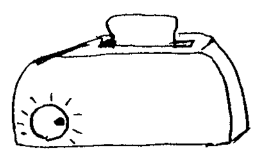
ECE 5610 DIGITAL CONTROL SYSTEMS

- **Control System:** A system designed to cause the output of a dynamic system ("the plant") to achieve a desirable behavior by manipulating some inputs (the "control" inputs)

  - **Example 1:** An automatic landing system that manipulates the elevator angle and the throttle to keep the airplane in a desired flight path

- **Example 2:** A Toaster:

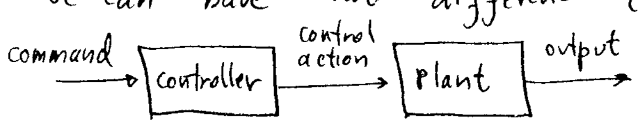


Here the controller (a timer) decides the amount of time that the toaster stays on, based upon the "darkness" setting.

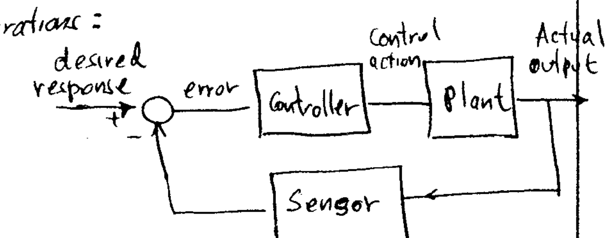
- **Elements of a control system:**
  - (1) **Plant:** The system or process to be controlled (i.e. airplane)
  - (2) **Sensors:** To measure the "behavior" of the plant (i.e deviation from the desired flight path)
  - (3) **Controller:** Decides the values of the inputs to the plant based upon the command.

- Question: Do we always need these three components?
- Answer: **No**: Obviously we need (1) and (3), but (2) may be missing.

We can have two different configurations:

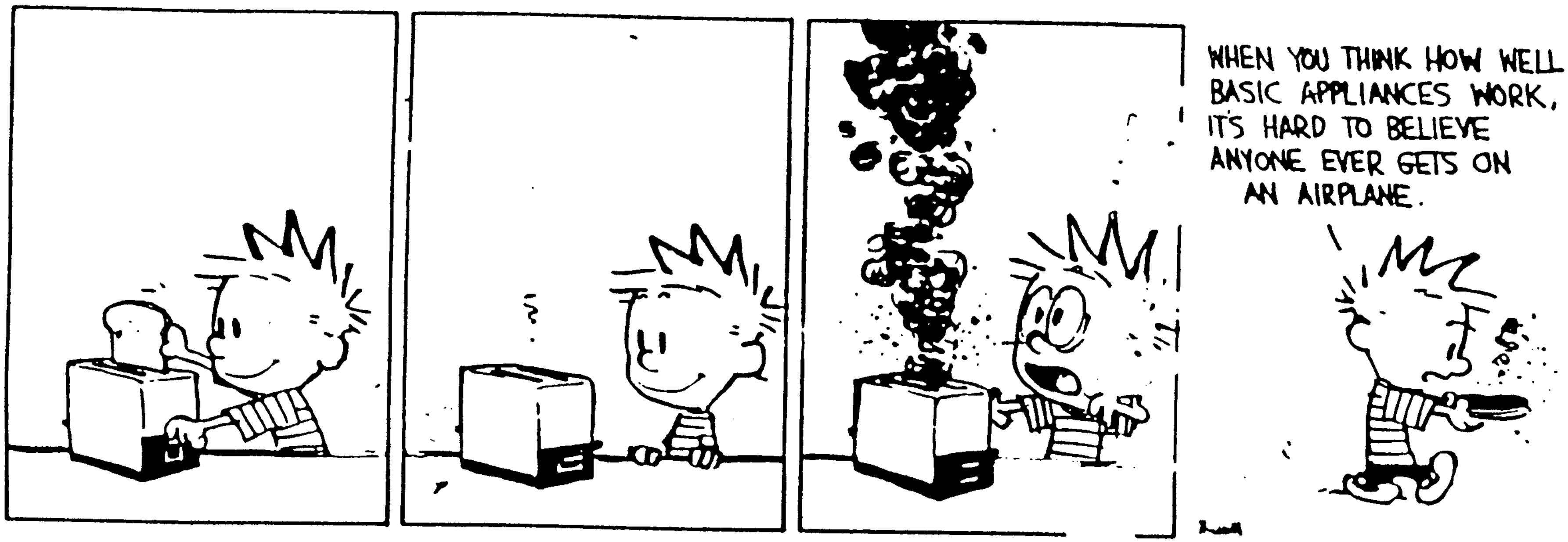


"Open loop" control
(no need for sensors)



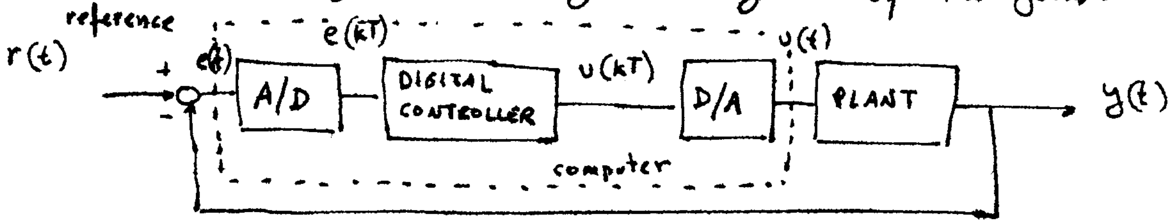
"Closed-loop" control

- <u>Why feedback?</u>

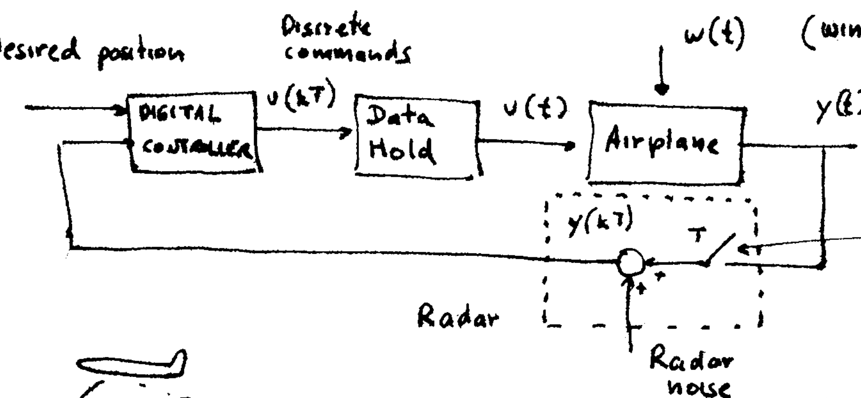


Introduction to Digital Control Systems

This course is concerned with the analysis and design of control systems that incorporate digital elements (usually a computer or microprocessor) in the feedback loop. These systems involve both continuous and discrete time signals, leading to diagrams of the form:



Example 1: An automatic landing system: goal: keep the airplane centered on the runway (lateral control system) an following a pre-specified trajectory

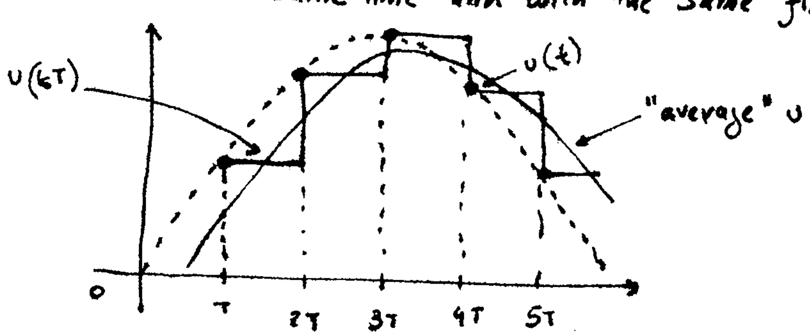


Sampler: device that samples the continuous input and generates a train of pulses

Radar measures the position $y(t)$ every T seconds
Data holds clamps the value of $u(kT)$ until the next command is received

Note that this is intrinsically a sampled-data system since the radar is pulsed every T seconds.

Assumption: All the numbers arrive at the digital controller at the same time and with the same fixed period T (the sampling period.)

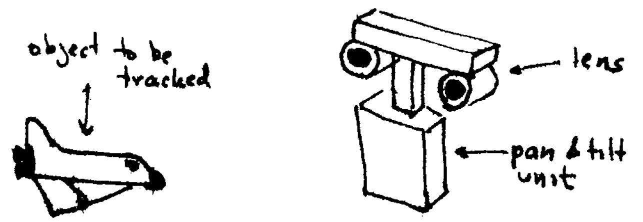


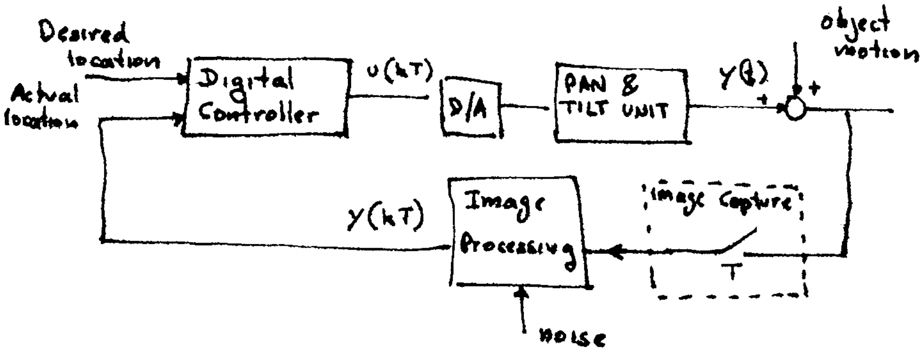
Note that the hold introduces a delay

(more on this later)

Example 2: **Active vision:** Close the loop using computer vision as a sensor

Applications: remote medicine
inspection
MEMS manufacturing
⋮
Automated spacecraft docking





As in the previous example, the system is intrinsically sampled-data, composed by the interconnection of digital elements (image processing, controller) and an analog plant (the robotic head).

As in the previous example, the digital processing introduces a substantial delay that creates stability (and performance) difficulties.

- **Why digital control ?**

   - Increased flexibility — analog systems are changed by rewiring
     computer controlled systems can be <u>reprogrammed</u>
     Also, you can add decision making capabilities to the loop

   - Economic reasons: Initial cost of a computer based controller may be high (although not usually the case now) but the cost of adding more loops is low. Moreover you get more compact & light weight controllers.

   - Better characteristics: No drift, better reliability, better noise rejection characteristics.

   - Many systems are inherently sampled: The two examples above, internal combustion engines, financial systems, etc

Historical Background:

1955: TRW and Texaco implemented a digital control system that determined the set-point for analog controllers

Technology: vacuum tubes
Mean Time Between Failures (MTBF) 50 - 100 hours
Time for ⊕: 1 ms
Time for ⊗: 20 ms

1962: Imperial Chemical Industries (U.K.) completely replaced analog controllers by a digital computer at a chemical process plant. The system regulated over 200 variables.

Technology: transistors
MTBF: 1000 hours (key improvement) (up by 1 order of magnitude)
Time for ⊕: 100 μs.
Time for ⊗: 1 ms
} down by 1 order of magnitude

1967: DEC (Digital Equipment Corp) introduces the PDP-8 computer. MTBF increases to 20 000 hours

1970: Estimated number of computer-controlled systems used in industry: 5.000

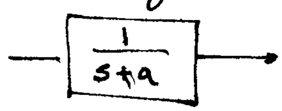
1975: Estimated number of computer-controlled systems in industry: 50,000

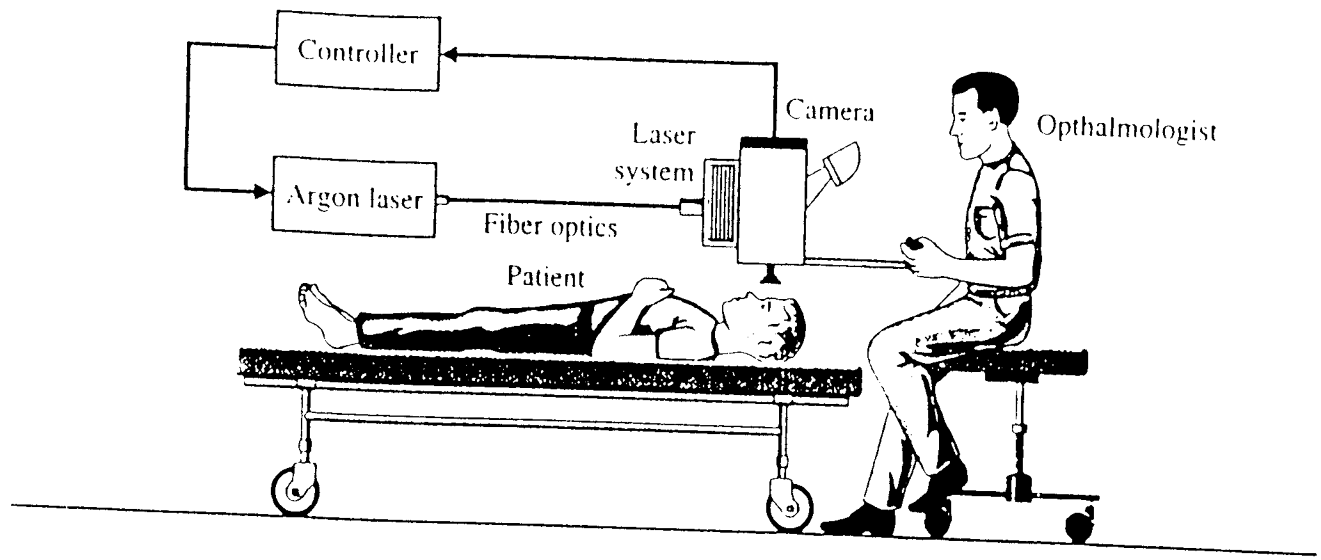
1998: Digital control systems are used everywhere, from automobiles (cruise control, fuel injection, ABS) to aircrafts (Airbus, MD-11, Boeing 757, 767, 777 are all "fly-by-wire")

- **Question:** OK, so we should implement the controllers using a computer (maybe), but why 5610? Can't we just use what we learned in 5580?

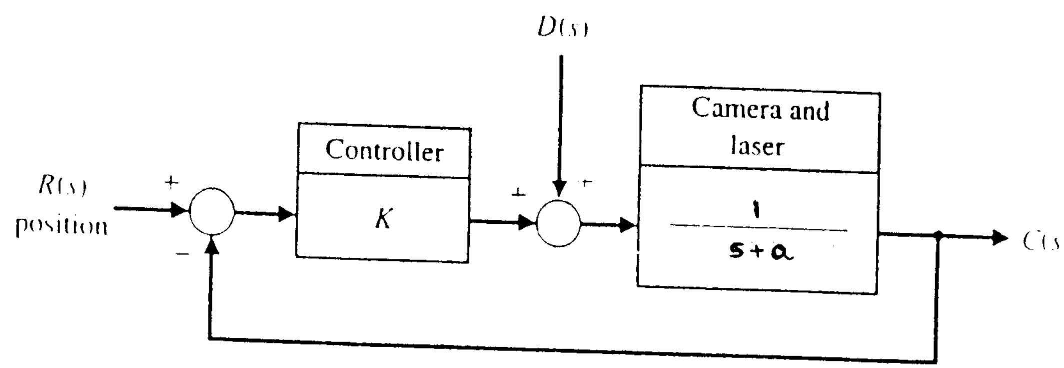
- **Answer:** Maybe. Let's try an example:

Design a controller to position the beam in a laser surgery system. For the sake of simplicity, model the dynamics as a first order system





(a)



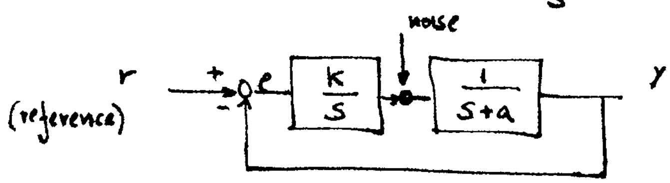
(b)

FIGURE DP4.4
Laser eye surgery system.

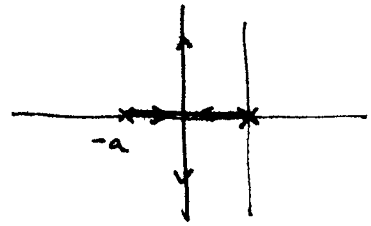
<u>Specs</u>: zero steady state error to a step command
"reasonable" settling time & overshoot

we know that in order to meet the specs we need to turn the system into a <u>Type I</u> $\Rightarrow$ use integral control (PI or PID)

Let's use PI, i.e. $C(s) = \frac{k}{s}$

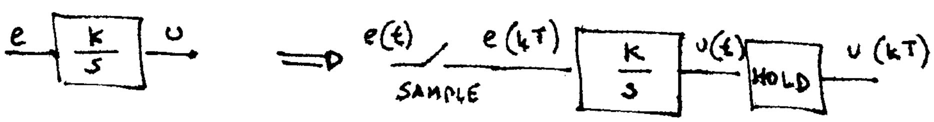


- Q: How do we select k ?
- A: Try root locus:

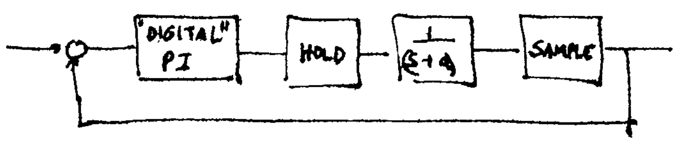


$\Rightarrow$ Stable for all k.
Critical damping yields fastest system without overshoot.

Suppose that now we want to go ahead an use a <u>digital</u> implementation of the PI controller:



$$u(t) = u((k-1)T) + k\int_{(k-1)T}^{t} e(kT)\,dt \Rightarrow u(kT) = kT\,e[(k-1)T] + u[(k-1)T]$$



Let's try a simulation:

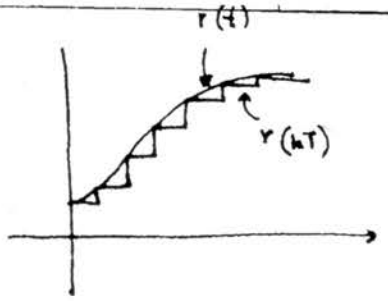
Works well for small values of T, however as we increase T performance degrades and then becomes unstable
(tough luck if <u>you</u> are the patient)

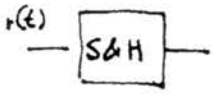
Moreover, the larger is T, the worse is the disturbance rejection.

- Q: What went wrong here
- A: With 20/20 hindsight: these factors may be related to the time delay introduced by the sampling and holding

Let's pursue this further:

Sample and hold action:





Recall (from 3464) that the transfer function of the S&H is given by:

$$G(s) = \frac{1 - e^{-sT}}{s}$$ (this comes from the impulse response)

$$\Rightarrow \quad G(j\omega) = \frac{1 - e^{-j\omega T}}{j\omega}$$

If ωT is small then

$$\frac{1 - e^{-j\omega T}}{j\omega} \cong \frac{1 - \left(1 - j\omega T + \frac{(j\omega T)^2}{2} + \cdots\right)}{j\omega} \cong T\left(1 - j\frac{\omega T}{2}\right)$$

$$\cong T e^{-j\frac{\omega T}{2}}$$ (a delay of $\frac{T}{2}$)

From 5580 we know that the effect of this delay is to <u>add</u> phase to the system (rotate the Nyquist plot clockwise) ⇒ if this additional phase exceeds the phase margin ⇒ we go <u>unstable</u>

In any case, the effect of the sampling and holding action is to <u>decrease</u> the phase margin (the digital system is <u>less</u> stable than its analog counterpart)

⇒ There is a trade-off between sampling rate and stability, regulator effectiveness and robustness (sensitivity) to parameter variations

$T \Uparrow$ $\Longrightarrow$	phase margin regulation effectiveness robustness	$\Big\}\ \Downarrow$

Other issue that needs to be considered is the effect of <u>quantizing</u> the signals in the loop (due to finite word length)

Recap: Digital control systems are:

- ⊕ More flexible
- ⊕ Cheaper
- ⊕ More reliable (high MTBF, low drift)

} than analog counterparts

However:

- ⊖ Less stable
- ⊖ Less effective
- ⊖ Less robust

} Unless T is chosen properly and adequate design techniques are used

(rather than "lets discretize an analog design" approach)

- **Some models that we will use later on:**
  - **Mechanical Translational Systems:**

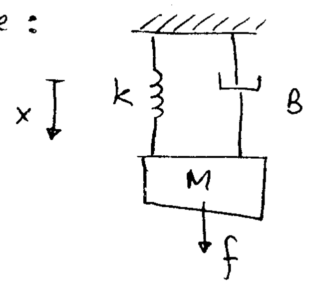
Basic Law: Newton's second law: $M\ddot{x} = \sum F$

Elements:

a) Spring $\quad F_k = -kx$

b) Viscous damping and friction $\quad F_b = -b\dot{x}$ $\quad$ (always opposes motion)

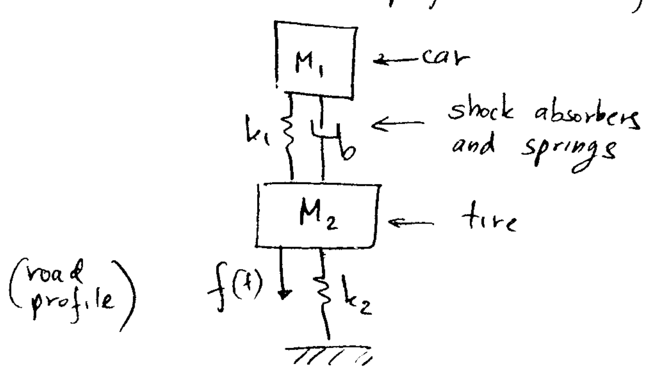
- Example:



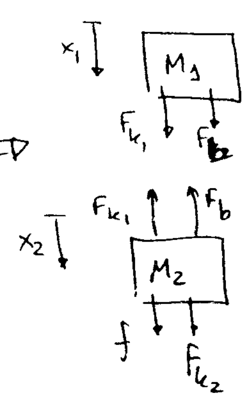
$$M\ddot{x} = -kx - b\dot{x}$$

$$\Rightarrow \boxed{M\ddot{x} + b\dot{x} + kx = 0}$$

- Example 2: Simplified model of an automobile suspension:



$\Longleftrightarrow$



$$M_1\ddot{x}_1 = -k_1(x_1 - x_2) - b(\dot{x}_1 - \dot{x}_2)$$

$$M_2\ddot{x}_2 = -k_1(x_2 - x_1) - b(\dot{x}_2 - \dot{x}_1) - k_2 x_2 + f$$

Taking Laplace transforms yields:

$$(M_1 s^2 + b_1 s + k_1)x_1 - (bs + k_1)x_2 = 0$$

$$-(bs + k_2)x_2 + (M_2 s^2 + b_1 s + k_1 + k_2)x_2 = F(s)$$

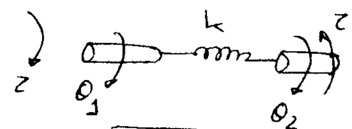
By solving these we can get the transfer functions $G_1(s) = \frac{X_1(s)}{F(s)}$ and $G_2(s) = \frac{X_2(s)}{F(s)}$

- **Mechanical Rotational Systems**

Basic Law: Newton's equation for rotational systems: $J\ddot{\theta} = \Sigma$ Torques

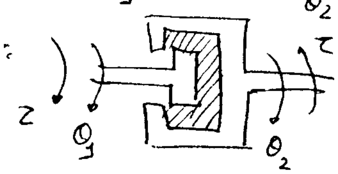
Elements: moment of inertia (similar to mass)
friction
torsion

Torsion spring:



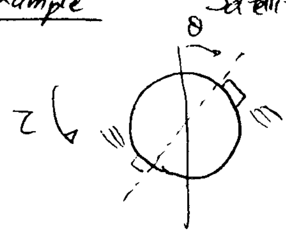
$$\tau = k(\theta_2 - \theta_1)$$

Viscous friction:



$$\tau = B(\dot{\theta}_2 - \dot{\theta}_1)$$

- Example Satellite attitude control: (with torque applied by 2 thrusters)

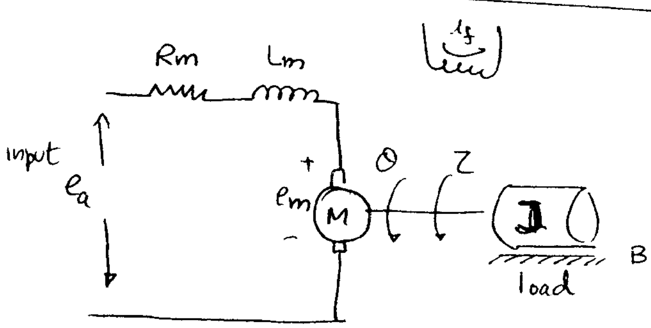


$$J\frac{d^2\theta}{dt^2} = \tau$$

(or, in Laplace domain: $Js^2\theta = \tau \Rightarrow \theta = \frac{1}{Js^2}\tau$: essentially a double integrator)

- **Electromechanical Systems:**

DC motor with independent excitation:



1) Electrical equation: $e_a = R_m i_a + L_m \dfrac{d i_a}{dt} + e_m \Leftrightarrow E_a(s) = (sL_m + R_m) I_a(s) + E_m(s)$

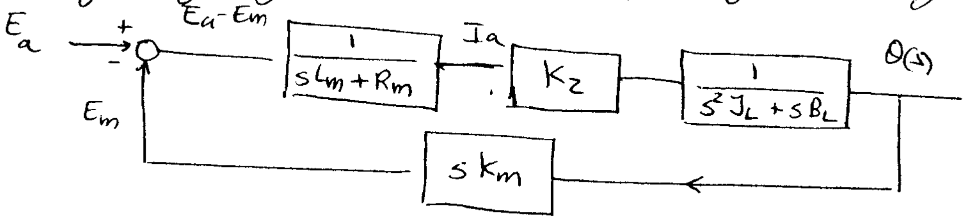
(e_m: back emf)

2) Back emf: $e_m = k_m \dot{\theta} \Leftrightarrow E_m(s) = s k_m \theta(s)$

3) Mechanical equation: $\tau = k_\tau i_a$

4) Newton's second equation: $J_L \dfrac{d\dot{\theta}}{dt^2} + B_L \dot{\theta} = \tau \Leftrightarrow (s^2 J_L + s B_L)\,\theta = \tau(s)$

Putting everything together yields the following block diagram:



<u>Surprise!</u> The system has <u>built-in</u> feedback (through the back emf)

- Q: How do we find the transfer function from $E_a(s)$ to $\theta(s)$?
- A: We could try solving the 4 simultaneous equations (messy) <u>or</u> applying Mason's formula to the loop above. The latter approach yields:

$$G(s) = \frac{G_1(s)}{1 + s k_m G_1(s)} \quad \text{where} \quad G_1(s) = \frac{K_\tau}{(sL_m + R_m)(s^2 J_L + s B_L)}$$

so in principle we get a <u>third order</u> system.

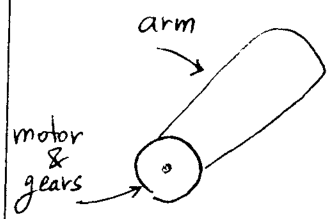
<u>Common simplifying assumption</u>: neglect L_m $(sL_m \simeq 0) \Rightarrow$

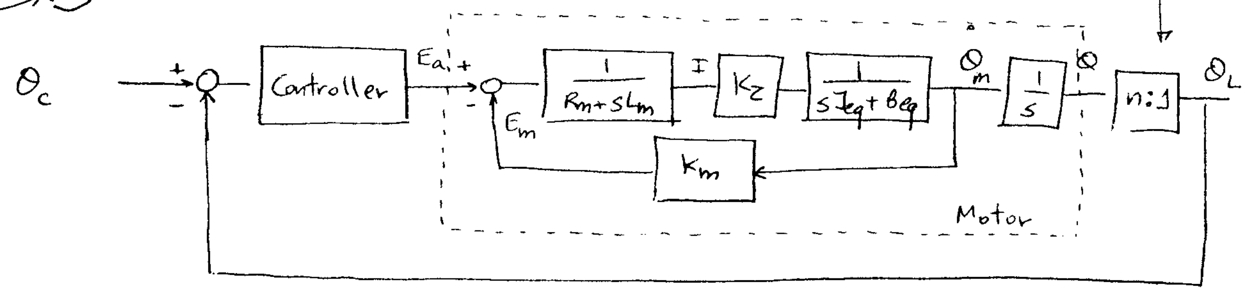
$$G(s) = \frac{\theta(s)}{E_a(s)} = \frac{\dfrac{k_\tau}{R_m s (sJ_L + B_L)}}{1 + \dfrac{k_\tau k_m \cancel{s}}{R_m \cancel{s} (sJ_L + B_L)}} = \frac{K_\tau}{\cancel{R_m} s \cancel{(sJ_L + B_L)}} \cdot \frac{1}{\dfrac{(R_m (sJ_L + B_L) + k_\tau k_m)}{\cancel{R_m} \cancel{(sJ_L + B_L)}}}$$

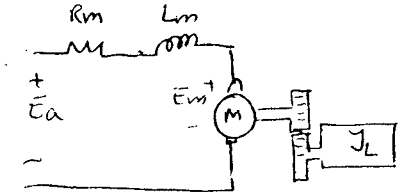
$$G(s) = \left(\frac{K_\tau}{R_m J_L}\right) \cdot \frac{1}{s\left(s + \dfrac{k_\tau k_m + B_L}{J_L R_m}\right)} = \boxed{\frac{K}{s(s+a)}}$$

(looks like the cascade of a pure integrator and a first order lag)

- Example of use: Position control of a single link, single joint, rigid robotic arm. or of the robotic head in room 209, Hayden







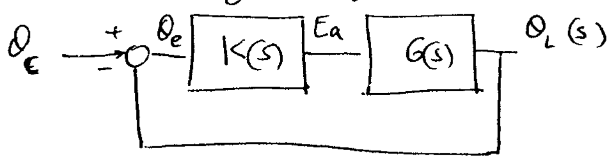
Here J_{eq} = DC motor inertia + (arm. inertia)$\cdot n^2$

$= J_m + J_{arm} \cdot n^2$

$B_{eq} = B_m + B_{arm} \cdot n^2$

Again, you get a third order system unless you neglect L_m

The block diagram of the closed-loop system is given by:



where $K(s)$ is the transfer function of the controller and $G(s)$ is the T. F. of the arm (including reduction gears)

To find the closed-loop transfer function $\frac{\theta_L}{\theta_c}$ we could, for instance write down the equations:

$\theta_e = \theta_c - \theta_L$, $\quad \theta_L = G(s) K(s) \theta_e$

Eliminating θ_e yields: $G \cdot K(\theta_c - \theta_L) = \theta_L$ // $GK \theta_c = (1+GK)\theta_L$

$$\Rightarrow \boxed{\frac{\theta_L}{\theta_c} = \frac{GK}{1+GK}}$$

This is a special case of Mason's formula:

Signal Flow Diagrams and Mason's Formula

(Reference: sections 2.3 and 2.4, Linear Control Systems, Phillips & Harbor)

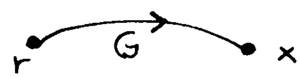
They provide an alternative representation of Transfer Function relationships and an alternative (often simpler) to Cramer's rule or block diagram manipulations for computing T. F.

Rules:

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

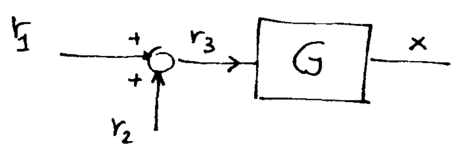
Block Diagram:

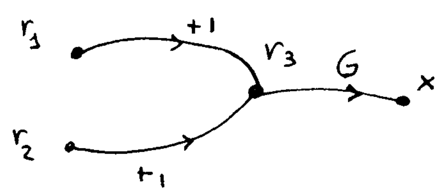
Signal flow:



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

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

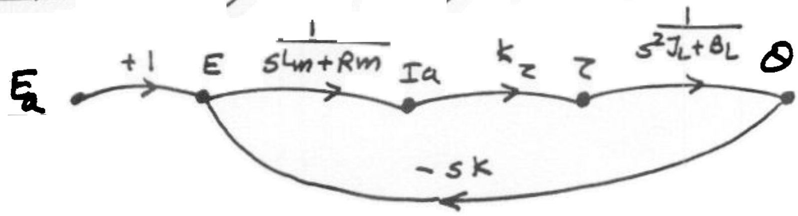




$r_3 = (+1)\cdot r_1 + (+1)\cdot r_2$

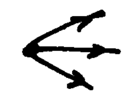
$x = G\, r_3$

Example: Signal flow graph representation of the DC motor



Some Terminology:

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



sink node: A node with incoming signals only

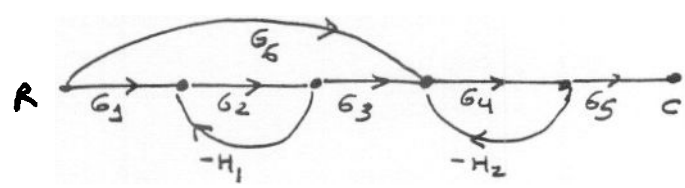
Path: Continuous connection of branches between 2 nodes (directed)

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 <u>nodes</u> in common.

Example:



2 loops: $-G_2 H_1$ (L_1)
$-G_4 H_2$ (L_2)

Path $G_6 G_4 G_5$ does not touch L_1
Path $G_1 G_2 G_3 G_4 G_5$ 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

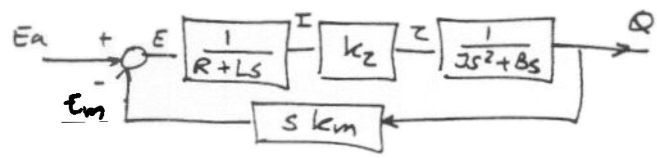
$$T_{CR} = \frac{1}{\Delta} \sum_{k=1}^{P} M_k \Delta_k = \frac{1}{\Delta}\left(M_1 \Delta_1 + \qquad + M_P \Delta_P\right)$$

Where :

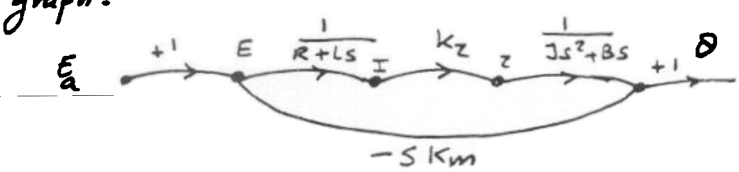
- $\Delta = 1 - \left(\Sigma \text{ gains individual loops}\right) + \Sigma\left(\text{products of } \underline{\text{non-touching}} \text{ pairs of loops}\right)$
  $- \Sigma\left(\text{products of triplets of } \underline{\text{non-touching}} \text{ loops}\right)$
  $+ \cdots$

- M_k = Gain of the k^{th} path between R and C
- Δ_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 : $L_1 = \dfrac{-k_z k_m s}{(R+Ls)(Js+B)s} \Rightarrow \Delta = 1 + \dfrac{k_z k_m s}{(R+Ls)(Js+B)s}$

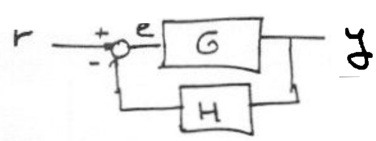
Only 1 path from E_a to Θ : $M_1 = \dfrac{k_z}{(R+Ls)(Js+B)s}$

$\Delta_1 = 1$

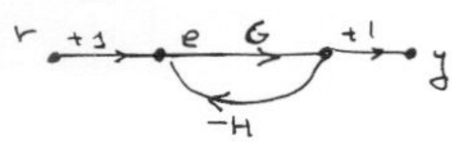
$$T_{\Theta E_a} = \frac{1}{\Delta}\cdot M_1 \Delta_1 =$$

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

Note : This is a special case of :



In signal flow form :

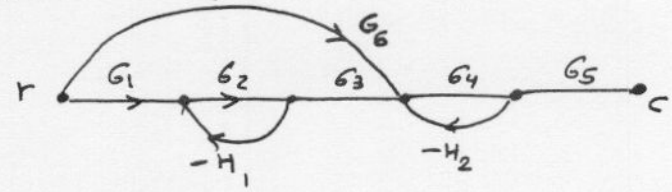


$L_1 = -GH$

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

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

Example 2: (Example 2.14 text)



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 $\Rightarrow \Delta_2 = 1 + G_2 H_1$

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

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

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

(This essentially says $y = y_1$)

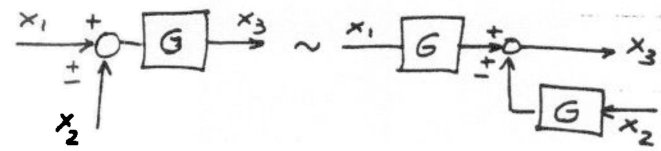
However, the restriction of the input being a source node can't be dealt with in this form.

- Basic operations on systems (Block Diagrams)

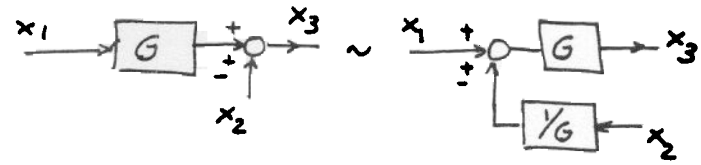
Here we are going to learn how to operate on block diagrams. This will allow us to obtain simpler (hopefully) diagrams, easier to solve
The basic operations are:

1) $x \rightarrow \boxed{G_1} - \boxed{G_2} \xrightarrow{y}$ $\sim$ $x \rightarrow \boxed{G_1 G_2} \rightarrow y$ (cascade)

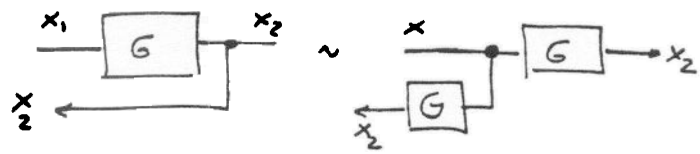
2) Moving a summing junction behind a block



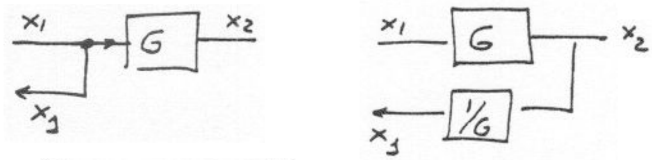
3) Moving a summing junction ahead of a block



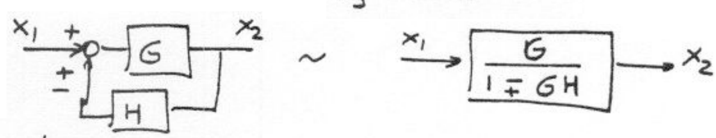
4) Moving a "tap" ahead of a block



5) Moving a "tap" behind a block

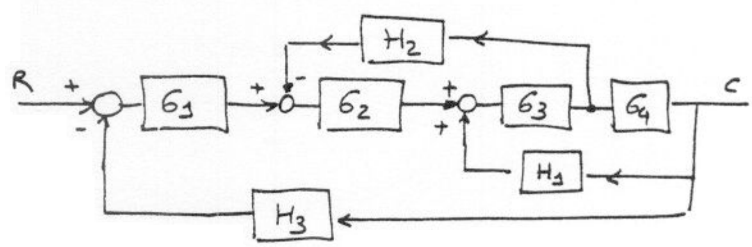


6) Eliminating a feedback loop:



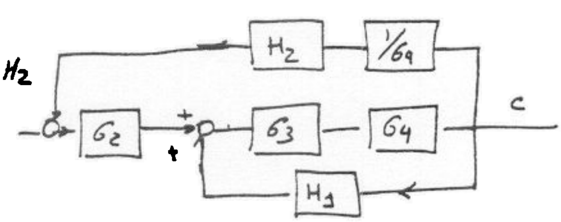
(This follows from Mason's formula)

• Example of application:

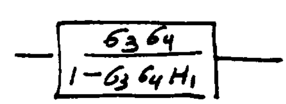


Want to find T_{CR}

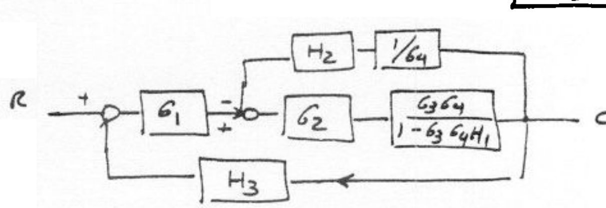
• Step 1: Move the tap for H_2 behind G_4



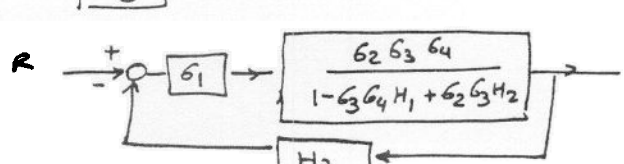
• Step 2: Eliminate the feedback loop $G_3 G_4 H_1$



Now we have:



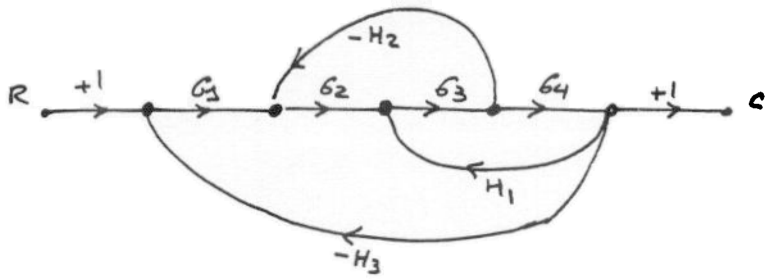
• Step 3: Eliminate the inner loop:



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

$$R \longrightarrow \boxed{\dfrac{G_1 G_2 G_3 G_4}{1 - G_3 G_4 H_1 + G_2 G_3 H_2 + G_2 G_3 G_4 G_1 H_3}} \longrightarrow C$$

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)

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

$\Rightarrow \dfrac{M}{\Delta} =$ $G_2 G_3 H_2 \; - G_3 \; {}_4 H_1 + G_1 G_2 G_3 G_4 H_3$