
```
function y = lab3(t)
t = 0:0.01:6.5;
t_sawf = (t>=4)&(t<6); t_saw = t(t_sawf);
t_zerof = (t>=2)&(t<4); t_zero = t(t_zerof);
t_sinf = (t>=6)&(t<6.5); t_sin = t(t_sinf);
t_constf = (t>=0)&(t<1.5); t_const = t(t_constf);
t_noise = (t>=1.5)&(t<=2); t_noise = t(t_noise);
% t_sin = 6:0.01:6.5;
A0 = 0; A = 2; f = 1;
delay = 6;
y_sin = A0 + A*sin(2*pi*f*(t_sin-delay));
plot(t_sin,y_sin)
axis([0 6.5 -1 4])

% Linear function
% t_saw = 4:0.01:6;
k = -2;
delay = 6;
y_saw = k*(t_saw-delay);
hold on
plot(t_saw,y_saw)

% Constant signal
% t_const = 0:0.01:1.5;
y_const = zeros(size(t_const))+2;
plot(t_const,y_const)

% Zero signal
% t_zero = 2:0.01:4;
y_zero = zeros(size(t_zero));
plot(t_zero,y_zero)

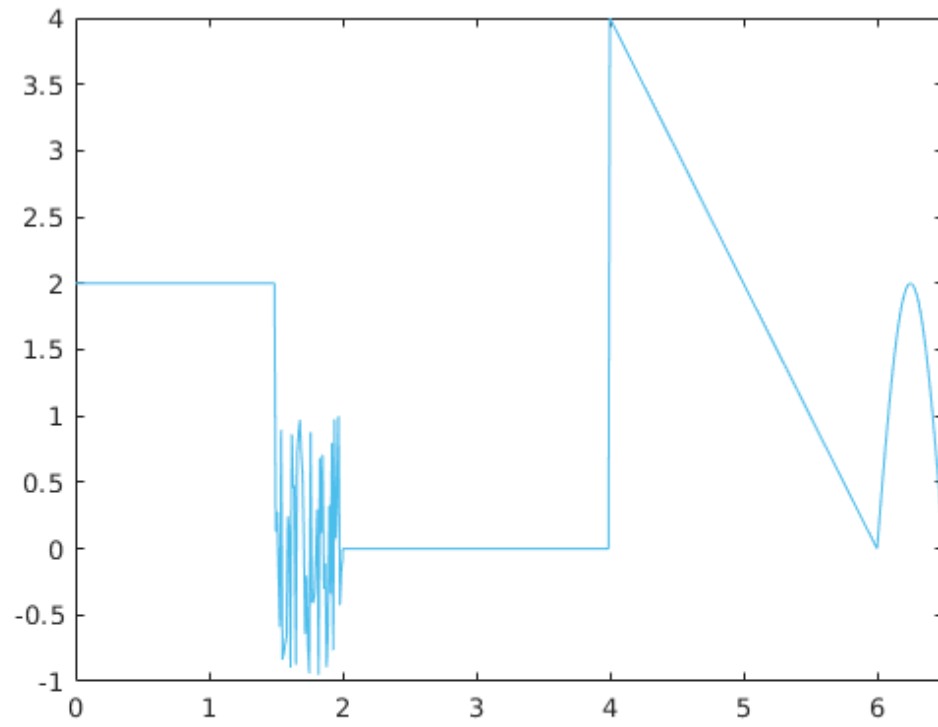
% Noise signal
% t_noise = 1.5:0.01:2;
y_noise = 2*rand(size(t_noise))-1;
plot(t_noise,y_noise)

% We will join the signal together
t = [t_const,t_noise,t_zero,t_saw,t_sin];
y = [y_const,y_noise,y_zero,y_saw,y_sin];
if nargin == 0
    plot(t,y)
    y=[];
end

% Today I learned how to create my own signal graph.
% And the test remind me of commands difference between matlab and
% other programming languages
% (C, Python etc.).

ans =
```

[]



Published with MATLAB® R2018a