Imperial College London

DE2.3 Electronics 2 for Design Engineers

Tutorial 2

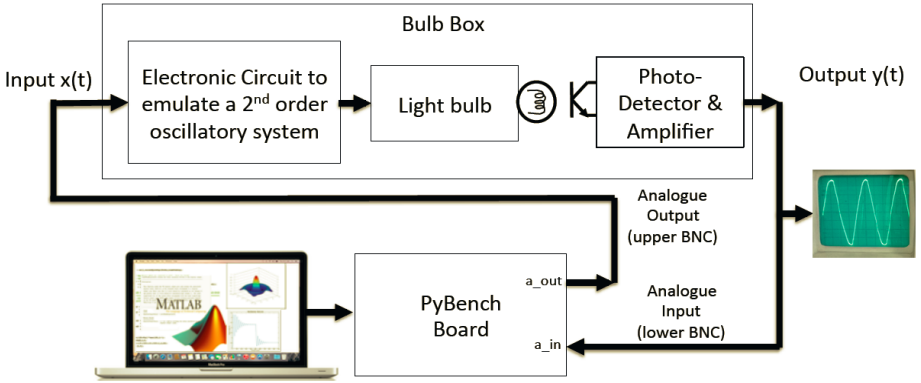
System Characterisation Lab 2 Explained

Prof Peter YK Cheung

Dyson School of Design Engineering

URL: www.ee.ic.ac.uk/pcheung/teaching/DE2_EE/
E-mail: p.cheung@imperial.ac.uk

Bulb Box

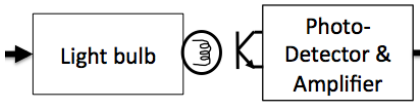


- We are interested in mathematical modelling system.
- Bulb Box is designed to behaviour like a 2nd order system + a non-linear system with some delay (the light bulb)
- We want to verify that the mathematical model is a good representation.
- We also wan to explore the limitations of this model

Key aspects of Lab 2

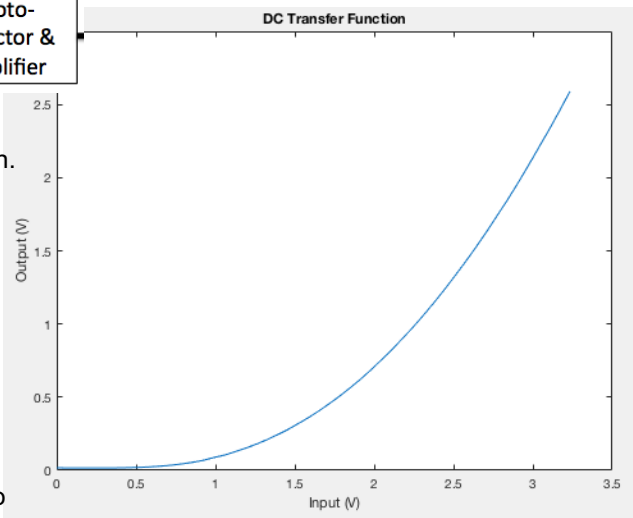
1. DC characteristics – no time variation, just what goes in, and what comes out.
2. Steady state response to sinusoidal signals at different frequencies – we call this frequency response.
3. Use of Matlab for modelling and simulation.
4. Transient behaviour of the system – we call this step response.
5. Impact of non-linearity in the system.

Exercise 1 – DC Characteristic

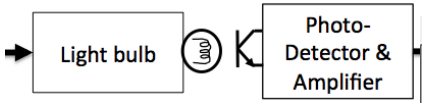


- y = F (x)
- F is a non-linear function.
- F is a quadratic function because:

  light intensity $\propto x^2$
- Light is dependent temperature of filament in bulb
- Temperature is dependent on energy consumed
- Energy is proportional to x^2.

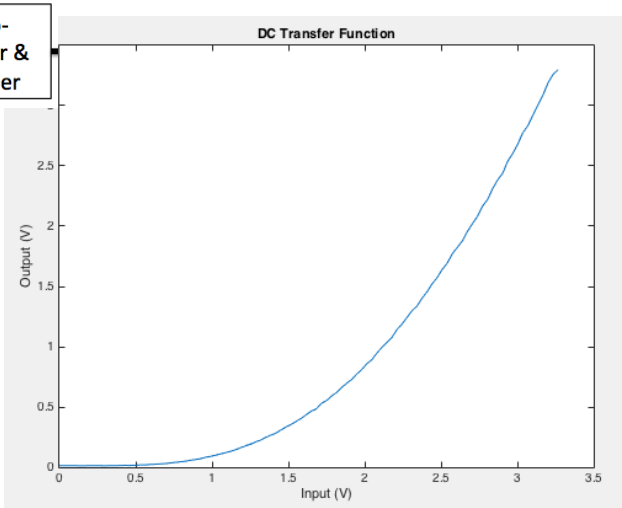


Exercise 1 – DC Characteristic



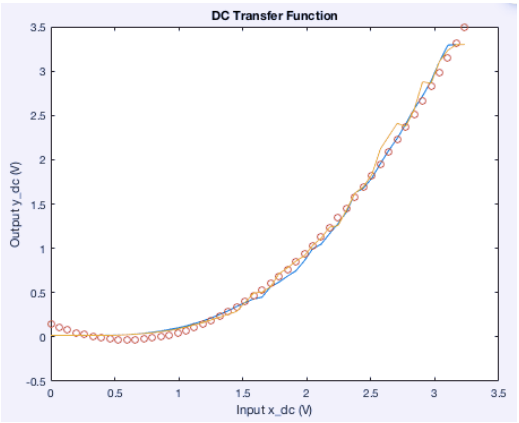
- y = F (x)
- F is a non-linear function.
- F is a quadratic function because:

  light intensity $\propto x^2$
- Light is dependent temperature of filament in bulb
- Temperature is dependent on energy consumed
- Energy is proportional to x^2.

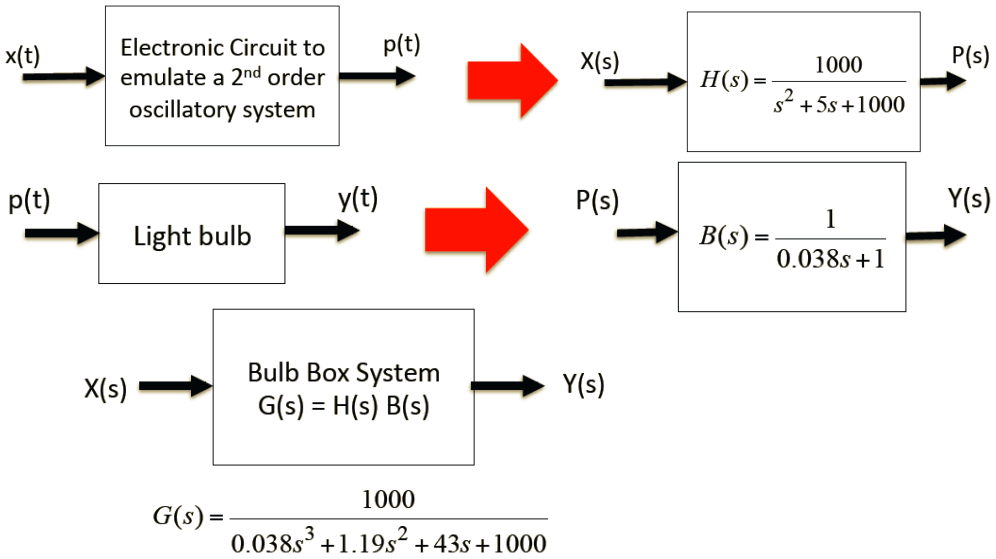


Exercise 1 – My solution

```
% Lab 3 Exercise 1 - DC Characteristics
clear all
pb = PyBench('/dev/tty.usbmodem1422');
pb.samp_freq = 100;
NSTEPS=50;
tic
for i = [1:NSTEPS]
    v = (i-1)*3.3/NSTEPS;
    x_dc(i) = v;
    pb.dc(v);
    pause(2);
    data = pb.get_block(10);
    y_dc(i) = mean(data);
end
pb.dc(0.0);
toc
plot(x_dc,y_dc)
xlabel('Input x\_dc (V)');
ylabel('Output y\_dc (V)');
title('DC Transfer Function');
% Fit a polynomial function to data
p = polyfit(x_dc,y_dc,2);
f = polyval(p,x_dc,'x');
```



Exercise 2 – Modeling dynamics in a system



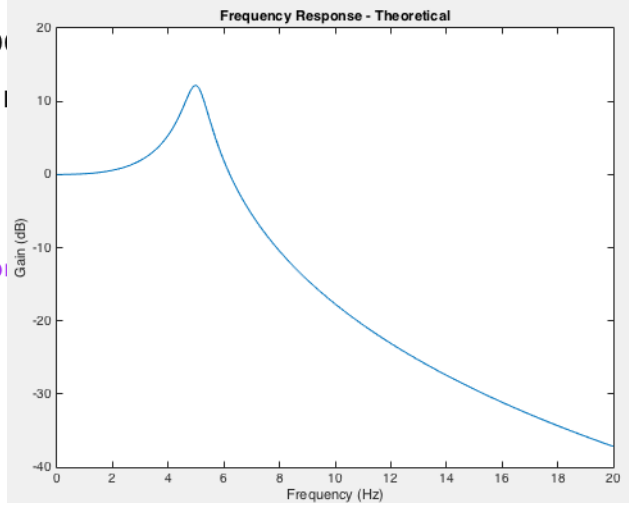
Exercise 2 – Predict the frequency response

```
% Program to plot theoretical freq. response of Bulb Box
clear all
f=(0:0.1:20);
D = [0.038 1.19 43 1000];  % specify denominator
s= i*2*pi*f;
G = 1000./abs(polyval(D,s));
Gdb = 20*log10(G);
figure;
plot(f,Gdb);
xlabel('Frequency (Hz)');
ylabel('Gain (dB)');
title('Frequency Response - Theoretical');
```

$$G(s) = \frac{1000}{0.038s^3 + 1.19s^2 + 43s + 1000}$$

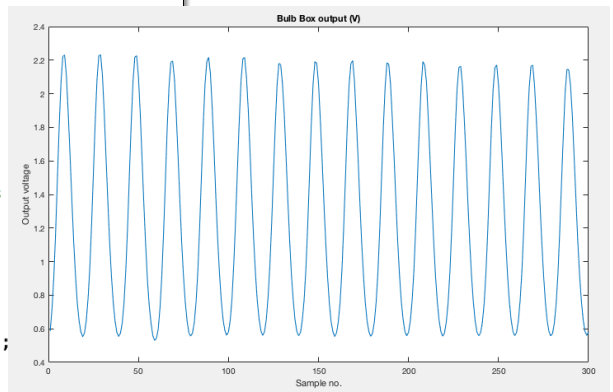
Exercise 2 – Predict the frequency response

```
% Program to plot theoretical freq. response of Bulb Box
clear all
f=(0:0.1:20);
D = [0.038 1.19 43 100
s= i*2*pi*f;
G = 1000./abs(polyval(
Gdb = 20*log10(G);
figure;
plot(f,Gdb);
xlabel('Frequency (Hz)
ylabel('Gain (dB)');
title('Frequency Respo
```

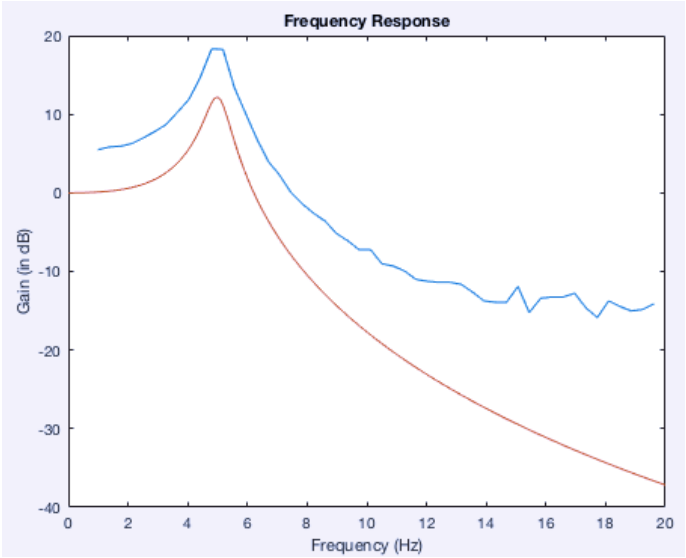


Exercise 3 – Measure the frequency response

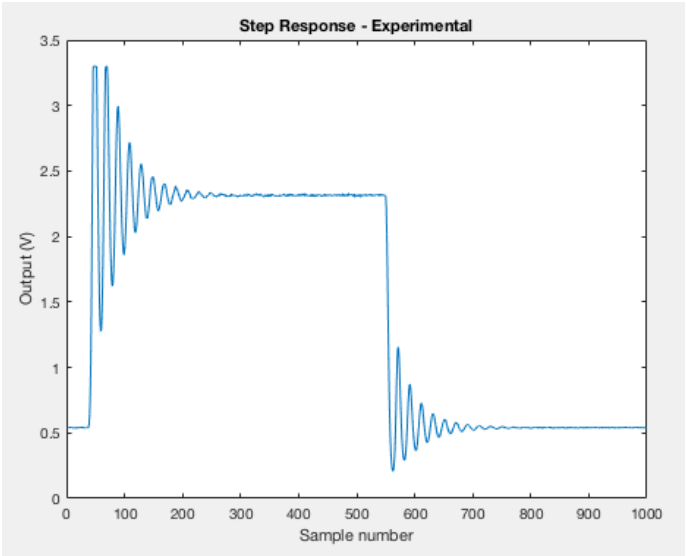
```
1   % Measure system gain at frequency f_sig
2   %
3 - clear all
4 - pb = PyBench('COM4');
5   % Generate a sine wave at sig_freq Hz
6 - max_x = 2.1;
7 - min_x = 1.9;
8 - f_sig = 5.0;
9 - pb=pb.set_sig_freq(f_sig);
10 - pb=pb.set_max_v(max_x);
11 - pb=pb.set_min_v(min_x);
12 - pb.sine();
13 - pause(2)
14  % Capture output y(t)
15 - pb=pb.set_samp_freq(100);  %
16 - N = 300;     % no of samples
17 - y = pb.get_block(N);
18  % plot signal
19 - plot(y);
20 - xlabel('Sample no.');
21 - ylabel('Output voltage');
22 - title('Bulb Box output (V)');
23  % Compute Gain in dB
24 - x_pk2pk = max_x - min_x;
25 - y_pk2pk = max(y) - min(y);
26 - G = 20*log10(y_pk2pk/x_pk2pk)
```



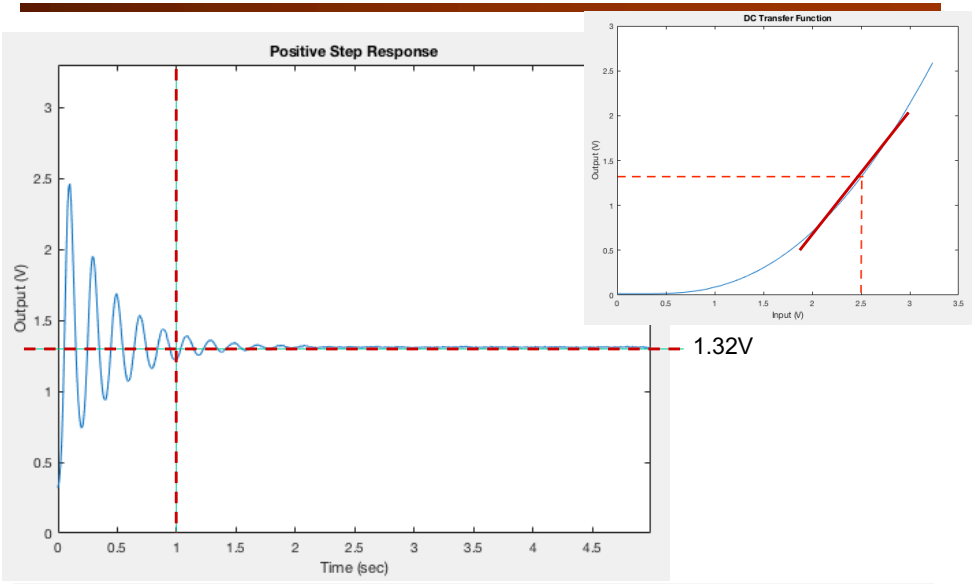
Exercise 3 – Theory vs Measurements



Exercise 4 – Step Response



Exercise 4 – Positive Step



Exercise 4 – Negative Step