CONTROL

SARAH ALKASASBEH

26/2

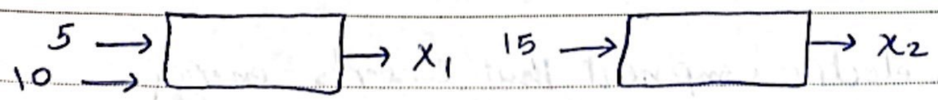
* Real time systems :- systems are subjected to real-time, ie., the response should be guaranteed within a specified timing constraint or the system should meet the specified deadline

- characteristics of real-time systems :-
  1) Time constraints → the task should be completed within time interval (meet the specified deadline) / (fast response)
  2) A task in the real-time systems <u>must not</u> start before its specified time.

- Notes:- • Not every embedded system is a real time system
  ↳ • Not every embedded system is control system
  ↳ • real-times systems (some of them) may not need controllers.
  ↳ • The response in controllers is slow compared to the response in embedded systems.

28/2

- linear time-invariant systems (LTI) → both linear and time invariant where linear systems are systems whose outputs for a linear combination of inputs are the same as a linear combination of individual responses to those inputs. for example :-



if $x_1 = x_2$ → linear system

- Time invarint system is one whose response to inputs doesn't change with time.

• تركيزنا بال control systems على (LTI)، بس المشكلة بالطبيعة جميع الخصائص الفيزيائية
تكون non-linear ← تم حل المشكلة باستخدام (approximation)
علاقة تقريب non-linear model ال linear model

• Analog Control system → the Controller is made up of resistors, and capacitors (complex), (hard to modify), (and they are subject to noise and distortion)
↳ highly sensitive

• Digital Control system → uses digital computers, microcontrollers, microprocessors ...etc.

• ex:- Basic analog control system :- potentiometers (variable ~~resistance~~ resistor)

• In digital control systems, computer functions may be passive or supervising, or ~~they~~ they may be active controllers (PLC "programmable logic controllers)
• SCADA systems → used for controlling, monitoring, and analyzing devices and processes (both active and passive)

• Transducer is an electric component that converts energy from one form to another.

$$\text{Transducer} \begin{cases} \text{sensors} \\ \text{actuators} \end{cases}$$

Scanned with CamScanner

- sensors → convert temperature to voltage or current (example)
- actuators → convert voltage or current to another physical form.

Note :- digital control systems are less susceptible to noise as well as there is no tolerance effect.

- sampling frequency = 2 * frequency of the signal
  ↳ when converting from analog to digital

↑ frequency ↑ power consmption

- two types of control systems :- 1) open loop 2) closed loop
- Open loop control system → there is no feedback path Provided (non-feedback control system)

تؤدي الوظيفة باسبط التكاليف والتصاميم

↳ there is no sensors
↳ we don't know anything about the quality or the state of the control
↳ it only gives ~~a~~ commands
↳ one forward path (feed forward path) from input to output
↳ unable to deal with any external changes (refer to the slides)

- disturbances :-
  1) noise → affects the control signal
  2) disturbances at the output

~~xxxx~~

- Closed loop control systems → there is a feedback path in addition to feed forward path.
  ↳ 1) Negative → works on (target − sensor output)
  ↳ feedback path
- Closed loop control systems → has the ability to self-correct while the open-loop system doesn't.
  two types 1) negative → (target − feedback path)
  ↓ error signal
  - the target and feedback ~~xxxx~~ must be the same unit.

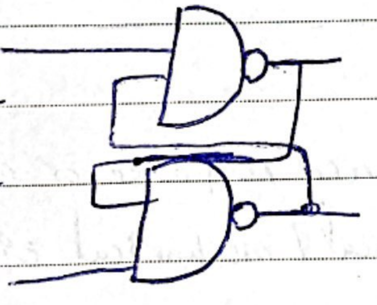
2) Positive → (target + feedback path)
↳ not preferred
↳ actuating signal here is not the error signal

notes:-

- Open loop control systems have fast response because there is no measurement and feedback of output. The response of the closed loop control systems is slow due to presense of feedback.
- closed loop control systems require more maintenance (more expensive).
- closed loop control systems are more accurate
- closed loop control sysems are less stable (unstability)

- positive closed loop control systems example → latches

- There is a feedback path and the latch value remains the same.



- There may be multible feedback paths from different sources or from the same source. Some of them may be used in the control process or not.

- Why do we rarely see HDDs with speeds more than 7200 RPMs?
  - because when the speed increases, the force on the disc increases, which may damage it. it also brings negative effects such as temperature increase.

- " " " " " " more than <u>5400</u> in laptops?
  - because it's portable. when the speed increases it'll be more sensitive to any external effects while it's working

- steady-state error :- the difference between the input command and the output of a system

- in HDDs, the steady-state error must be almost zero percent.

- natural response :- unforced response or characteristic response
  - ↳ has physical / mechanical equations that control it.
  - ↳ with time, the effect of the system disappears, and the output only affected by the input. ⟶ stable system

- forced response ⟶ the system is stable if the input is within its range

---

Digitization operations ⟶ done by ADC
- ↳ sampling
- ↳ quantization

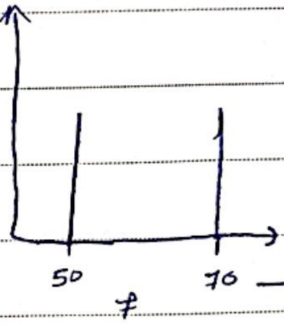
- Sampling → discretization of time
- Quantization → discretization of range (ex:- if temperature is between 0 and 1 consider it 1).
- if we use the sampling frequency less than twice the maximum frequency component in the signal, that is called <u>undersampling</u>

$$\boxed{f_{sampling} = 2 \times f_{max}}$$

- undersampling results in data loss.

- Fourier transform is a mathematical model which helps to transform the signals between different domains (from time domain to frequency domain or vice versa.)

ex :- $\sin(50t) + \sin(70t)$



→ max frequency component.

$f_{sampling} = 2 * 70 = 140 Hz$ ← or sampling rate

- $T_{processing} < T_{sampling}$

- Original signal max frequency $<$ $\frac{f_{sampling}}{2}$
  ↳ ممكن تكون نص fsampling او اقل

- distance between samples = $1/f_{sampling}$ = T_s
  (time)

- Cycle time = 1 / Processor frequency

$T_{processing}$ = cycle time x # of ~~cycles~~ instructions x # of cycles

$T_{processing} < T_{sampling}$
↳ $1/f_{sampling}$

- if $T_{processing} > T_{sampling}$ we can use faster processor but this solution is considered expensive, or we can decrease the size of our program / using Algorithms (with better performance ex:- O(logn)) / Compiler optimization
  ↳ اسهل من ال Algorithm

Compiler optimization
- speed → if memory is large enough
- size → ~~if~~ if memory size is small

Deadlines
- relative → نحسب ال deadline من لحظة وصول ال event
- absolute → غير مرتبط بال event وبس مرتبط بالوقت سواء في event ال 8 او 2 بعمل deadline

ال event ما is نفذ ال deadline في ال ~~...~~ ص

• Functional Requirements → "essential"

A. Data collection / Acquisition → through feedback loops (sensors)

↳ if Analog → use Analog to digital converter.

B. Signal Conditioning → amplification / filtering / Converting and any other processes required to make sensor output suitable for processing.

C. Plant / Process Control → Controlling.

D. Alarm monitoring / Data logging / Man Machine interaction → scada systems

↳ optional.

• Analog signals are more sensitive to noise (gaussian noise)

• EEC → error detection and correction.

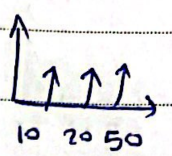
• low pass filter → is a filter that passes signals with a frequency lower than a selected cutoff frequency and attenuates signals with frequencies higher. (RC circuits)

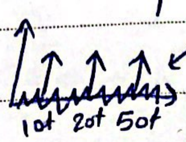
↳ . Losses

• $\sin(10t) + \sin(20t) + \sin(50t)$

→ we can't determine the max. freq. so we have to get rid of this noise before sampling using low pass filters.

Fourier transform →





actual signal due to noise

- High pass filter → filter designed to pass all frequencies above its cut-off frequency, and attenuates signals with frequencies lower.

- Band pass filter → passes frequencies within a certain range.

- Non-functional Requirements:

1. Dependability → the quality of being able to be trusted and being very likely to do what people expect.

Reliability　　Safety

fault-tolerance　　Security

"survivability"

- Reliability → the probability that a product or system will perform its intended function adequately for a specified period of time.

- Note: Slide 44 → survival function 4 is the best one / survival function 1 more realistic in the industry (years insted of months)

- λ → constant failure rate $< 10^{-7}$ (high reliability)

$$\lambda = \frac{\#\text{of failures}}{\#\text{of systems} \times \text{operating time}}$$

- fault tolerance → the ability of a system to continue operating without interruption when one or more of its components fail
  ↳ ex:- odd number of sensors / lockstep processor

Note: lockstep processor → fault-tolerant that run the same set of operations at the same time in parallel, the output from lockstep operations can be compared to determine if there has been a fault if there are at least two systems, and the error can be corrected if there are at least three systems.

- Safety stands for accident avoidance (unintentional harm), and security for crime prevention (intentional harm).

Scanned with CamScanner

- Fail-safe system → in the event of any fault appearing, the system will always go to a safe status. (another action to do to reach the safe state.)
- Fail-operational → guarantee the full or degraded operation of a function even if a failure occurs. (يستمر بعمل الوظيفة الأساسية)

- SCADA system (hybrid system) "Supervisory Control and Data Acquisition system) → used for Controlling, monitoring, and analyzing industrial devices and Processes. The system consists of both software and hardware components.

- SCADA System Architecture layers :-

1. Level 0 (Field level) → the lowest level and includes the Physical devices that gather data from the processes. (sensors / actuators)

2. Level 1 (Programmable Devices for direct Control)
   ↳ PLCs → local
   ↳ RTUs → Remote

3. Level 2 (Local plant control and HMI) → observation and local control
   ↳ "Human Machine interface"
   ↳ sending commands to PLCs and RTUs
4. level 3 (Coordination and Production control)
5. level 4 (Central control and Production Scheduling) → more software

---

"end of chapter 1".

"Engineering is the Art of approximation" :) "Chapter 2" 26/3

. System modeling is the process of developing abstract models of a system. (تصميم يعبر عن الشكل الغائي)

. Laplace : A mathematical tool that converts any differential and integral equations into algebraic equations. (from time domain to frequency s domain).

. The superposition property of linear systems states that the response of a linear system to a sum of signals is the sum of the responses to each individual input signal.

• linear systems must apply the superposition principle.

• A homogeneous system of linear equations is one in which all of the constant terms are zero. (no constant term)

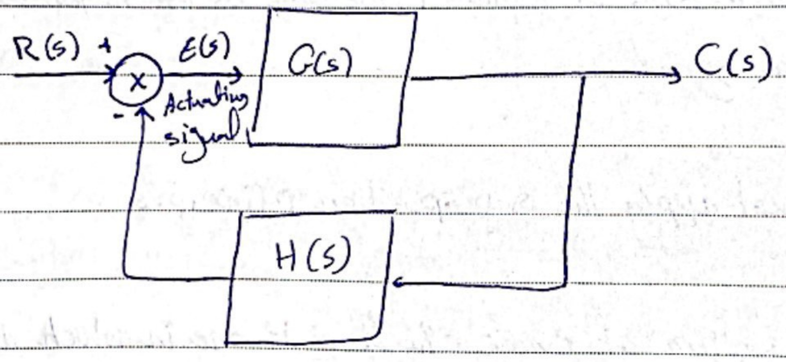
Laplace equation $\rightarrow$ $\mathcal{L}_t[f(t)](s) \equiv \int_0^{\infty} f(t)e^{-st}\,dt$

$s = \sigma - j\omega t$ if $\sigma = 0 \rightarrow$ fourier transform

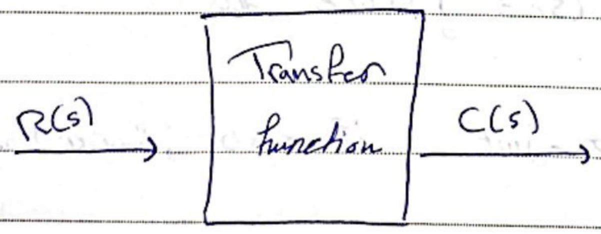
- Transfer function → represents the relationship between the output signal of a control system and the input signal.

- Actuating signal → An algebraic ~~[illegible]~~ addition or subtraction of input signal and feedback signal.
  - addition → Positive feedback system
  - subtraction → negative feedback system

- $R(s)$ → reference input / $C(s)$ → controlled output



we need to simplify it to :-



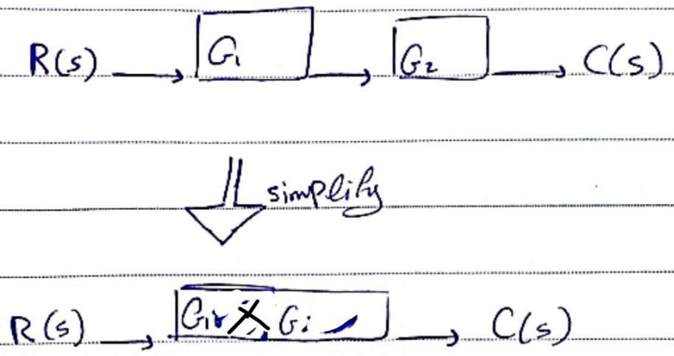
if we have n-systems we have n $\frac{C}{R}$

$$\frac{C_1(s)}{R_1(s)} + \frac{C_2(s)}{R_2(s)} + \dots \frac{C_n(s)}{R_n(s)}$$

- negative feedback system → $\dfrac{C(s)}{R(s)} = \dfrac{G(s)}{1+G(s)H(s)}$
- Positive feedback system → $\dfrac{C(s)}{R(s)} = \dfrac{G(s)}{1-G(s)H(s)}$

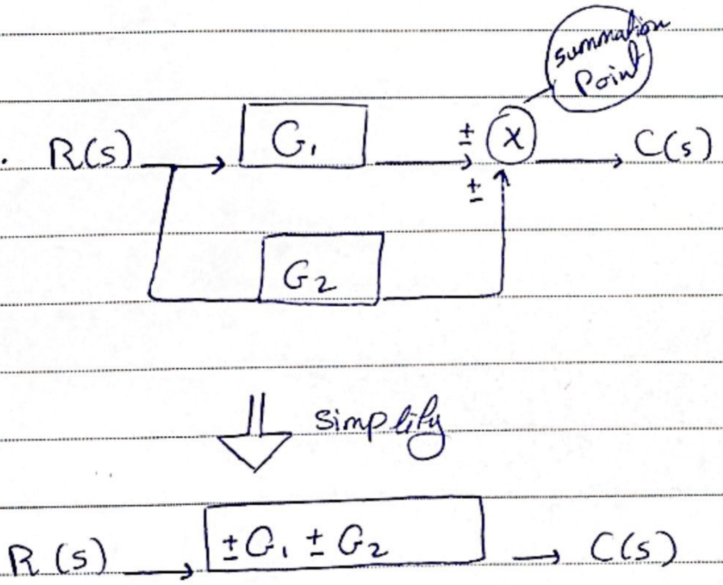
→ Transfer functions of positive and negative feedback systems

• Block Manipulation



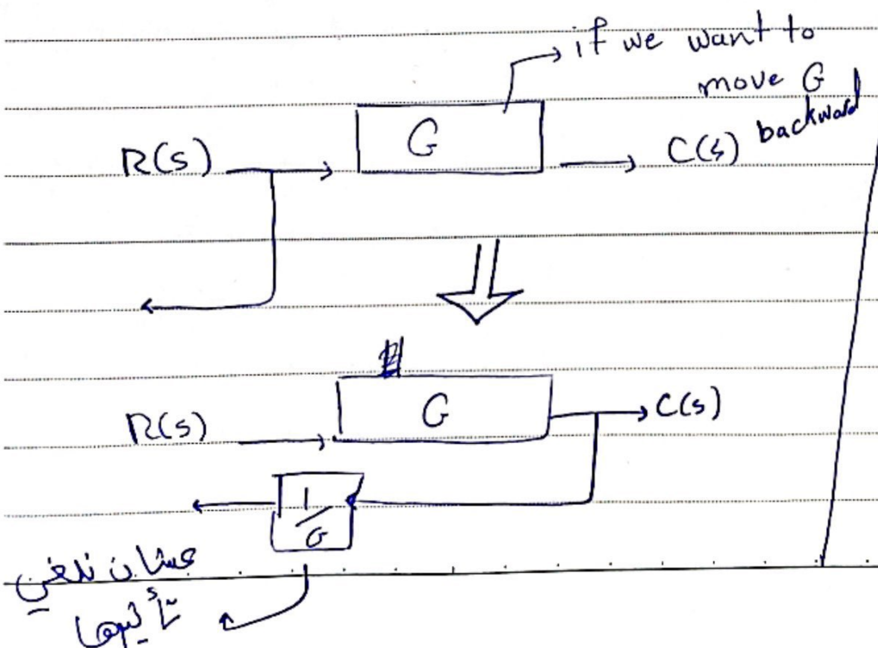
• Cascade systems "series"

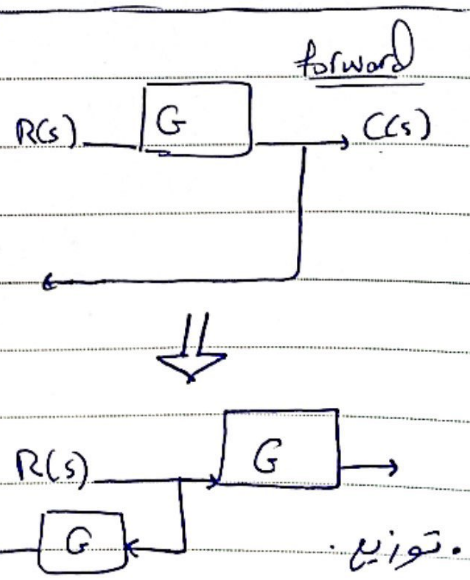
• multiplication



• Parallel system

• addition or subtraction according to ~~the~~ the sign enters the summation point

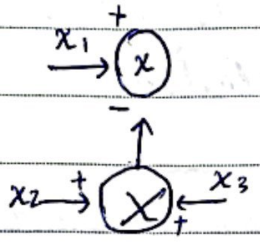




• توزيع

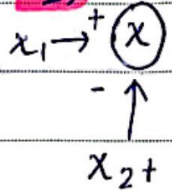
- Moving a Summing point
- مش حفظ.

Rule 1 :- Rearrangement of summing points



1) $x_2 + x_3$

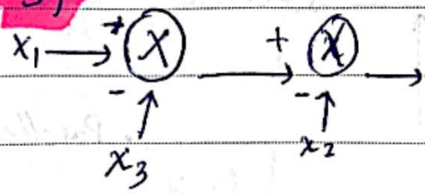
$\Rightarrow$ 2)



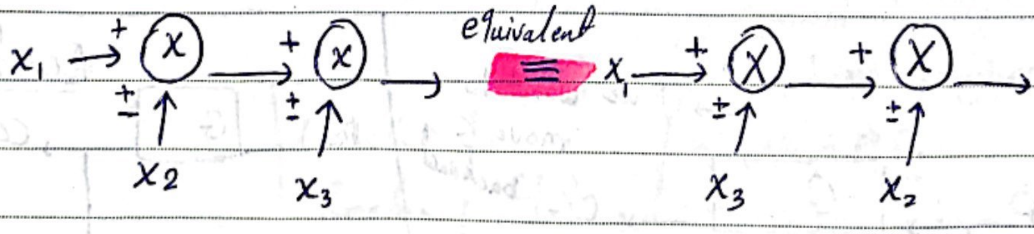
$\Rightarrow x_1 - (x_2 + x_3)$

$= x_1 - x_2 - x_3$

3)

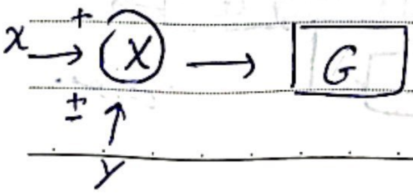


Rule 2 :- Interchang of Summing Points → الترتيب ما بفرق

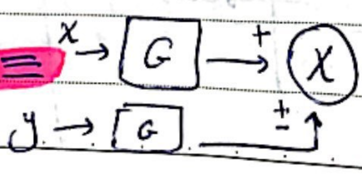


Rule 3 :- Moving a summing point ahead of a block

بالأصل عبارة عن $GX \pm GY$. (Summation point ال block قبل ننقل)



$\equiv$

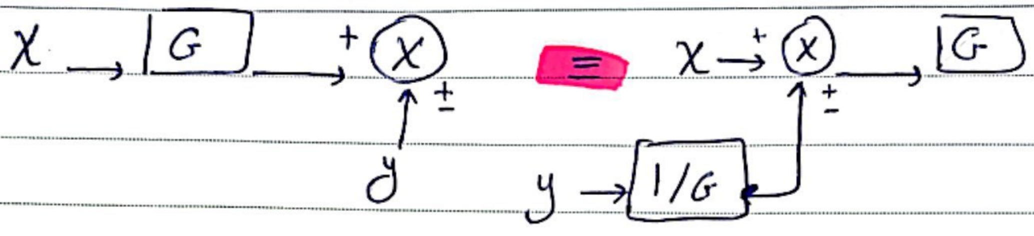


To keep x, y multiplied by G.

Rule 4: Moving a summing point beyond a block.

(نقل ال block بعد ال Summation Point.)

فقط x مضروبة ب G.



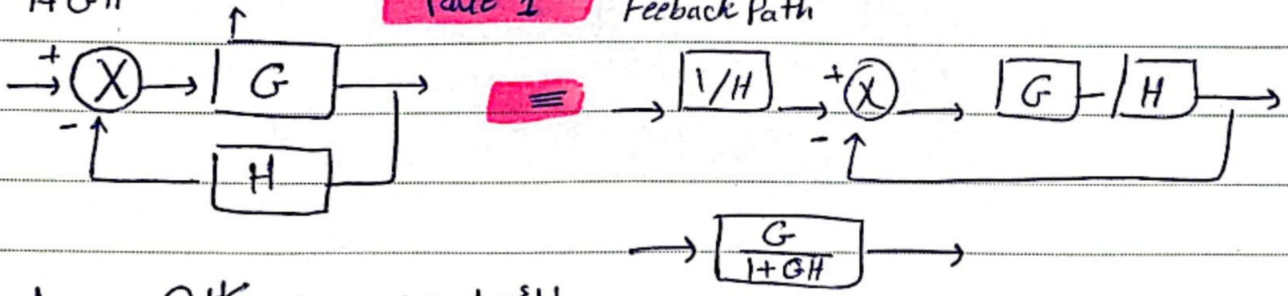
to keep only X multiplied with G.

Unity path → a path that doesn't contain any block, whether it's forward path or feedback path. = "block with 1"

Changing feedback and forward path (Converting to a unity path).

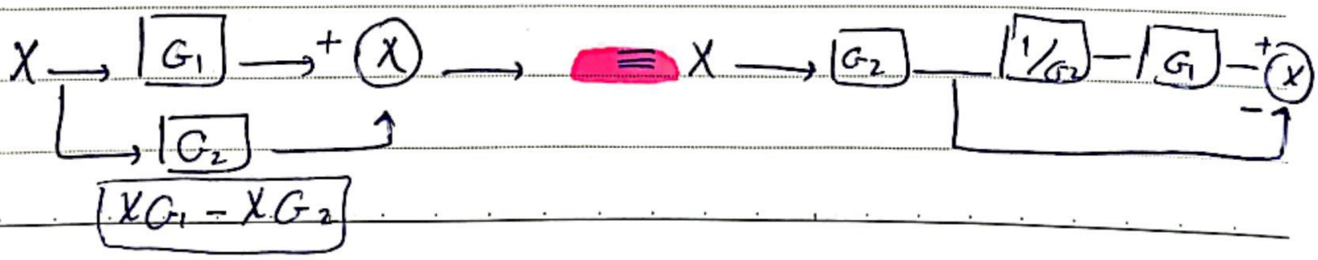
$\frac{G}{1+GH}$ ← تعطيني

Rule 1 Feedback Path



$\frac{1}{\cancel{H}} \cdot \frac{G\cancel{H}}{1+GH}$ ← بالأصل كانت

Rule 2 Forward Path



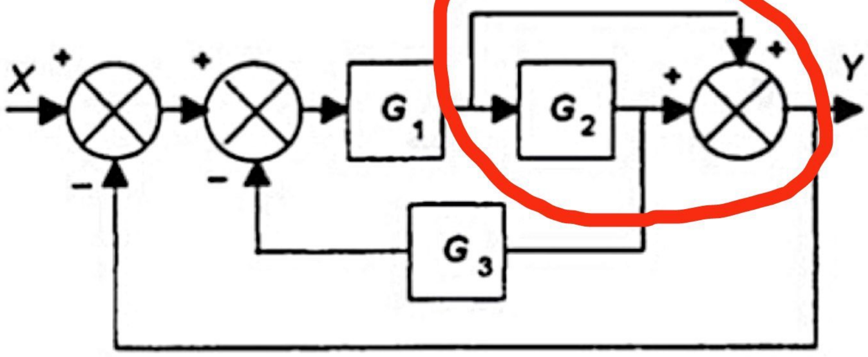
Scanned with CamScanner

Example 1 on Control System Block Simplification

9

نقطة البداية
لازم نتخلص من نقاط التشعب او
التفرعات (G1 و G2 يكونوا series)

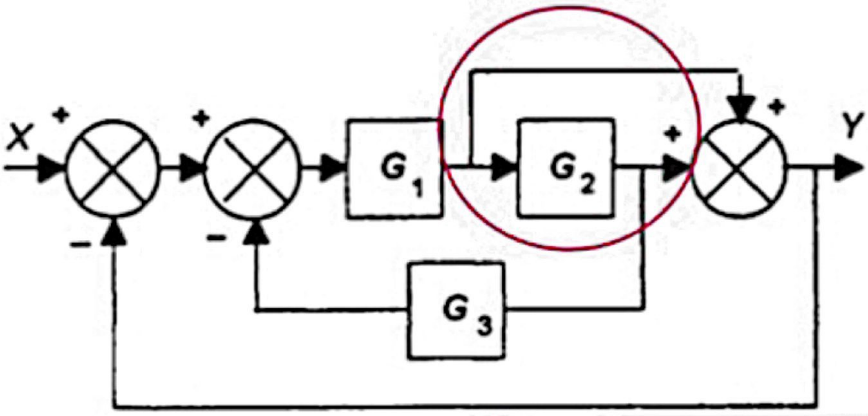
Simplify the following control circuit using the block manipulation rules:

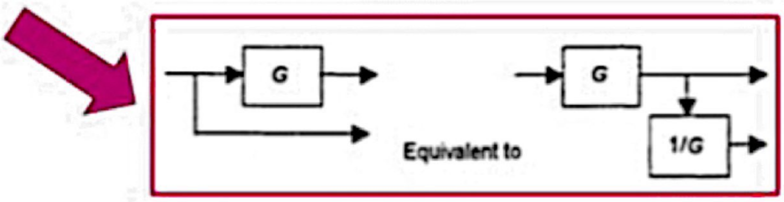


Example 1 on Control System Block Simplification

10

Simplify the following control circuit using the block manipulation rules:

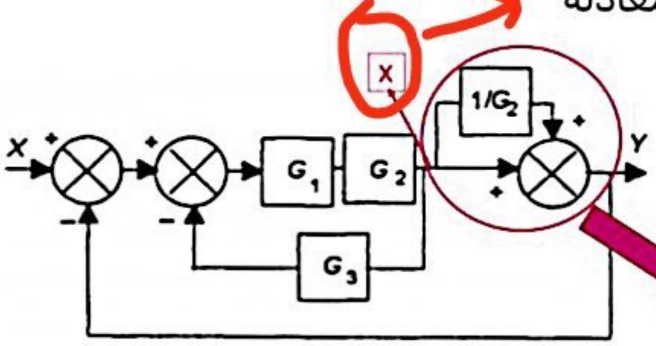




Example 1 on Control System Block Simplification

Simplify the following control circuit using the block manipulation rules:

افترضناه لنحل المعادلة



حل المعادلة رياضيا

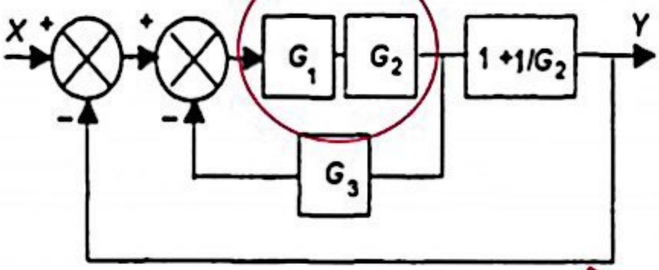
$$Y = X + \frac{X}{G_2}$$

$$Y = X\left(1 + \frac{1}{G_2}\right)$$

$$\frac{Y}{X} = 1 + \frac{1}{G_2}$$

Example 1 on Control System Block Simplification

Simplify the following control circuit using the block manipulation rules:

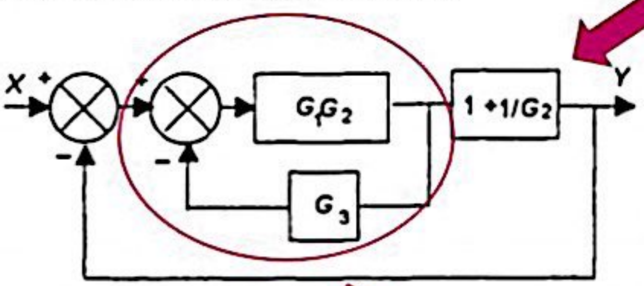


$G_{overall} = G_1G_2$

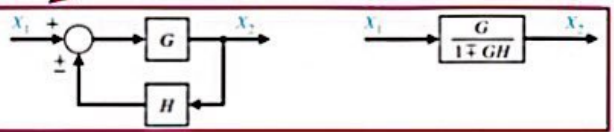
Example 1 on Control System Block Simplification

Simplify the following control circuit using the block manipulation rules:

$G_{overall}(s) = G_1G_2$



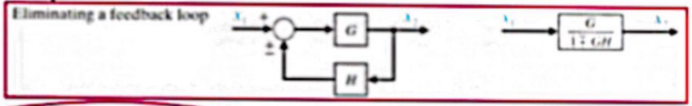
Eliminating a feedback loop

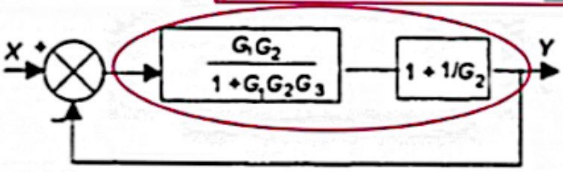


Example 1 on Control System Block Simplification

14

Simplify the following control circuit using the block manipulation rules:





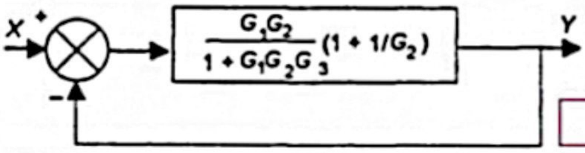
$G_{overall} = G_xG_y$

Example 1 on Control System Block Simplification

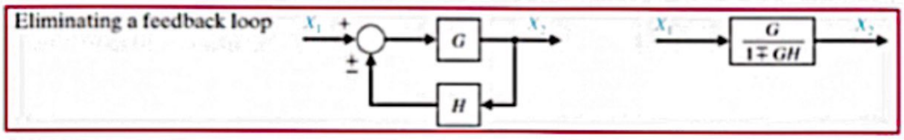
15

Simplify the following control circuit using the block manipulation rules:

$G_{overall} = G_xG_y$



H = 1 for empty path

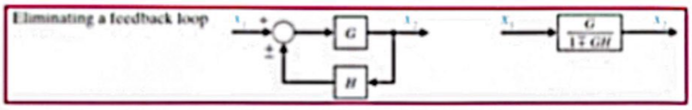


Example 1 on Control Block Simplification

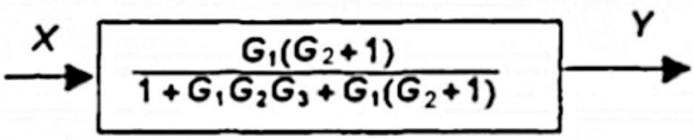
16

Simplify the following control circuit using the block manipulation rules:

G1,G2,G3
عبارة عن functions كانوا بالاساس
بال time domain



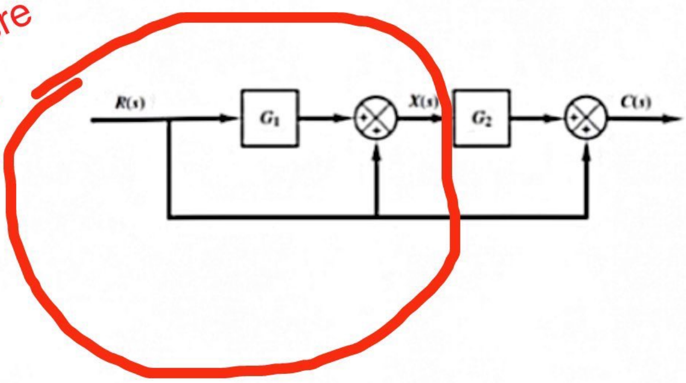
after simplification



Example 2 on Control System Block Simplification

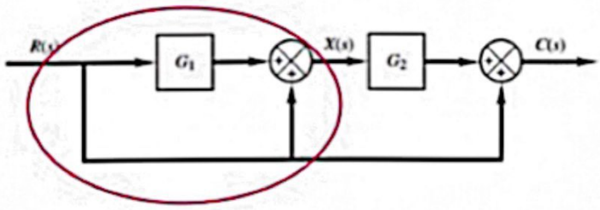
Simplify the following control circuit using the block manipulation rules:

Start from here



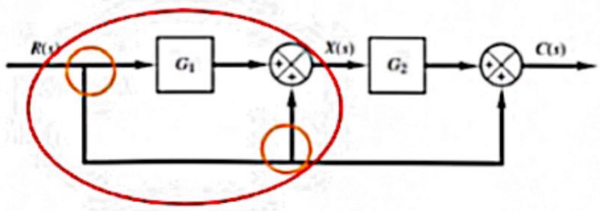
Example 2 on Control Block Simplification

Simplify the following control circuit using the block manipulation rules:



Example 2 on Control Block Simplification

Simplify the following control circuit using the block manipulation rules:

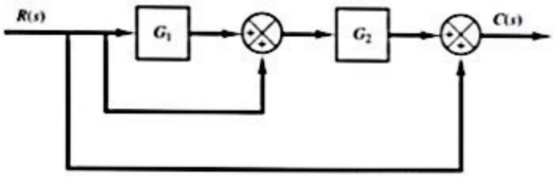


رتبهم بطريقة افضل

Example 2 on Control Block Simplification

20

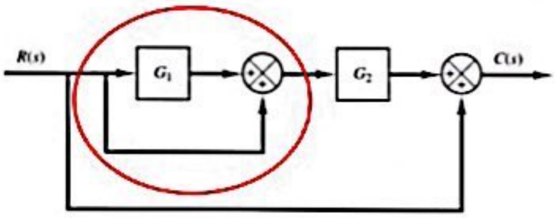
Simplify the following control circuit using the block manipulation rules:



Example 2 on Control Block Simplification

21

Simplify the following control circuit using the block manipulation rules:

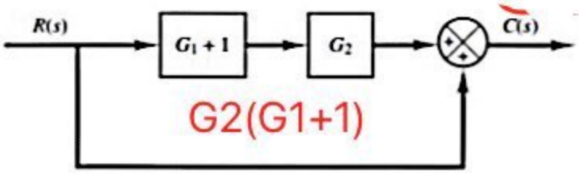


$$y = XG1 + X$$
$$y = X(G1 + 1)$$
$$y/X = G1 + 1$$

Example 2 on Control Block Simplification

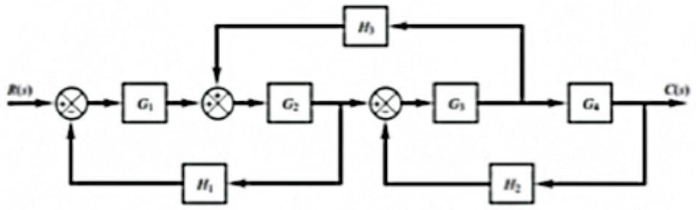
22

Simplify the following control circuit using the block manipulation rules:



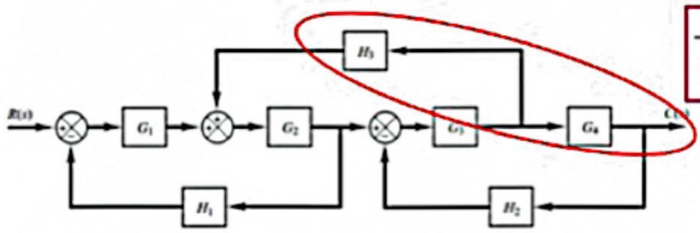
Example 3 on Control Block Simplification

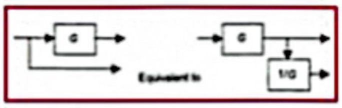
Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



Example 3 on Control Block Simplification

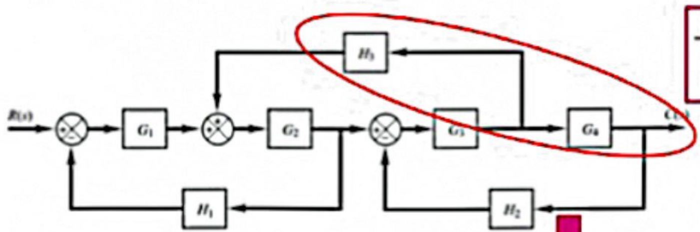
Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).

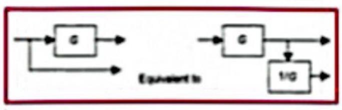




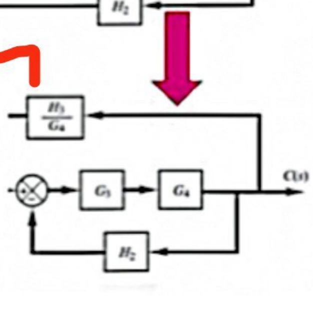
Example 3 on Control Block Simplification

Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).





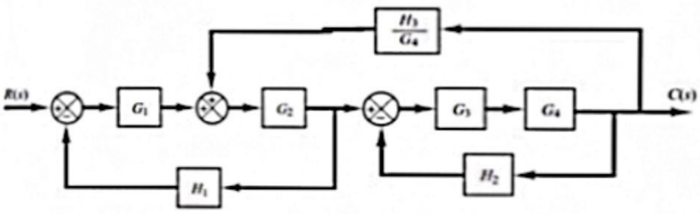
لانهم كانوا series وهاد ال block
بعد ما ضربهم



Example 3 on Control Block Simplification

27

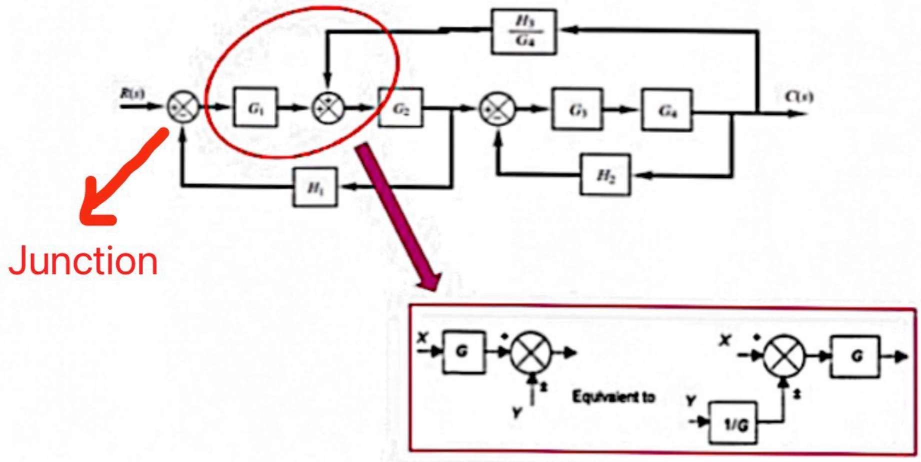
Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



Example 3 on Control Block Simplification

28

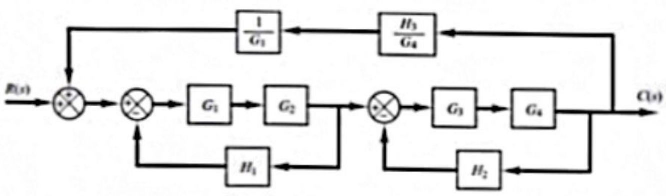
Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



Example 3 on Control Block Simplification

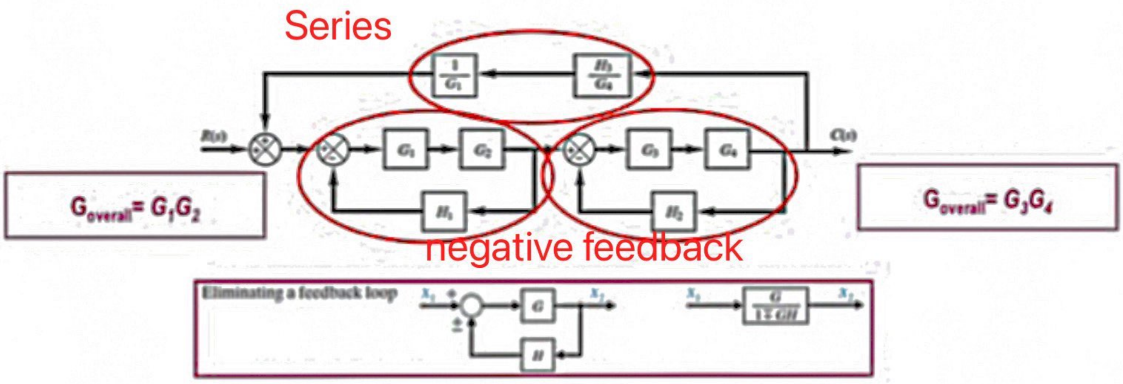
29

Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



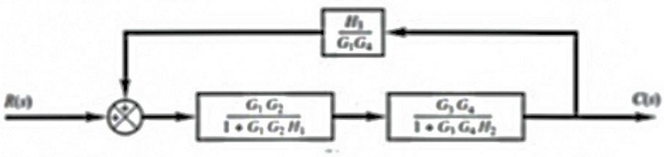
Example 3 on Control Block Simplification

Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



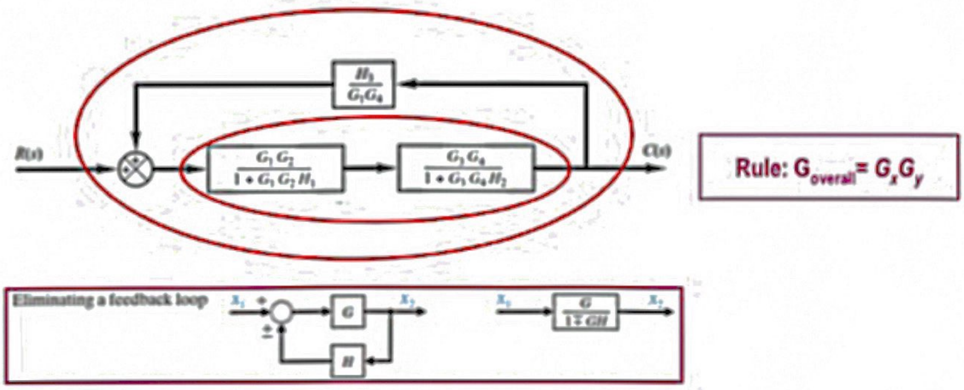
Example 3 on Control Block Simplification

Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



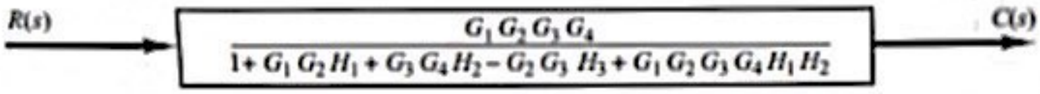
Example 3 on Control Block Simplification

Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



Example 3 on Control Block Simplification

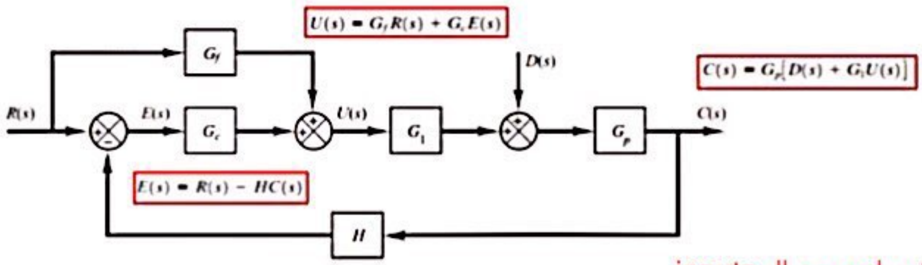
Simplify the block diagram shown in the Figure. Then obtain the closed-loop transfer function C(s)/R(s).



Example 4 on Control Block Simplification

Using Equations

Mathematically obtain transfer functions C(s)/R(s) and C(s)/D(s) of the system shown in the Figure:



-This is a MIMO system
-D(s) is disturbance
-We have 2 transfer functions

بنصفر واحد من ال inputs
عشان نحصل على C/R بنصفر D بالنهاية
وعشان نحصل على C/D بنصفر R بالنهاية
باستخدام superposition كونه linear

Example 4 on Control Block Simplification

Using Equations

Mathematically obtain transfer functions C(s)/R(s) and C(s)/D(s) of the system shown in the Figure:

Substitute U_s into C_s

$$C(s) = G_p D(s) + G_1 G_p[G_f R(s) + G_c E(s)]$$

Substitute E_s into the above equation:

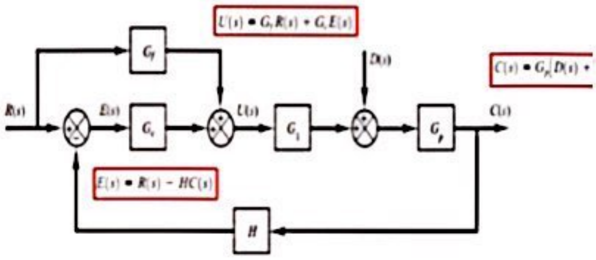
$$C(s) = G_p D(s) + G_1 G_p\{G_f R(s) + G_c[R(s) - HC(s)]\}$$

Solve for C_s:

$$C(s) + G_1 G_p G_c HC(s) = G_p D(s) + G_1 G_p (G_f + G_c) R(s)$$

Rearrange:

$$C(s) = \frac{G_p D(s) + G_1 G_p (G_f + G_c) R(s)}{1 + G_1 G_p G_c H}$$



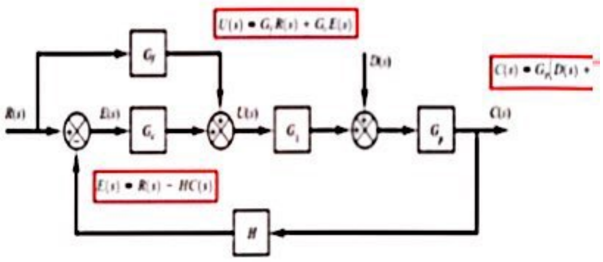
دايما بنحاول نخلي ال outputs
على جهة وال inputs على جهة

Example 4 on Control Block Simplification

Using Equations

Mathematically obtain the transfer functions C(s)/R(s) and C(s)/D(s) of the system shown in the Figure:

$$C(s) = \frac{G_p D(s) + G_1 G_p (G_f + G_c) R(s)}{1 + G_1 G_p G_c H}$$



Notice that this is a multiple input system, one for the actual Input R(s), and another for some disturbance D(s)

To obtain the transfer functions C(s)/R(s) and C(s)/D(s), we must only let one input present, and all other inputs are 0, therefore:

C(s)/R(s) = $\frac{C(s)}{R(s)} = \frac{G_1 G_p (G_f + G_c)}{1 + G_1 G_p G_c H}$

C(s)/D(s) = $\frac{C(s)}{D(s)} = \frac{G_p}{1 + G_1 G_p G_c H}$

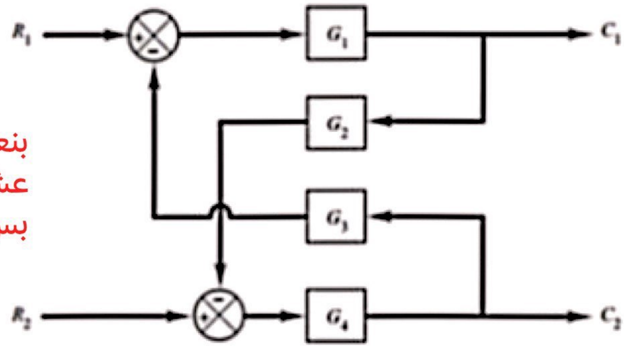
Example 5 on Control Block Simplification MIMO system

Using Equations

Figure 2–24 shows a system with two inputs and two outputs. Derive $C_1(s)/R_1(s)$, $C_1(s)/R_2(s)$, $C_2(s)/R_1(s)$, and $C_2(s)/R_2(s)$

Solution:

-We have 4 transfer functions because there are 2 inputs and 2 outputs



$$C_1 = G_1(R_1 - G_3C_2)$$

$$C_2 = G_4(R_2 - G_2C_1)$$

بنعوض C1 بمعادلة C2 او العكس
عشان بهمنا يكون في output واحد
بس بالمعادلة

Substitute C_2 into C_1

$$C_1 = G_1[R_1 - G_3G_4(R_2 - G_2C_1)]$$

Substitute C_1 into C_2

$$C_2 = G_4[R_2 - G_2G_1(R_1 - G_3C_2)]$$

Solve the equation for C_1

$$C_1 = \frac{G_1R_1 - G_1G_3G_4R_2}{1 - G_1G_2G_3G_4}$$

Solve the equation for C_2

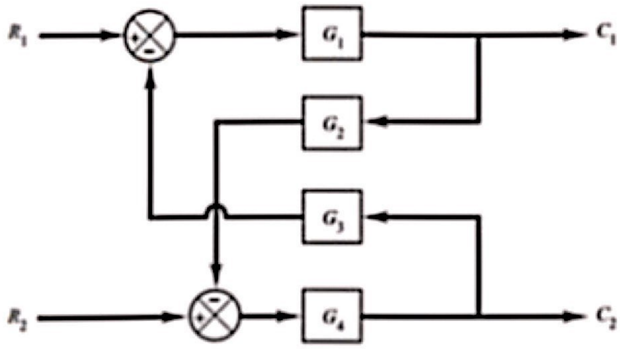
$$C_2 = \frac{-G_1G_2G_4R_1 + G_4R_2}{1 - G_1G_2G_3G_4}$$

Example 5 on Control Block Simplification

Using Equations

Figure 2–24 shows a system with two inputs and two outputs. Derive $C_1(s)/R_1(s)$, $C_1(s)/R_2(s)$, $C_2(s)/R_1(s)$, and $C_2(s)/R_2(s)$

We now have these two relationships:



$$C_1 = \frac{G_1R_1 - G_1G_3G_4R_2}{1 - G_1G_2G_3G_4} \qquad C_2 = \frac{-G_1G_2G_4R_1 + G_4R_2}{1 - G_1G_2G_3G_4}$$

We can derive the transfer functions of $C_1(s)/R_1(s)$ and $C_2(s)/R_1(s)$ by setting R_2 to 0, so

$$\frac{C_1(s)}{R_1(s)} = \frac{G_1}{1 - G_1G_2G_3G_4},$$

$$\frac{C_2(s)}{R_1(s)} = -\frac{G_1G_2G_4}{1 - G_1G_2G_3G_4}.$$

Similarly, we can derive the transfer functions of $C_1(s)/R_2(s)$ and $C_2(s)/R_2(s)$ by setting R_1 to 0, so

$$\frac{C_1(s)}{R_2(s)} = -\frac{G_1G_3G_4}{1 - G_1G_2G_3G_4}$$

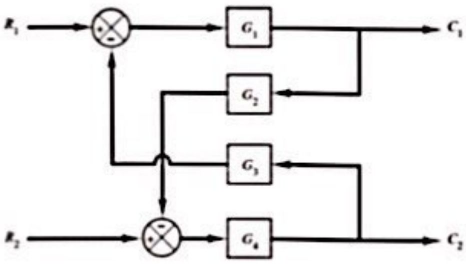
$$\frac{C_2(s)}{R_2(s)} = \frac{G_4}{1 - G_1G_2G_3G_4}$$

Example 5 on Control Block Simplification طريقة ثانية للحل

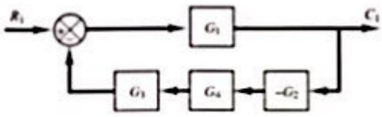
Using Equations – Alternative Solution

When setting $R_2 = 0$, the block diagram can actually be simplified to:

مرة صفرنا R2 وطلعنا معادلة R1
C1/R1, C2/R1
ومرة صفرنا R1 وطلعنا معادلة
R2

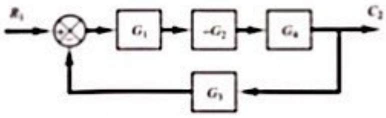


(a)



$$\frac{C_1}{R_1} = \frac{G_1}{1 - G_1 G_2 G_3 G_4}$$

(b)

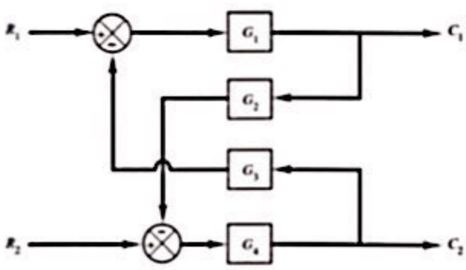


$$\frac{C_2}{R_1} = \frac{-G_1 G_2 G_4}{1 - G_1 G_2 G_3 G_4}$$

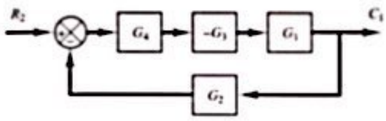
Example 5 on Control Block Simplification

Using Equations – Alternative Solution

When setting $R_1 = 0$, the block diagram actually can be simplified to:

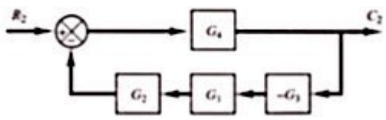


(c)



$$\frac{C_1}{R_2} = \frac{-G_1 G_3 G_4}{1 - G_1 G_2 G_3 G_4}$$

(d)

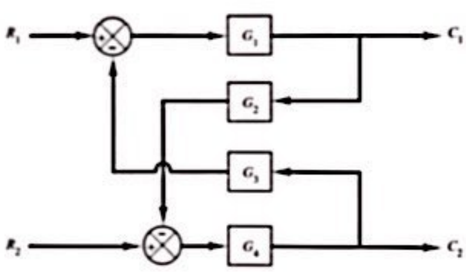


$$\frac{C_2}{R_2} = \frac{G_4}{1 - G_1 G_2 G_3 G_4}$$

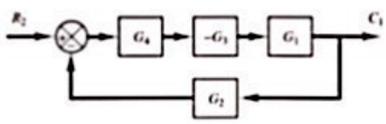
Example 5 on Control Block Simplification

Using Equations – Alternative Solution

When setting $R_1 = 0$, the block diagram actually can be simplified to:

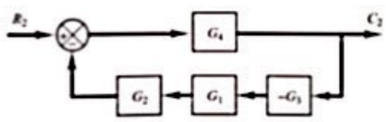


(c)



$$\frac{C_1}{R_2} = \frac{-G_1 G_3 G_4}{1 - G_1 G_2 G_3 G_4}$$

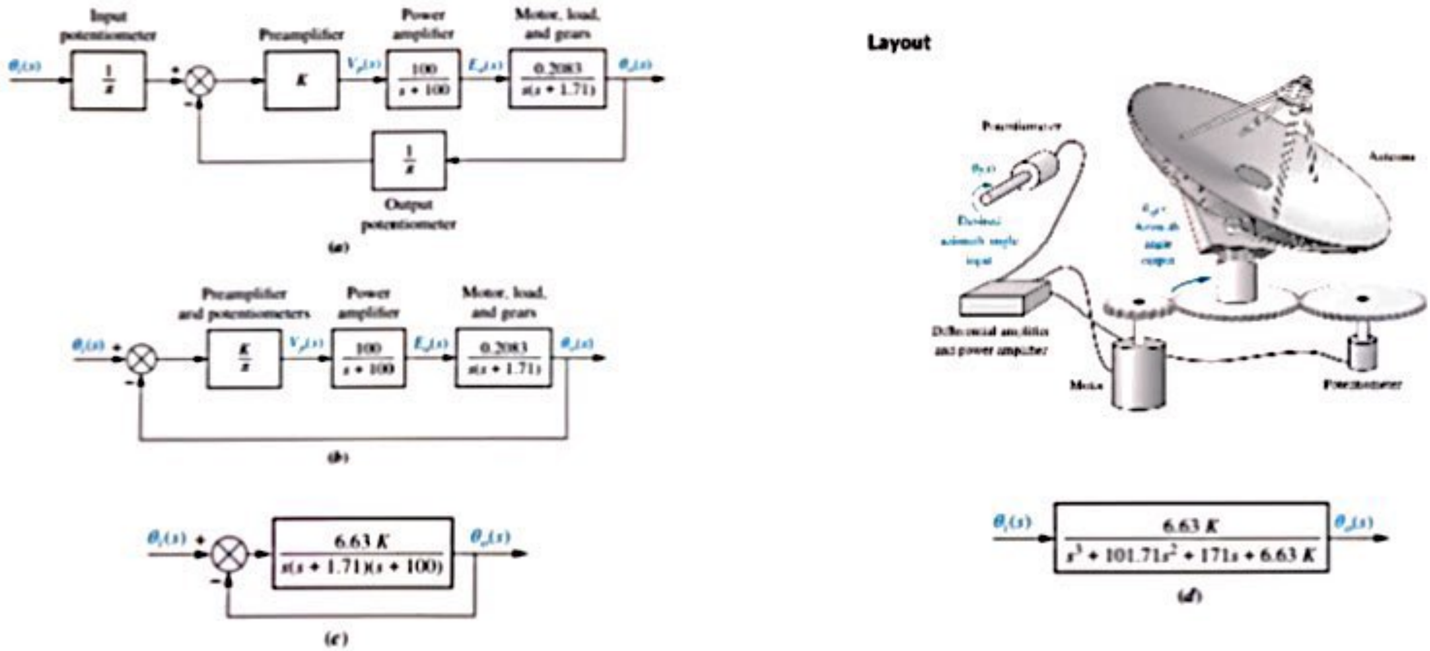
(d)



$$\frac{C_2}{R_2} = \frac{G_4}{1 - G_1 G_2 G_3 G_4}$$

Example 6 on Control Block Simplification

Antenna Azimuth Control System



References and Textbook Material

The material in these slides are based on:

Control Systems Engineering, Norman S. Nise, 7th Edition (2014), John Wiley And Sons

- **Chapter 5 – Reduction of Multiple Subsystems**
  Sections *5.1, 5.2*

Instrumentation and Control Systems, 1st Edition , 2004, Elsevier (Newer versions available 3rd edition 2021, but I used the first one)

- **Chapter 9 – Transfer Function**
  Sections *9.1, 9.2, 9.3, 9.4, 9.5*

"chapter 3"

Time Response

Time response → The amount of time required for the outputs to respond or react to the inputs.

- Forced response is what the system does ~~with the zero~~ when turned on, but with the initial condition set to zero (from state to state "steady state") also called steady-state response. "input"

- Natural response is how does the system transfer from a state to the steady-state but with the influence of the system's properties, and how its properties affect the transition process. This effect must decrease over time until it disappears so that we can reach the steady-state. "شكل الانتقال"

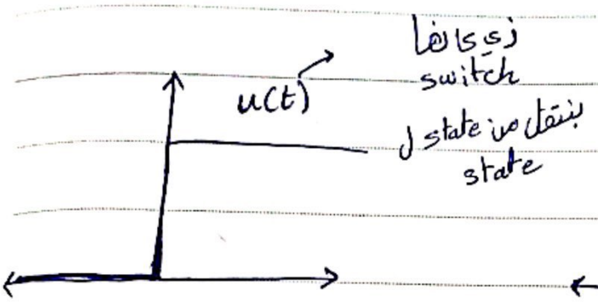
- first order system → المقام من الدرجة الأولى مثلاً ($\frac{1}{s}$) "أكبر درجة في المقام هي 1"

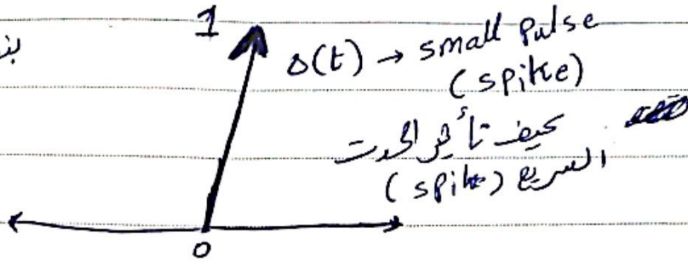
- second order system → أكبر درجة في المقام هي 2 مثلاً ($\frac{1}{s^2}$)

- Poles → أصفار المقام / zeros → أصفار البسط
  - ↳ they will determine if the system is stable or not, so we'll focus on them

in s-plane (representation of blocks) → Poles are represented by an (X), and zeros are represented by (O) . هيك بنقدر نحول ال transfer function من رسمة s-plane

• يجب ان تكون جميع ال Poles على left half من ال s-plane ليكون stable.





$u(t) \longrightarrow \frac{1}{s}$, so when the input of a system is $\frac{1}{s}$, consider it a switch.

(↳ time domain) (↳ s-domain)

ex:-
• input $\frac{1}{s}$ → forced input ~~response~~
• Natural response → شكل الانتقال | is it liner / smooth ... etc
↳ determined by the tranfer function of a system. ↳ لازم تأثيره بالغاية يختفي

$$R(s) = \frac{1}{s} \longrightarrow \boxed{\frac{s+2}{s+5}} \xrightarrow{C(s)}$$

G(s) → Poles يعطوني شكل الانتقال

• شكل الانتقال من لحظة التشغيل ولحظة الثبات, transient response

$$C(s) = \frac{(s+2)}{s(s+5)} = \frac{2/5}{s} + \frac{3/5}{s+5}$$ → ~~[illegible]~~ Partial fraction (in matlab)

laplace table

عشان نرجع ال function لاصلها ونستخدم

from table :- $C(t) = \frac{2}{5} + \frac{3}{5}e^{-5t}$

↳ يختفي مع الزمن ↳

so → $\frac{2}{5}$ is the steady-state

• zeros with Poles ~~affect~~ generate the final amplitudes

inputs → to reach the steady-state

Poles → الطريقة الي وصلنا فيها لل $\frac{2}{5}$ وكان فيها e^{-5t} ب smooth

- كل ما كان ال Poles ابعد باتجاه left x-axis تكون الاخذ راجع وبوصل ال system لل steady-state بشكل اسرع .

- بال right half plane البيرل output تكبر يعني بصير unstable.

- $\frac{k_2}{s+2} \rightarrow k_2 e^{-2t}$ / $\frac{k}{s+3} \rightarrow k e^{-3t}$ ... etc

  ↳ natural response ~~witch~~ which will decay over time.

- من ال time-domain صعب نحدد اذا ال system تكون stable

- a → exponential frequency → time to reach steady state.
  - as a increases, the system reaches the steady-state faster.
- time constant = $1/a$
- لو عندي e^{-5t} فال Time constant بساوي $\frac{1}{5}$ يعني بعد $\frac{1}{5}$ ثانية ال system بوصل ل 63% من ال steady-state .

- Note :- ilaplace (F) command in matlab returns the inverse laplace transform of F (in time domain).

- rise time → The time for a signal to go from 10% of its value to 90% of its value.

$$T_r = t_{90} - t_{10}$$

Scanned with CamScanner

- settling time → time to reach 2% of the final value.

Rise time , settling time, Time const الـ بعرف $\boxed{\text{first order systems}}$ بال .

- Rise time $(T_r) = t_{90} - t_{10} = \frac{2.2}{a}$
- settling time $(T_s) = 4/a$

} → مش مطلوب الاشتقاق

<u>problem</u>

- A system has a transfer function $G(s) = 50/s+50$ find the time constant T_c, settling time T_s, and rise time T_r.

Answer :- $a = 50$, so time constant $T_c = \frac{1}{50} = 0.02$

$T_r = 2.2/50 = 0.044$

$T_s = 4/a = 0.08$

- In second order systems → changing a will change the form of the response not only the speed.

- Overdamped → This function has a pole that comes from from the unit step function (steady-state) and <u>Two</u> <u>real</u> poles $-\sigma_1, -\sigma_2$ that come from the system. (no complex roots)

- Note : euler identity → $e^{jw} = \sin(wt) + j\cos(wt)$

• Underdamped → roots are real and complex (Poles).

↳ بالبداية بتكون sin wave

ومع الزمن ال amplitude بتقل لا فها

↓ exp. ب enveloped

half بتكون بال

18/4

• Undamped → Purely imaginary poles without a real part (no exponential Parts)

↳ so, sin wave

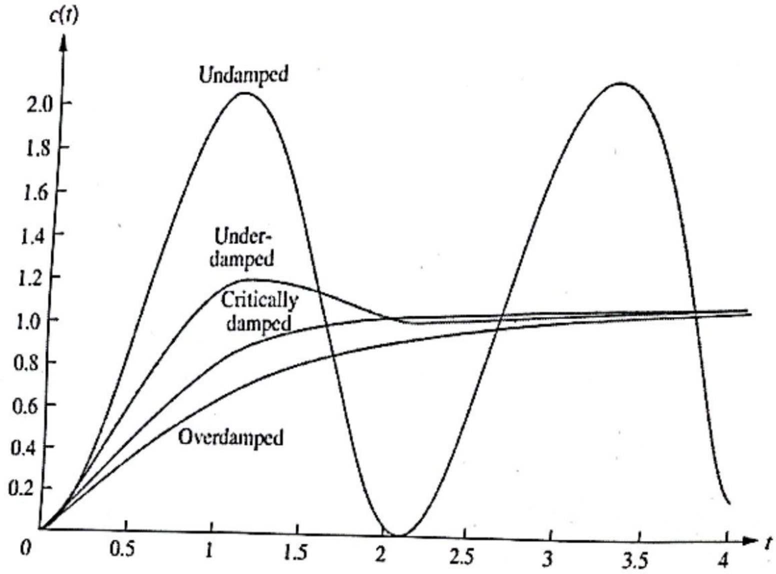
• Critically damped → both roots are the same. (متساويين)

• بتشبه ال overdamped بس بالاداء واضح اكثر

• The output can be estimated as → $c(t) = k e^{-\sigma_1 t} + k_2 t e^{-\sigma_1 t}$

↳ ظهرت t عبارة عن independent Variable → يعني صار time variant

Second Order Systems

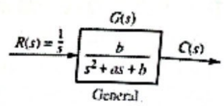


لما اعطيه input رح يوصل للoutput بوحدة من هاي الأشكال. (بده وقت ليوصل للجواب).

بس بدنا نصمم سستم يختار من هاي الأشكال، بحيث تكون استجابته مناسبة.

The General Second Order System

20



Natural Frequency, ω_n of a second-order system is the frequency of oscillation of the system without damping

Damping Ratio, ζ compares the exponential decay frequency of the envelope to the natural frequency

$$\boxed{\zeta = \frac{\text{Exponential decay frequency}}{\text{Natural frequency (rad/second)}} = \frac{1}{2\pi}\frac{\text{Natural period (seconds)}}{\text{Exponential time constant}}}$$

For the underdamped system, the complex poles have a real part, σ

We can quickly derive the value of the real part of the poles from the second order equation $s^2 + as + b$

The real part of the pole σ is equal to $-a/2$.

$$\zeta = \frac{\text{Exponential decay frequency}}{\text{Natural frequency (rad/second)}} = \frac{|\sigma|}{\omega_n} = \frac{a/2}{\omega_n} \longrightarrow a = 2\zeta\omega_n \longrightarrow \boxed{G(s) = \frac{\omega_n^2}{s^2 + \underbrace{2\zeta\omega_n}_{a} s + \underbrace{\omega_n^2}_{b}}}$$

$$\omega_n = \sqrt{b}$$

$$s_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$$

Scanned with CamScanner

- Underdamped systems
  - → decay frequency (exp.)
  - → natural frequency (sin wave)

- ζ (overall damping ratio) $= \dfrac{\text{decay frequency}}{\text{Natural frequency}} = \dfrac{\text{Natural Period} \cdot \frac{1}{2\pi}}{\text{exp. time constant}}$

  "How the exp. part is damping the sinusoidal part."

  equivalent

second order system
$$\frac{b}{s^2 + as + b} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

$$\boxed{\begin{aligned} \omega_n^2 &= b \\ 2\zeta\omega_n &= a \end{aligned}}$$

$\boxed{\text{natural frequency } (\omega_n) = \sqrt{\omega_n^2} = \sqrt{b}}$

عشان نقدر نشتق ال natural freq. / exp. decay frequency بالنظر

$\boxed{\text{decay frequency } (a) = 2\zeta\omega_n}$

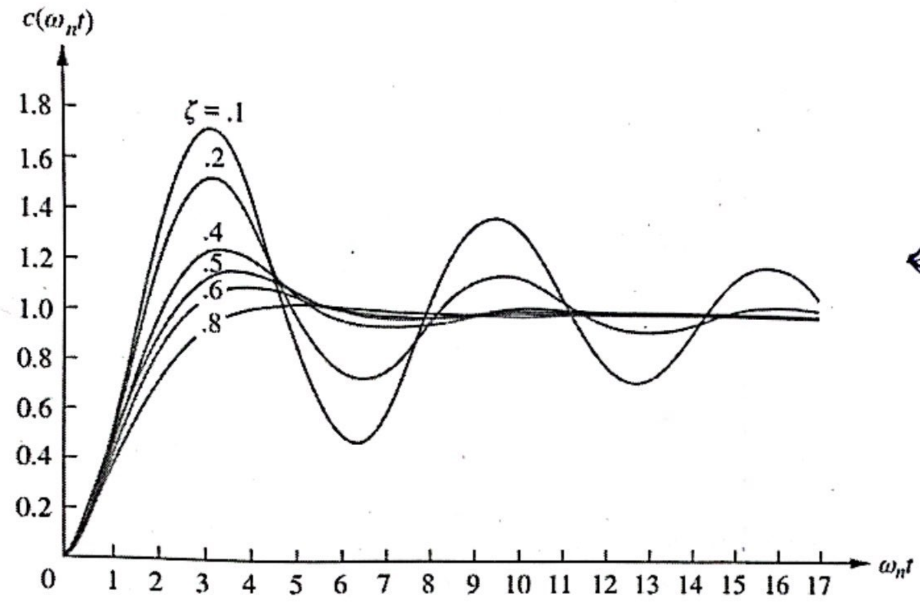
Note :- لما نزيد ال damping ration بيصير ال exp part اقوى وبزيد ال damping
بيصير ال exp decay اعلى وبروح باتجاه الشمال اكثر.

$\zeta = 0 \rightarrow$ Undamped
$0 < \zeta < 1 \rightarrow$ Underdamped
$\zeta = 1 \rightarrow$ Critically damped
$\zeta > 1 \rightarrow$ Overdamped.

بنعرف شكل الانتقال من قيمة ζ.

Effect of ζ (dampening ratio) Visualized

21



نفس ال Frequency (ω_n)

. اذا كانت جميع ال sin waves تتقاطع بنفس النقطة ← نفس ال Frequency

$\boxed{\uparrow \zeta \ \uparrow \text{damping}}$

وبتصير ال Poles عالشمال اكثر

. كلهم بالآخر بوصلوا لل steady state بس باختلاف طريقة الوصول .

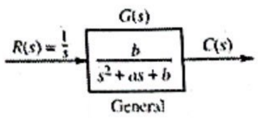
Finding ζ and ω_n For a Second-Order System - Example

22

Given the transfer function of G(s), find ζ and ω_n.

$$G(s) = \frac{36}{s^2 + 4.2s + 36}$$

مطابق للشكل العام البسط في b والمقام في b . غير هيك لازم نحولها للشكل العام .



$$\boxed{G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}}$$

$\omega_n^2 = 36$, therefore $\omega_n = 6$ rad/s

Also, $2\zeta\omega_n = 4.2$, so after substituting the value of $\omega_n = 6$, w get $\zeta = 0.35$

الهم علاقة بال speed ←

• Other parameters associated with the underdamped response :-

1) Rise time (T_r) → time to go from 0.1 to 0.9 of the final val.
2) Peak time (T_p) → time to reach the max. peak.
3) percent overshoot → اعلى قمة بوصلها قبل ما يوصل ال steady-state
4) settling time (T_s) → reach and stay within 2% of the steady-state value على بعد 2% من النتيجة النهائية.

Note • T_r ما اله معادلة فعملوا empirical study واوجدوا علاقة رياضية بتربط ζ مع T_r مضروبة ب ω_n

---

• as the poles move in the vertical direction, the frequency changes. (envelope remains the same).

معامل الخسف ما

• Changing frequency within the same envelope doesn't affect settling time, but it affects the overshoot (as frequency increases the overshoot increases and peak time decreases)

↑ frequency. Higher overshoot, lower peak time and rise time

• changing the damping ratio doesn't affect the peak time, but affects the overshoot and rise time.

↑ damping ratio ↓ overshoot ↑ T_r

- Changing the envelope and the frequency → overshoots are the same / The farther the poles are from the origin, the more rapid the response.

- Third order system ~~...~~ → has 3 poles
  ↳ we'll assume that the third pole is a const. number.

- Case I → لما تكون ال third pole قريبة بال left half صعب نطبق معادلات ال second order.
- Case II → " " " " " بعيدة " " بمقدار، نطبق المعادلات وبنعامله معاملة ال second order sys. → او اكثر بعيدة بمقدار 5 اضعاف عن ال roots الأساسية
  ↳ dominant roots
- Case III → اذا كان third pole = ∞ بنعتبره second order system.

- Additional real zero.
  - اذا كان ال zero بعيد 5 اضعاف عن a بجوز تطبيق المعادلات ونعامله معاملة second order system.

- if the zero is in the right half → nonminimum - phase system
  ↓
  تكون stable بس الاستجابة عكس الي لازم يكون
  لانه ال roots
  ال left

Exercise

- Find the step response of the following transfer functions and compare them. Which one of T_2 or T_3 can we apply the performance equations we learnt to?
- It is clear that T_1 is a second-order system while T_2 and T_3 are third order systems
- There is a frequency component ω_n = sqrt(24.542) = 4.954, therefore ζ = 4/(2* 4.954) = 0.4037, so T_1 is an underdamped system
- We notice that T_2 and T_3 have identical complex poles like T_1 but with extra real poles, one at -10 for T_2, and one at -3 for T_3, so expect the shapes of T_2 and T_3 to have similar shapes to T_1, but the effect of the dampening by the extra real pole will change them.
- We must know if the new shape will still resemble a second order system, so that we can approximate the third-order system as a second-order system and use the equations or not.

$$T_1(s) = \frac{24.542}{s^2 + 4s + 24.542}$$

$$T_2(s) = \frac{245.42}{(s+10)(s^2 + 4s + 24.542)}$$ → Third

$$T_3(s) = \frac{73.626}{(s+3)(s^2 + 4s + 24.542)}$$

بجوز تطبيق المعادلات هون

ال 10 ابعد بـ 5 مرات عن ال 2 فبنعتبر second order

ممكن باستخدام ال matlab نرسم plot لكل system ونشوف اي واحد اقرب لل second order.

Solutions- Numerical

- We should find the location of the complex poles , and see if the real pole is five times further away to the left
- We know from the equations that the complex pole location will be at $-\zeta\omega_n = -\sigma/2 = -0.4037 \times 4.954 = -2$
- For T_2, α_r = -10 which five times larger than -2, so this system can be approximated as a 2nd order system. We can use the equations we learnt for settling time, overshoot, rise time, and peak time to ***APPROXIMATE*** the solutions.
- For T_3, α_r = -3 which 1.5 times larger than -2, this system <u>**cannot**</u> be approximated as a 2nd order system. We cannot use the equations we learnt for settling time, overshoot, rise time, and peak time.

"steady-state error"

- step inputs represent Constant position and thus are useful in determining the ability of the control system to position itself with respect to target.

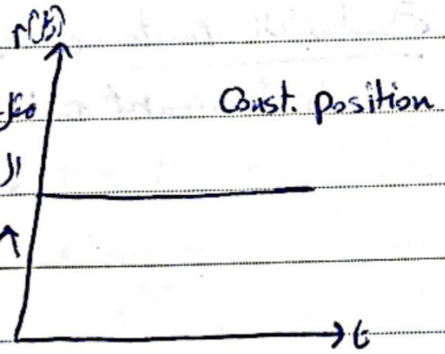
- Ramp inputs represent Constant-velocity inputs to a position control system by their linearly increasing amplitude

- Parabolas, represent constant acceleration inputs to position control systems.

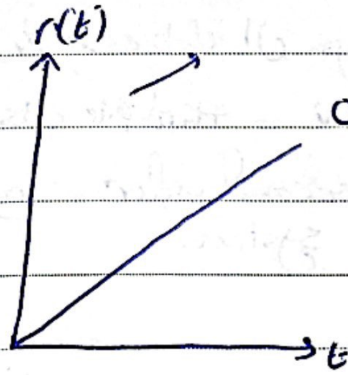
- So, inputs :-

بكل لحظة بكون تزيد عالخط

مثلا تحرك الريش عشان ↑ القناه

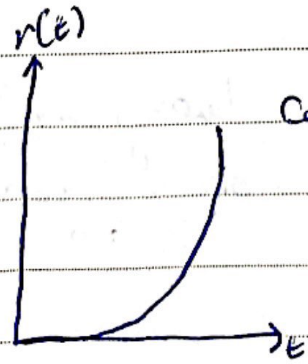


step



Ramp

← مثل الصواريخ



Const. acceleration.

Parabola

• steady-state error reasons :-

1) our assumptions were that the systems are linear, but in reality they are not linear.
2) initially systems work as linear then non-linear
3) related to the design of the control systems

$$e_{steady\text{-}state} = \frac{1}{k} C_{steady\text{-}state}$$

• لازم يكون ال error قليل جداً عشان ($k \approx \infty$) ومستحيل تكون ∞

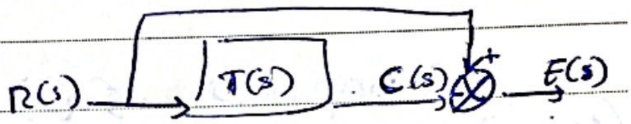
• Final Value Theorem

• هدفنا انه كل ما يمشي الوقت ال error يقرب للصفر وعند $t=\infty$ = يصير "0"

$$e(\infty) = \lim_{s \to 0} s\,E(s)$$

• Example :-

$E(s) = R(s) \cdot (1 - T(s))$



$$E(s) = \frac{s^2 + 7s + 5}{s(s^2 + 7s + 10)}$$

$$e(\infty) = \lim_{s \to 0} s\,E(s)$$

in matlab

```
syms s
E = " "
limit(s*E, s, 0)
ans = 1/2
```

لـ تعتبر قيمة عالية.

ال system بعيد 50% عن ال steady-state

• دائماً اول خطوة لازم نتأكد انه ال system يكون stable عن طريق ال poles.

لكل ال system (لازم يكون stable).

Slide 9

$$e(\infty) = \lim_{s\to 0} \frac{sR(s)}{1+G(s)}$$

↳ the answer may = 0 → Ideal case

↳ " " " be finite

↳ " " " be infinite (error كبير)

. ال stability تحسب لل system كامل فلازم نطلع معادلة ال sys. ونستخدم tf('s')

. رح نتعامل مع closed-loop system with unity feedback

. بنعوض R(s) بالحالات الثلاثة (step / Ramp / parabola)

1) step input (on/off) → $e(\infty) = \frac{1}{1+\lim_{s\to 0} G(s)}$ $\left(\lim_{s\to 0} G(s)\right) = K_p$

↳ $\frac{1}{s}$ $(1u(t))$

2) ramp input → $e(\infty) = \frac{1}{\lim_{s\to 0} s(G(s))}$ $\boxed{\lim_{s\to 0} sG(s)}$ K_v

$(1tu(t))$ ↳ $\frac{1}{s^2}$

3) parabola input → $\frac{1}{\lim_{s\to 0} s^2 G(s)}$

$(1t^2u(t))$ ↳ $\frac{1}{s^3}$

↳ $\lim_{s\to 0} s^2 G(s) = K_a$

. بهمنا ال ٣ حالات
تكون ال error = 0
عن طريق انه تكون lim = ∞
فتكون المقام = ∞

. يعني بهمنا kp / kv / ka تكونوا ∞ فبتقل نسبة الخطأ لصفر
وعشان تكون lim g(s) = ∞ لازم تكون في "s" اضافية ، وكل "s" اضافية
↳ Integrator
بنعكس تأثيرها على كل inputs

• في حال طلعوا معنا ال poles بال RHP، يعني ال system بكون unstable، هنا بكمل حل وما بنطلع e(∞) .

. Note: kv / kp / ka → static error constants

• "Poly" command → roots بوصف المعادلة باستخدام ال
• "tf" command → transfer function
• "pole" command → same as roots ~~rlocus plot ال...~~

• لما تكون "s" اضافية وحدة :- بكون ال step جواب صفر
ال ramp جواب finite
ال Parab. جواب infinite

• لما تكون "s^2" اضافية :- " " " " صفر
" ramp " صفر
" Parab. جواب finite

• "dcgain" command → بحسب K مباشرة
بس لازم ننتبه بحالة ka / kv
بكون G(s) مضروبة ب "s" و "s^2" قبل dcgain .
بنستخدم ال conv

• كل ما نزيد بعدد ال integrators بقل ال error .
ودائماً لما يكون عنا او معطى قيم ل k يعني stable system .

- Steady-State errors with disturbances ($D(s)$)
  - we will assume that $D(s) = \frac{1}{s}$ "step funct"
- when we apply Final Value theorem

$$e_D = -\frac{1}{\lim_{s\to 0}\frac{1}{G_2(s)} + \lim_{s\to 0} G_1(s)}$$ (error due to disturbance only)

↑dc gain of $G_1(s)$ ↓ error by disturbance
↑dc gain of $G_2(s)$ ~~...~~ ↓ error by disturbance
↓
فعشان نقلل الخطأ الخارجي لازم نزيد G_1 ونقلل G_2 (dcgain)
ال dcgain بتم تطبيقها على ال forwardpath فقط.
ال conv ما بتكون بين السستم كامل وانما فقط G يعني سلايد ٩٢ في خطأ.

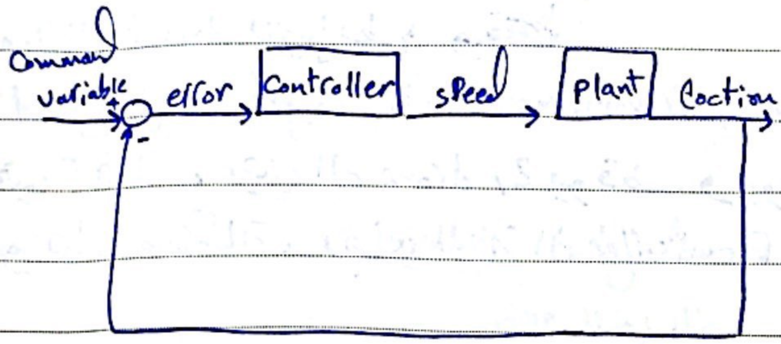
- Steady-state error for Nonunity feedback system
  * بنحوله ل unity بإضافة خطين ↳
  1   -1

slide 21 → $e(\infty) = -4$
↳ يعني لو ال steady-state
كانت 5 وهو اعطاني 20
وهاد خطأ كبير.

18/5

"Quick Intro to PID Controllers"

. PID controller → most famous type of controllers that is designed using software or hardware



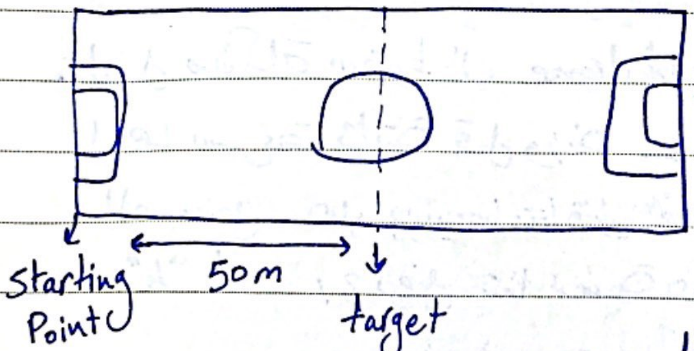
. PID → 3 controllers

Proportional integral derivative

ممكن يكون P لحاله

او PI او PD

او PID



⇒ input = target = 50m

feedback → target - current point = error

↑ current point ↓ error

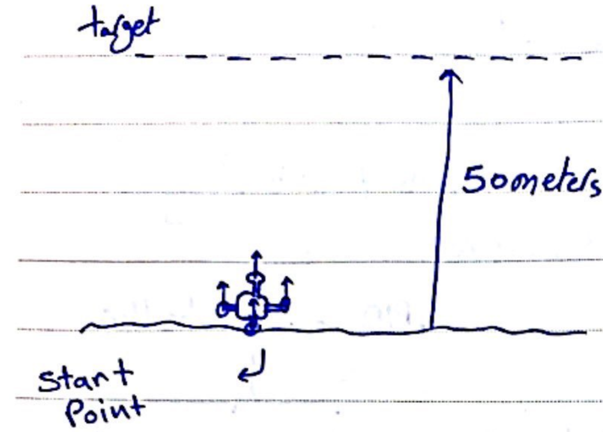
. Here the input command is Proportional with the error.

. بالبداية ال controller يعطي امر بالمشي بسرعة باتجاه ال target، وكل ما قرب من ال target بتصير الحركة ابطئ لحدما يوقف

⇓

Proportional control → مكون من block وحدة

بكون جواتها "k" بالبداية بتكون k كبيرة وبتصير تقل لتوصل لعدد معين Proportional لقيمة k .



error عندها ال 50 بكون

. هلأ عنا ال application
و صار drone .
. سرعة المراوح بالبداية بتكون عالية لأنه بعيد عن ال target
فال error عالي، وكل ما يقترب من ال target بتصير بطيئ.
. لما يوصل لل target ال Proportional , 2 يعطي قيمة صفر، يعني ال drone , 2 يوقف و يوقع.
و هاي مشكلة، بهاي الحالة ال P controller ما بزبط.

. ال Proportional بيسمى present control لانه بس بشتغل على الوضع الحالي.

. لنحل مشكلة سقوط ال drone لما يوصل لل target لازم يكون للمراوح
الها سرعة ثابتة تحمل وزنه ← بس المشكلة هون صارت انه
ال gain صار يزيد ما يتناقص فعلياً ولا مرة بوصل ال drone لل 50
"k" ف , 2 يظل عنا نسبة خطأ. (steady-state error)

→ lets assume we need 100 rpm to hover

Error	x	Gain	=	100 rpm (speed)
50	x	2	=	100
20	x	5	=	100
10	x	10	=	100
1	x	100	=	100

← الحل ← نحوله من P-controller ل PI

← ال integral controller بيسموهم . They remember the past

↓

as is they have memory.

. ال P-controller لـ 2 يعمل وظيفته ويوصل ال drone لاعلى ارتفاع

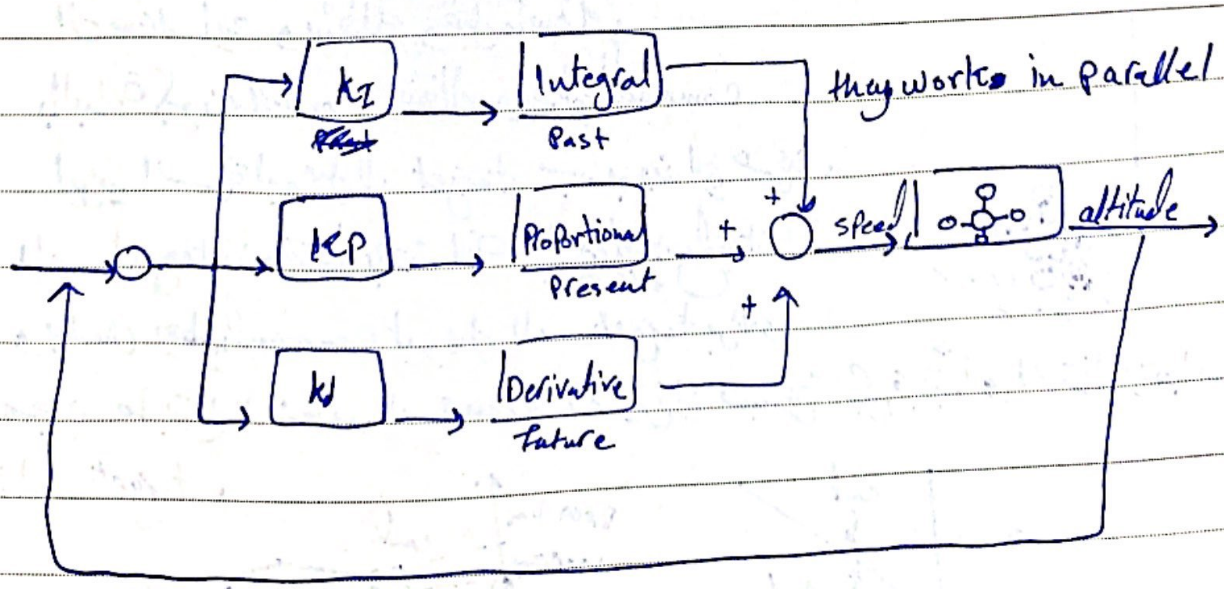
بتكون بس تراقب كيف الخطأ يقل

. ال I-controller بيشتغل بعد ال P-controller، بشوف ال steadystate error وبصير يعطي اوامر حتى يرفعه لل target وبضل يلف بسرعة محددة

. اضافة ال integrator يعني اضافة block بال hardware او code

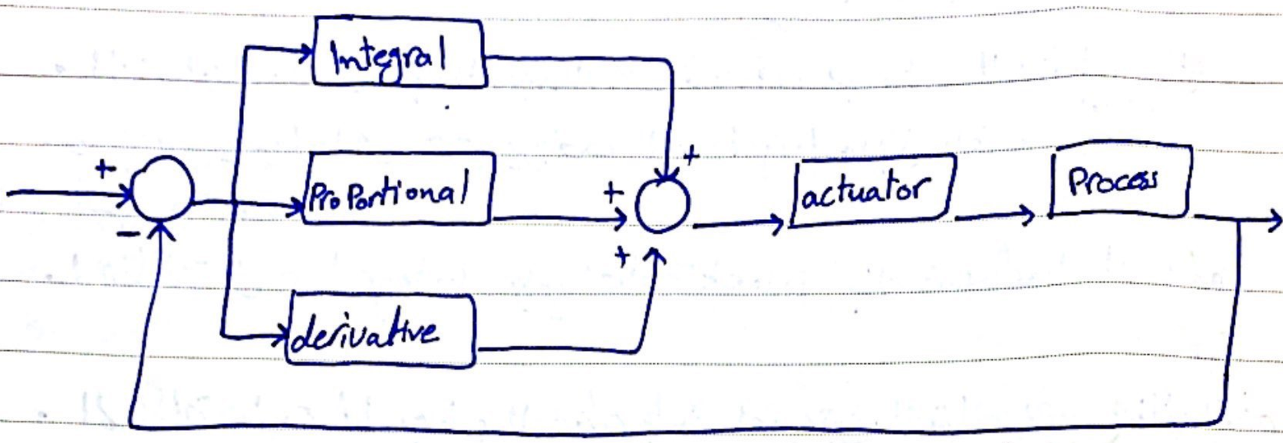
. المشكلة صارت لما يعطي ال integrator امر معين احتمال يصير عنا overshoot ويطلع عن ال target فلازمنا controller يعرف المستقبل ، فبنستخدم ال derivative الي وظيفتها تراقب معدل التغير في الخطأ، وبناءً على معدل التناقص او التسارع يعطي اوامر. اذا ال drone صارت بدها تطلع عن ال 50 meter بصير يعطي اوامر باتجاه معاكس .

. مش شرط بكل ال Applications يكونوا ال 3 موجودين .



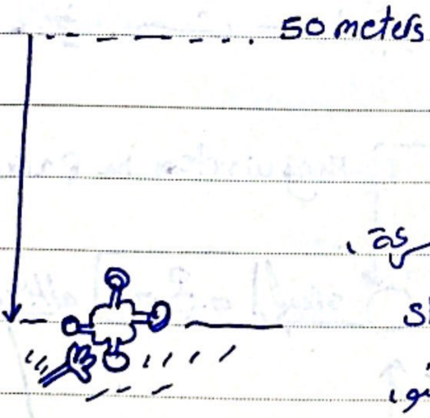
21/5

PID Control

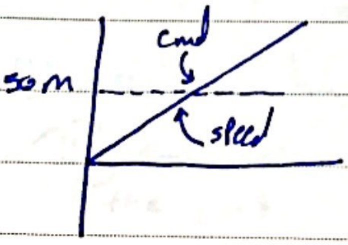


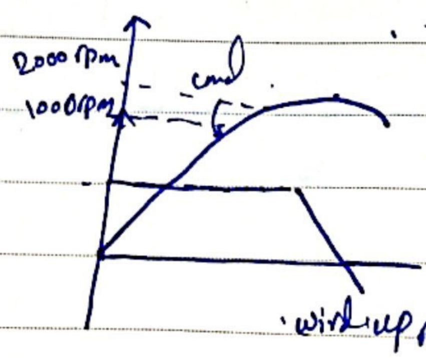
. في مشاكل بتظهر لما نضيف ال Integral منها :-

Integral wind up problem



. لنفرض لهاد المثال انه شخص ماسك ال drone ايده وشغله وضل ماسكه بالبداية بكون P-controller شغال وبعطي command ليزيد السرعة ليوصل ال target بس بدون اي حركة ، ال Integral وقتها بعتقد انه في steady-states error وبلش يعطي commands ليوصل ال target برضو ، مجرد ما يترك الشخص ال drone رح يرتفع بسرعة كبيرة جدا واحتمال يتجاوز ال target .





. تعدى ال 1000 ولبعد ال 2000 لتقدر يرجع للوضع الصح ← wind up problem .

. How to solve the wind up Problem?

using Clambing

. clambing ← عبارة عن if-statment/circuit بينه ال controller وال actuator

. بدنا نتأكد انه الأوامر الي بتطلع من ال controller بتتجاوز طاقة ال actuator وإذا
لـ من الداتا شيت

. مثلاً لو كانت سرعة الدوران القصوى لل drone بتساوي 1000، لو اعطى ال controller امر اكثر منه 1000 يضل 1000، ما نعطيه command يزيد (كاسقف if-stament).

كاسقف 1000 حسب المثال السابق.

. لازم نحقق شرطين بال clamping :-

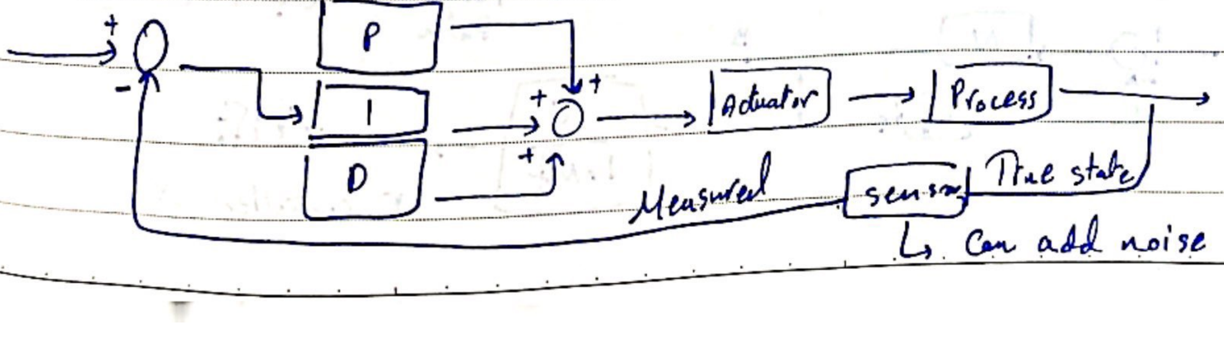
1) يجب الوصول ال saturation level ليضل عندها

2) إذا كان لازم يتحرك لفوق واشارة ال error موجبة، او لازم يتحرك لتحت واشارة ال error سالبة، اذا تساووا بالإشارة لازم ما يكمل حركة.

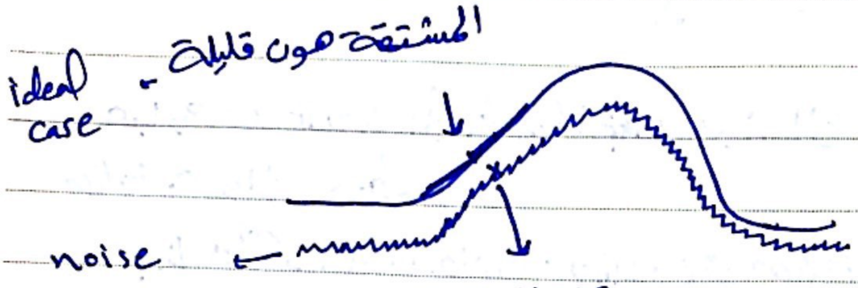
. لما يتحققوا الشرطين بفصل ال integrator.

* شو القيمة الي لازم أعطيها لل clamping؟ مثلاً لو كانت الطاقة الاستيعابية هي 1000 rpm، فهل ال clamping اعطيها 1000؟ دايماً بنحطها أقل (900 مثلاً). لانه مع الزمن طاقته القصوى رح تقل.

. المشاكل الي بتواجه ال derivative سببها الرئيسي هو ال noise بال sensors



. ال derivative بعتمد على معدل التغير في الخطأ، والي هو المشتقة



عند نفس النقطة. ولانه ال signal مايلة صار معدل الخطأ كبير واصبحت قراءات ال D خاطئة.

• الحل ← نستخدم Filter، وهون ال low-pass filter عشان يحجب ال frequency العالية

يحجب → attenuation

• ال low-pass filter اله cut-off freq. ، اي اعلى freq منها بتم حجبه.

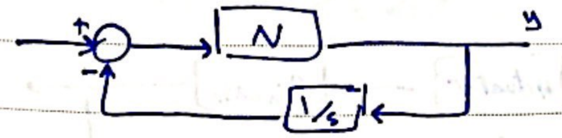
→ [low-pass filter] → [Derivative]

laplace	what is it
S	derivative
$\frac{1}{s}$	integral
$\frac{N}{s+N}$	low pass with cut-off = N

• المشتقة بنضرب بـ s والتكامل بنقسم s.

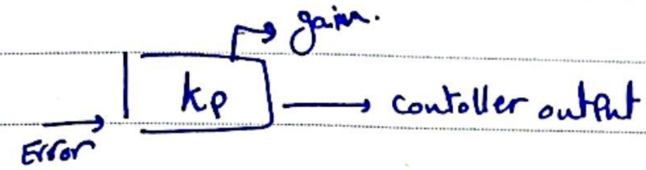
• وعبدا انه التصميم المناسب لل filter هو $\left(\frac{N}{s+N}\right)$ وال N هو ال cut-off

• بنقدر نصمم ال low pass مع ال derivative باستخدام ال integral بال feedback وال proportional بال forward path.



$= \boxed{\frac{N}{1+N\frac{1}{s}}}$ → تكافئ ال D-controller

. P- controller



$$\frac{Kp}{1+Kp}$$

• شو ما اعط قيمة ال gain بضل في عنا نسبة خطأ

• شو ما زدنا بال gain بتقل نسبة الخطأ بس مستحيل توصل للصفر

إلا لو كانت ال gain تساوي infinity وهاد مستحيل

• اهمية وجود ال P انه بقلل من ال disturbances.

• كل ما أزيد قيمة kp بقل ال disturbance ~~x~~ ~~وبتقل ال~~ damping ال overshoot

وبصير ~~smooth~~ smoother .

. PD - Controller

• ال P بتقلل ال disturbance وال D بتقلل ال oscillation

وبعمل smoothing لل signal وهذا اشي كتير مرغوب

• زيادة kd بتزيد ال dampingratio فبقل ال oscillation وال underdamped effect.

• "بتعمل damping لل overshoot" . &

. PI - controller

• ال I وظيفته الأساسية تمنع الخطأ من ال system وال steady-state

---

• اختيار نوع ال controller بيعتمد على نوع ال system

وطبيعته .

23/5

"Brief Introduction to Digital control"

. digital control → based on software (digital computer)

- A/D → Analog to digital (~~sen~~ from sensors)
- D/A → convert the digital commands to analog (digital to analog).
  ↳ ex:- PWM "Pulse width modulation".

. Analog signals are contin. in time and in range.
- A/D → discretization the ~~ag~~ analog signal
  ↳ both time and range

A/D (2 circuits)

sampling ↓ take a sample and hold it (discretization of time)

quantizing ↳ discretization of range
ex:- $V(\cancel{1} \rightarrow 2) = 0$

. remember :- $f_{sampling} = 2 \times f_{max}$

- n-bit A/D → # of levels = 2^n

↑ # of levels
- more accurate
- slower
- more expensive

- A/D ≡ ADC

- ZOH → generates the stair-step approximation. (discrete values)
  ↳ sample and hold circuit.

- example :- 3-bit A/D with range from 0V to 5V
  ↳ we have 8 levels

- range of each level → $\frac{5-0}{8} = 0.625$

- 0 → 0.625 → 000
- 0.625 → 1.25 → 001
- 1.25 → 1.875 → 010

⋮

etc

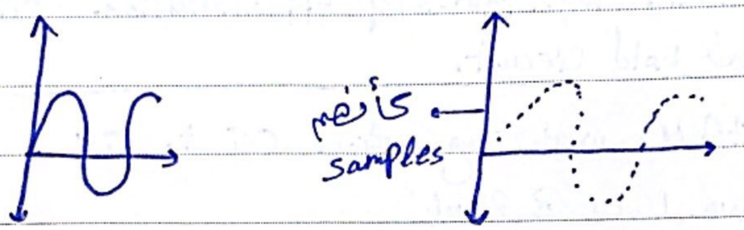
- في مشكلة ← نسبة خطأ عالية لانه مثلاً بعتبر 0 و 0.62499 نفس الأشي مع انه الأقرب الها 001 / الحل نزيد ال resolution عن طريق زيادة عدد ال levels .

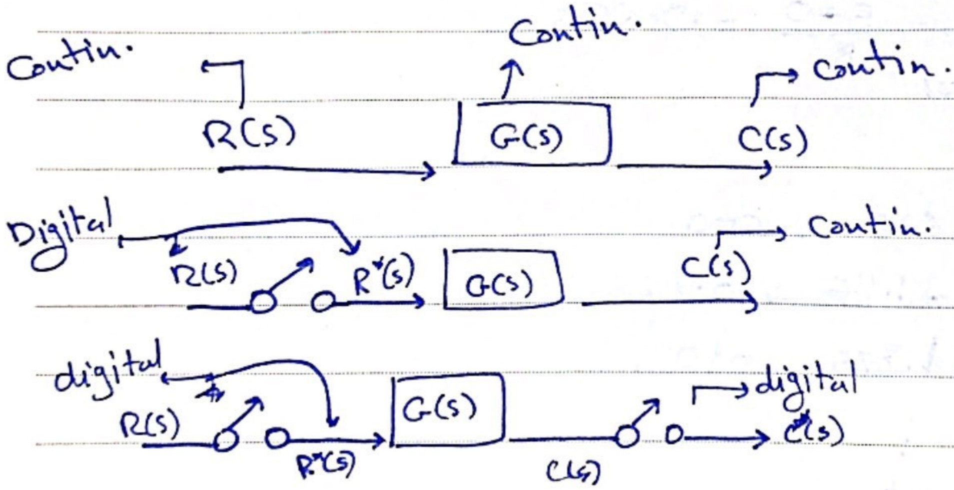
- The laplace transform for the zero-hold circuit is given by :-

$$G_h(s) = \frac{1-e^{-Ts}}{s}$$ where T = sampling period.

- Ztrans (From time domin to frequency domain for discrete values/signals)
  ↳ instead of laplace command in Analog.

- n → instead of t | • z → instead of s
- Contin. time sin(n) • discrete time sin(n)





↳ Phantom (imaginary)
" مش موجود physically بس "
تعبير عن ال discrete

- C2d → Convert the block to digital block
  بيغير G(s) بـ G(z).

$$R(s) \multimap R^*(s) \rightarrow \underbrace{\boxed{G_1(s)} \xrightarrow{\text{contin.}} \boxed{G_2(s)}}_{\text{contin.}} \xrightarrow{C(s)} \multimap C^*(s) \rightarrow$$

$$\equiv$$

$$R(z) \rightarrow \boxed{G_2G_1(z)} \rightarrow C(z)$$

but if there is a sampling switch between $G_1(s)$ and $G_2(s)$

$$R(z) \rightarrow \boxed{G_2(z)G_1(z)}$$

• اي سستم اله output وهاد ال output ليس input ~~ممم~~
لى سستم تاني (يعني بكون output نهائي) ← بنقدر
نضيف عليه phantom sampler.

• Phantom ممكن نضيفه على نقاط التفرع.
sampler بشرط يكونوا على نفس ال sampling freqeny

then we can start
with block simplification

- In digital systems stability
  - take the magnitude of poles
    - if the magnitude $< 1 \rightarrow$ stable
    - " " " $> 1 \rightarrow$ unstable
    - if there is only one "1" $\rightarrow$ marginally stable
    - if there are more than one root with magnitude $= 1 \rightarrow$ unstable.

- remember :- magnitude of complex numbers $= \sqrt{a^2+b^2}$ or use abs() command

---

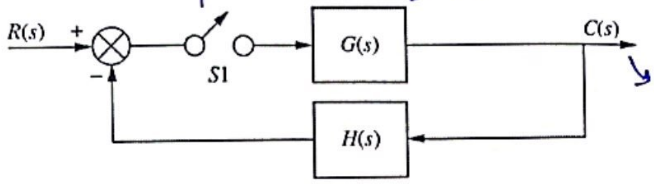
- why does the sampling rate affects the stability of the system?

- لما تكون ال sampling rate كبيرة يعني كوننا نقيس بسرعة عالية، وممكن تكون أكبر من سرعة الاستجابة للسستم.

- Hardware in the loop $\rightarrow$ in simulink "matlab", to test the hardware on sampling rate or parameter values

Digital Block Simplification Example

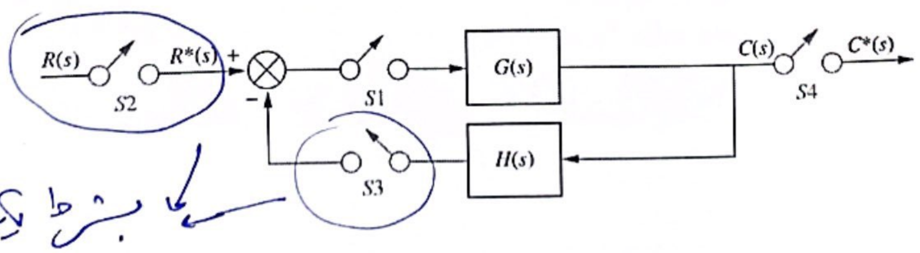
sample بتعمل ل R و H فبنوديه



Phantom sampler

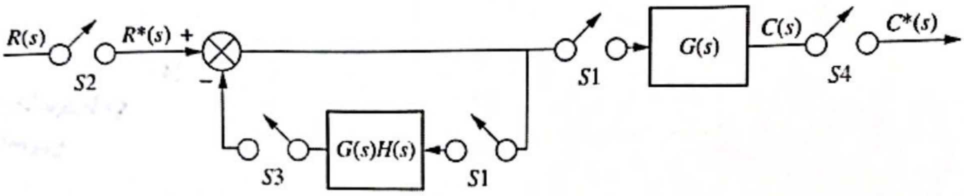
Digital Block Simplification Example – cont.

- Adding Phantom samplers per the rules



بشرط تكون على نفس ال frequency

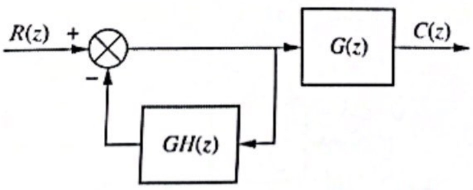
Digital Block Simplification Example – cont.

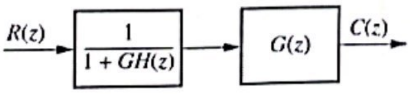


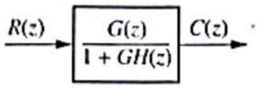
عشان نحقق شرط

sample-block-sample

Digital Block Simplification Example – cont.

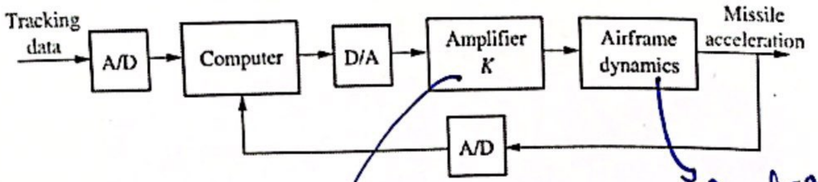






Example

- The missile shown can be aerodynamically controlled by torques created by the deflection of control surfaces on the missile's body. The commands to deflect these control surfaces come from a computer that uses tracking data along with programmed guidance equations to determine whether the missile is on track. The information from the guidance equations is used to develop flight control commands for the missile. A simplified model is shown as well.
- Here the computer performs the function of controller by using tracking information to develop input commands to the missile. An accelerometer in the missile detects the actual acceleration, which is fed back to the computer.
- Find the closed-loop digital transfer function for this system and determine if the system is stable for K = 20 and K = 100 with a sampling interval of T = 0.1 second.

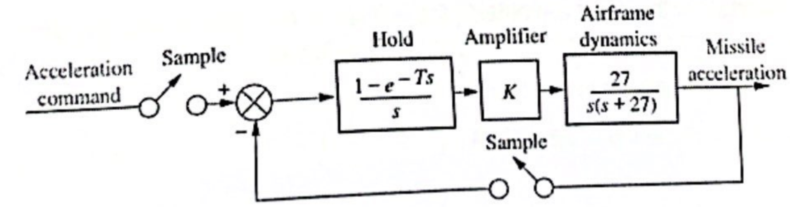


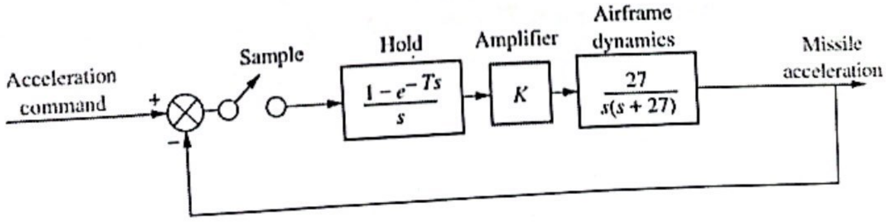
Analog system

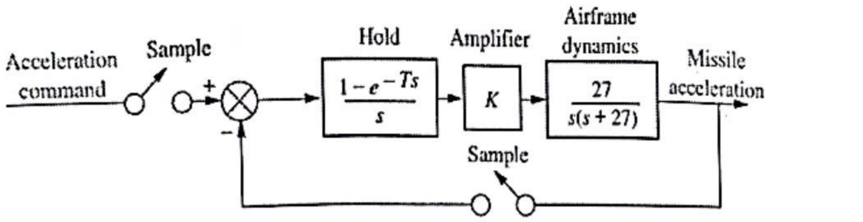
عشان نزيد ال voltage
بكون اله قيمة معينة
ممنوع تزيد عنها

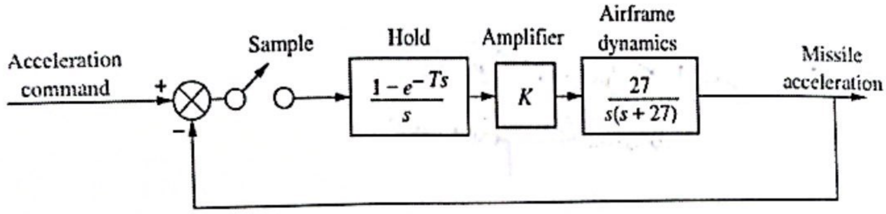
Computer ≡ ZOH circuit.

Example – cont.









Example MATLAB Solution I

عبارة عن السستم مع ال zoH بنفسه بهاي الطريقة

$s^2 + 27s + 0$

```
numg=27;             % Define numerator of Ga(s).
deng=[1 27 0];       % Define denominator of Ga(s).
Ga=tf(numg,deng); % Create and display Ga(s).

Gz=c2d(Ga,0.1,'zoh') % Find G(z) assuming Ga(s) in cascade with z.o.h. and display.
```

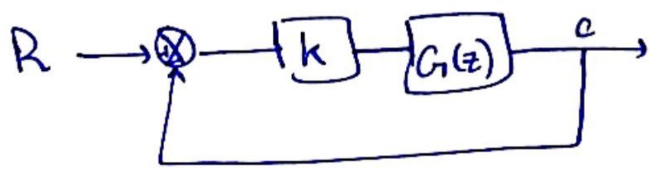
given

```
Gz =

    0.06545 z + 0.02783
  ----------------------
  z^2 - 1.067 z + 0.06721

Sample time: 0.1 seconds
Discrete-time transfer function.
```

السستم كامل باستثناء قيمة K



Example MATLAB Solution II

50 system

```
for K=1:0.1:50; % Set range of K to look for % stability.
    Tz=feedback(K*Gz,1); % Find T(z).
    r=pole(Tz);          % Get poles for this value of K.
    rm=max(abs(r));      % Find pole with maximum absolute value for this value of K.
    if rm>=1, % See if pole is outside unit circle.
        break; % Stop if pole is found outside unit circle.
    end
  end
display(K)
```

```
K = 33.6000
```

اي قيمة اعلى من K بصير السستم unstable

```
display(r)
```

```
r = 2×1 complex
   -0.5660 + 0.8257i
   -0.5660 - 0.8257i
```


Steady-State Errors in Digital Systems

- Any general conclusion about the steady-state error is difficult because of the dependence of those conclusions upon the placement of the sampler in the loop.
- Remember that the position of the sampler could change the open-loop transfer function.
- In the discussion of analog systems, there was only one open-loop transfer function, G(s), upon which the general theory of steady-state error was based and from which came the standard definitions of static error constants.
- For digital systems, however, the placement of the sampler changes the open-loop transfer function and thus precludes any general conclusions. In this chapter, we assume the typical placement of the sampler after the error and in the position of the cascade controller, and we derive our conclusions accordingly about the steady-state error of digital systems

Good Luck!