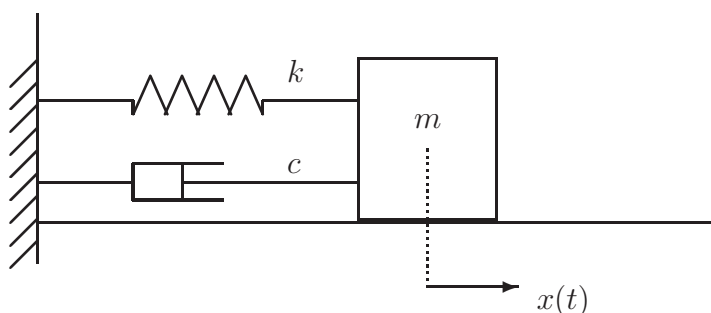


Visualisering av dämpad svängande kloss

En kloss med massan m kan röra sig friktionsfritt längs en horisontell linje. Klossen är fäst i en fjäder med fjäderkonstanten k och en stötdämpare, vars dämpning är proportionell mot hastigheten och motriktad rörelsen.



Om $x(t)$ är avvikelsen från jämviktsläget vid tiden t så ger Newtons andra lag ekvationen

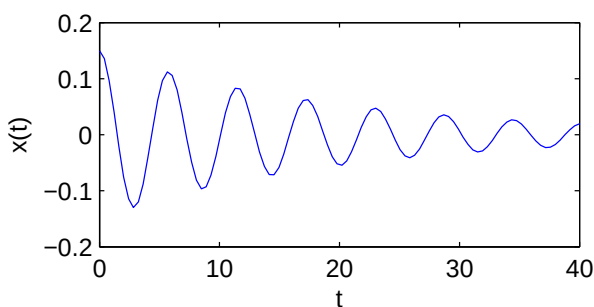
$$mx'' = -kx - cx'$$

På samma sätt som i exempel 5 i "Tillämpningar i matematik med MATLAB" skriver vi om ekvationen som ett system av första ordningen. Vi löser sedan med `ode45` enligt

```
f=@(t,u,k,m,c)[u(2);-k*u(1)/m-c/m*u(2)];  
m=0.1; k=0.12; c=0.01; x0=0.15; v0=0;  
tspan=linspace(0,40,100); u0=[x0;v0];  
[t,U]=ode45(@(t,u)f(t,u,k,m,c),tspan,u0);
```

och ritar upp grafen av $x(t)$ med

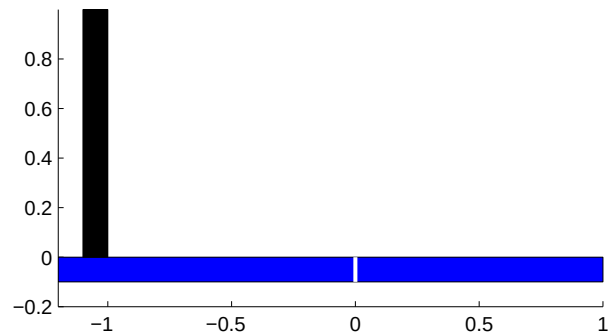
```
plot(t,U(:,1))  
xlabel('t'), ylabel('x(t)'), axis([0 40 -1 1])
```



Nu vet vi hur klossen skall svänga och kan anpassa måttsättningen därefter.

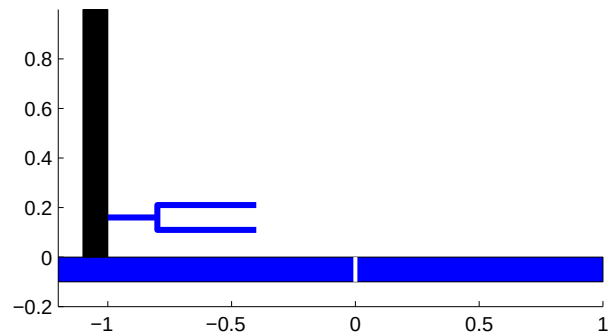
Vi ritar vägg och bord (med ett vitt streck vid neutralläget) med

```
clf, axis equal, axis([-1.2 1.0 -0.2 1])
hold on
fill([-1.2 1.0 1.0 -1.2],[-0.1 -0.1 0 0],'b')
fill([-1.1 -1 -1 -1.1],[0 0 1 1],'k')
plot([0 0],[0 -0.1],'w','linewidth',2)
```



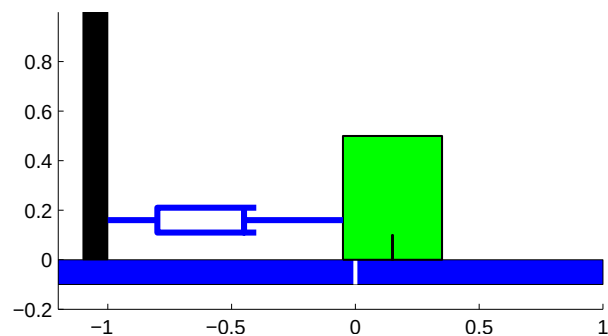
och den fasta delen av dämparen

```
b=0.08; d=0.05;
plot([-1 -0.8],2*b+[0 0],'b','linewidth',3);
plot([-0.4 -0.8 -0.8 -0.4],2*b+[-d -d d d],'b','linewidth',3);
```



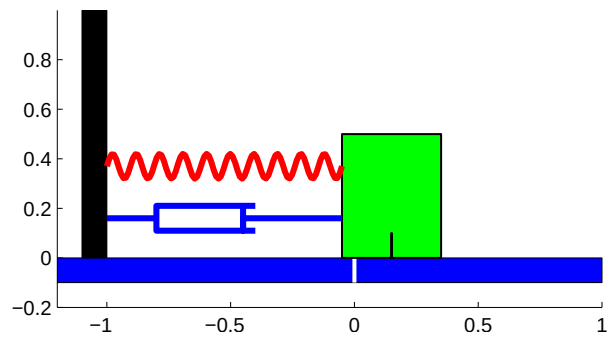
Den rörliga delen av dämparen, dvs. kolven, samt klossen utdragen till begynnelseläge x_0 .

```
kloss_x=[-0.2 0 0 0 0.2 0.2 -0.2]; kloss_y=[0 0 0.1 0 0 0.5 0.5];
kolv_x=[-0.2 -0.6 -0.6 -0.6]; kolv_y=2*b+[0 0 -d d];
kloss_h=fill(x0+kloss_x,kloss_y,'g','linewidth',1);
kolv_h=plot(x0+kolv_x,kolv_y,'b','linewidth',3);
```



Dags för den utdragna fjädern.

```
s=linspace(0,1); a=0.8+x0;
spring_h=plot(-1+a*s,4*b+d*(1+sin(20*pi*s)),'r','linewidth',3);
```



Sedan följer animeringen

```
for j=1:length(t)
    set(kloss_h,'xdata',kloss_x+U(j,1))
    a=0.8+U(j,1); set(spring_h,'xdata',-1+a*s);
    set(kolv_h,'xdata',kolv_x+U(j,1));
    pause(0.1)
end
hold off
```