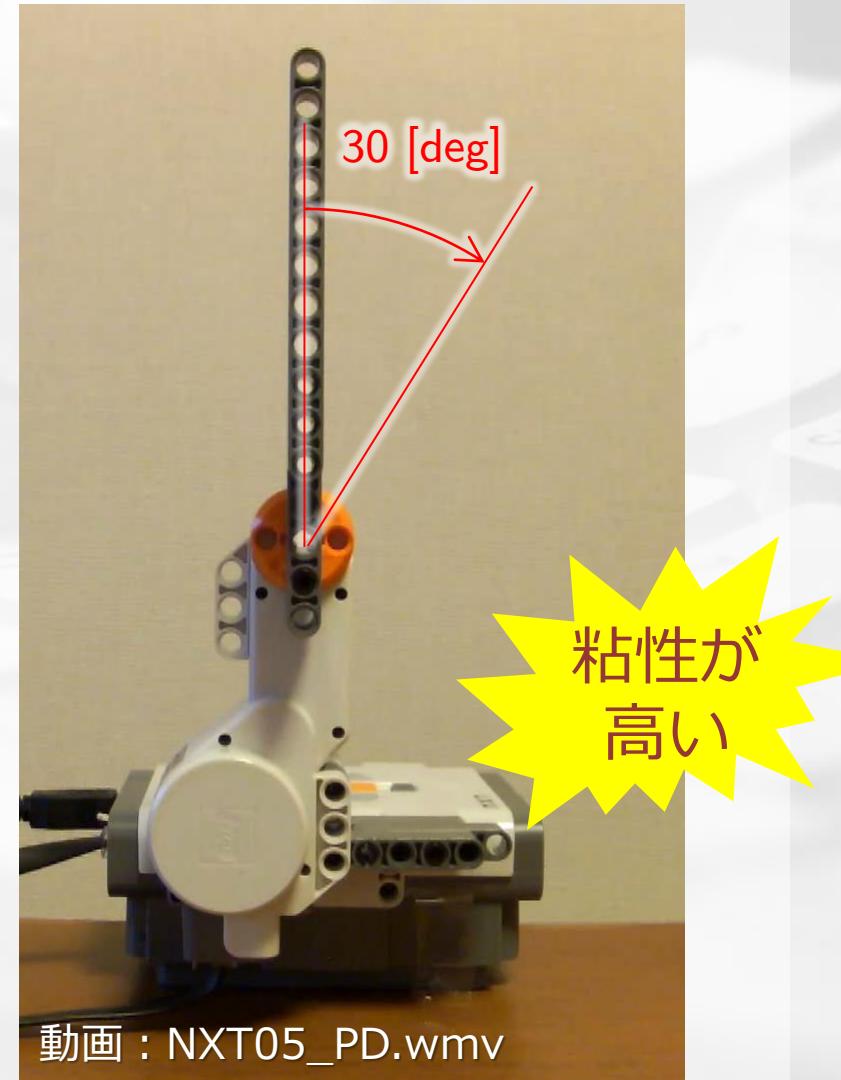
```
>> h = 0.01; Tf = 0.02;  
>> kP = 3;  
>> kD = 0.2;
```

# PD 制御

P 制御 :  $k_P = 3, k_D = 0$

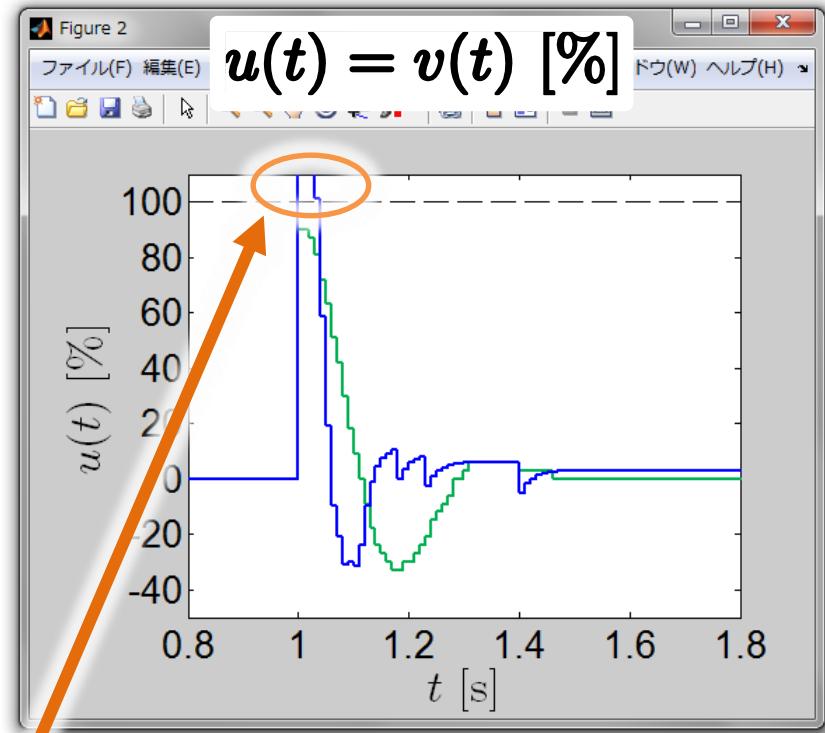
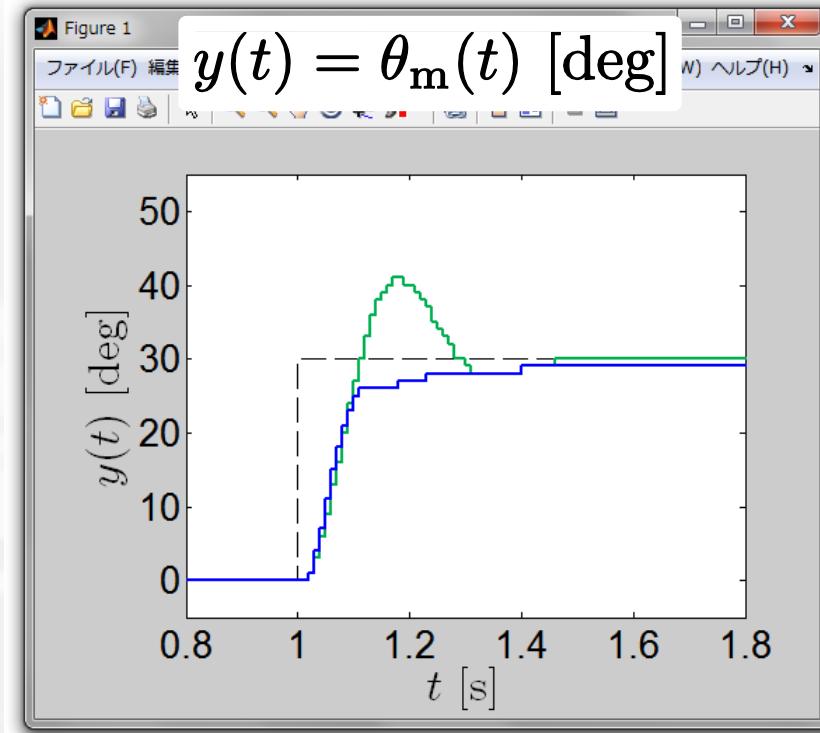


PD 制御 :  $k_P = 3, k_D = 0.2$



# PD 制御

— P 制御 :  $k_P = 3, k_D = 0$   
— PD 制御 :  $k_P = 3, k_D = 0.2$



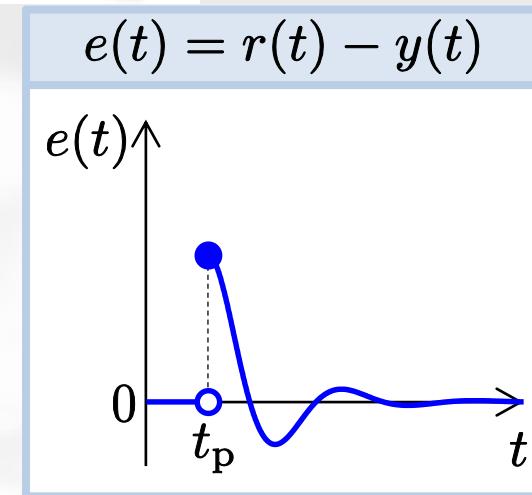
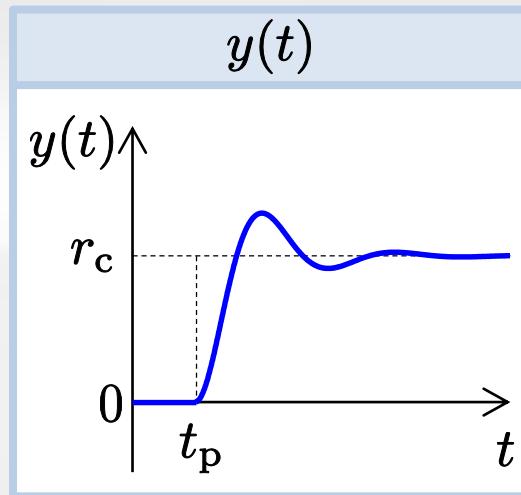
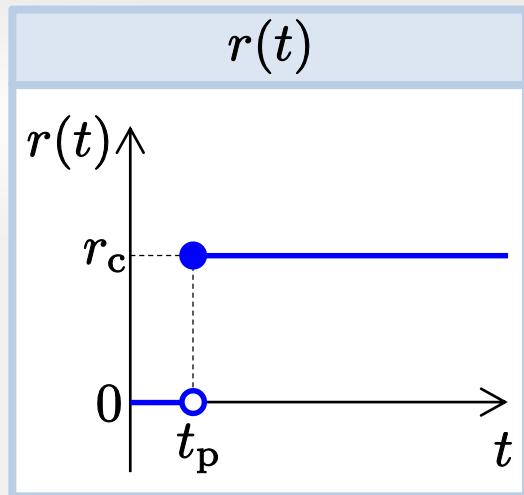
振動を抑制！！

入力が過大

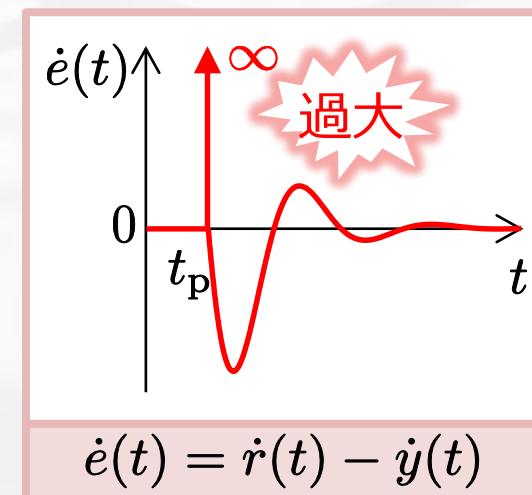
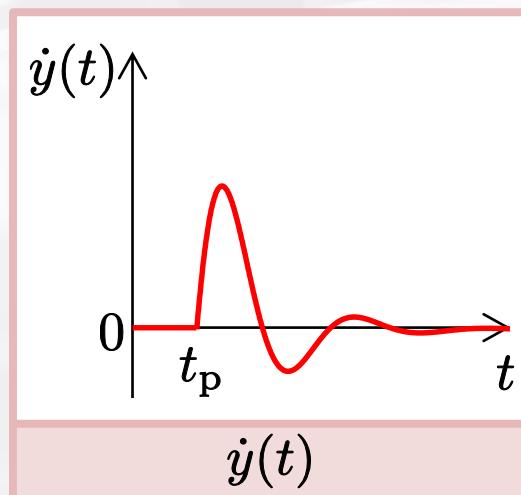
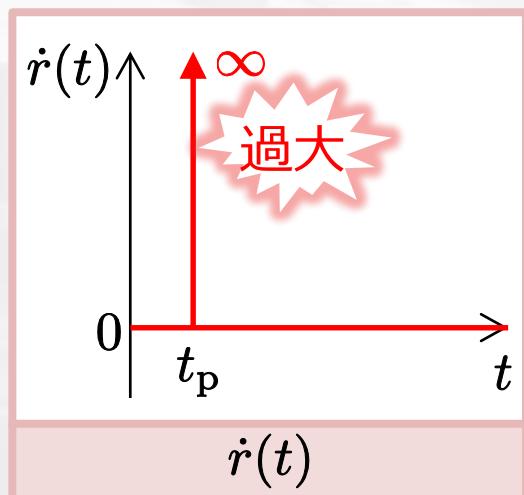
$u(t) > 100$  であるとき  
 $u(t) = 100$  に制限

# PD 制御

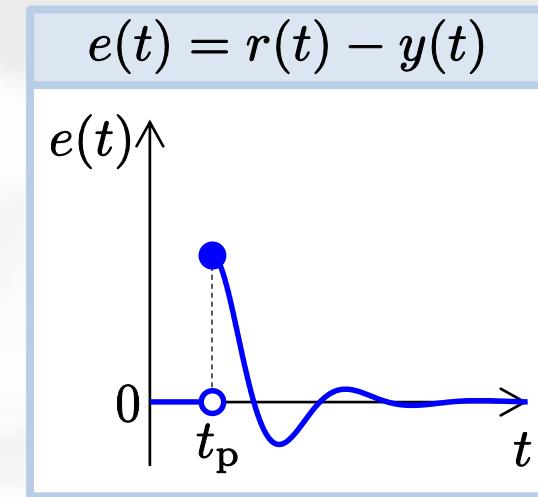
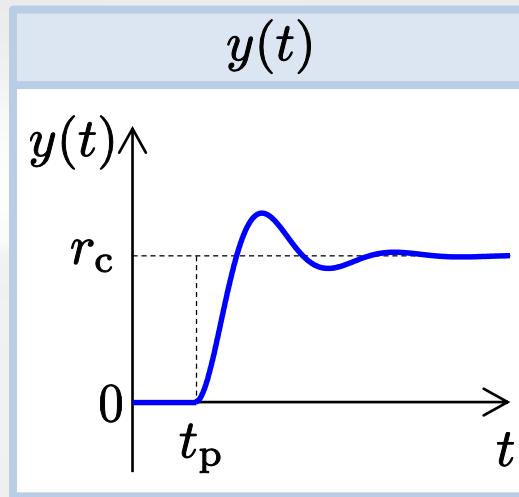
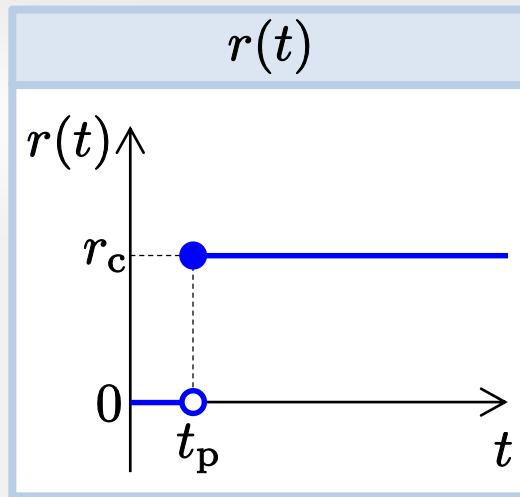
入力が過大となるのはなぜ



微分動作 :  $k_D \dot{e}(t)$

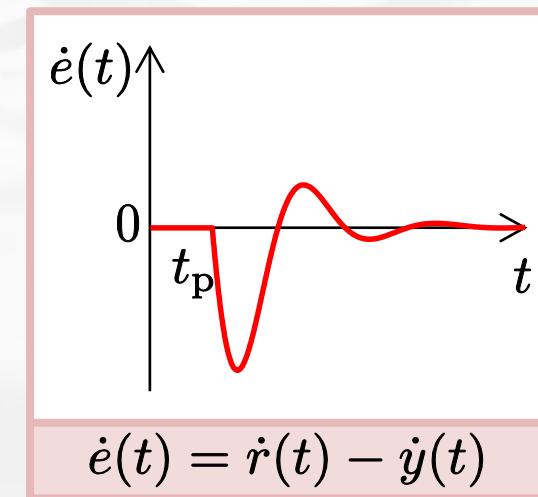
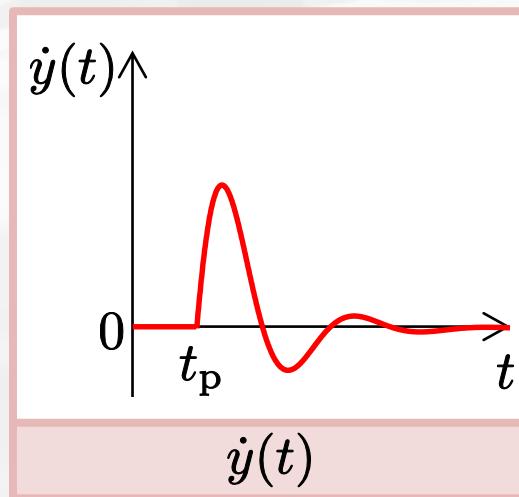
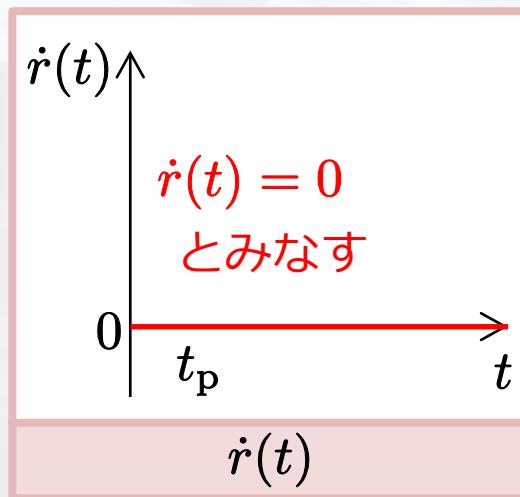


# P-D 制御 (微分先行型 PD 制御)



入力を小さくするには…

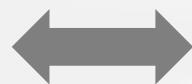
微分動作 :  $-k_D \dot{y}(t)$



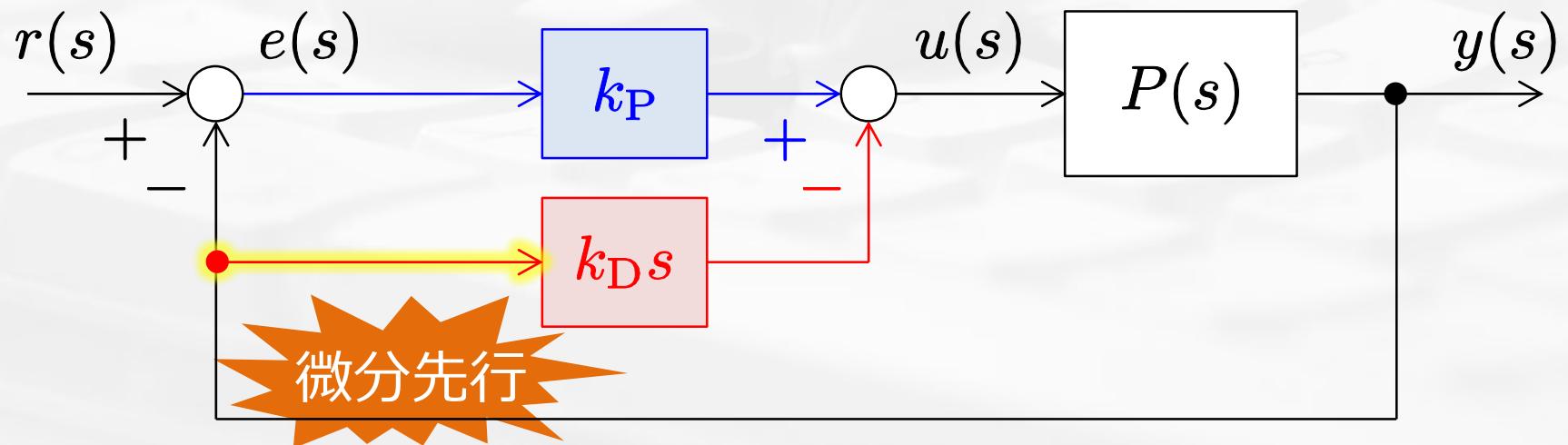
# P-D 制御 (微分先行型 PD 制御)

## P-D コントローラ

$$u(t) = k_P e(t) - k_D \dot{y}(t)$$



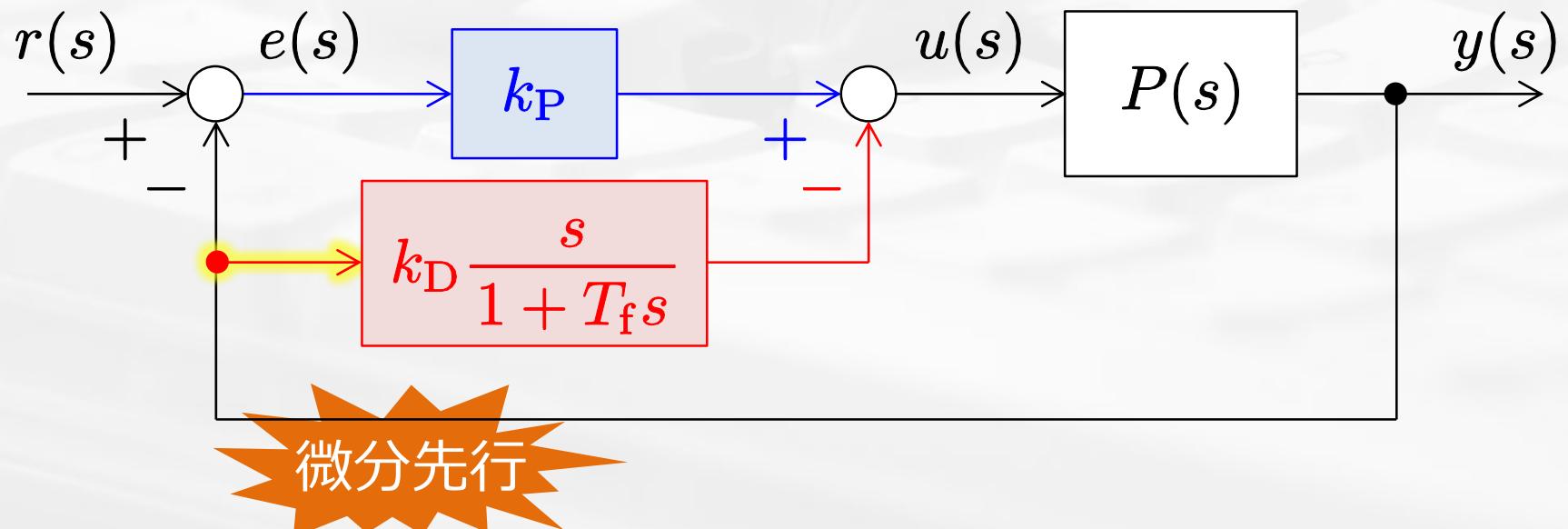
$$u(s) = k_P e(s) - k_D s y(s)$$



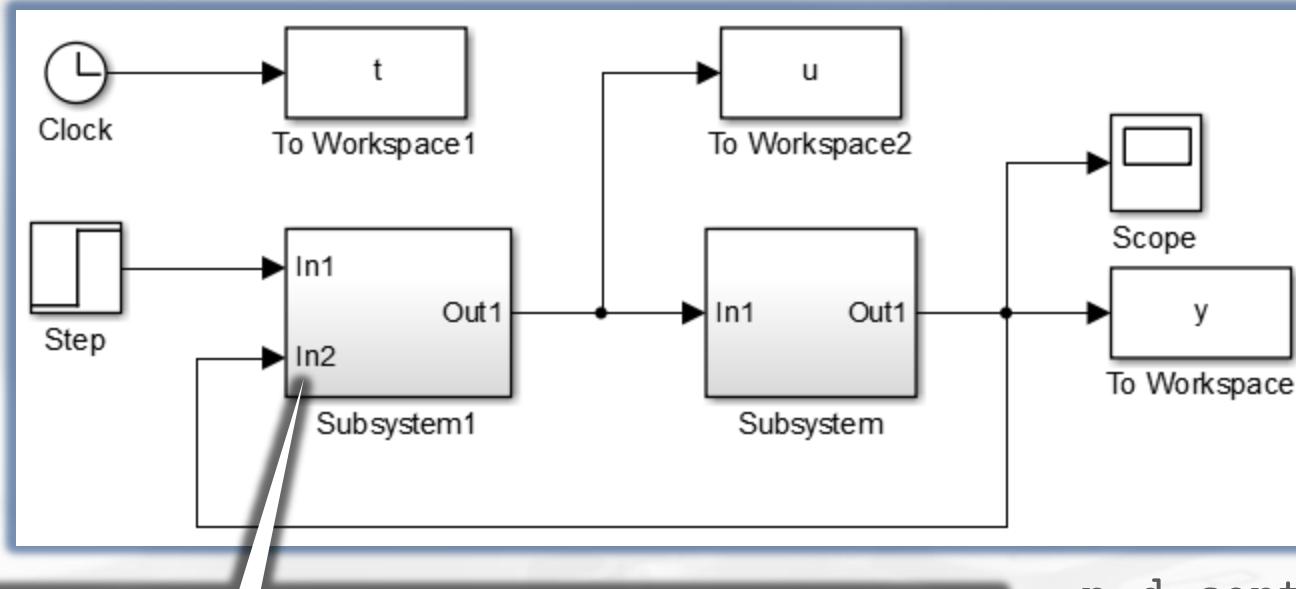
# P-D 制御 (微分先行型 PD 制御)

P-D コントローラ ..... 不完全微分

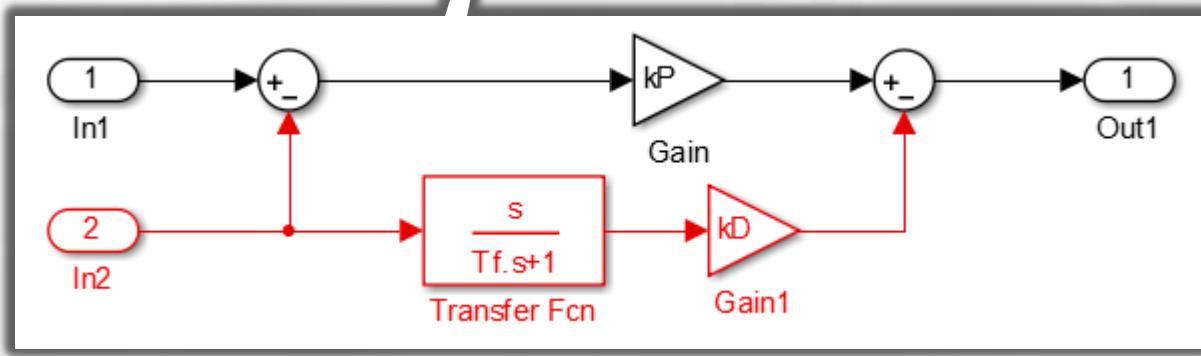
$$u(s) = k_P e(s) - k_D \frac{s}{1 + T_f s} y(s)$$



# P-D 制御 (微分先行型 PD 制御)



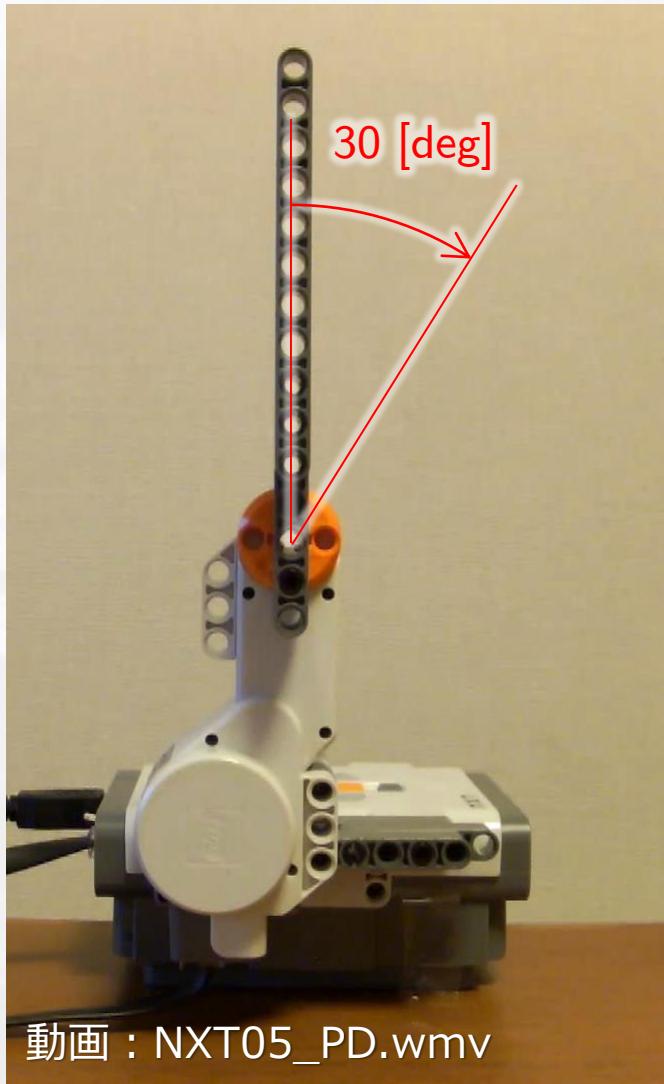
p\_d\_cont\_c.slx



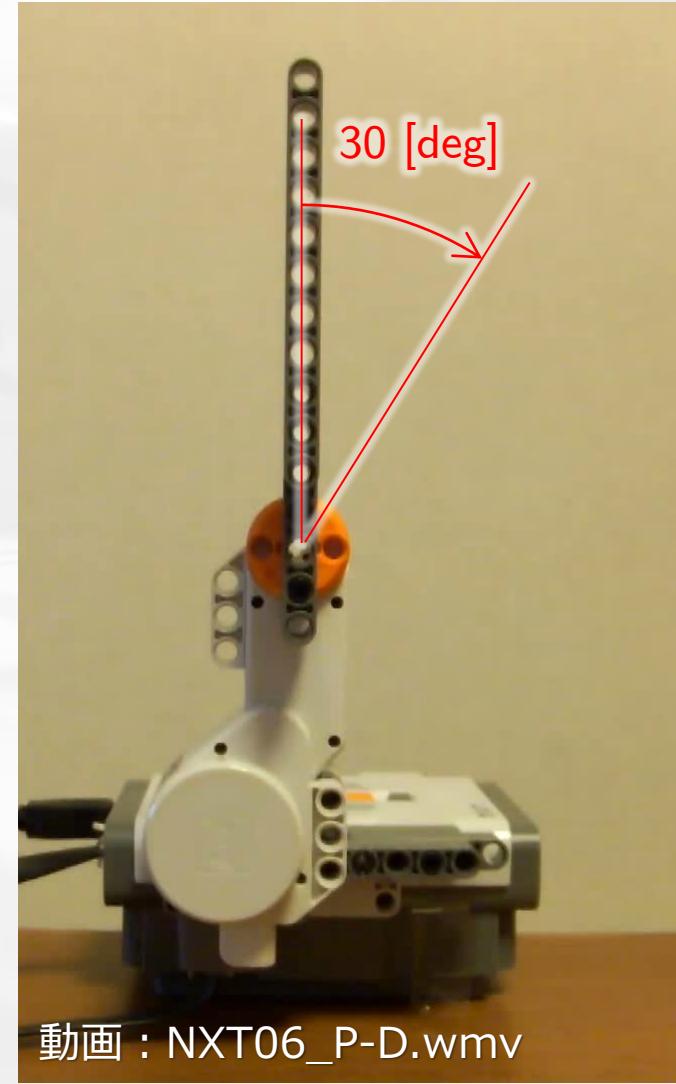
```
>> h = 0.01; Tf = 0.02;  
>> kP = 3;  
>> kD = 0.2;
```

# P-D 制御 (微分先行型 PD 制御)

PD 制御 :  $k_P = 3$ ,  $k_D = 0.2$

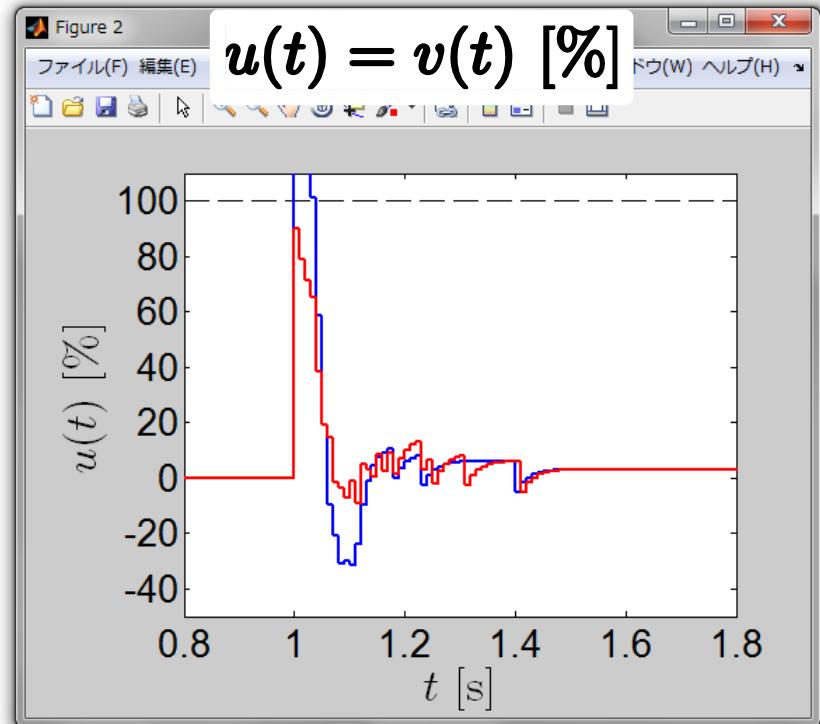
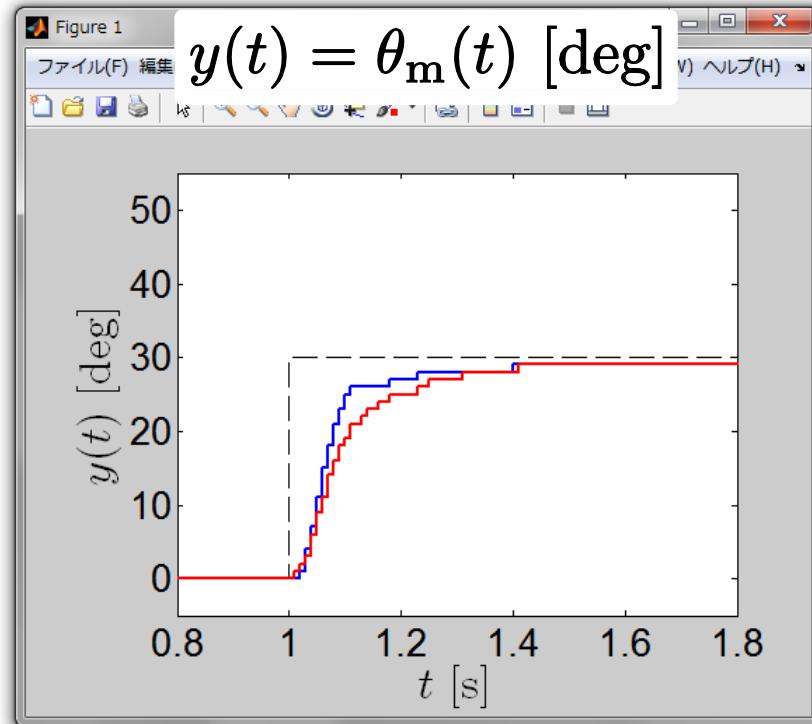


P-D 制御 :  $k_P = 3$ ,  $k_D = 0.2$



# P-D 制御 (微分先行型 PD 制御)

— PD 制御 :  $k_P = 3, k_D = 0.2$   
— P-D 制御 :  $k_P = 3, k_D = 0.2$



- PD, P-D 制御とともに定常偏差が残る

- P-D 制御では入力の大きさを抑制

# P-D 制御 (微分先行型 PD 制御)

マス・ばね・ダンパ系

$$M\ddot{y}(t) + \mu\dot{y}(t) + ky(t) = u(t)$$



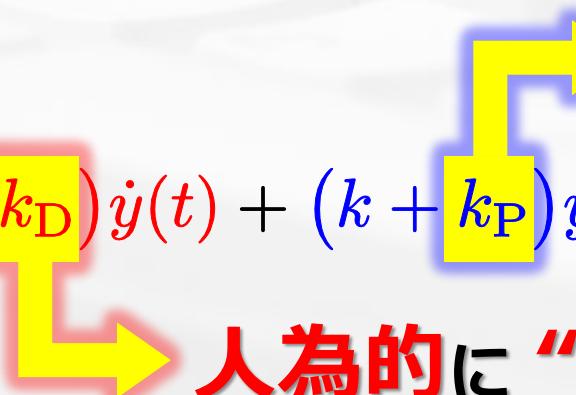
P-D コントローラ

$$u(t) = k_P(r(t) - y(t)) - k_D\dot{y}(t)$$



人為的に “ばね”  
を強くする

$$M\ddot{y}(t) + (\mu + k_D)\dot{y}(t) + (k + k_P)y(t) = k_P r(t)$$



人為的に “ダンパ”  
を強くする

# P-D 制御 (微分先行型 PD 制御)



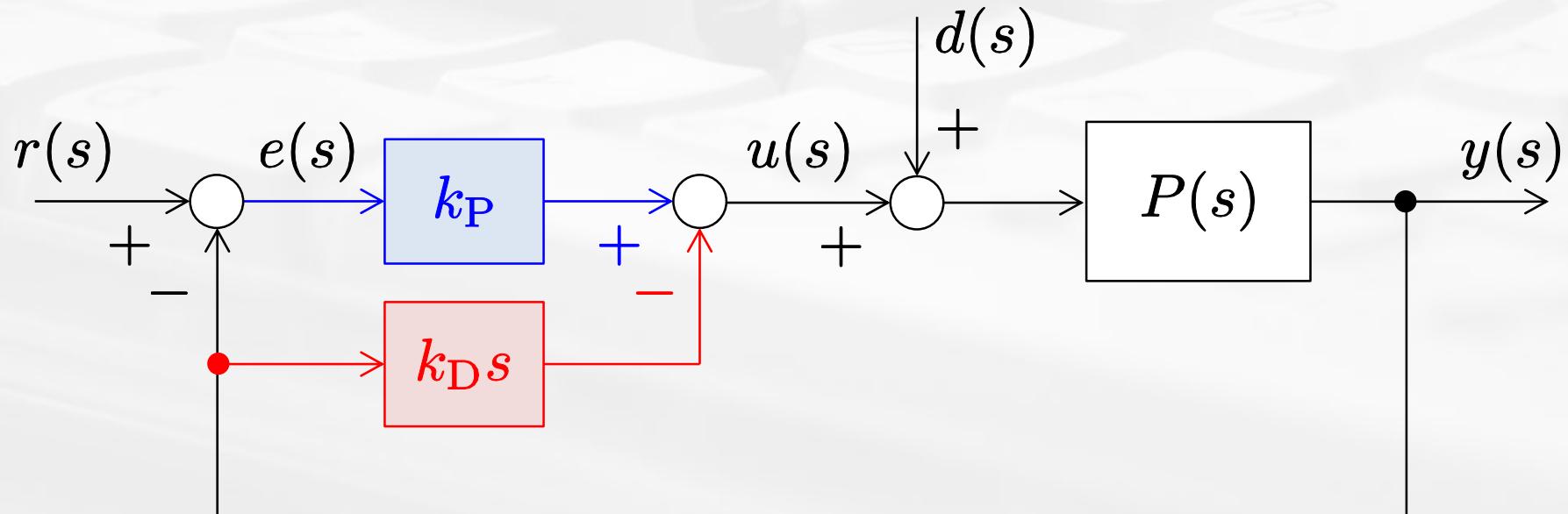
# 理論的な解析

# P-D 制御 (微分先行型 PD 制御)

2 次遅れ  
要素

$$y(s) = G_{yr}(s)r(s) + G_{yd}(s)d(s)$$

$$\begin{cases} G_{yr}(s) = \frac{b\textcolor{blue}{k}_P}{s^2 + (a + b\textcolor{red}{k}_D)s + b\textcolor{blue}{k}_P} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \\ G_{yd}(s) = \frac{b}{s^2 + (a + b\textcolor{red}{k}_D)s + b\textcolor{blue}{k}_P} = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \end{cases}$$



# P-D 制御 (微分先行型 PD 制御)

2 次遅れ  
要素

$$y(s) = G_{yr}(s)r(s) + G_{yd}(s)d(s)$$

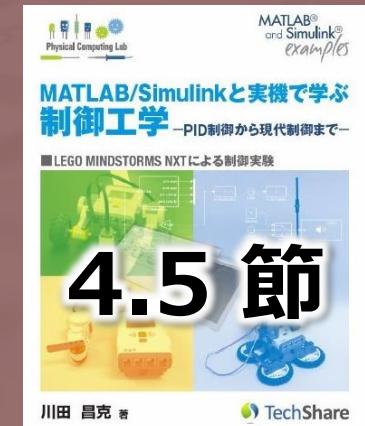
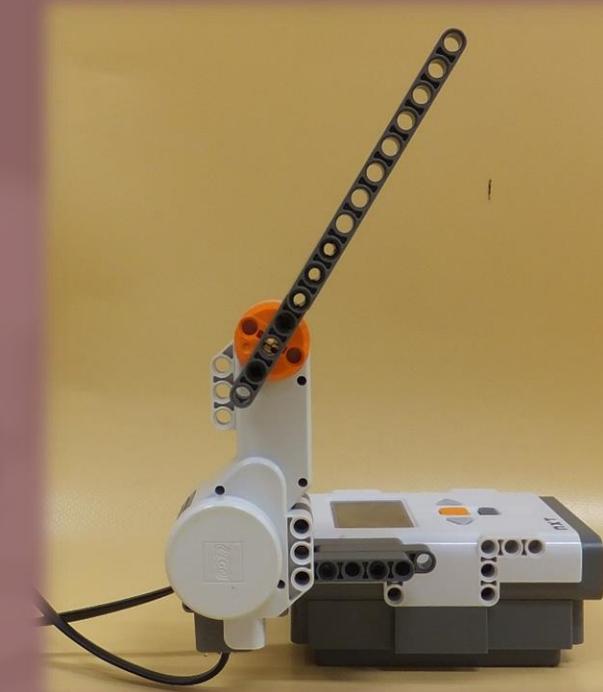
$$\begin{cases} G_{yr}(s) = \frac{b\textcolor{blue}{k}_P}{s^2 + (a + b\textcolor{red}{k}_D)s + b\textcolor{blue}{k}_P} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \\ G_{yd}(s) = \frac{b}{s^2 + (a + b\textcolor{red}{k}_D)s + b\textcolor{blue}{k}_P} = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \end{cases}$$

$$\left\{ \begin{array}{l} \text{固有角周波数} \quad \omega_n = \sqrt{b\textcolor{blue}{k}_P} \\ \text{減衰係数} \quad \zeta = \frac{a + b\textcolor{red}{k}_D}{2\omega_n} = \frac{a + b\textcolor{red}{k}_D}{2\sqrt{b\textcolor{blue}{k}_P}} \rightarrow \text{改善!} \\ \text{ゲイン} \quad K = \frac{1}{\textcolor{blue}{k}_P} \end{array} \right.$$

- 過渡特性は改善
- 定常偏差が残る

# PID 制御の基礎

- ON/OFF 制御
- PID 制御
  - P 制御
  - 過渡特性を改善しよう
    - PD 制御と P-D 制御
  - 定常特性を改善しよう
    - PI-D 制御
  - I-PD 制御



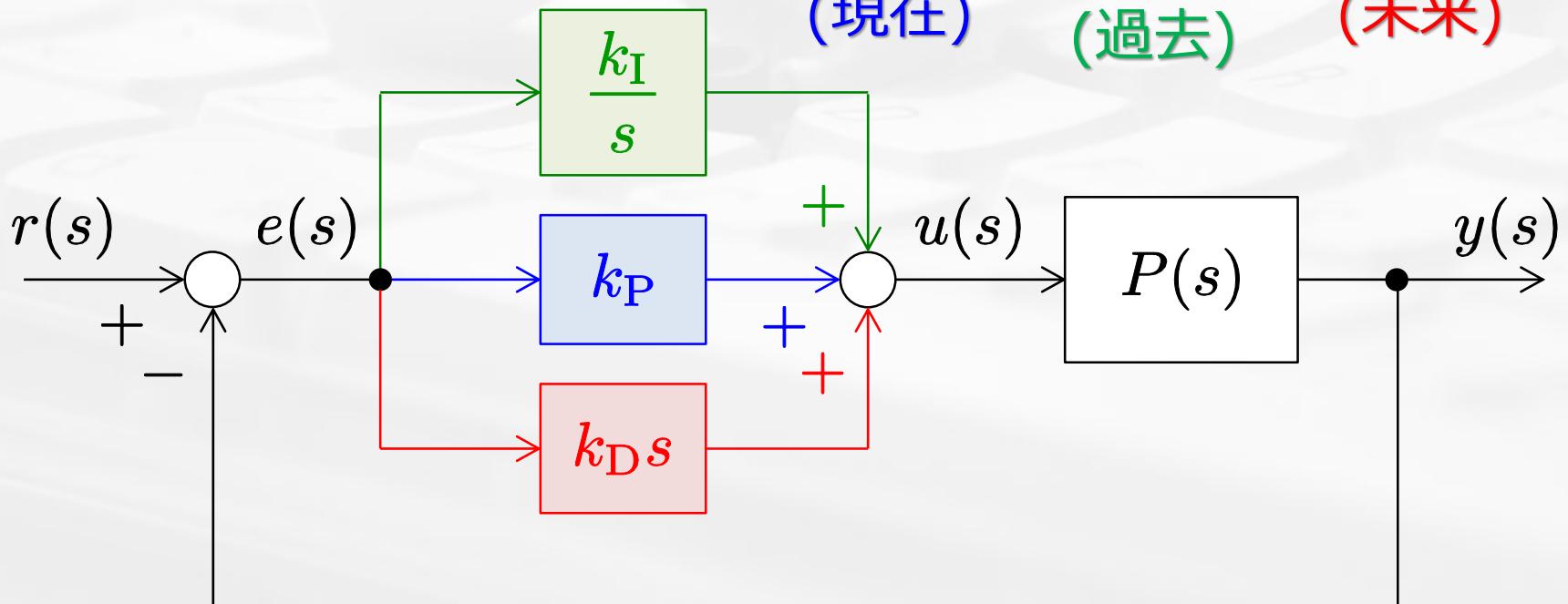
# PID 制御

Integral: 積分

## PID コントローラ

$$u(t) = k_P e(t) + k_I \int_0^t e(t) dt + k_D \dot{e}(t)$$

$$u(s) = \underbrace{k_P e(s)}_{\text{P 動作 (現在)}} + \underbrace{\frac{k_I}{s} e(s)}_{\text{I 動作 (過去)}} + \underbrace{k_D s e(s)}_{\text{D 動作 (未来)}}$$



# PID 制御

## PID コントローラ

$$u(t) = k_P e(t) + k_I \int_0^t e(t) dt + k_D \dot{e}(t)$$

P コントローラ .....  $k_I = 0, k_D = 0$

$$u(t) = k_P e(t)$$

PD コントローラ .....  $k_I = 0$

$$u(t) = k_P e(t) + k_D \dot{e}(t)$$

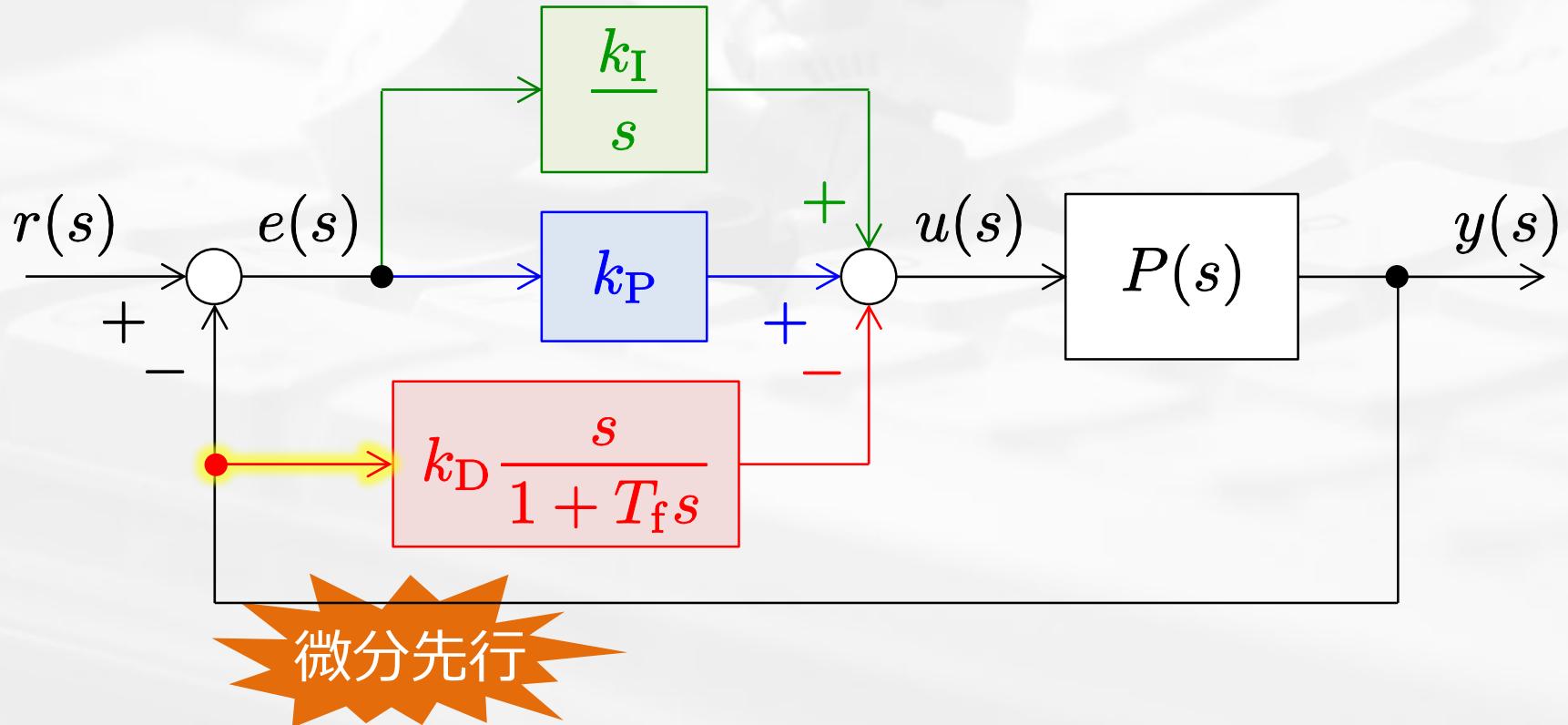
PI コントローラ .....  $k_D = 0$

$$u(t) = k_P e(t) + k_I \int_0^t e(t) dt$$

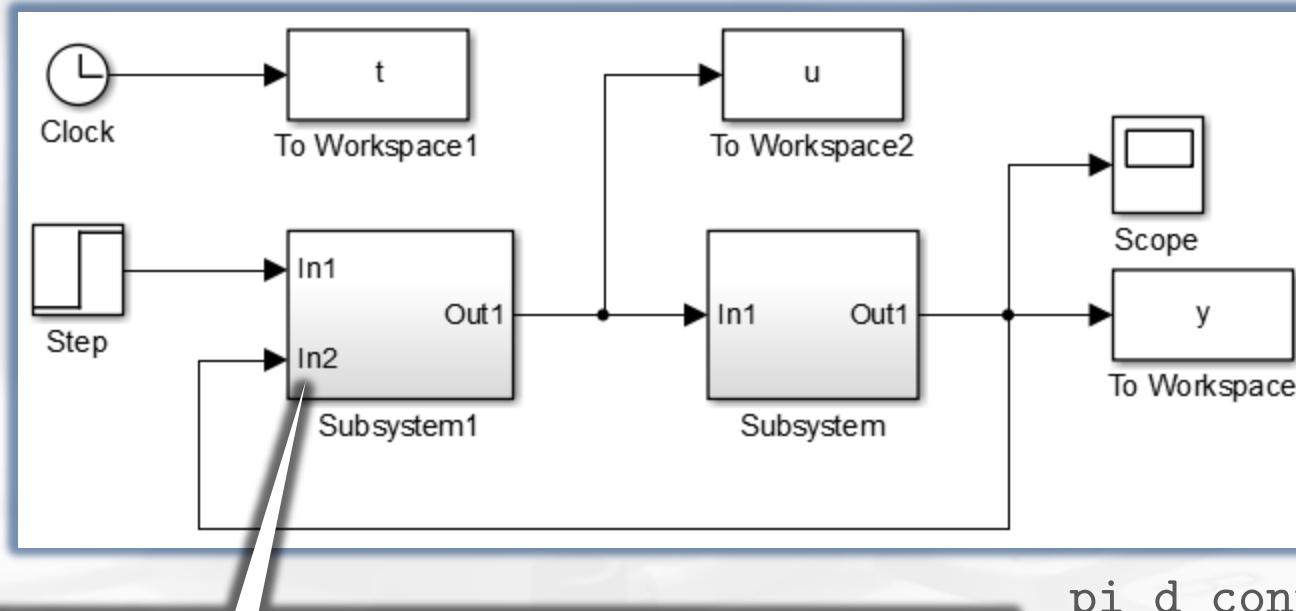
# PI-D 制御 (微分先行型 PID 制御)

PI-D コントローラ .....不完全微分

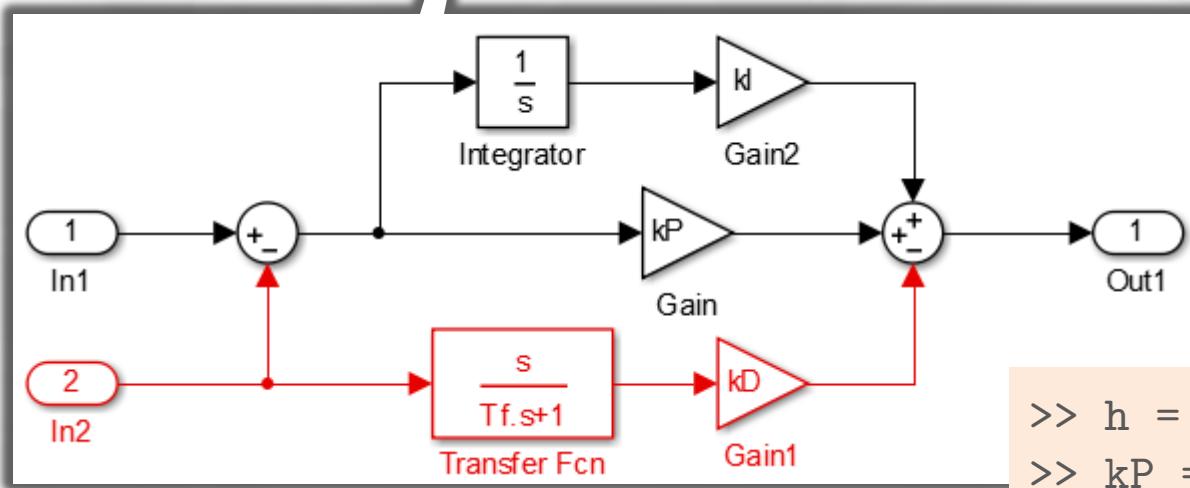
$$u(s) = k_{\text{P}}e(s) + \frac{k_{\text{I}}}{s}e(s) - k_{\text{D}} \frac{s}{1 + T_{\text{f}}s}y(s)$$



# PI-D 制御 (微分先行型 PID 制御)



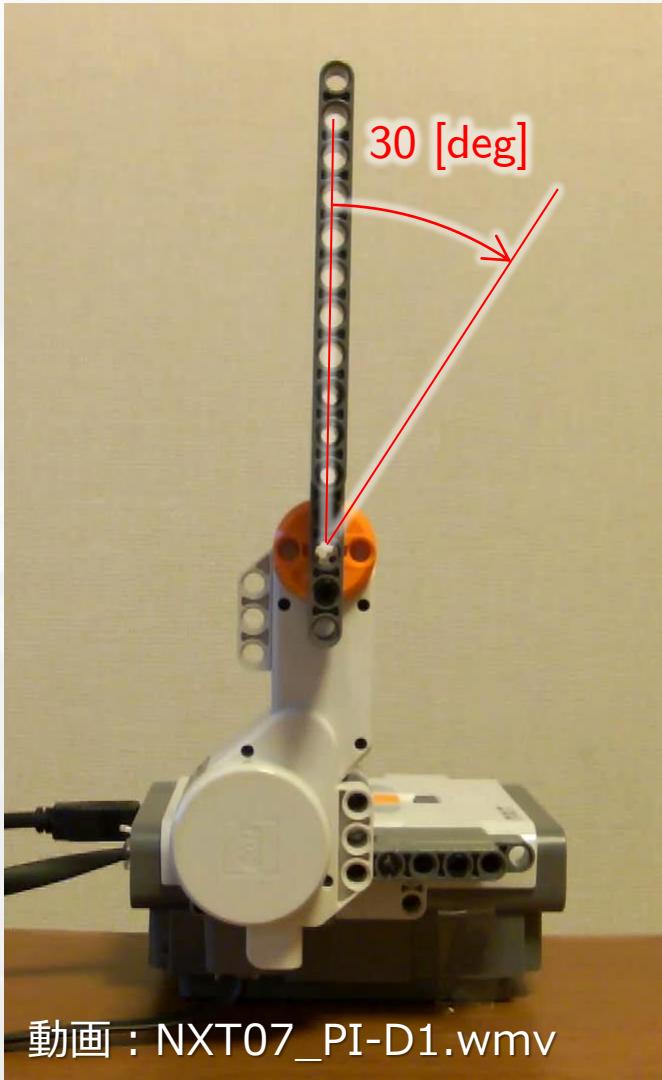
pi\_d\_cont\_c.slx



```
>> h = 0.01; Tf = 0.02;  
>> kP = 3; kD = 0.2;  
>> kI = 10;
```

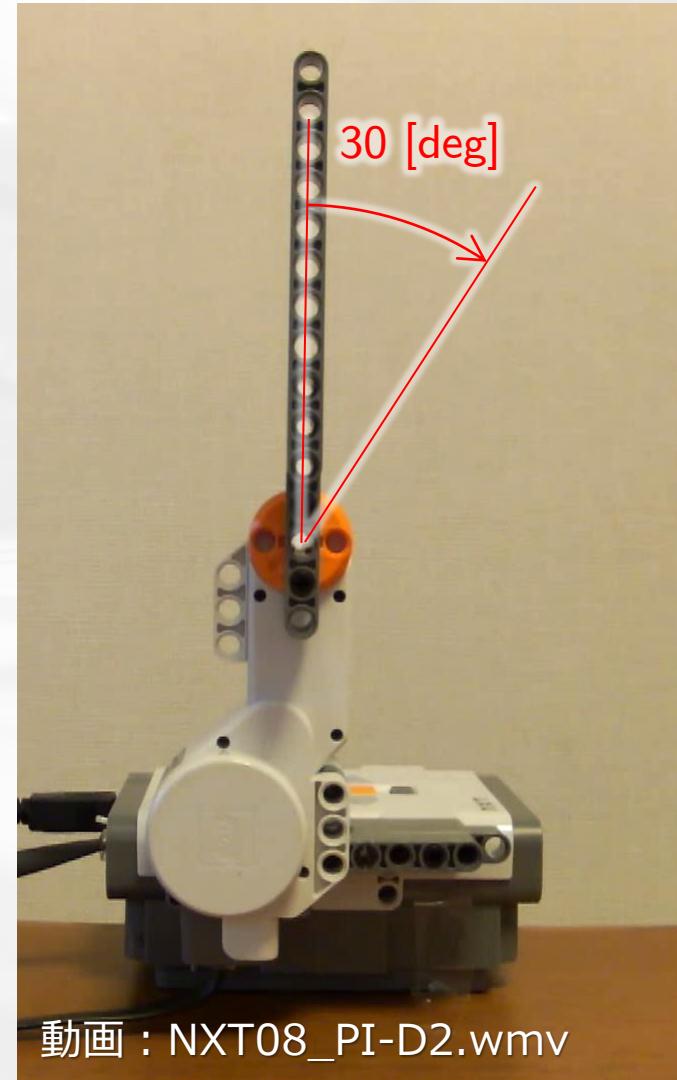
# PI-D 制御 (微分先行型 PID 制御)

$k_P = 3, k_D = 0.2, k_I = 10$



動画 : NXT07\_PI-D1.wmv

$k_P = 3, k_D = 0.2, k_I = 20$

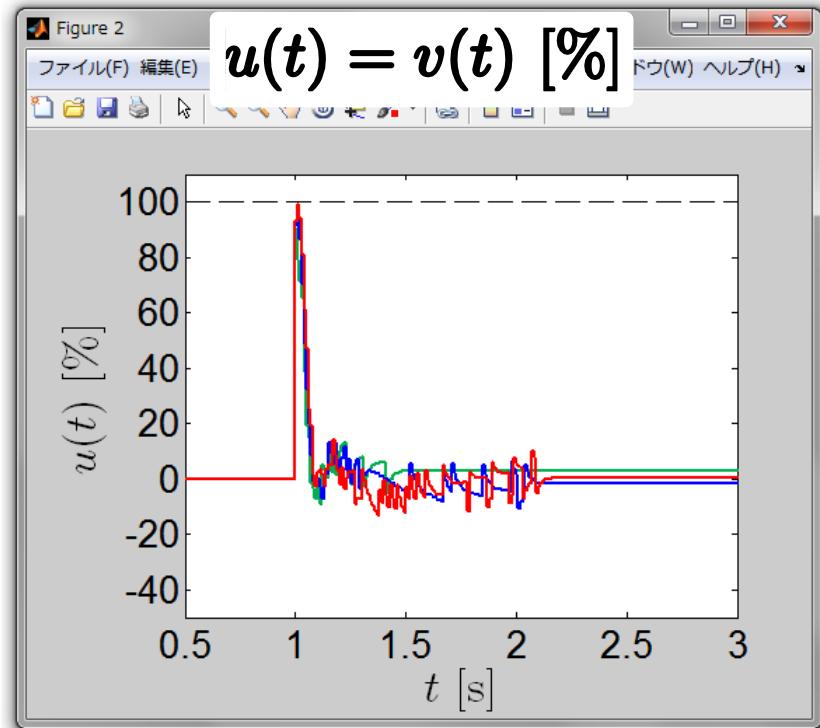
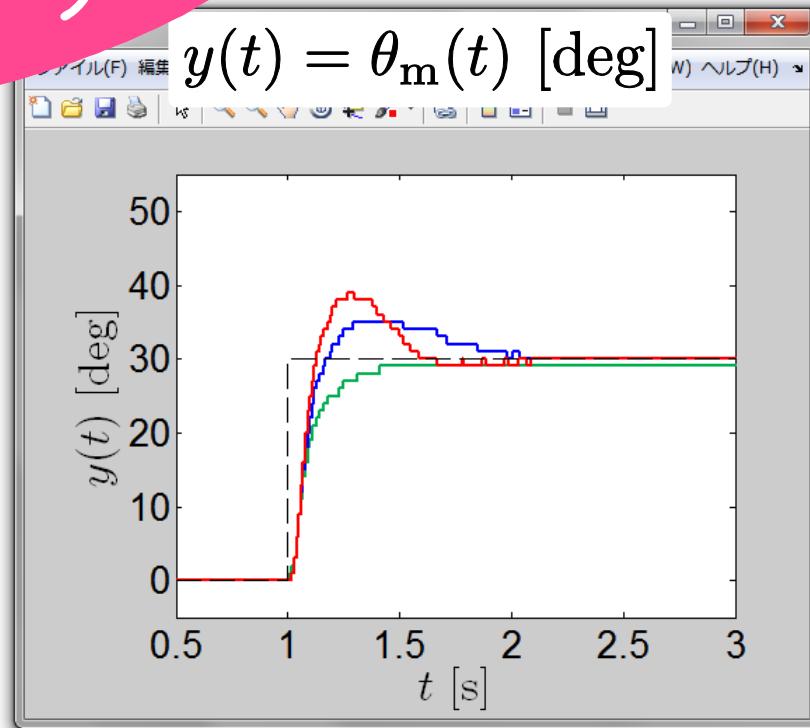


動画 : NXT08\_PI-D2.wmv

# PI-D 制御 (微分先行型 PID 制御)

愛(／)  
の力

- P-D 制御 :  $k_P = 3, k_D = 0.2, k_I = 0$
- PI-D 制御 :  $k_P = 3, k_D = 0.2, k_I = 10$
- PI-D 制御 :  $k_P = 3, k_D = 0.2, k_I = 20$



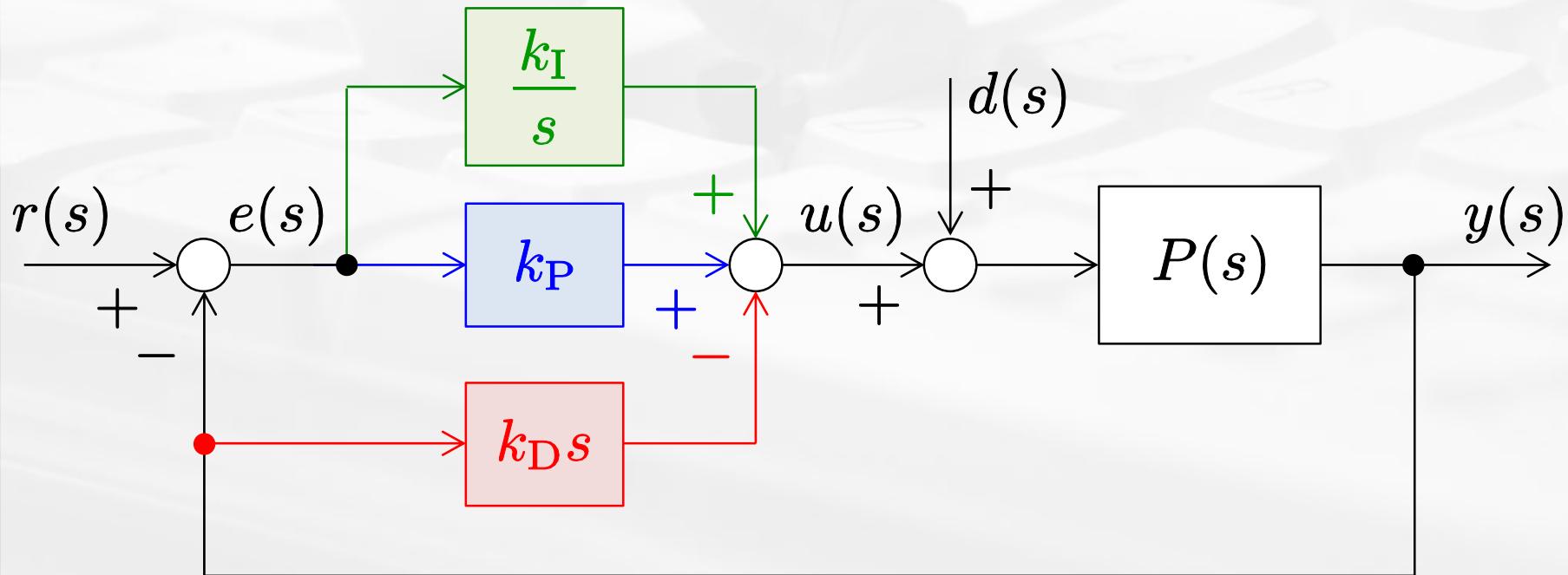
- 定常偏差が 0 となる
- オーバーシュートを生じる
- 収束が遅い

# 理論的な解析

# PI-D 制御 (微分先行型 PID 制御)

$$y(s) = G_{yr}(s)r(s) + G_{yd}(s)d(s)$$

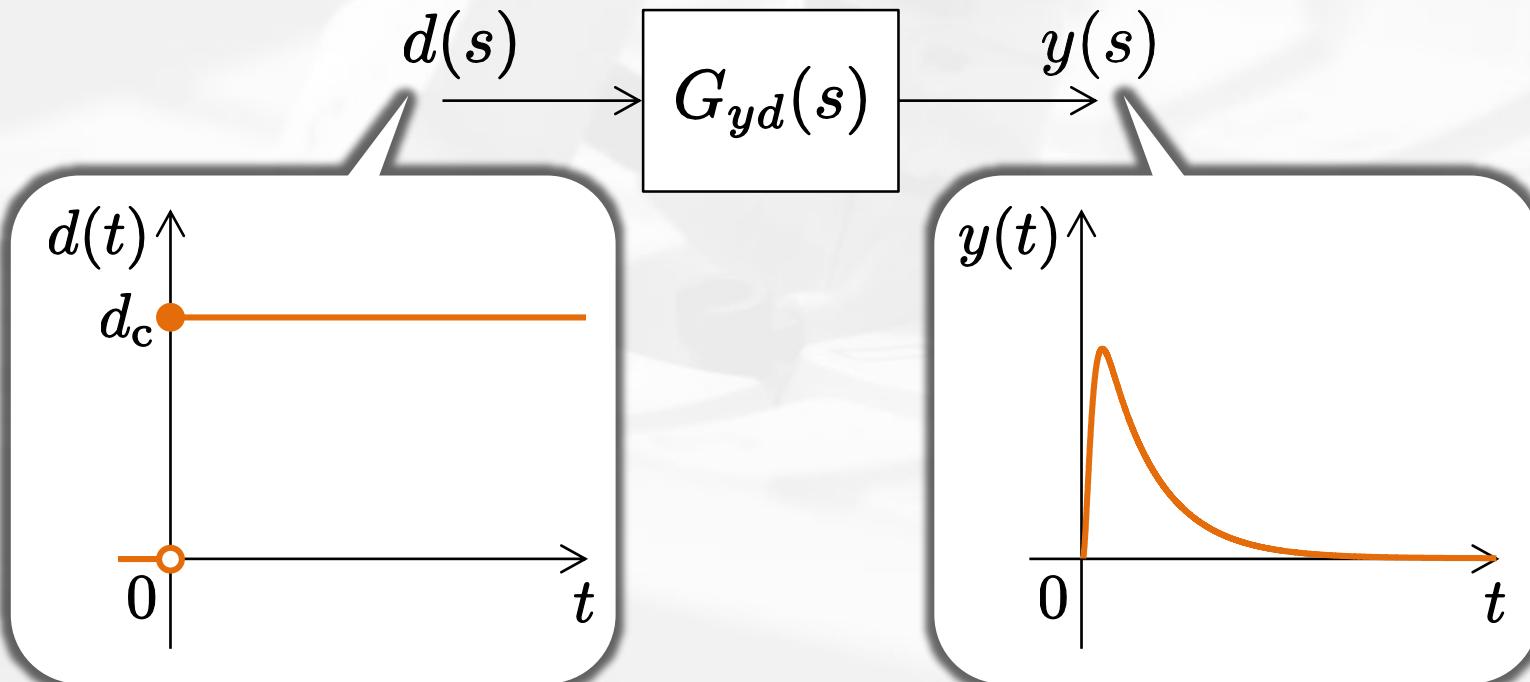
$$\begin{cases} G_{yr}(s) = \frac{b\mathbf{k}_P s + b\mathbf{k}_I}{s^3 + (a + b\mathbf{k}_D)s^2 + b\mathbf{k}_P s + b\mathbf{k}_I} \\ G_{yd}(s) = \frac{bs}{s^3 + (a + b\mathbf{k}_D)s^2 + b\mathbf{k}_P s + b\mathbf{k}_I} \end{cases}$$



# PI-D 制御 (微分先行型 PID 制御)

$$r(t) = 0$$

$$y(s) = G_{yd}(s)d(s), \quad G_{yd}(s) = \frac{b s}{s^3 + (a + b k_D)s^2 + b k_P s + b k_I}$$



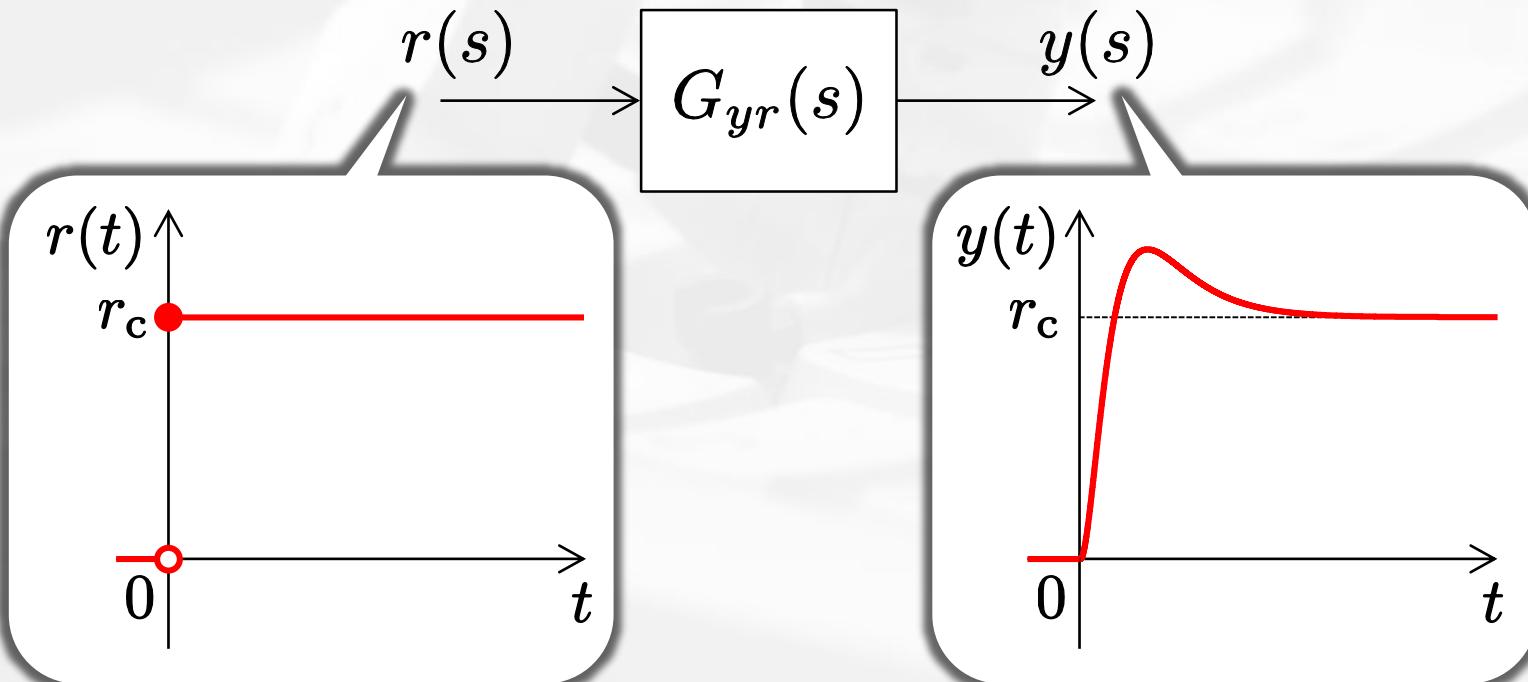
$$y_\infty = \lim_{s \rightarrow 0} s y(s) = 0$$

定值外乱が加わっても  
定常偏差は 0

# PI-D 制御 (微分先行型 PID 制御)

$$d(t) = 0$$

$$y(s) = G_{yr}(s)r(s), \quad G_{yr}(s) = \frac{bk_P s + bk_I}{s^3 + (a + bk_D)s^2 + bk_P s + bk_I}$$

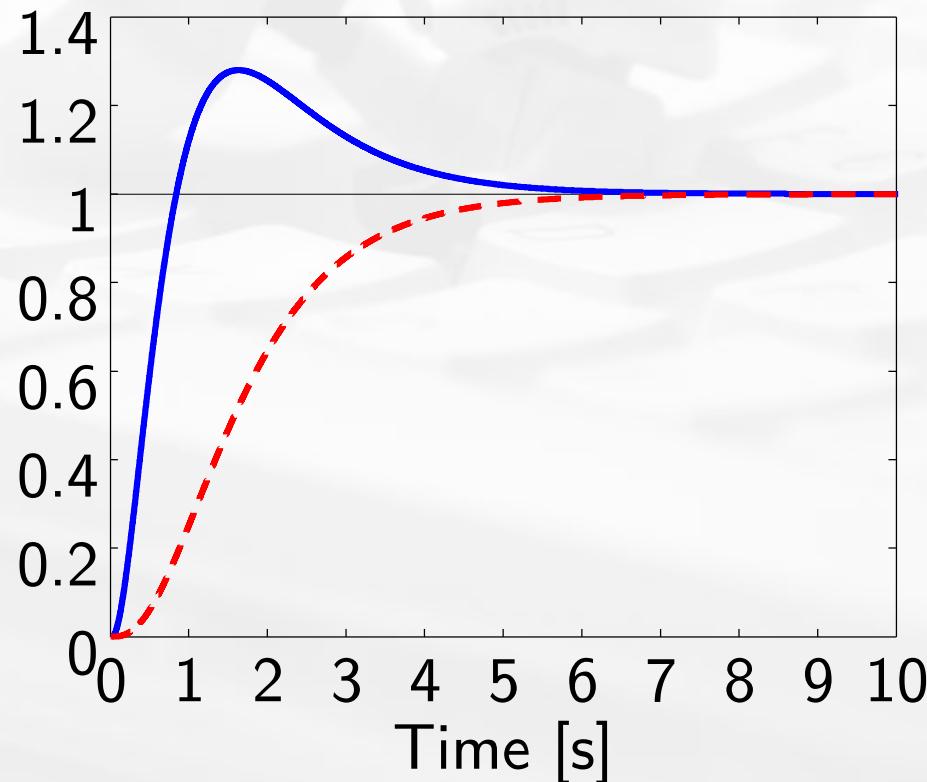


零点に起因する  
オーバーシュートを生じる

# 零点に起因するオーバーシュート

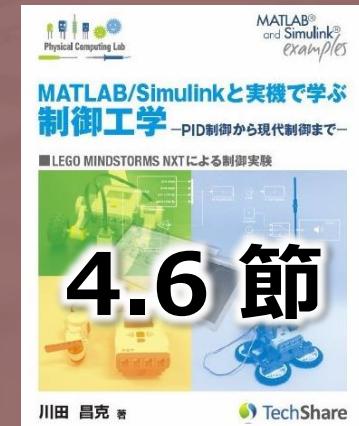
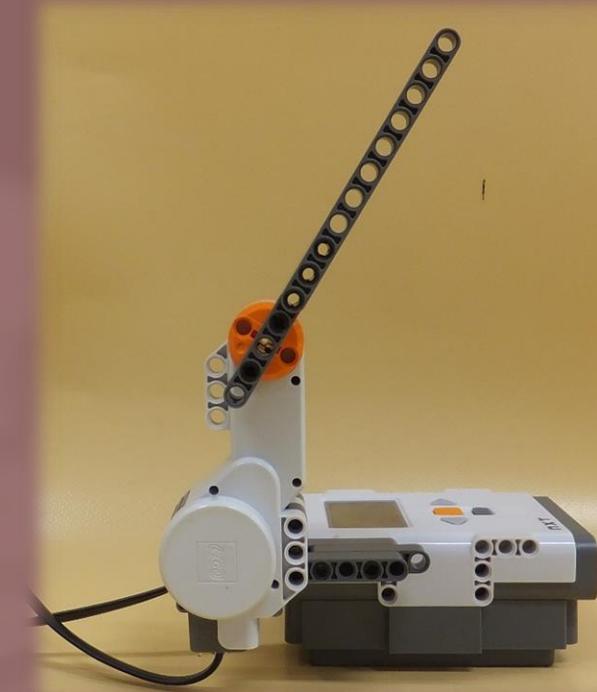
$$G_{yr}(s) = \frac{6(2s+1)}{(s+1)(s+2)(s+3)}$$
 極 :  $-1, -2, -3$  零点 :  $-1/2$

$$G_{yr}(s) = \frac{6}{(s+1)(s+2)(s+3)}$$
 極 :  $-1, -2, -3$  零点 : なし



# PID 制御の基礎

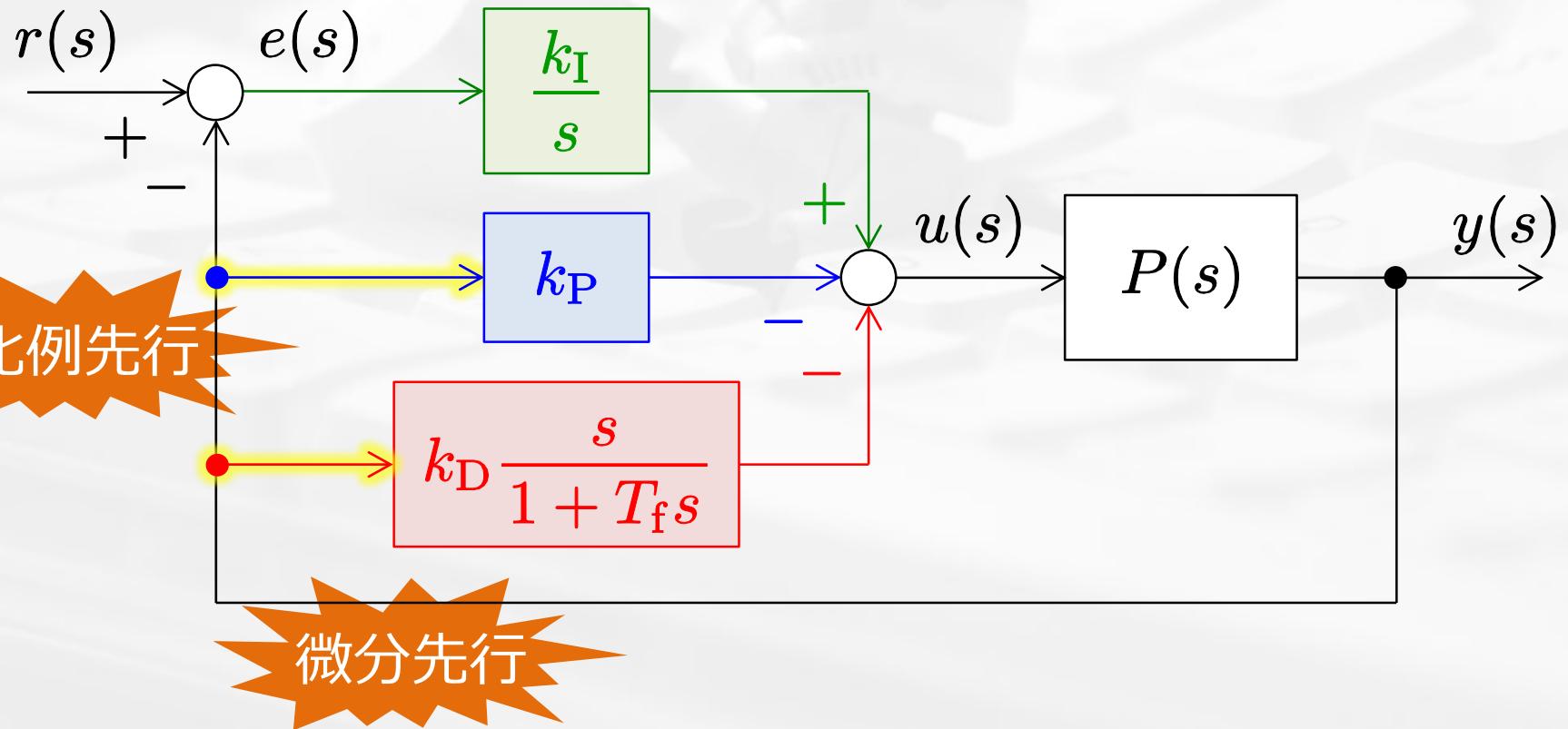
- ON/OFF 制御
- PID 制御
  - P 制御
  - 過渡特性を改善しよう
    - PD 制御と P-D 制御
  - 定常特性を改善しよう
    - PI-D 制御
  - I-PD 制御



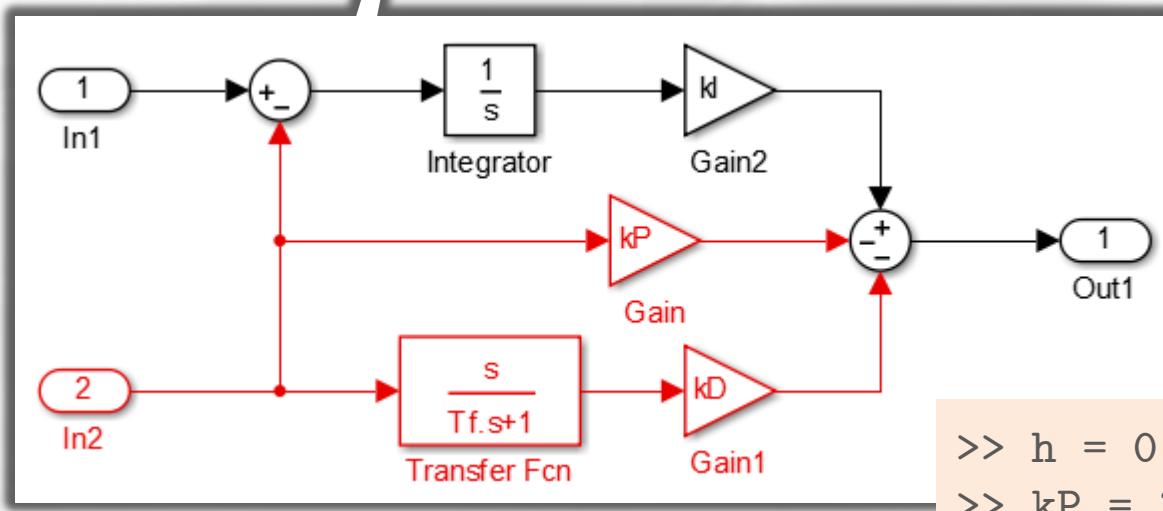
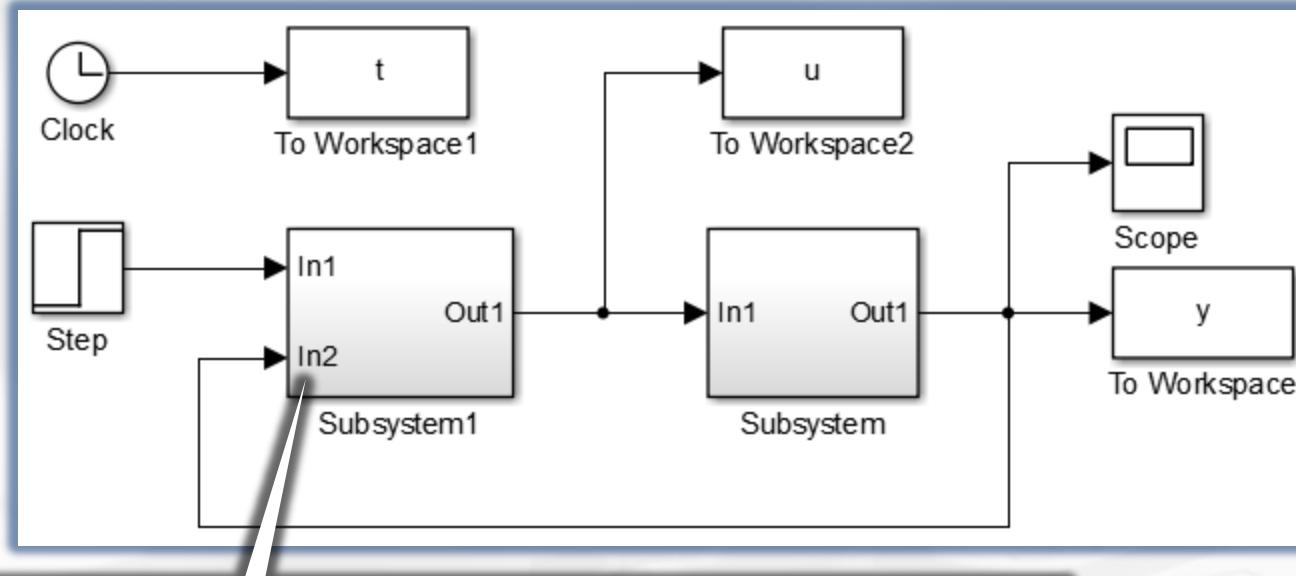
# I-PD 制御 (比例・微分先行型 PID 制御)

I-PD コントローラ .....不完全微分

$$u(s) = -k_P y(s) + \frac{k_I}{s} e(s) - k_D \frac{s}{1 + T_f s} y(s)$$



# I-PD 制御 (比例・微分先行型 PID 制御)



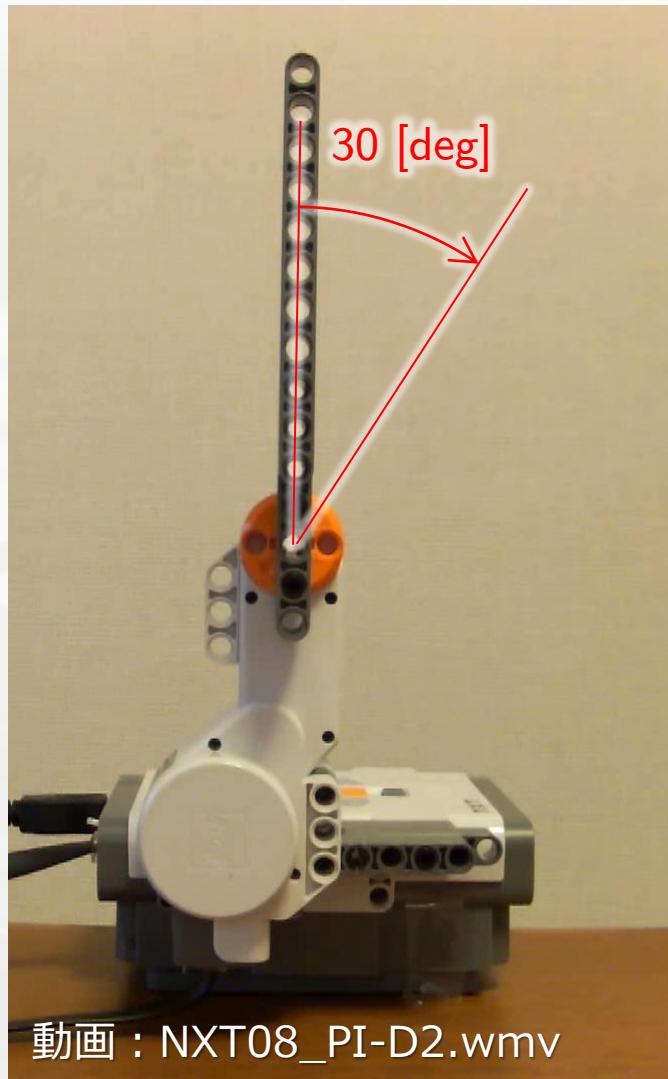
i\_pd\_cont\_c.slx

```
>> h = 0.01; Tf = 0.02;  
>> kP = 3; kD = 0.2;  
>> kI = 20;
```

# I-PD 制御 (比例・微分先行型 PID 制御)

PI-D 制御 :

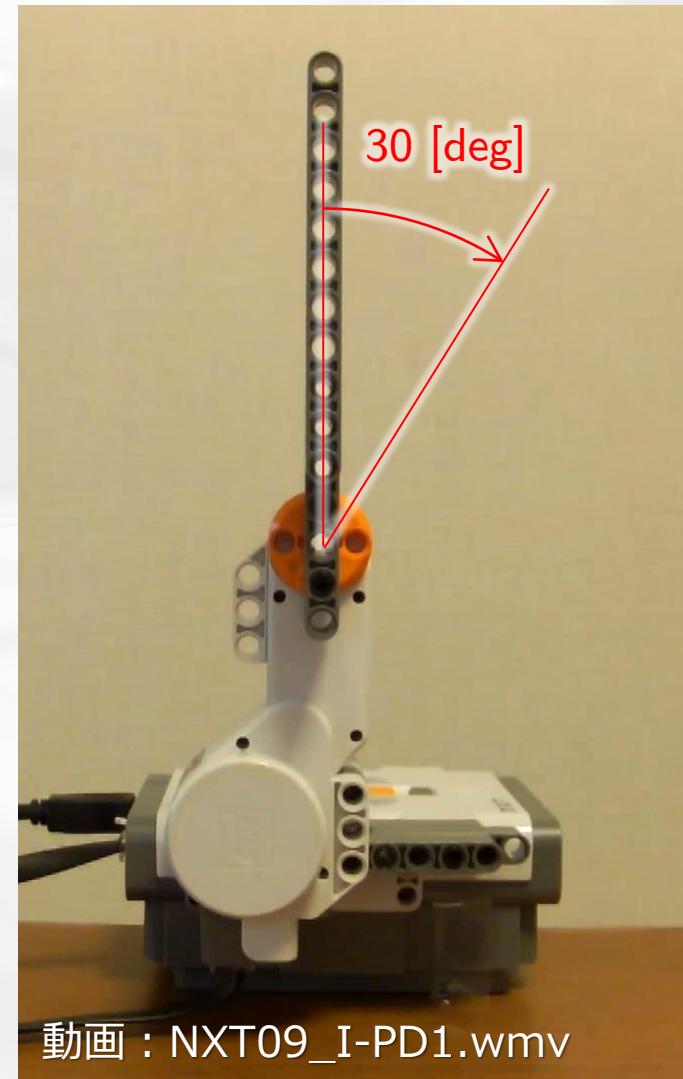
$$k_P = 3, k_D = 0.2, k_I = 20$$



動画 : NXT08\_PI-D2.wmv

I-PD 制御 :

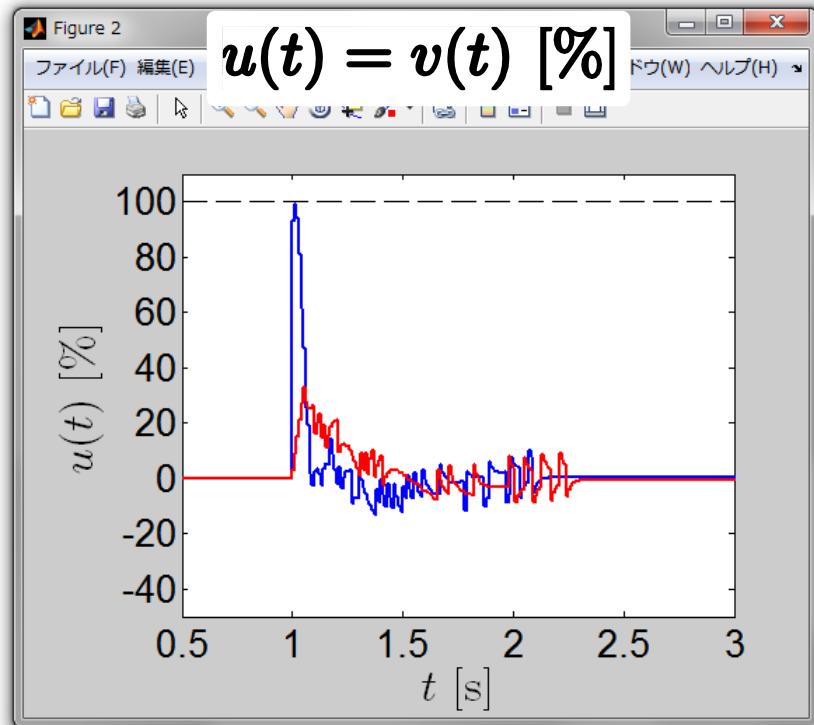
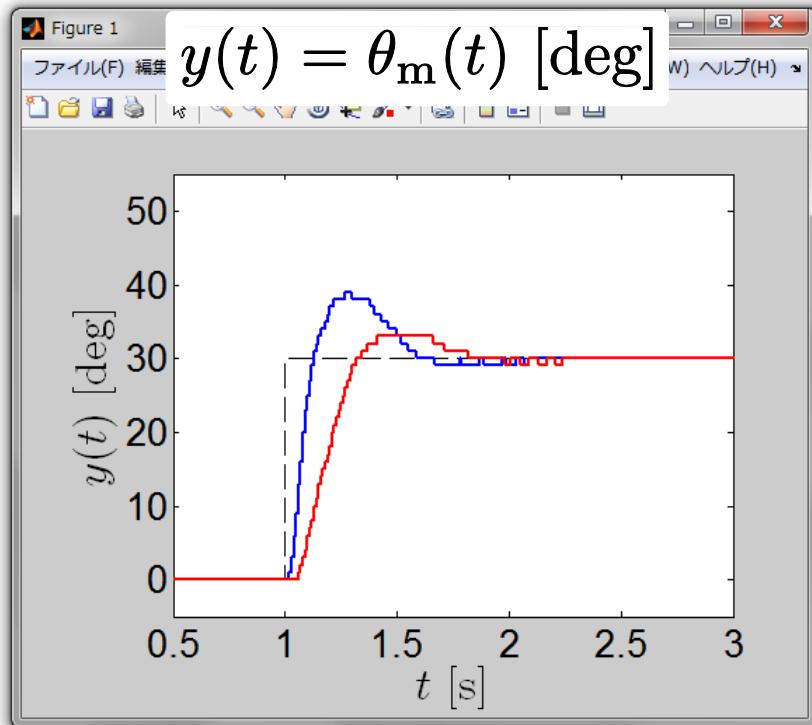
$$k_P = 3, k_D = 0.2, k_I = 20$$



動画 : NXT09\_I-PD1.wmv

# I-PD 制御 (比例・微分先行型 PID 制御)

— PI-D 制御 :  $k_P = 3, k_D = 0.2, k_I = 20$   
— I-PD 制御 :  $k_P = 3, k_D = 0.2, k_I = 20$

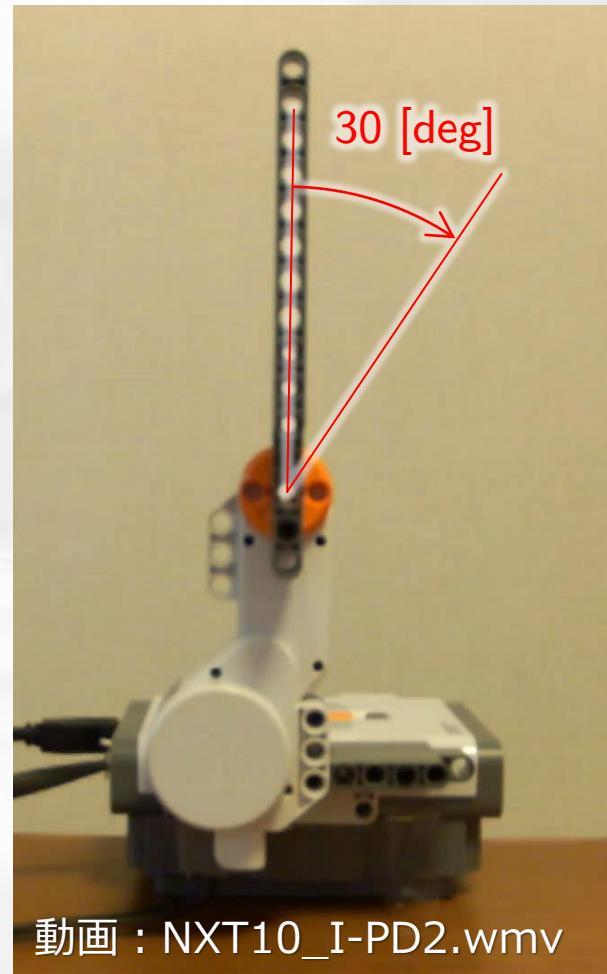
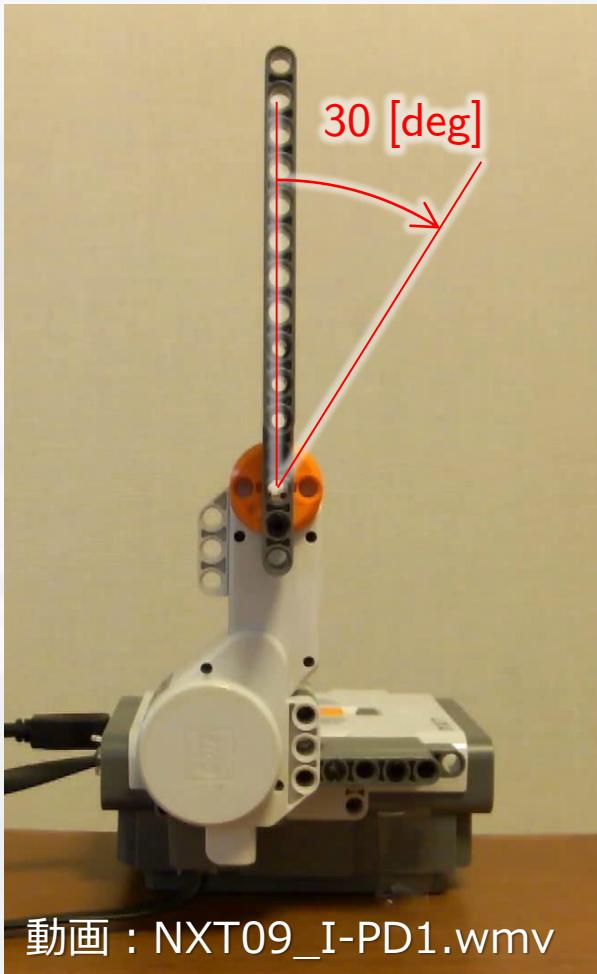


■ I-PD 制御の方がオーバーシュートが小さいが、立ち上がりが遅い

■ I-PD 制御の方が入力が小さい

# I-PD 制御 (比例・微分先行型 PID 制御)

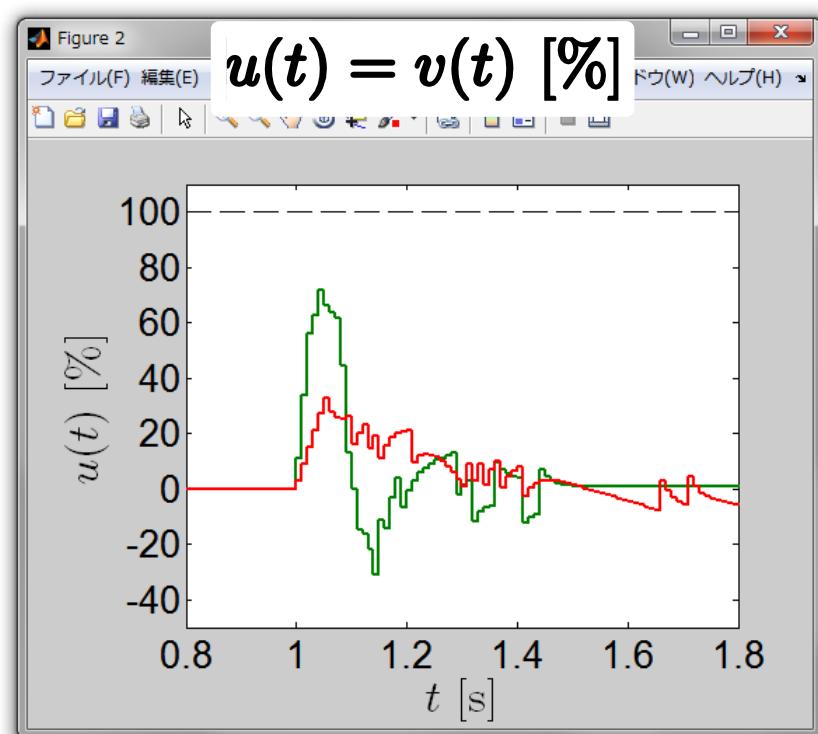
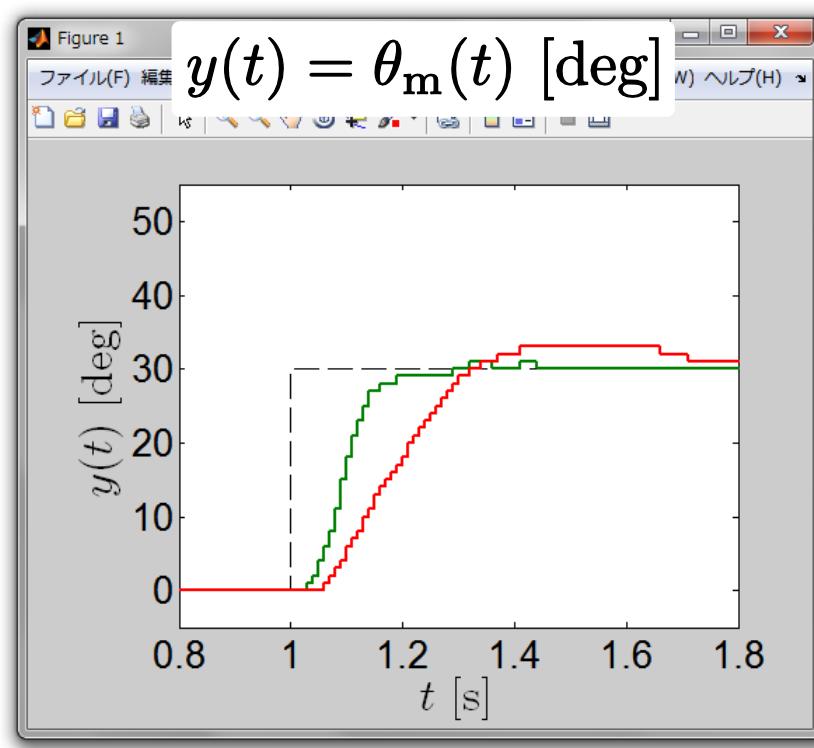
$$k_P = 3, k_D = 0.2, k_I = 20 \quad k_P = 7.5, k_D = 0.2, k_I = 75$$



入力の大きさに余裕があるので、  
ゲインを大きくして速応性を改善

# I-PD 制御 (比例・微分先行型 PID 制御)

- I-PD 制御 :  $k_P = 3, k_D = 0.2, k_I = 20$
- I-PD 制御 :  $k_P = 7.5, k_D = 0.2, k_I = 75$

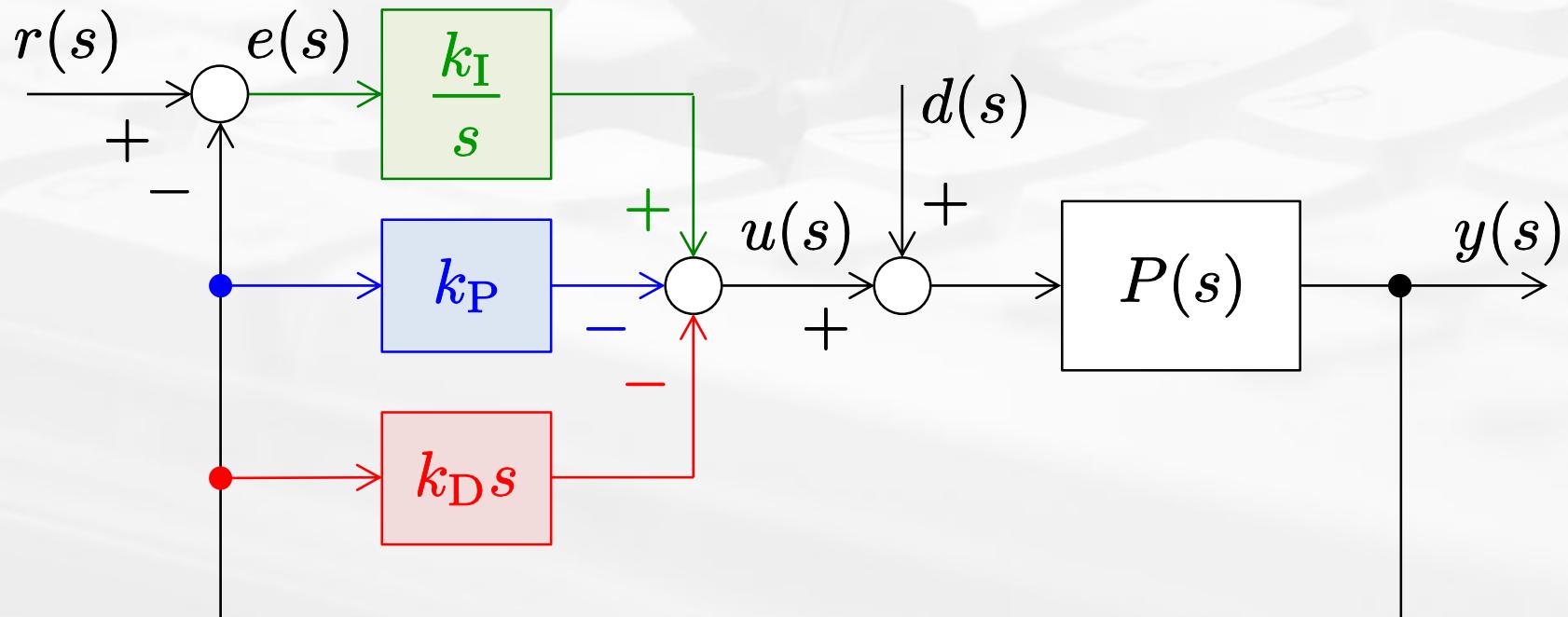


# 理論的な解析

# I-PD 制御 (比例・微分先行型 PID 制御)

$$y(s) = G_{yr}(s)r(s) + G_{yd}(s)d(s)$$

$$\begin{cases} G_{yr}(s) = \frac{b\mathbf{k_I}}{s^3 + (a + b\mathbf{k_D})s^2 + b\mathbf{k_P}s + b\mathbf{k_I}} \\ G_{yd}(s) = \frac{bs}{s^3 + (a + b\mathbf{k_D})s^2 + b\mathbf{k_P}s + b\mathbf{k_I}} \end{cases}$$



# I-PD 制御 (比例・微分先行型 PID 制御)

$$y(s) = G_{yr}(s)r(s) + G_{yd}(s)d(s)$$

$$\left\{ \begin{array}{l} G_{yr}(s) = \frac{bk_I}{s^3 + (a + bk_D)s^2 + bk_P s + bk_I} \\ G_{yd}(s) = \frac{bs}{s^3 + (a + bk_D)s^2 + bk_P s + bk_I} \end{array} \right.$$



零点を持たない



改善!

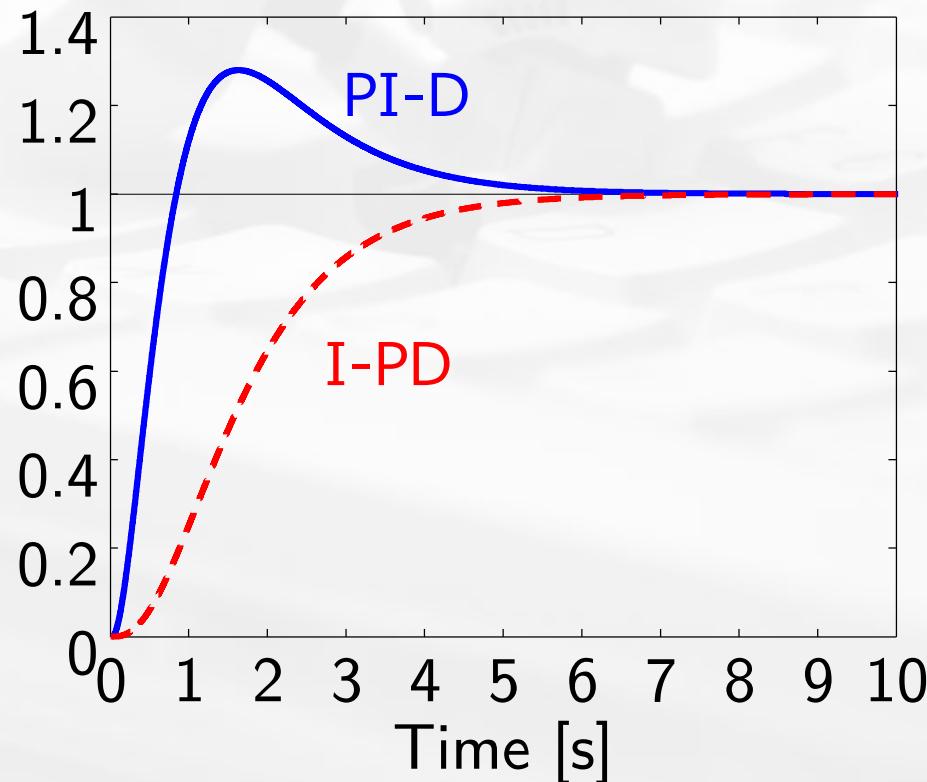


PI-D 制御と同じ

# 零点に起因するオーバーシュート

$$G_{yr}(s) = \frac{6(2s+1)}{(s+1)(s+2)(s+3)}$$
 極 :  $-1, -2, -3$  零点 :  $-1/2$

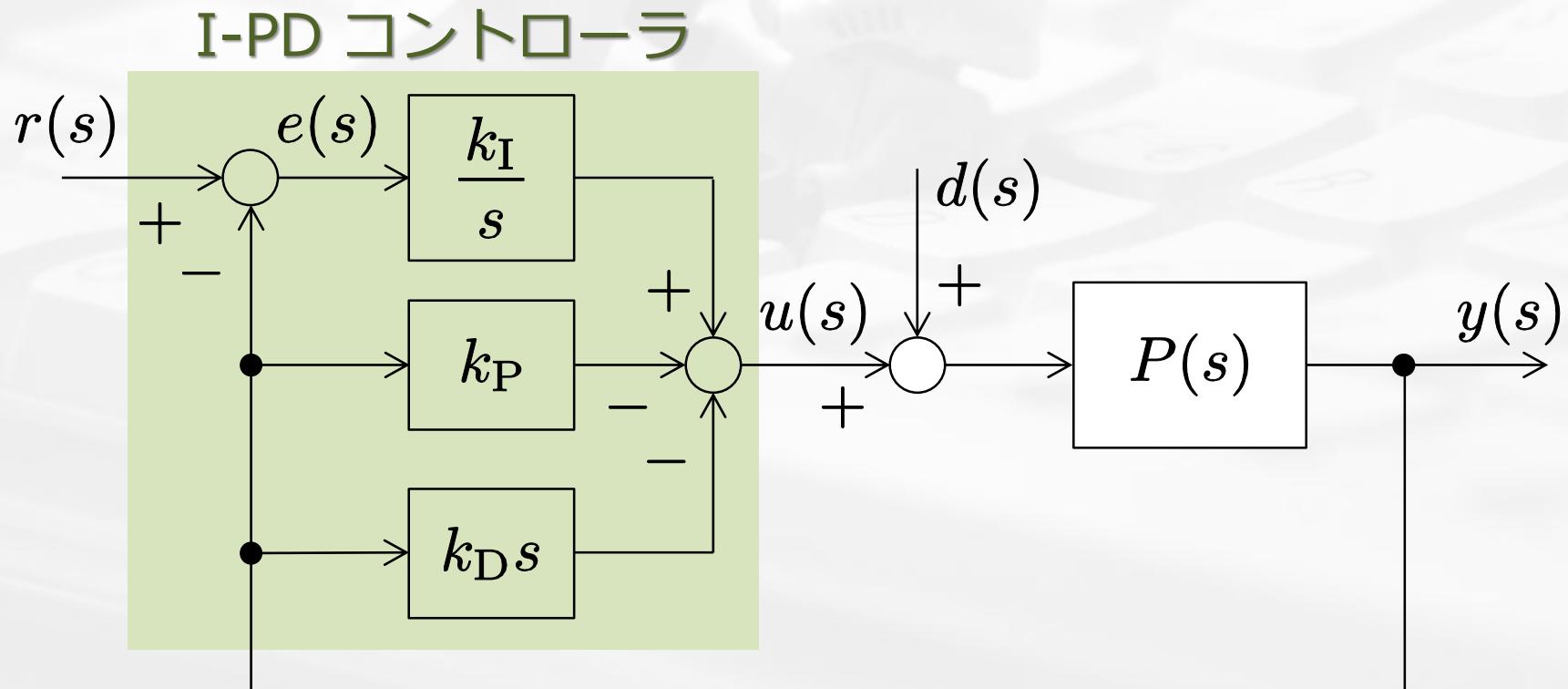
$$G_{yr}(s) = \frac{6}{(s+1)(s+2)(s+3)}$$
 極 :  $-1, -2, -3$  零点 : なし



# I-PD 制御 = 目標値フィルタ + PID 制御

## I-PD コントローラ

$$u(s) = -k_P y(s) + \frac{k_I}{s} e(s) - k_D s y(s)$$

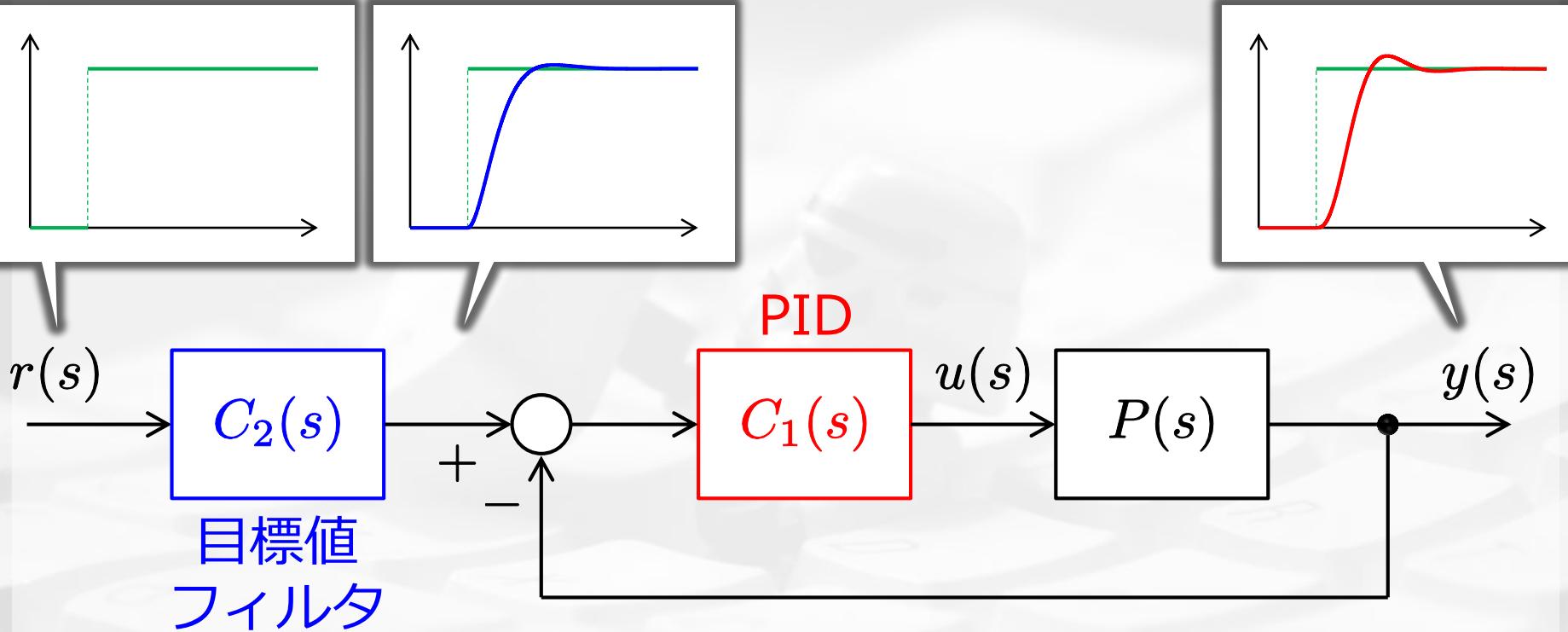


# I-PD 制御 = 目標値フィルタ + PID 制御

## I-PD コントローラ

$$\begin{aligned} u(s) &= -k_P y(s) + \frac{k_I}{s} e(s) - k_D s y(s) \\ &= \frac{k_I}{s} r(s) - \left( k_P + \frac{k_I}{s} + k_D s \right) y(s) \\ &= \frac{k_I}{s} r(s) - \frac{k_D s^2 + k_P s + k_I}{s} y(s) \\ &= \frac{k_D s^2 + k_P s + k_I}{s} \left( \frac{k_I}{k_D s^2 + k_P s + k_I} r(s) - y(s) \right) \\ &= \left( k_P + \frac{k_I}{s} + k_D s \right) \left( \frac{k_I}{k_D s^2 + k_P s + k_I} r(s) - y(s) \right) \\ &\quad C_1(s) \qquad \qquad \qquad C_2(s) \end{aligned}$$

# I-PD 制御 = 目標値フィルタ + PID 制御



$$\left\{ \begin{array}{l} C_1(s) = k_P + \frac{k_I}{s} + k_D s \quad : \text{PID コントローラ} \\ C_2(s) = \frac{k_I}{k_D s^2 + k_P s + k_I} \quad : \text{目標値フィルタ} \\ \qquad \qquad \qquad (2 \text{ 次遅れ要素}) \end{array} \right.$$