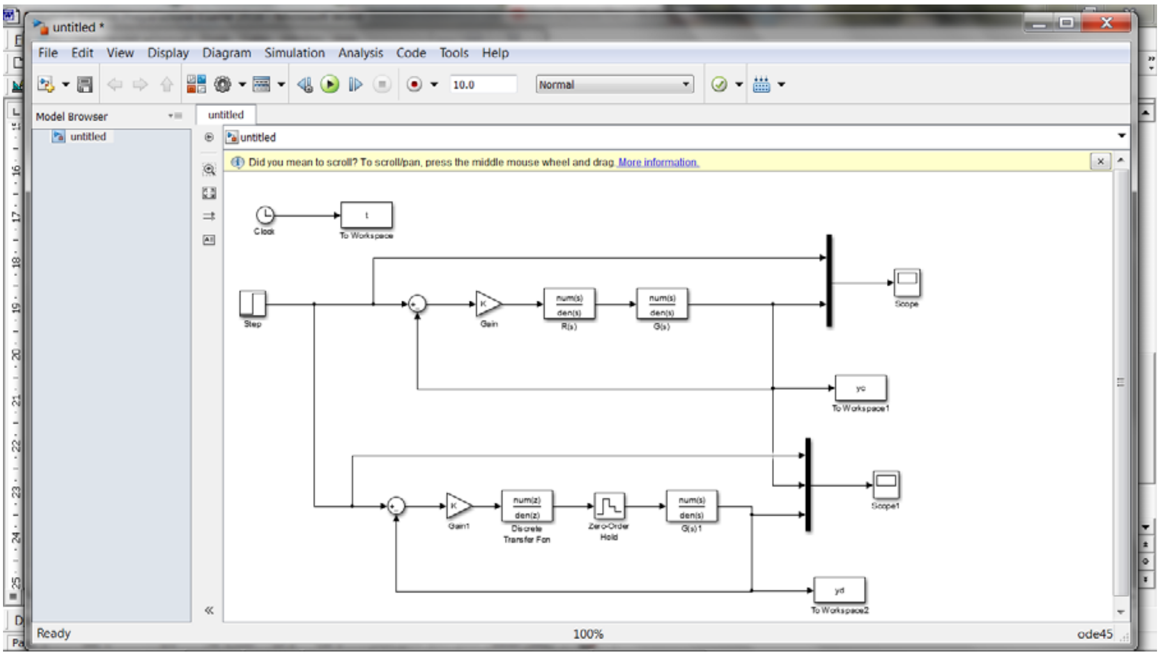
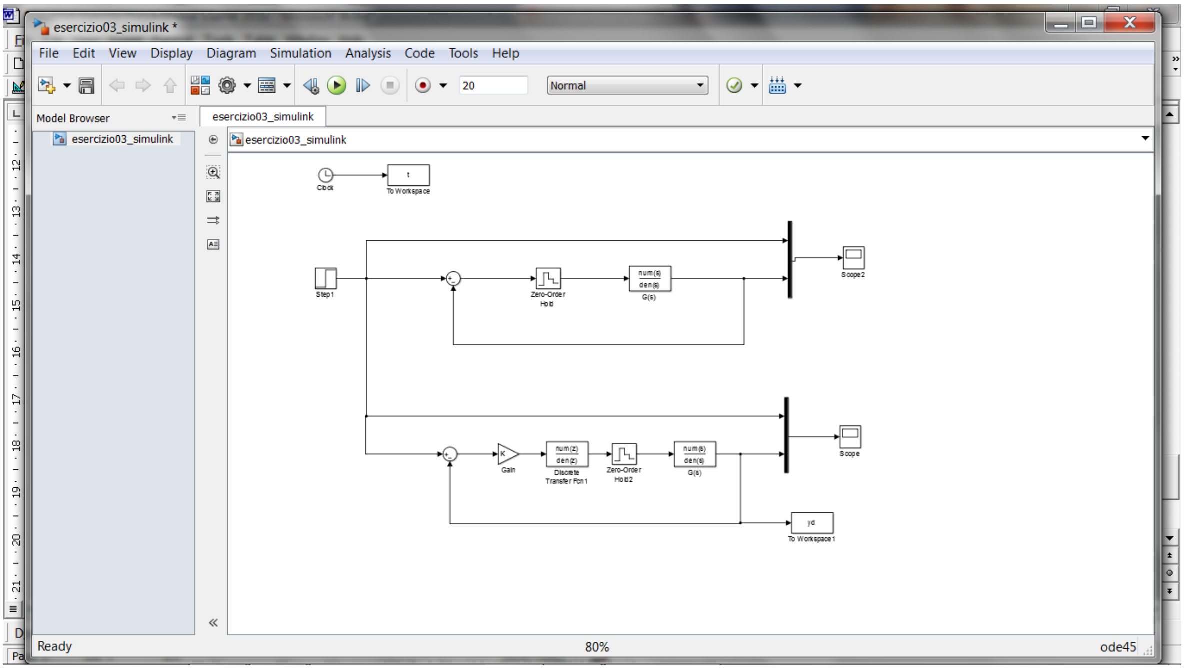
Solutions to the Phase Lead Design Example

```
s = tf('s');

Gs = 1/(s*(s+3)*(s+21));

[numGs,denGs] = tfdata(Gs,'v'); % Required for the Simulink implementation

Rs = (s/3 + 1)/(s/100 + 1);      % Zero in -3 e pole in -100 per la phase
                                 % lead network

[numRs,denRs] = tfdata(Rs,'v'); % Simulink implementation


Ga = Rs * Gs;                    % Overall transfer function

rlocus(Ga)                       % The root locus is depicted

grid                             % The constant delta locus is depicted

K = rlocfind(Ga)                 % Look for a point near to delta = 0.64

                                 % The required gain is:
                                 % K = 640. Try to find this value by
                                 % a trial and error procedure in order
                                 % to satisfy both the settling time Ta and
                                 % the maximum overshoot S%


                                 % Note that in general S% increases with K
                                 % whilst Ta decreases while increasing K
                                 % In this way, a trade-off value should be
                                 % determined

                                 % The Simulink implementation is provided below
```



Solutions to Phase Lag Design Example

```
s = tf('s');

Gs = 1/((s+1)*(s+2)*(s+10));

[numGs,denGs] = tfdata(Gs,'v'); % Simulink implementation


Cs=(s+1)/s                      % Continuous time regulator

[numCs,denCs]=tfdata(Cs,'v');   % Simulink implementation

Ga = Cs * Gs;                   % Overall controlled system


rlocus(Ga)                      % It generates the root locus for the
                                % overall closed loop system

grid                            % Constant delta points of the locus

K = rlocfind(Ga)                % It determines the gain near to delta = 0.9

                                % The staring value is K = 10.81


                                % The gain K satisfying the requirements
                                % is:
                                % K = 21. This value is determined using
                                % a trial and error procedure:
                                % In general, Ta decreases if K increases
                                % S% increases if K increases
                                %
                                % The final value is K = 10.81
%
% The Simulink implementation is reported below
```



Soluzion to Phase Lag Design Example

```
s = tf('s');

Gs = 0.1/(s*(s+1)*(s+10));

[numGs,denGs] = tfdata(Gs,'v'); % Simulink implementation

Rs = (s/10 + 1)/(s/2 + 1);      % Zero in -3 e pole in -100 for the
                                 % phase lag network

[numRs,denRs] = tfdata(Rs,'v'); % Simulink implementation


Ga = Rs * Gs;                    % Overall controlled system

rlocus(Ga)                       % The continuous time root locus is depicted

grid                             % The points at constant delta are depicted

K = rlocfind(Ga)                 % A point near to delta = 0.6 is determined

% The initial value can be K = 35 and a trial and error procedure is followed.


                                 % The gain K satisfying the requirements is:
                                 % K = 42. This value is finally found by
                                 % following a trial and error procedure.
                                 %
                                 % The following meta-rules can help the design:
                                 % In general, Ta increases is K decreases

                                 % Other possible values for K are also
                                 % 41 or 40, the performances are satisfied.

% The Simulink implementation is reported below
```