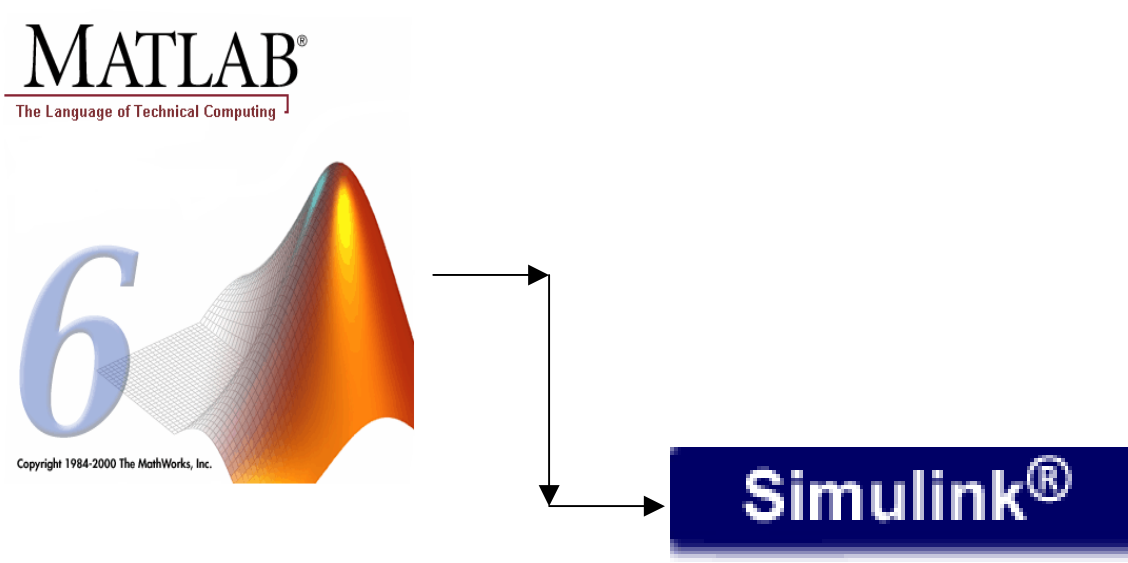




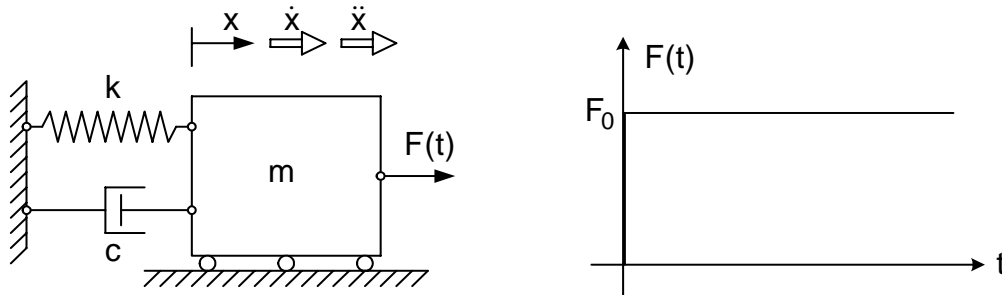
# SIMULINK 5.0 Harjoitus

© Matti Lähteenmäki  
2004  
[www.tpu.fi/~mlahteen/](http://www.tpu.fi/~mlahteen/)



## Harjoitustehtävä 1.

Tarkastellaan kuvan mukaisen yhden vapausasteen jousi-massa-vaimennin systeemin vaakasuuntaista pakkovärähtelyä, kun kuormituksena on hetkellä  $t=0$  vaikuttava askelkuormitus. Systeemin parametrit ovat  $m = 44\text{ kg}$ ,  $k = 1550\text{ N/m}$ ,  $c = 68\text{ Ns/m}$  ja  $F_0 = 88\text{ N}$ . Systeemi lähtee liikkeelle levosta ilman alkunopeutta, ts.  $x(0) = 0$  ja  $\dot{x}(0) = 0$ .



Oheisesta vapaakappalekuvasta saadaan liikeyhtälö soveltamalla Newtonin II lakia

$$\rightarrow m\ddot{x} = -kx - c\dot{x} + F(t)$$

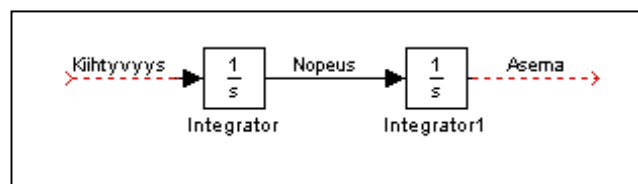
Jakamalla massalla saadaan tulos

$$\ddot{x} = \frac{1}{m}(-kx - c\dot{x} + F(t))$$

Kiihtyvyys  $\ddot{x}$  saadaan siis, kun kaikki kappaleeseen vaikuttavat voimat lasketaan yhteen ja tulos jaetaan massalla. Integroimalla kiihtyvyys saadaan nopeus ja integroimalla nopeus saadaan asema.

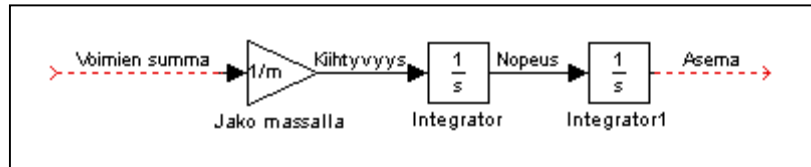
Avataan MATLAB, Simulink ja uusi malli-ikkuna. Mallinnetaan ensin kiihtyvyyden integrointi  $\iint \ddot{x} dt = \int \dot{x} dt = x$ .

Lisää kaksi Integrator lohkoa Continuous kirjastosta ja yhdistä ne signaaliiviivalla. Piirrä vielä viivasegmentit tyhjäksi jäävistä porteista. Nimeä signaalit seuraavan kuvan mukaisesti (Kiihtyvyys, Nopeus, Asema).



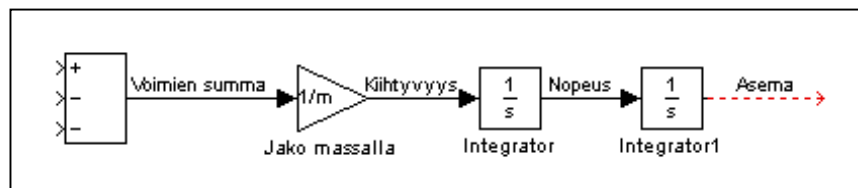
Koska kiihtyvyys on voimien summa jaettuna massalla, on vasemmanpuoleiseen Integrator lohkon sisään menevä signaali jaettava massalla.

Lisää Gain lohko Math Operations kirjastosta ja kytke se Kiihtyvyyssignaaliin. Avaa Gain lohkon parametri-ikkuna kaksoisklikkaamalla lohkoa ja muuta Gain parametrin arvoksi  $1/m$  (muuttujan  $m$  arvo annetaan myöhemmin MATLABin komentoikkunassa). Vedä Gain lohkoa hieman suuremmaksi, jotta parametri  $1/m$  näkyy. Muuta Gain lohkon nimeksi Jako massalla. Piirrä viivasegmentti lohkon sisäänmenoon ja nimeä se (Voimien summa). Malli on nyt seuraavan kuvan mukainen.



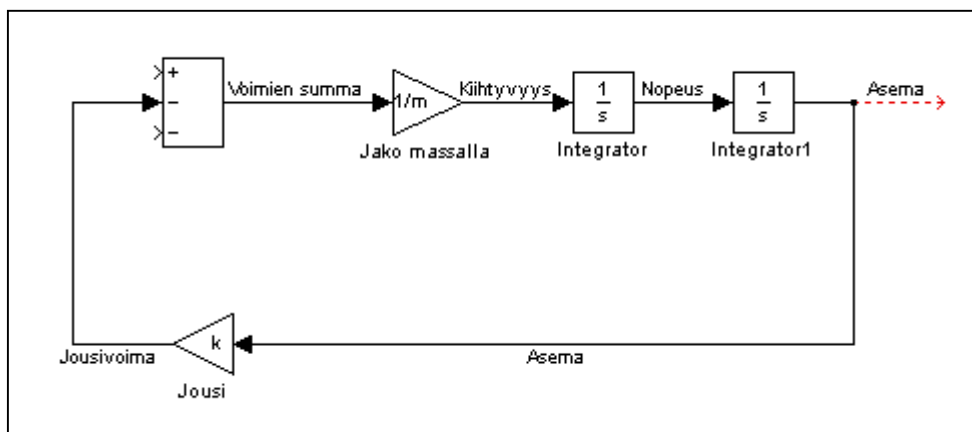
Mallinnetaan seuraavaksi Gain lohkon tulevien voimien summaus.

Lisää Sum lohko Math Operations kirjastosta ja kytke se Voimien summa signaaliin. Muuta Sum lohkon parametri Icon shape arvoon rectangular ja List of signs arvoon  $+-$ . Vedä Sum lohkoa hieman suuremmaksi, jotta porttien etumerkit näkyvät selvemmin. Malli on nyt seuraava.



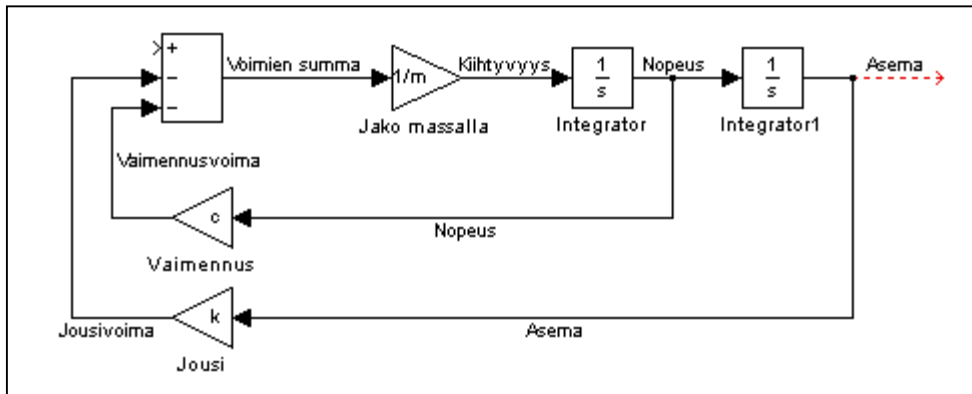
Lisätään sitten malliin jousivoima, joka on  $kx$ , missä  $x$  on asema ja  $k$  jousivakio, joka annetaan myöhemmin MATLABin komentoikkunassa.

Lisää Gain lohko Math Operations kirjastosta ja muuta sen suunta (Flip block Format valikosta). Muuta Gain lohkon nimeksi Jousi ja Gain parametrin arvoksi  $k$ . Haaroita Asema signaali (hiiren oikea näppäin) ja kytke se Jousi lohkon sisäänmenoon. Kytke Jousi lohkon ulostulo Sum lohkon keskimmäiseen porttiin ja nimeä signaali (Jousivoima). Malli näyttää nyt seuraavan kuvan mukaiselta.



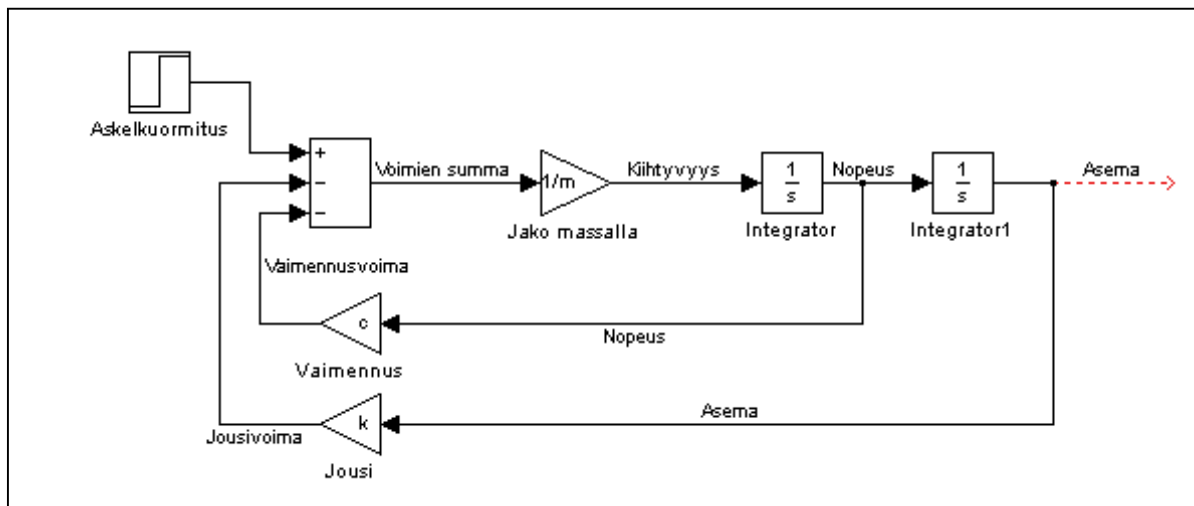
Lisätään seuraavaksi malliin vaimennusvoima, joka on  $c\dot{x}$ , missä  $\dot{x}$  on asema ja  $c$  vaimennusvakio, joka annetaan myöhemmin MATLABin komentoikkunassa.

Lisää Gain lohko Math Operations kirjastosta ja muuta sen suunta. Muuta Gain lohkon nimeksi Vaimennus ja Gain parametrin arvoksi  $c$ . Haaroita Nopeus signaali ja kytke se Vaimennus lohkon sisäänmenoon. Kytke Vaimennus lohkon ulostulo Sum lohkon alimmaiseseen porttiin ja nimeä signaali (Vaimennusvoima). Malli näyttää nyt seuraavalta.



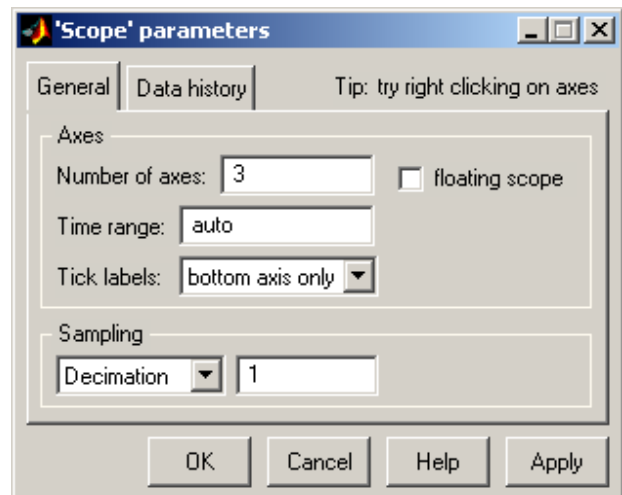
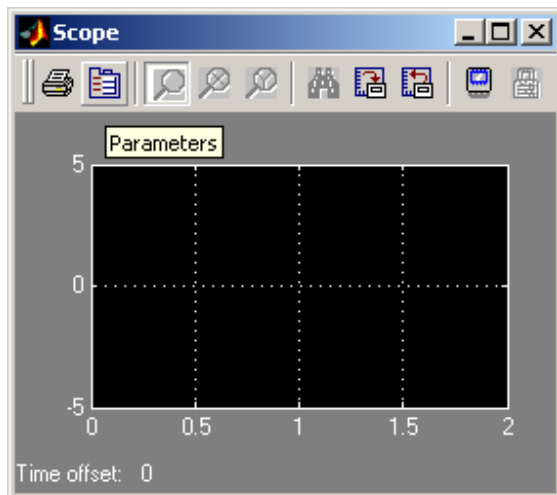
Viimeisenä kuormituksena lisätään malliin ulkoinen voima, joka tässä tapauksessa on askelkuormitus  $F_0$ , joka alkaa vaikuttaa hetkellä  $t=0$ . Parametri  $F_0$  annetaan myöhemmin MATLABin komentoikkunassa.

Lisää Step lohko Sources kirjastosta ja muuta sen parametrin Step time arvoksi 0 ja parametrin Final value arvoksi  $F_0$ . Muuta lohkon nimeksi Askelkuormitus ja kytke lohko Sum lohkon ylimpään porttiin, jolloin malli on seuraavan kuvan mukainen.

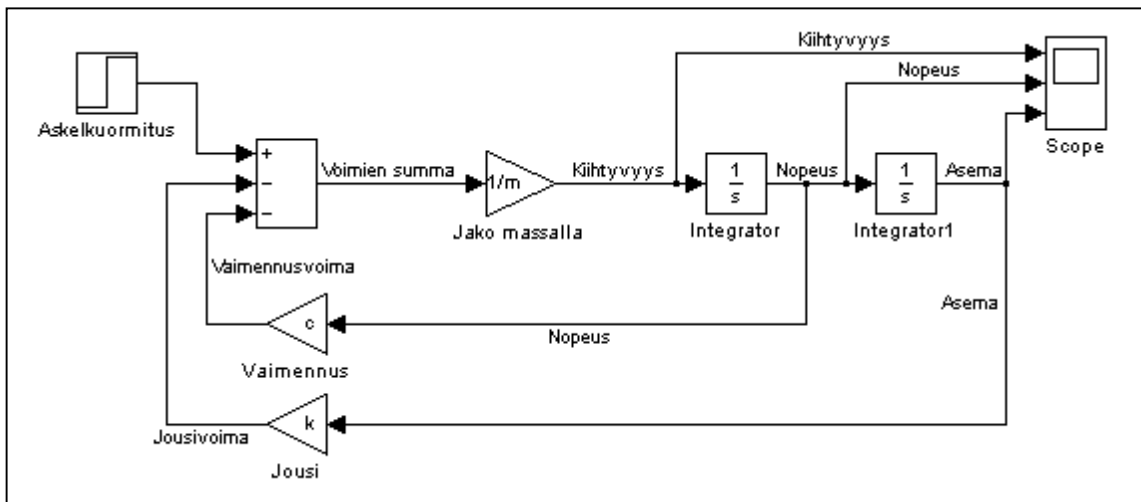


Tulostuksena otetaan ensin aseman, nopeuden ja kiihtyvyyden kuvaajat ajan funktiona Scope lohkon.

Lisää Scope lohko Sinks kirjastosta ja avaa se kaksoisklikkaamalla. Klikkaa työkalurivin Parameters painiketta ja muuta parametrin Number of axes arvoksi 3.



Vedä Scope lohkoa hieman suuremmaksi, jotta sen 3 porttia erottuvat paremmin. Haaroita Kiihtyvyys, Nopeus ja Asema signaalit ja kytke haarat Scope lohkoon. Malli näyttää nyt seuraavan kuvan mukaiselta.

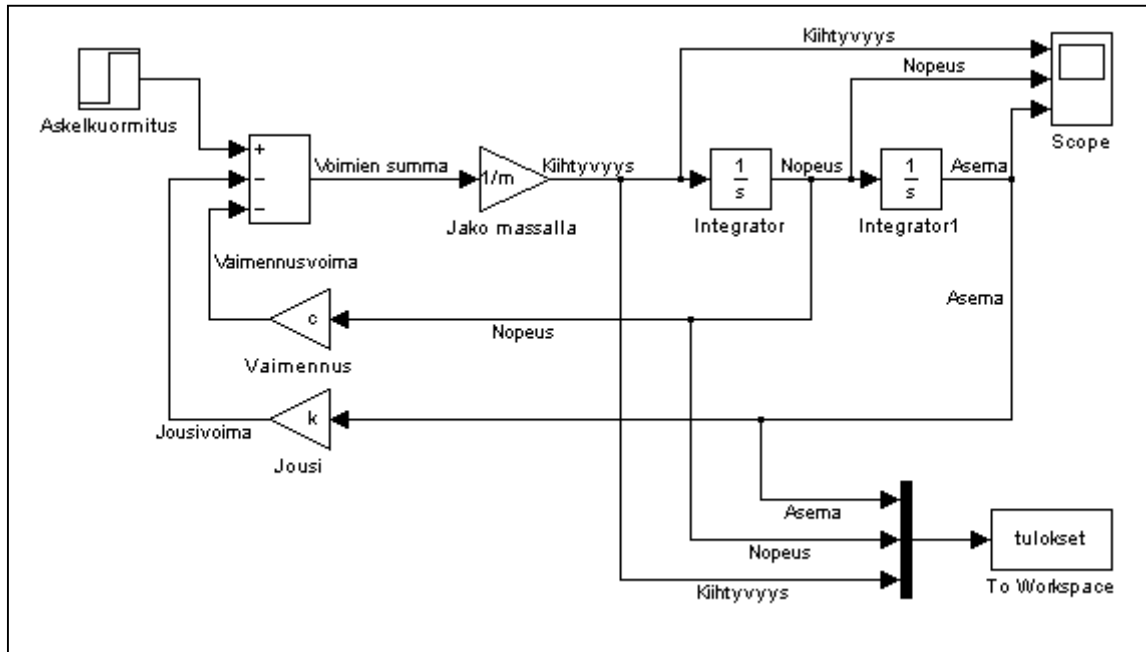


Toisena tulostuksena otetaan aseman, nopeuden ja kiihtyvyyden arvot matriisina MATLABin muuttuja-avaruuteen.

Lisää Mux lohko Signal Routing kirjastosta ja muuta sen parametrin Number of inputs arvoksi 3. Vedä Mux lohkoa hieman suuremmaksi, jotta sen 3 sisäänmenoporttia erottuvat paremmin. Haaroita Kiihtyvyys, Nopeus ja Asema signaalit ja kytke haarat Mux lohkoon. Lisää To Workspace lohko Sinks kirjastosta ja muuta sen parametrin Variable name arvoksi tulokset, parametrin Limit data points to last arvoksi 1000 ja parametrin Save format arvoksi Array. Kytke Mux lohko To Workspace lohkoon. Valmis malli on seuraavan sivun yläreunan kuvassa.

Annetaan systeemin parametrit MATLABin komentoikkunassa (kuva seuraavalla sivulla).

Määritellään simulointiparametrit malli-ikkunan Simulation valikon Simulation parameters keskusteluikkunassa (kuva seuraavalla sivulla). Muuta parametrin Stop time arvoksi 6.0, parametrin Max step time arvoksi 0.006 ja parametrin Relative tolerance ja Absolute tolerance arvoiksi 1e-9.

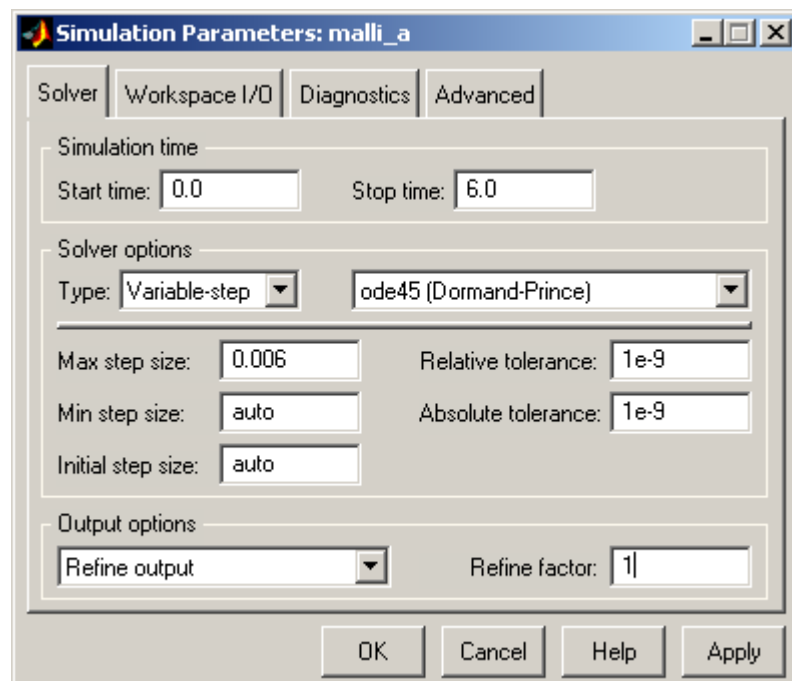


```

Comm...
File Edit View
Web Window
Help

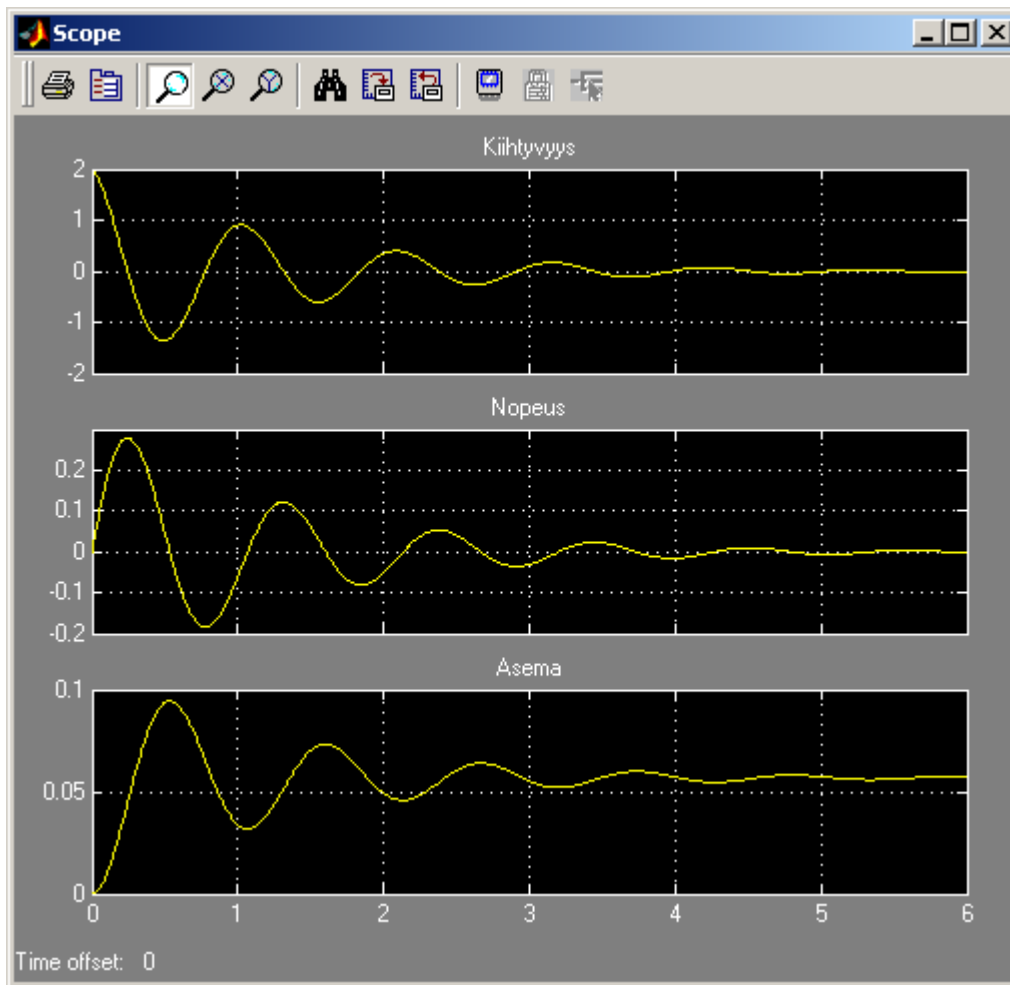
>> m=44;
>> k=1550;
>> c=68;
>> F0=88;
>>

```



Käynnistä simulointi malli-ikkunan työkalurivin Start simulation painikkeesta. Avaa simuloinnin päätyttyä Scope lohko kaksoisklikkaamalla ja vedä tarvittaessa sitä suuremmaksi. Klikkaa Scope lohkon työkalurivillä olevaa Autoscale painiketta (kiikarit), jotta käyrien pystyakselit tulevat sopiviksi. Scope lohkon tulevat käyrät ovat seuraavan sivun yläreunan kuvassa.

Totea, että tulokset ovat MATLABin muuttuja-avaruudessa avaamalla muuttujat tulokset ja tout (näytteenottoajat) taulukkoeditoriin (kuva seuraavalla sivulla).

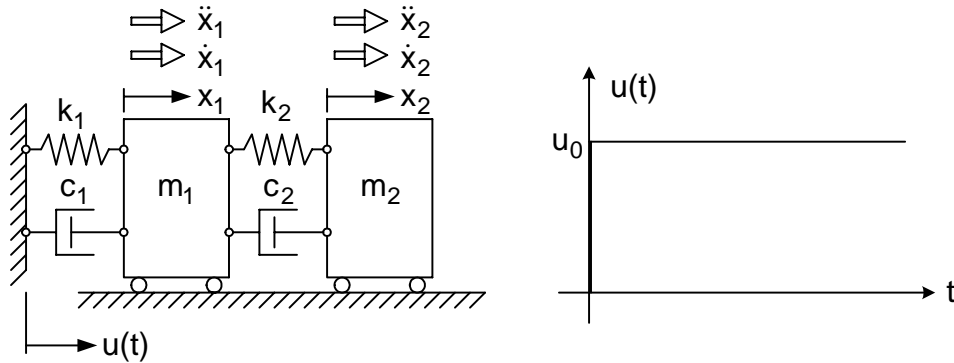


Name	Size	Bytes	Class
FO	1x1	8	double array
c	1x1	8	double array
k	1x1	8	double array
m	1x1	8	double array
tout	1000x1	8000	double array
tulokset	1000x3	24000	double array

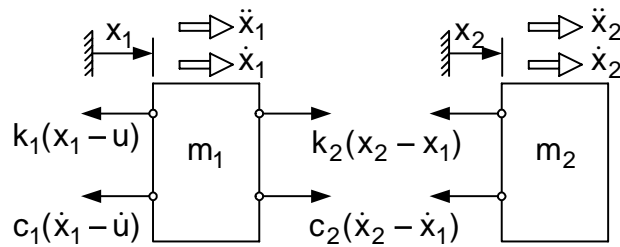
	1	2	3
49	0.058086	0.26736	-0.45941
50	0.059682	0.26445	-0.51112
51	0.061259	0.26123	-0.5617
52	0.062816	0.25771	-0.61111
53	0.064351	0.2539	-0.65929
54	0.065862	0.24981	-0.7062
55	0.067348	0.24543	-0.75178
56	0.068807	0.24079	-0.79599

## Harjoitustehtävä 2.

Tarkastellaan kuvan mukaisen kahden vapausasteen jousi-massa-vaimennin systeemin vaakasuuntaista pakkovärähtelyä, kun kuormituksena on hetkellä  $t=0$  vaikuttava tuen askelsiirtymä. Systeemin parametrit ovat  $m_1=10\text{kg}$ ,  $m_2=5\text{kg}$ ,  $k_1=3000\text{N/m}$ ,  $k_2=2000\text{N/m}$ ,  $c_1=55\text{Ns/m}$ ,  $c_2=33\text{Ns/m}$  ja  $u_0=0,005\text{m}$ . Systeemi lähtee liikkeelle levosta ilman alkunopeutta, ts.  $x_1(0)=0$ ,  $x_2(0)=0$ ,  $\dot{x}_1(0)=0$  ja  $\dot{x}_2(0)=0$ .



Oheisesta vapaakappalekuvista saadaan liikeyhtälöt soveltamalla Newtonin II lakia



$$\rightarrow m_1 \ddot{x}_1 = -k_1(x_1 - u) + k_2(x_2 - x_1) - c_1(\dot{x}_1 - \dot{u}) + c_2(\dot{x}_2 - \dot{x}_1)$$

$$\rightarrow m_2 \ddot{x}_2 = -k_2(x_2 - x_1) - c_2(\dot{x}_2 - \dot{x}_1)$$

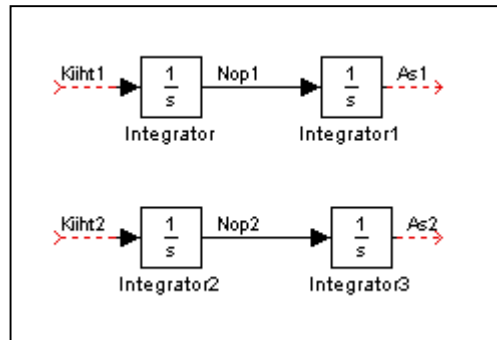
Jakamalla massoilla saadaan tulokset

$$\ddot{x}_1 = \frac{1}{m_1} [-k_1(x_1 - u) + k_2(x_2 - x_1) - c_1(\dot{x}_1 - \dot{u}) + c_2(\dot{x}_2 - \dot{x}_1)]$$

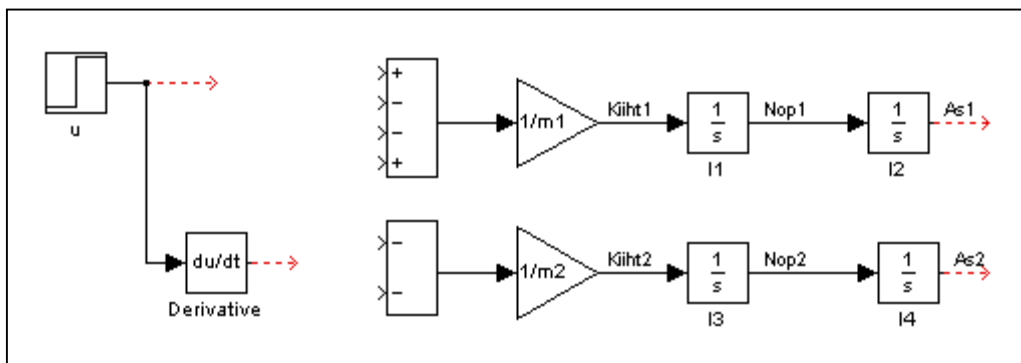
$$\ddot{x}_2 = \frac{1}{m_2} [-k_2(x_2 - x_1) - c_2(\dot{x}_2 - \dot{x}_1)]$$

Avaa MATLAB, Simulink ja uusi malli-ikkuna. Mallinna ensin kiihtyvyyksien integrointi  $\iint \ddot{x}_1 dt = \int \dot{x}_1 dt = x_1$  ja  $\iiint \ddot{x}_2 dt = \int \dot{x}_2 dt = x_2$  seuraavan kuvan mukaisesti.

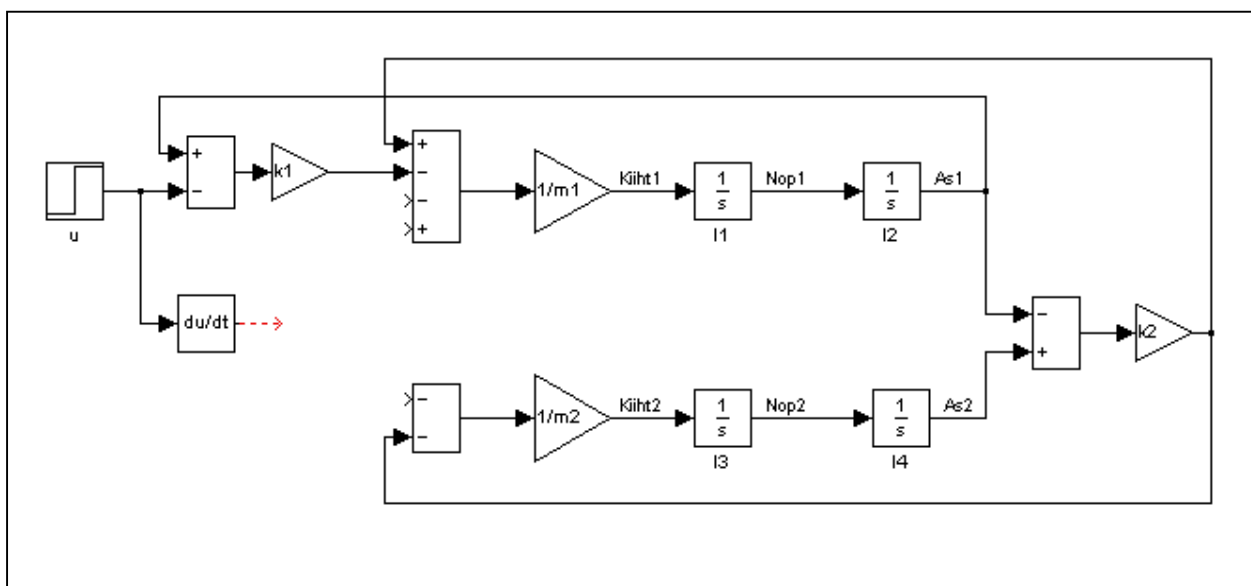




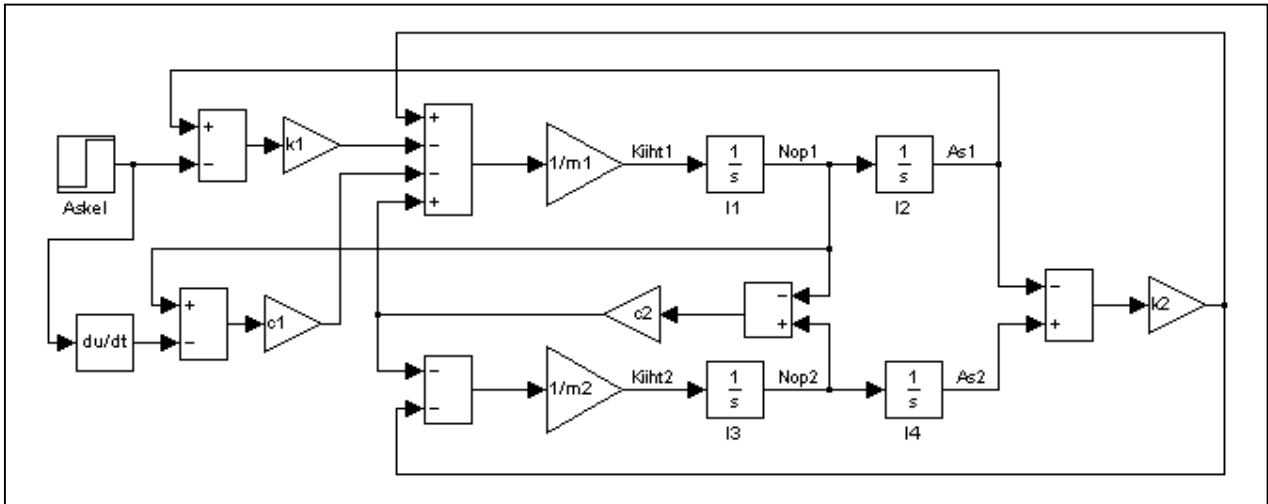
Lisää massoilla jakamiset, voimien summaukset sekä askelkuormitus ja sen derivointi. Derivative lohko on Continuous kirjastossa. Muuta Step lohkon parametrin Step time arvoksi 0 ja parametrin Final value arvoksi u0. Piilota Gain lohkojen nimet, lyhennä Integrator lohkojen nimiä ja anna Step lohkolle nimi u. Malli on tässä vaiheessa seuraavan kuvan mukainen. Piilota mallista vielä Derivative lohkon nimi.



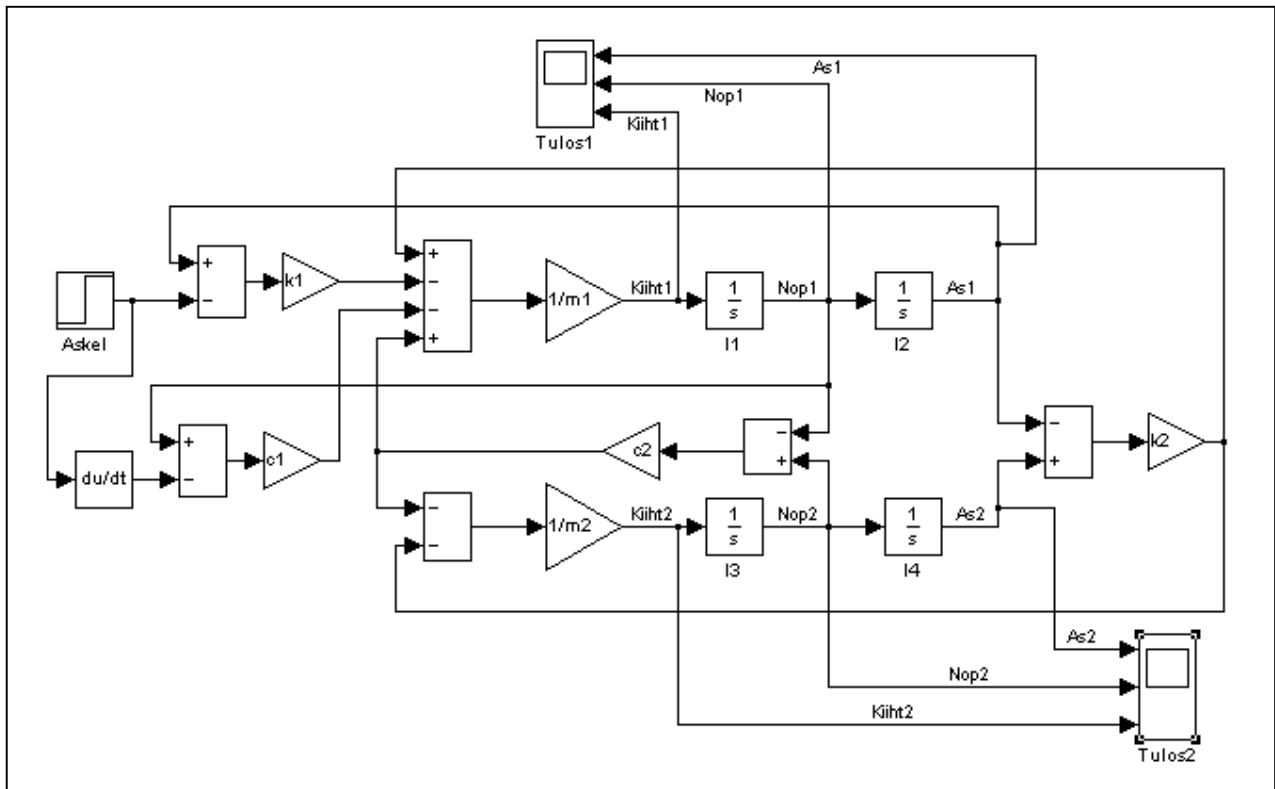
Lisää jousivoimien mallintamiseen tarvittavat Gain lohkot ( $k_1$  ja  $k_2$ ) sekä jousien pituuden muutokset laskevat Sum lohkot ja tee tarvittavat kytkennät lohkojen välille seuraavan kuvan mukaisesti.



Lisää vaimennus voimien mallintamiseen tarvittavat Gain lohkot (c1 ja c2) sekä vaimentimien kohdalla olevat nopeudet laskevat Sum lohkot ja tee tarvittavat kytkennät lohkojen välille seuraavan kuvan mukaisesti.



Ota simuloinnin tuloksina asemien, nopeuksien ja kiihtyvyyksien kuvaajat kahteen Scope lohkoon (Tulos1 ja Tulos2) seuraavassa kuvassa esitetyllä tavalla.



Anna malliin liittyvät parametrien arvot MATLABin komentoikkunassa ja määrittele simulointiparametrit Simulation valikosta seuraavan kuvan mukaisesti.

```

Command Window
File Edit View Web Window
Help
>>
>>
>> m1=10; m2=5;
>> k1=3000; k2=2000;
>> c1=55; c2=33;
>> u0=0.005;
>> |

```

Simulation Parameters: malli\_b

Solver Workspace I/O Diagnostics Advanced

Simulation time  
Start time: 0.0 Stop time: 4.0

Solver options  
Type: Variable-step ode45 (Dormand-Prince)

Max step size: 0.005 Relative tolerance: 1e-9  
Min step size: auto Absolute tolerance: 1e-9  
Initial step size: auto

Output options  
Refine output Refine factor: 1

OK Cancel Help Apply

Suorita simulointi ja avaa Scope lohkot. Tulokset näyttävät seuraavilta (suurena ja skaalaa Scope lohkoja tarvittaessa).

