تقدم لجنة ElCoM الاكاديمية

تلخيص لمادة:

مختبر تحكم آلي

Day / /

$T(s) = \frac{s^2+3s+2}{s^2+2s+4}$ transfer fn.

relation bet. output and input with initial value = 0

P = 5; → لا يظهر النتيجة على command window ولكن ينفذها

n = [1 3 2];

d = [1 2 4];

① r = roots(n) → النتيجة roots

r = roots(d) → roots $\boxed{\text{Complex}}$ / Poly

second order under damped

② roots in left (-1) → Stable

cls → clear command window

clear delete data

②

Poly([-1 -2]) → النتيجة تكون [1 3 2]

roots

(x+1)(x+2)

$x^2+2x+x+2$

x^2+3x+2

③

Poly val(n, # or [ ]) يعوض

Ex Polyval(n,1) = n(1)

>> help polyval

④ result of Pol × Pol

n.d X

n*d X

n.*d [1 6 8] element by element

t = conv(n,d) ↵

t = 1 5 12 16 8

$s^4+5s^3+12s^2+16s+8$

Day / /

2

$T = \underset{\substack{\text{transfer} \\ \text{function}}}{tf}(n, d)$

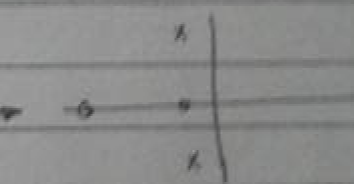
$f = tf([1 \ 0 \ 1], [1 \ 0 \ 2 \ 1])$

$$f = \frac{s^2 + 1}{s^3 + 2s + 1}$$

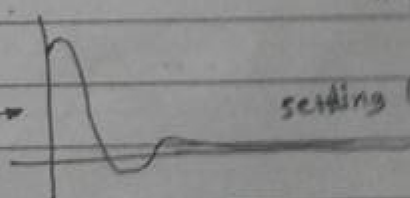
pole(T) → مقام

zero(T) → بسط

$\boxed{\text{pzmap}(T)}$ →



step(T) →



damp(T) → damping ratio nature frequency

stepinfo(T) Rise time

$$T(s) = \frac{s^2 + 3s + 2}{s^2 + 2s + 4}$$

dc gain(T) → $\frac{0.5 \text{ out}}{1 \text{ input}} = 0.5$

at time ∞ (steady state) $\boxed{s = 0}$

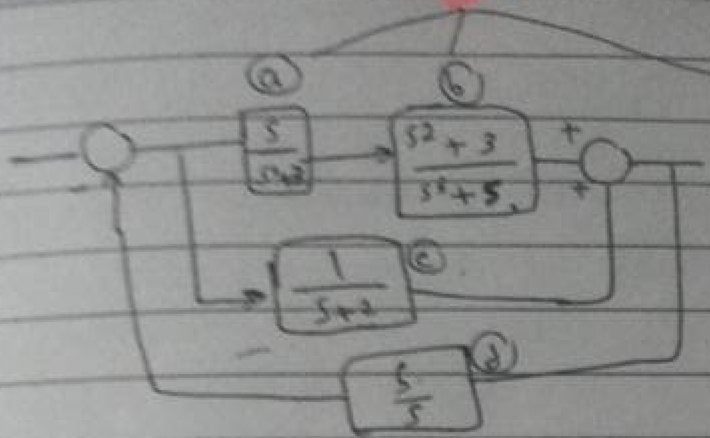
كل ما [illegible] الصفر holds on

step(5*T) $\frac{2.5 \text{ out}}{5 \text{ input}} = 0.5$

gain

input لو غيرت يفضل ثابت

①

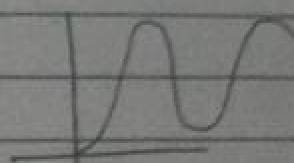


fifth order → behave

but behave as second

$\frac{s}{s^2+3} \times \frac{s^2+3}{s^2+5}$

$\frac{1}{s^2+1}$



critically damped system

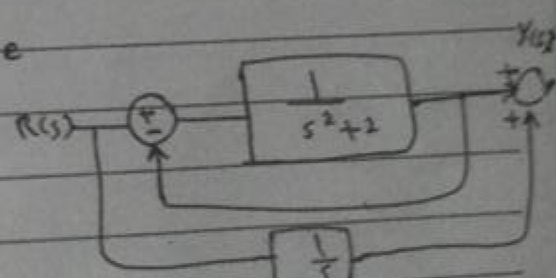
```
a = tf([1 0], [1 0 3]);
b =
```

Parallel يختلف عن feedback

```
e = series(a,b);
f = parallel(e,c);
F = feedback(f,d)
    feedback(f,d,-1) → negative feedback
    feedback(f,d,+1) → +ve
    feedback(f,d,1)
```

* إذا لم تغير الإشارة تلقائياً يكون unstable

negative → stable
+ve → unstable




```
w = minreal(e)
```

يختصر لأبسط صورة

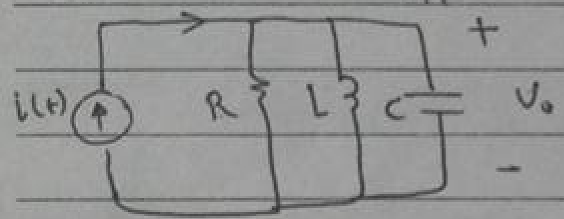
Modlening

1. electrical . KVL $\rightarrow \sum \Delta v = zero$ conservation of energy

KCL $\rightarrow \sum_{node} i = zero$ " of mass

2. mechanical system $\rightarrow$ Newtons 2^{nd}

$\sum F = ma$



$i(t) = i_R + i_L + i_C$

$= \frac{v_o}{R} + \frac{1}{L}\int V_o + C\frac{dv_o}{dt}$

$\dot{i}(t) = \frac{\dot{v}_o}{R} + \frac{1}{L}v_o + C\ddot{v}_o$

higher

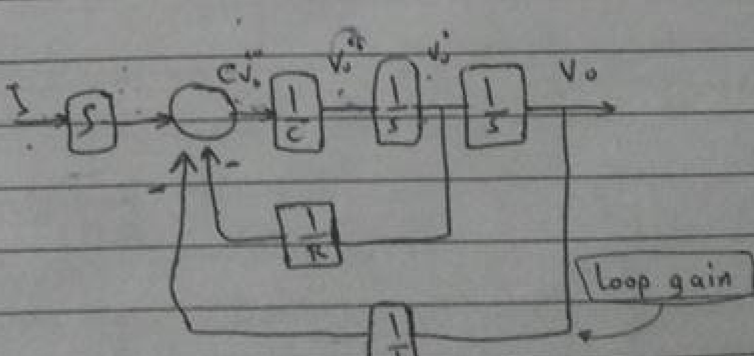
$\boxed{C\ddot{V} = \dot{I} - \frac{1}{R}\dot{V}_o - \frac{1}{L}V_o}$

TF

it must single input and single output

SISO

$C S^2 V = S I - \frac{1}{R} S V - \frac{1}{L} V$



Simulink

input from [source] Ex step

continout $\boxed{du/dt}$

math operation Ex

$$\frac{V}{I} = \frac{S}{S^2 + \frac{1}{RC}S + \frac{1}{LC}}$$

$L = 10\,mH$

$R = 1\,K\Omega$ $C = 4\,\mu F$

i step input

$\boxed{\text{simout}}$ → output on matlab workspace

save format $\boxed{\text{Array}}$

double click

plot(tout, vout)

mux

لرسم أكتر من
شكل على نفس
الرسمة

① Accuracy

$e = \text{input} - \text{output}$

② Disturbance rejection

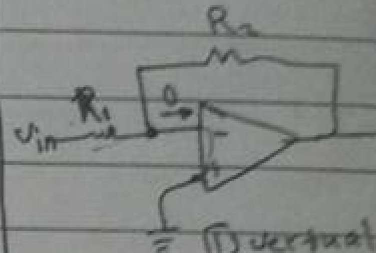
$DR = \dfrac{V_{oi}}{V_{od}}$

$V_o = V_{oi} + V_{od}$

③ $S_G^T = \dfrac{\Delta T}{\Delta G}\dfrac{G}{T} = \dfrac{\Delta T/T}{\Delta G/G}$

$S_{G_4}^T = \dfrac{\delta T}{\delta G_4}\dfrac{G}{T}$

sensitivity for the system when S is open



① virtual short
Vnon = Vinverter

② I input = zero

③ KCL at each node

Home work *
G1, G2

open

$V_{oi} = G_1 G_2 V_{in}$

$V_{od} = G_2 V_d$

$V_o = G_1 G_2 V_{in} + G_2 V_d$

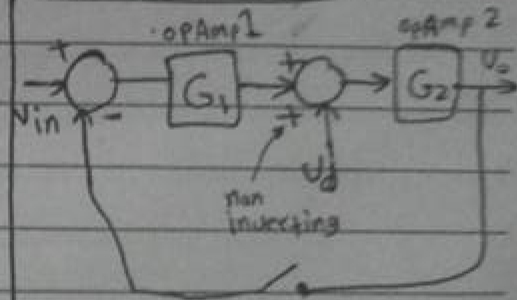
closed

$V_{oi} = \dfrac{G_1 G_2}{1 + G_1 G_2} V_{in}$

$V_{od} = \dfrac{G_2}{1 + G_1 G_2} V_d$

open: $DR = G_1 \dfrac{V_{in}}{V_d}$ closed: $DR = G_1 \dfrac{V_{in}}{V_d}$

① $\underbrace{\qquad\qquad}_{=}$



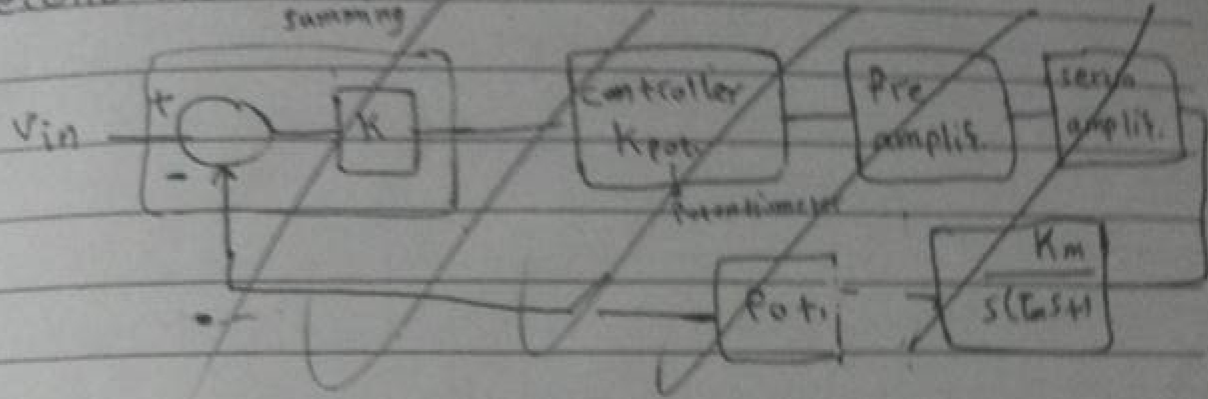
super position
assume $V_d = 0$

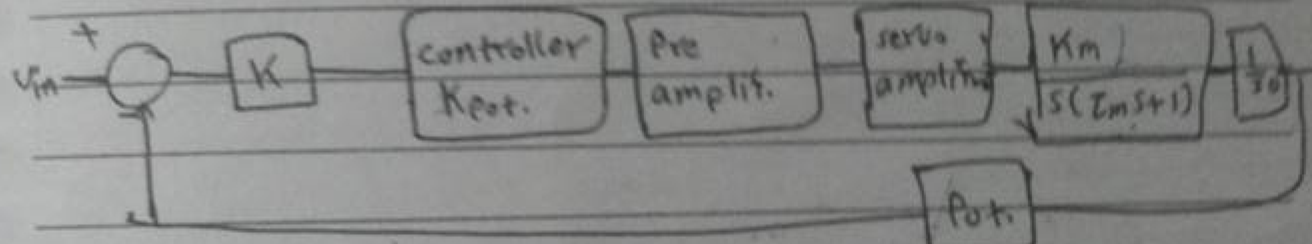
$V_{oi} = \_\_\_ V_{in}$ ①

assume V_i = zero

$V_{od} = \_\_\_ V_d$ ②

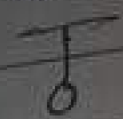
second order



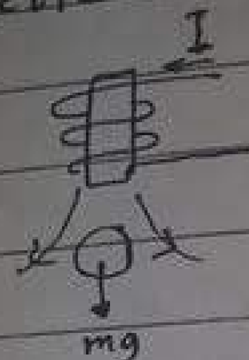


$$-\frac{R_1 \| R_2}{R_1 + R_2}$$

1) Inverted Pendulum



2) Magnetic levetation



$$T(s) = \frac{y}{I} = \frac{-a}{s^2 - b}$$

distance

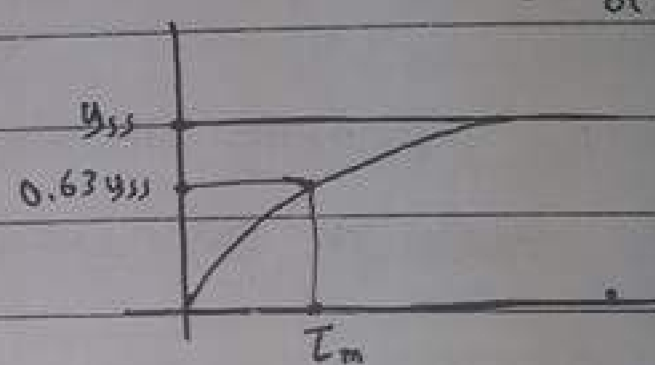
current

3) DC Motor "speed"

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

$$= \frac{17.2}{66s + 1}$$

1st order



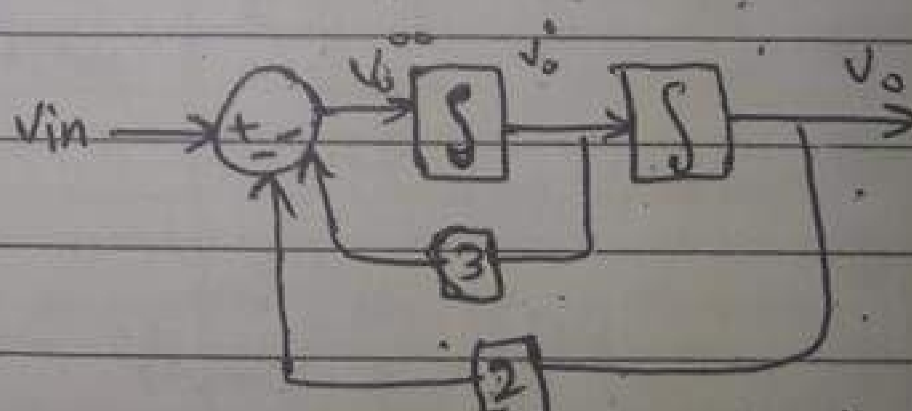
$$K_m = \frac{y_{ss}}{I/P}$$

$\tau_m \longrightarrow 0.63\,y_{ss}$

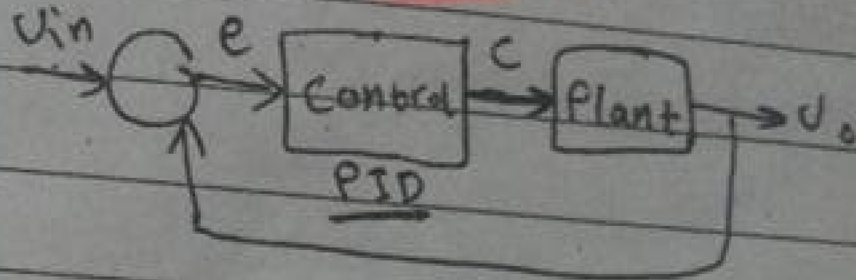
$$\frac{V_o}{V_{in}} = \frac{1}{s^2 + 3s + 2}$$

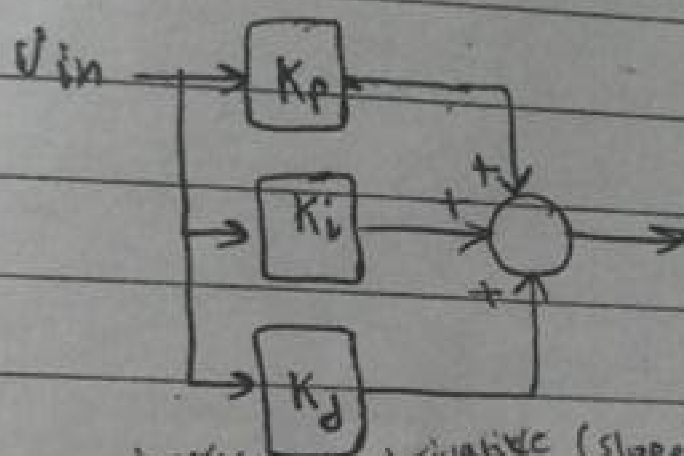
$$\ddot{v}_o + 3\dot{v}_o + 2v_o = V_{in}$$

$$\ddot{v}_o = V_{in} - 3\dot{v}_o - 2v_o$$



PID_c:





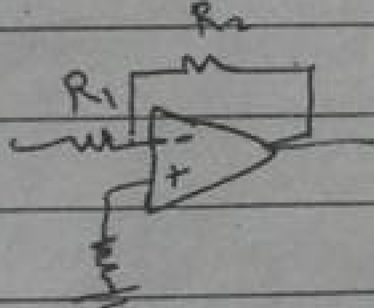
integr. (Area)   derivative (slope)

$$\frac{V_o}{V_{in}} = \left(K_p + \frac{K_i}{s} + K_d s\right)$$

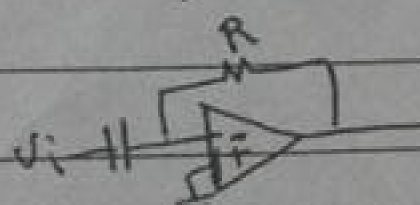
K_p → without time

$\frac{K_i}{s}$ → past time (eleminate steady state error) $e_{ss} = 0$

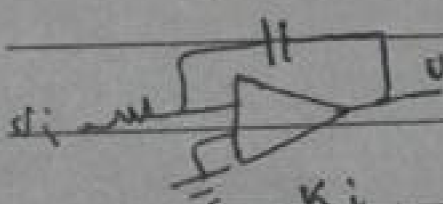
$K_d s$ → future time (If the gain increase error increase)



K_p



$\frac{R}{\frac{1}{CS}} \Rightarrow RCS$ → Kd



$V_o = -\frac{1}{RCS}$

Ki

prepared by Eng. Sarah AL-Bargothi

Tuning the PID Controller

Tuning the PID controller can be like learning to roller blade, ski or maybe riding a bull. Until you've done it a few times, the literature you've read really doesn't hit home. But after few attempts (and falls), you find it wasn't so bad after all - in fact it was kind of fun!

Although you'll find many methods and theories on tuning a PID, here's a straight forward approach to get you up and soloing quickly.

1. SET KP. Starting with KP=0, KI=0 and KD=0, increase KP until the output starts overshooting and ringing significantly.

2. SET KD. Increase KD until the overshoot is reduced to an acceptable level.

3. SET KI. Increase KI until the final error is equal to zero.

the other two. For this reason, the table should only be used as a reference when you are determining the values for Ki, Kp and Kd.

Implementing the PID Controller Using Op-Amp

Figure 1 shows how to separately implement a P, D and an I controllers.

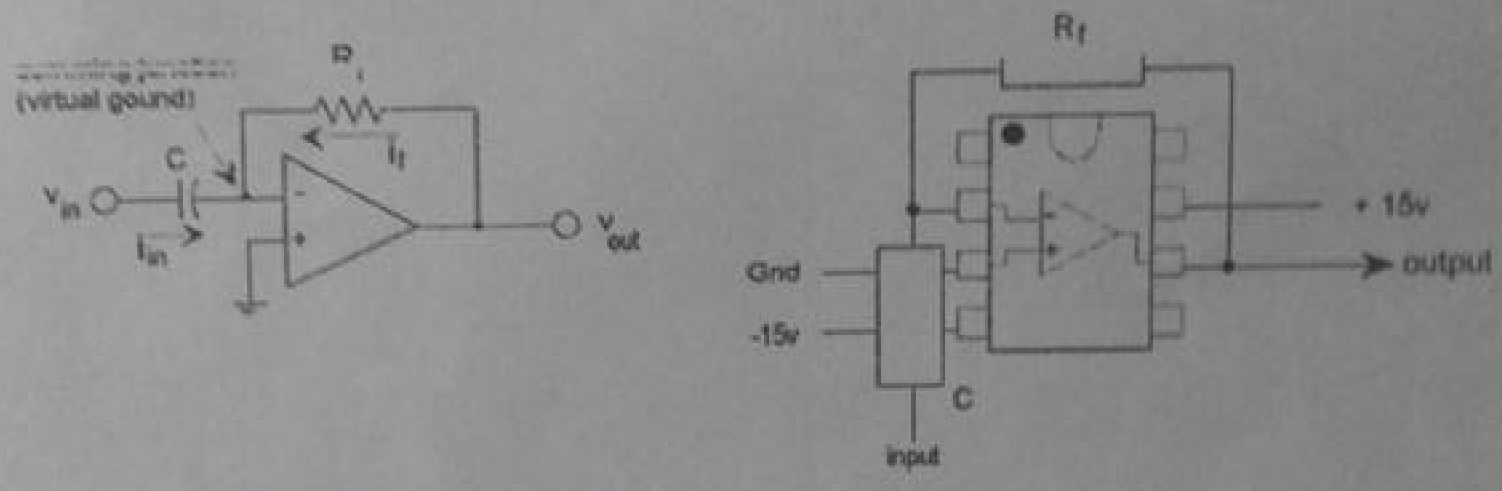


Figure 1.a

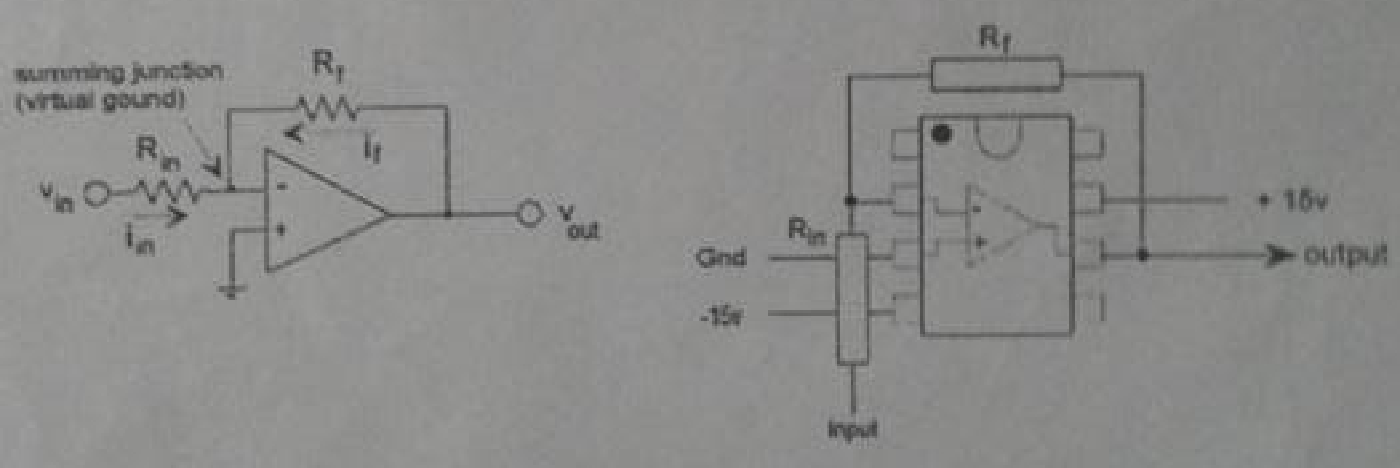


Figure 1.b

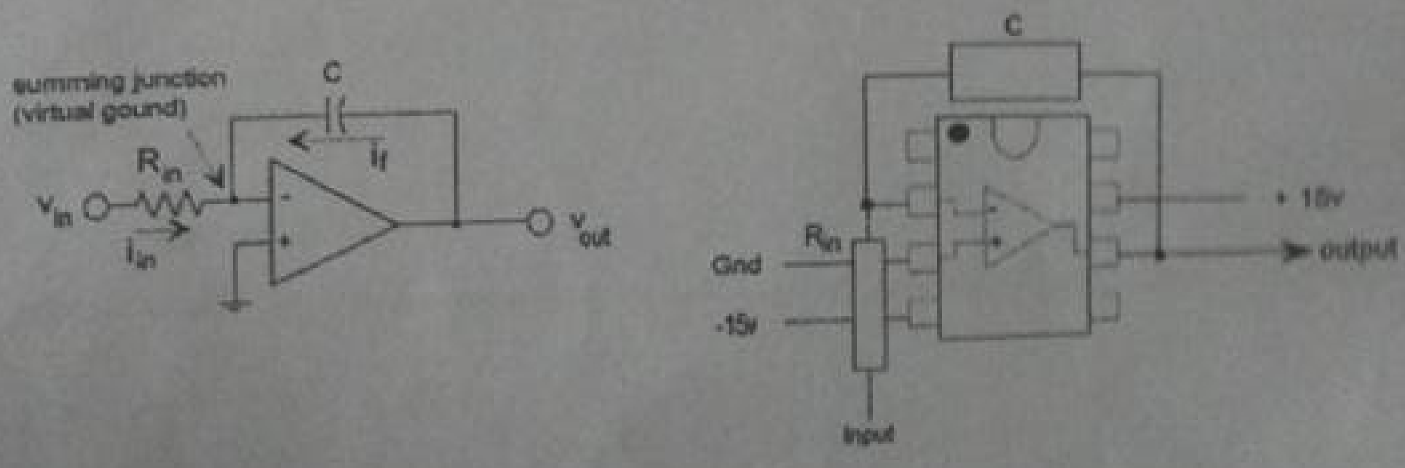


Figure 1.c

prepared by Eng. Sarah AL-Bargothi

Tuning the PID Controller

Tuning the PID controller can be like learning to roller blade, ski or maybe riding a bull. Until you've done it a few times, the literature you've read really doesn't hit home. But after few attempts (and falls), you find it wasn't so bad after all - in fact it was kind of fun!

Although you'll find many methods and theories on tuning a PID, here's a straight forward approach to get you up and soloing quickly.

1. SET KP. Starting with KP=0, KI=0 and KD=0, increase KP until the output starts overshooting and ringing significantly.

2. SET KD. Increase KD until the overshoot is reduced to an acceptable level.

3. SET KI. Increase KI until the final error is equal to zero.