This week’s agenda

- **Linear Differential Equations**
- **Obtaining Transfer Functions**
- **Block Diagrams**
- **An Introduction to Stability for Transfer Functions**
- **Concept of Feedback and Closed Loop**
- **Basic Control Actions, P-I-D Effects**

P-2 Linear Differential Equations

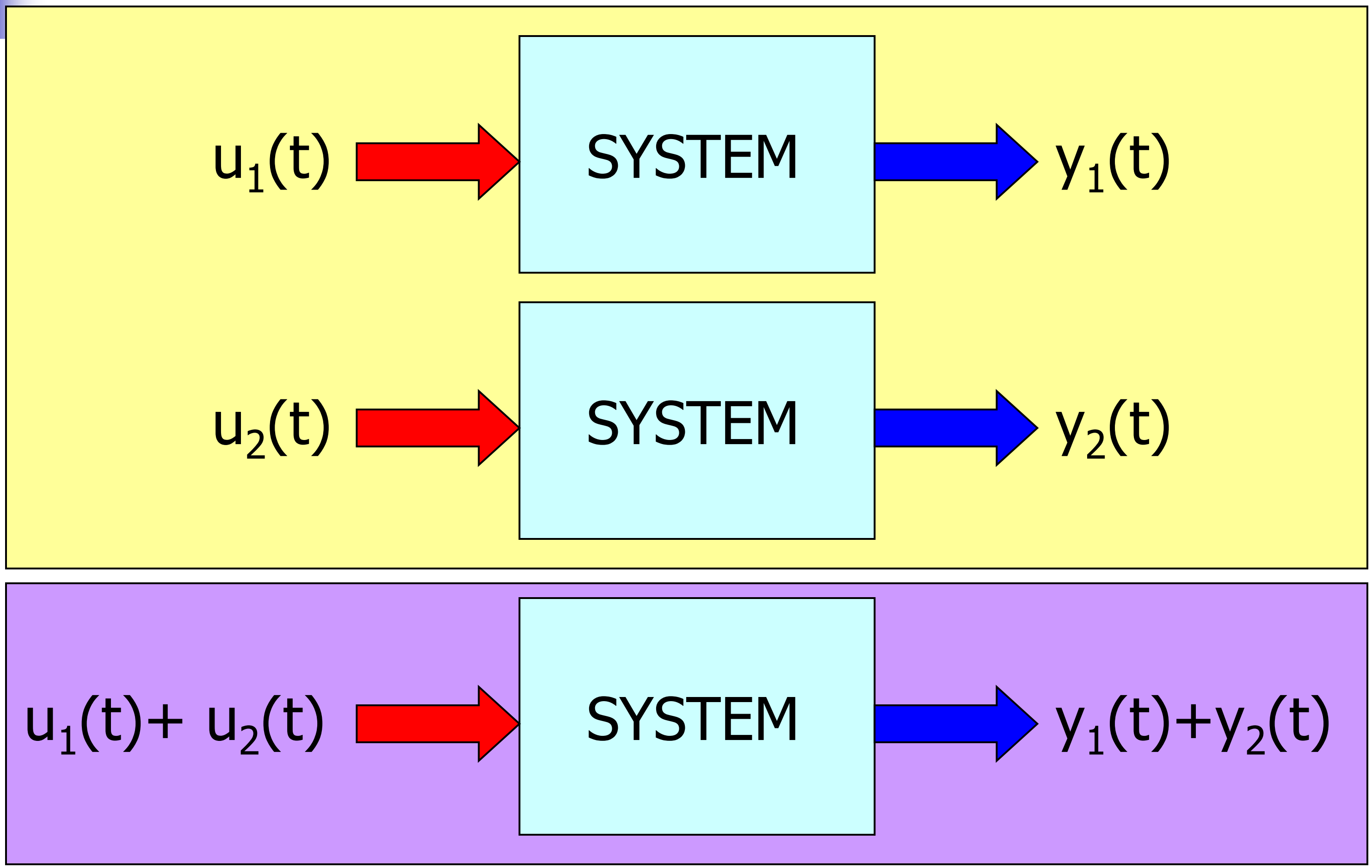
Why do we need differential equations?

- To characterize the dynamics
- To obtain a model (which may not be unique)
- To be able to analyze the behavior

Finally, to be able to design a controller

- Model may depend on your perspective and the goals of the design
- Simplicity versus Accuracy tradeoff arises

When is a dynamics linear?



The system is **linear** if the principle of **superposition** applies

Linear Time Invariant (LTI) Systems
Linear Time Varying (LTV) Systems

A differential equation is linear if the coefficients are constants or functions only of the independent variable (e.g. time below).

$$\frac{d^2x(t)}{dt^2} = a\frac{dx(t)}{dt} + bx(t) + c$$

$$\ddot{x}(t) = a\dot{x}(t) + bx(t) + c$$

→ **LTI**

$$\ddot{x}(t) = a(t)\dot{x}(t) + b(t)x(t) + c(t)$$

→ **LTV**

Nonlinear Systems

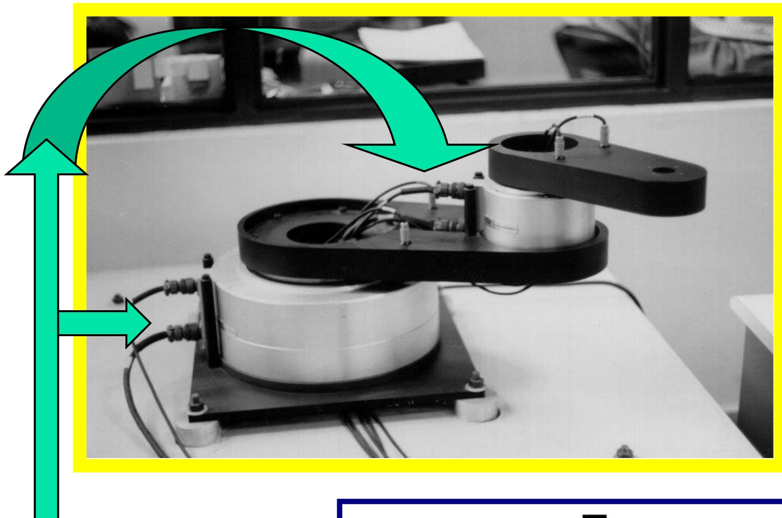
A system is nonlinear if the principle of superposition does not apply

$$\ddot{x} + \dot{x}^2 + x = A\sin\omega t$$

$$\ddot{x} + (x^2 - 1)\dot{x} + x = 0$$

$$\ddot{x} + \dot{x} + x + x^3 = 0$$

A More Realistic Example - 2DOF Robot



Control Inputs

$\underline{u}$

Dynamics is characterized by

$$\boxed{M\left(\underline{x}\right)\ddot{\underline{x}} + \underline{V}\left(\underline{x}, \dot{\underline{x}}\right) = \underline{u} - \underline{f}_{c}}$$

where

$$M\left(\underline{x}\right) = \begin{bmatrix} p_1 + 2p_3\cos\left(x_e\right) & p_2 + p_3\cos\left(x_e\right) \\ p_2 + p_3\cos\left(x_e\right) & p_2 \end{bmatrix}$$

$$V\left(\underline{x}, \dot{\underline{x}}\right) = \begin{bmatrix} -\dot{x}_e\left(2\dot{x}_b + \dot{x}_e\right)p_3\sin\left(x_e\right) \\ \dot{x}_b^2\, p_3\sin\left(x_e\right) \end{bmatrix}$$

Linearization of $z=f(x)$

- Consider $z=f(x)$ is the system
- (x_0, z_0) is the operating point
- Perform Taylor series expansion around the operating point

Only if these terms are negligibly small!

$$z = f(x_0) + \frac{df}{dx}(x - x_0) + \boxed{\frac{1}{2!}\frac{d^2 f}{dx^2}(x - x_0)^2 + \cdots}$$

$$z \cong z_0 + K(x - x_0) \text{ where } K = \left(\frac{df}{dx}\right)_{x=x_0}$$

Linearization of $z=f(x,y)$

- Consider $z=f(x,y)$ is the system
- (x_0,y_0,z_0) is the operating point
- Perform Taylor series expansion around the operating point

Only if these terms are negligibly small!

$$z = f(x_0, y_0) + \frac{\partial f}{\partial x}(x - x_0) + \frac{\partial f}{\partial y}(y - y_0)$$

$$+\frac{1}{2!}\left(\frac{\partial^2 f}{\partial x^2}(x-x_0)^2 + 2\frac{\partial^2 f}{\partial x \partial y}(x-x_0)(y-y_0) + \frac{\partial^2 f}{\partial y^2}(y-y_0)^2\right) + \cdots$$

$$z \cong z_0 + K_1(x - x_0) + K_2(y - y_0) \text{ where } K_1 = \left(\frac{\partial f}{\partial x}\right)_{\substack{x=x_0 \\ y=y_0}} \text{ and } K_2 = \left(\frac{\partial f}{\partial y}\right)_{\substack{x=x_0 \\ y=y_0}}$$

P-2 Obtaining Transfer Functions

Consider the system, whose dynamics is given by the following differential equation

$$\boxed{\begin{aligned} &a_0 x^{(n)} + a_1 x^{(n-1)} + \cdots + a_{n-1}\dot{x} + a_n x \\ &= b_0 u^{(m)} + b_1 u^{(m-1)} + \cdots + b_{m-1}\dot{u} + b_m u \end{aligned}}$$

where, x is the output, u is the input

Assume all initial conditions are zero and take the Laplace transform. Remember

Real Differentiation

$$\boxed{L\{\frac{d}{dt}f(t)\} = sF(s) - f(0)}$$

$$\boxed{L\{f(t)\} = F(s)}$$

$$\boxed{\begin{aligned} &a_0 s^n X(s) + a_1 s^{n-1} X(s) + \cdots + a_{n-1} s X(s) + a_n X(s) \\ &= b_0 s^m U(s) + b_1 s^{m-1} U(s) + \cdots + b_{m-1} s U(s) + b_m U(s) \end{aligned}}$$

We get

$$\boxed{\begin{aligned}&\left(a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n\right) X(s)\\ &\quad = \left(b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1} s + b_m\right) U(s)\end{aligned}}$$

$$\boxed{\frac{X(s)}{U(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n} = G(s)}$$

Transfer function is G(s)

Note that while studying with transfer functions all initial conditions are assumed to be zero

What does a transfer function tell us?

Transfer function (TF)

$$\boxed{\frac{X(s)}{U(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1}s + b_m}{a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1}s + a_n} = G(s)}$$

- TF states the relation between input and output
- TF is a property of system, no matter what the input is
- TF does not tell anything about the structure of the system
- TF enables us to understand the behavior of the system
- TF can be found experimentally by studying the response of the system for various inputs
- TF is the Laplace transform of g(t), the impulse response of the system

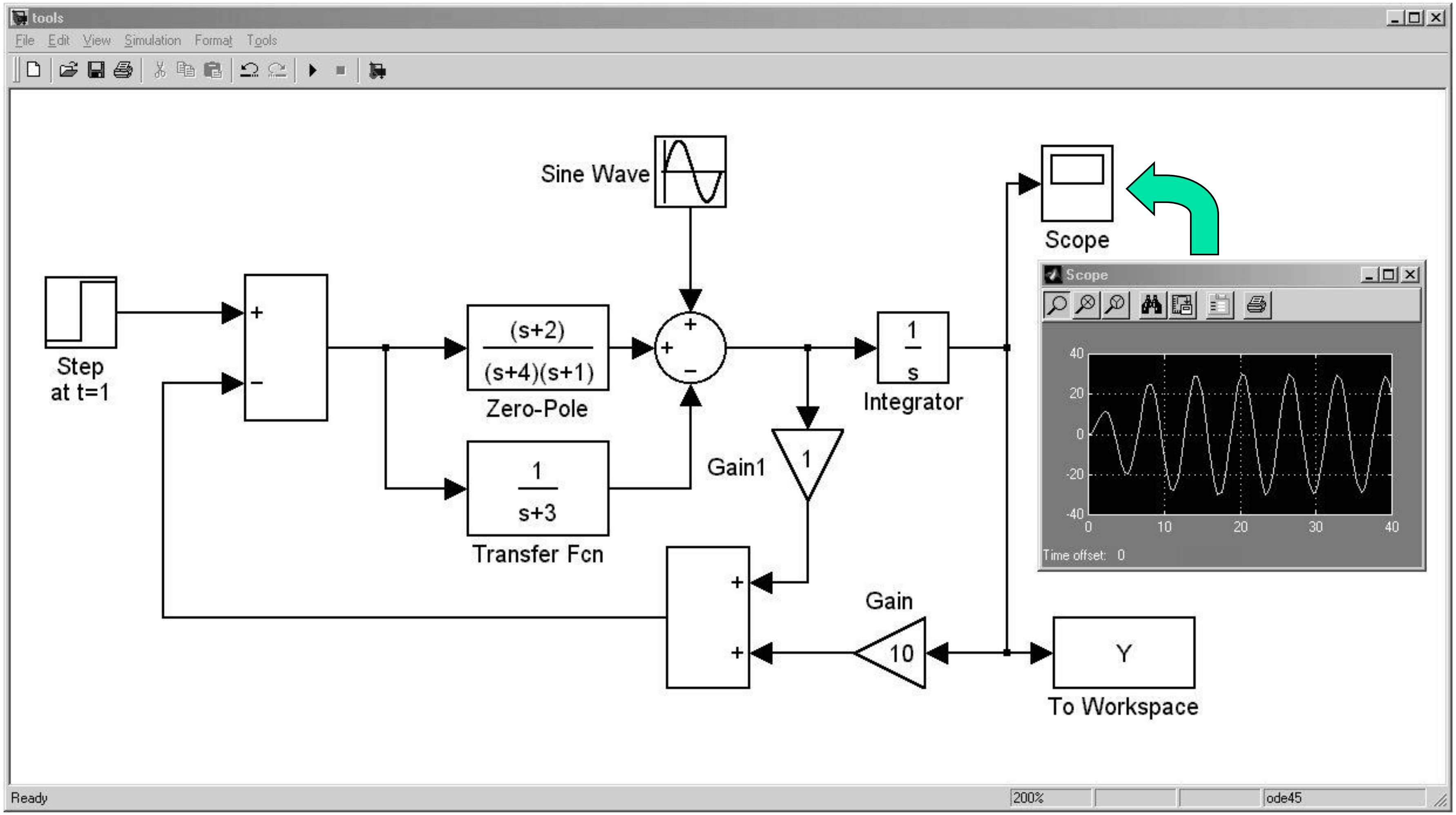
Transfer function (TF)

$$\frac{X(s)}{U(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \cdots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \cdots + a_{n-1} s + a_n} = G(s)$$

- Above TF, i.e. $G(s)$, is n^{th} order
- We assume that $n \geq m$
- If $a_0=1$, the denominator polynomial is said to be monic
- If $b_0=1$, the numerator polynomial is said to be monic

P-2 Block Diagrams

Tools we will mainly use (Matlab-Simulink)



P-2 An Introduction to Stability for Transfer Functions

Consider

$$T(s)=\frac{N(s)}{D(s)}=\frac{b_0 s^m + b_1 s^{m-1} + \cdots + b_m}{s^n + a_1 s^{n-1} + \cdots + a_n}\begin{array}{l}\rightarrow \text{Numerator}\\ \rightarrow \text{Denominator}\end{array}$$

Rewrite this as

$$T(s)=\frac{N(s)}{D(s)}=\frac{b_0\prod_{i=1}^{m}(s+z_i)}{\prod_{i=1}^{n}(s+p_i)}$$

$s=-z_i$ **for** i**=1,2,…,**m **are the zeros of the system**

$s=-p_i$ **for** i**=1,2,…,**n **are the poles of the system**

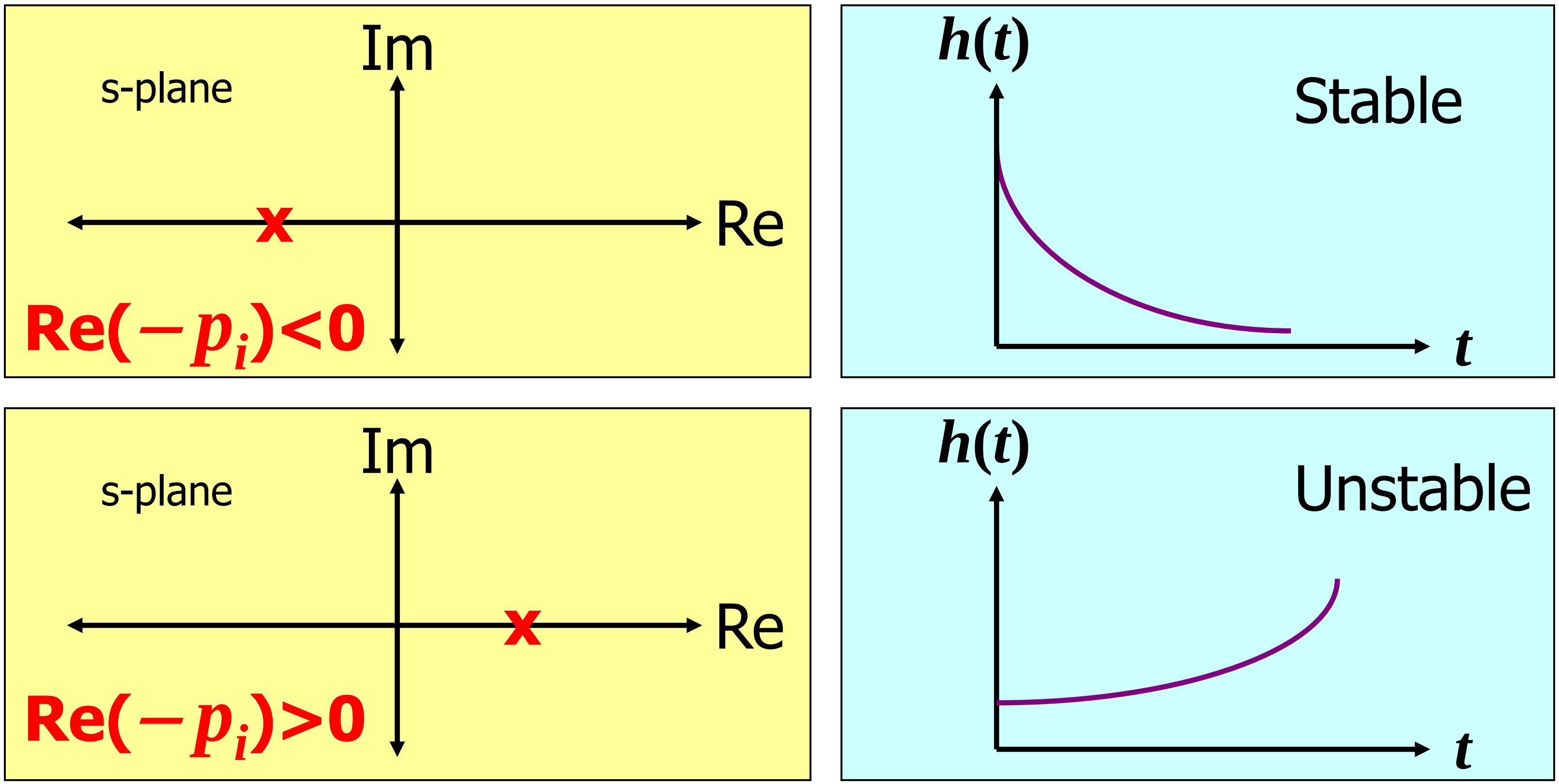
Stability in terms of TF poles

If the real parts of the poles are negative, then the transfer function is stable

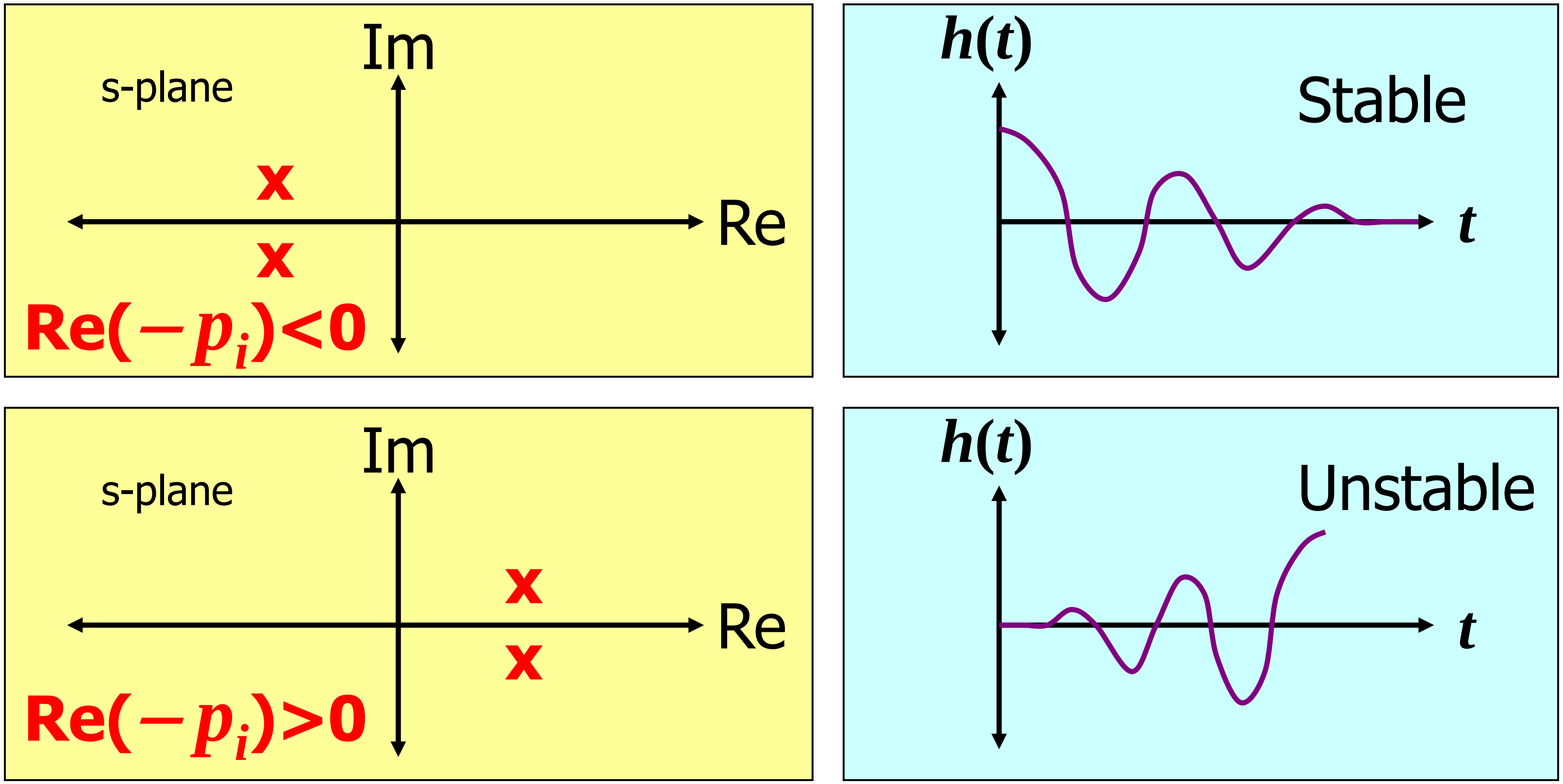
$\text{Re}(-p_i)<0 \Longleftrightarrow \text{Re}(p_i)>0 \Longleftrightarrow$ TF Stable

$$\boxed{T(s)=\frac{N(s)}{D(s)}=\frac{b_0\prod_{i=1}^{m}(s+z_i)}{\prod_{i=1}^{n}(s+p_i)}}$$

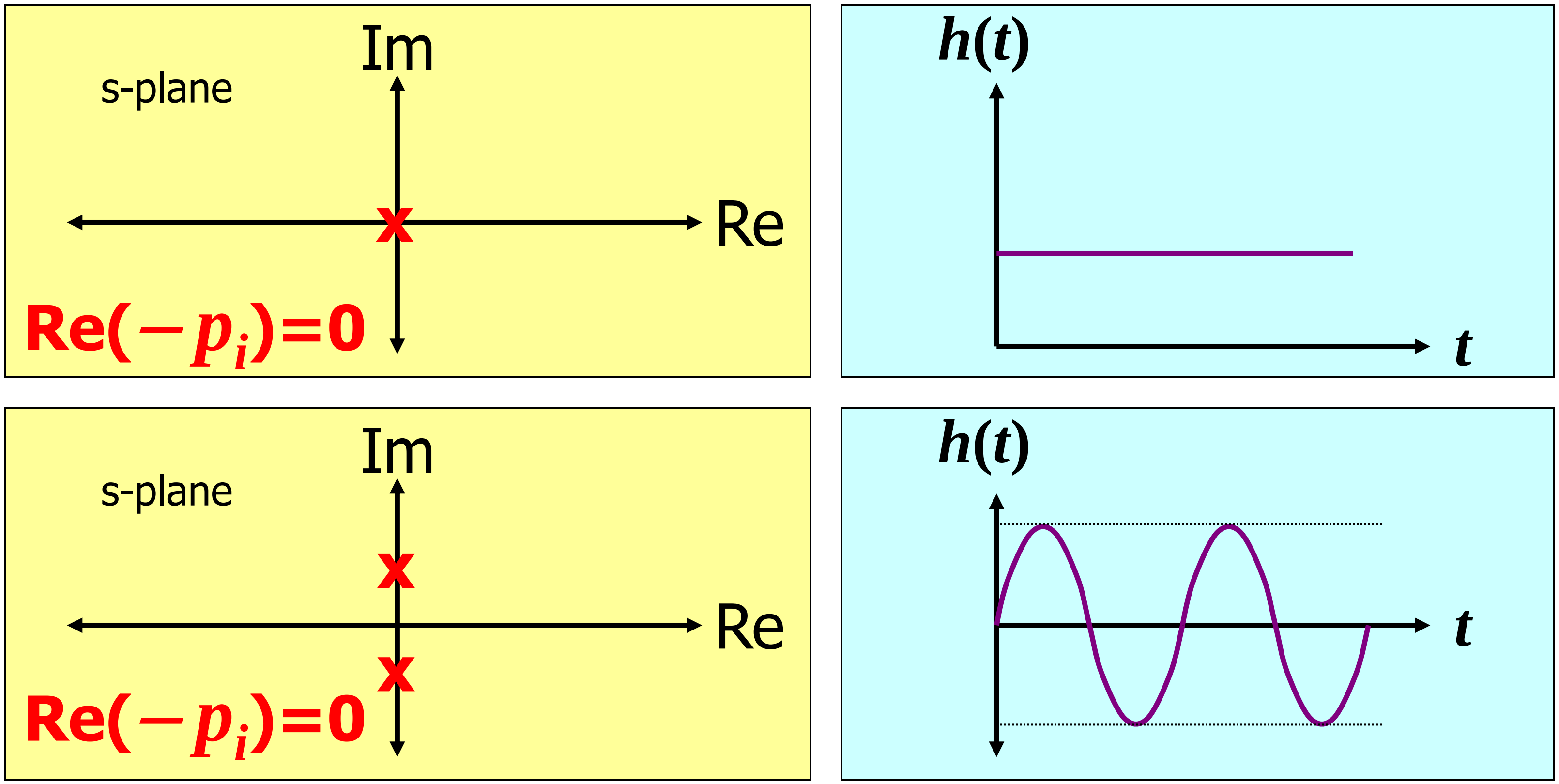
What is the meaning of this?
Poles with zero imaginary parts



What is the meaning of this?
Poles with nonzero imaginary parts



What is the meaning of this? Poles on the imaginary axis



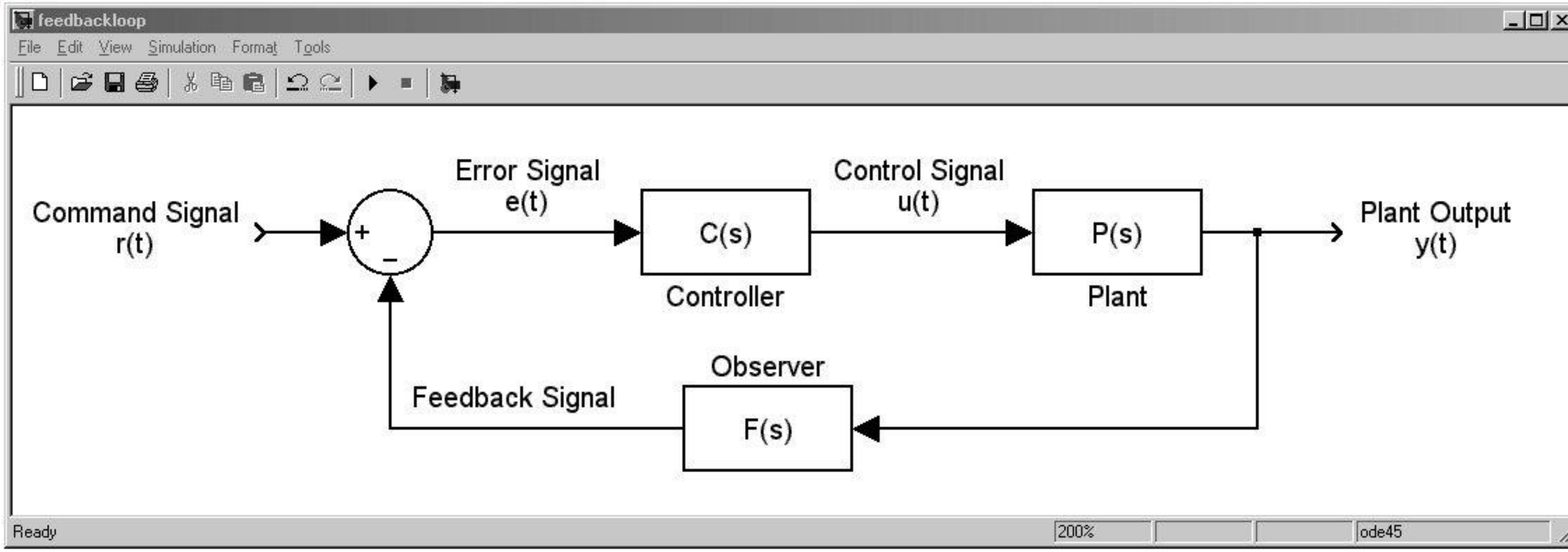
Neither stable nor unstable

A TF is said to be stable if all the roots of the denominator have negative real parts

Poles determine the stability of a TF

Zeros may be stable or unstable as well, but the stability of the TF is determined by the poles

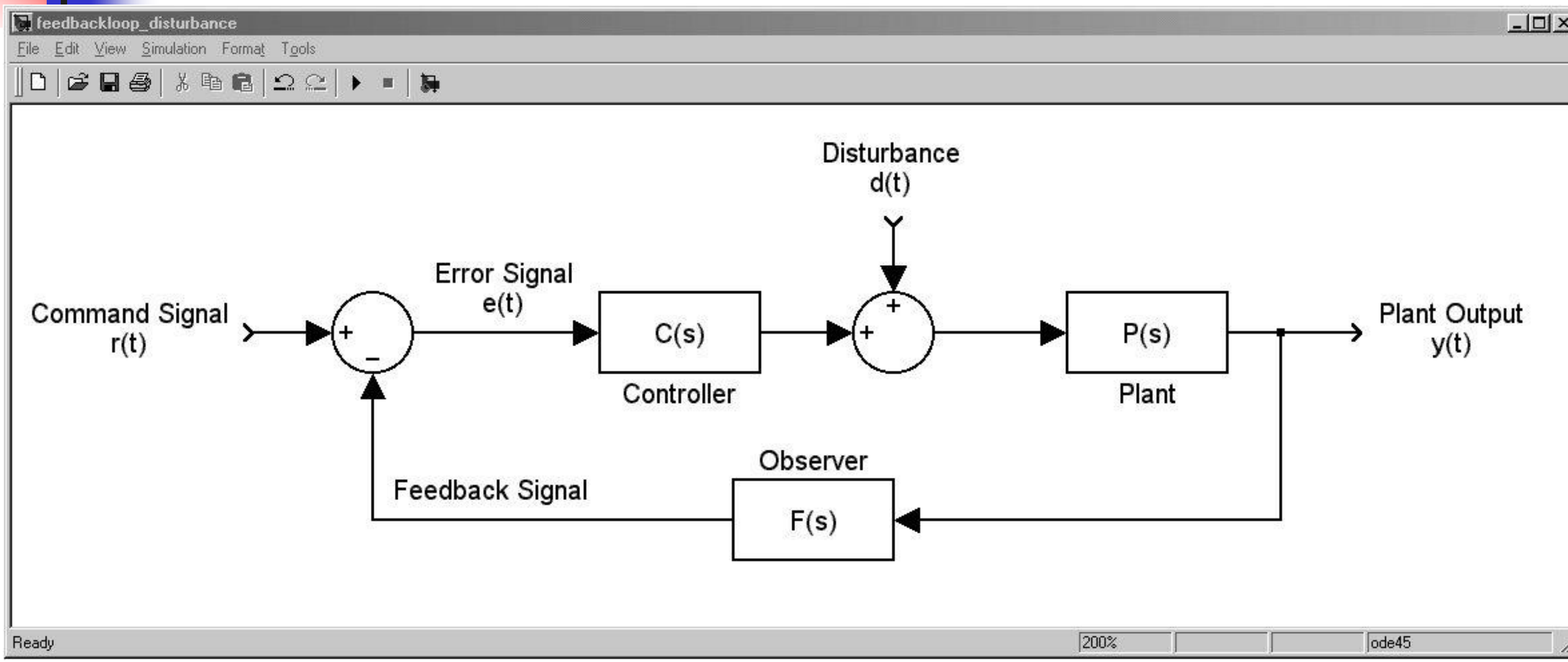
P-2 Concept of Feedback and Closed Loop



$$\frac{Y(s)}{R(s)} = \frac{P(s)C(s)}{1+P(s)C(s)F(s)} \Leftrightarrow Y(s) = \frac{P(s)C(s)}{1+P(s)C(s)F(s)} R(s)$$

$$\frac{E(s)}{R(s)} = \frac{1}{1+P(s)C(s)F(s)} \Leftrightarrow E(s) = \frac{R(s)}{1+P(s)C(s)F(s)}$$

What are the advantages of feedback?

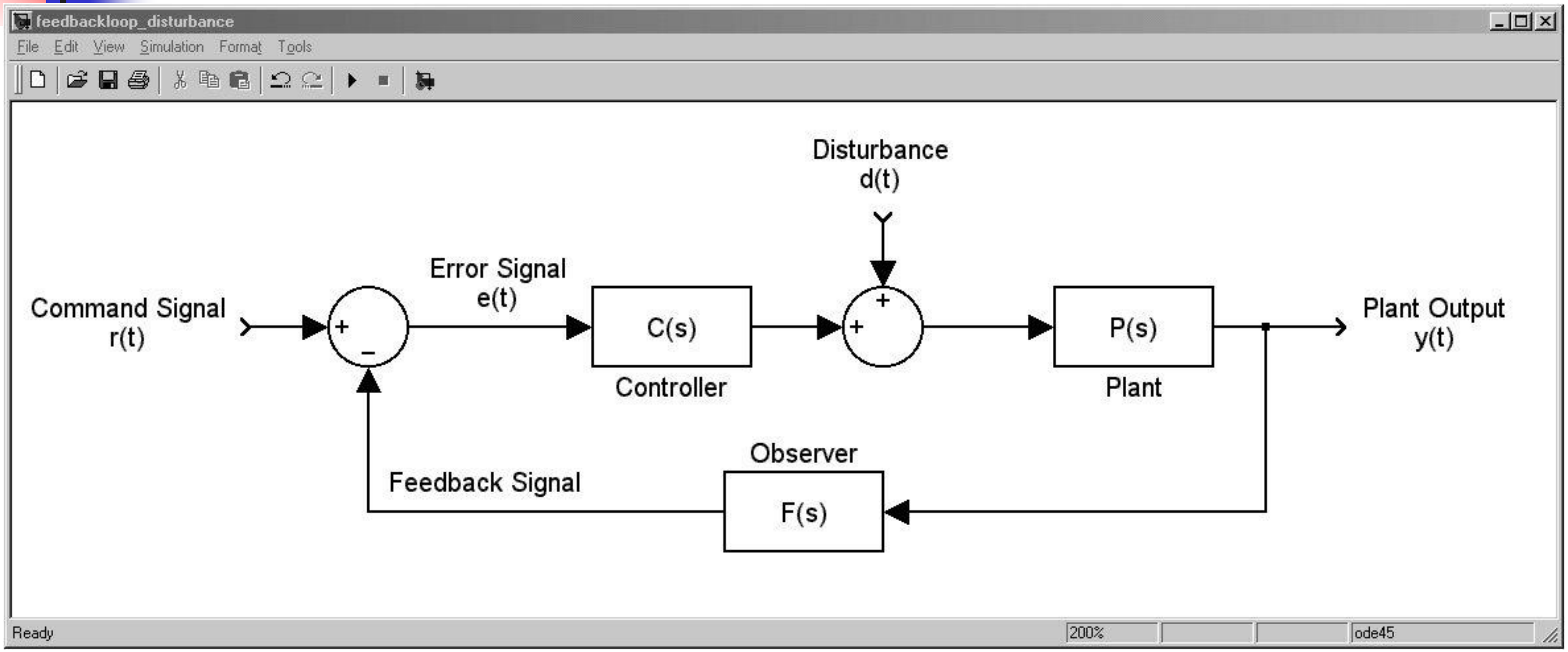


$$T_R(s) = \frac{Y_R(s)}{R(s)} = \frac{P(s)C(s)}{1+P(s)C(s)F(s)}$$

$$T_D(s) = \frac{Y_D(s)}{D(s)} = \frac{P(s)}{1+P(s)C(s)F(s)}$$

Superposition!

$$Y(s) = Y_R(s) + Y_D(s)$$



$$Y(s) = \frac{P(s)}{1 + P(s)C(s)F(s)}\left[C(s)R(s) + D(s)\right]$$

$$Y(s) = \frac{P(s)}{1+P(s)C(s)F(s)}[C(s)R(s)+D(s)]$$

$$\left.\begin{aligned} |C(s)F(s)| &>> 1 \\ |P(s)C(s)F(s)| &>> 1 \end{aligned}\right\} \Rightarrow \begin{aligned} \frac{Y_D(s)}{D(s)} &\approx 0 \\ \frac{Y_R(s)}{R(s)} &\approx \frac{1}{F(s)} \end{aligned}$$

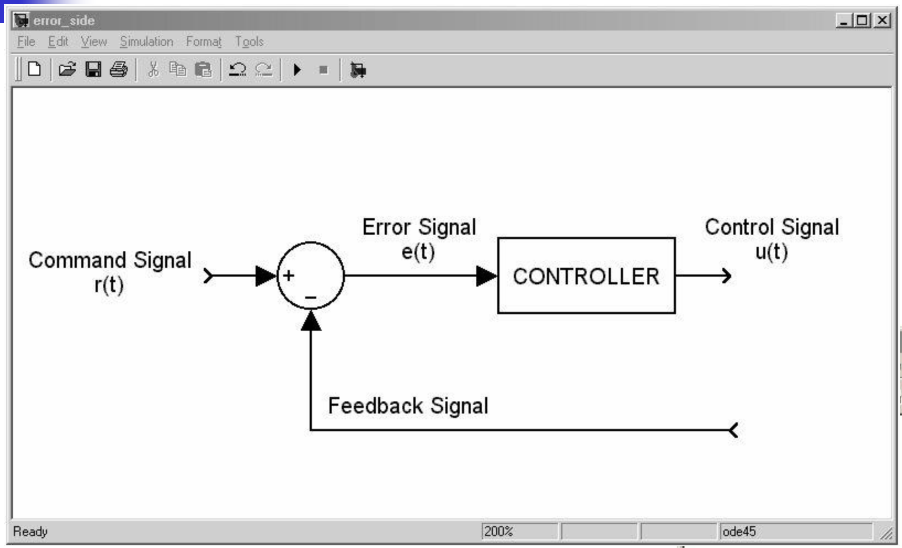
Effect of disturbance is suppressed considerably

Variations on P(s) and C(s) do not affect the closed loop TF. Think about the case when F(s)=1

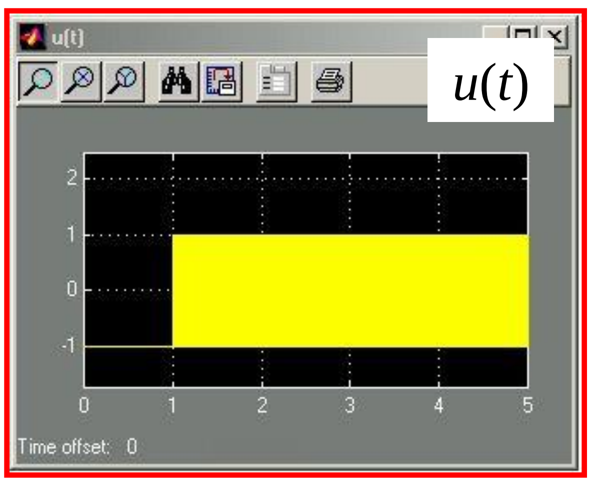
P-2 Basic Control Actions, P-I-D Effects

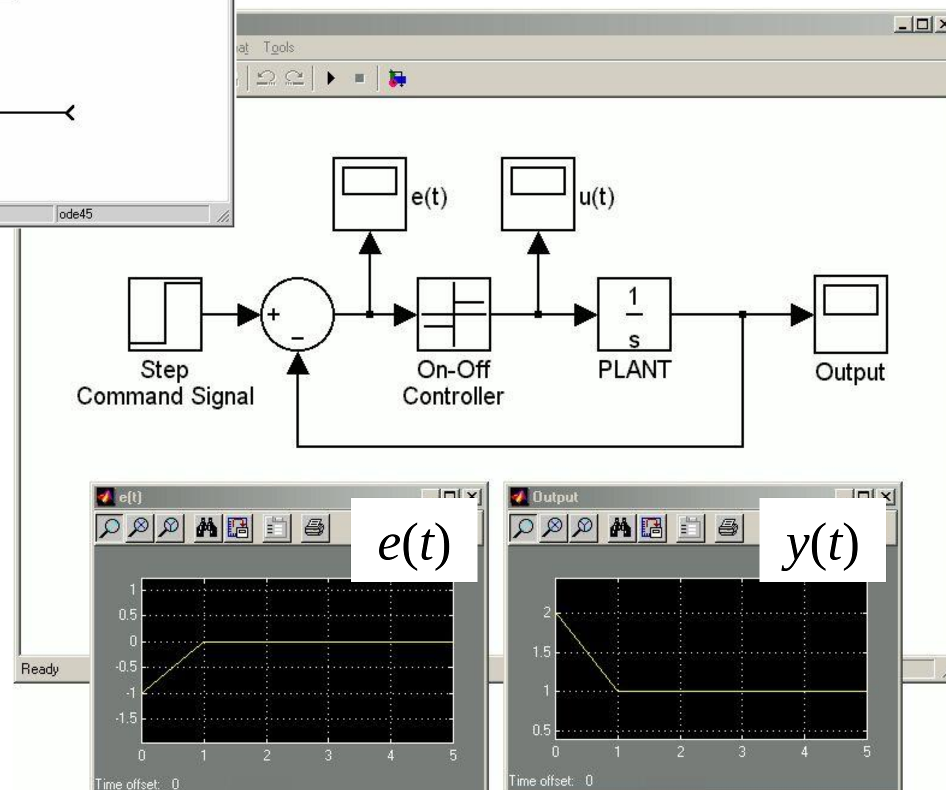
- **On-Off Controller**
- **Proportional (P) Controller**
- **Integral (I) Controller**
- **Proportional-Integral (PI) Controller**
- **Proportional-Derivative (PD) Controller**
- **Proportional-Integral-Derivative (PID) Controller**

On-Off Controller



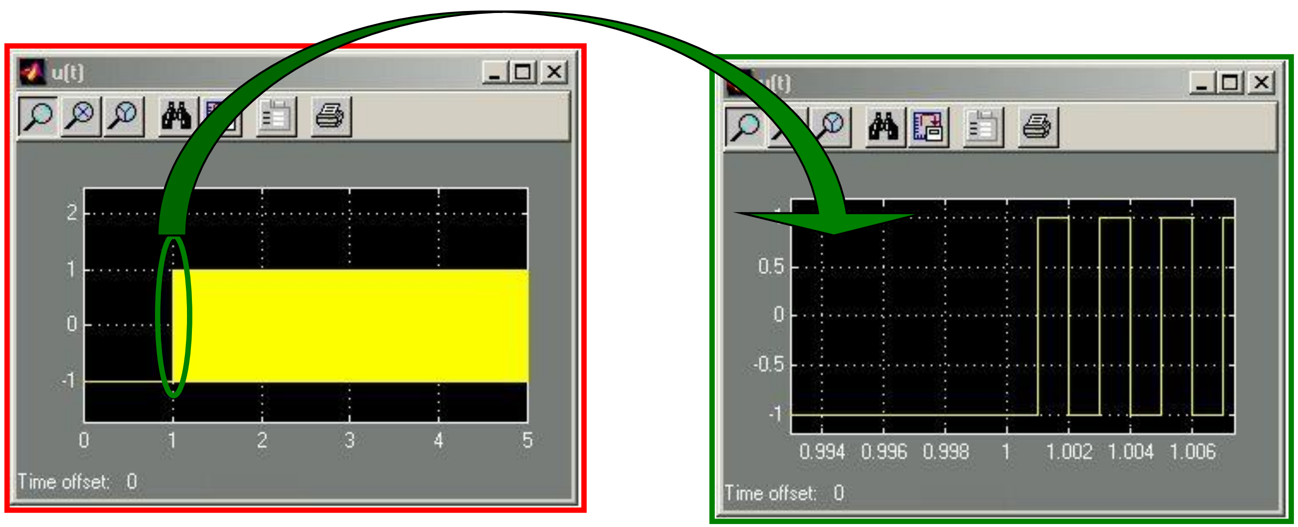
$$u(t) = \begin{cases} U_1 & if\ e(t) > 0 \\ U_2 & if\ e(t) < 0 \end{cases}$$





On-Off Controller

- Initial condition was 2.0
- Reference input was 1.0 (step input at t=0)
- Calculated error = -1.0 ($0 \leq t < 1$)
- Apply minimum (U_2) control value ($U_1=1$, $U_2=-1$)
- This would bring the output to 1 (Command Sig.)
- Around zero output, what is your control?

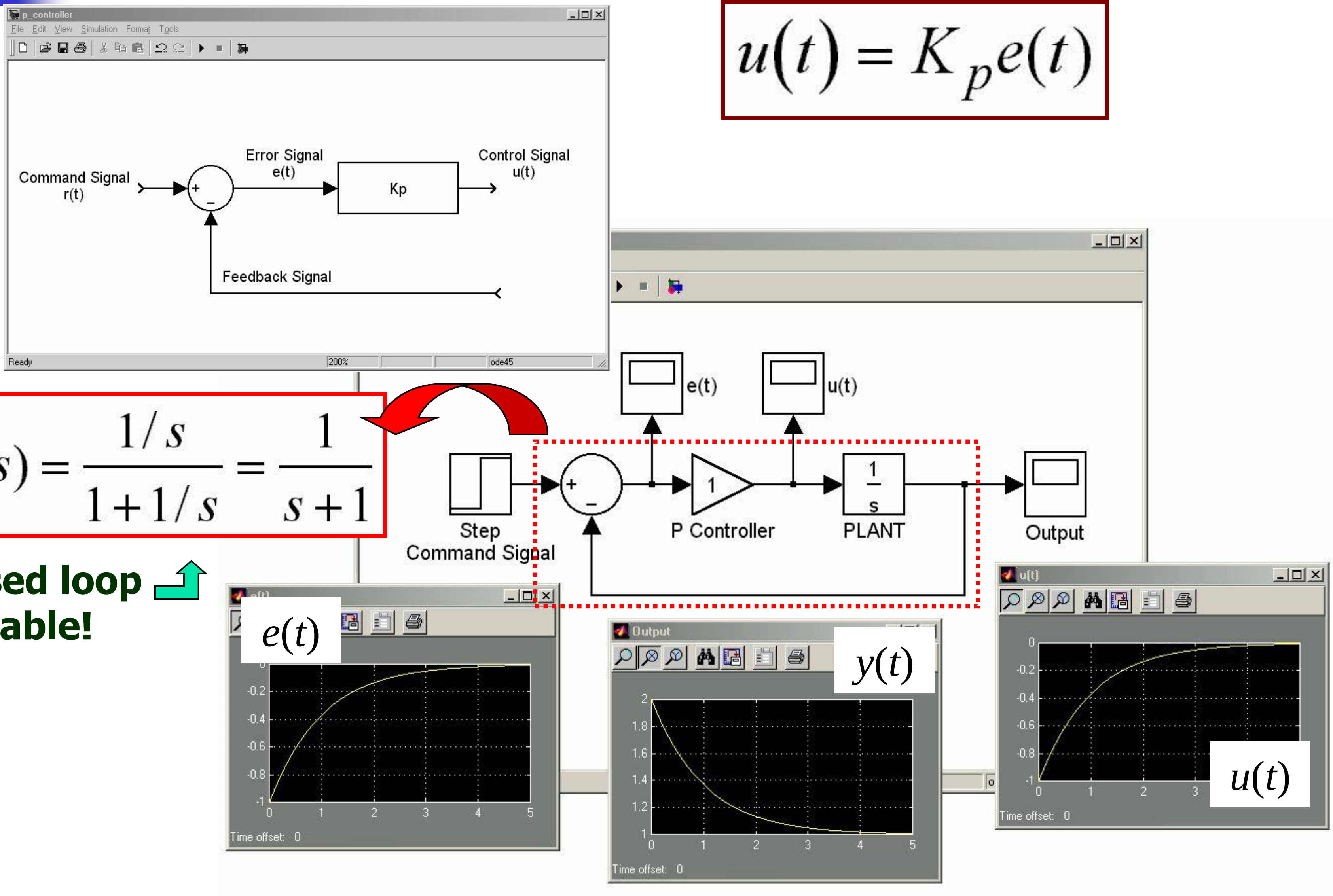


On-Off Controller
Remarks on Simulation

- Ideally, the switching frequency is infinity!
- Simulation step size was 1 msec
- This example shows how On-Off type controller works

Proportional (P) Controller

$$u(t) = K_p e(t)$$



$$T(s) = \frac{1/s}{1+1/s} = \frac{1}{s+1}$$

Closed loop is stable!

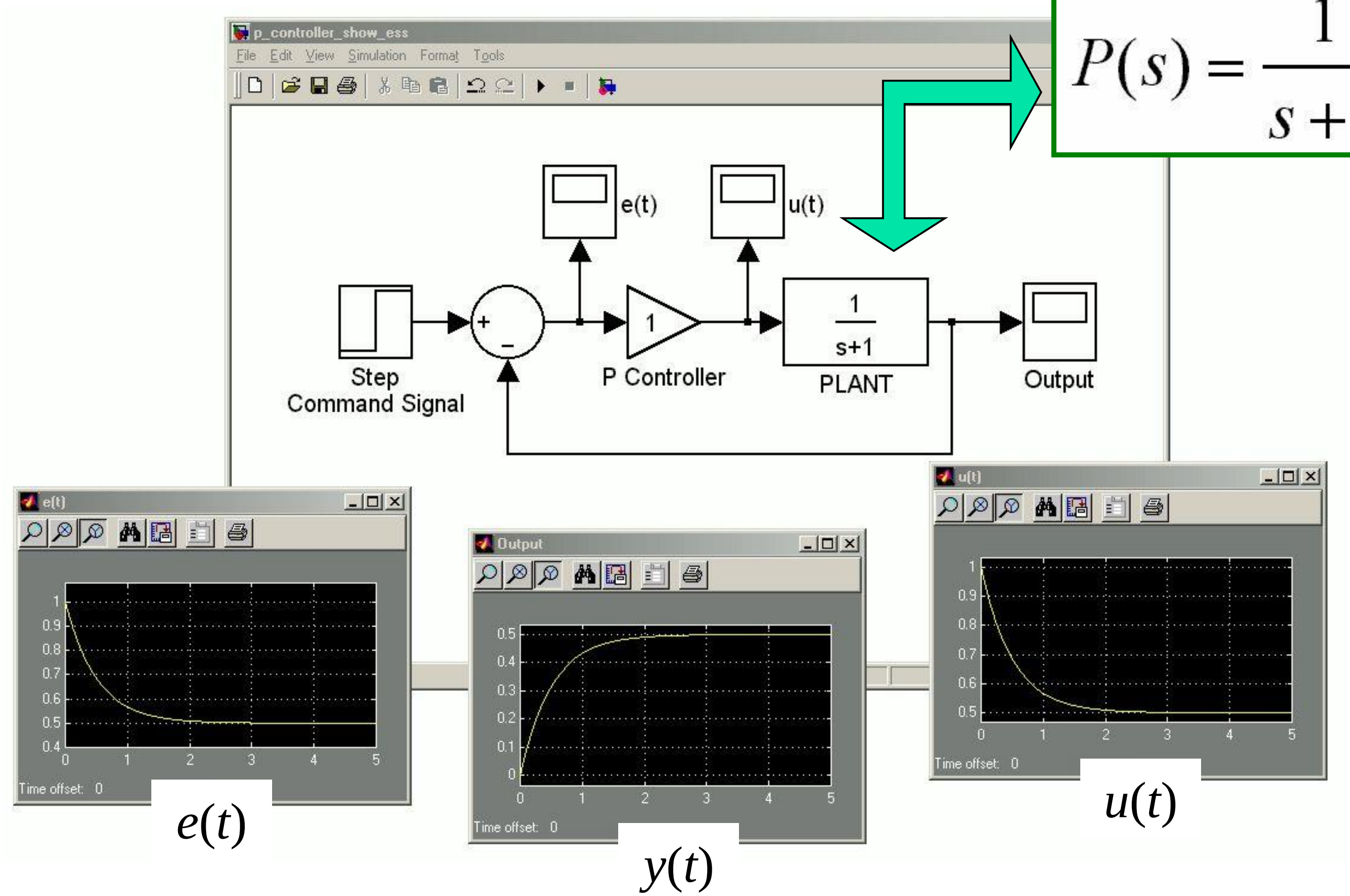
Proportional (P) Controller
Remarks

- Initial condition was 2.0
- Reference input was 1.0 (step input at t=0)
- Calculated error converges to zero
- This would bring the output to 1 (Command Sig.)

Integral (I) Controller

First see what P controller performs with the plant

$$P(s) = \frac{1}{s+1}$$



$e(t)$ $y(t)$ $u(t)$

Here is what happened inside...

When is this stable?

$$T(s)=\frac{P(s)C(s)}{1+P(s)C(s)}=\frac{K_p}{s+1+K_p}$$

$$Y(s)=\frac{K_p}{s+1+K_p}R(s),\quad R(s)=\frac{1}{s}$$

$$\lim_{t\to\infty} y(t)=\lim_{s\to 0} sY(s)=\frac{K_p}{1+K_p}$$

$$\lim_{t\to\infty} y(t)=\frac{1}{2}\ \text{for}\ K_p=1$$

PLANT

$$\boxed{P(s)=\frac{1}{s+1}}$$

CONTROLLER

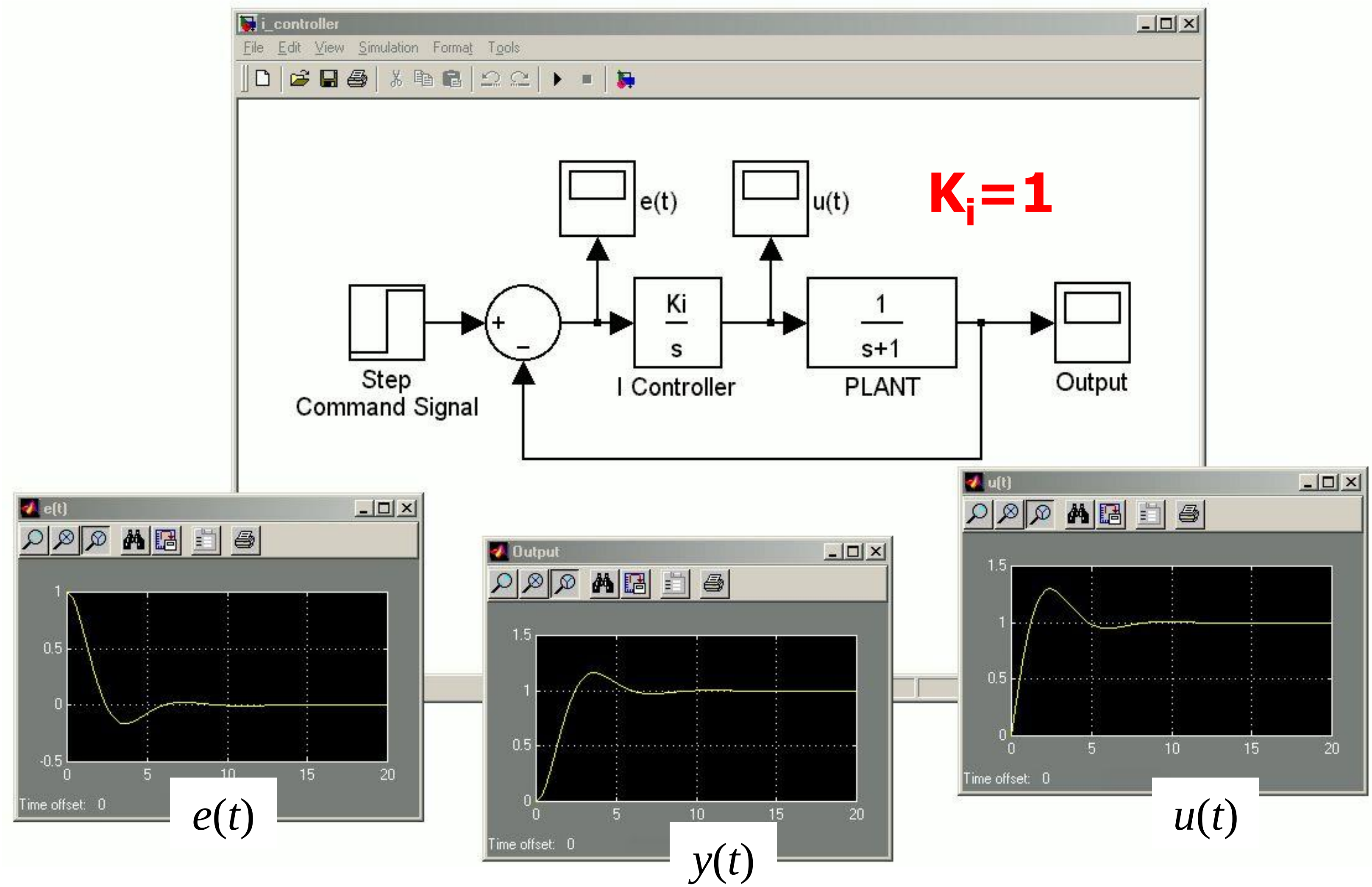
$$\boxed{C(s)=1}$$

$\neq$ 1
Steady State Error!

Integral (I) Controller

- **When there is steady state error, integral action is required**
- **Transfer functions having no integrator (no pole at s=0) would output steady state error to step input**
- **We will turn back to this later… Now consider the same simulation with $C(s)=K_i/s$ (Set $K_i=1$)**

Integral (I) Controller



$e(t)$ $y(t)$ $u(t)$

Integral (I) Controller

When is this TF stable?

$$T(s) = \frac{P(s)C(s)}{1+P(s)C(s)} = \frac{K_i}{s^2+s+K_i}$$

$$Y(s) = \frac{K_i}{s^2+s+K_i} R(s), \quad R(s) = \frac{1}{s}$$

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = 1 \qquad r(t) = 1$$

Note that

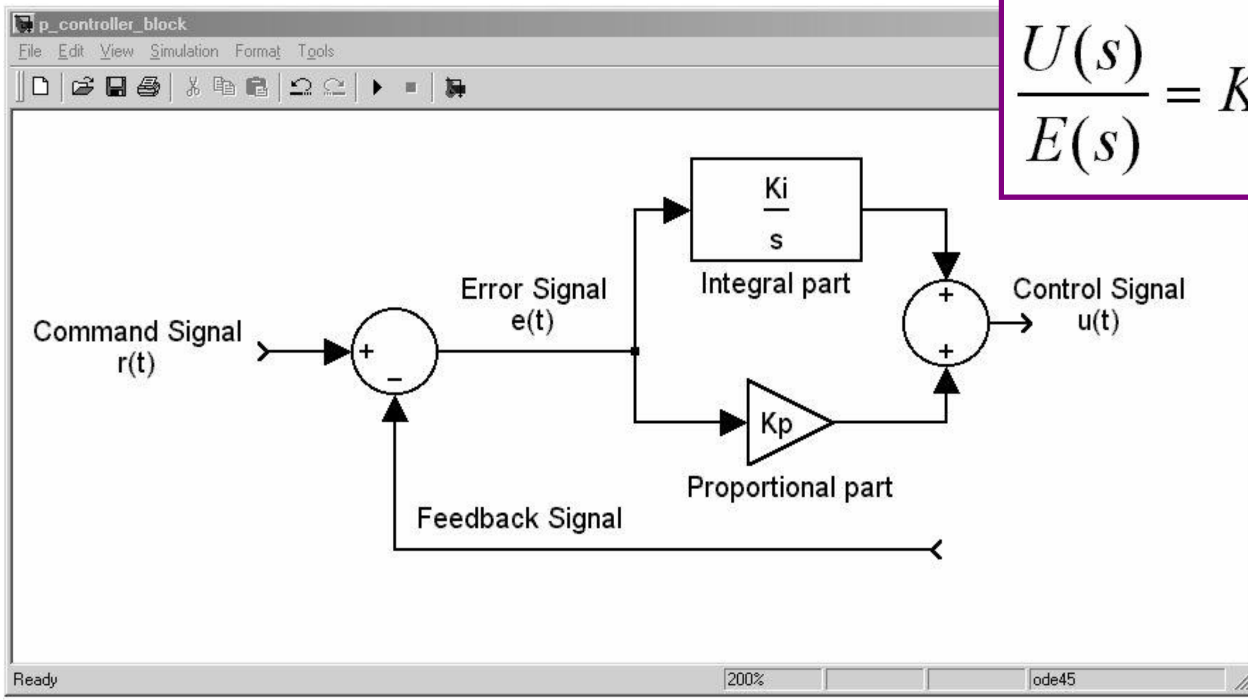
- **P controller calculates the control input based on the current value of the error**
- **I controller calculates the control input based on the accumulated (integrated) value of the error**

- **A combination of both would possess the two properties collectively. This type of a controller is called PI controller**

Proportional-Integral (PI) Controller

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt$$

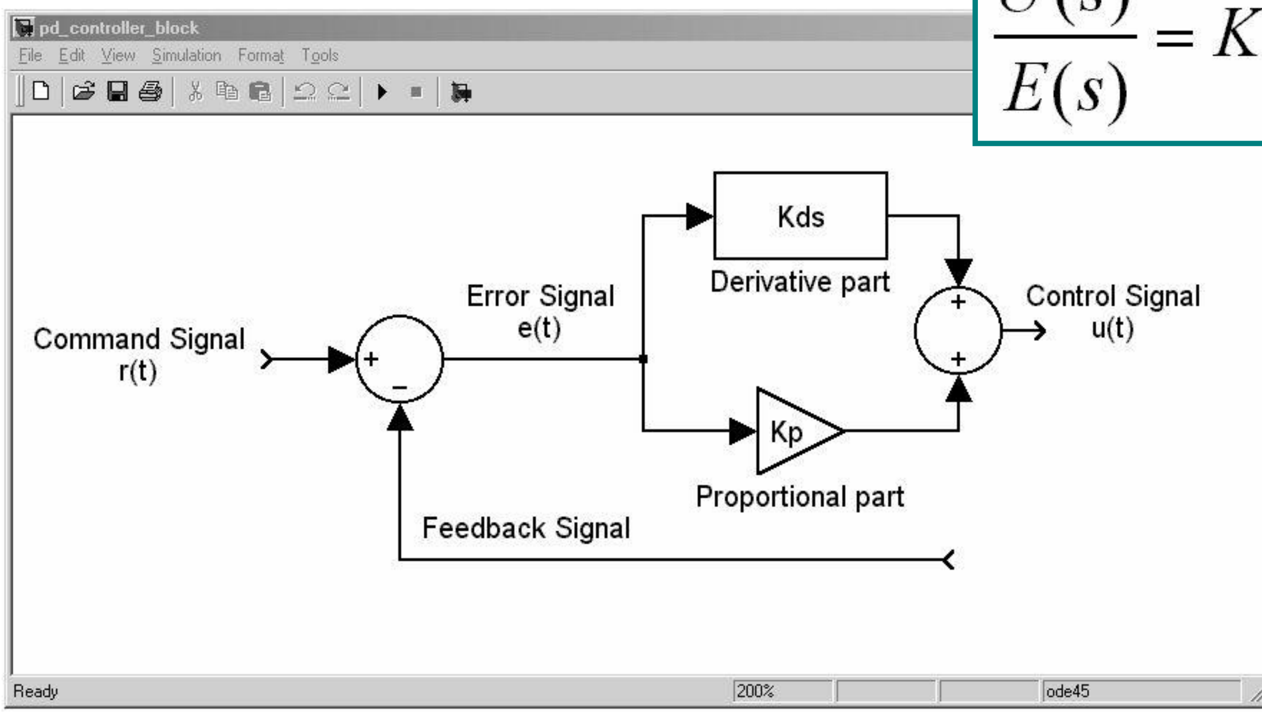
$$\frac{U(s)}{E(s)} = K_p + \frac{K_i}{s}$$



Proportional-Derivative (PD) Controller

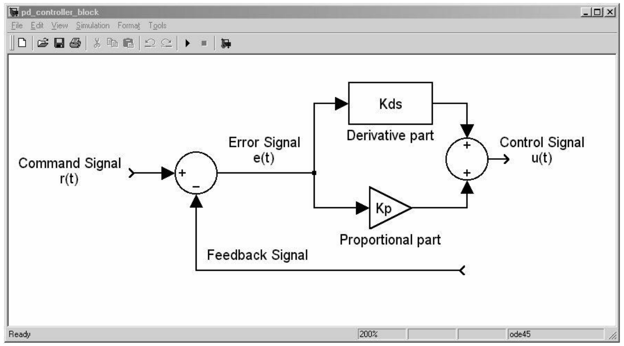
$$u(t) = K_p e(t) + K_d \frac{de(t)}{dt}$$

$$\frac{U(s)}{E(s)} = K_p + K_d s$$



Note that

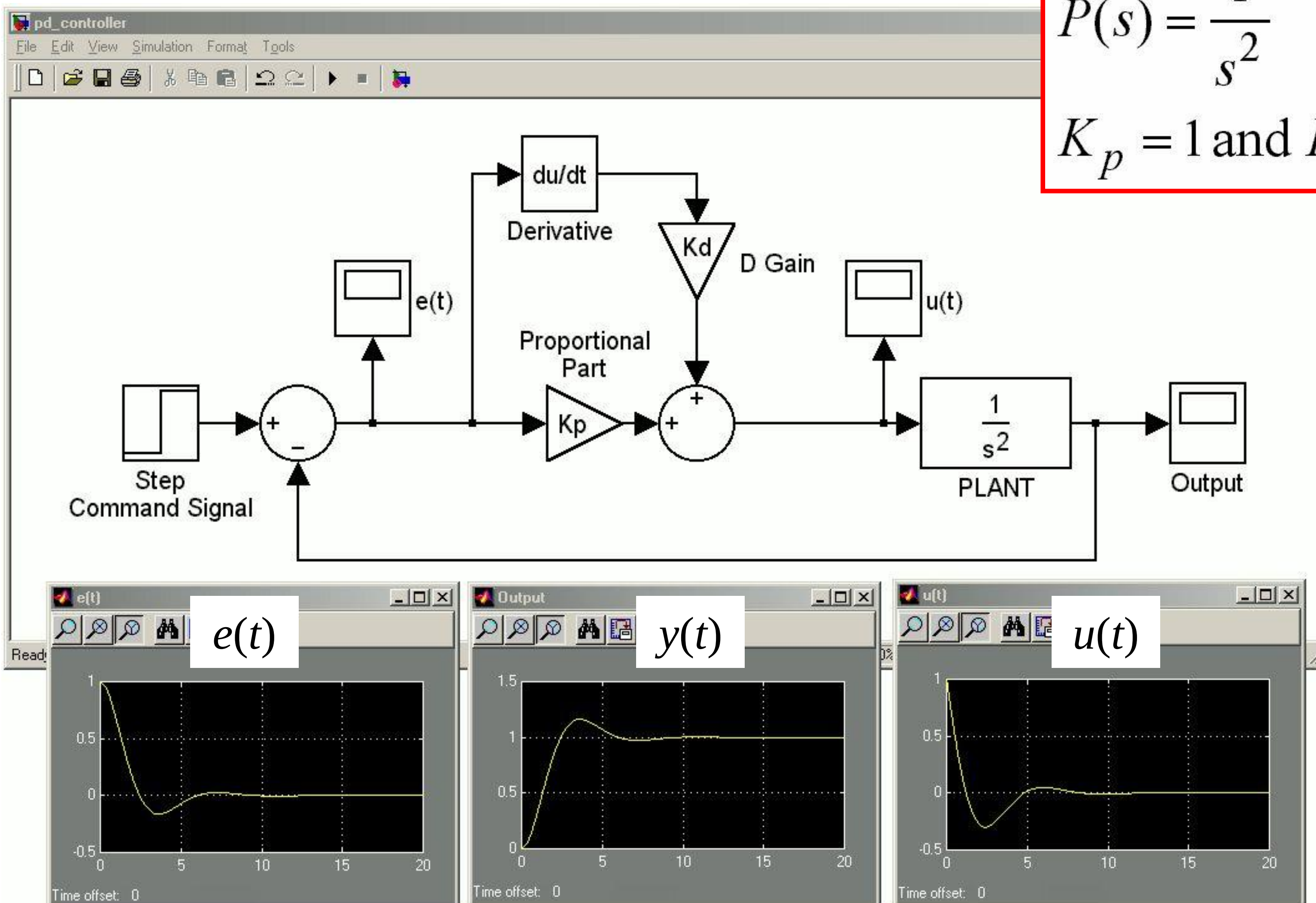
- If the feedback signal is noisy, e(t) will be noisy
- Differentiation of a noisy signal can lead to an excessively large output! Several modifications can be proposed...
- Derivative action introduces anticipatory behavior since it is based on the slope of the error signal
- A combination of P-D actions would possess the two properties collectively. This type of a controller is called PD controller



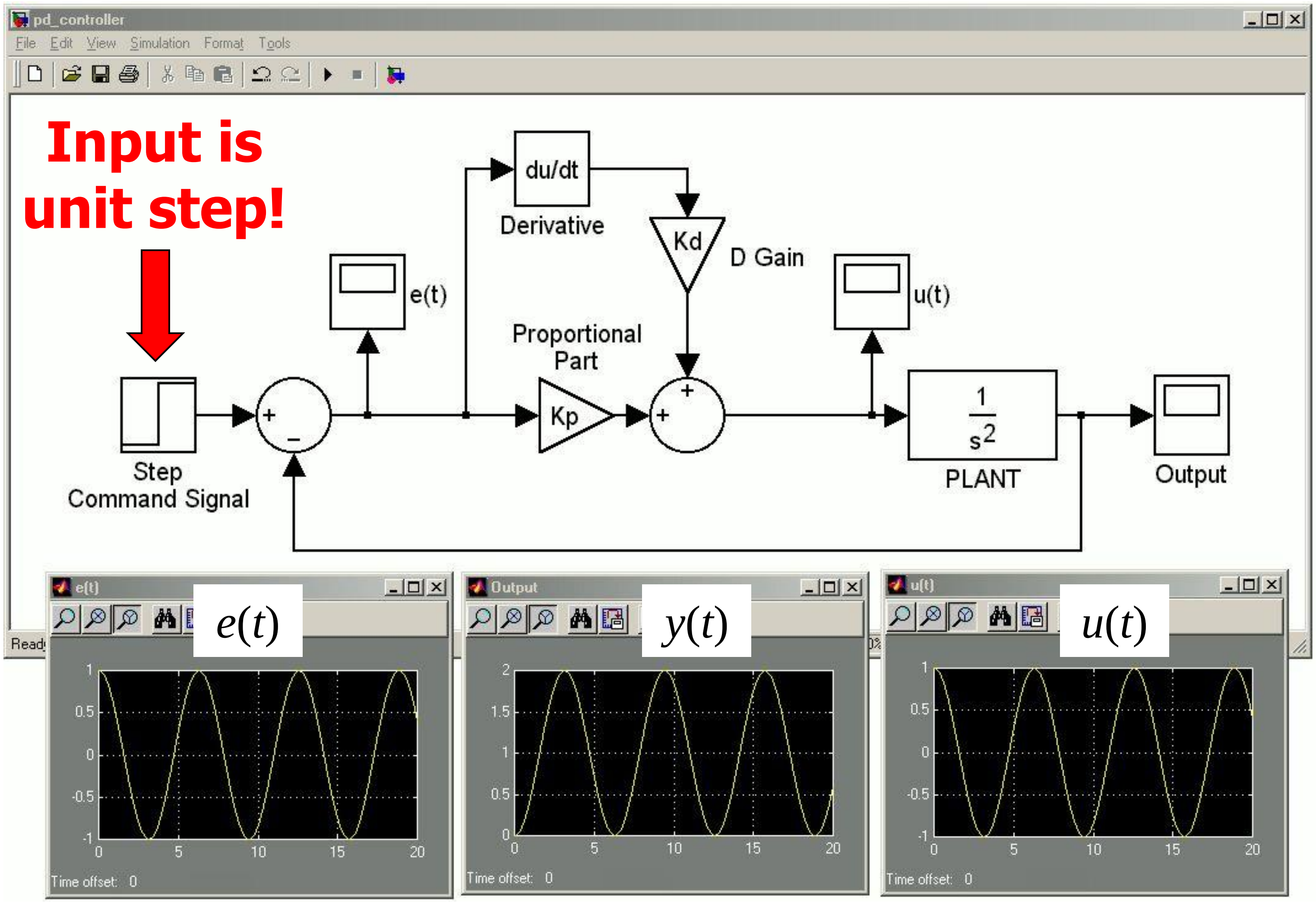
Proportional-Derivative (PD) Controller
An Example

$$P(s) = \frac{1}{s^2}$$

$$K_p = 1 \text{ and } K_d = 1$$



Proportional-Derivative (PD) Controller
An Example: Now set $K_d=0$



Proportional-Derivative (PD) Controller
An Example: Let's analyze what happened...

$$T(s) = \frac{K_d s + K_p}{s^2 + K_d s + K_p}$$

CL Transfer Function

$$T(s) = \frac{K_p}{s^2 + K_p}$$

CLTF with K_d=0

$$R(s) = \frac{1}{s}$$

Unit Step

$$y(t) = L^{-1}\left\{ \frac{1}{s} \frac{K_p}{s^2 + K_p} \right\} = 1(t) - \cos\left(\sqrt{K_p}\, t \right)$$

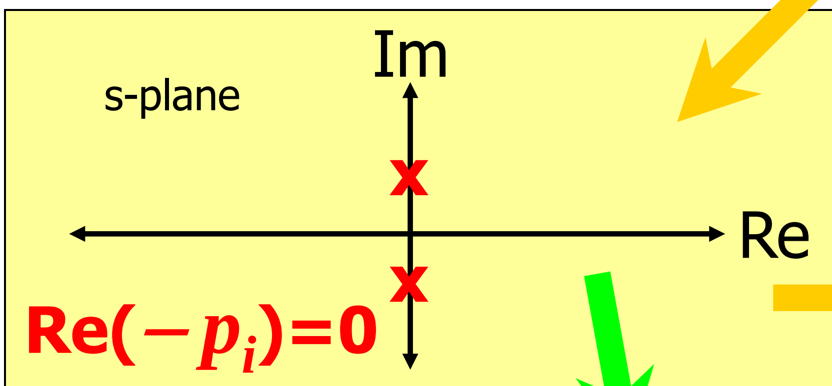
With only proportional controller, the output oscillates in response to constant input

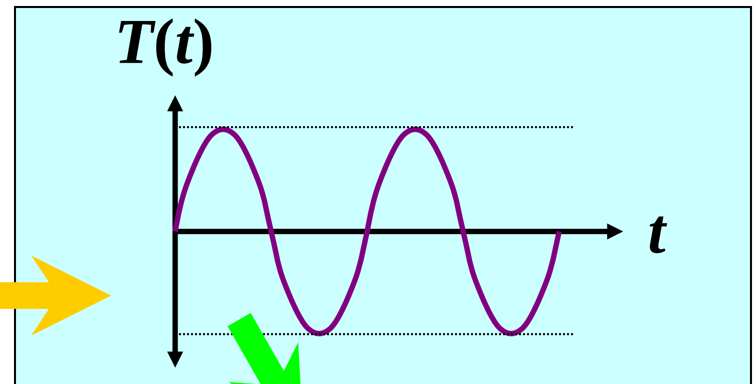
Proportional-Derivative (PD) Controller
An Example: Let's see in terms of stability

$$T(s)=\frac{K_d s+K_p}{s^2+K_d s+K_p}$$

$$T(s)=\frac{K_p}{s^2+K_p}$$

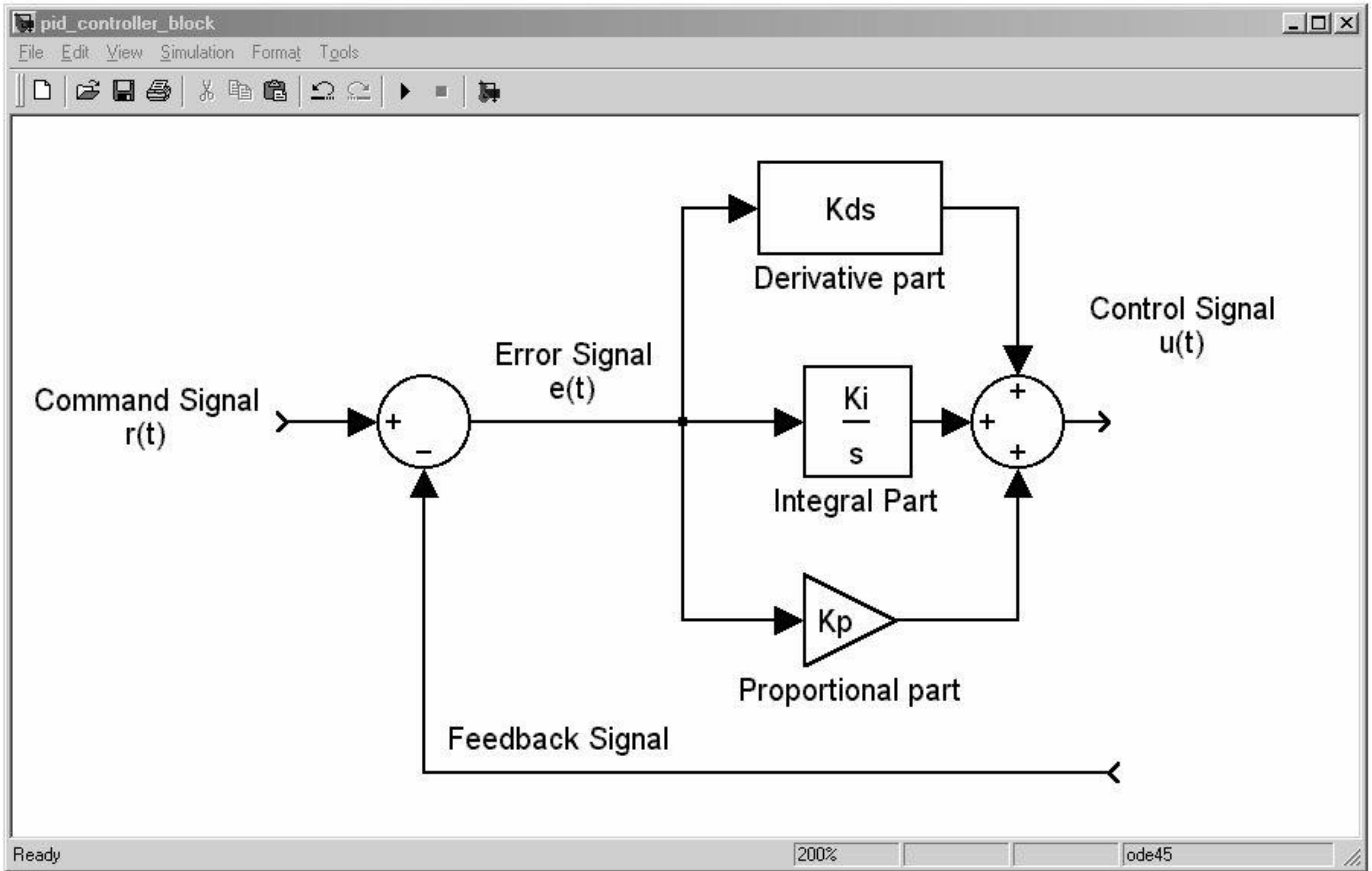
$$R(s)=\frac{1}{s}$$





$$y(t)=L^{-1}\left\{\frac{1}{s}\frac{K_p}{s^2+K_p}\right\}=1(t)-\cos\left(\sqrt{K_p}\,t\right)$$

Proportional-Integral-Derivative (PID) Controller



Proportional-Integral-Derivative (PID) Controller

- Over 95% of the controllers operating in industry are of type PID
- PID Controller utilizes the information contained in the current value, accumulated value and the tendency of the error signal
- Hardware/Software implementation of the PID controller is easy

Proportional-Integral-Derivative (PID) Controller

- **If the plant transfer function is changing, PID controller may not account for the entire set of combinations**

PID Controller
Questions & Answers

Q: Can we so freely assign the controller parameters?
A: NO

Q: What constraints do we have in designing a PID controller?
A: First requirement is the stability, then the design specifications must be met

Q: How to check stability compactly? What are design specifications?
A: Next week's agenda...