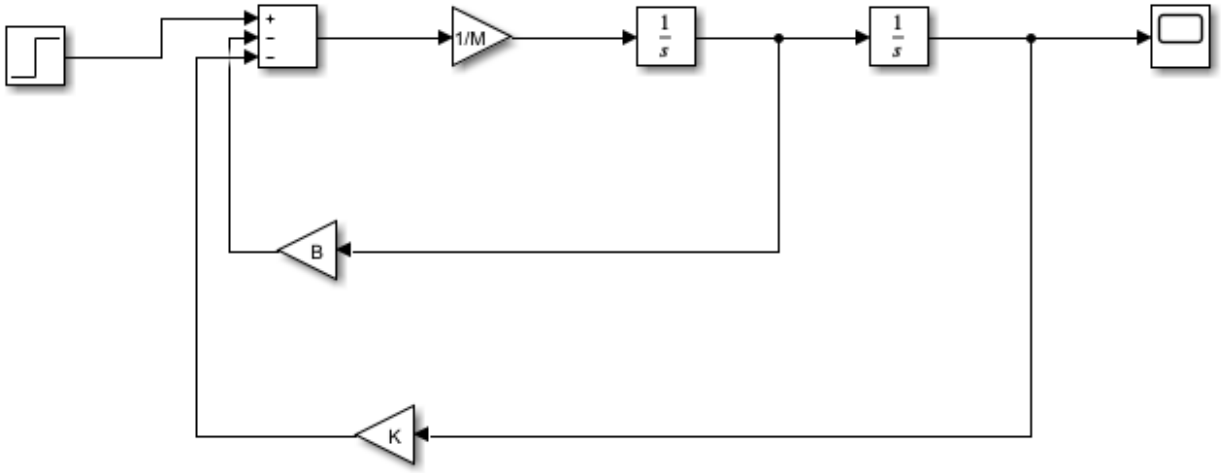
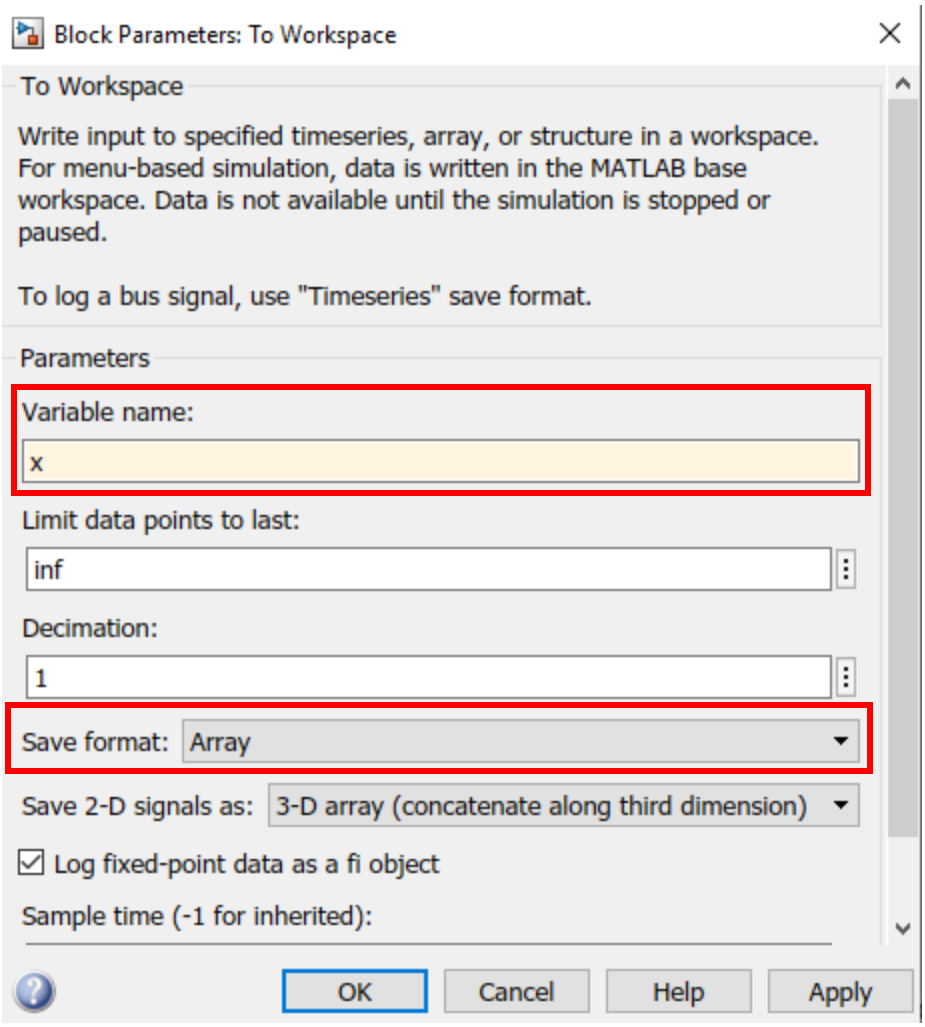
Here, a more detailed overview of how to do page 7 and 8 of lab manual (17 and 18 of pdf) is presented.

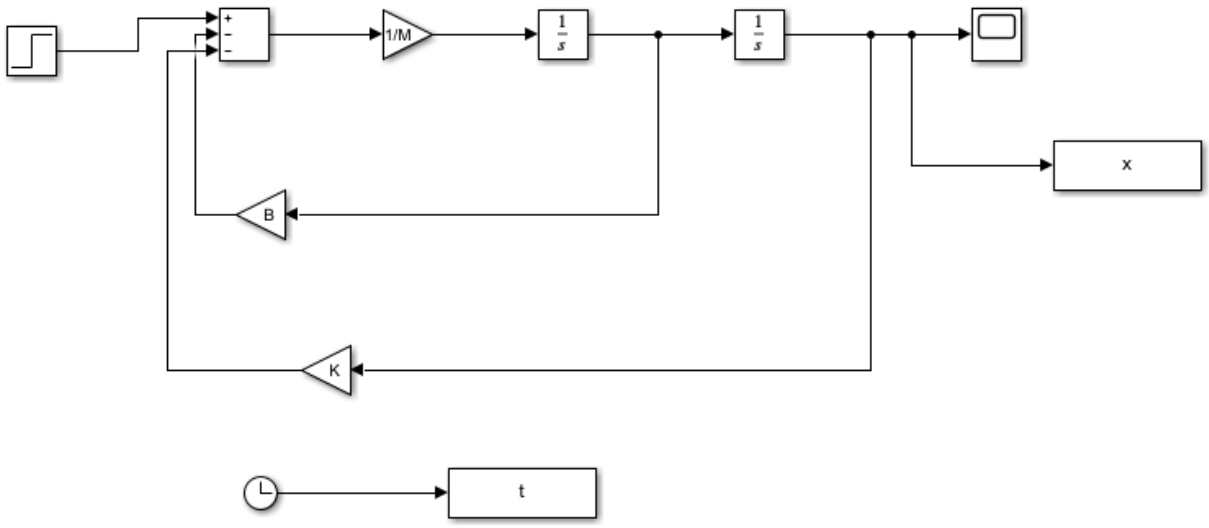
First, make sure that in the Simulink block diagram, the values of gains are parametric values of B, K, and 1/M, meaning that make sure you have not given numeric values to gains already as we are going to change them using MATLAB script.



Then, please add the 'clock' block, and two 'To Workspace' blocks. In order to add blocks quickly without going to the library browser, double click anywhere in the Simulink environment and type down 'To Workspace' for example to see a list of blocks, and then add the one you want from the list. Double click on the 'To Workspace' blocks, and modify their name and save type. Modify the name of one of them to '`x`' and one of them to '`t`'. Also change the save format of both of them to 'Array'. The save format is how the data will be collected from the SIMULINIK to pass it to the MATLAB script's workspace.

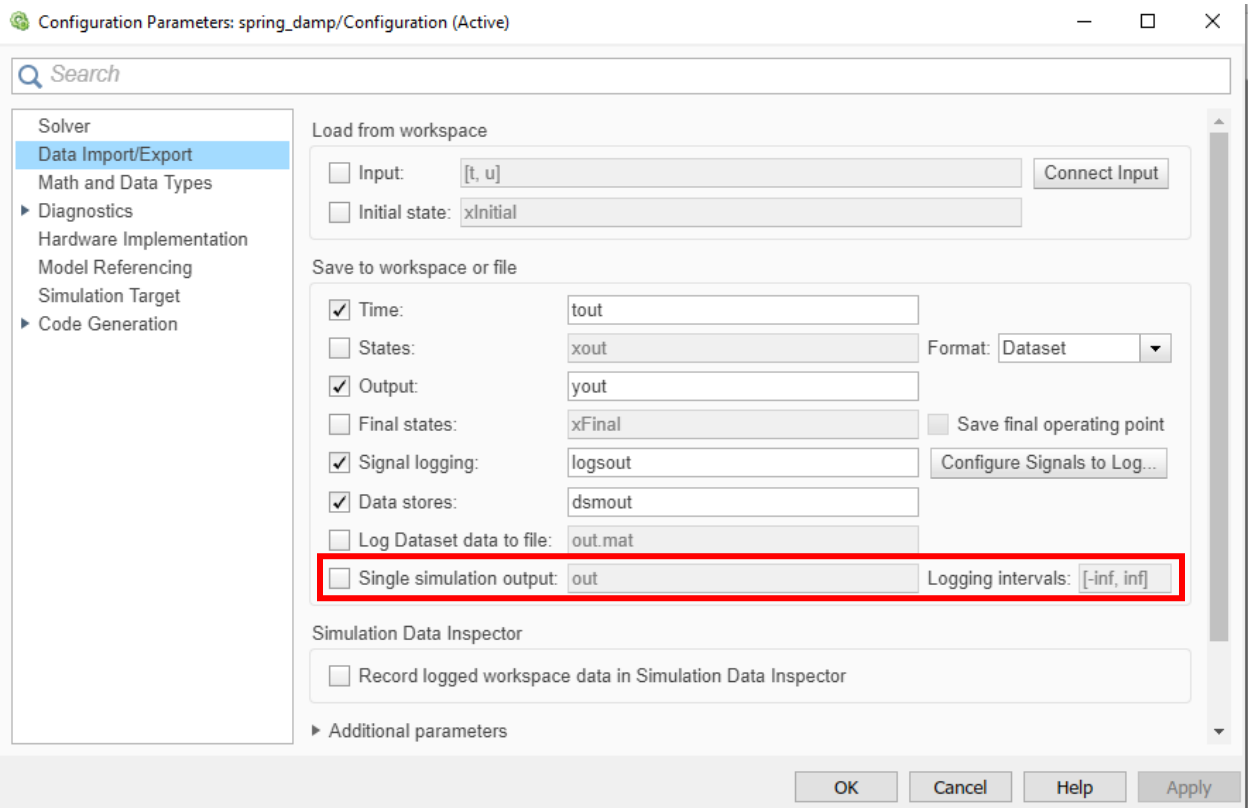


Make the following connections between the blocks:



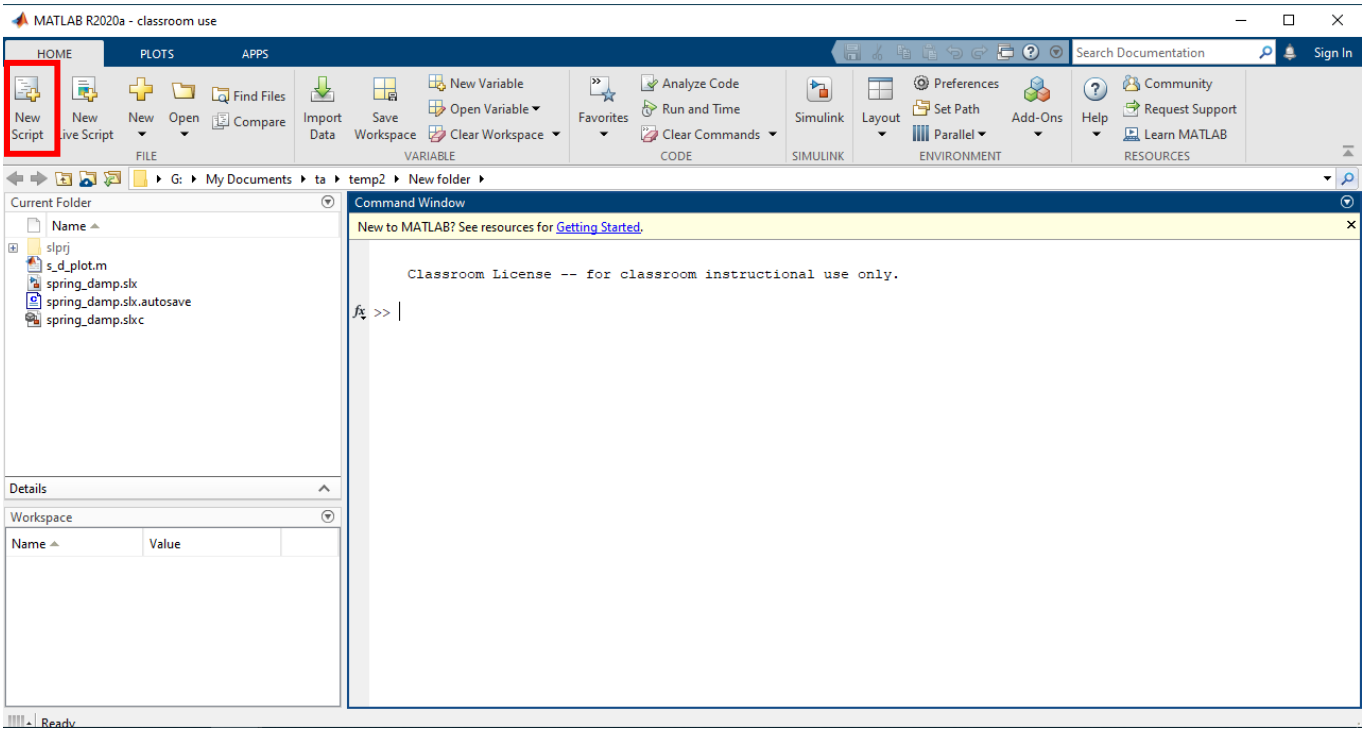
If your 'To Workspace' blocks are in '`out.t`' and '`out.x`' format, from the top of the SIMULINK tabs go the 'Modelling' tab and then select the gear icon above the 'Model Settings' (Model Configuration Parameters can be access via Ctrl+E as well).

In the opened window, go to the 'Data Import/Export' then uncheck 'Single Simulation Output' if it is checked.



Press ok, and now you should have '`t`' and '`x`' instead of '`t.out`' and '`x.out`'.

Save your SIMULINK file with the name of '`spring_damp`'. Now, we want to create a MATLAB script file or m file. To create a new script file, when you boot MATLAB, from the top select 'New Script'. Note that you script file should be save in the same folder as your SIMULINK file we created previously in order to run what we write later.



Now, please write down the following code in your script environment:

```
clc;
M=2; %kg
K=16; %N/m
B=4; %Ns/m
fafinal = 8; %N
sim('spring_damp')
plot(t,x);
hold on
B=8; sim('spring_damp');plot(t,x)
B=12; sim('spring_damp');plot(t,x)
B=25; sim('spring_damp');plot(t,x)
hold off
```

Note that MATLAB script is a programming environment like python and C++. When you run a code in it, it starts going down through your code and execute your program line by line.

Here when we write `M=2,` like other programming languages, it means that it allocates a space in the memory to a variable named '`M`' and then it pours the value of '`2`' in this variable.

The semicolon `;` at the end of each line prevents the value of the variable to be shown each time the line is executed when running in the command window.

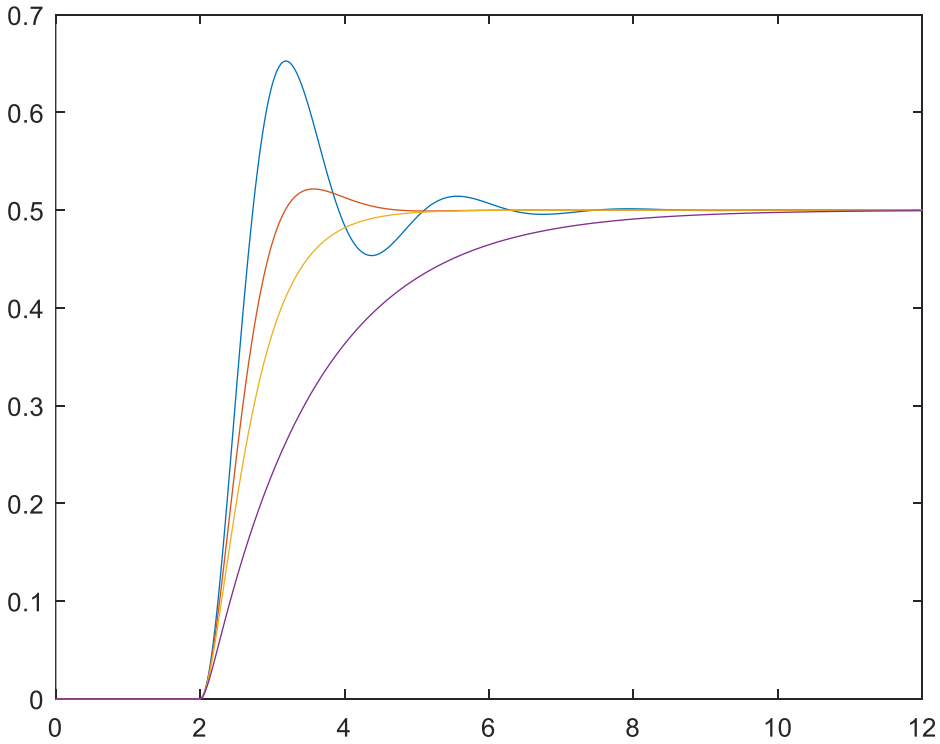
With `sim('spring_damp')` line, the SIMULINK file will be run.

`plot(t,x)` line simply plots x vs t.

without `hold on` and `hold off`, whenever we use plot command, the new plot will be overwritten on the previous one, which we don't want and we want all plots in one figure simultaneously.

This code basically changes damper constant B each time, run the Simulink, and plot the new vectors `of x` and `t`.

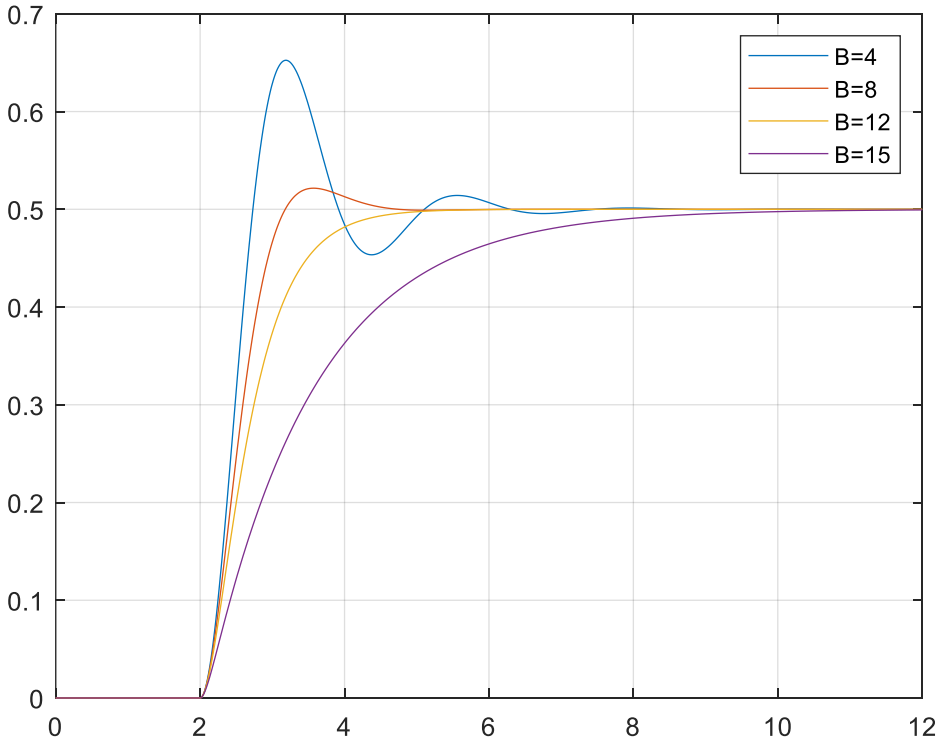
When you run the code from the green button at the top, you must see the following figure.



If you add these lines at the end of your code,

```
grid on
legend('B=4','B=8','B=12','B=15')
```

, your figure becomes like the following:



`grid on` will add a grid to your figure.

With `legend('B=4','B=8','B=12','B=15')` we can depict which line belongs to which case