Details Note

Theme: "Control Theory". Date: / - /

Sat Sun Mon Tue wed Thu Fri

Defination of Control Theory

Control theory is a branch of mathematical and engineering that deals with the behavior of dynamical systems.

Intuition → It focuses on designing system that can control the behavior of other systems to produce specific outputs.

Example: Line following robot. The controller decides how much to adjust based on how far the robot is from the line.

Types of control system:

Type-01

→ Open loop control system (Non-feedback Control)

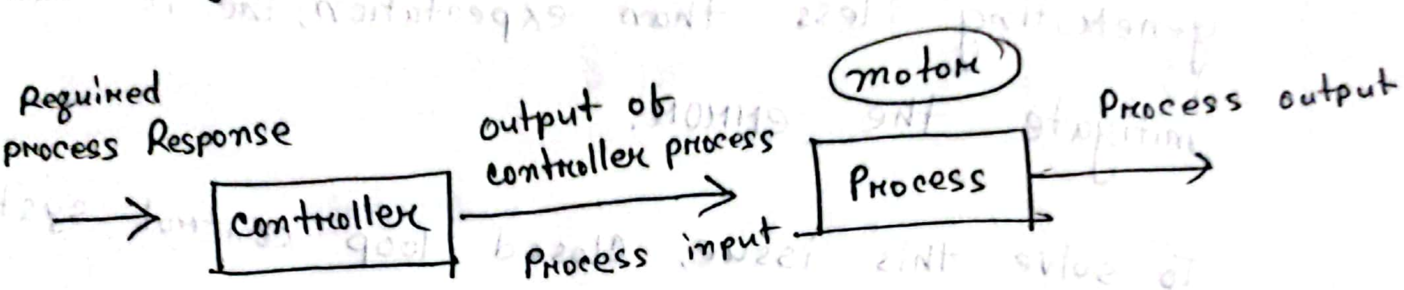
In an open loop control system, the controller sends commands to the actuator and the robot executes them without any feedback.

Advantage: Simple and inexpensive

Disadvantage: Lack of accuracy and adaptability.

Example: A robot arm that moves a certain distance without any feedback.

Working Mechanism:

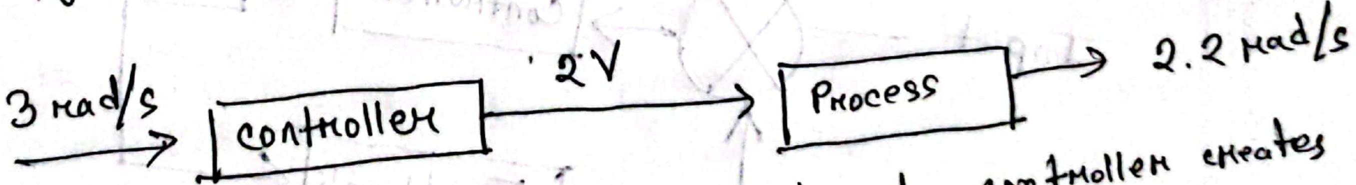


Now,

From the process output, we want to get 3 rad/s that is the desired value we gave as a input. (Required process Response = Input)

N:B → process = Motor,

So, we will give 3 rad/s as a input and we will expect that from the process output, we will get 3 rad/s. ~~Bu~~ This things controlled by an algorithm which is set inside the controller. Now if we give 3rad/s as a input,



Lets assume, for the given input, controller creates 2V which will go in the process and finally the output will be 2.2 ~~V~~ rad/s because of the

Theme:

Date: / /

Sat Sun Mon Tue wed Thu Fri

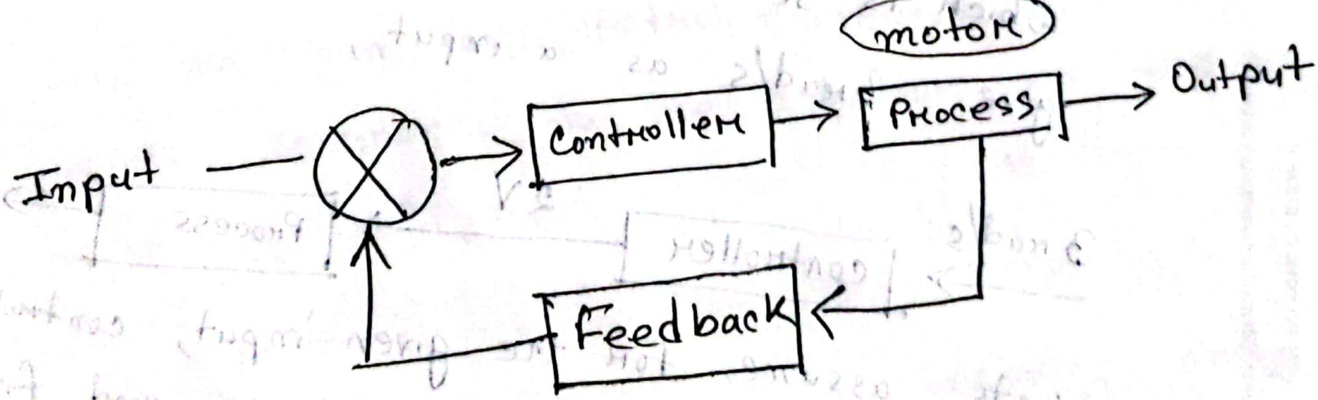
disturbance which we created on the motor. But as there are no options to give this msg to the controller that ~~p~~ output process is generating less than expectation, ~~the~~ it cannot mitigate the error.

To solve this issue, closed loop control system has been introduced.

Type-02

Closed Loop Control system:

In a closed loop system, the controller continuosly receives feedback from sensors and adjust the command.



Advantage: Ability to correct disturbance, greater precision and adaptability.

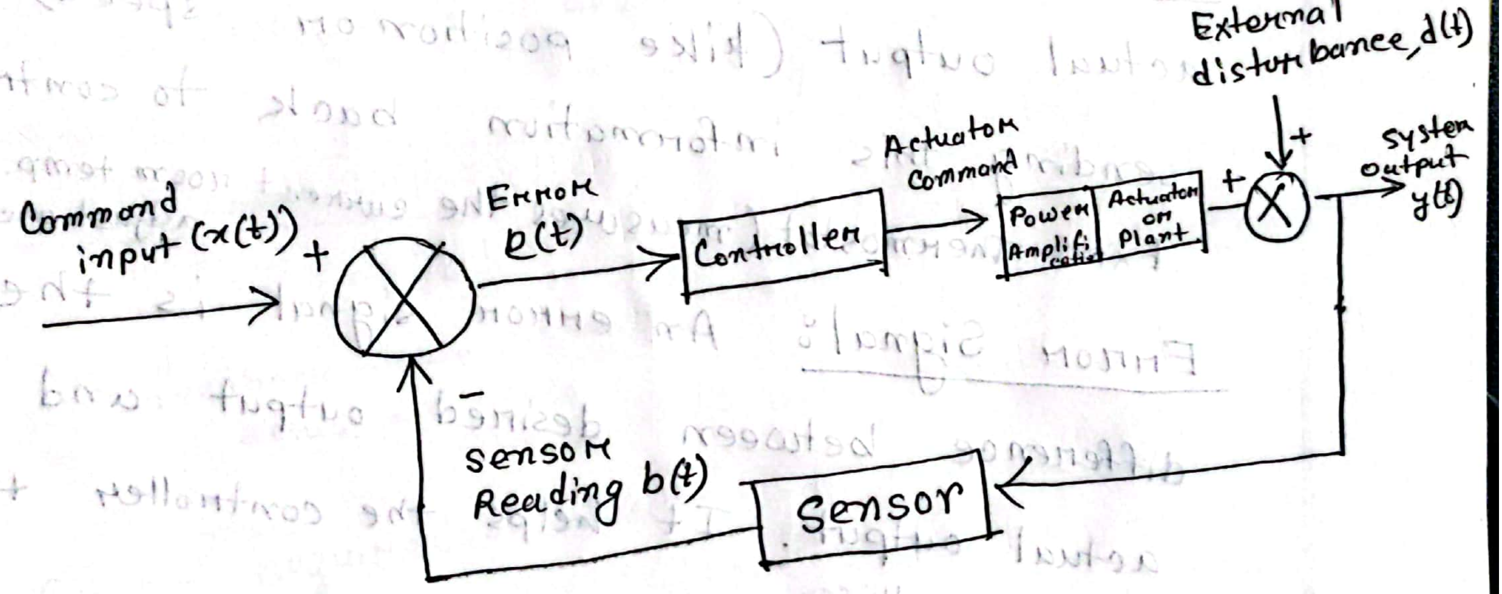
CamScanner

Disadvantage: More complex and costly.

Example: A self balancing robot.

Working Mechanism:

Feedback diagram



Characteristics of feedback:

~~Amp~~ **Power amplification:** It boosts low power signal to high power level so that even if ~~at~~ ~~Actuator or motor/~~ controller sends low power, ~~process can)~~ system can control larger mechanical outputs.

Theme:

Date: / /

Sat Sun Mon Tue wed Thu Fri

Actuator: An actuator is a device that can takes the control signal ~~and converts~~ from controller and converts them into physical action. Example: movement or force.

Feedback: It is the process of measuring actual output (like position or speed) and sending this information back to controller.
Exp: Thermostat (measures the current room temp. and adjust accordingly)

Error Signal: An error signal is the difference between desired output and actual output. It helps the controller to minimize this error.

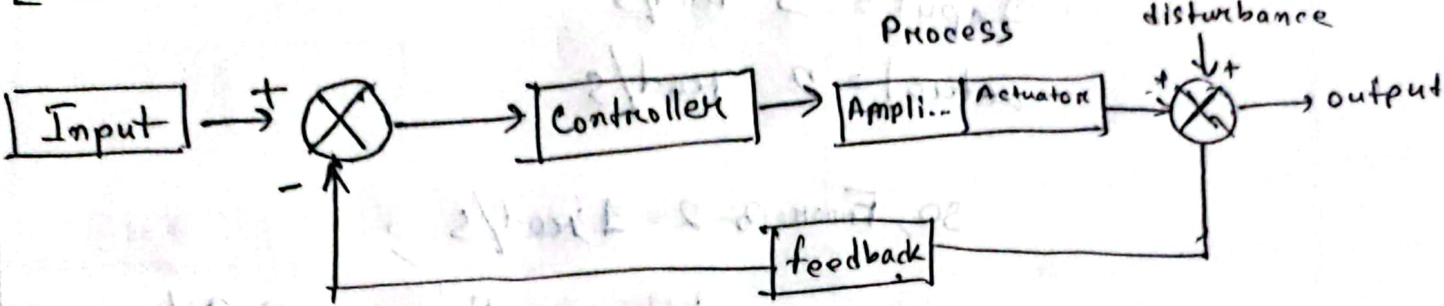
Controller: A controller in closed loop system processes the error signal and generates the appropriate control actions to bring the system's actual output closer to the desired value.

Theme: Date: / /

Sat Sun Mon Tue wed Thu Fri

Mathematical intuition of Closed-loop Control system:

[N:B ⇒ All the values are assumption based to clear the concept]:



Our goal → $\boxed{\text{required} = \text{actual}}$

Let's assume,
input = 3 rad/s

Initial ~~steps~~ loop

input = 3 rad/s
feedback value = 0 rad/s
actual = 0 rad/s

So,
Error = required - actual = 3 − 0
= 3 rad/s

Then, Controller update voltage to 0.5V
and actual output produce = 1 rad/s (assumption)

Second ~~sp~~ ~~step:~~ loop:

input = 3 rad/s
actual = 1 rad/s
Error produce, 3 − 1 = 2 rad/s; controller update voltage = 1.2V

Let's assume actual = 2 rad/s.

Theme:

Date: / /

Sat Sun Mon Tue wed Thu Fri

Third loop:

Input = 3 rad/s
actual = 2 rad/s

so, Error = 3 − 2 = 1 rad/s;

controller update voltage 2.2V.

Actual = 3.3 rad/s

As our goal was to reach 3 rad/s but our value crossed it. Now,

Fourth loop

input = 3
actual = 3.3

Error = 3 − 3.3 = −0.3 !

so, controller decrease voltage to 2V,

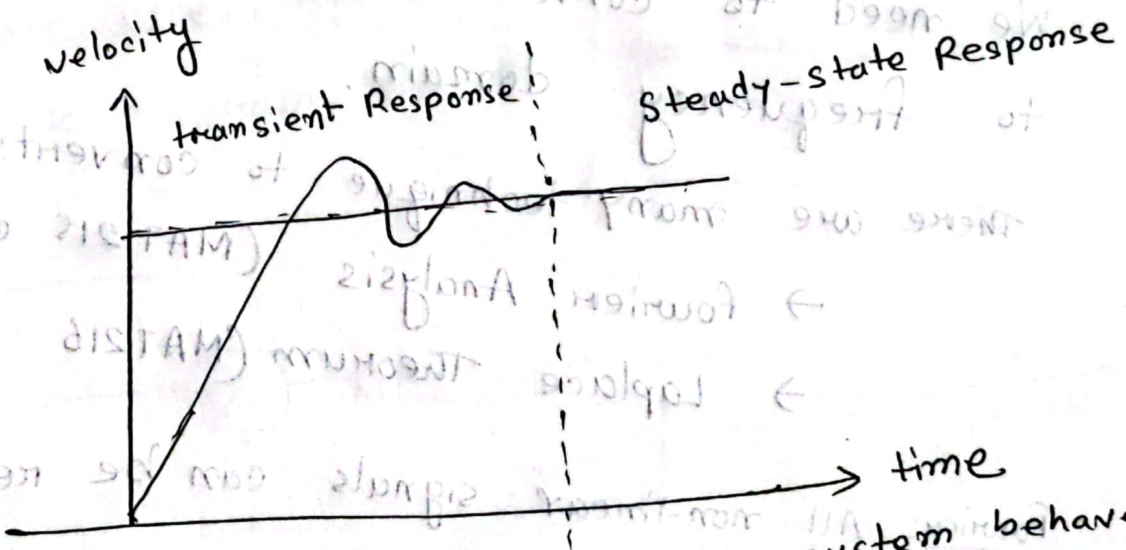
$\boxed{\text{actual} = 3V = \text{Required}}$.

So,

$\boxed{\text{Error} = \text{Required value} - \text{Actual value}}$ **

Stability: Stability ensures, the system's response remained bounded over time, even in the presence of disturbances or changes in operating conditions. Exp: balancing pencil on the tip of a finger. Eventually pencils come to equillibrium position even though there are disturbance.

Ex



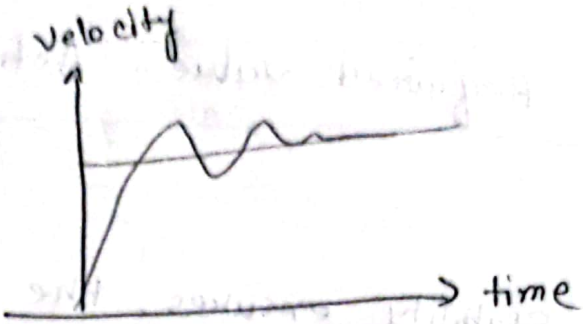
Transient Response: Refers to how system behaves during the transition. Exp: Turning on a light switch.

Steady State Response: Describes the behavior of the system once it is settled.
Exp: Elevator.

Theme:

Date: / /

Sat Sun Mon Tue wed Thu Fri



In the x-axis we can see time domain which is not appropriate in every cases. ~~So, we need~~ Because, it can be difficult to analyze complex system behaviors directly from time-based signals. So,

We need to convert from time domain to frequency domain.

There are many technique to convert:

(MAT 215 course content)

→ fourier Analysis (MAT 215 " ").

→ Laplace Theorum (MAT 216 " ").

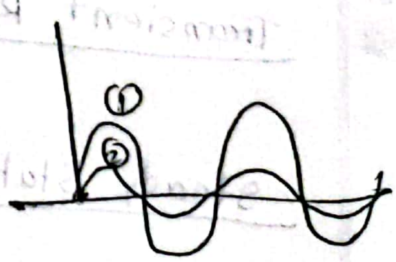
Fourier: All non-linear signals can be represented by sine and cosine.

$A_1 \sin w_1 t \rightarrow (A_1, w_1)$

$A_2 \sin w_2 t \rightarrow (A_2, w_2)$

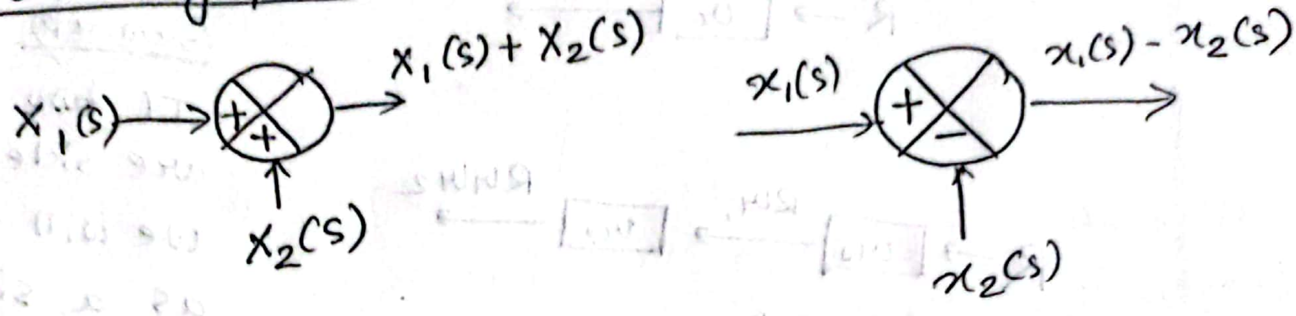
$\approx A_n \sin w_n t + A_{n-1} \sin w_{n-1} t + \cdots$

$+ A_n \cos w_n t + A_{n-1} \cos w_{n-1} t + \cdots$



"Block Diagram"

summing point:



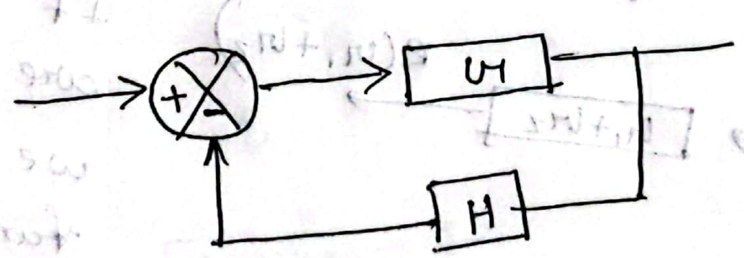
Special Mention:

→ Normally 2 inputs but can be 3 in some cases!

→ Arrow inwards!

There are intotal 7 rules to solve a block diagram. We will go through all.

Rule-01



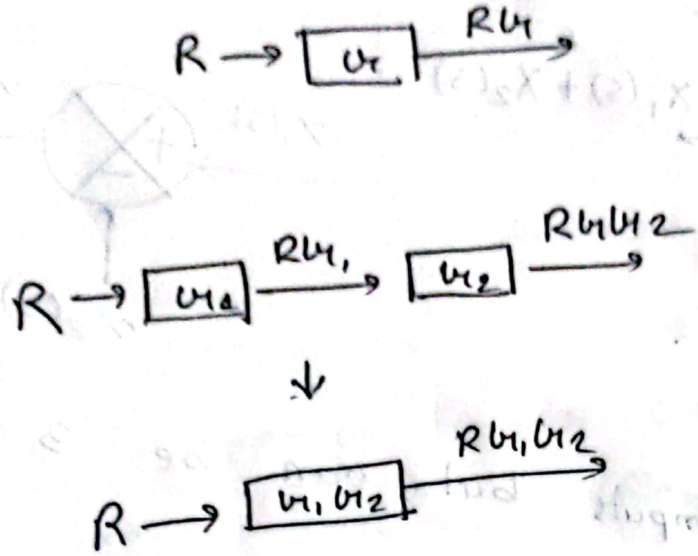
$$\rightarrow \frac{G}{1+GH}$$

If signs are different, we will add in the denominator otherwise, if signs are same, we will subtract in the denomintor like $\left(\frac{G}{1-GH}\right)$ this.

Theme: Date: / /

Sat Sun Mon Tue wed Thu Fri

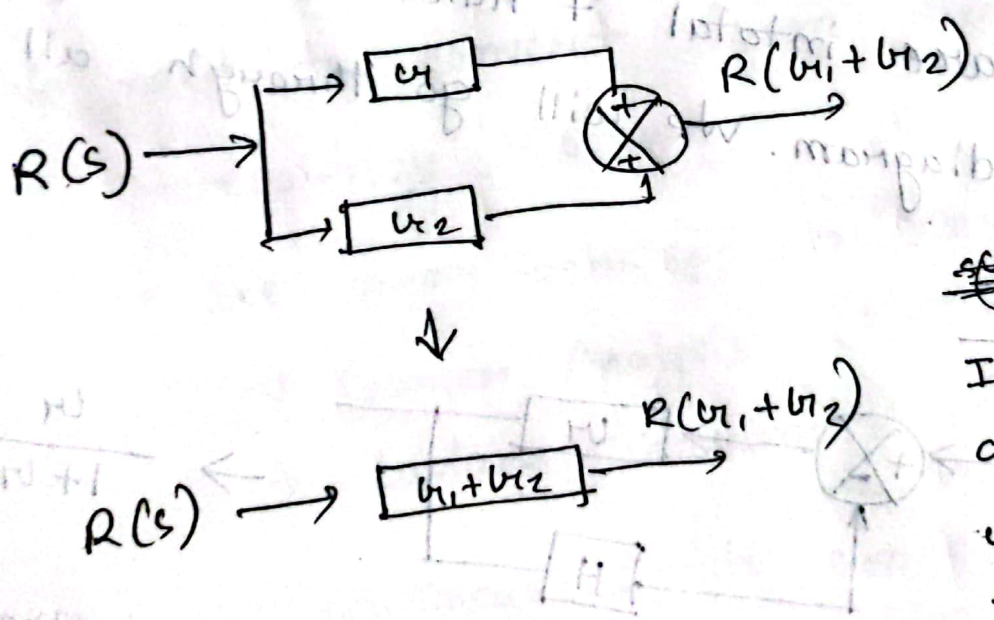
Rule-02



~~(Sum up)~~

If two blocks are side by side, we will make them as a single block and multiply them.

Rule: 03



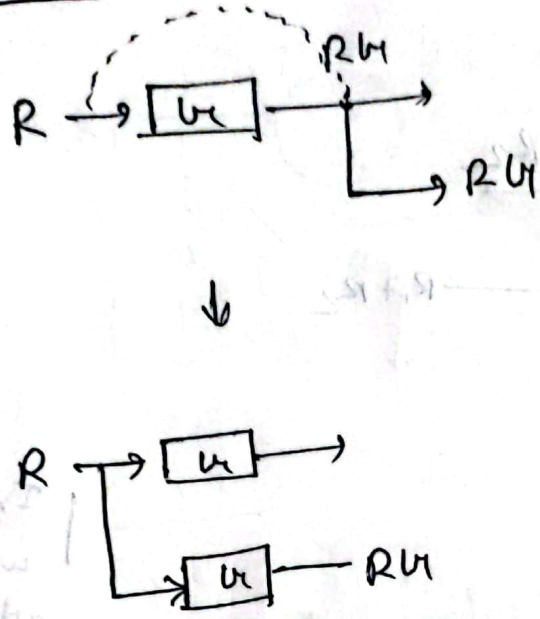
~~(Sum up)~~

If two blocks are in parallel we can add the function inside and make them as a single block.

Theme: Date: / /

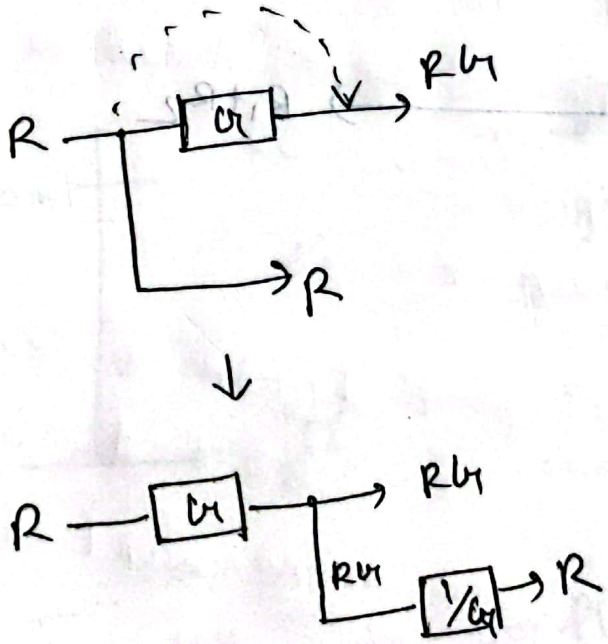
Sat Sun Mon Tue wed Thu Fri

Rule-04



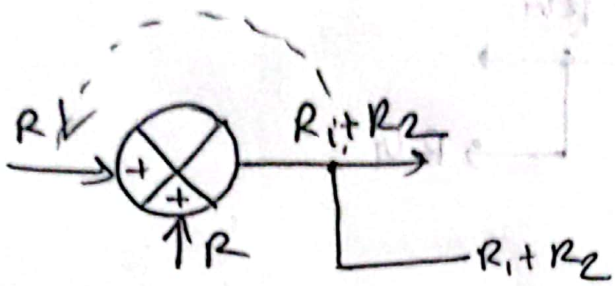
If we want to shift the ~~right~~ right side value of the block to the left side, we need to add another block.

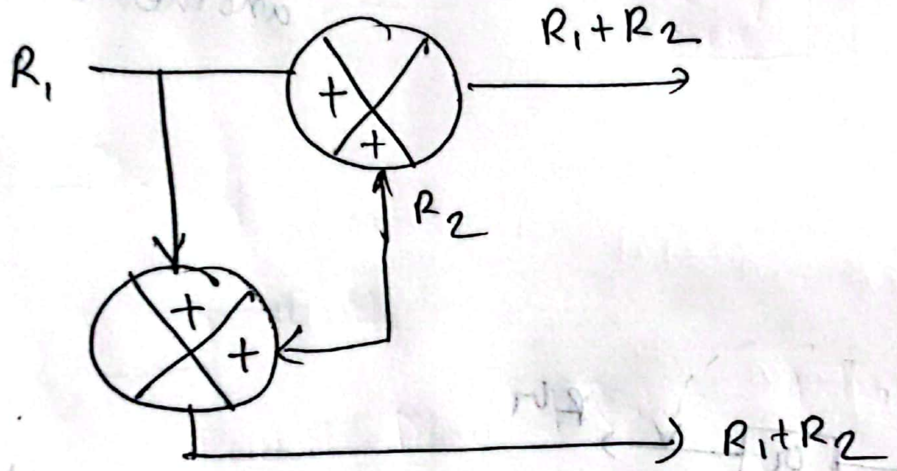
Rule-05



~~to~~ to shift the left side value of the block to the right side, we need to devide the gain.

Rule - 06





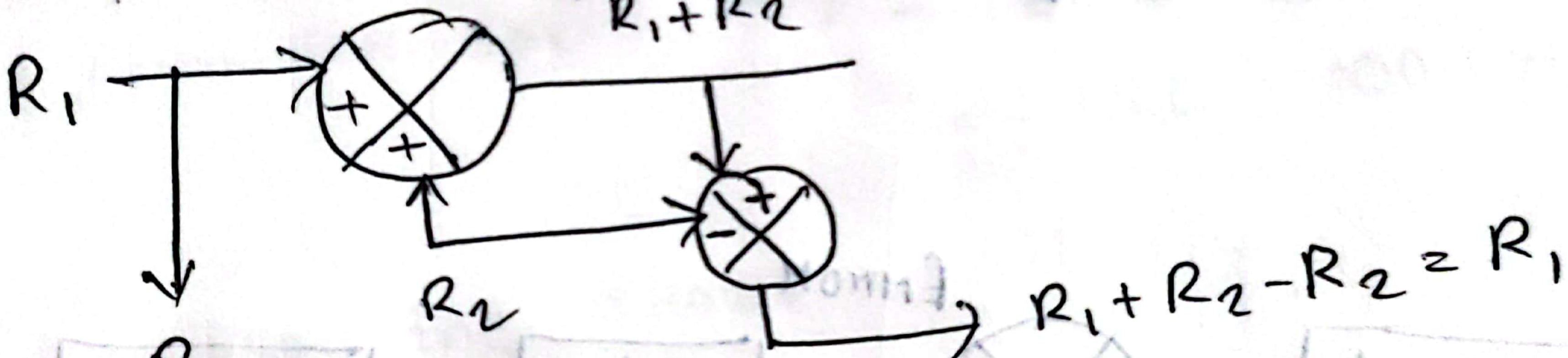
In this case, we need to add another summing point.

Theme:

Date: / /

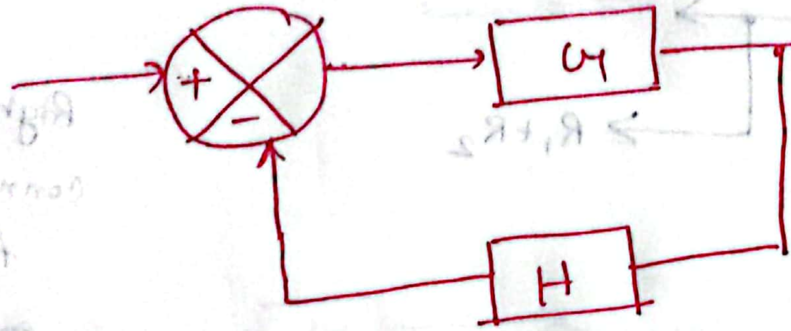
☐ Sat ☐ Sun ☐ Mon ☐ Tue ☐ wed ☐ Thu ☐ Fri

Rule-07



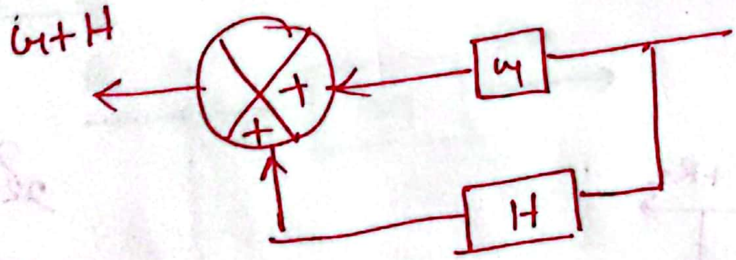
(proportional- inte

Intuition (Essential for basic)



$$\frac{G}{1+GH}$$

যদি একটা block থেকে input আর একটা block থেকে output হয়, তখন হবে $\frac{G}{1+GH}$ (Rule-01)



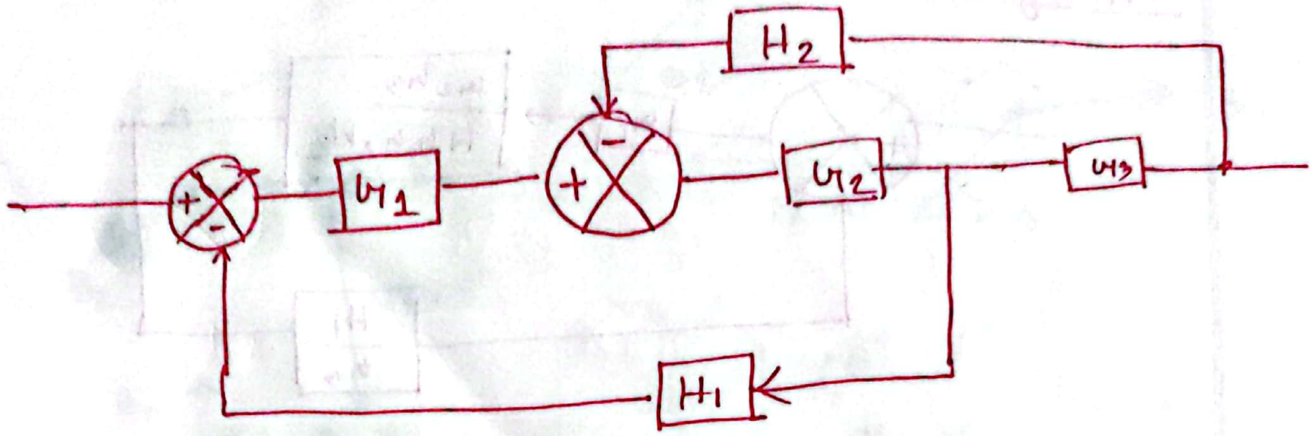
দুইটা block থেকে input গেলে (+)

Theme:

Date: / /

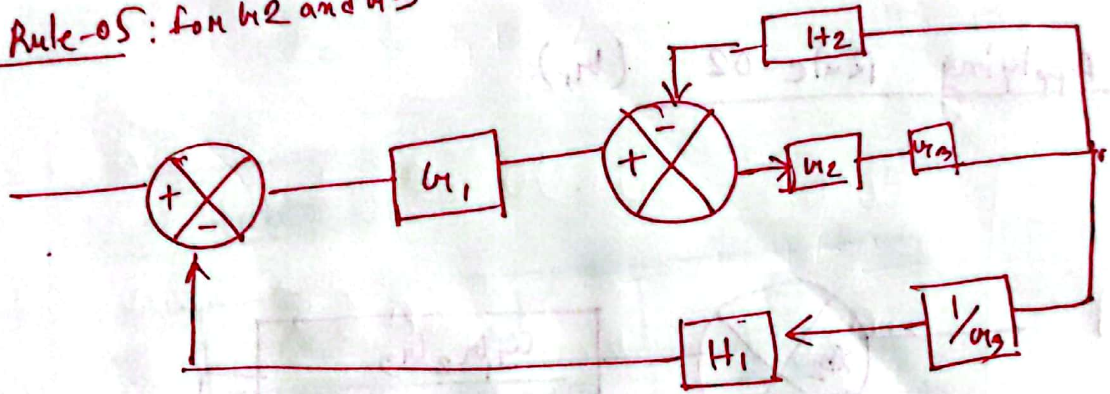
☐ Sat ☐ Sun ☐ Mon ☐ Tue ☐ wed ☐ Thu ☐ Fri

Problem:

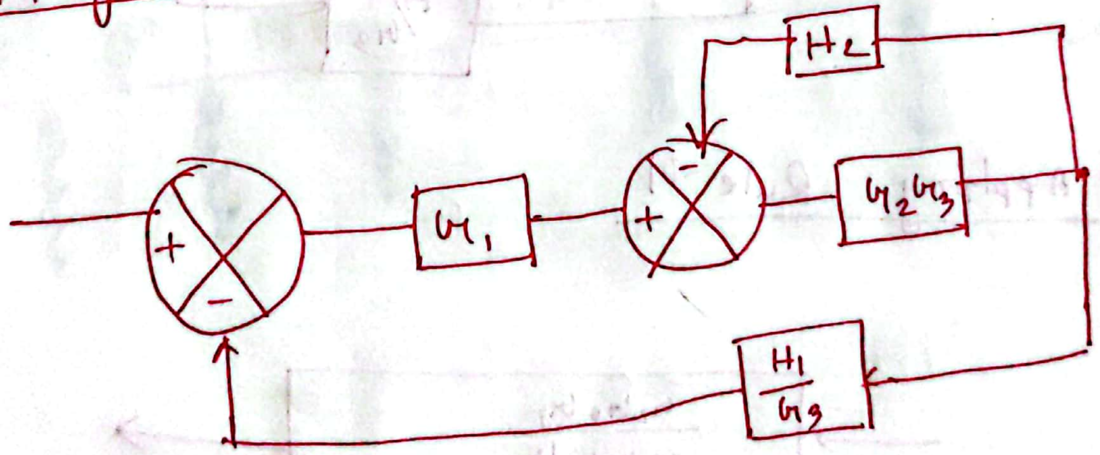


Applying Rule-05: for G2 and G3

⇒



⇒ Applying Rule-02

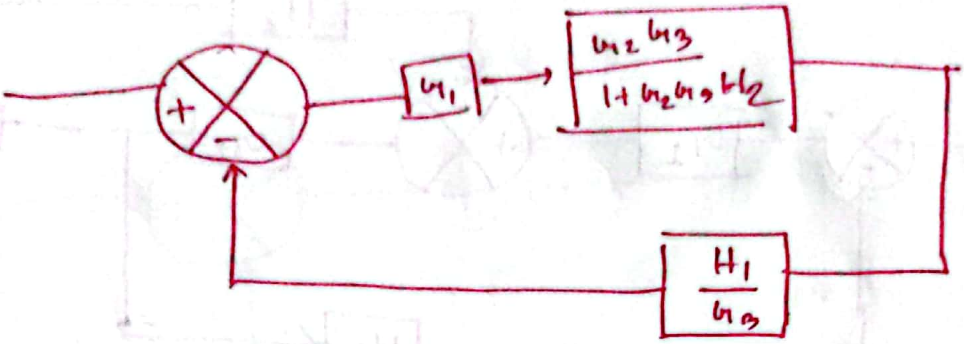


Date: / /

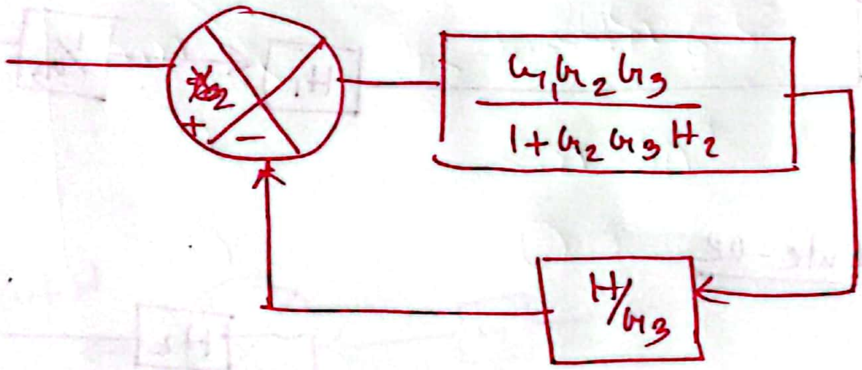
Sat Sun Mon Tue wed Thu Fri

Theme:

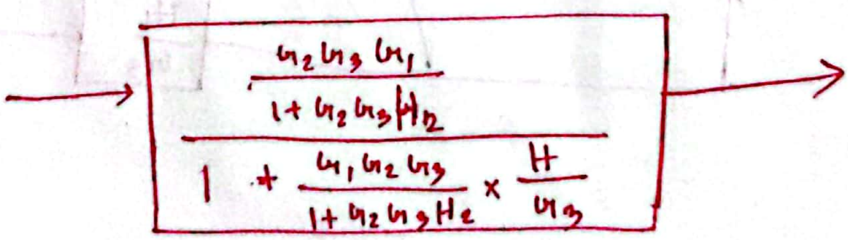
Applying Rule -1 (For H_2)



Applying Rule-02 (G_1)

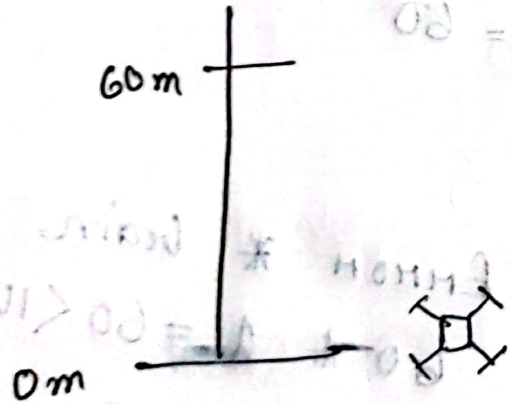


Applying Rule-01



PID Control Theory (proportional-integral-derivative control).

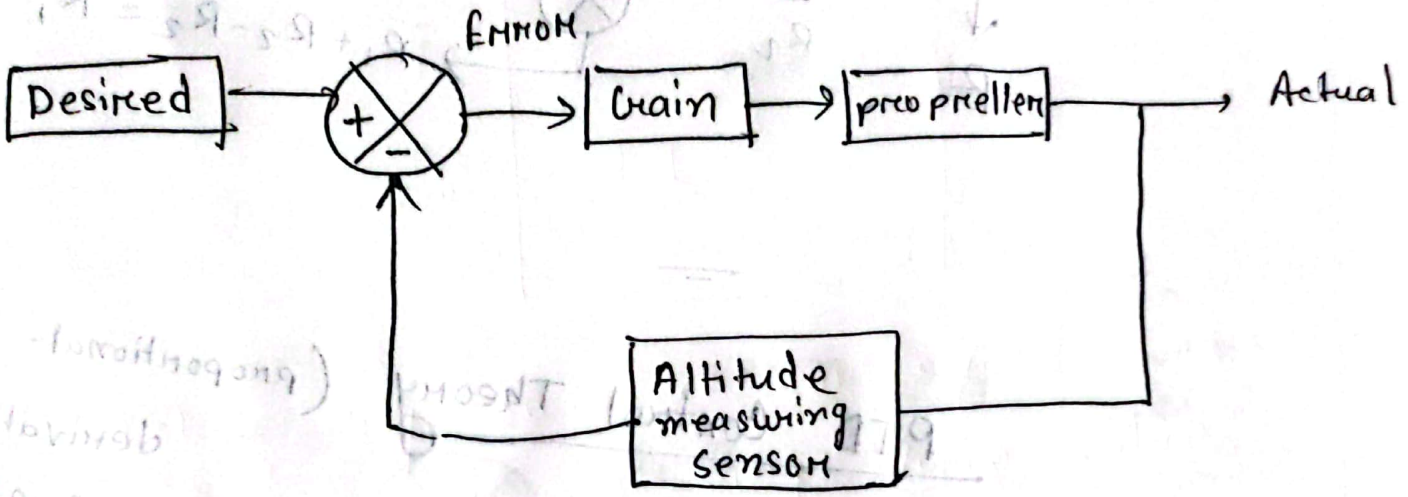
We will try to understand the entire concept with a task:



There is a drone on the 0m height and our goal is to fly that drone higher so that it can reach the ~~de~~ goal!.

[Condition; to ~~pl~~ fly, propeller speed minimum 100 rpm

[N:B: For solving this problem, we will take the idea of closed loop control system]



1st loop:

Desired = 60m

Actual = 0m

Error = Desired value - Actual value

Error = 60 - 0 = 60

Suppose, Gain = 1

propeller speed = Error * Gain

= 60 * 1 = 60 < 100 rpm

So, the drone won't fly.

Theme: Date: / /

Sat Sun Mon Tue wed Thu Fri

Suppose, Gain = 2

Propeller speed (P.S) = $60 * 2$

$= 120 > 100$ rpm

So the drone will fly.

In that case, suppose, the actual value produced 5m.

Now, Desired = 60

Actual = 5

$\therefore$ Error = $60-5$

$= 55$

$\therefore$ P.S = $55 * 2$

$= 110$ rpm > 100 rpm

So, the robot will fly.

Then, actual value produced suppose 10m.

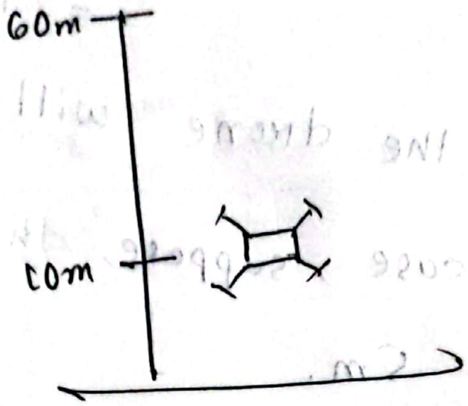
So, error = $(60-10)$

$= 50$

P.S = $50 * 2 = 100$ rpm

CamScanner

So, for gain = 2; maximum height = 10m



Let's try to make our system more efficient by increasing Gain value.

Assuming, Gain = 10

Desired = 60m

Actual = 10m

Error = 60 - 10
= 50

1st iteration

∴ P. S = 600 rpm > 100 rpm (will fly)

Assuming Actual value = 10m

Now, Error = (60 - 10)
= 50

2nd iteration

So, P.S = 50*10 = 500 rpm > 100 rpm (will fly)

Now, Assuming, ~~train = 20~~ Actual = 20

Error = 60 - 20
= 40

3rd iteration

P.S = 40*10 = 400 > 100 rpm will fly

Again, Actual = 30

Error = 60 - 30
= 30

4th iteration

P.S = 300 > 1 rpm

Actual = 40

Error = 60 - 40
= 20

5th iteration:

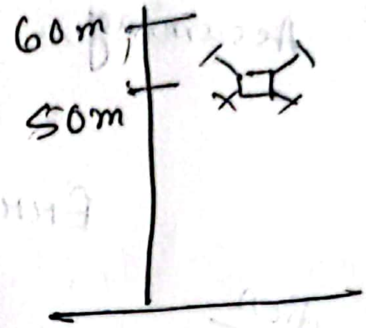
∴ P.S = 20*10 = 200 > 100 rpm

Actual = 50

∴ Error = 60 - 50 = 10

6th iteration P.S = 10*10 = 100 rpm

So, drone can reach upto 50m for gain 10



We know,

Propeller speed = Gain * Error

$$\therefore \text{Error} = \frac{100}{\text{Gain}}$$

for exact 60m reach,

error = 0

so that we can reach upto 60m.

But, $\text{Gain} = \frac{100}{\text{error}}$

$$= \frac{100}{0} = \infty$$

so, it will be infinity gain if we make error 0. So, we can't

Theme: Date: / /

Sat Sun Mon Tue wed Thu Fri

make error 0.

Let's take, error = 1

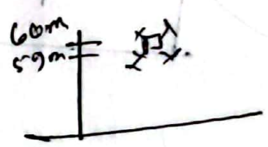
$$\therefore \text{Gain} = \frac{100}{1}$$

$$= 100$$

$$\boxed{\text{Actual} = \cancel{60 - 1}\ \text{Desired} - \text{Error}}$$

$$\text{Actual} = 60 - 1$$

$$= 59$$ (close to height 60)



Now, what if we take Gain = 110

$$G \times e = 100$$

$$\therefore e = \frac{100}{110}$$

$$= 0.91$$

$$\text{Actual} = 59.09$$

drone will fly more. Then ~~Actual~~ propagation speed will become negative. ~~Ae~~ Drone will change its direction and come downwards. Then again, for G=110, it will fly

Date: / /

Sat Sun Mon Tue wed Thu Fri

Theme:

up. In this way, it will keep moving up and down.

$$P = G \times E$$

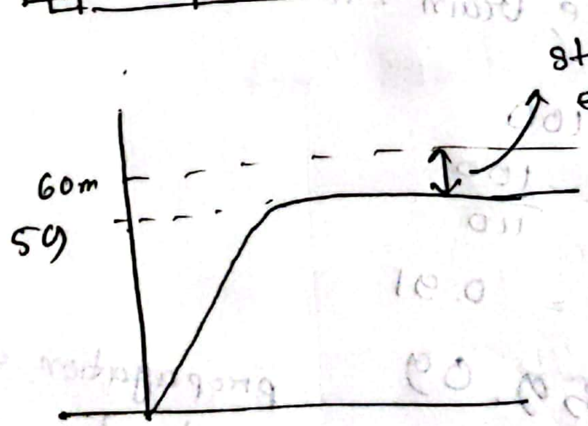
Here, G is a constant value. So,

$$P \propto E$$

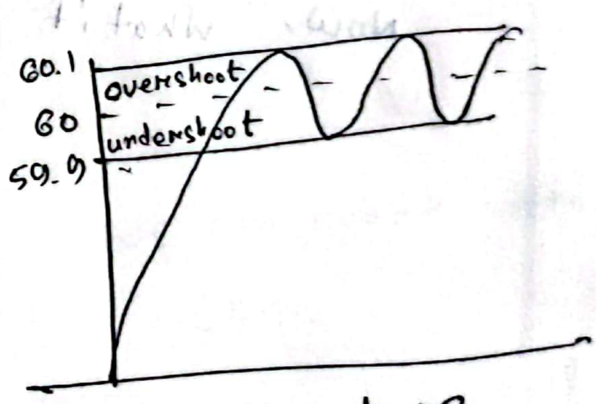
formula for proportional control

$\boxed{\text{Output, } P = K_p \times \text{Error}}$ | K_p = constant proportional.

Proportional Control:



Graph-01



Graph-02

As we discussed above, if we increase value of gain extremely, it will fluctuate.

To solve this issue, we will introduce Integral control along with proportional control, which combinedly makes PI control.

PI Control: (Proportional - Integral control)

$$\text{Output} = \overbrace{K_p * e}^{\text{proportional control}} + \overbrace{K_i * \Sigma e}^{\text{integral control.}}$$

$$= K_p * \text{error} + K_i * (\text{error}1 + \text{error}2 + \text{error}3 + \ldots)$$

In this method, the drone will fly higher. Once it passes 60m, errors will become negative and eventually it will come down. So, it is solving the issue of fluctuation problem of P-control but PI control has a problem. It overshoots.

To solve this problem, we will introduce PID ~~prob~~ control.

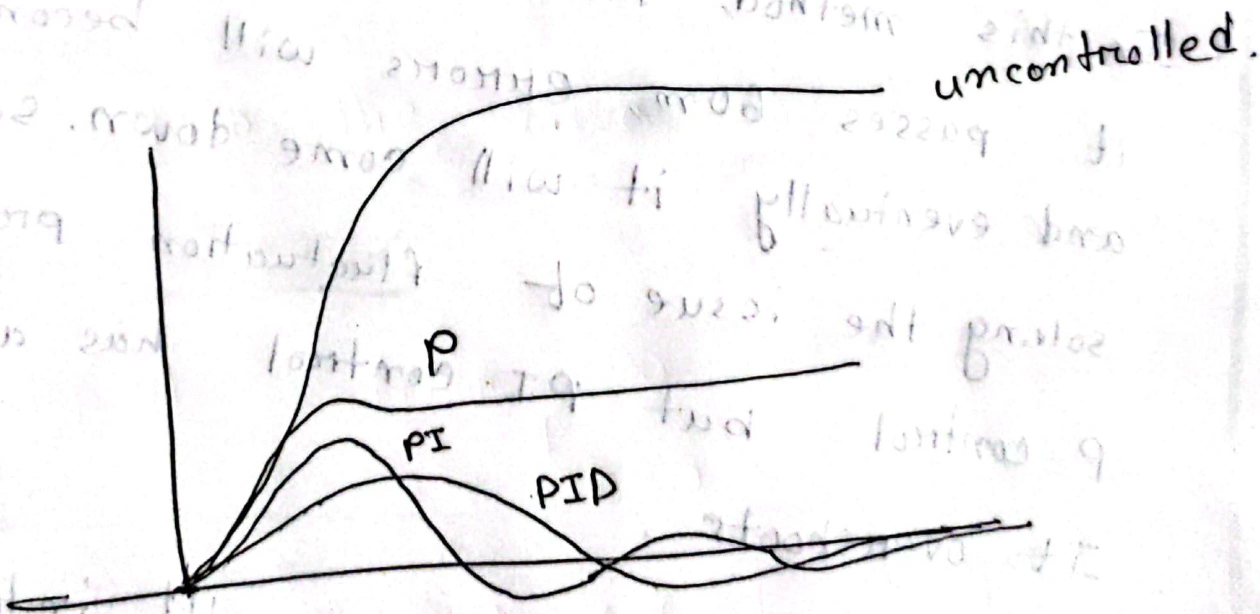
Theme: Date: / /

Sat Sun Mon Tue wed Thu Fri

PID Control: (Proportional - Integrated - Derivative Control).

Output = $K_p \times$ Error + $K_i *$ Σ Error + $K_d *$ (Δ Error)

proportional and integral control pushes drone upwards and Derivative control pushes drone downwards. It solves the overshoot problem.



Theme: Date: / /

Sat Sun Mon Tue wed Thu Fri

How to get PID parameter value?

2 ways.

1. Manually
2. Ziegler Nichols Rule.

⇒ Ziegler Nichols Rules for Tunning PID controller:

for P control

$K_p = 0.5 \times K$

PI control

$K_p = 0.45 \times K$

$K_i = \frac{1.2}{P}$

PID control

$K_p = 0.6 \times K$

$K_i = \frac{2.0}{P}$

$K_d = P/8.0$

Here, K = result in gain

P = Oscillation period.

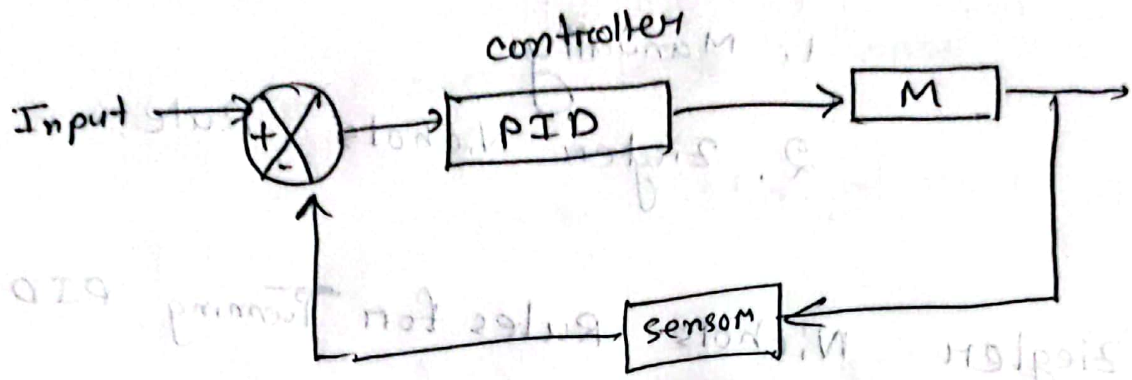
Advantage of PID

1. Simple : easy to understand
2. Effective : Accurate and stable
3. Robust : can be adopted to different Robot system

Application:

1. Arm positioning
2. Robot Navigation
3. Speed control
4. Balance Control.

A) Practise Problem:



Given, required value = 30 rad/s

$K_p = 0.5$

$k_i = 0.6$

$k_d = 0.2$

Simulate the value of output! (3-4 steps)

Solution:

Step-01 $\quad Error_1 = (30 - 0)$

$= 30$

$$Output = K_p * E_1 + K_i * \Sigma E + K_d * \Delta E$$

$$= K_p * E_1 + K_i * E_1 + K_d * (E_1 - E_0)$$

$$= 0.5 * 30 + 0.6 * 30 + 0.2 * (30 - 0)$$

$$= 15 + 18 + 6 = 39$$

$=$

step-02

$$Error = 30 - 39$$
$$E_2 = -9$$

$$Output = K_p * E_2 + K_i * (E_1 + E_2) + K_d * (E_2 - E_1)$$
$$= 0.5 * (-9) + 0.6 * (30-9) + 0.2(-9-30)$$
$$= -4.5 + 12.6 - 7.8$$
$$= 0.3$$

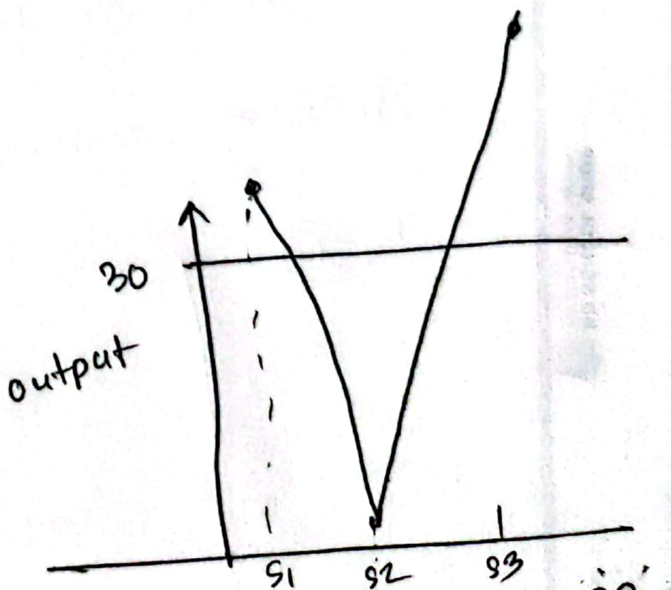
step-03

$$Error = 30 - 0.3$$
$$= 29.7$$

$$output = 0.5 * 29.7 + 0.6 (E_1 + E_2 + E_3) + K_d * (E_3 - E$$
$$= 53.01$$

⋮

And so on.



output response.

Prachurja Bhattacharjee