EECE 5610 Digital Control Systems

Lecture 3

Milad Siami
Assistant Professor of ECE
Email: m.siami@northeastern.edu

Northeastern University
College of Engineering

Updates

- HW#1 is posted
- Lectures 1-2 and recorded videos are posted
- Quick Review of Laplace Transform is posted

Signal Flow Diagrams and Mason's Formula

They provide an alternative representation of Transfer Function relationships and an alternative (often simpler) to Cramer's rule or block diagram manipulations for computing T. F.

Signal Flow Diagrams and Mason's Formula

Rules:

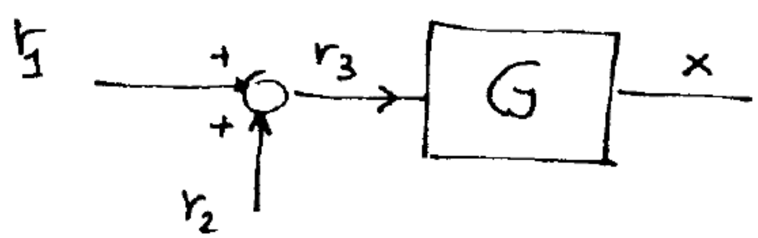
- Each signal is represented by a node
- Each transfer function is represented by a branch (arrow)

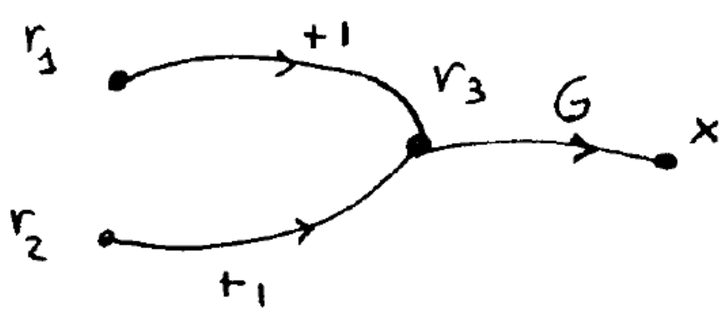
Block Diagram:

Signal flow:

$$x(s) = G(s)\,r(s)$$

- Summing junctions are represented implicitly: all the inputs converging to a node are added together:



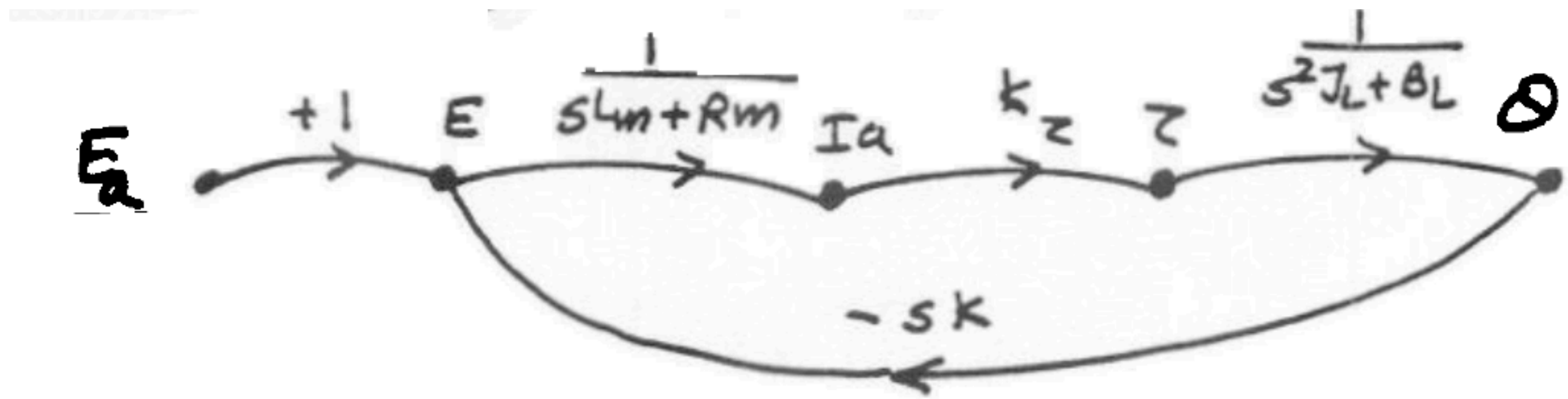


$$r_3 = (+1)\cdot r_1 + (+1)\cdot r_2$$

$$x = G\, r_3$$

Signal Flow Diagrams and Mason's Formula

Question: Find the signal flow graph representation of the DC motor:



Signal Flow Diagrams and Mason's Formula

Some Terminology:

source node: A node that has all signals flowing away from it.

sink node: A node with incoming signals only

Path: Continuous connection of branches between 2 nodes (directed)

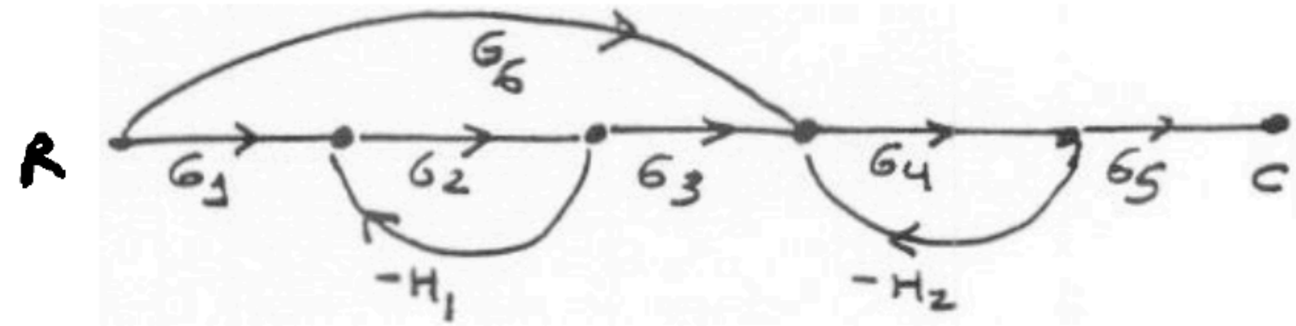
Loop: Closed path (i.e starting node = finishing node)

Path (loop) Gain: Product of all T.F. of all the branches in the path (loop)

Non Touching loops (paths): Loops that do not have any **nodes** in common.

Signal Flow Diagrams and Mason's Formula

Example:



2 loops: $-G_2H_1$ (L_1)

$-G_4H_2$ (L_2)

Path $G_6G_4G_5$ does not touch L_1

Path $G_1G_2G_3G_4G_5$ touches both L_1 and L_2

- Mason's Formula (section 2.4) Provides an alternative to Cramer's rule or elimination for finding Transfer Functions

$$T_{CR} = \frac{1}{\Delta}\sum_{k=1}^{P} M_k\Delta_k = \frac{1}{\Delta}\left(M_1\Delta_1 + \quad + M_p\Delta_p\right)$$

Signal Flow Diagrams and Mason's Formula

- Mason's Formula (section 2.4) Provides an alternative to Cramer's rule or elimination for finding Transfer Functions

$$T_{CR} = \frac{1}{\Delta} \sum_{k=1}^{P} M_k \Delta_k = \frac{1}{\Delta} \left(M_1 \Delta_1 + \ldots + M_P \Delta_P \right)$$

Where:

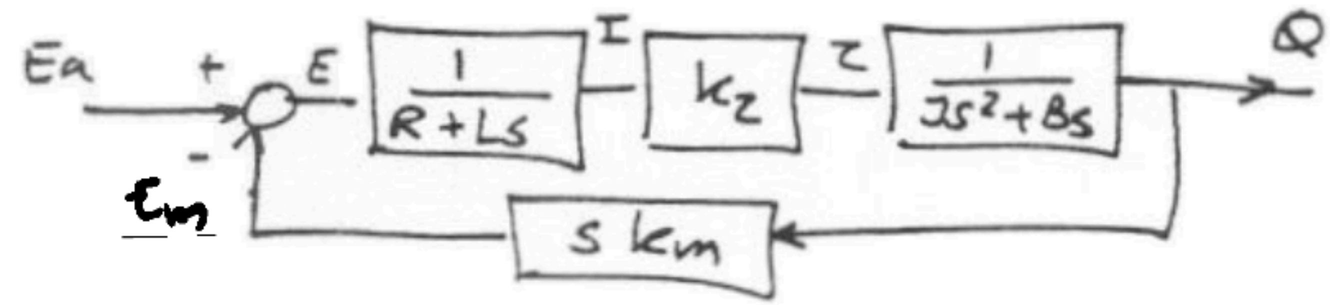
- $\Delta = 1 - \left(\Sigma \text{ gains individual loops} \right) + \Sigma \left(\text{products of } \underline{\text{non-touching}} \text{ pairs of loops} \right)$

  $- \; \Sigma \left(\text{products of triplets of } \underline{\text{non-touching}} \text{ loops} \right)$

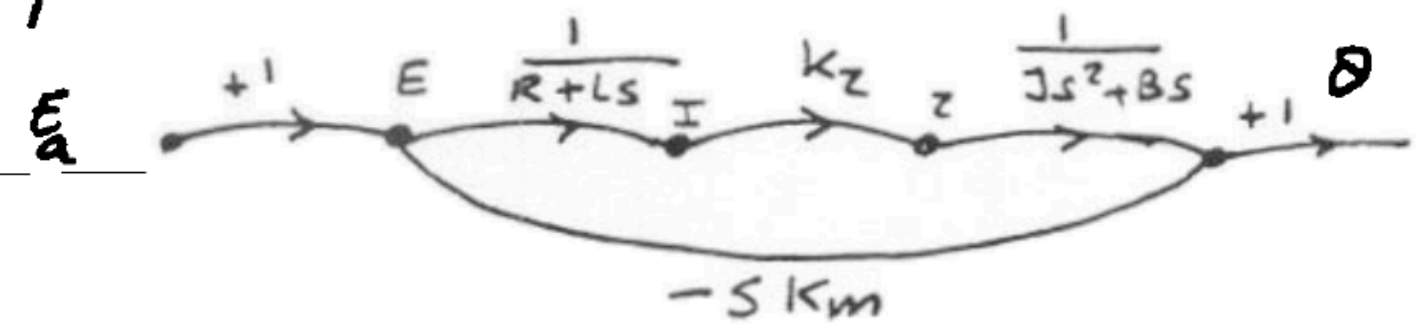
  $+ \; \ldots$

- $M_k =$ Gain of the k^{th} path between R and C
- $\Delta_k =$ Value of Δ when the nodes in the path M_k are removed from the graph

Example 1: DC motor:



Signal flow graph:



1 loop: $L_1 = \dfrac{-k_z k_m s}{(R+Ls)(Js+B)s} \Rightarrow \Delta = 1 + \dfrac{k_z k_m s}{(R+Ls)(Js+B)s}$

Only 1 path from E_a to θ: $M_1 = \dfrac{k_z}{(R+Ls)(Js+B)s}$

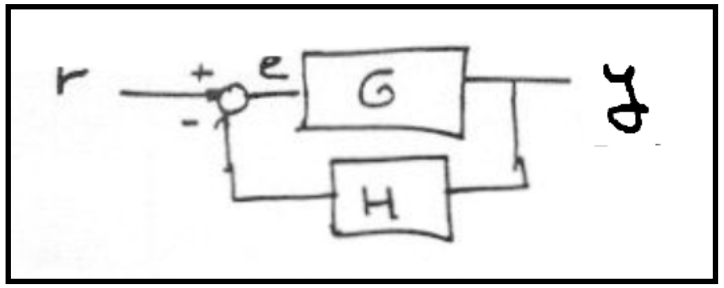
$\Delta_1 = 1$

$$T_{\theta E_a} = \frac{1}{\Delta} \cdot M_1 \Delta_1 = \frac{M_1 \Delta_1}{\Delta}$$

$$= \frac{\dfrac{k_z}{(R+Ls)(Js+B)s}}{1 + \dfrac{k_z k_m s}{(R+Ls)(Js+B)s}} = \frac{K_z}{(R+Ls)(Js+B)s + k_m k_z s} \quad \#$$

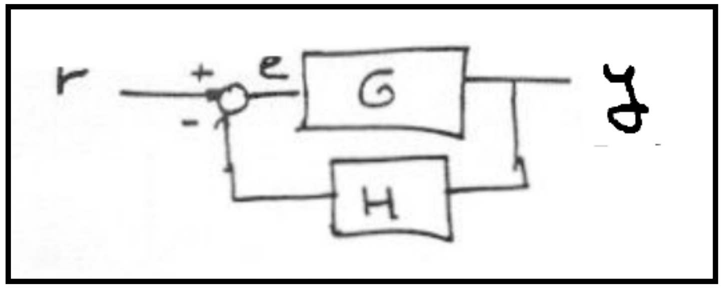
Signal Flow Diagrams and Mason’s Formula

Note: This is a special case of

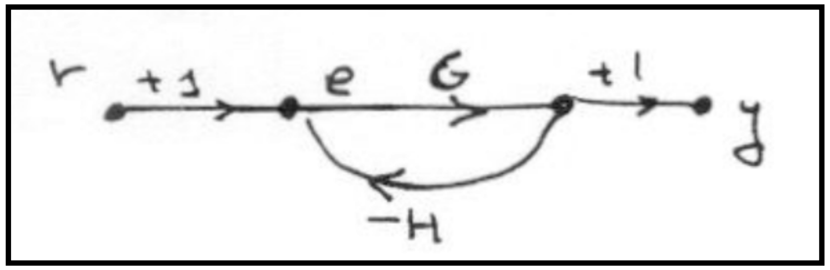


Signal Flow Diagrams and Mason's Formula

Note: This is a special case of

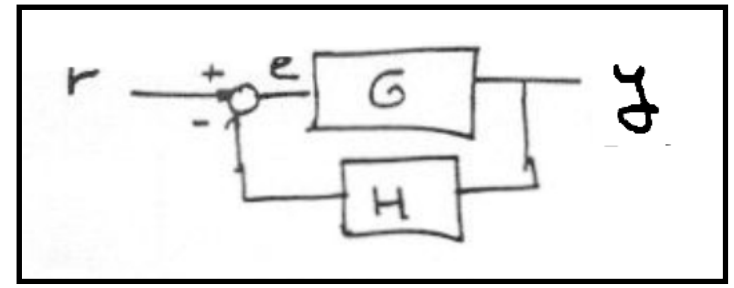


In the signal flow form:

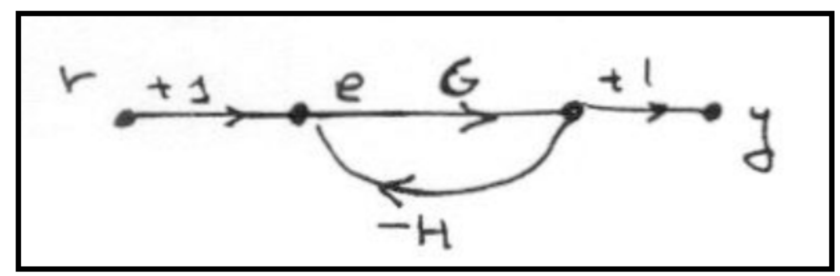


Signal Flow Diagrams and Mason’s Formula

Note: This is a special case of



In the signal flow form:



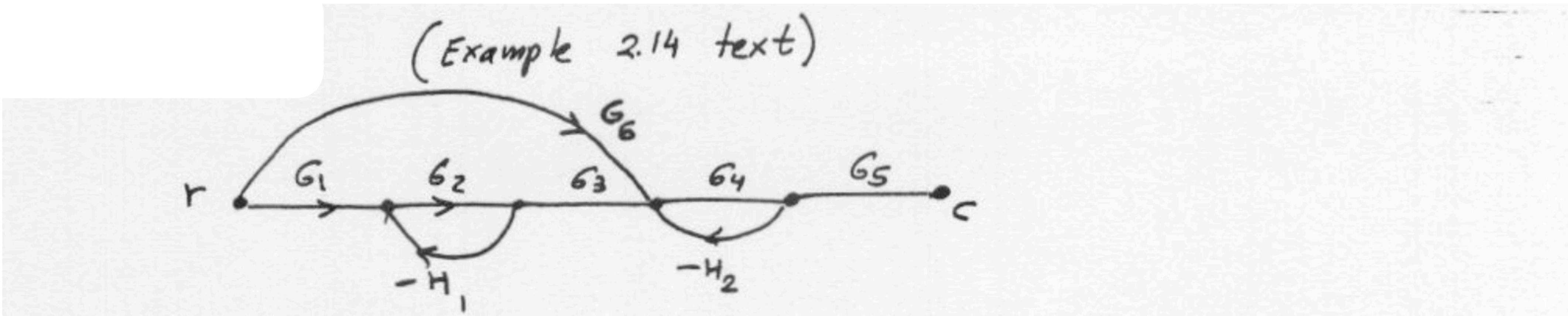
$$L_1 = -GH$$

$$\Delta = 1 - L_1 = 1 + GH$$

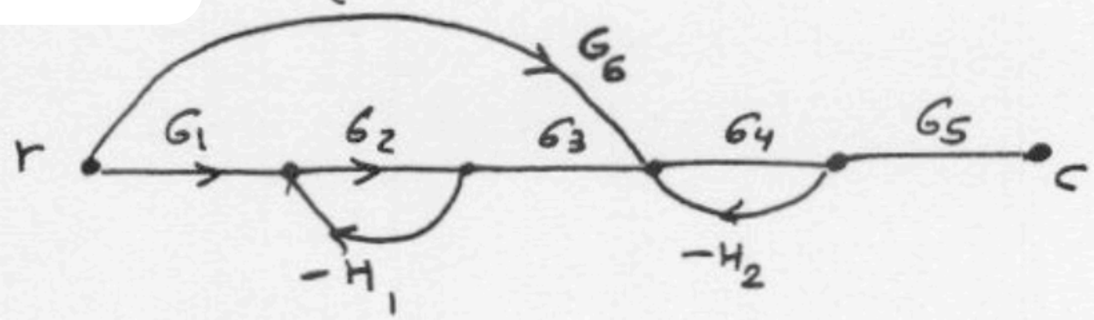
$$T_{yr} = \frac{M}{\Delta} = \frac{G}{1+GH} \quad \#$$

Example:

Example:



Example:



Number of paths between r & c: 2 — $M_1 = G_1 G_2 G_3 G_4 G_5$

$M_2 = G_6 G_4 G_5$

Number of loops: 2

$L_1 = -G_2 H_1$ (non touching)

$L_2 = -G_4 H_2$

$$\Delta = 1 - \Sigma \text{ loops} + \underset{(N.T.)}{\Sigma \text{ pairs}} = 1 + G_2 H_1 + G_4 H_2 + G_2 G_4 H_1 H_2$$

Path M_1 touches both loops $\Rightarrow \Delta_1 = 1$

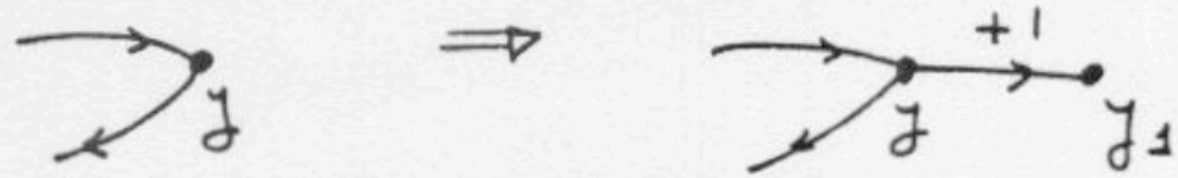
Path M_2 touches only L_2 $\Rightarrow \Delta_2 = 1 + G_2 H_1$

$$T_{cr} = \frac{1}{\Delta}\left(M_1 \Delta_1 + M_2 \Delta_2\right) = \frac{G_1 G_2 G_3 G_4 G_5 + G_6 G_4 G_5 (1 + G_2 H_1)}{1 + G_2 H_1 + G_4 H_2 + G_2 G_4 H_1 H_2}$$

Important: Technically, Mason's formula is valid to compute the TF **ONLY** from a **source** node to a **sink** node

Important: Technically, Mason's formula is valid to compute the TF **ONLY** from a **source** node to a **sink** node

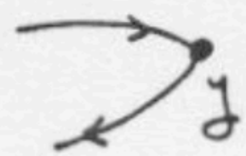
The restriction of the output being a sink node is easy to remove: add an extra branch with gain 1



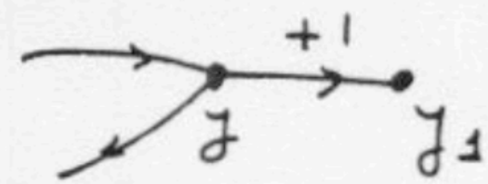
(This essentially says $y = y_1$)

Important: Technically, Mason's formula is valid to compute the TF ONLY from a source node to a sink node

The restriction of the output being a sink node is easy to remove: add an extra branch with gain 1



$\Longrightarrow$



(This essentially says $y = y_1$)

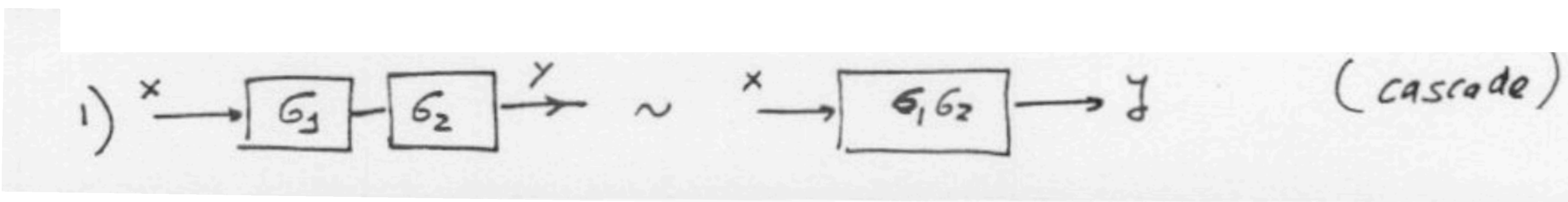
However, the restriction of the input being a source node can't be dealt with in this form.

Basic Operations on Systems (Block Diagrams)

Here we are going to learn how to operate on block diagrams. This will allow us to obtain simpler (hopefully) diagram

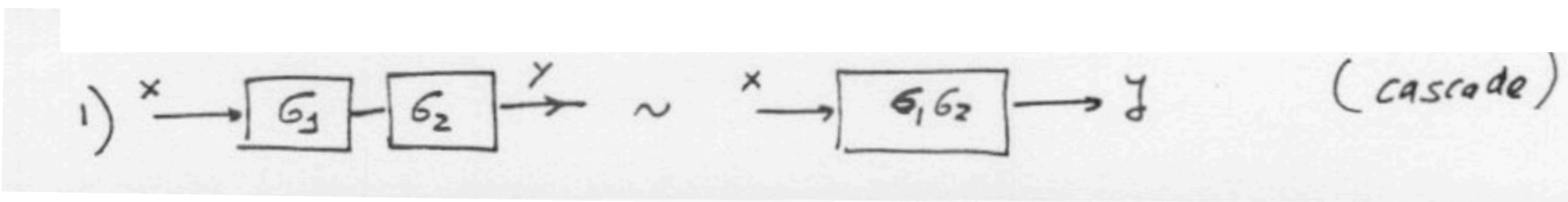
Basic Operations on Systems (Block Diagrams)

Here we are going to learn how to operate on block diagrams. This will allow us to obtain simpler (hopefully) diagram

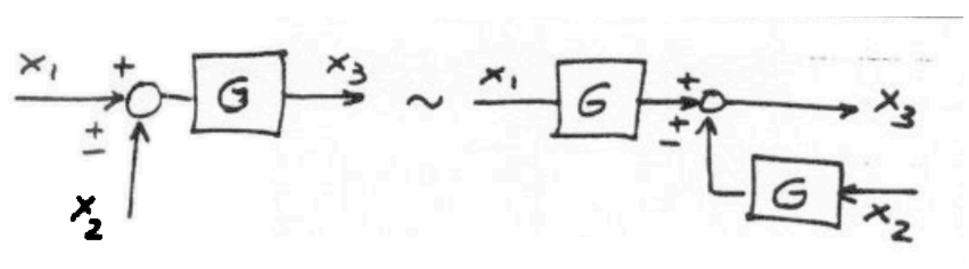


Basic Operations on Systems (Block Diagrams)

Here we are going to learn how to operate on block diagrams. This will allow us to obtain simpler (hopefully) diagram



2) Moving a summing junction behind a block



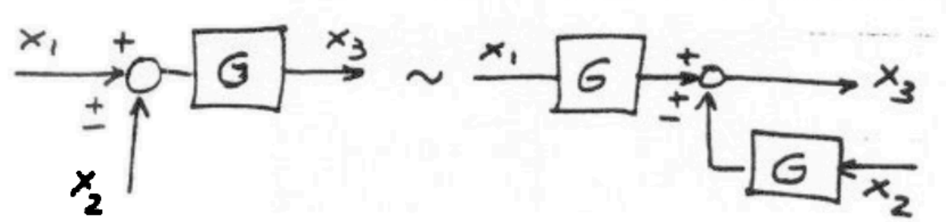
Basic Operations on Systems (Block Diagrams)

Here we are going to learn how to operate on block diagrams. This will allow us to obtain simpler (hopefully) diagram

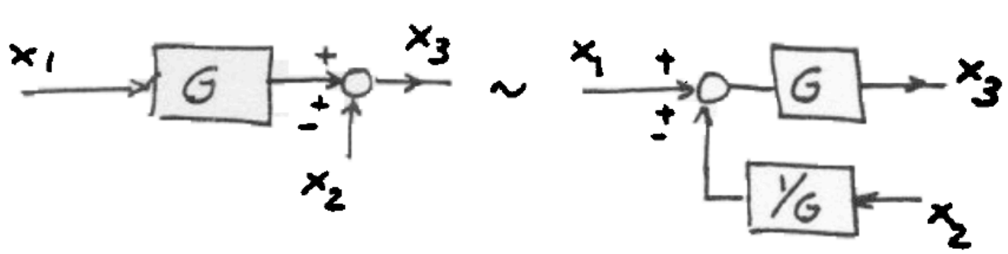
1)

(cascade)

2) Moving a summing junction behind a block



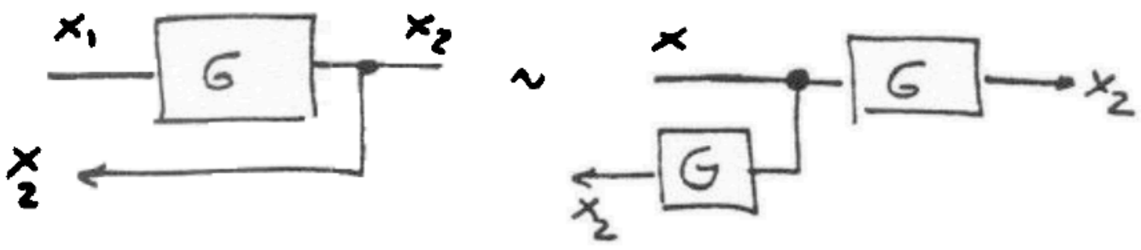
3) Moving a summing junction ahead of a block



Basic Operations on Systems (Block Diagrams)

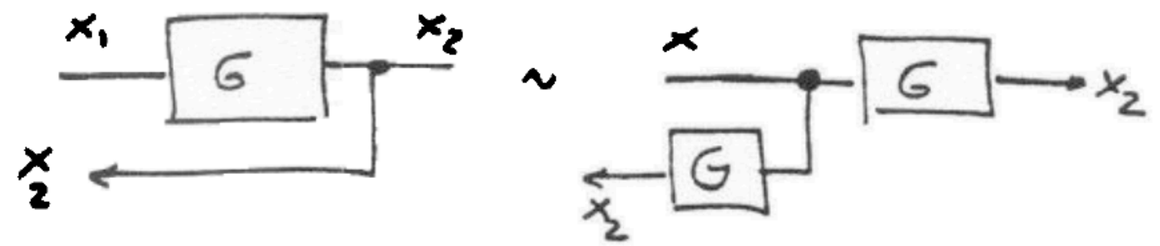
Basic Operations on Systems (Block Diagrams)

4) Moving a "tap" ahead of a block

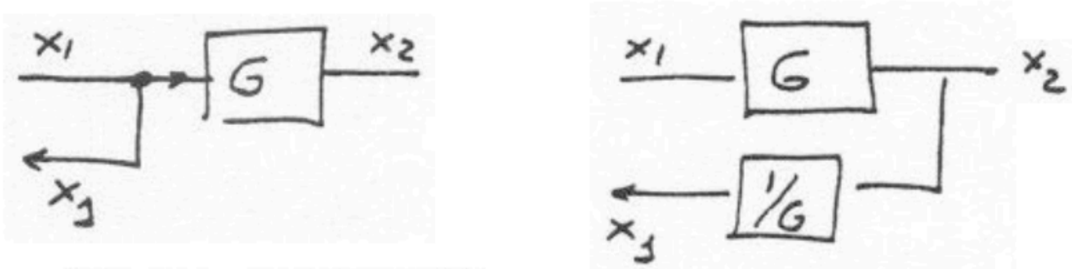


Basic Operations on Systems (Block Diagrams)

4) Moving a "tap" ahead of a block

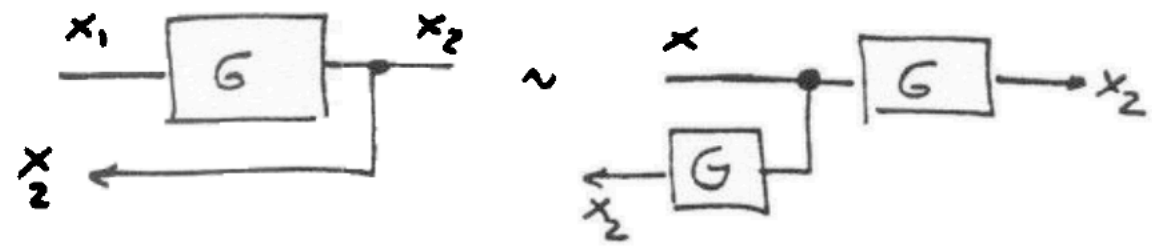


5) Moving a "tap" behind a block

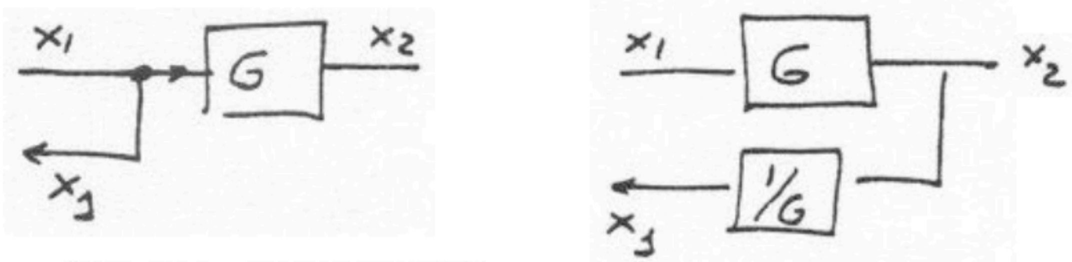


Basic Operations on Systems (Block Diagrams)

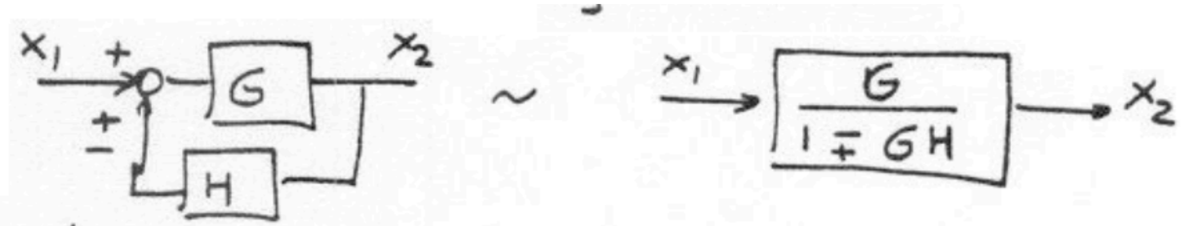
4) Moving a "tap" ahead of a block



5) Moving a "tap" behind a block



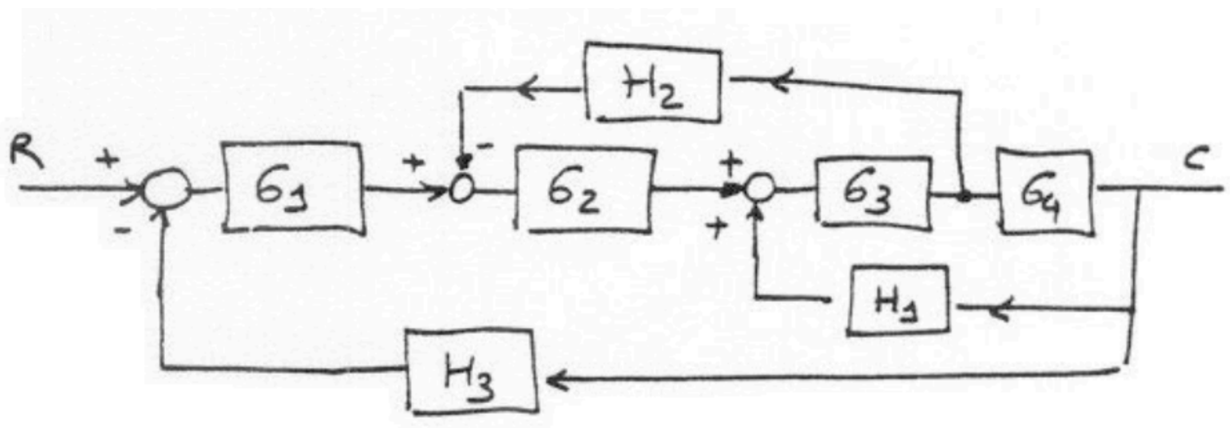
6) Eliminating a feedback loop:



(This follows from Mason's formula)

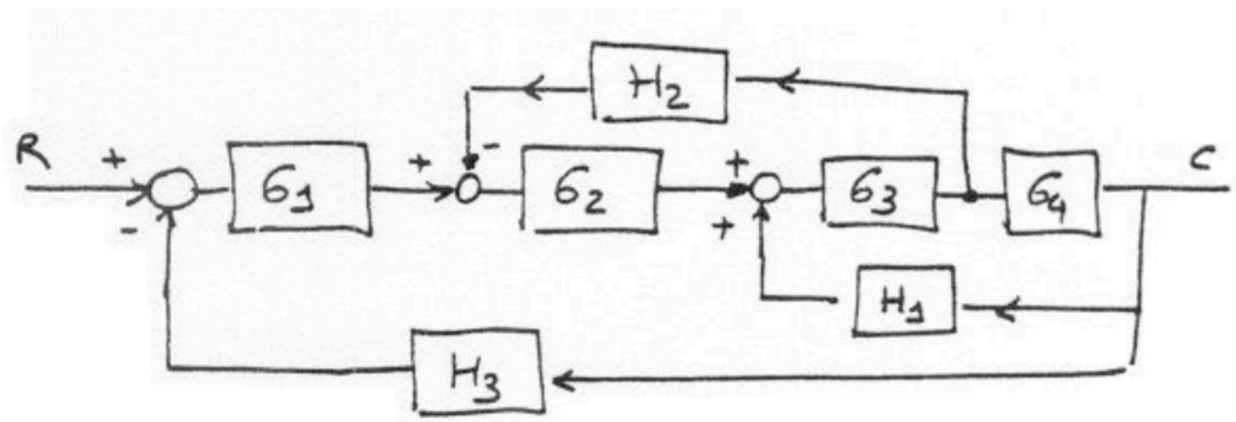
Example

Want to find T_{RC}

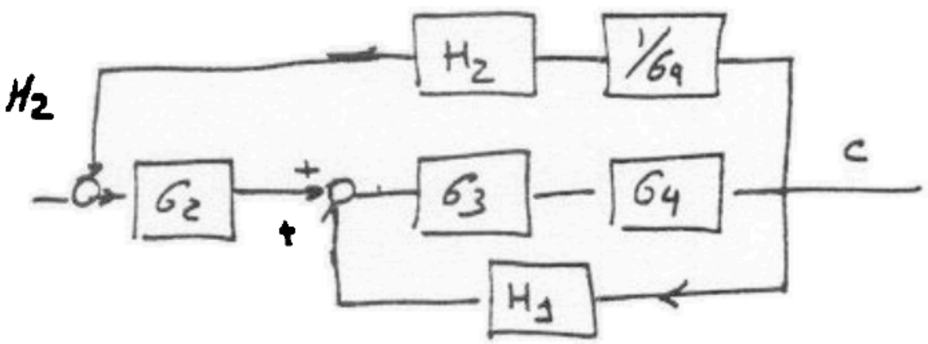


Example

Want to find T_{RC}

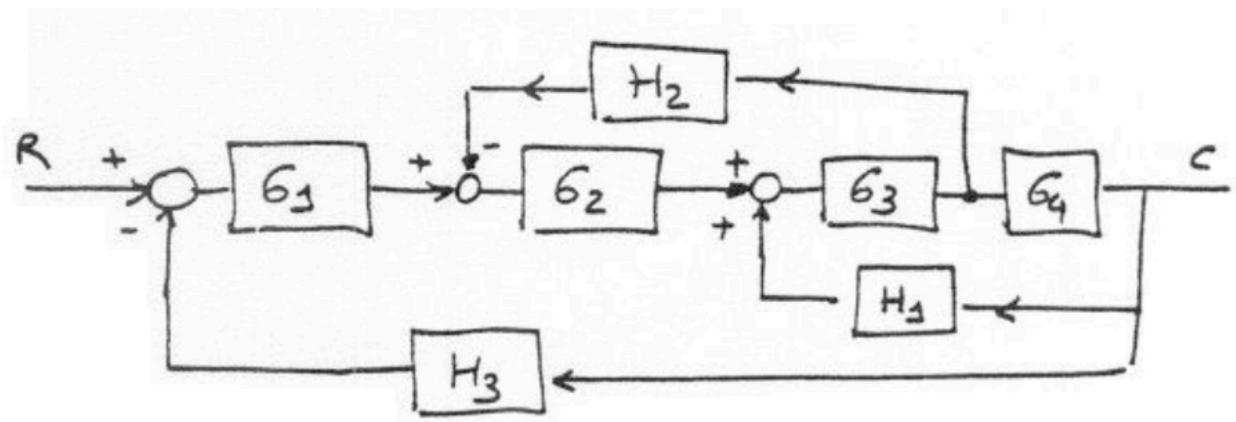


- Step 1: Move the tap for H_2 behind G_4

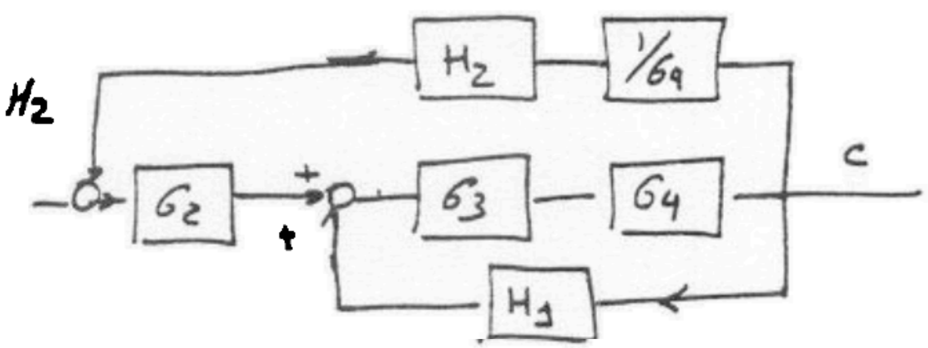


Example

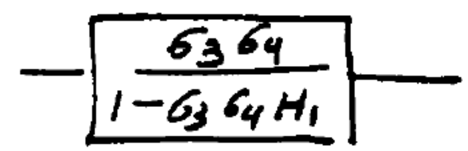
Want to find T_{RC}



- Step 1: Move the tap for H_2 behind G_4

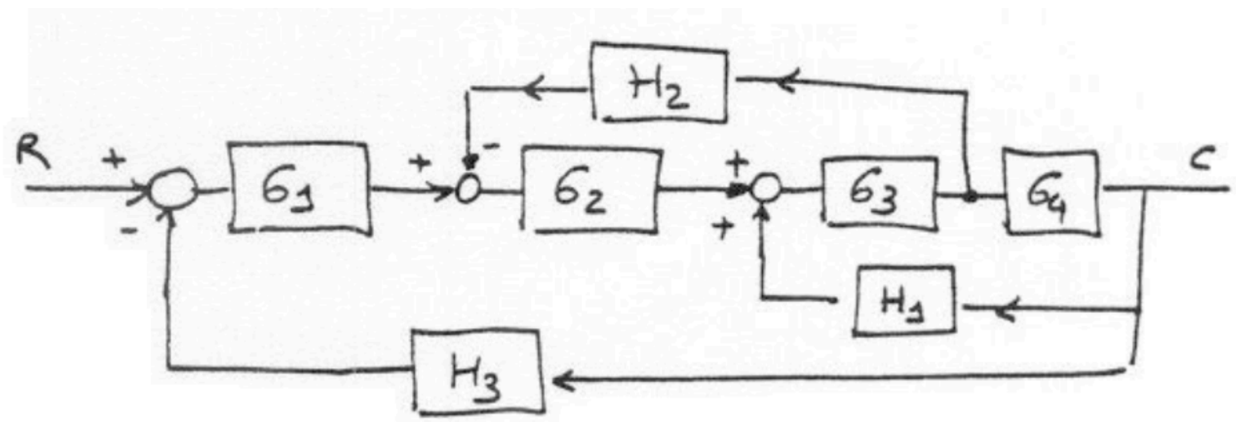


- Step 2: Eliminate the feedback loop $G_3 G_4 H_1$

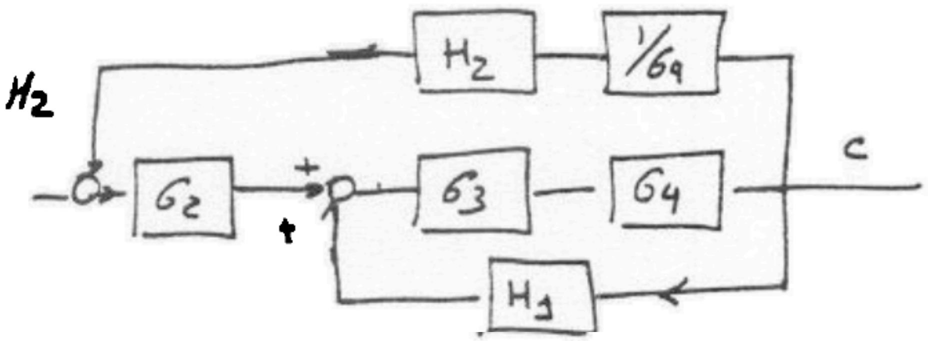


Example

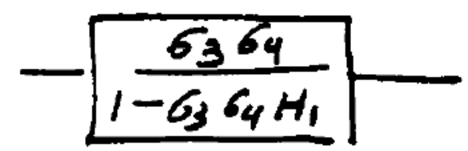
Want to find T_{RC}



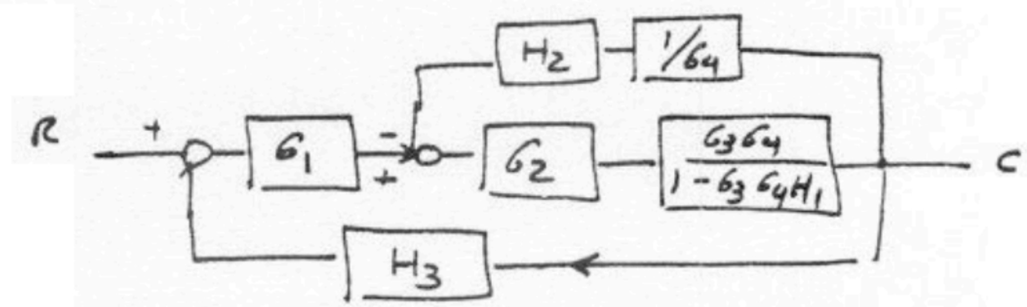
- Step 1: Move the tap for H_2 behind G_4



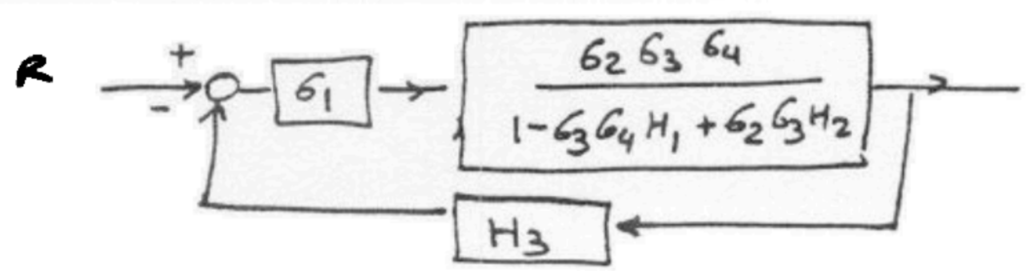
- Step 2: Eliminate the feedback loop $G_3 G_4 H_1$



Now we have:

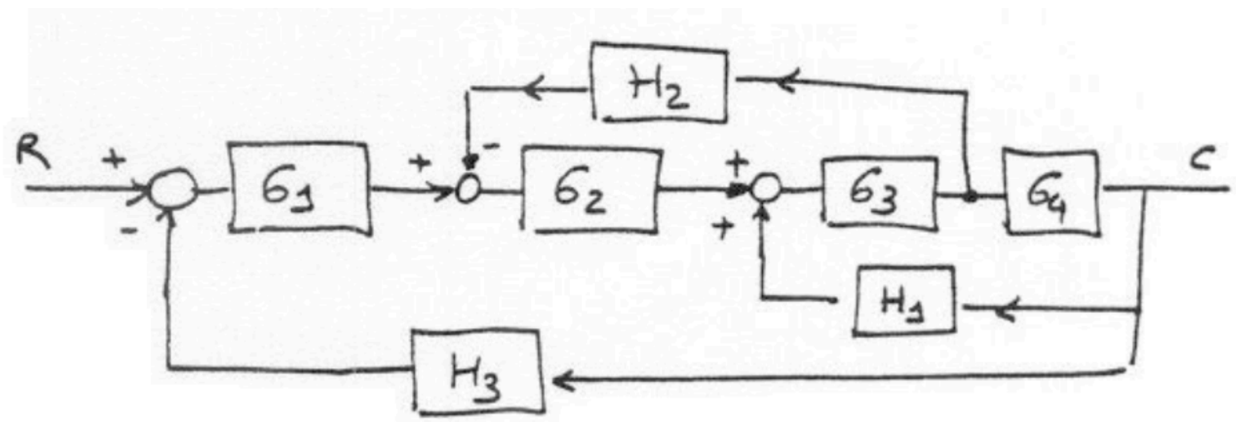


- Step 3: Eliminate the inner loop:

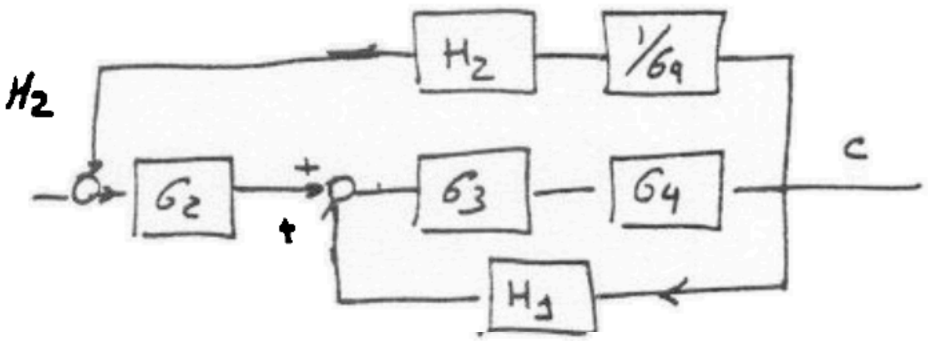


Example

Want to find T_{RC}



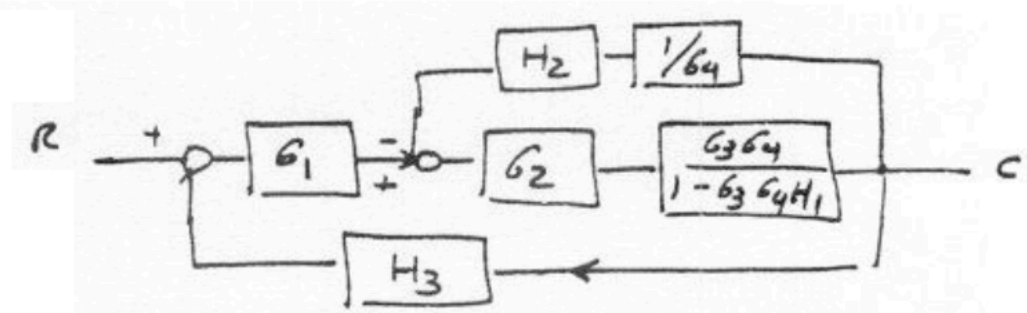
- Step 1: Move the tap for H_2 behind G_4



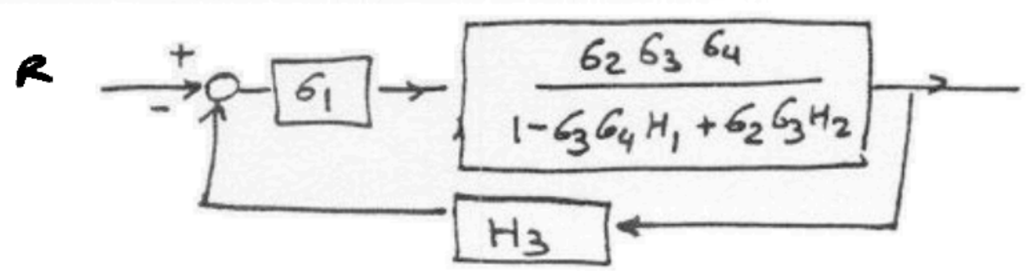
- Step 2: Eliminate the feedback loop $G_3 G_4 H_1$

$$\boxed{\frac{G_3G_4}{1-G_3G_4H_1}}$$

Now we have:



- Step 3: Eliminate the inner loop:

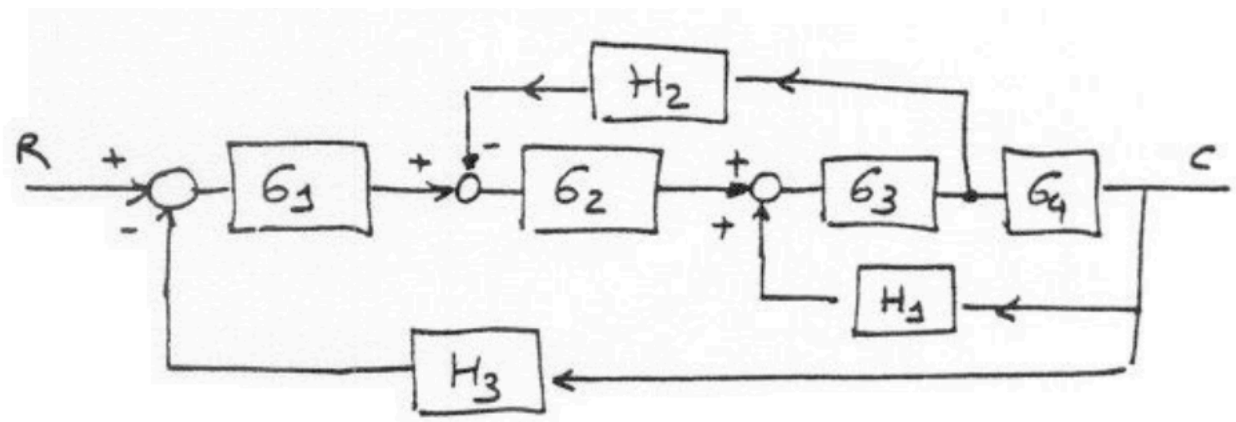


- Step 4: Collapse the final loop (i.e use Mason's again)

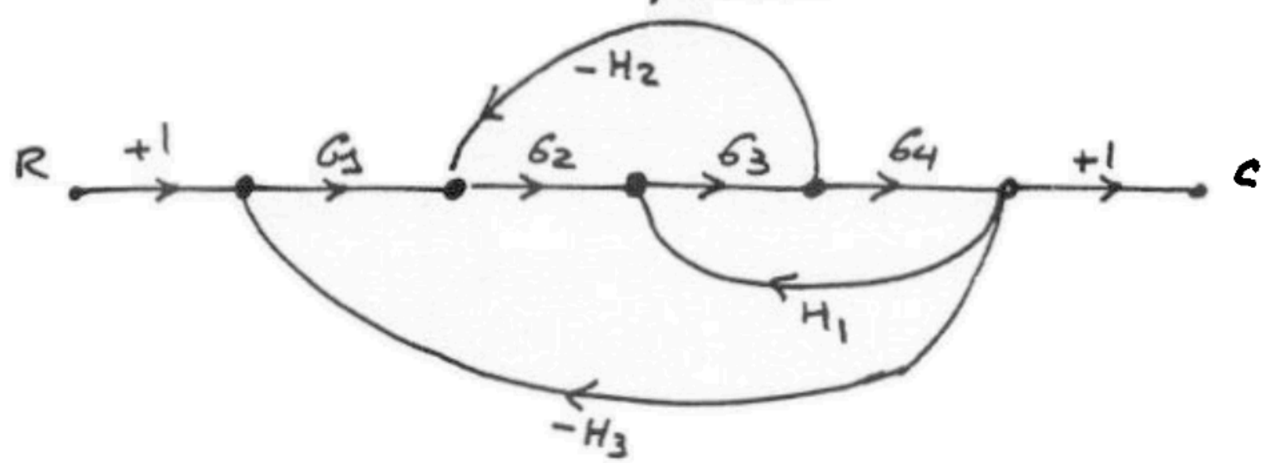
$$R \longrightarrow \boxed{\frac{G_1G_2G_3G_4}{1-G_3G_4H_1+G_2G_3H_2+G_2G_3G_4G_1H_3}} \longrightarrow C$$

Example

Want to find T_{RC}



Alternative solution: Transform to a signal flow graph and use Mason's



Loops: $-G_2 G_3 H_2$
$G_3 G_4 H_1$
$-G_1 G_2 G_3 G_4 H_3$

(all touching)

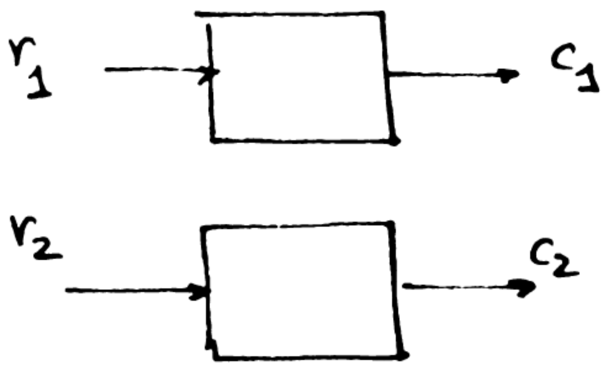
$$\Rightarrow \frac{M}{\Delta} = \cdots G_2 G_3 H_2 - G_3 G_4 H_1 + G_1 G_2 G_3 G_4 H_3$$

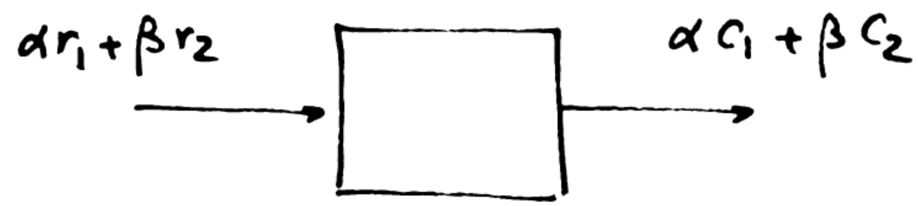
Forward path (only one) $M = G_1 G_2 G_3 G_4$

Discrete Time Systems and z-transform

Discrete Time Systems and z-transform

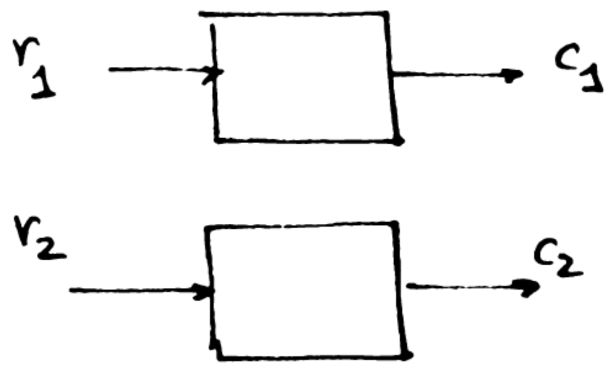
Linear System: Satisfies the principle of superposition

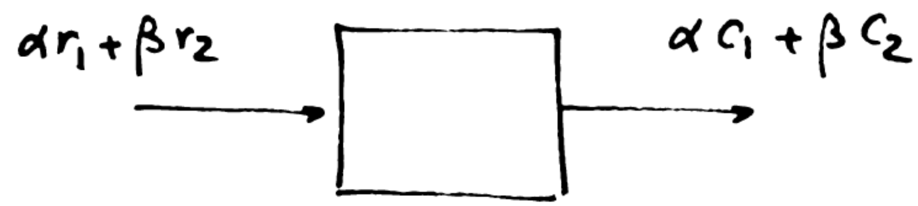




Discrete Time Systems and z-transform

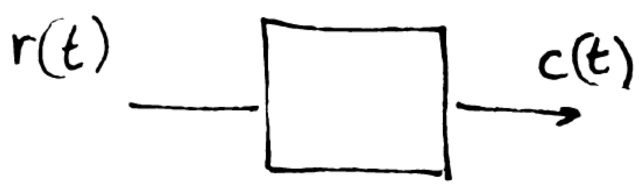
<u>Linear System:</u> Satisfies the principle of superposition

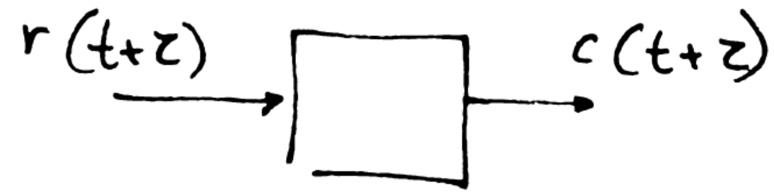




<u>Linear Time Invariant</u> (or Linear Shift Invariant)

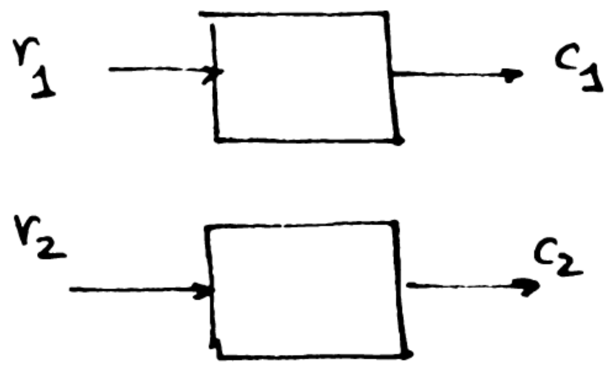
Input/Output relationship is independent of time

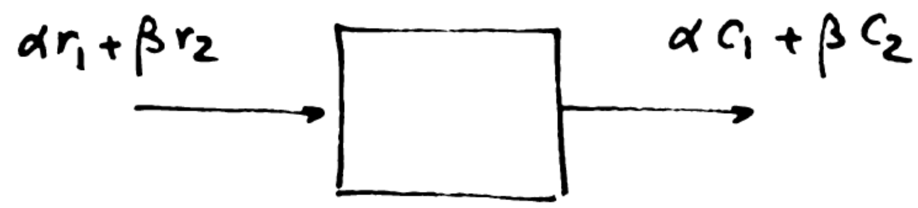




Discrete Time Systems and z-transform

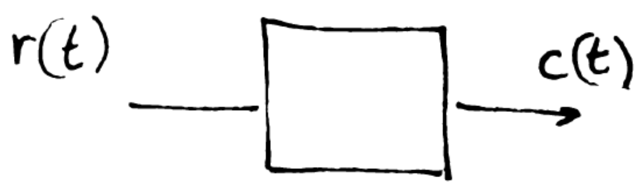
<u>Linear System:</u> Satisfies the principle of superposition

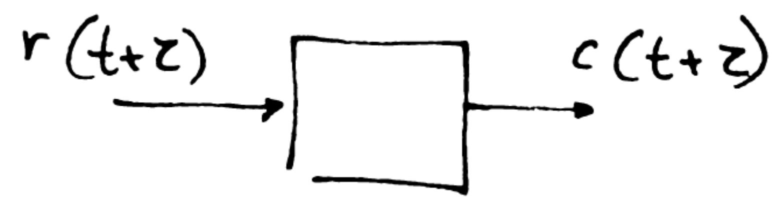




<u>Linear Time Invariant</u> (or Linear Shift Invariant)

Input/Output relationship is independent of time





For our purposes: LTI discrete time system is described by a set of <u>difference</u> equations with constant coefficients.

$$x(k) + a_1 x(k-1) + \cdots a_{n-1} x(k-n+1) + a_n x(k-n) = b_0 e(k) + b_1 e(k-1) + \cdots b_m e(k-m)$$

Discrete Time Systems and z-transform

Discrete Time Systems and z-transform

<u>Assumption</u>: (standing from now on) All the numbers arrive to the digital controller with the same fixed period T (the sampling period), and at the same time (single rate, synchronous sampling)

Discrete Time Systems and z-transform

<u>Assumption</u>: (standing from now on) All the numbers arrive to the digital controller with the same fixed period T (the sampling period), and at the same time (single rate, synchronous sampling)

Later we will see how to remove this assumption $\Rightarrow$ nonsynchronous sampling, multirate sampling

z-Transform

Z transform: (Reference: Chapter 2 text, Chapter 10, Oppenheim & Willsky)

Given a sequence: $e_{-k} \cdots e_{-1}\ e_0\ e_1 \cdots e_k$

we define its bilateral z transform by:

$$Z(e_k) = \sum_{-\infty}^{+\infty} e_k z^{-k}$$

where z is a complex variable. In 5610 we are going to use only single-sided (or unilateral) z-transforms:

$$Z(e_k) = \sum_{0}^{\infty} e_k z^{-k} = e_0 + \frac{e_1}{z} + \frac{e_2}{z^2} + \cdots +$$

(this amounts to setting $e_k = 0$ for $k < 0$)

z-Transform

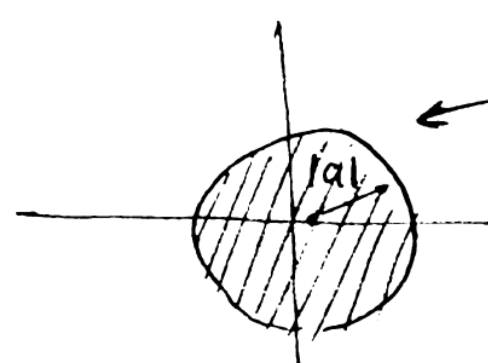
- Notation: $e_k \overset{Z}{\longleftrightarrow} E(z)$

- Note that we have an infinite series. Thus $E(z)$ is likely to be defined (i.e. finite) only for some regions of the complex plane.

Example: $e_k = a^k$

$$E(z) = 1 + \frac{a}{z} + \frac{a^2}{z^2} + \cdots + = \begin{cases} \dfrac{1}{1-\left(\frac{a}{z}\right)} & \text{if } \left|\frac{a}{z}\right| < 1 \\ \infty & \text{otherwise} \end{cases}$$

In this example the z-transform is well defined only in the region $\left|\frac{a}{z}\right| < 1 \Leftrightarrow |z| > |a|$, i.e. the exterior of a disk with radius $|a|$

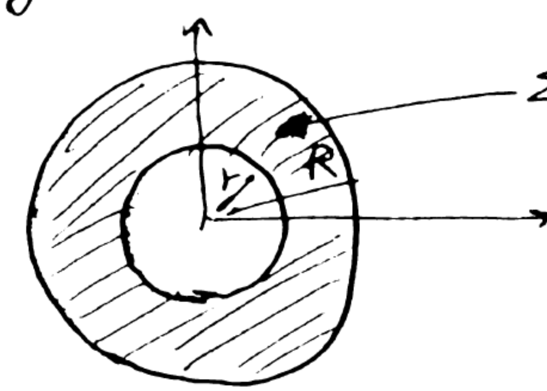


z transform defined here

• **Definition**: Region of Convergence: (ROC) The region in the z-plane where the ∞ sum $\sum_{-\infty}^{+\infty} e_k z^{-k}$ converges

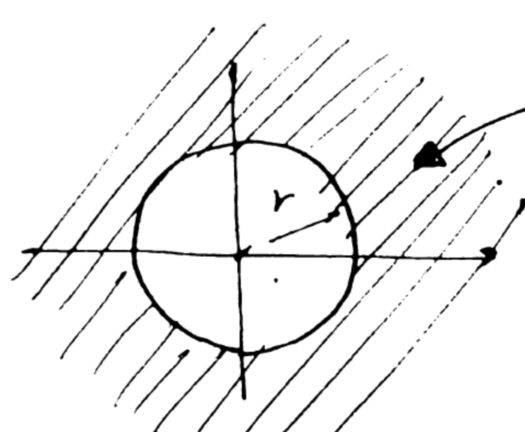
Facts

1) The ROC for general double-sided z-transforms is a ring centered about the origin: $|r| < |z| < |R|$



$\Sigma e_k z^{-k}$ well defined here

2) The ROC for unilateral z-transforms (the only ones we'll use in 429) is the exterior of a disk (i.e $|z| > r$)



z-transform well defined here

Q: Why do we care about ROC?

A: (a) we will need it in order to compute <u>inverse</u> z transforms

(b) It is related to the location of the <u>poles</u> of the T.F and to the concepts of <u>stability</u> and causality.

Q: Why do we care about ROC?

A: (a) we will need it in order to compute <u>inverse</u> z transforms

(b) It is related to the location of the <u>poles</u> of the T.F and to the concepts of <u>stability</u> and causality.

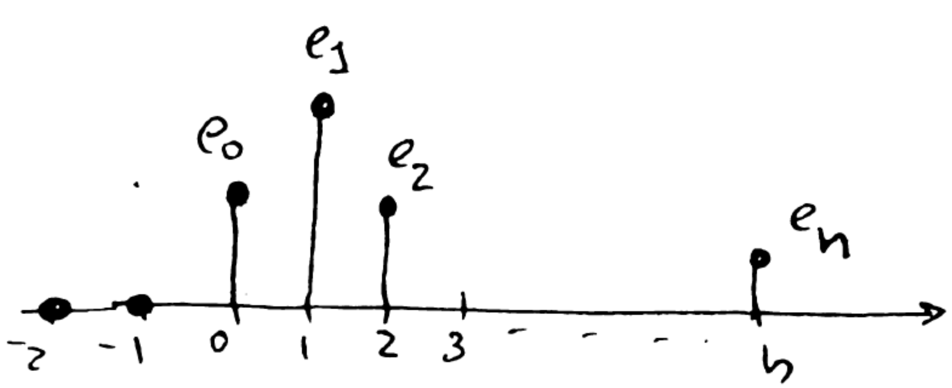
Properties of the z-transform

1) <u>Linearity</u> : $\mathcal{Z}\left[\alpha e_1(k) + \beta e_2(k)\right] = \alpha \mathcal{Z}(e_1) + \beta \mathcal{Z}(e_2)$

Proof: follows immediately from the definition

2) <u>Time shift</u>

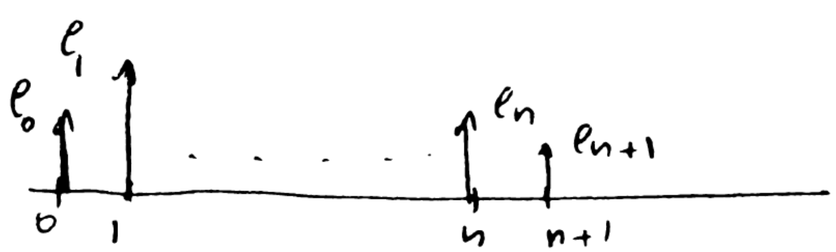
Assume that $e_k = 0$ for $k < 0$



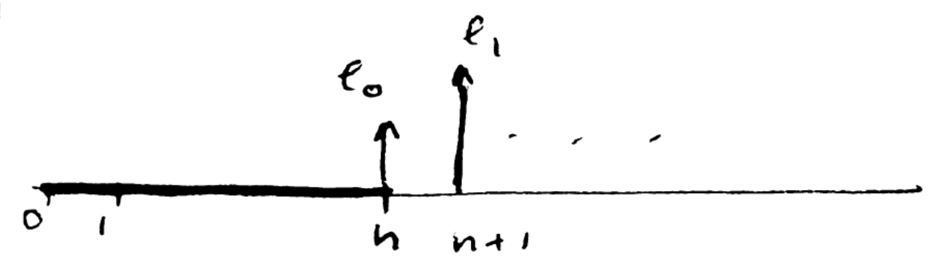
We will consider 2 cases: (a) <u>Time delay</u>

(b) <u>Time advance</u>

(u) Time delay: (similar to integration in continuous time)



delay n periods $\Longrightarrow$



$\Downarrow Z$ $\qquad\qquad$ $\Downarrow Z$

$E(z)$

$$Z\left[e(k-n)\right] = \sum_{0}^{\infty} e(k-n)z^{-k} = \underbrace{e_{-n} + e_{-n+1}z^{-1} + \cdots e_{-1}z^{-(n+1)}}_{0} + e_0 z^{-n} + e_1 z^{-(n+1)} + \cdots +$$

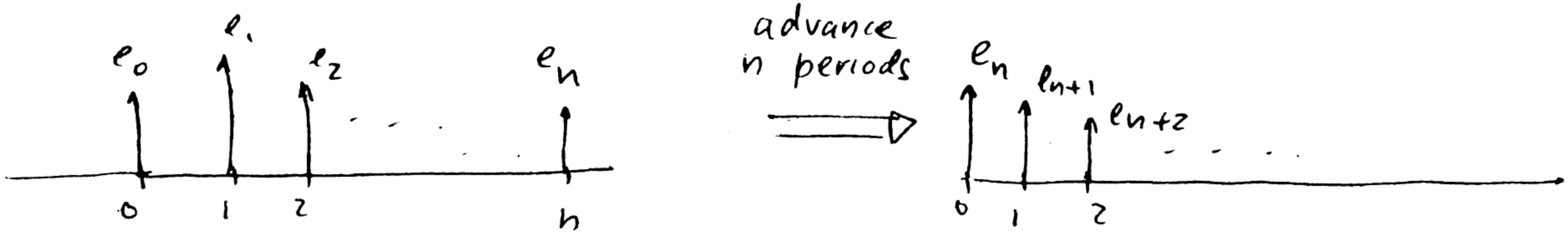
(all $e_i = 0$, $i < 0$)

$$= z^{-n}\Big[\underbrace{e_0 + e_1 z^{-1} + \cdots}_{E(z)}\Big] = z^{-n}E(z)$$

$$\Rightarrow \boxed{Z\left[e(k-n)\right] = \frac{1}{z^n} Z\left[e_k\right]}$$

Time delay of n periods $\Longleftrightarrow$ multiply $E(z)$ times $\frac{1}{z^n}$

(b) <u>Time advance</u>: (similar to differentiation in continuos time)



Advance 1 period: $Z\left[e(k+1)\right] = e_1 + e_2 z^{-1} + \cdots = z\left(e_1 z^{-1} + e_2 z^{-2} + \cdots\right)$

$$= z\left[-e_0 + e_1 + e_1 z^{-1} + e_2 z^{-2} + \cdots\right] = \boxed{z\left[E(z) - e_0\right]}$$

Similarly:

$$\boxed{Z\left[e(k+n)\right] = z^n\left[E(z) - \sum_{0}^{n-1} e_k z^{-k}\right]}$$

Note that (contrary to the delay case) here $Z\left[e(k+n)\right]$ cannot be expressed solely in terms of $E(z)$. Why the difference? Because now some of the values of $\{e_k\}$ have been lost and we need to account for these.

<u>Example</u>: $\quad Z(a^k) = \dfrac{z}{z-a}$.

$$Z(a^{k+1}) = Z(a\cdot a^k) \overset{\text{linearity}}{=} z\,\frac{a}{z-a} = z\left[\frac{z}{z-a} - 1\right] \quad \#$$

$$Z\left[a^{(k-1)}u(k-1)\right] = 0 + \frac{1}{z} + \frac{a}{z^2} + \cdots = \frac{1}{z-a} \quad \#$$

- **Scaling in z-plane:**

$$\mathcal{Z}\{(r^{-k}e_k)\} = E(rz)$$

Proof: $\mathcal{Z}\{(r^{-k}e_k)\} = \sum_0^{\infty}(e_r\, r^{-k})z^{-k} = \sum_0^{\infty} e_k\,(rz)^{-k} = E(rz)$ #

- **Scaling in z-plane:**

$$Z\{(r^{-k} e_k)\} = E(rz)$$

Proof: $$Z\{(r^{-k} e_k)\} = \sum_0^{\infty} (e_r r^{-k}) z^{-k} = \sum_0^{\infty} e_k (rz)^{-k} = E(rz) \quad \#$$

- **Two important Theorems:** (Initial and Final value theorems)

They allow for the calculation of limits and steady state values without having to actually find the transform

- **Final value theorem**:

$$\boxed{\lim_{k\to\infty} e(k) = \lim_{z\to 1} (z-1)\,E(z)}$$

(Physical significance: the behavior of $e(k)$ as $k\to\infty$ is related to the low frequency ($s=0 \leftrightarrow z=1$) components of $E(z)$)