Binghamton University
State University of New York

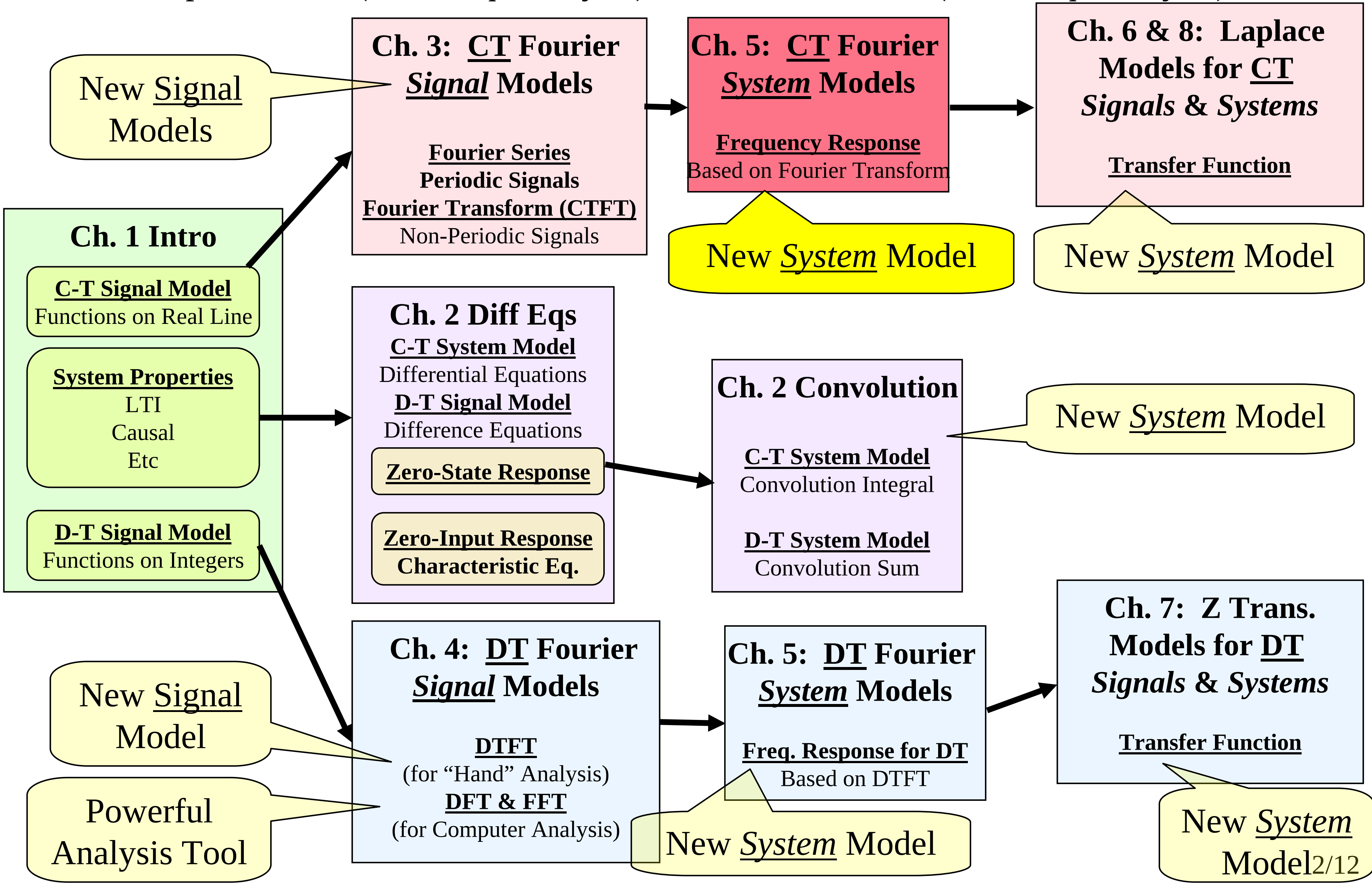
EECE 301
Signals & Systems
Prof. Mark Fowler

Note Set #18

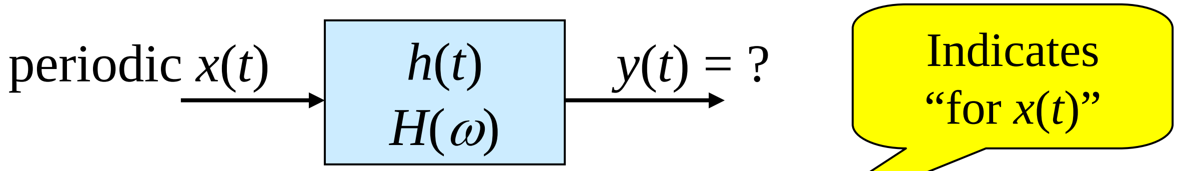
- C-T Systems: Frequency-Domain Analysis of Systems
- Reading Assignment: Section 5.2 of Kamen and Heck

Course Flow Diagram

The arrows here show conceptual flow between ideas. Note the parallel structure between the pink blocks (C-T Freq. Analysis) and the blue blocks (D-T Freq. Analysis).



5.2 Response to Periodic Inputs

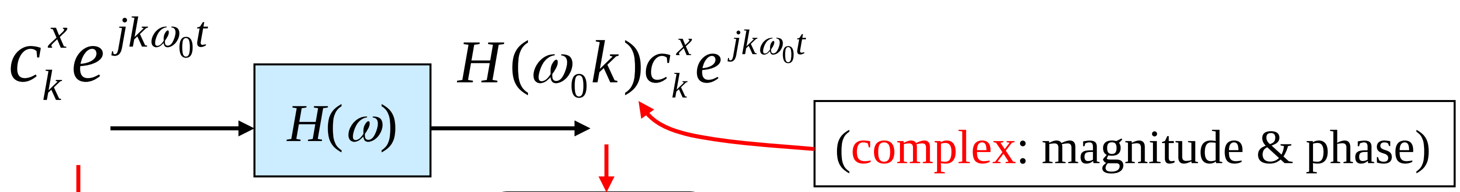


Since $x(t)$ is periodic, write it as FS: $$x(t) = \sum_{k=-\infty}^{\infty} c_k^x e^{jk\omega_0 t}$$

So, the input is a sum of terms

Linear System: So… Output = Sum of Individual Responses

But each individual response is to a complex sinusoid input $\Rightarrow$ EASY!



Sum these to get input

$$x(t) = \sum_{k=-\infty}^{\infty} c_k^x e^{jk\omega_0 t}$$

Sum these to get output

$$y(t) = \sum_{k=-\infty}^{\infty} \underbrace{[H(\omega_0 k) c_k^x]}_{} e^{jk\omega_0 t}$$

FS coefficient of $y(t)$

General Insights from this Analysis

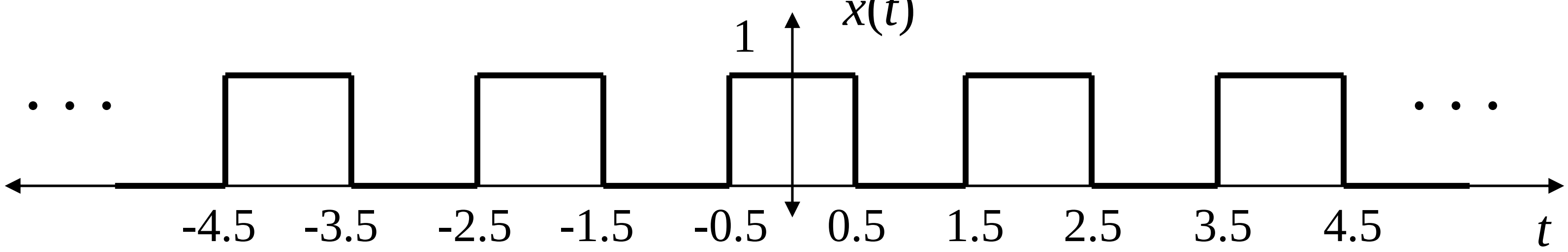
1. periodic in, periodic out

2. The system's frequency response $H(\omega)$ works to modify the input FS coefficients to create the output FS coefficients:

$$c_k^y = H(k\omega_0)c_k^x$$

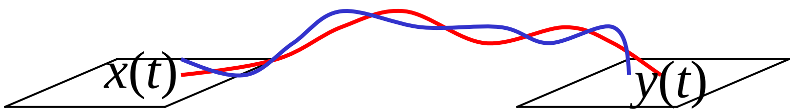
Example (Ex. 5.4 with Some Injected Reality)

Problem: suppose you have a circuit board that has a digital clock circuit on it. It makes the rectangular pulse train shown below:



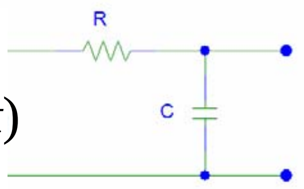
(Of course most digital clock circuits would run much faster)

Suppose you need to connect this clock signal to a circuit on another circuit board using a twisted pair of wires:



Q: What effect does the cable have on the clock signal at the 2nd board???

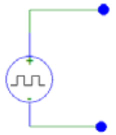
Pair of wires can be modeled as an RC circuit: $x(t)$



$y(t)$

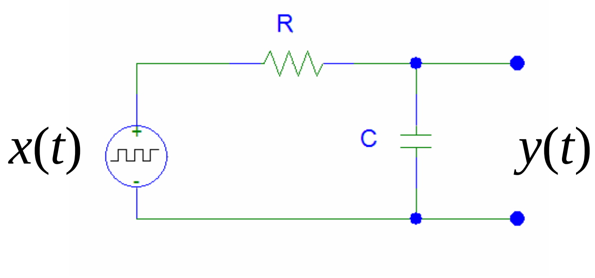
Assume: The circuit "driving" the cable has an infinitesimally small output impedance (that is good!):

Thevenin of driver: $x(t)$



Assume: The circuit being "driven" by the cable has infinite input impedance (that is good!)   i.e. No loading of the RC circuit

So…



→ (goes to driven circuit having infinite input impedance)

Goal: Perform an analysis to enable you to recommend an acceptable value of cable RC time constant   **(Analysis Drives Design!)**

Step 1: Analytically find FS of input and compute truncated FS sum:

From Ex. 3.4 we get:

$$c_k^x = \begin{cases} \dfrac{1}{k\pi}, & k = \pm 1, \pm 5, \pm 9, \ldots \\ -\dfrac{1}{k\pi}, & k = \pm 3, \pm 7, \pm 11, \ldots \\ 0, & k = \pm 2, \pm 4, \pm 6, \ldots \\ \dfrac{1}{2}, & k = 0 \end{cases}$$

Indicates "for $x(t)$"

$$x(t) \approx \sum_{k=-N}^{N} c_k^x e^{jk\omega_o t}$$

Then plot vs. time t

Step 2: Find cable's frequency response as a function of RC:

$$H(\omega) = \frac{1}{1 + j\omega RC}$$

(See Ex. in section 5.1)

Step 3 (optional) (But it really helps you see what is going on!)

Look at frequency domain plots of Input and System (for various RC values)

"stem" plot of FS coefficients' Magnitude $\left|c_k^x\right|$

"continuous" plot of Magnitude of system's Frequency Resp. $|H(\omega)|$

Step 4 (optional) (This also really helps you see what is going on)

Compute output FS coefficients: $c_k^y = H(k\omega_0)c_k^x$

Look at the result → "stem" plot of $\left|c_k^y\right|$

Step 5: Compute truncated FS sum to see output signal

$$y(t) \approx \sum_{k=-N}^{N} c_k^y e^{jk\omega_0 t}$$

Plot vs. time t

See plots on next 3 pages for three RC time constant values:

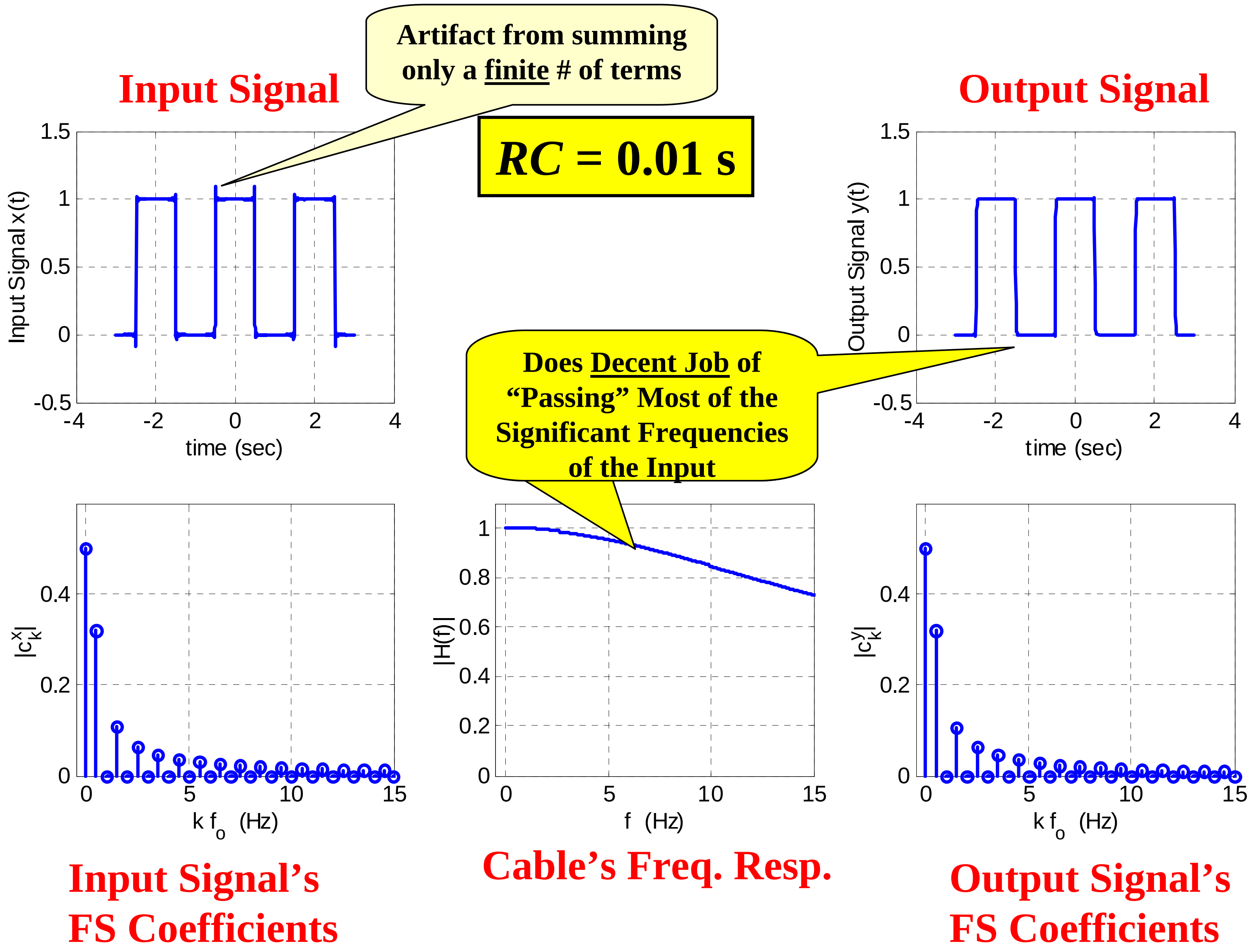
RC = 0.01 s

RC = 0.1 s

RC = 1 s

Note: Short RC time constant passes high frequencies better than long RC time constant

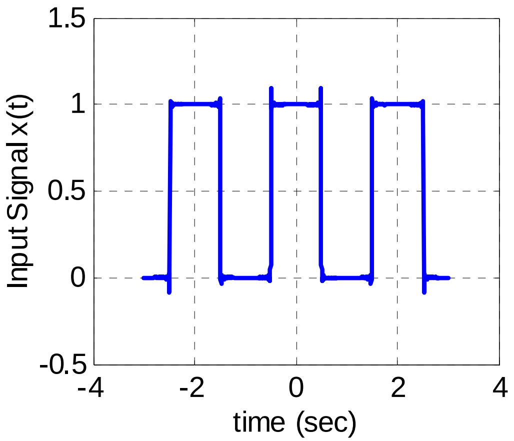
RC Circuit Analysis w/ Square Wave Input



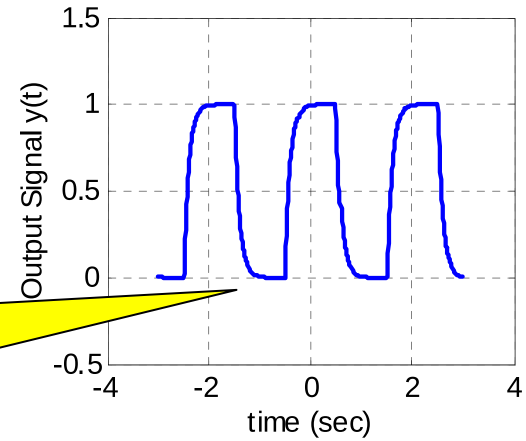
RC Circuit Analysis w/ Square Wave Input

RC = 0.1 s

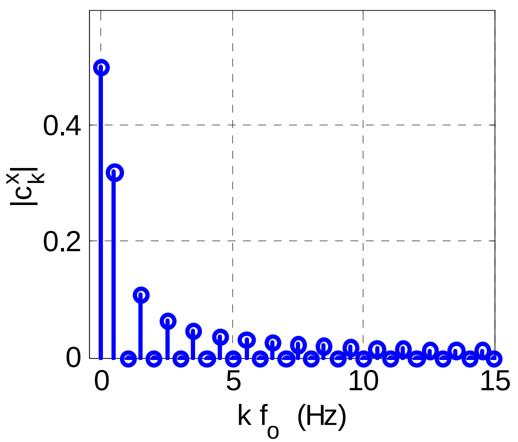
Input Signal



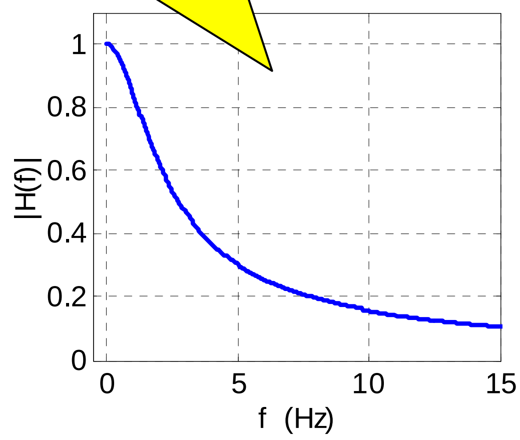
Output Signal



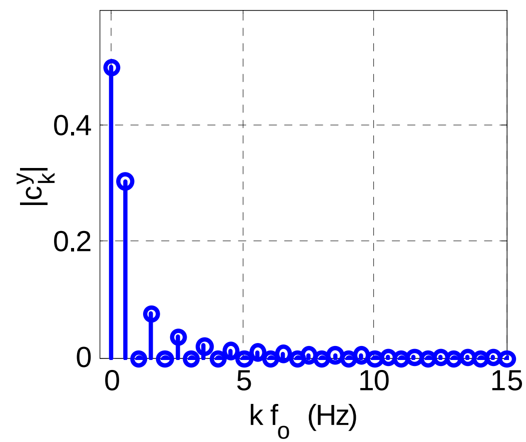
Does <u>Moderate Job</u> of "Passing" Most of the Significant Frequencies of the Input



Input Signal's FS Coefficients



Cable's Freq. Resp.

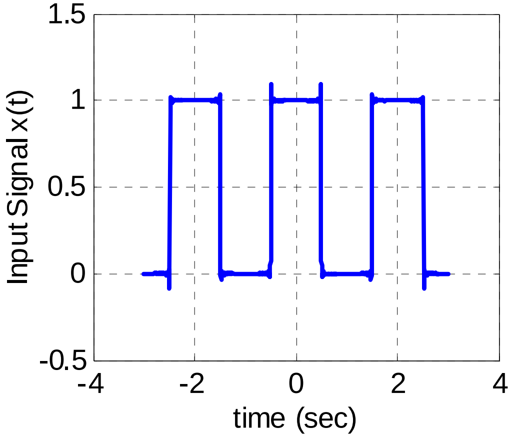


Output Signal's FS Coefficients

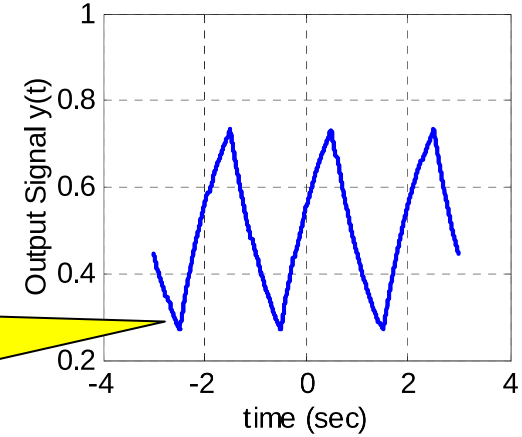
RC Circuit Analysis w/ Square Wave Input

***RC* = 1 s**

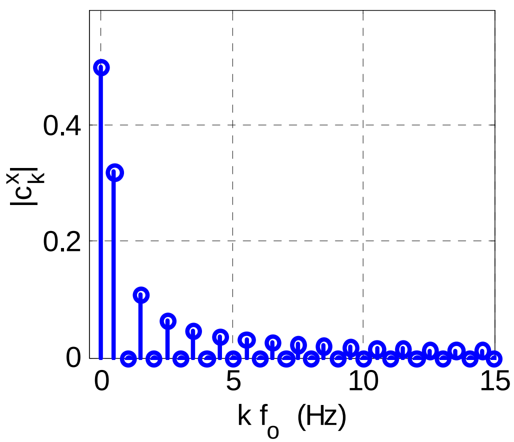
Input Signal



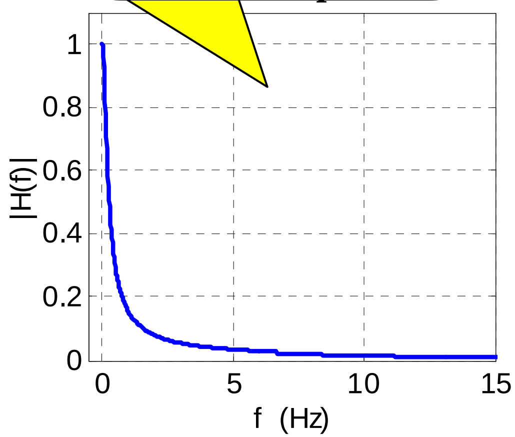
Output Signal



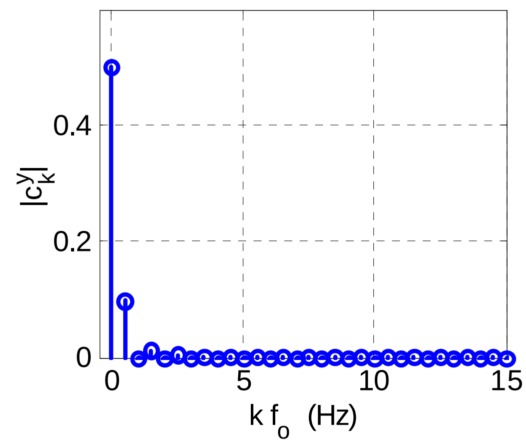
Does Poor Job of “Passing” Most of the Significant Frequencies of the Input



Input Signal’s FS Coefficients



Cable’s Freq. Resp.



Output Signal’s FS Coefficients

Insight from Example:

- We used a simple model for the cable to make it easy to analyze
  - But… the method would be the same even if we had a more detailed model for the cable
- The input clock signal has nice sharp transitions due to its significant high frequency components
- Cables that significantly suppressed the input's high frequency components provided a low-quality clock signal to the 2^{nd} board
- We made assumptions about the driver circuit and the driven circuit
  - The driver was assumed to have zero output resistance
    - If that were not true, its output impedance gets added to the resistor and that would further degrade the performance (in fact the driver's output impedance may be more than the cable resistance in which case it would be the dominant factor
  - The driven circuit was assumed to have infinite input impedance
    - If that were not true we would have to combine it in parallel with the capacitor's impedance… this would further degrade the performance
- Typically the RC value of a cable increases with length
  - So performance would decrease with length of cable

Matlab Code for Example's Plots

4 types of indices:
1,5,9,…
2,6,10,..
3,7,11,…
4,8,12,…

```
function [x,t]=example_5_4(RC)

k=1:4:200;
K=[k;k+1;k+2;k+3];
C_k=1./(K*pi);   % fill matrix with this form… keep first row
C_k(2,:)=zeros(size(k));   % replace 2nd row w/ zeros
C_k(3,:)=-1*C_k(3,:);     % replace 3rd row w/ negatives
C_k(4,:)=zeros(size(k));   % replace 4th row w/ zeros
c_x_k=C_k(:);   % turn into col. vector by going down matrix columns

t=-3:(6/800):3;   % create time vector with approp. spacing
k=(1:max(max(K)));  % create FS term index
[T,K]=meshgrid(t,k);  % create time matrix and index matrix

wo=pi;
EXP_pos=exp(j*T.*K*wo);  % Each row is a sinusoid term
EXP_neg=exp(-j*T.*K*wo);
%% Compute the FS summation to get approx. input time signal
x=0.5+sum(c_x_k(:,ones(1,length(t))).*EXP_pos)…
      +sum(conj(c_x_k(:,ones(1,length(t)))).*EXP_neg);
  % The above cmnd adds up the rows of EXP weigthed by the c_x_k
subplot(2,3,1)
plot(t,x)
xlabel('time (sec)')
ylabel('Input Signal x(t)')

subplot(2,3,4)
w_k=k*wo;   % create vector of FS frequencies
% In the next line we have to attach c_0=0.5 and its freq
stem([0 w_k]/(2*pi),[0.5 abs(c_x_k).'])   % plot vs freq in Hz
xlabel('k f_o  (Hz)')
ylabel('|c^x_k|')
axis([-0.5 15 0 0.6])
```

$$T = \begin{bmatrix} t_1 & t_2 & t_3 & \cdots \\ t_1 & t_2 & t_3 & \cdots \\ t_1 & t_2 & t_3 & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \qquad K = \begin{bmatrix} 1 & 1 & 1 & \cdots \\ 2 & 2 & 2 & \cdots \\ 3 & 3 & 3 & \cdots \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

```
subplot(2,3,5)
w=0:0.1:max(w_k);   % create finely-spaced frequency
H=1./(1+j*w*RC);   % compute Freq Resp @ these Freqs
plot(w/(2*pi),abs(H))  % plot vs. freq in Hz
xlabel('f  (Hz)')
ylabel('|H(f)|')
axis([-0.5 15 0 1.1])

subplot(2,3,6)
H_k=1./(1+j*w_k*RC);    % compute Freq Resp at FS freqs
H_0=1./(1+j*0*RC);
c_y_k=c_x_k.*(H_k.');  % compute output FS coeffs
stem([0 w_k]/(2*pi),[0.5*H_0 abs(c_y_k).'])
xlabel('k f_o  (Hz)')
ylabel('|c^y_k|')
axis([-0.5 15 0 0.6])

subplot(2,3,3)
%% FS summation to get approx. output time signal
y=0.5*H_0+sum(c_y_k(:,ones(1,length(t))).*EXP_pos)…
   +sum(conj(c_y_k(:,ones(1,length(t)))).*EXP_neg);
plot(t,y)
xlabel('time (sec)')
ylabel('Output Signal y(t)')
```