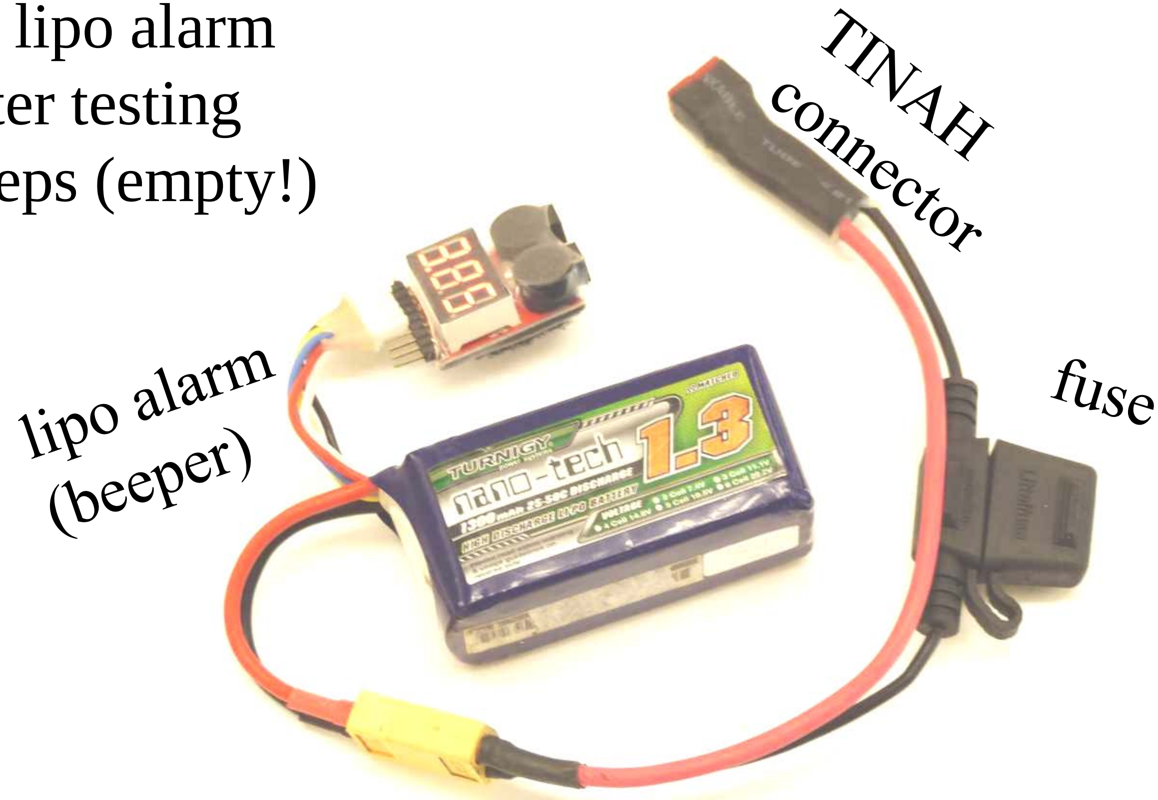
Thursday Lab: Lipos

Lipo = Lithium Polymer Battery = a powerful battery that will go up in flames when treated badly

For testing on the competition surface

- don’t take to your bench
- don’t operate without lipo alarm
- return to Bernhard after testing
- stop using when it beeps (empty!)

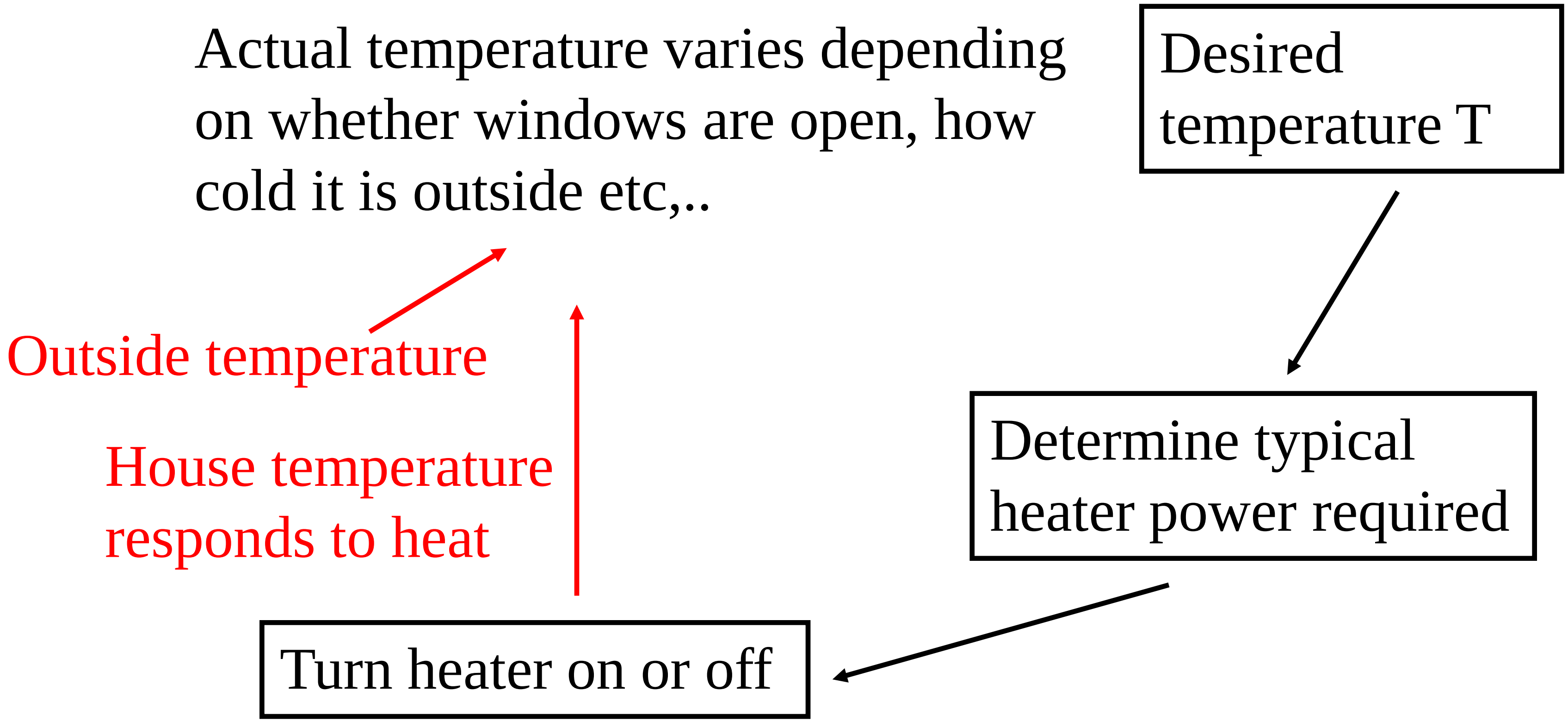


Lecture 4 – Introduction to control

Feedback control is a way of automatically adjusting a variable to a desired value despite possible external influence or variations.

Eg: Heating your house.

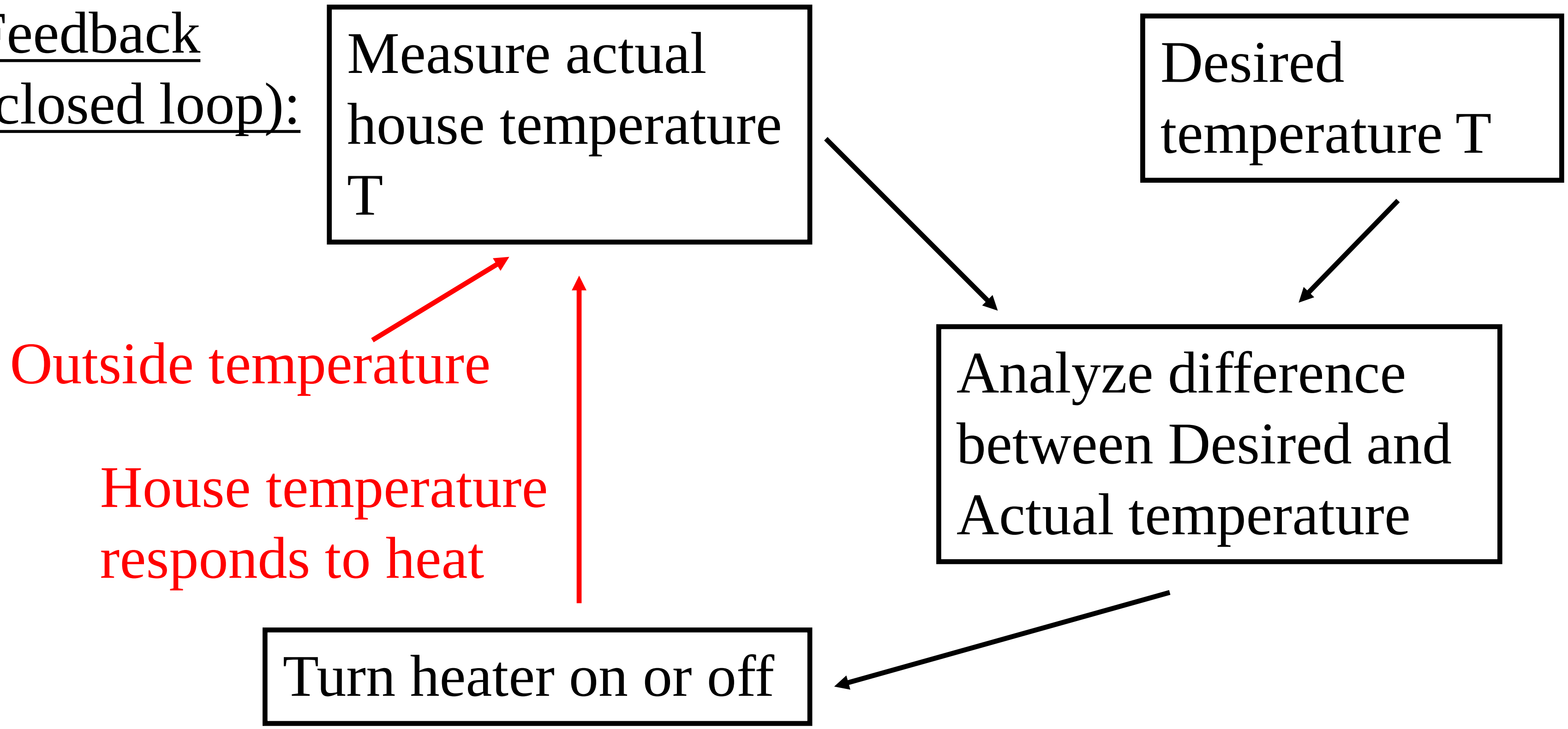
No feedback (open loop):



Control: Introduction

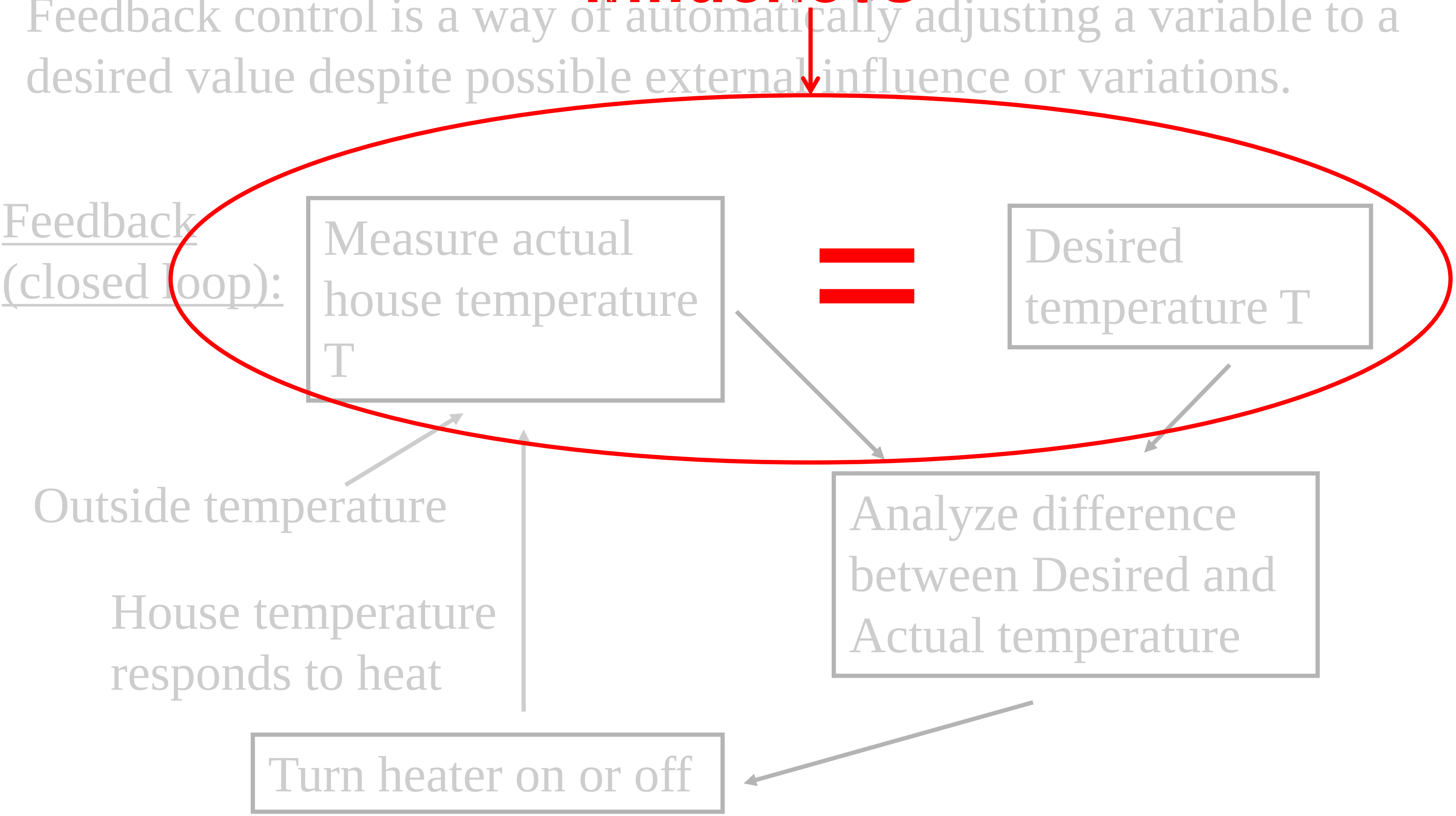
Feedback control is a way of automatically adjusting a variable to a desired value despite possible external influence or variations.

<u>Feedback (closed loop):</u>



The purpose of control theory is to make these two numbers the same despite external influences

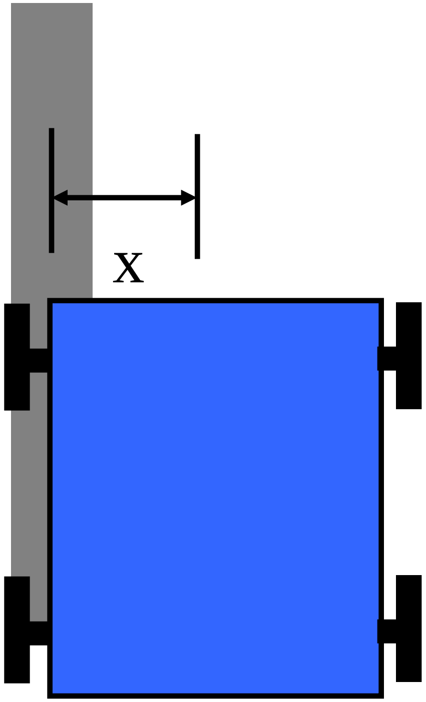
Feedback control is a way of automatically adjusting a variable to a desired value despite possible external influence or variations.



Control: Introduction

ON/OFF control:

X = distance between center of robot and center of tape




```
while(1)
    {
    if (x=0) go_straight();
    if (x>0) turn_left();
    if (x<0) turn right();
    }
```

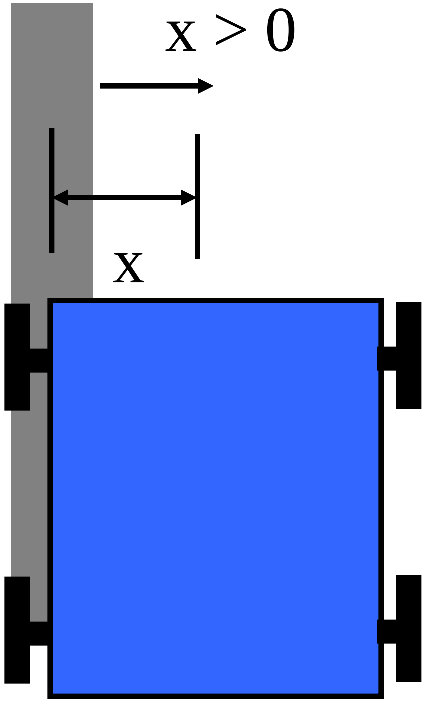
This tends to lead to oscillations around the center of the tape.

Control: Introduction

<u>Proportional control:</u>

X = distance between center of robot and center of tape

steer(int dir); - a hypothetical function that steers robot left (dir<0) or right (dir>0) in a radius of 600”/dir.

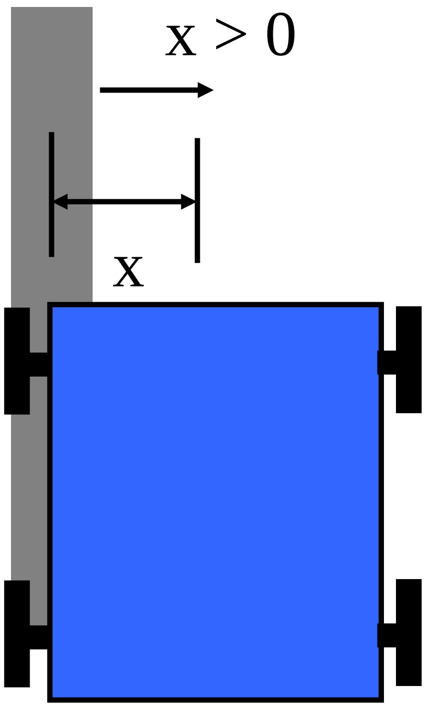



```
while(1)
        {
        steer(K*X);
        }
```

K is the proportional gain of this feedback loop and MUST be negative.

This is much better and more accurate than ON/OFF control, though it will still have significant error and oscillate for large values of K.

Control: Introduction



Proportional control:

X = distance between center of robot and center of tape

steer(int dir); - a hypothetical function that steers robot left (dir<0) or right (dir>0) in a radius of 600”/dir.

```
while(1)
        {
        steer(K*X);
        }
```

So what is the right algorithm?????

How do we optimize the robot to follow tape better?

Transfer functions revisited (Laplace transform notation: **s~jω**)

V(s) is the Laplace transform of v(t).

$$V(s) = \int_{0-}^{\infty} v(t)e^{-st}dt$$

Some rules:

1) Proportionality:

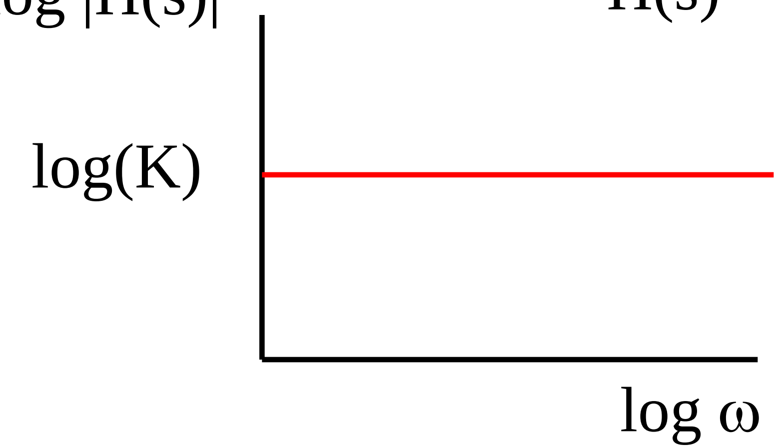
$v_{out}(t) = K*v_{in}(t)$ ← Time domain

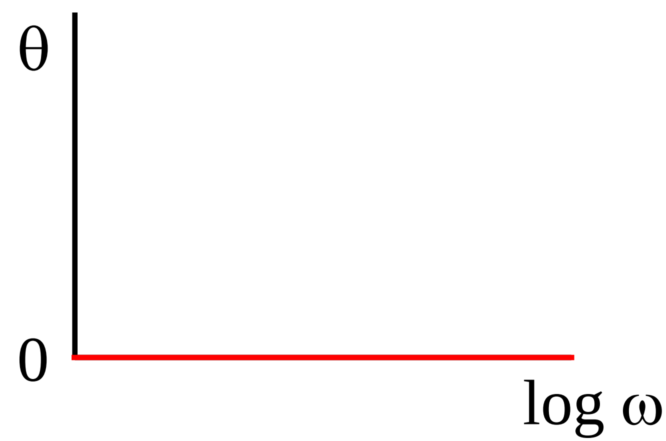
$H(s) = V_{out}(s)/ V_{in}(s)$

$V_{out}(s) = K*V_{in}(s)$ ← Frequency domain

$H(s) = K$

10log |H(s)|





2) <u>Integration</u>:

$$v_{out}(t) = K\int v_{in}(t)dt$$

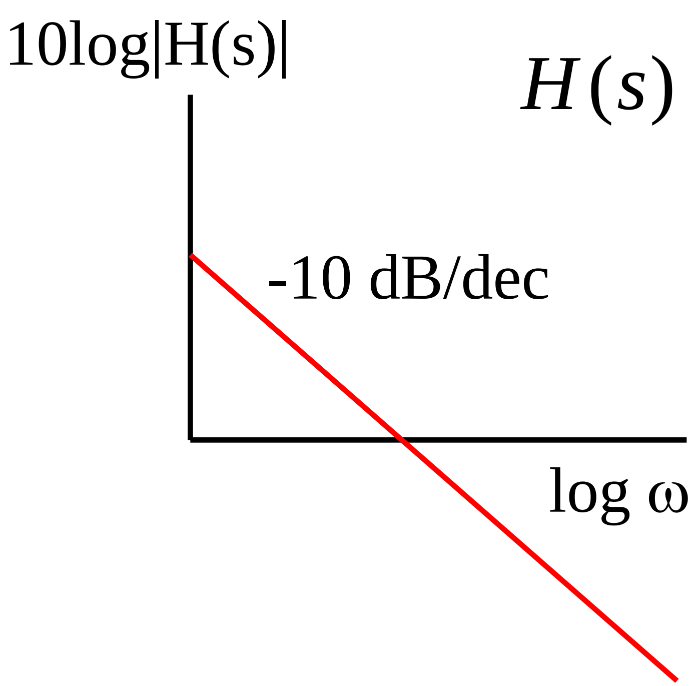
Time domain

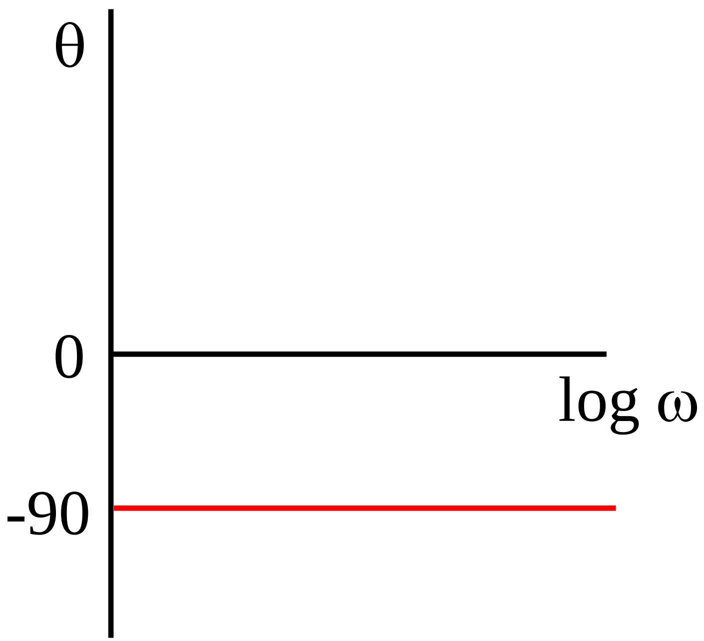
$$V_{out}(s) = \frac{KV_{in}(s)}{s}$$

Frequency domain

$$H(s) = \frac{K}{s}$$

Pole





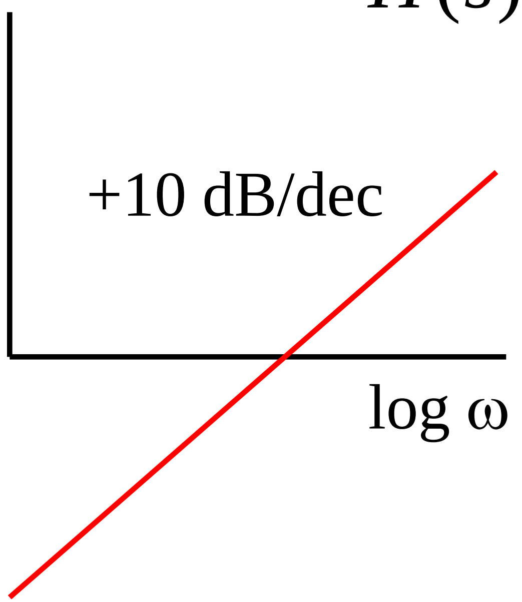
3) Differentiation:

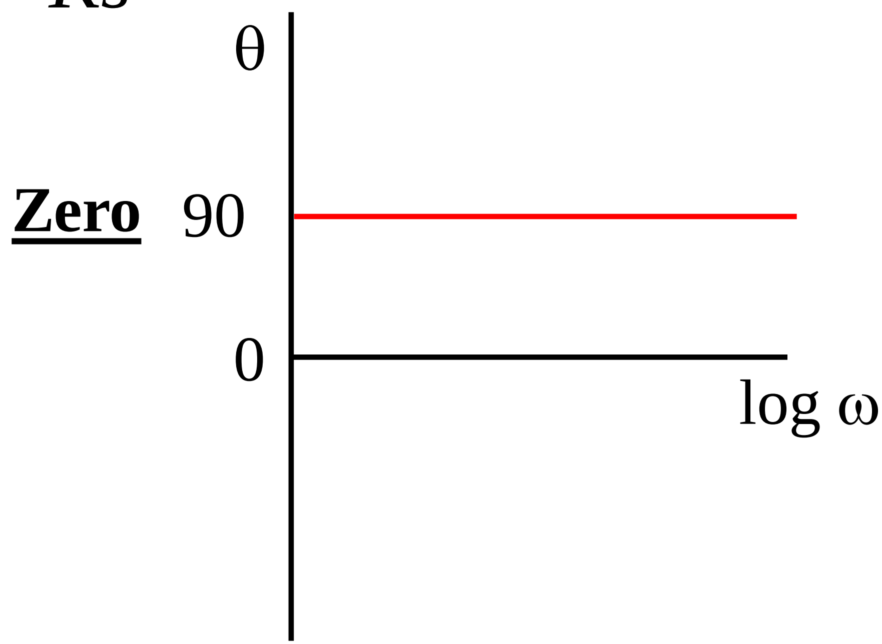
$$v_{out}(t) = K\frac{dv_{in}(t)}{dt}$$

$$V_{out}(s) = KsV_{in}(s)$$

10log|H(s)|

$$H(s) = Ks$$

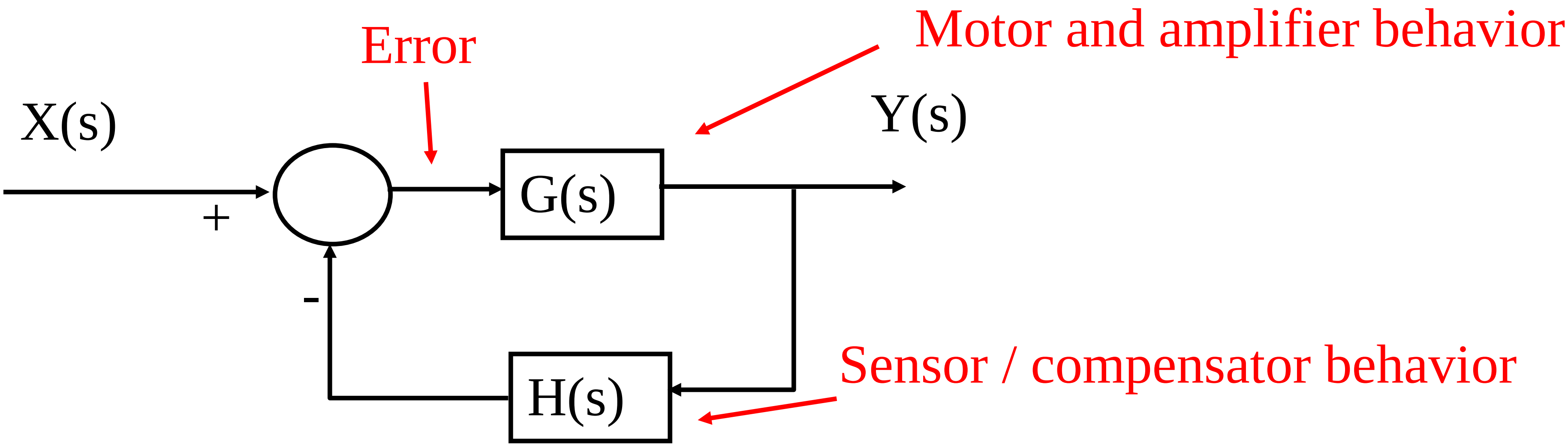




Feedback loops

Y = variable you'd like to control (eg: shaft angle of a servo motor)

X = your desired value of Y (eg: 10 degrees)



Y = G(X-HY)

Y(1+GH) = GX

Y/X = G/(1+GH)

$$\frac{Y}{X} = \frac{G}{1+GH}$$

G = forward transfer function, GH = loop transfer function

Feedback loops: stability

$$\frac{Y}{X} = \frac{G}{1+GH}$$

This loop will be unstable if GH = -1

→ |GH|=1, phase(GH)= ±180 deg.

G(s)H(s) = -1 implies $\frac{Y}{X} = \infty$ for some value of s

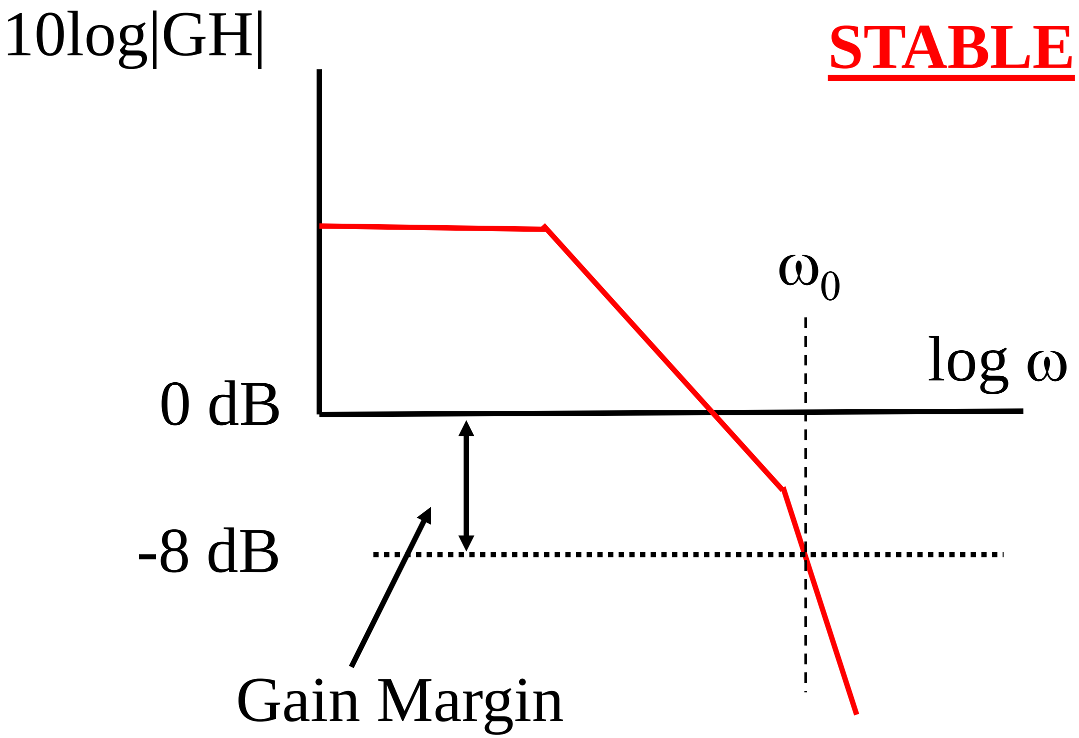
i.e. there will exist a frequency for which the loop will provide infinite amplification

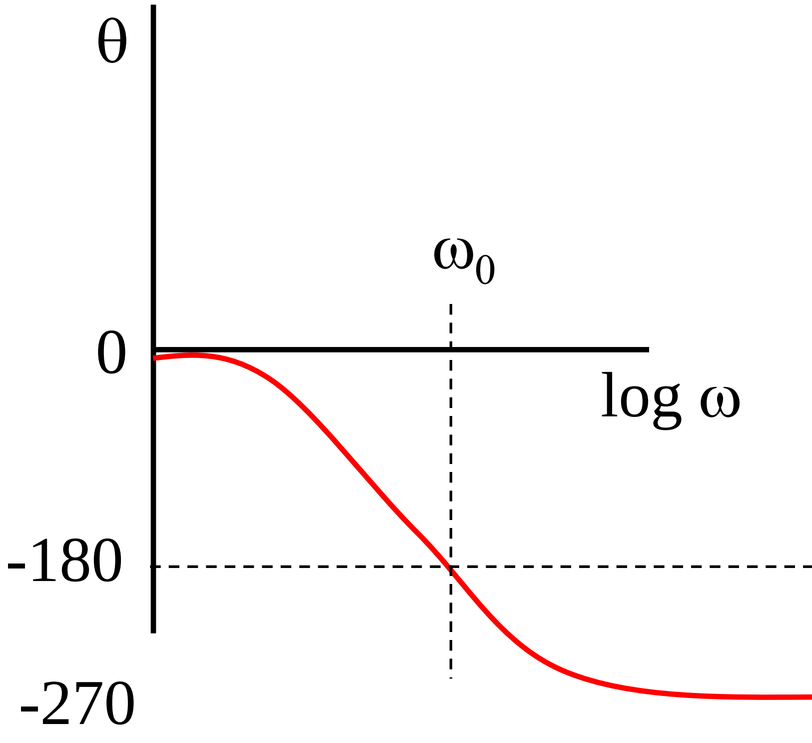
Loop Stability

$$\frac{Y}{X} = \frac{G}{1 + GH}$$

Partial stability criterion:
|GH| < 1 where the phase of GH is ± 180 deg.

STABLE



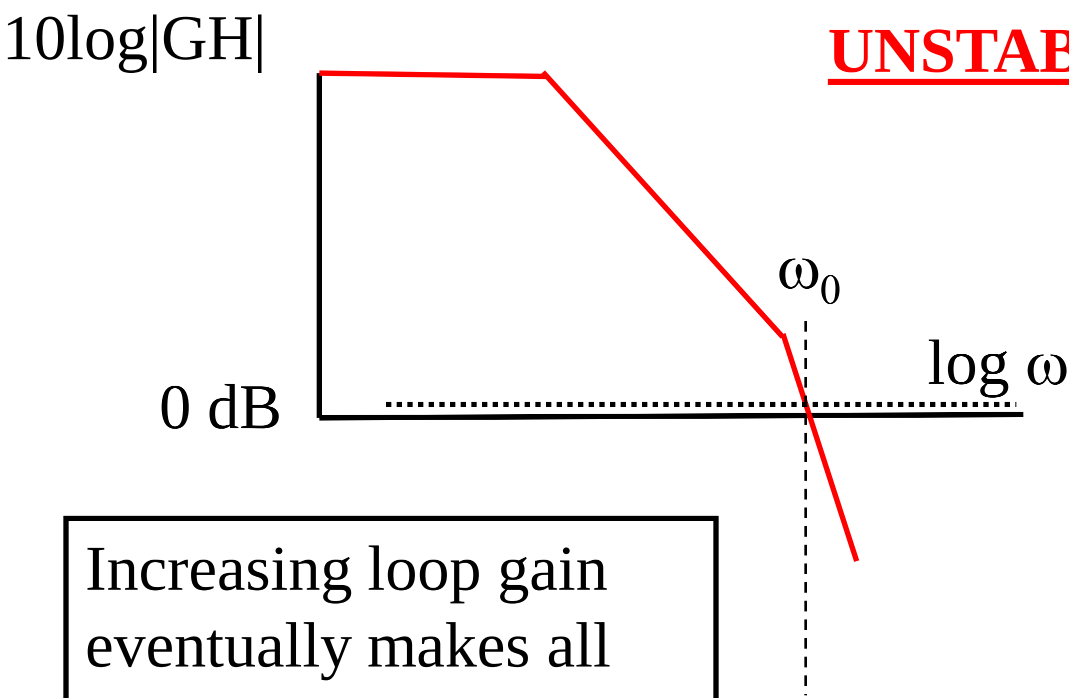


Loop Stability

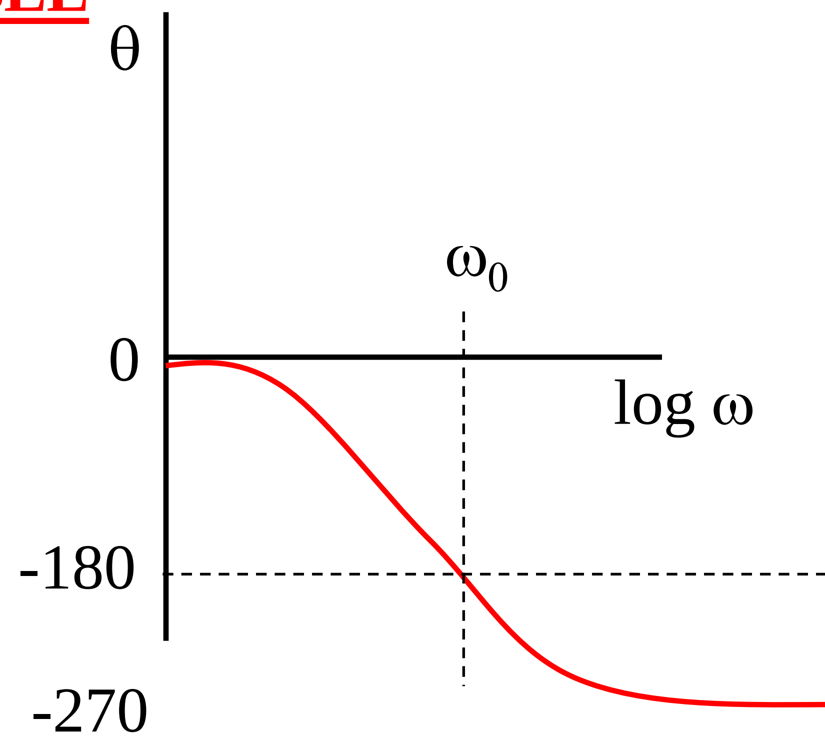
$$\frac{Y}{X} = \frac{KG}{1 + KGH}$$

Partial stability criterion:

|GH| < 1 where the phase of GH is ± 180 deg.



UNSTABLE



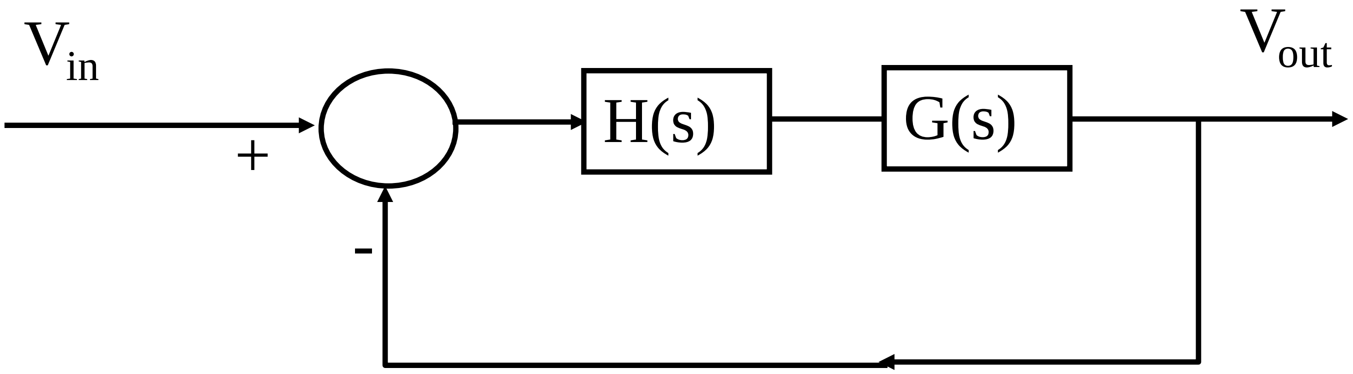
Increasing loop gain eventually makes all systems unstable

Stability Summary

$$\frac{Y}{X} = \frac{KG}{1 + KGH}$$

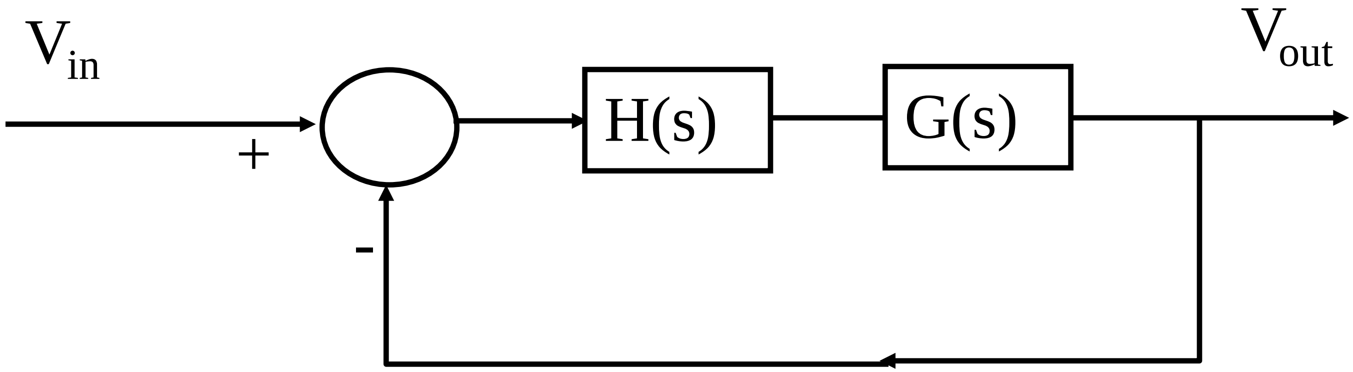
- Having one or fewer poles in the plant function KGH ensures that the loop is never unstable.
- The more poles exist in KGH, the harder it will be to control.
- Problems will start to occur when controlling at frequencies above the pole frequencies.
- Increasing loop gain eventually makes all systems unstable due to unexpected high frequency poles.

Compensation



- A feedback system is usually divided into two transfer functions:
  - The “plant” function (G(s)) which usually you cannot alter (motor characteristics etc.)
  - A compensator circuit H(s) that you can design to optimize the feedback loop
- A common type of “all-purpose” compensation is PID:
  - Proportional (K_p)
  - Integral (K_i/s)
  - Derivative (sK_d)

PID Compensation



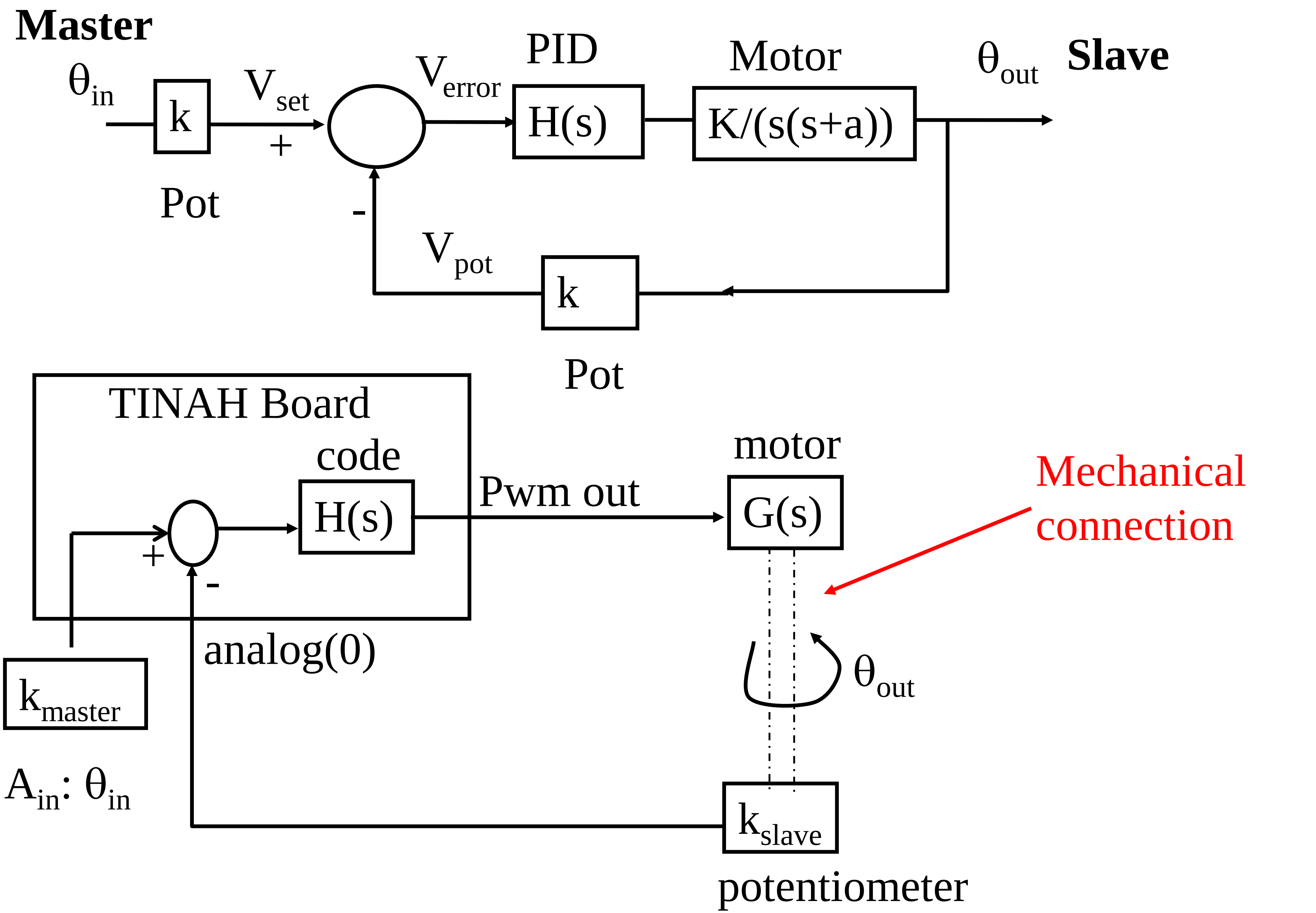
Typical PID transfer function:

$H(s) = K_{tot}(K_p + K_i/s + sK_d)$

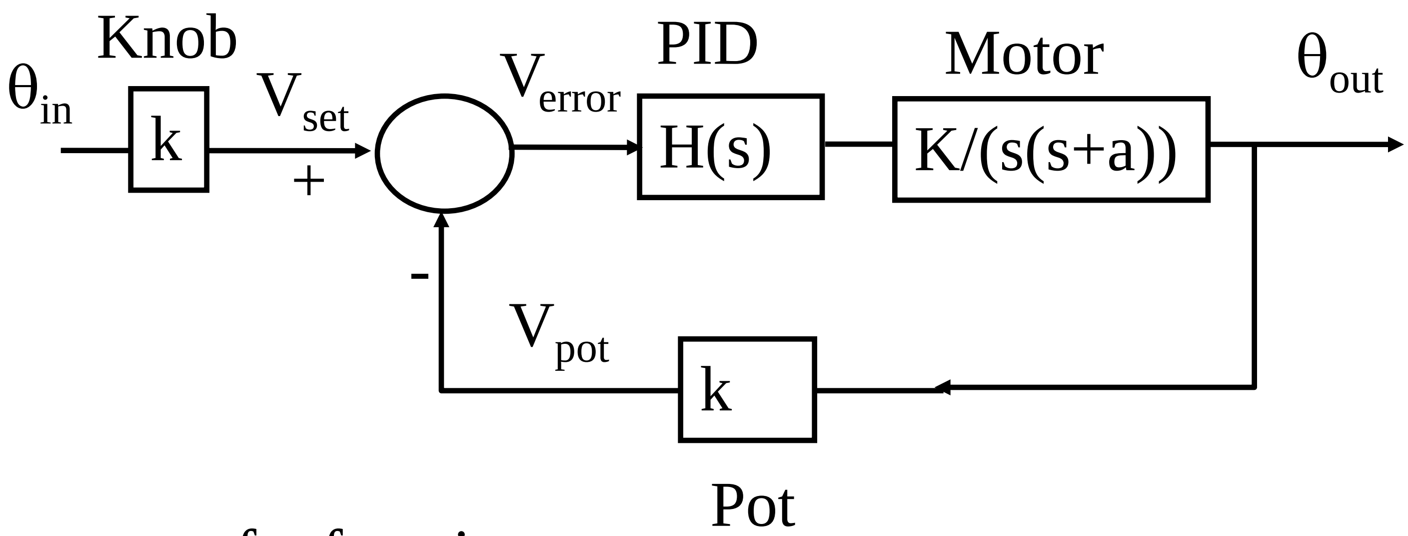
The various gains (K_{tot}, K_p, K_i, K_d) are adjusted to control how much of each type of compensation is applied for a specific plant function G(s).

This adjustment is referred to as "tuning" and is often done iteratively (a slightly improved form of trial and error) when the plant function G is not well known.

PID example: position servo (demo)



PID example: position servo



Motor transfer function:

$$\theta = \int \omega_{\max} dt$$ (at low frequencies: G=K/s)

$$\theta = \int\int \alpha dt$$ (at high frequencies: G=K/s^2)

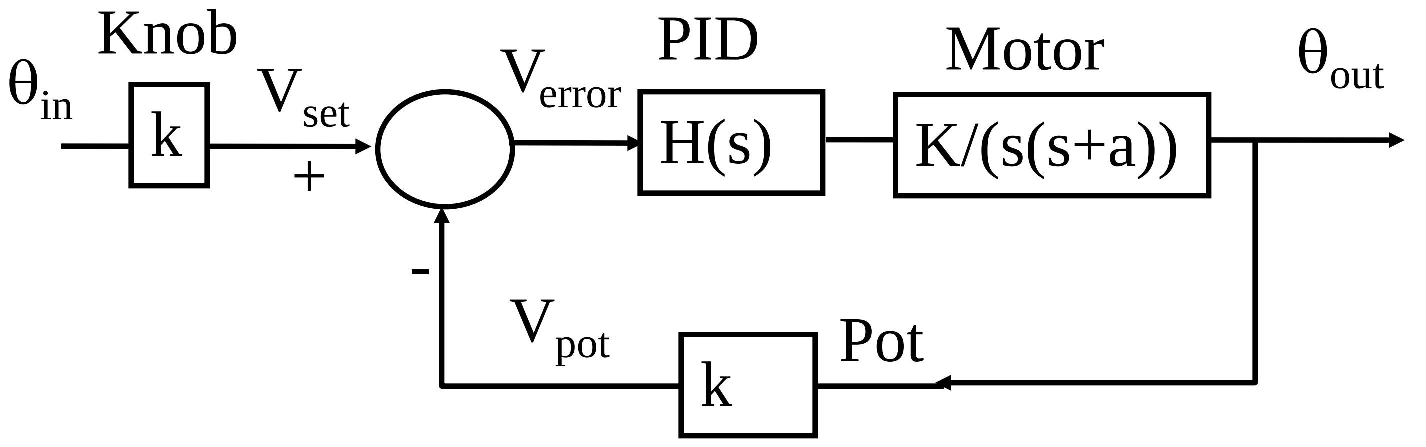
$$\alpha = \frac{Torque}{Inertia}$$

$$G(s) = \frac{K}{s(s+a)}$$

$$\theta = \int \omega$$

$$\omega = \int \alpha$$

PID example: position servo



Loop transfer function (stability analysis):

$$G(s) = \frac{K}{s(s+a)}$$

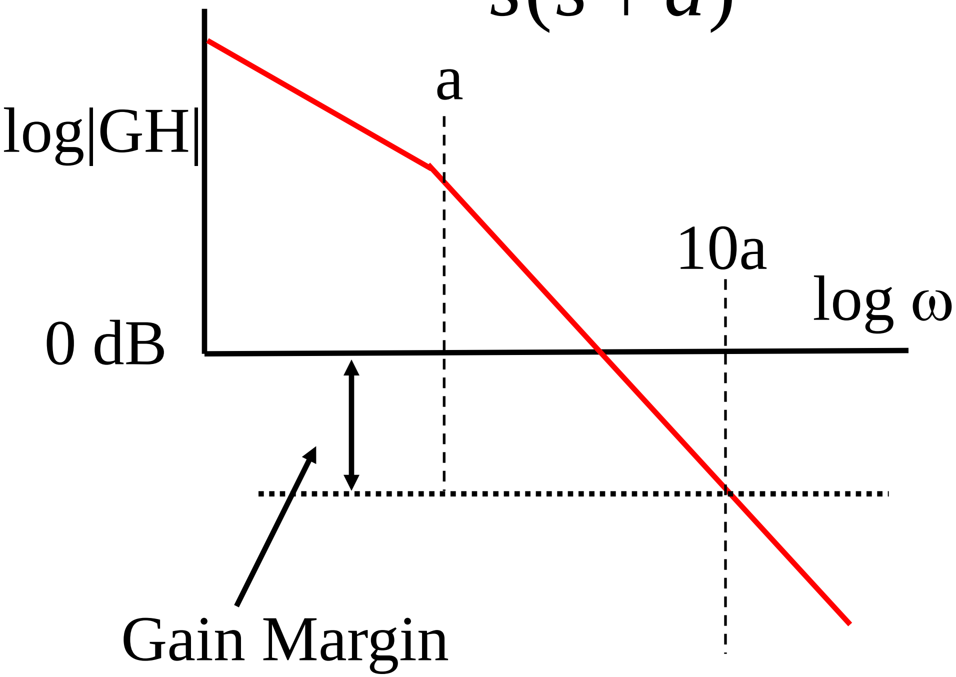
H(s) =?

Try proportional control: $H(s) = K_p$

Stability: position servo – P control

Loop transfer function (P only):

$$GH(s) = \frac{KK_p}{s(s+a)}$$

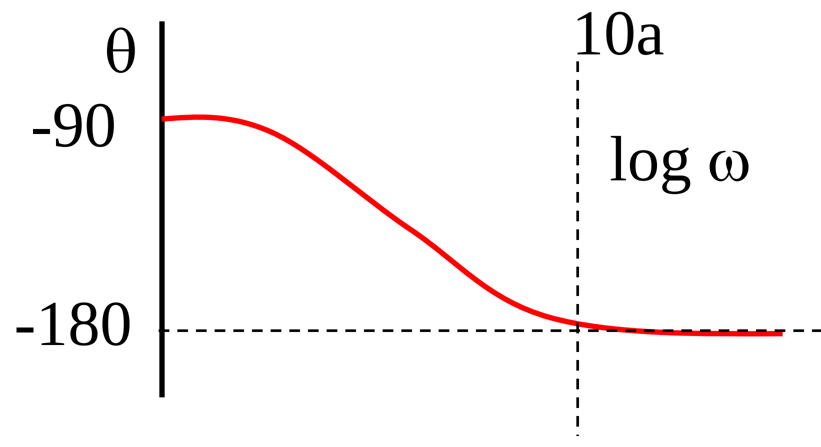


Stable for limited gain

$$\theta_{out} = \frac{KK_p}{s(s+a)} V_{error}$$

$$V_{error} = \frac{s(s+a)}{KK_p} \theta_{out}$$

$V_{error} = 0$ at s=0!

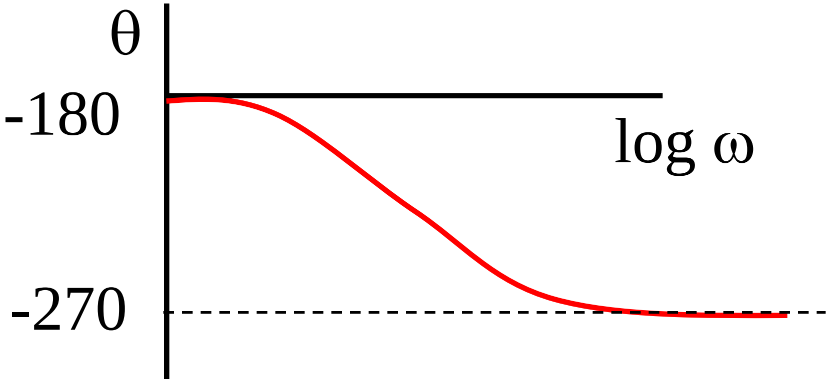


Stability: position servo – I control

Open loop transfer function (I only):

$$H(s) = \frac{K_i}{s}$$

$$GH(s) = \frac{KK_i}{s^2(s+a)}$$



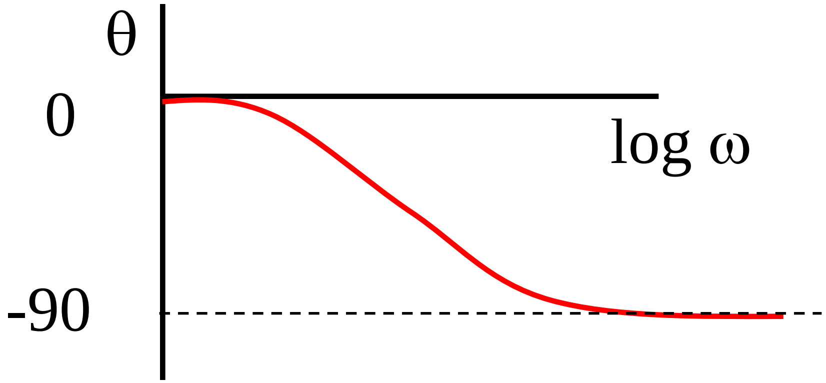
Phase crosses –180 at DC, with infinite DC gain!
Inherently unstable at s=0

Stability: position servo – D control

Open loop transfer function (D only):

$$H(s) = sK_d$$

$$GH(s) = \frac{KK_d}{(s+a)}$$



Phase always less than –180
Stable even for large gains!

$$V_{error} = \frac{(s+a)}{KK_d}\theta_{out}$$

SS error ≠ 0!

Problems:

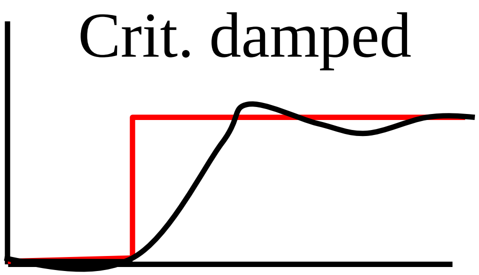
- May be hard to implement due to amplification of fast transients.
- Can be combined with P gain to add high gain stability and low SS error
- Model is not complete – loop will still be unstable at very high gains.

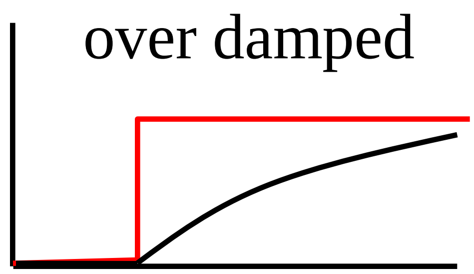
Tuning PID

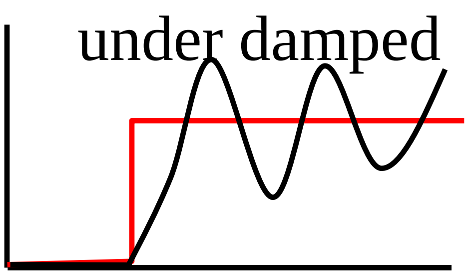
Often PID tuning is done by nearly trial and error. Here is a common Procedure which works for many (but not all) plant functions.

USE external pots or menus to adjust!!!!!

- Set P=I=D=0
- Increase P slightly and ensure that the sign of the gains is correct.
- Increase P until oscillations begin
- Increase D to dampen oscillations
- Iterate increasing P and D until fast response is achieved with little overshoot
- Increase I to remove any Steady State error.
- If overshoot is too large try decreasing P and D.
- Test with step response:



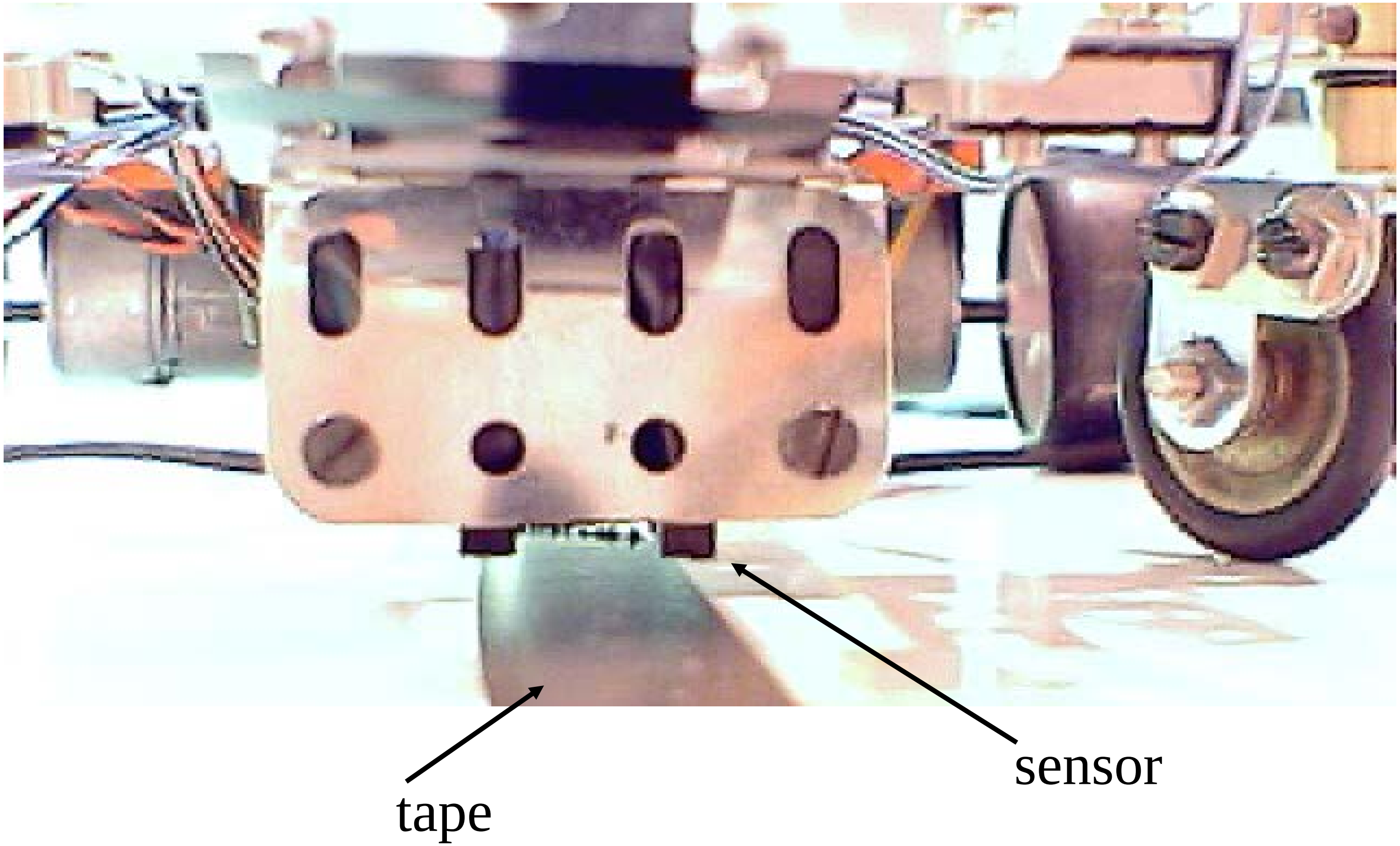




Control: Introduction

How to measure X (distance from tape):

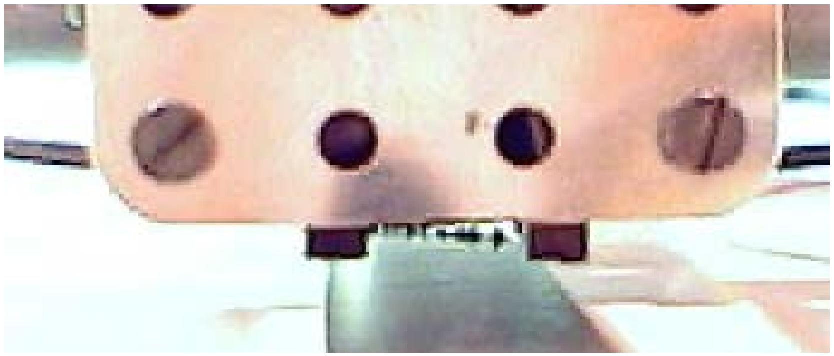
Use QRD1114 reflectance sensors to detect lack of reflectance from tape.

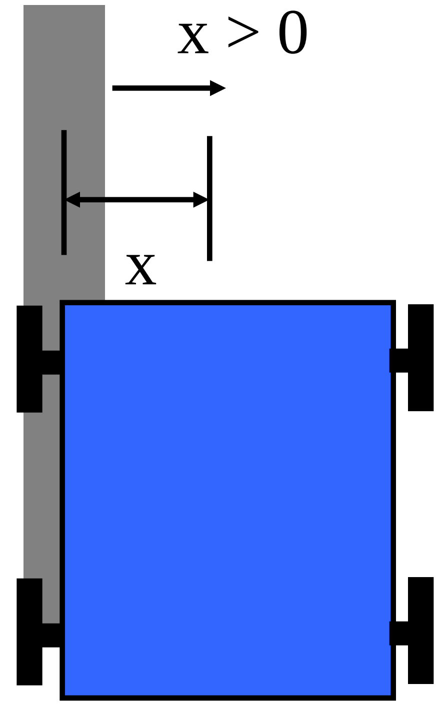


Control: Introduction

How to measure X:

X = distance between center of robot and center of tape



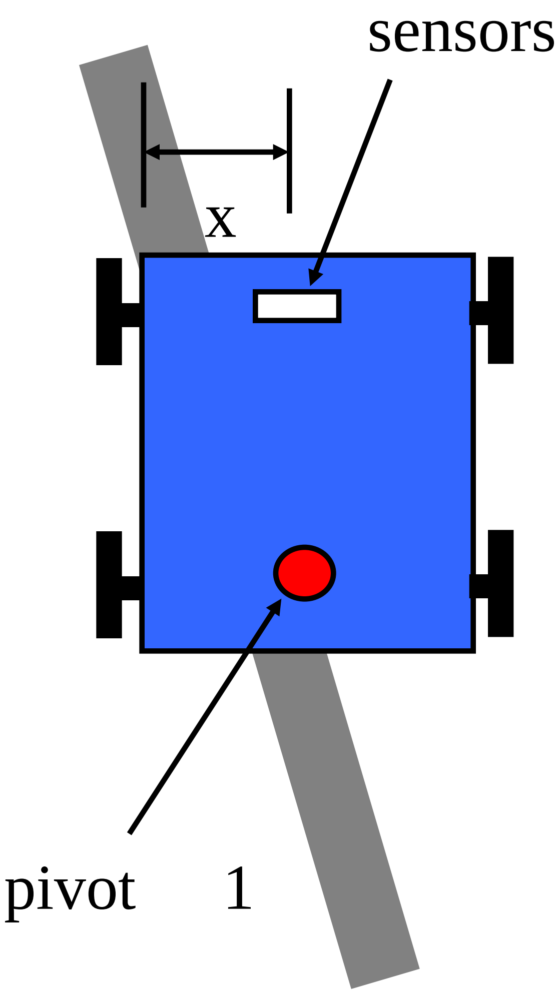


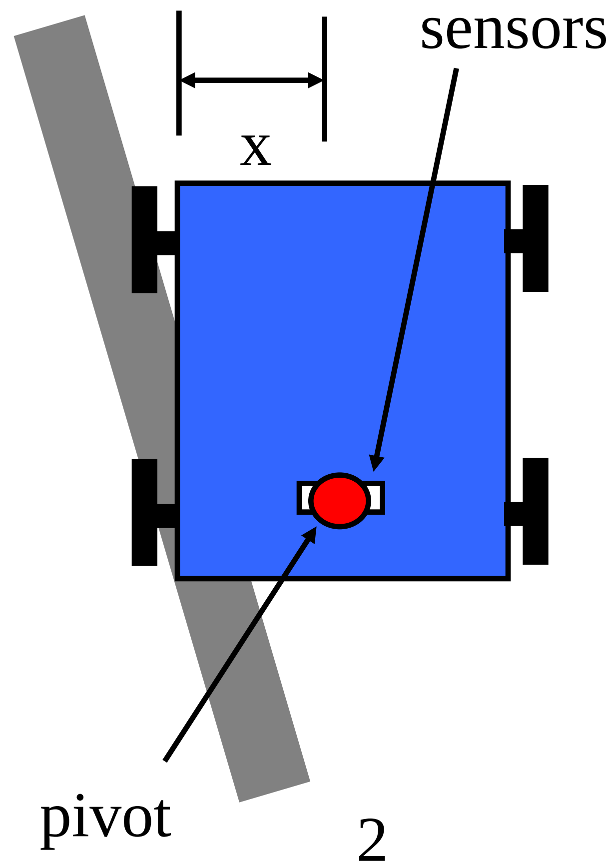
You can form a rough approximation of X by digital to analog conversion of your digital inputs with history:

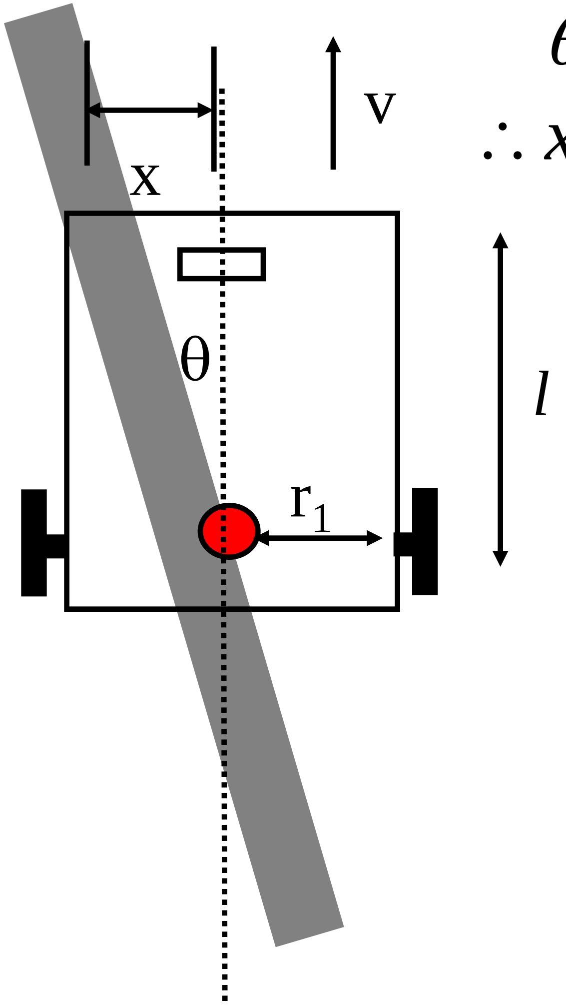
Situation	**Left sensor**	**Right Sensor**	**X**
Both sensors on tape	1	1	0
Left sensor off tape, right on	0	1	-1
Right sensor off tape, left on	1	0	+1
Both sensors off (right was last on)	0	0	-5
Both sensors off (left was last on)	0	0	+5

Please consider the following problem for a robot with differential rear drive steering:

Which robot configuration has more poles in the transfer function between I (current to motors) and x (distance of sensors from tape)?







$\theta_{in} = 0$ (we want robot to follow tape)

$\therefore x_{in} = 0$

Actual x value in time domain:

$$x = l\sin\theta + \int v\sin\theta dt$$

$$\approx l\theta + \int v\theta dt$$

Actual X in frequency domain:

$$X \approx l\theta + \frac{v\theta}{s}$$

$X \sim l\theta$ **at low *v***

$X = \dfrac{v\theta}{s}$ for $l = 0$

$$\theta = \frac{K_{bot} I_{pwm}}{s(s+a)}$$

where $a \propto \dfrac{(T/D)r_1}{I_{bot}}$

I_{bot} is the chassis moment of inertia

T is motor torque

D is wheel diameter

Design tips for stability

- Minimize robot polar moment of inertia (I_{robot})
- Maximize robot torque about polar axis (r_1/D_{wheel})
- Maximize distance from polar axis to tape sensors (l)
- Minimize sensor dead band
- Change gear ratio / wheel size to increase torque / reduce speed if you find stability is only achieved at very low motor powers.

$$X \approx l\theta + \frac{v\theta}{s}$$

$$\theta = \frac{Kr_1 I_{pwm} / D_{wheel}}{s(s + Kr_1 I_{pwm} / I_{robot} D_{wheel})}$$

Inertia Ratio Definition

Inertia Ratio

In motion control the inertia ratio is defined as follows:

$$Inertia\ Ratio = \frac{I_l}{I_m}$$

$Where$

$I_l = load\ inertia$

$I_m = motor\ inertia$

For optimal power transmission the inertia ratio is 1:1

$$Power = Force * Velocity$$

If velocity and mass are fixed, then power improvement is created by acceleration improvement

Inertia Optimization Proof

$$T = I \times \propto = \left(J_m + \frac{J_L}{{G_r}^2}\right) \times \propto_M$$

Where:

T_M = Motor Torque

J_M = Motor Inertia

J_L = Load Inertia

G_r = Gear Ratio

α_M = Acceleration of Motor

Motor acceleration:

$$\propto_M = \propto_L \times G_r$$

$$\propto_L = \frac{T_M \times G_r}{J_M \times G_{r^2} + J_L}$$

Inertia Optimization Proof

Taking the derivative of α_L with respect to G_r:

$$\frac{d \propto_L}{dG_r} = \frac{\left(J_M G_r{}^2 + J_L\right)(T_M G_r)' - (T_M G_r)\left(J_M G_r{}^2 + J_L\right)'}{\left(J_M G_r{}^2 + J_L\right)^2}$$

$$\frac{d \propto_L}{dG_r} = \frac{\left(J_M G_r{}^2 + J_L\right)T_M - (T_M G_r)(2J_M G_r)}{\left(J_M G_r{}^2 + J_L\right)^2}$$

Simplifying to :

$$\frac{d \propto_L}{dG_r} = \frac{\left(J_L - J_M G_r{}^2\right)T_M}{\left(J_M G_r{}^2 + J_L\right)^2}$$

Inertia Optimization Proof

To find the gear ratio that results in the maximum acceleration the derivative is set equal to zero.

$$0 = \frac{(J_L - J_M {G_r}^2)T_M}{(J_M {G_r}^2 + J_L)^2}$$

$$0 = J_L - J_M {G_r}^2$$

Simplify to:

$$G_r = \sqrt{\frac{J_L}{J_M}}$$

This demonstrates the claim for optimal power transmission at a 1:1 ratio

Inertia Ratio Recommendations

Typical Inertia Ratio Industry Recommendations

Stepper Motor Driven Systems:

1:1 or as close to 1:1 as is reasonable for the system

Servo Systems:

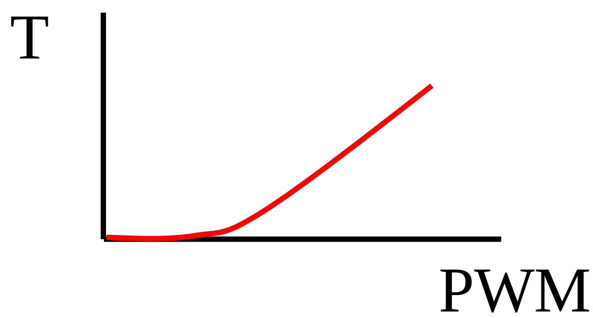
5:1 to 10:1 are typical industry recommendations, but specific system goals will move this range.

*****Performance goals driven by application requirements will ultimately determine what ratio is acceptable. Remember a lower ratio allows a system to respond faster and have tighter dynamic control.***

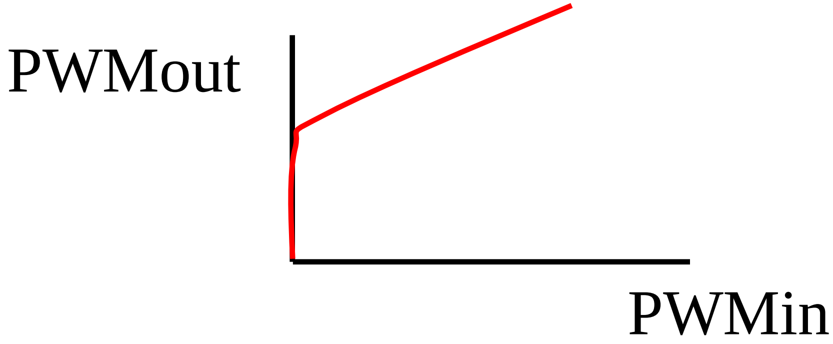
Linearization of non-linear functions

Control can be very difficult if G is non-linear.

PWM drive (combined with friction) yields a very non-linear torque curve:



Solution: Linearize this curve in software by mapping PWM to desired Torque



Analog PID in software (Servo control)

```
loop
    pot = analog(6);
    set = knob();

    error = set-pot;
    p=kp*error;
    d=kd*(error-lasterr);
    i=ki*error+i;
        if (i>maxi) i = maxi;
        if (i<-maxi) i = -maxi;
    g = p+i+d;
    motor(3,g);
    lasterr=error;
```

Feedback potentiometer

Set point

Proportional

Derivative

Integration

Anti-windup

Because i is an integral, it will build up to large values over time for a constant error. An anti-windup check must be put in place to avoid it overwhelming P and D control when the error is removed.