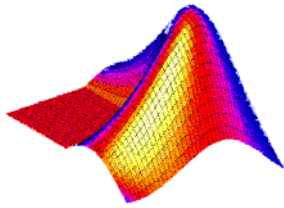


MATLAB / SIMULINK / STATEFLOW/POWER SYSTEM BLOCKSET

ЭХИС-ийн электроникийн системийн баг
Профессор: Проф. Доктор Ж. Дашдорж

Агуулга

- MATLAB гэж юу вэ ?
- MATLAB-ын бүтэц
- MATLAB-ыг ажиллуулах
- Вектор
- Матриц
- Скрипт (Skript) ба функцүүд
- Програмын ажиллагааг удирдах
- График
- Операторууд

1 MATLAB гэж юу вэ?

MATLAB = MATrix LABoratory

Математик- техникт хэрэглэдэг программ хангамж

LINPACK/EISPACK-хэмээх дэд програмын багцаас үүссэн

(Зохиогч: The Mathworks Inc., Natick, MA, <http://www.mathworks.com>)

Matlab нь доорхи зориулалтаар хэрэглэгддэг интерактив ба программчлагдах хэрэгсэл юм. Үүнд:

- Тоон математик
- Вектор ба матрицийн тооцоо
- График

Өргөн хэрэглэдэг салбаруудыг тоочвол:

- Өгөгдлийн шинжилгээ
- Загварчлах
- Симуляци хийх (Simulation)
- Сигнал ба дүрс боловсруулах
- Программ хангамжийн үнэлгээ
- Шугаман алгебрийн сургалт

Matlab – ын бүтэц

Зураг 1.1 –ээс үзвэл Matlab -ын гол цөмийг “ Matlab Command Window” (Matlab-ын командын цонх), интерпретер (Interpreter), “Command Line Editor” үүсгэдэг. Интерпретер нь бэлэн байгаа тухайгаа тэмдгээр мэдээлдэг бол “Command Line Editor” нь товчлуур гараас мэдээлэл оруулах боломж олгодог.

Matlab Release 12- оос эхлэн дурын өөрчлөн тохируулах “Command Window” -той болсон билээ.

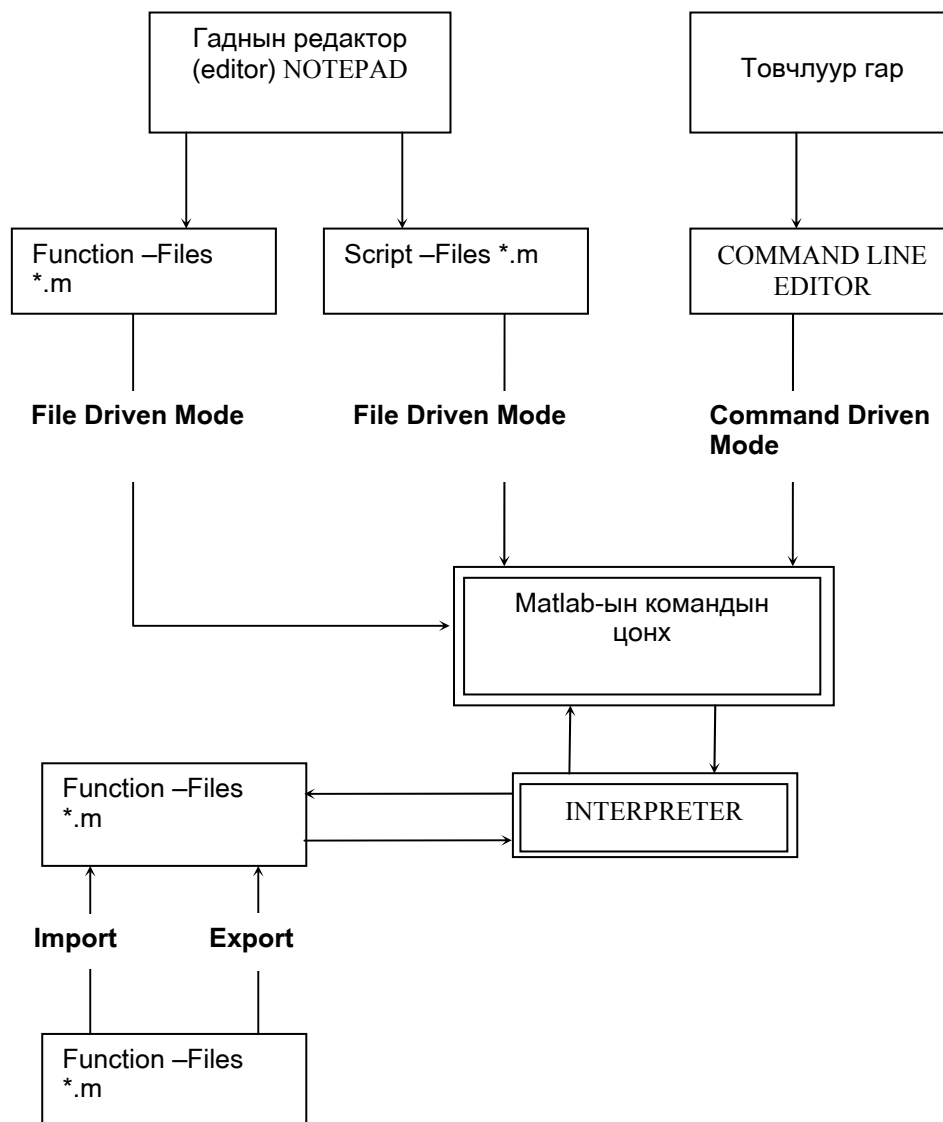
Matlab нь интерпретерийн хэл бөгөөд ажлын хоёр янзын горимтой. Үүнд:

1. “Command Driven Mode”

“Command Driven Mode” – оор “Matlab Command Window” –д зааврыг (комманд буюу оператор) оруулж , Return товчлуураар дуусгахад заавар шууд биелэгддэг.

2. “File Driven Mode”

Гаднын редакторлогч программаар зааврыг оруулах ба уг программд нэр өгч *.m өргөтгөлтэйгээр хадгалдаг. Хэрэв “Matlab Command Window”-д уг нэрийг бичиж , Return товчийг дарвал программ ажиллаж , заавар буюу операторууд биелэгддэг. Интерпретер нь санах ойн хэсэг болох “Workspace” -той харилцаж ажиллах ба энд бүх хувьсахуудыг хадгалдаг. Иймд “Workspace” -ийн өгөгдлийг гаднын файлд *.mat өргөтгөлтэй эсвэл ASCII форматын файл мэт хадгалж, мөн эндээс дуудаж хэрэглэх боломжтой.



Зураг 1.1 Matlab-ын бүтэц

Matlab- ыг ажиллуулах

Дисплейний дэлгэц дээр байгаа MATLAB нэртэй дүрсийг хоёр дахин дарах буюу эсвэл Start -> Programs-> Matlab менюгээр программыг ажиллуулна.

MATLAB- Prompt -ын (>>) дараа **intro** команд бичиж MATLAB -ын тухай тайлбарыг гаргаж хялбархан сурч болох юм. Үүнээс гадна янз бүрийн салбарт хэрэглэх боломжийг танилцуулсан Demo-г (үзүүлбэр) ашиглавал улам хялбар сурах боломж гардаг.

График ба дүрс гаргах Demos:

- *penny*
- *lorenz*
- *bucky*

Өгөгдлийн шинжилгээ хийх Demos:

- *quake*
- *census*
- *fitdemo*

Matlab нь программчлалын хэл болох тухай Demos:

- *xplang*

Үзүүлбэр программуудыг шууд дуудаж ажиллуулж болох ба харин текст болон график цонхууд бие биенээ хаахгүй байлгах хэрэгтэй. Үүнийг жижиг мониторын дэлгэцэнд гаргаж (< 17 инч) харуулахад төвөгтэй байдаг.

demo ашиглаж бүх үзүүлбэр программын жагсаалтыг гаргаж ажиллуулж болдог.

Matlab - д программ зохиохыг ойлгохын тулд программ хангамжтай хамт нийлүүлсэн M- файлуудыг (.m өргөтгөлтэй программ ба функц) гаргаж болдог.

Үүний тулд type <команд> ашигладаг. Жишээлбэл type mean гэж бичиж файлыг дэлгэцэнд гаргадаг.

Заавар:

- Тусламж зааврыг *help Funktion* гэж бичиж гаргадаг.
- Хувьсах хэмжигдэхүүнийг том ба жижиг үсгээр бичих нь ялгаатай юм. *M* ба *m* гэж бичвэл хоёр өөр хувьсах болдог.!!
- Matlab -д хэрэглэдэг чухал өгөгдлийн төрөл нь матриц юм. (Нэг тоо байвал 1x1 хэмжээст матриц, вектор нь 1xn буюу nx1 хэмжээст матриц юм) Матрицийн элемент нь Matlab-д хэрэглэдэг илэрхийлэл байж болно. Жишээ нь sqrt(3) буюу 1+2*4, 1+2i (Энд i нь комплекс тоо sqrt(-1) юм).

Вектор

Prompt ">>" –ын дараа

```
» a=[1 2 3 4 5 6 7]
```

гэж оруулахад Matlab нь доорхи өгөгдлийг гаргана.

```
a =
```

```
1 2 3 4 5 6 7
```

Олон элементтэй векторт өгөгдлийг дэлгэцэнд гаргахыг хориглож болдог. Үүний тулд мөрийн төгсгөлд цэгтэй таслал (;) бичдэг.

```
» a=[1 2 3 4 5 6 7];
```

Векторын хооронд үйлдэл хийх хялбархан юм. Үүнд:

```
» b = a + 5
```

```
b =
```

```
6 7 8 9 10 11 12
```

a, b векторуудын хэмжээс ижил тул тэдгээрийг хооронд нь нэмэх

```
» c = a + b
```

```
c =
```

```
7 9 11 13 15 17 19
```

буюу тэдгээрийн харгалзах элементүүдийн хооронд үржих үйлдэл (Array product) хийж болдог.

```
» d = a .* b
```

```
d =
```

```
6 14 24 36 50 66 84
```

(Үйлдлийн тэмдэгийн өмнөх цэг нь Matlab-д элемент элементээр нь үйлдэл гүйцэтгэхийг заадаг)

Аливаа векторыг хөрвүүлж болдог.

```
» c = c'
```

```
c =
```

```
7
```

```
9
```

```
11
```

```
13
```

```
15
```

```
17
```

```
19
```

Хоёр векторын скаляр үржвэр нь:

```
» a*c
```

```
ans =
```

```
420
```

гэж гарна. **ans** (answer) гэсэн хувьсах нь ямар нэг өөр хувьсахад олгохыг заагаагүй бол программаас автоматоор гардаг. Жишээ нь:

```
» sin(pi/2)
```

```
ans =
```

```
1
```

ans -ыг цааш нь тооцоонд хэрэглэвэл:

```
» ans*2
```

```
ans =
```

```
2
```

Хувьсахуудыг санах ойд (Workspace) хадгалдаг ба тэдгээрийг *clear* командаар арилгадаг. Санах ойд хадгалсан (Workspace) хувьсахуудыг *who* буюу дэлгэрэнгүй харах бол *whos* командаар дэлгэцэнд гаргаж хардаг:

```
» who
```

```
Your variables are:
```

```
a ans b c d
```

```
»clear ans d
```

```
»who
```

```
Your variables are:
```

```
a b c
```

```
»whos
```

```
Name Size Bytes Class
```

```
a 1x7 56 double array
```

```
b 1x7 56 double array
```

```
c 7x1 56 double array
```

Тодорхой дэс дарааллаар үүсэх элементтэй векторыг доорхи байдлаар үүсгэж болдог:

```
» f=0:10
```

```
f =
```

```
0 1 2 3 4 5 6 7 8 9 10
```

```
» g=10:2:20
```

```
g =
    10 12 14 16 18 20
```

Векторын шаардлагатай элементийг индексээр нь тодорхойлж олдог:

```
» f(5)
ans =
     4
```

```
» f(2:4)
ans =
     1 2 3
```

Мөн тухайн элементийг өөр элементээр орлуулах буюу устгаж болно:

```
» f(5)=100
f =
     0 1 2 3 100 5 6 7 8 9 10
```

```
» f(1:3)=[1 1 1]
f =
     1 1 1 3 100 5 6 7 8 9 10
```

```
» f(4:10)=[];
f =
     1 1 1 10
```

Тэгш өнцөгт хаалт [] ашиглан векторуудыг хооронд нь холбох болон матриц болгож хувиргаж болдог:

```
» k=[1 2 3];
» l=[k k k]
l =
     1 2 3 1 2 3 1 2 3
```

```
» M =[k; k; k]
M =
     1 2 3
     1 2 3
     1 2 3
```

Матриц

Matlab нь матрицийн хооронд тооцоо хийх ба түүнийг хувиргах өргөн их боломжтой юм. Ийм үйлдлүүд нь векторын функцтэй төстэй хэрэглэгддэг. M матрицийн аливаа нэг элемент буюу дэд матрицийг доорхи байдлаар гаргадаг:

```
» M(2,2)
ans =
     2
```

```
» M(2:3,2:3)
ans =
     2 3
     2 3
```

Нэг мөр буюу баганыг бүхлээр нь гаргах бол давхар цэг (:) ашигладаг:

```
» M(2,:)
ans =
     1 2 3
```

Матрицад үйлдэл хийх тусгай функц нь , жишээлбэл:

```
» diag(M)
```

```
ans =
```

```
1
2
3
```

буюу матрицийг урвуулах үйлдэл (инверс):

```
» A=[1 2 3;5 7 6;1 4 6]
```

```
A =
```

```
1 2 3
5 7 6
1 4 6
```

```
» inv(A)
```

```
ans =
```

```
2.0000 0.0000 -1.0000
-2.6667 0.3333 1.0000
1.4444 -0.2222 -0.3333
```

```
» A*inv(A)
```

```
ans =
```

```
1.0000 0.0000 0.0000
0.0000 1.0000 0.0000
0.0000 0.0000 1.0000
```

M матрицийн урвуулсан матрицийг тооцоолно уу?

3 x 3 хэмжээтэй, нормал тархсан санамсаргүй тоон утгатай матрицийг үүсгэхийн тулд доорхи командыг ашигладаг:

```
» randn(3)
```

```
ans =
```

```
-0.5033 -1.0711 0.5333
0.0915 -0.1636 0.4024
-1.0127 -0.4402 -1.3607
```

(Matlab –аас өгөгдөл гаргахыг цаашид харуулахгүй)

Доорхи тэгшитгэлийн системийг бодохын тулд

$$x_1 + x_2 = 2$$

$$4x_1 - 9x_2 = -5$$

коэффициентын матриц

```
» A=[1 1;4 -9]
```

ба системийн баруун талын хэсэг

```
» b=[2;-5]
```

векторыг оруулбал, шийд вектор $[x_1, x_2]^T$ - ийг

```
» A \ b
```

буюу

```
» inv(A) * b гэж тодорхойлно.
```

Матрицийг мөр мөрөөр оруулж болох ба түүнчлэн хувьсах хэмжигдэхүүн, арифметик илэрхийлэл хэрэглэж болно. Гэхдээ эдгээрийн хооронд нэг үсгийн хэмжээтэй хоосон зай үлдээх шаардлагатай.

```
» B = [k(2) 2+3
        1+k(2) sqrt(k(2))]
```

Энэ үед мөн матрицийг матрицаар үүсгэх боломжтой:

```
» C = [A b]
```

Матрицийн хооронд нэмэх, хасах ба үржих үйлдлийг доорхи маягаар хийж болно.

$$\gg C = A + B - A * B$$

Үүнээс гадна ямар нэг матрицийг скаляраар үржих болон скалярыг матрицийн элемент бүр дээр нэмэх / элемент бүрээс скалярыг хасах үйлдэл хийх буюу матрицийг хөрвүүлж болно:

$$\gg 2*A$$

$$\gg A+3$$

$$\gg A'$$

Мөн матрицийн хувьд „Цэг цэгээр үйлдэл хийх“, өөрөөр хэлбэл элемент бүрээр нь үйлдэл гүйцэтгэж болно:

$$\gg C = A .* B$$

Матрицийн элементэд мөн дан дангаар нь хандаж болно:

$$\gg A(1,1) = A(1,2) + 2$$

Matlab – д матрицийн хооронд **хоёр** төрлийн хуваах үйлдэл хийх боломжтой:

1. Зүүн талын хуваах үйлдэл

$$X=A \setminus B$$

нь

$$A*X=B$$

тэгшитгэлийн шийдийг гаргадаг.

A, B – ийн мөрийн тоо ижил байх ёстой ба X нь B – ийн адил баганын тоотой байдаг.

2. Баруун талын хуваах үйлдэл

$$B/A$$

нь

$$X*A=B$$

тэгшитгэлийн шийдийг гаргадаг. Харин A, B –ийн баганын тоо ижил байх ёстой ба X нь B-ийн адил мөрийн тоотой байдаг.

Скрипт (Skript) ба функцүүд

Matlab программ хангамжийн өргөн тархсан нэг шалтгаан нь Matlab ашиглан программ зохиоход хялбар байдагт оршино. Хоёр төрлийн программын бүтэц байдаг:

- **Скрипт**

Энэ нь Matlab-ын командуудаас бүрдэх энгийн мөрүүдтэй юм. Ийм файлыг Editor -оор үүсгэдэг ба ямар нэг нэр өгч, жишээ нь **name.m** нэрээр **m** төрлийн файл мэт хадгалдаг. Хэрэв файлын нэрийг (Жишээ нь name-г **.m** өргөтгөлгүй) оруулж өгвөл Matlab -ын командын дараалал нь Matlab –ын цонхонд ажилладаг. Скрипт нь параметр дамжуулдаггүй. Аливаа **m**-файлын агуулгыг *type name* гэж дэлгэцэнд гаргаж болдог.

- **Функцүүд**

Скриптийн адил үүсгэдэг, хадгалдаг ба дууддаг. Харин функц нь параметр дамжуулах чадвартай. Мөн локал, глобал (ерөнхий) хувьсахууд ба (Version 5-оос дээш) локал дэд программыг үүсгэдэг. **fname** нэр бүхий функцийг **m**-файлыг доорхи байдлаар үүсгэдэг:

- **Функцийг тодорхойлох**

Энд

`function dummy=fname(p1,p2,...)` хэлбэрийн мөрийг бичдэг.

function нь түлхүүр үг юм. **dummy** нь дурын сонгож нэрлэсэн гаралтын хувьсах юм. Энд мөн векторыг хэрэглэж болно. Жишээ нь

[dummy1,dummy2]. Программ ажиллахад гаралтын хувьсахад үр дүнг

олгодог. **fname** нь Matlab – аас дуудах функцийг нэр юм. **p1,p2,..** зэрэг нь (дурын олон) дамжуулах параметр юм. Эдгээрийг **Call by Value** мэт дамжуулдаг. Өөрөөр хэлбэл функцэд ямар нэг өөрчлөлт гарвал функцийг гадна талд үйлчилдэггүй. Гадна талд нь үйлчлэх шаардлага гарвал,

хувьсахыг функцийн дотор ба гадна талд (Matlab-ын цонх) үйлчлэхээр ерөнхий (глобал) зарлах ёстой .

- **H1- мөрүүд**

Ийм тайлбарын (комментар) мөр нь функцийн нэр ба функцийн тухай товч тайлбарыг агуулдаг. Хэрэв Matlab-д **help fname** гэж оруулбал энэ ба дараагийн тайлбар мөрүүдийг харуулдаг. Хэрэв **lookfor**-д тодорхой нэр томъёолол өгч Matlab- д хайлт хийвэл H1- мөрийг хайж олдог.

Жишээлбэл:

```
% FNAME(p1,p2)
```

- **Туслах текст**

Энэ хэсэгт параметрын төрөл буюу алдаа гарах боломж зэргийг тайлбарласан туслах текстийг бичдэг. Ингэж ямар нэг Online маягийн тусламжийг (Help) гаргаж болох юм.

Жишээ нь:

```
% FNAME(p1,p2)
```

```
%      p1,p2 нь ижил тоотой элементтэй
```

```
%      векторууд байх
```

```
%      ёстой.
```

- **Хувьсахыг зарлах**

Ийм зарлалтыг хэрэглэх ёсгүй. Хэрэглэгдэх хувьсахууд нь автоматоор **локал** гэж зохицдог. Ерөнхий хувьсахууд нь функцийн гадна талд (Matlab- ын цонх буюу дуудагч скриптэд) **ба** дотор талд нь **global** түлхүүр үгээр тодорхойлогддог. Зөвхөн энэ үед л хувьсахын өөрчлөлт нь функцийн дотор ба гадна талд үйлчилдэг. **Хэвийн тохиолдолд глобал хувьсах хэрэглэхийг аль болох зайлсхийх хэрэгтэй.** Учир нь том программ дотор ямар функцийг яг ингэж тодорхойлсон тухай баримжаагаа хурдан алддаг.

- **Функцийн их бие**

Функцийн их биед өөрийн нь тооцооны хэсгийг бичдэг. Функцийн тооцооны утга нь дээр өгөгдсөн **dummy** хувьсахад олгогддог. Функцийн их биед Matlab- д хүчин төгөлдөр оператор бүрийг бичиж болно. Ялангуяа бусад өөр функцийг дуудаж болдог.

- **Тайлбар (Комментар)**

Мөн энд тайлбар бичиж , хавсаргаж болно . Тайлбарыг % тэмдэгийн дараа бичдэг.

- **Дэд функц**

Зөвхөн анхны функцийн хувьд ил бусад дэд функцийг хавсаргаж болдог. Жишээлбэл m- файлуудын жишээнүүдийг Matlab- ын каталогт агуулдаг. Тэгвэл энд аливаа шугаман комплекс дүрслэлийг тооцоолох нэгэн функц өгөгдсөн гэж үзье. Үүнд:

```
function W=linear(Z)
```

```
%LINEAR Шугаман функц
```

```
%      LINEAR(Z) нь доорхи параметр бүхий
```

```
%      шугаман функцийг тооцоолдог:
```

```
%      a=1/2*exp(i*pi/3)      Эргэх хэмжээ
```

```
%      b=1+i                  Шилжилт
```

```
%      W=a.*Z +b
```

```
%      Z нь бодит буюу комплекс тоо эсвэл матриц
```

```
%      a,b бодит буюу комплекс параметр
```

```
a=1/2*exp(i*pi/3);
```

```
b=1+i;
```

```
W=a.*Z+b;
```


Программын ажиллагааг удирдах

Matlab программ нь Matlab-ын дурын операторуудыг агуулдаг. Энд зөвхөн программд хэрэглэгддэг хэд хэдэн операторууд байдаг. Эдгээр операторууд нь программын ажиллагааг удирдах зориулалттай тусгай операторууд юм. Үүнд:

if, else, elseif

if- операторын синтакс нь:

if логик нөхцөл 1

Matlab-ын командууд

..

end

Хэрэв логик нөхцөл үнэн байвал Matlab-ын команд биелэгдэх ба үгүй бол энэ команд биелэгдэхгүй. Хэрэв анхны нөхцөл биелэгдэхгүй бол **elseif** гэдэг нөхцөлийг дахин цаашид салаалах боломж болгож хэрэглэдэг. Хэрэв мөн энэ нөхцөл биелэгдэхгүй бол **else** – г хэрэглэж **болох** юм.

if логик нөхцөл 1

Matlab-ын командууд

..

elseif логик нөхцөл 2

Matlab-ын командууд

..

else

Matlab-ын команд

..

end

Жишээлбэл, sigpi гэсэн тэмдэг өөрчлөх функц нь ямар нэг тоо π -ээс их бол 1-ийг, π -тэй тэнцүү бол 0, харин π -ээс бага бол -1 –ийг гаргах ёстой гэвэл:

```
function out=sigpi(x)
```

```
if x>pi
```

```
    out=1;
```

```
elseif x==pi
```

```
    out=0
```

```
else
```

```
    out=-1
```

```
end
```

switch, case

switch- оператор нь доорхи синтакстай:

switch *илэрхийлэл* (скаляр буюу тэмпэгт мөр (String))

case Утга1

Matlab оператор

case Утга 2

Matlab оператор

.

.

.

otherwise

Matlab оператор

end

Дээрхи оператор эхлээд *илэрхийллийн* утгыг тооцоолж гаргадаг. Хэрэв Matlab операторын case – ын утга (Утга1 .. Утга n) илэрхийллийн утгатай таарч байвал салаалалтыг гүйцэтгэдэг. Ямарч тохиолдол таарахгүй байвал

otherwise- оор тэмдэглэсэн салаанд шилждэг. С хэлнээс ялгаатай нь программ биелэгдэх бүрт дараагийн тохиолдолруу шилжих бус, харин switch-бүтцийн төгсгөл рүү салаалдагт оршино. Жишээ нь, -1 –ээс 5 хүртэл өөрчлөгдөх тоон дээр бүхэл тоон хэсэг 1 –ийг нэмж, шатлалтай функц гаргах программын хэсэг нь хязгаараас өөр утга өгөгдвөл NaN (not a number- тоон утга биш) утгыг гаргана.

```
function out=stufе(x)
```

```
switch fix(x)
```

```
case 0
```

```
    out=1;
```

```
case 1
```

```
    out=2;
```

```
case 2
```

```
    out=3;
```

```
case 3
```

```
    out=4;
```

```
case 4
```

```
    out=5;
```

```
case 5
```

```
    out=6;
```

```
otherwise
```

```
    out=NaN;
```

```
end
```

for-давталт

нь Matlab-ын операторыг тодорхой тоотой давтан гүйцэтгэхэд хэрэглэгддэг.

Синтакс нь:

for index=*Анхны утга*:*Өөрчлөгдөх алхам*:*Эцсийн утга*

Matlab-ын операторууд

..

```
end
```

Хэрэв өөрчлөгдөх алхмыг өгөөгүй байвал 1 байна гэж үзнэ. Алхмыг мөн эерэг буюу сөрөг утгатай сонгож болдог. index –ийг давталтанд хэрэглэдэг ба анх эхлэн ажиллахад түүний утга нь анхны утгатай тэнцүү ба алхам бүрийн **дараа** өөрчлөгдөх алхмын хэмжээгээр ихэсдэг (хэрэв эерэг бол). Хэрэв өөрчлөгдөх алхам эерэг бол давталт нь index – ийг эцсийн утгаас хэтэртэл биелэгдэнэ (гэвч үүний дараа дахин биелэгдэхгүй). Жишээ нь:

```
x=1;
```

```
for i=1:1.1:3
```

```
    x=x+i;
```

```
end
```

Энэ давталт нь хоёр удаа биелэгдэнэ. x нь 1 утгатай давталтанд орж, эхний давталтанд 2 ба хоёр дахь давталтанд 4.1 болж ихсэнэ. Энэ үед i нь 2.1 утгатай байна.

Хэрэв өөрчлөгдөх алхам сөрөг бол давталт нь index – ийг эцсийн утгаас бага болтол биелэгдэнэ.

while-давталт

нь нөхцөл 1 (TRUE) утгатай байвал биелэгддэг. Синтакс нь:

```
while Нөхцөл
```

Matlab-оператор

..

```
end
```

Жишээ нь :

```
n=1
while n<10
    disp(n)
    n=n+2;
end
```

Энэ давталт нь дэс дараалсан 1,3,5,7,9 утгыг дэлгэцэнд харуулна.

break

оператор нь while- ба for- давталтын биелэлтийг дуусгана.

Мэдээлэл оруулах ба гаргах

Товчлуур гараас мэдээлэл оруулах ба дэлгэцэнд мэдээлэл гаргахын тулд **input** ба **disp** зааврыг ашиглаж болдог.

Input оператор нь программыг ажиллах явцад тодорхой хувьсагчид утга олгодог. Энэ үед дрограммын ажиллагаа түр зогсох ба товчлуур гараас өгөгдсөн утгыг хувьсахад олгодог.

```
x=input('Parameter x=');
```

disp оператор нь программ ажиллах явцад гарах мэдээллийг дэлгэцэнд гаргадаг.

```
disp('Display Information');
```

График

Matlab – д энгийн ху-тэнхлэг зурахаас эхлэн хөдөлгөөнтэй зураг гаргаж , өгөгдлийг визуальчлах (зураг зурах) өргөн их боломж байдаг.

```
» x=0:0.1:2*pi;
» y=sin(x);
» x=0:0.1:2*pi;
» y1=sin(x);
» y2=cos(x);
» plot(x,y1,x,y2);
» title('Sinus and Cosinus');
» xlabel('X')
» ylabel('Y')
» legend('sin','cos');
```

Мөн зургийг (Legend) хулганы (mouse) тусламжтай дурын байрлалд шилжүүлж болдог.

Хэвийн тархалтын утгатай 3 хэмжээст (3-D-Plot) зургийг гаргахын тулд доорхи операторыг бичдэг:

```
n=5000,plot3(randn(1,n),randn(1,n),randn(1,n),'.')
```

n - ийн утгыг компьютерийн хүчин чадлаас хамаарч их утгатай сонгож болдог. Мөн хэд хэдэн операторыг хооронд нь таслалаар зааглаж нэг мөрөнд бичиж болно .

Matlab- ын тэмдгийг (Logos) зурах утгын матрицийг доорхи байдлаар бичиж оруулдаг:

```
A=membrane;
```

Энэ утгын тусламжтай зарим графикийг гаргах боломжийг үзүүлбэл :

```
» contour(A,15) % 15 шатлалтай шугамын дүрс
» mesh(A) %Сүлжээний дүрслэл
» colormap(bone);surf(A);shading interp;axis off %Бодит фото шиг
» pcolor(A); shading interp; colorbar %Өнгөний кодлолтой хавтгай
» colormap(jet) %Янз бүрийн өнгөний багц: Мөн түүнчлэн
» colormap(copper) %Ж=нь хуурмаг өнгөний дүрслэл
```

PLOT функц

Matlab - ын давуу тал нь сигнал , спектр ба давтамжтай процессыг дүрслэн гаргах

боломж сайтай явдал юм. Эдгээр хэмжигдэхүүний утгууд мөр буюу багана векторын хэлбэрээр байрладаг.

PLOT функц нь зураг ,дүрс гаргах гол заавар юм.

```
plot ([xmin],[ymin],[xmax],[ymax],[x],[y],[‘df’]);
```

Үүнд:

xmin , ymin Зургийн цонхны зүүн доод талын булангийн цэгийн координат

xmax, ymax Зургийн цонхны баруун дээд талын булангийн цэгийн координат

x Абциссыг илэрхийлэх мөрөн вектор

y Босоо тэнхлэгийг илэрхийлэх мөрөн вектор , өөрөөр хэлбэл сигнал юм.

d Сигналын утгыг цэгээр эсвэл шугамаар хооронд нь холбох эсэхийг заах тэмдэг

Шугам: d нь **-/--:/-** .

Цэг: d нь **./+/*/o/x**

f Өнгийг заах тэмдэг: r/g/b/y

Спектр: Фурьегийн хурдан хувиргалт

Фурьегийн хурдан хувиргалт нь (товчлон англиар FFT – Fast Fourier Transformation) Фурьегийн дискрет хувиргалтыг (Discrete Fourier Transformation – DFT) тооцоолох арга юм.Өөрөөр хэлбэл Фурьегийн дискрет хувиргалтын нийлбэрийг олох явдал юм. Фурьегийн дискрет хувиргалт нь N дарааллын элемент бүхий комплекс тоо $x(n)$ -ийн цуваа юм.

$$\underline{y}(m) = \frac{1}{N} \sum_{n=0}^{N-1} \underline{x}(n+1) e^{-j2\pi \frac{m \cdot n}{N}} \quad (1)$$

Энэ хувиргалтыг Matlab –ын **fft** функцээр доорхи байдлаар хоёр хувиргалт хийж хэрэгжүүлдэг. Үүнд:

$$\underline{y}(m+1) = \sum_{n=0}^{N-1} \underline{x}(n+1) e^{-j2\pi \frac{m \cdot n}{N}} \quad (2)$$

Харин энд $1/N$ коэффициент Matlab -ын функцэд оролцоогүй ба энэ нь Фурьегийн инверс дискрет хувиргалтын үед гарч ирдэг. Үүний зэрэгцээ индекслэлтэд шилжилт гардаг. Энэ нь матрицийг индекслэхэд тэгийг хэрэглэдэггүйтэй холбоотой юм. Иймд давтамжийн вектор $y(i)$ –ийн индекс ба үүнтэй холбоотой давтамжийн хооронд доорхи уялдаа гардаг.

$$f' = \frac{i-1}{N} \quad (3)$$

Томъёо (1) ёсоор синусын дараалалд шинжилгээ хийхэд спектрийн шулууны өндөрийн хэмжээ нь амплитуд утгын хагастай тэнцүү байдаг. Иймд программчлах үедээ $1/N$, 2 коэффициент ба индексийн шилжилтийг тооцох ёстойг анхаарах хэрэгтэй.

Фурьегийн хурдан хувиргалтын Matlab функц нь:

<Давтамжийн вектор> = **fft** (<Векторын дараалал>, <Дарааллын элементийн тоо>);

Операторууд

Matlab нь тоо (Скаляр) ба матрицийн хувьд биелэгддэг арифметикийн стандарт үйлдэл болон тэдгээрийг гүйцэтгэх дүрмүүдтэй байдаг:

+	Нэмэх ба матриц нэмэх үйлдэл
-	Хасах ба матрицийн хасах үйлдэл
*	Үржих ба матрицийн коммутатив бус үржих үйлдэл
/, \	Хуваах ба матрицийн баруун болон зүүн хуваах үйлдэл
^	Экспонент ба матрицийн экспонент (зэрэгт дэвшүүлэх)
'	Комплекс тоон матрицийн хосмогийг хөрвүүлэх

Матрицийн хооронд элемент, элементээр үйлдэл гүйцэтгэх бол **цэгтэй** үйлдлийг хэрэглэдэг:

.*	Хоёр матрицийн хооронд элемент бүрээр үржих
./, \.	Хоёр матрицийн хооронд элемент бүрээр хуваах үйлдэл хийх
.^	Хоёр матрицийг элемент бүрээр экспонентлэх (зэрэгт дэвшүүлэх)
.'	Комплекс тоон матрицийг хөрвүүлэх (хосмоггүй)

Урьдчилан тодорхойлогдсон функц ба тогтмол

Matlab –д **sqrt** (язгуур), **exp**, **sin** гэх мэт олон янзын элементар функцүүд байдаг.

Matlab – ын элементар функцийг жагсаалтыг

» *help elfun*

гэж бичиж гаргадаг.

Тусгай функцийг жагсаалтыг (Бессель, алдааны функц г.м.)

» *help specfun*

ба матрицтай ажиллах жич функцийг жагсаалтыг

» *help elmat* гэж тус тус гаргана.

Ихэнх функцийг аргумент нь комплекс тоо ба үр дүн нь мөн комплекс тоо байж болно. Жишээ нь:

» *sqrt(-1)*

Гол төлөв хэрэглэдэг урьдчилан тодорхойлсон тогтмолууд:

pi	3.14159265...
j, i	Хуурмаг хэсэг, <i>sqrt(-1)</i>
eps	2 ⁽⁻⁵²⁾ , нарийвчлал , Matlab- ын функцийг нарийвчлалыг тодорхойлоход хэрэглэдэг . eps – г зааж олгож болох (Жишээ нь: eps=1.e-10) ба арилгаж болно (<i>clear eps</i>).
inf	Хязгааргүй утгатай үр дүн, Хэрэв тоог нойлд хуваавал гардаг.
nan	Математикийн хувьд тодорхойлогдоогүй тоо, Жишээ нь 0/0

2 SIMULINK

Simulink нь динамик системийн загварчлал хийхэд зориулагдсан графикайн хандлагат программ хангамж юм. Matlab-тай Simulink-ийг хамтруулан динамик системийн анализ, загварчлал, дүрслэл зураг гаргах ба симуляци хийх иж бүрэн, цогц ажлын орчинг үүсгэж болдог. Энэ систем нь холбоо мэдээллийн техникийн системийн ажиллагааг симуляцлахад маш тохиромжтой байдаг. Simulink –ийн тусламжтай шугаман, шугаман бус, дискрет ба тасралтгүй, гибрид (холимог) төрлийн системийг загварчилж болдог. Системийн загвар үүсгэхэд гурван янзын боломжийг сонгож болно. Үүнд:

- График –блокийн хандалттай арга
- Matlab-д программчлах
- C буюу FORTRAN программчлал

Энэхүү гарын авлагын зорилго нь Simulink-ийн программын багцыг практикт хэрхэн хэрэглэх тухай тайлбарлах юм.

- Simulink гэж юу вэ? Түүнтэй хэрхэн ажиллах вэ?
- Сигнал боловсруулалтанд Simulink-ийг хэрхэн хэрэглэх вэ?

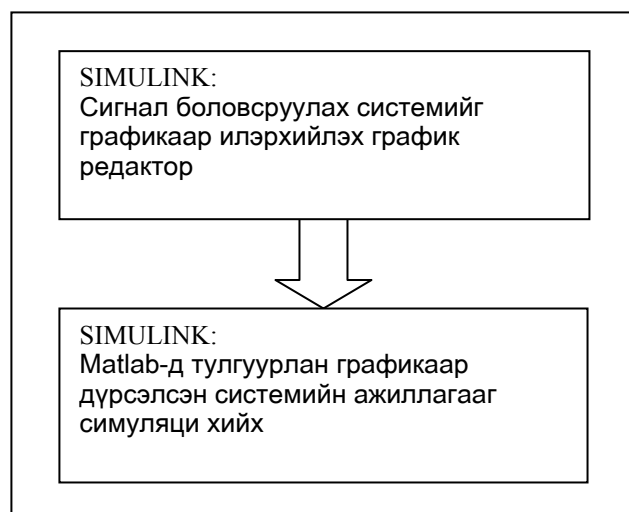
гэсэн асуултанд хариулах хэрэгтэй юм. Нөгөө талаас энэ гарын авлага нь практикт Simulink-ийг хэрхэн ашиглах тухай мэдэгдэхүүн өгөх зорилготой юм.

Бүтэц зохион байгуулалт

Юуны өмнө simulink-ийн статик бүтцийг тайлбарлая. График орчин болох янз бүрийн цонхууд нь чухал үүрэгтэй байдаг тул эдгээрийн тухай эхлээд тайлбарлах юм.

Тойм

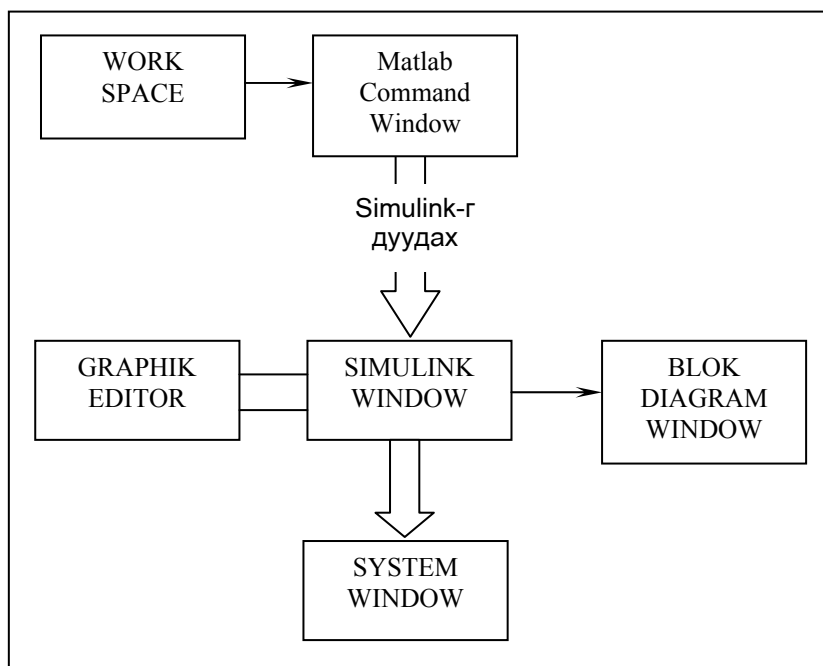
Симулинк нь Matlab-тай холбоо бүхий графикайн редактортой (editor) юм. График редактор нь тоон ба аналог сигнал боловсруулах системийг графикаар дүрслэх боломж олгодог. Matlab-тай холбогддог тул ийм системийг загварчлах бүрэн боломжтой юм. Доорхи зурагт Simulink-тэй ажиллах, ажлын үндсэн хоёр алхамыг үзүүлжээ.



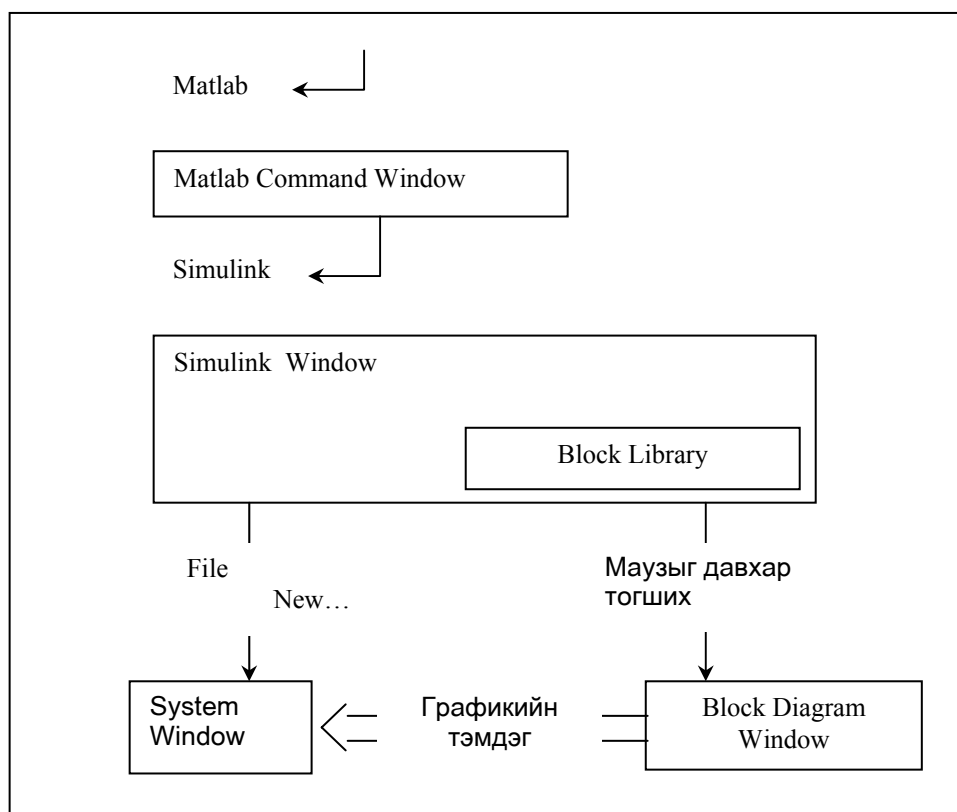
Зураг 2.1 Simulink-ийн ажлын алхам ба хэрэгсэл

График редактороор тодорхойлогдсон системийг Matlab-д тулгуурлан симуляци хийдэг.

Simulink- тэй ажиллахад блок ба цонхууд хэрхэн харилцан ажиллаж байгааг зураг 2.2 -д, цонхууд хэрхэн харилцан холбогдохыг зураг 2.3 -д тус тус үзүүлэв.



Зураг 2.2 Simulink-ийн бүтэц



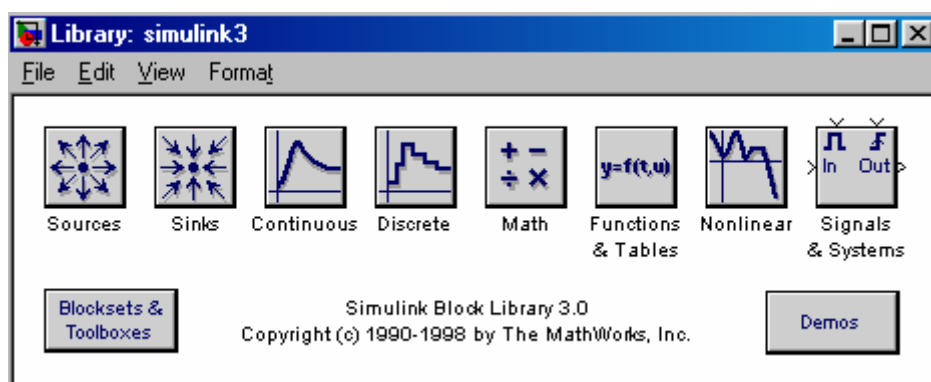
Зураг 2.3 Simulink-ийн цонхны харилцан уялдаа

“Matlab Command Window” –оос программы simulink гэдэг командаар (Release 12-д simulink 3) ажиллуулдаг. Ингэхэд Симулинкийн “Simulink Block Library” –г агуулсан сангийн цонх нээгддэг (Зураг 2.4). Симулинкийн цонхноос өшөө хоёр цонхыг нээх боломжтой. “File/New/Model” менюгээр загварын цонхыг нээдэг. Энэ цонхонд тоон ба аналог сигнал боловсруулах системийн загварчлал явагддаг.

“Simulink Block Library” –ийн блок дээр маузыг давхар тогшиход графикийн тэмдэг бүхий сангийн цонх нээгддэг. Системийг байгуулахын тулд графикийн тэмдэгийг маузаар татаж, загварын цонхонд шилжүүлэх замаар системийг бүхэлд нь үүсгэдэг.

Simulink Window

Simulink-ийн сангийн цонх нь энд нээгддэг бүх цонхны адил File, Edit, View, Format ба Help гэсэн менютэй (цэс) байдаг.

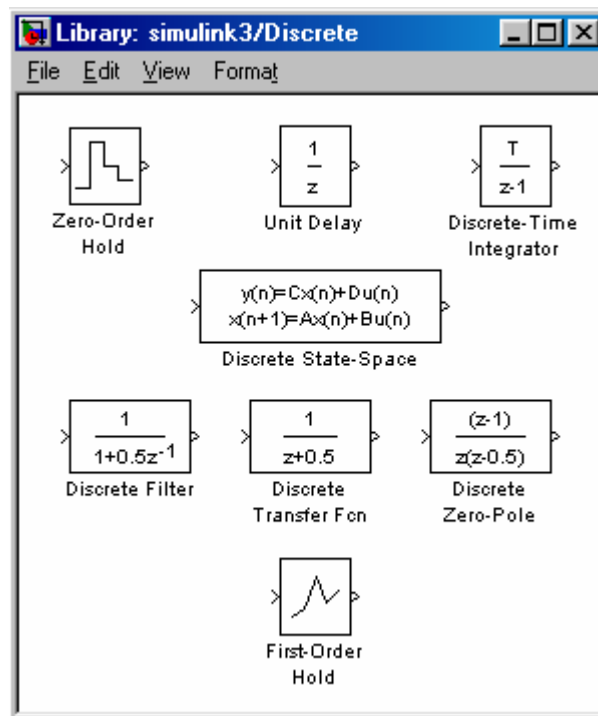


Зураг 2.4 Simulink-ийн сангийн цонх

Бид эхлээд зөвхөн File гэдэг цэсний тухай тайлбарлая. Энэ цэс нь файльтай уялдсан үйлдэл гүйцэтгэх боломж олгодог. Эхлээд New ба Open цэсний тухай үзье. New/Model-оор загварын цонх нээгдэх ба энд шинэ системийг загварчилдаг. Загварчилсаны дараа Save-ийн тусламжтай *.mdl өргөтгөл бүхий нэр өгсөн файлыг хадгалдаг. Open нь үүсгэж зохиосон системийн загварыг дахин дуудаж нээх боломжтой юм.

Сангийн цонх

“Simulink Block Library” –ийн тэмдэг дээр хоёр дахин тогшиж дахин сангийн өөр цонхыг нээдэг. Жишээлбэл, Зураг 2.5-д “Discrete” хэмээх бүлгийн зургийг үзүүлжээ. Энэ цонхны блокод хандаж цаашдын дэд сандаа хандах боломж гардаг юм. Ингэснээр дурын иерархи дараалал үүсгэх боломжтой болдог.

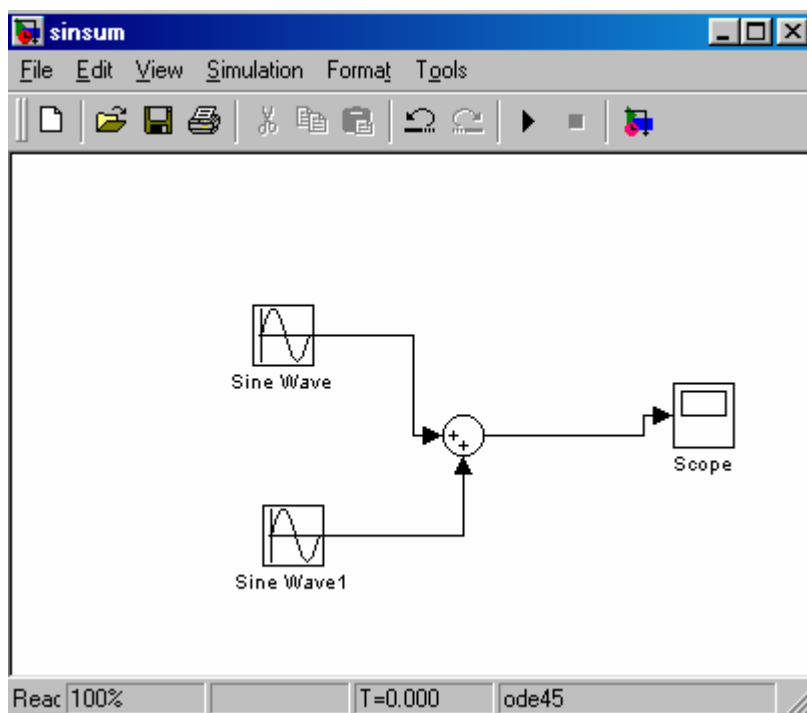


Зураг 2.5 Discrete сангийн цонх

Загварын цонх

Загварын шинэ цонхыг File/New/Model цэсээр нээдэг юм.

Энэ цонхонд системийн загварчлалыг гүйцэтгэдэг. Ингэхдээ сангийн цонхонд байгаа хэсгүүдийг шаардлагатай бол гаргаж, хооронд нь холбох боломжтой юм. Зураг 2.6-д хоёр синус генератор, нэг нийлбэрлэгч ба осциллограф зэргээс бүрдсэн, sinsum.mdl нэрээр хадгалагдсан жижиг системийг үзүүлжээ.



Зураг 2.6 Загварын цонх

Графикийн редактор

Ийм редактор нь (editor) маузын тусламжтай график үйлдлүүдийг гүйцэтгэх боломж олгодог. Үйлдэл гүйцэтгэхэд загварын цонхны Edit цэс чухал үүрэгтэй юм. Энэ цэсэнд Cut, Copy, Paste, Clear ба Select all зэрэг дэд цэс байрладаг. Маузын тусламжтай график тэмдгийг нэг цонхноос нөгөө өөр цонхонд зөөж шилжүүлэх, шугам татаж тэмдэгийг системтэй хооронд нь холбох зэрэг үйлдэл гүйцэтгэж болно. График тэмдэгийн цаана нь Matlab-ын программ байх ба тэмдэгүүдийг шулуунаар холбох нь Matlab-ын программын хооронд хувьсахыг автоматаар шилжүүлэхтэй адил харгалзаа үүсдэг.

Simulink системийг ажиллуулах

Аливаа систем нь файлд *.mdl өргөтгөлтэй хадгалагдсан байх ёстой. Ийм системийг янз бүрийн аргаар нээж болдог. Хэрэв файл нь тухайн каталогт байрлаж байвал файлын нэрийг "Matlab Command Window"-д бичиж нээнэ. Мөн File/Open менюгээр нээж болдог.

Загварчлалыг ажиллуулах

Загварын цонхонд системийн загвар бүрэн хийгдсэн бол Simulation цэс нь симуляци гүйцэтгэх тохируулгыг гаргадаг. Энд гол төлөв Start, Stop, ба "Simulation Parameters....." дэд цэс гарч ирдэг. Загварчлалыг ажиллуулахаас өмнө загварчлалын параметрыг шалгах хэрэгтэй. Start цэсийг дарж загварчлалыг ажиллуулдаг.

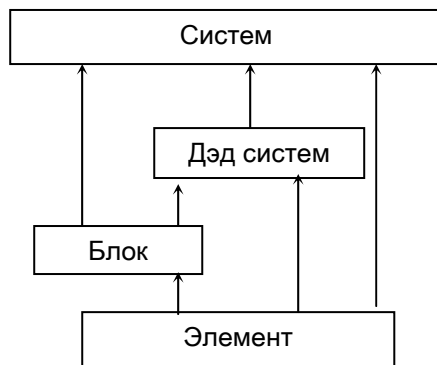
Системийн загварчлал

Аливаа аналог буюу тоон сигнал боловсруулах системийн загварчлалын үндэс нь элемент, блок, дэд систем буюу бүхэл системийг дүрслэх график тэмдгийн сан юм.

Энэ сангууд нь Simulink-ийн чадварыг тодорхойлдог юм. Үүнээс гадна Matlab-ын адил өөрийн үүсгэсэн системээр санг өргөжүүлэх боломж байдаг.

Library блок (Simulink Block Library)

Simulink-ийн сангийн цонх (зураг) дахь сангийн тэмдэгтүүд нь дотроо дэд бүлэг ба блокод хандах боломж агуулсан байдаг. Сигнал боловсруулах систем нь иерархи (дэс дараалах) бүтэцтэй байдаг. Ийм системийн бүтцийг Зураг 2.7 –д үзүүлээ.



Зураг 2.7 Системийн иерархи бүтэц

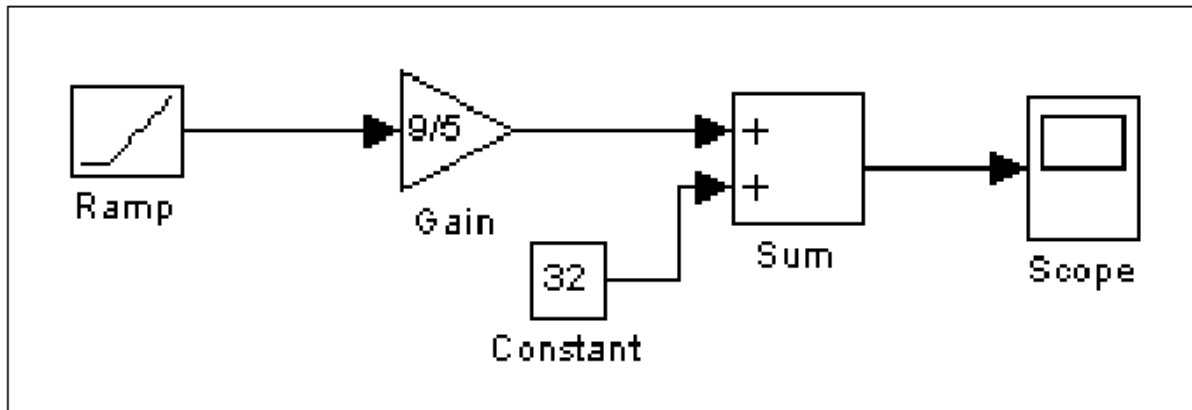
Хүснэгт 2.1 “Simulink Block Library” –ийн өргөн хэрэглэгддэг тэмдгүүд

Бүлэг	Тэмдэг	Түвшин	Тайлбар
Signal sources	Clock	Элемент	Загварчлалын алхамын такт
	Constant	Элемент	Тогтмол утга
	Step input	Элемент	Огцом өсөлттэй дараалал
Signal sinks	Graph	Систем	Осциллограф
	XY- Graph	Систем	XY- Осциллограф
	To Workspace	Хувьсах	Векторын хэлбэрээр хадгалах
Discretete systems	Unit delay	Элемент	Нэг алхамаар саатуулах
	Filter	Блок	Тоон фильтр
Linear system	Sum	Элемент	Нэмэх элемент
	Gain	Элемент	Коэффициент гаргах
Nonlinear system	Product	Элемент	Үржигч
	Saturation	Блок	Ханалттай тодорхойломж
	Quantizer	Блок	Амплитудын утга өгөгч
	Lookup Table	Блок	Хүснэгт (ой)
	Transport delay	Элемент	Саатал (аналог)
	Function	Блок	Скаляр функц бүхий блок
	Matlab – Function	Дэд систем	Матрицын функцтэй блок
Connections	Mux	Блок	Матриц дахь хэд хэдэн скаляр
	Demux	Блок	Mux-ийн эсрэг

Simulink системийг практикт хэрэглэх жишээ, дасгалыг дараагийн хэсгүүдэд тодорхой авч үзэх тул энд зөвхөн хоёр жишээг товчхон үзүүлээ.

Жишээ 1 :

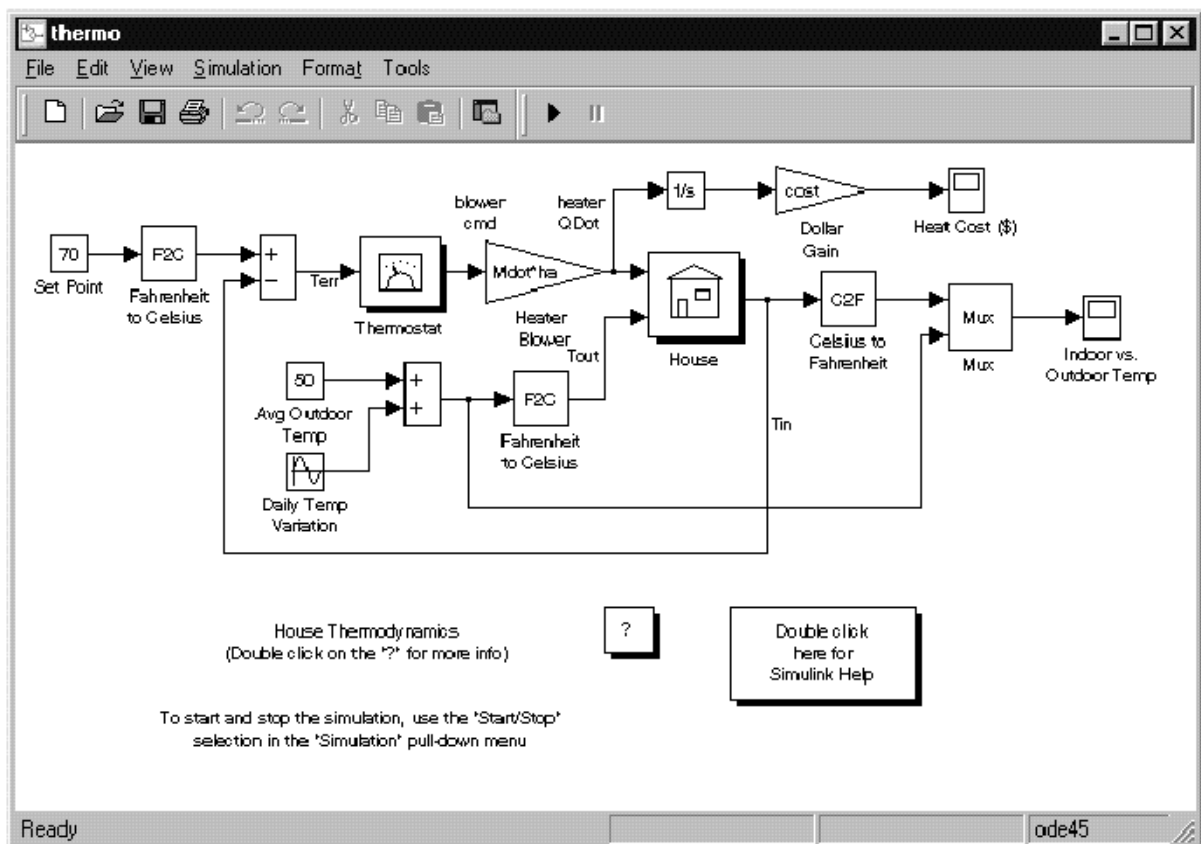
$T_f = 9/5(T_c) + 32$ тэгшитгэлийн блок-схемийг Зураг 2.8 -д үзүүлээ.



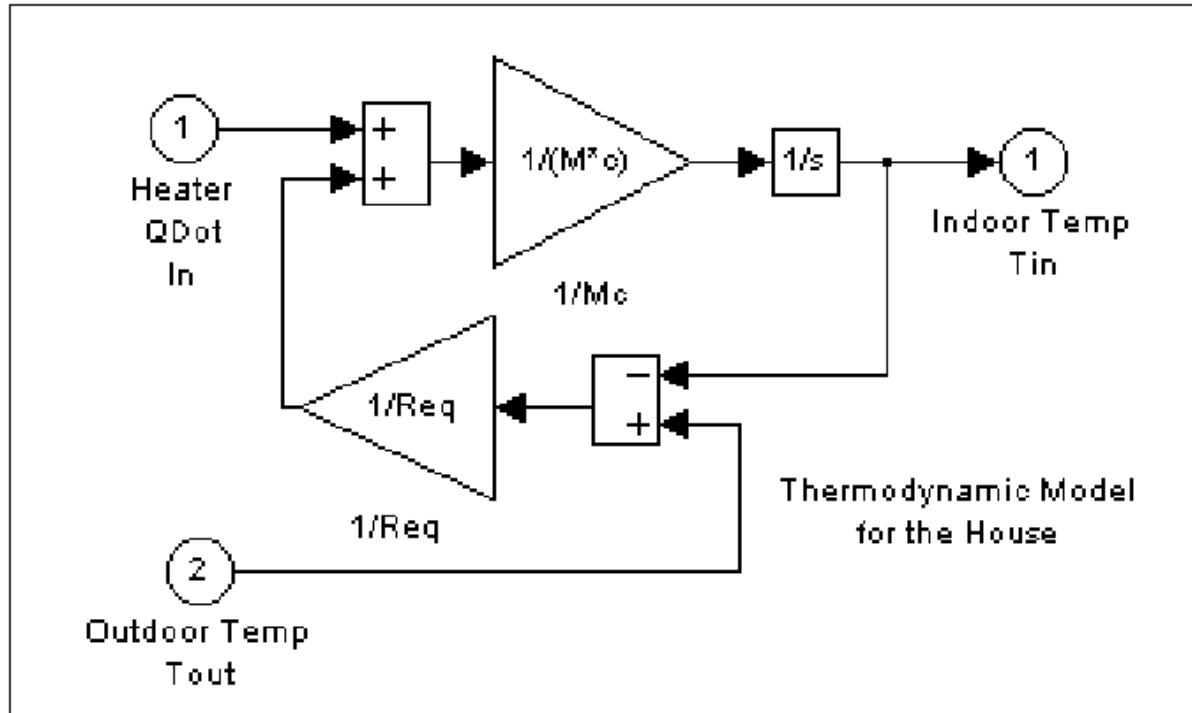
Зураг 2.8 Тэгшитгэлийн блок-схем

Жишээ 2:

Орон сууцны термодинамикийг загварчлах Simulink –ийн блокийг Зураг 2.9–д үзүүлсэн (www.mathworks.com- оос авав) ба орон сууцыг (Зураг 2.9–д House блок) дэд систем болгож дүрслэв (Зураг 2.10) .



Зураг 2.9 Орон сууцны термодинамикийн загварчлал



Зураг 2.10 Орон сууц (House) дэд систем

3 Stateflow ба түүний онолын үндэс

MATLAB-ын Stateflow хэрэгсэлийн тусламжтай төгсгөлтэй автоматыг загварчилдаг. Төгсгөлтэй автоматыг (Finite State Machine) өгөгдлийн хүснэгтээр илэрхийлж болно. Гэвч хамгийн зохистой арга нь төгсгөлтэй автоматыг графикаар илэрхийлэх явдал юм. Иймд Matlab-д төлөв хоорондын хамаарлыг маш энгийн гаргаж, нүдэнд харагдахуйц илэрхийлдэг. Төгсгөлтэй автомат нь үйл явцаар удирдагдах систем юм.

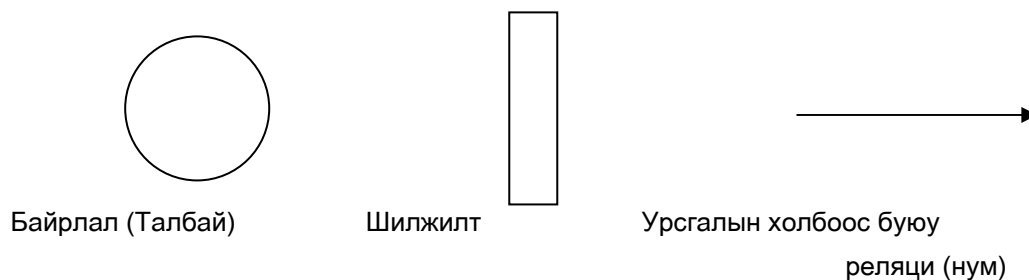
Үйл явцаар удирдах системийн онолын үндэс нь сонгодог Петри сүлжээ (Petri-Net) юм. Matlab-ын Stateflow хэрэгсэл нь Петри сүлжээнд байгаа бүх боломжийг агуулдаггүй. Гэвч Петри сүлжээнд тодорхойлогдоогүй бусад зүйлийг тодорхойлсон байдаг.

Сонгодог Петри сүлжээг өргөтгөсөн нэг хэлбэр нь тодорхой бус Петри сүлжээ юм. Ийм сүлжээнд элемент болгонд тодорхой бус чанарыг нэмж оруулсан тул сонгодог Петри сүлжээг бодвол уян хатан байдаг. Тухайлбал: тодорхой үйл явцад үзүүлэх хариу мэдрэмж нь ямагт ижил байдаггүй, гол нь цаашдын нөхцөлүүдээс хамаардаг.

Жишээлбэл би гаднын танихгүй хүнийг мэндлэхэд найзынхаа мэндлэхээс өөрөөр мэдрэнэ. Ийм үйл явцад миний хариулт ямагт ижил бус, харин тодорхой бус (Fuzzy) байна.

Петри сүлжээн дэх объектууд

Петри сүлжээ нь гурван янзын график элементээс бүрддэг. Үүнд:



Элемент бүр нь түүний хэлбэрээс гадна тодорхой нэртэй болон бусад шинж чанартай байдаг. Байрлал ба шилжилтийг мөн зангилаа гэж зарим үед тодорхойлдог.

Байрлалын (Талбай) шинж чанарууд

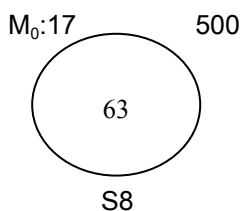
Байрлал буюу талбай гэдгийг нэрнээс нь хүртэл авч үзвэл ямар нэг зүйлийг байрлуулах талбай мэт авч үзэж болно.

Жишээ болгож автомашин тавих гаражийг авч үзвэл энэ нь суудлын автомашин тавих зогсоол юм.

Бидний авч үзэж буй жишээн дээр үзвэл энэ талбайд хамгийн ихдээ 500 машин тавьж болно гэж үзье. Энэ талбайн багтаамжийг (энд 500 машин талбай) байрлалын гадна талд тэмдэглэдэг. Багтаамжийг нь бичээгүй байвал хязгааргүй гэж үзнэ.

Одоо байгаа төлөв байдал болох машины гаражид байгаа автомашины тоог байрлалын дотор талд нь тэмдэглэдэг.

Гаражид байгаа автомашины тоог тэмдэг (марк) гэж нэрлэдэг. Анхнаасаа тэмдэгийг цэгээр тэмдэглэдэг байсан боловч тойрог дотор 63 цэгийг зурах нь практик ач холбогдол муутай тул цэгийн тоог тойрог дотор бичдэг.



S8	S	Байрлалын нэр (Гаражийн нэр)
500	K	Байрлалын багтаамж
63	M	Тухайн хугацааны моментод байгаа тэмдэгтүүд
17	M ₀	Анхны тэмдэгийн тоо

Хэрэв бид гараж хэмээх системийг авч үзэх үед хичнээн машин гаражид байгааг буюу тэмдэгийн тоог хугацааны тэг моментод ($M_0:17$) авч үзэж байж болно.

Байрлал нь нэгэн төрлийн агуулах ба бидний тохиолдолд машинууд буюу гаражид байгаа машины тоо юм. Байрлал нь мэдээллийг агуулдаг.

Мөн түүнчлэн байрлал нь нөхцөлийг бүрдүүлдэг. Хэрэв гаражид 500 машин байвал, дахиж ямар ч машин оруулах боломжгүй. Ингэхээр гаражид машин орохын тулд гараж дүүрээгүй байх нөхцөл биелэгдэх ёстой. Эсвэл эсрэгээр авч үзвэл гаражаас нэг машин гарч явна гэвэл дор хаяж нэг машин гаражид байх ёстой.

Шилжилтийн шинж чанарууд

Шилжилт нь боломжтой үйл явцыг илэрхийлдэг. Бидний гаражийн хувьд авч үзвэл үйл явцын хувьд нэг шилжилт нь “Нэг машин гаражид орохыг завдаж байна” юм. Өөр нэг үйл явцын шилжилт нь “Нэг машин гаражаас гарах гэж байна” гэж болно.



t 2 Т шилжилтийн нэр
(үйл явцын нэр)

Шилжилт

Шилжилтийн ганцхан шинж чанар нь шилжилтийн нэр буюу өөрөөр хэлбэл түүнд шилжилт мэдэрдэг үйл явцын нэр юм.

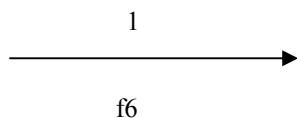
Ийм нэг үйл явц тааралдсан гэвэл, хэрэв нөхцөлтэй холбоотой нөхцөл биелэгдвэл л энэ шилжилт нь зөвхөн залгагдана.

Жишээлбэл “Нэг машин гаражид орохыг завдаж байна” гэвэл энэ нь хэрэв гараж дүүрээгүй байвал л боломжтой.

Хэрэв нэг шилжилт биелэгдвэл, нөхцөл нь өөрчлөгддөг.

Урсгалын холбоосын (нум) шинж чанарууд

Урсгалын холбоос нь байрлал ба шилжилтийг холбодог. Энэ нь байрлал ба шилжилтийн хоорондын хамаарлыг (реляци) тодорхойлж харуулдаг. Хэрэв урсгалын холбоосын сумын дагуу дагалдвал өгөгдлийн урсгалын дагуу шилжих юм. Ямар нэг урсгалын реляци нь зөвхөн нэг байрлалыг шилжилттэй холбох буюу эсрэгээр байна. Ижил төрлийн зангилааг (байрлал ба шилжилт) хооронд нь холбож болохгүй.



f6 F Урсгалын реляцийн нэр (нум)

1 W Урсгалын реляцийн (нум) жин

Урсгалын реляци (нум)

Урсгалын реляци нь хоёр шинж чанартай. Үүнд жирийн тохиолдолд тэмдэглэгддэггүй урсгалын реляцийн нэр хамаарах ба энэ нь ямар байрлал ба ямар шилжилт урсгалын реляцаар холбогдсоныг илэрхийлдэг.

Нөгөө нэг шинж чанар нь урсгалын реляцийн жин юм. Бидний авч үзэж буй гараж нь машин орох зөвхөн нэг хаалгатай юм. Ингэхээр тухайн хугацааны үед зөвхөн нэг машин л орох тул урсгалын реляцийн жин нь $W:1$ юм.

Гараж нь машин гарч явах хоёр хаалгатай тул машин гарч явах урсгалын реляцийн жин нь $W:2$ юм. Бидний жишээ болох $W:2$ жин бүхий урсгалын реляци нь хоёр машин нэгэн зэрэг гарч явж болохыг заах бус харин хоёр машин нэгэн зэрэг гарч явах ёстойг заадаг.

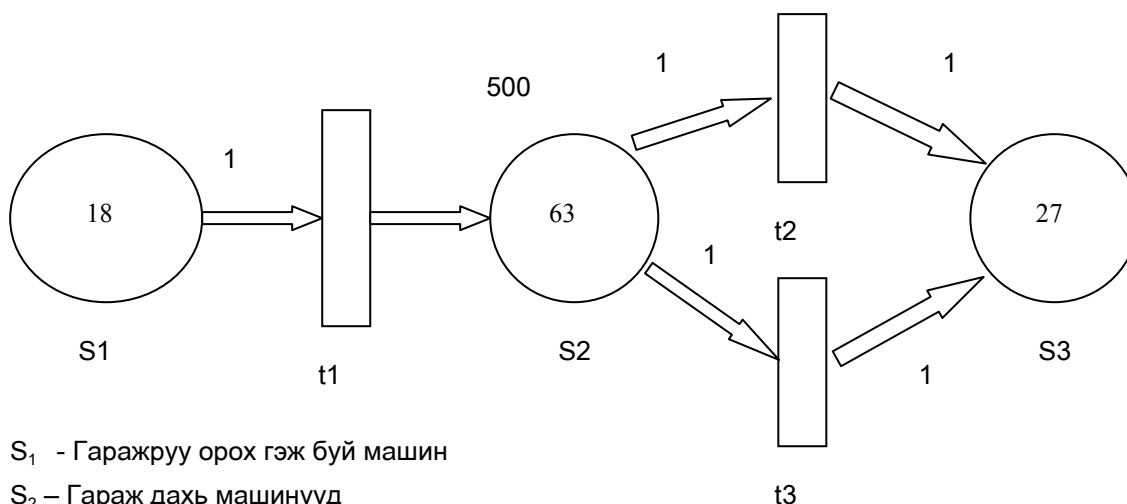
Бидний авч үзэж буй гаражийг өөрөөр дүрсэлж болох юм. Тухайлбал, нумын жин тус бүр нь $W:1$ бүхий хоёр зэрэгцээ сумыг зурж болно. Тэгвэл энэ нь хоёр машин нэгэн зэрэг гаражаас гарч болохыг харуулж байна. Гэхдээ мөн зөвхөн нэг машин гаражаас гарч явах боломжтой.

Товч дүгнэлт:

- Байрлал нь нөхцөл ба нөхцөлүүд нь мэдээллийг агуулдаг
- Шилжилт нь үйл явц ба үйл явц нь өөрчлөгддөг.
- Нум нь мэдээлэл / нөхцөл ба үйл явцын үйлдэл, холбоог илэрхийлдэг.

Гараж нь Петри сүлжээ болох нь

Энд гаражийг жишээ болгон авч үзлээ. Энэ загварт гараж нь гурван хаалгатай юм. Нэг хаалга нь орох ба нөгөө хоёр хаалга нь гарахад зориулагдсан юм. Цаашилбал гараж нь хамгийн ихдээ 500 машин зогсоолтой юм.



S_1 - Гаражруу орох гэж буй машин

S_2 - Гараж дахь машинууд

S_3 - Гаражаас гарсан автомашинууд

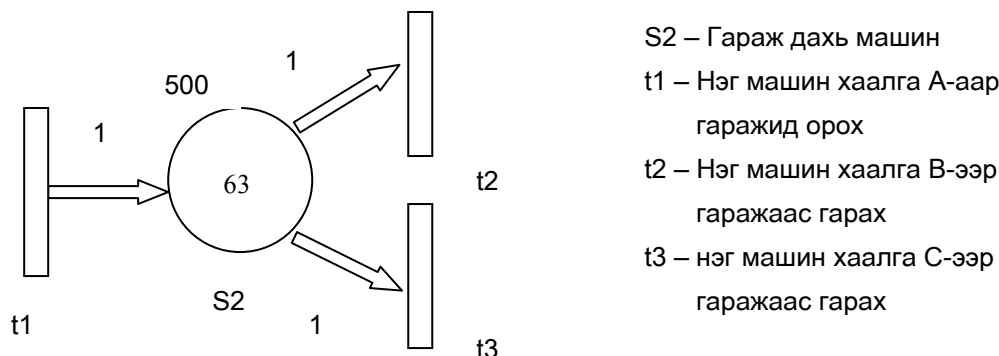
t_1 - Хаалга А-аар гаражид нэг машин орж байна

t_2 - хаалга В-ээр нэг машин гаражаас гарч байна

t_3 - хаалга С-ээр нэг машин гаражаас гарч байна

“S3” байрлал байхгүй байж болох юм. Энэ нь ямар ч ач холбогдолгүй ба зөвхөн нэг өдөрт гарч буй машины тоо буюу нийт машины тоог гаргахад л хэрэгтэй байж болох юм.

Мөн түүнчлэн “S1” байрлал байхгүй байж болох юм. Энд харин гараж дүүрэн байгаа буюу үгүй байгаа нь л сонирхогдоно. Тэгвэл бид загварыг доорх байдлаар хялбарчилж болох юм.



Петри сүлжээг формал тодорхойлох нь

Петри сүлжээ нь зургаан жагсаалтаар формал тодорхойлогддог:

$$N = (S, T, F, K, W, M_0)$$

S: Байрлал $S = \{s_1, s_2, \dots, s_n\}$ -ийн хоосон бус төгсгөлтэй олонлог

T: Шилжилт $T = \{t_1, t_2, \dots, t_m\}$ -ийн хоосон бус төгсгөлтэй олонлог

F: Урсгалын реляци $F = (S \times T) \cup (T \times S)$ -ийн хоосон бус төгсгөлтэй олонлог

K: Байрлал $K: S \rightarrow \mathbb{N} \setminus \{0\}$ - ийн багтаамж

W: Урсгалын реляци $W: F \rightarrow \mathbb{N} \setminus \{0\}$ - ийн жин

M_0 : Анхны тэмдгийг олгох $M_0: S \rightarrow \mathbb{N}$

Тэмдэглээний тайлбар:

\subseteq - Дэд олонлог буюу тэнцүү

$S \times T$ - S ба T-ийн реляци (холбоо, хамаарах)

$T \times S$ - T ба S-ийн реляци (холбоо, хамаарал)

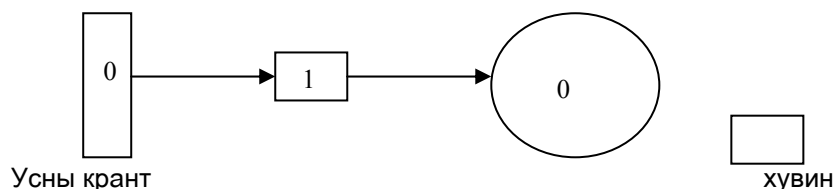
\cup - Олонлогийн нэгдэл

\rightarrow - Цэгцтэй харгалзана

Тодорхой бус Петри сүлжээ

Тодорхой бус Петри сүлжээнд (Fuzzy Petri Net) зангилаа (байрлал ба шилжилт) бүрд үнэлгээний функц харгалздаг.

Байрлалын хувьд энэ нь хичнээн тэмдэгтэй байхыг хүсч байгааг харуулдаг. Шилжилтийн хувьд энэ функц нь шилжилт ямар хүчтэйгээр залгагдахыг харуулдаг.



Товчхон жишээ:

Усны хувийг дүүргэхийн тулд усны краныг бүрэн нээдэг. Хувин ойролцоогоор 90% дүүрвэл, хувингаас ус халихаас болгоомжлон краныг аажмаар эргүүлж, хувинг дүүргэдэг. Энэ нь төгсгөлийн төлөвт аажмаар хүрэх хандлага юм.

Stateflow нь Simulink-ийн графикийн боломжийг өргөжүүлж, төгсгөлтэй төлөвт автоматын (finite state machine) ажиллагааг загварчлах ба симуляци хийх боломж олгодог программ хангамж юм. Энэ нь үйл явцын удирдлагатай төгсгөлтэй төлөв байдал бүхий системийг загварчлахад хэрэглэдэг хэрэгсэл билээ. Simulink системтэй харьцуулбал хоёр янзын үндсэн ялгаа байдаг. Үүнд, Stateflow нь үйл явцын удирдлагатай ба зөвхөн дотоод буюу гаднын үйл явц (event, триггерлэх дохио) үйлчлэхэд л ажилладаг. Тэгвэл Simulink систем нь интегралчлалын тогтмол буюу хувьсах алхамтайгаар үйл явцаас үл хамааран тооцоо хийдэг. Simulink-д төлөв байдал хэмээх ойлголт нь динамик системийн төлөвийн тасралтгүй утгын хэмжигдэхүүнийг тэмдэглэдэг.

Тэгвэл Stateflow загварын төлөв байдал нь идэвхитэй эсвэл идэвхигүй гэсэн хоёр утгатай л байдаг. Ийм автоматын нэг тод жишээ бол гудамжны гэрлэн дохио юм. Гэрлэн дохионд **улаан, шар, ногоон** гэсэн төлөв байдал байх ба төлөв байдал бүр нь идэвхитэй (ассан) буюу идэвхигүй (унтарсан) байж болно.

Төгсгөлтэй автомат нь графикийн редактороор төлөвийн шилжилтийн диаграмм мэт дүрслэгдэх ба Simulink-ээр тэмдэг (chart) мэт загварчлагддаг. Төлөвийн шилжилт нь үйл явц (event) ба нөхцлөөс хамаарч зөвшөөрөгддөг. Simulink –тэй харьцуулбал Stateflow нь үйл явцын удирдлагатай юм. Өөрөөр хэлбэл аливаа нэг chart (тэмдэг) нь зөвхөн урьдчилан тодорхойлсон үйл явц (event) тааралдахад л ажилладаг. Stateflow нь Matlab ба Simulink –д иж бүрэн нэгдсэн ба Stateflow–г ашиглахад Simulink ямагт хэрэглэгддэг юм. Загварчлал хийх явцад Simulink –тэй сигнал солилцох ба Matlab-ын бүх функцийг ашигладаг. Stateflow ажиллаж эхлэхээс өмнө нь Chart -аас Simulink-ийн S-функцийг дэд каталог **sfprj** дотор үүсгэдэг. Функц нь автоматоор үүсэх ба хэрэглэгчийн оролцоог шаарддаггүй юм. Энэ үед дараах алхмыг гүйцэтгэдэг. Үүнд:

- C - кодыг хөрвүүлэх компиляци хийх
- mex файлыг үүсгэх (Windows – ын орчинд *.dll ба Unix/Linux – ын орчинд *.mex)

Stateflow-Coder-ын тусламжтай Stateflow Chart -аас бодит хугацааны C -кодыг автоматаар үүсгэх боломжтой. Ингэснээр Stateflow загварыг ямарч асуудалгүйгээр Hardware-in-the-Loop буюу Rapid-Control-Prototyping системд нэгтгэж болдог. Stateflow-г ашиглахын тулд C-компиляторыг (C-Compiler) заавал хэрэглэх шаардалагатай. Анх эхлэн хэрэглэхээс өмнө компиляторыг суурилуулах процессыг (Setup) хийх хэрэгтэй. Үүний тулд Matlab-ын командын цонхонд
>>mex –setup
командыг өгдөг.

Ингэхэд Windows-ын орчинд Lcc, Unix/Linux-т GNU C-компилятор Gcc-г сонгох шаардлагатай. Ингэж үүсгэсэн суурилуулах мэдээлэл нь файлд хадгалагдаж, компиляторын цаашдын ажиллагаанд бэлэн болдог.

3.1 Stateflow-ын элемент

Stateflow-ын үндэс нь санамжийн ой бүхий төлөвийн шилжилтийн диаграмм юм. Гэвч Stateflow-ын элементээс санамж ойгүй (stateless) урсгалын диаграммыг гаргаж болдог. Төлөвийн шилжилтийн диаграмм ба урсгалын диаграммыг chart хэмээх график дүрсээр илэрхийлж, SIMULINK –тэй холбодог. Stateflow-ын графикийн бус элемент болох chart(үйл явц –event, янз бүрийн төрлийн өгөгдөл, Simulink –рүү орох гарах) нь Data Dictionary-д хадгалагдах ба Stateflow Explorer –ээр удирдагддаг. Stateflow нь объект хандлагат бүтэцтэй юм. Өөрөөр хэлбэл элемент (объект) бүр тодорхой нэгэн эх(ээж) –объектыг (parent –төрөл) эзэмших ба өөртөө дурын олон тооны хүү -объектыг (children) үүсгэдэг. Эдгээр хүү объект нь мөн хүү объектын эх нь Stateflow chart өөрөө юм. Chart-д агуулагдах төлөв байдал, үйл явц (event), хувьсах буюу функцүүд нь chart –ын хүү юм. Ийм эх – хүү объектын хамаарал нь иерархи (дэс дараалсан) үүсгэх боломж олгодог. Эх-хүүгийн холбоог тэмдэглэхдээ объектуудын хооронд зааглах цэг (.) бичдэг. Хэрэв kind1 нэртэй хүү-объект mutter2 нэр бүхий эх-объектынх байвал mutter2.kind1 гэж илэрхийлдэг. Stateflow chart нь Simulink-д жирийн дэд систем шиг ойлгогддог. Иймд SIMULINK –ийн нэг загварт хэд хэдэн chart байж болох ба chart бүр нь SIMULINK ба бусад chart-тай (SIMULINK-ээр дамжуулан) өгөгдөл солилцож болно. Бүх chart-ыг нийлүүлж төлөвийн машин (State Machine) гэж нэрлэдэг. Симуляци явагдахын өмнө State Machine –аас дан ганц SIMULINK –ийн S-функцийг үүсгэдэг (C-компиляторыг хэрэглэх замаар). SIMULINK загварт хоосон chart оруулахын тулд

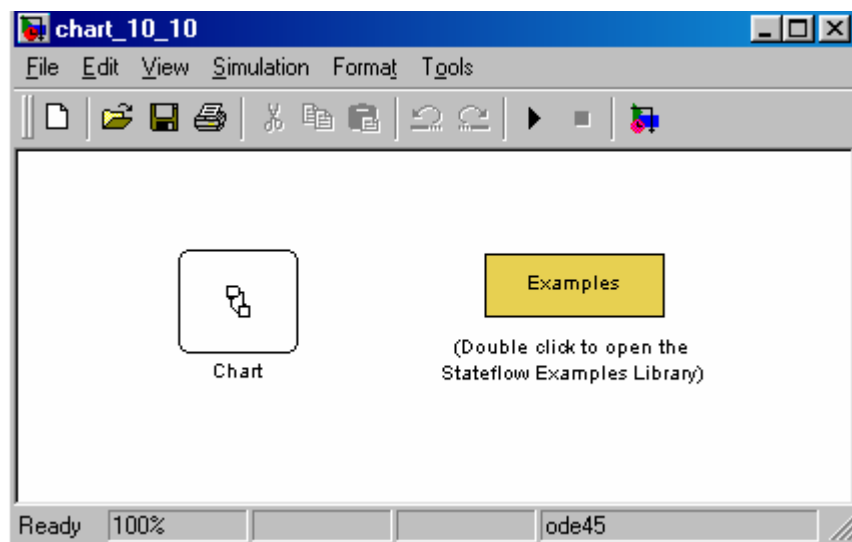
```
>>stateflow
```

```
товчлон
```

```
>>sf эсвэл >>sfnew
```

```
команд өгч Stateflow-ын санг (Library) нээдэг.
```

```
Stateflow Library (сан) нь хоосон chart-ын зэрэгцээ маузаар давхар тогшиход
```



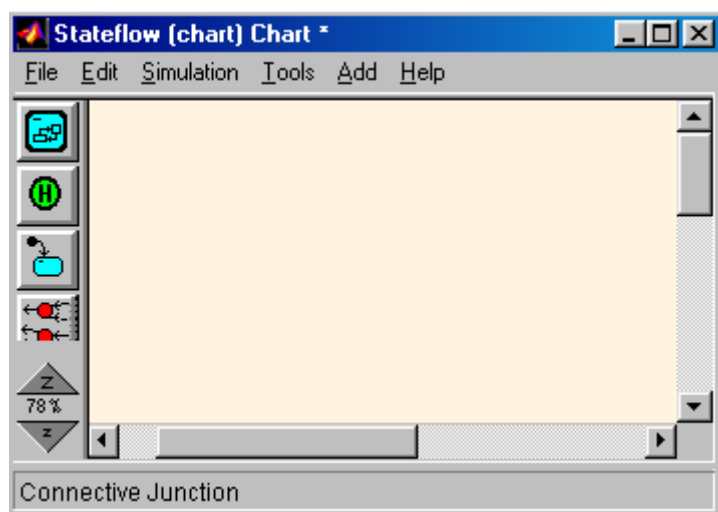
Зураг 3.1 Stateflow –ын сан

нээгдэж ажилладаг жишээг агуулсан байдаг (зураг 3.1).

Хоосон chart-г маузын зүүн товчлуураар татаж SIMULINK загвар руу оруулдаг.

Хоосон chart-г хоёр дахин тогшвол Stateflow-ын график редактор (editor) нээгддэг.

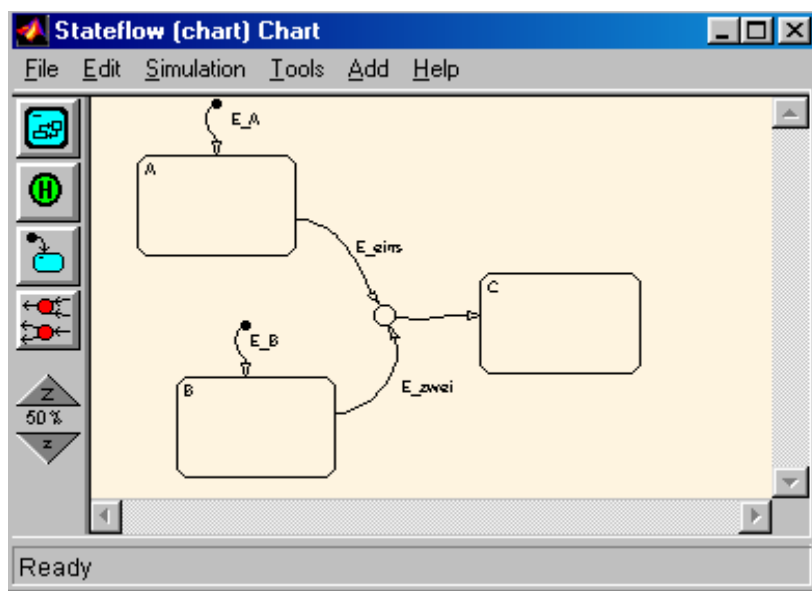
Зураг 3.2-д график элементийг байгуулах хэрэгсэл бүхий баганатай редакторыг үзүүлжээ.



Зураг 3.2 Stateflow-ын график редактор

Хэрэв зүүн талын (зураг 3.2) баганаас ямарч хэрэгсэлийг сонгоогүй бол редакторын маузын заагчаар төлөвийн шилжилтийг (transition) зурж болдог. Stateflow-ын chart-ын бусад элементүүдийг хэрэгсэлийн баганаас сонгож оруулдаг. Chart нь төлөв байдал (State Tool), төлөвийн шилжилт (редакторын стандарт тохируулга), Connective Junctions (Connective Junction Tool) (холболтын зангилаа), History Junction (History Junction Tool) болон төлөв ба шилжилт дэх тэмдэгээс (Label) бүрддэг. Зураг 3.2 –д байгаа толгой талын цэсээс гадна, тэмдэглэгдсэн объектод маузын баруун товчийг дарж, Shortcut цэсийг гаргаж янз бүрийн үйлдэл гүйцэтгэж болдог. SIMULINK-ийн адил товчлуурын хослолоор тэмдэглэх, хуулах, тайрах, оруулах, шилжүүлэх, арилгах болон хэд хэдэн объектыг тэмдэглэх зэрэг үйлдлийг гүйцэтгэдэг.

Зураг 3.3-д бүх үндсэн элементээс бүрдэх Stateflow-ын chart-г үзүүлжээ.



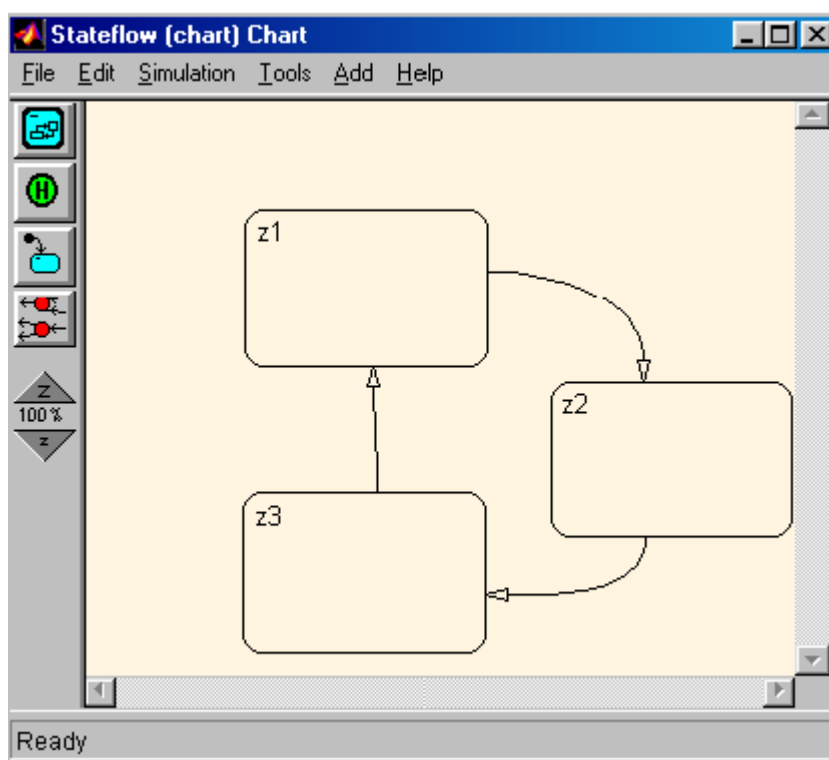
Зураг 3.3 Stateflow chart-ын жишээ

3.2 Chart –ын график элемент

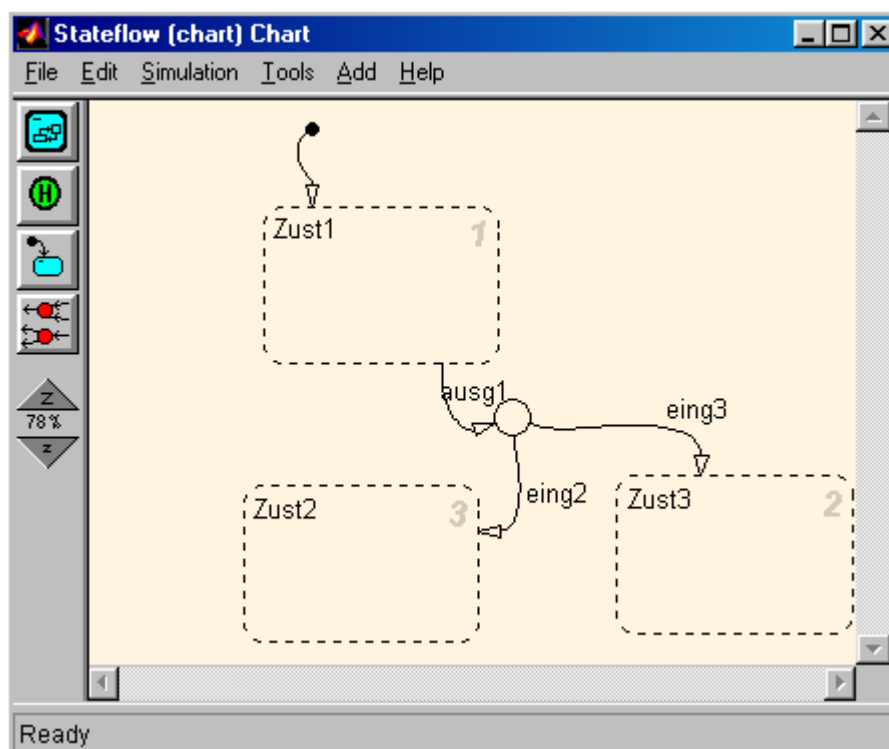
Chart –ын график элемент нь боломжит төлөв байдал, төлөвийн зөвшөөрөгдөх шилжилт, шилжилтийн нөхцөл ба гүйцэтгэх үйл явц зэргийг тодорхойлдог.

Төлөв байдал

Төлөв байдал нь аливаа системийн тухайн үед оршин байгаа хэв шинжийг (modus) илэрхийлдэг. Chart нь дурын тооны төлөв байдалтай байж болдог. Төлөв байдлыг цуваа төлөв байдал (OR-төлөв) ба зэрэгцээ төлөв байдал (AND-төлөв) гэж хоёр төрөлд хуваадаг. Төлөвийн төрлийг Shortcut цэсний Decomposition цэсээр сонгодог. Цуваа төлөв байдлыг хэрэглэх үед тухайн моментод зөвхөн нэг төлөв идэвхтэй байдаг (зураг 3.4) бол зэрэгцээ төлөв байдалд хэд хэдэн төлөв байдал нэгэн зэрэг идэвхтэй байдаг (Зураг 3.5). Графикийн редакторт цуваа төлөв нь тасралтгүй, зэрэгцээ төлөв нь тасралттай шугамаар зурагддаг.

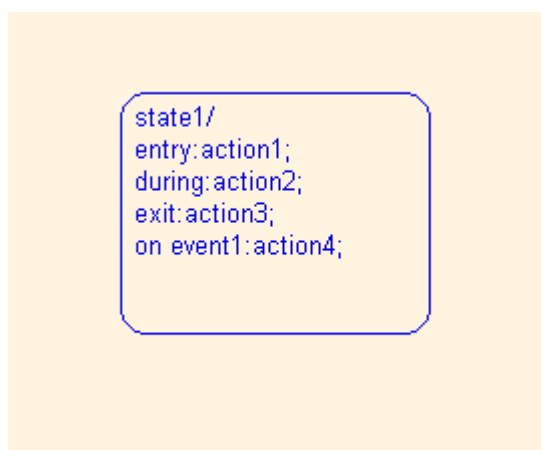


Зураг 3.4 OR-төлөв бүхий chart



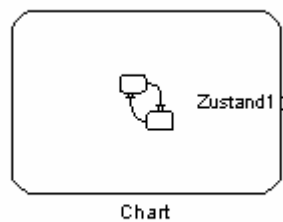
Зураг 3.5 Зэрэгцээ (AND) төлөв бүхий chart

Chart дахь аливаа төлөв байдлыг график редакторын State Tool –оор сонгох ба хүсч буй байрлалдаа төлөвийн тэмдэгийг байрлуулдаг. Шинэ төлөв байдал бүр эхлээд “?” тэмдэгтэй (Label) байх ба маузны зүүн товчлуураар дарж, өөр нэр өгдөг. Нэг иерархи түвшинд байгаа бүх төлөвүүд өөр өөр нэртэй байх ёстой. Төлөвийн тэмдэг нь нэрнээс гадна сонголт бүхий үйл явцтай байдаг. Төлөвийн тэмдгийн жишээг зураг 3.6-д үзүүлээ.



Зураг 3.6 Иж бүрэн тэмдэг (Label) бүхий төлөв байдал

Хэрэв төлөвийн идэвхижилт нь Stateflow Chart-ын гадна Simulink –ийн загварт шаардлагатай бол Output State Activity дэд цэсийг сонгож хэрэгжүүлдэг. Ингэхэд Data Dictionary –д State төрөл бүхий төлөвийн нэртэй хувьсах үүсэх ба Chart –т нэмэлт гаралтыг гаргаж харуулдаг (Зураг 3.7).



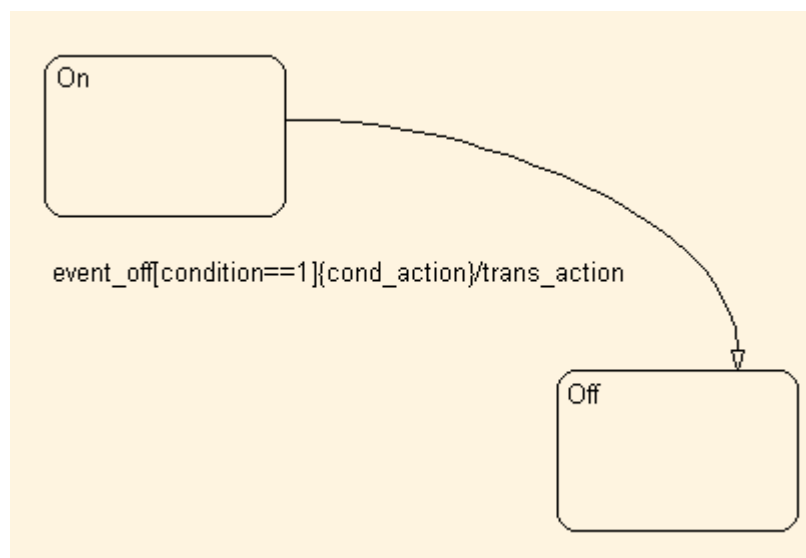
Зураг 3.7 Төлөвийн идэвхижилтийн Simulink- рүү гарах гаралт

Шилжилт (transition)

Шилжилт нь үүсгүүр объектод chart –ын шилжих боломжийг илэрхийлдэг. Шилжилт нь хоёр төлөвийн хооронд болон төлөв байдал ба холболтын цэгийн хооронд (connective junstion) байрладаг. Хоёр төлөвийн хоорондын шилжилтийг гол төлөв хэрэглэдэг. Шилжилтийг маузны зүүн товчлуураар үүсгэдэг. Шилжилт нь мөн тэмдэгтэй (Label) байдаг. Тэмдэг нь шилжилт хүчинтэй байх нөхцөлийг тодорхойлдог (зураг 3.8).

Шилжилтийн ерөнхий хэлбэр:

`event[condition]{condition-action}/transition-action`



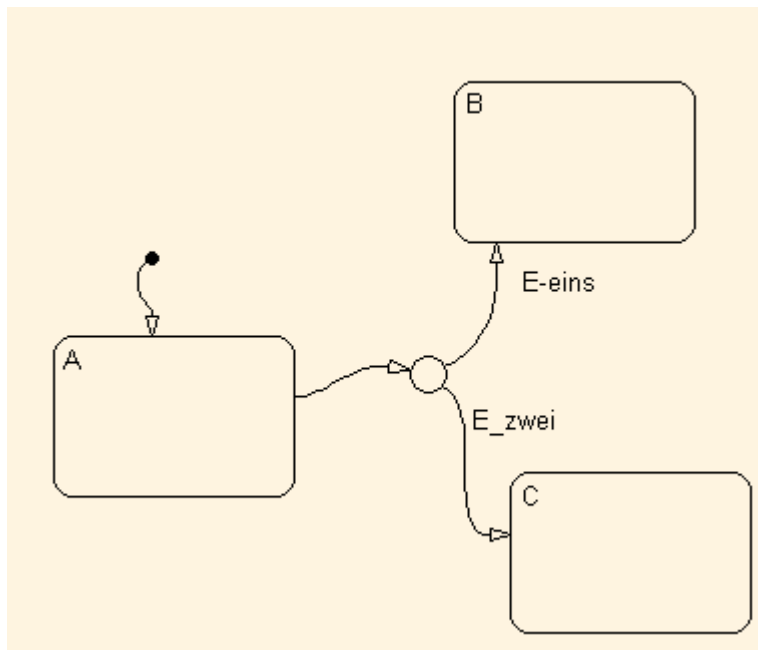
Зураг 3.8 Залгах/салгах залгуурын жишээгээр шилжилтийн иж бүрэн тэмдэгийг дүрсэлсэн нь

3.3 Холболтын цэг (Connective Junstions)

Холболтын цэг нь аливаа шилжилтийн хэд хэдэн боломжтой салаанаас хэрхэн шийдвэр гаргах болмжийг тодорхойлдог. Ингэснээр for, do-while давталт ба if-нөхцлийг хэрэгжүүлэх боломж гардаг. Stateflow chart-д холболтын цэгийг тойргоор

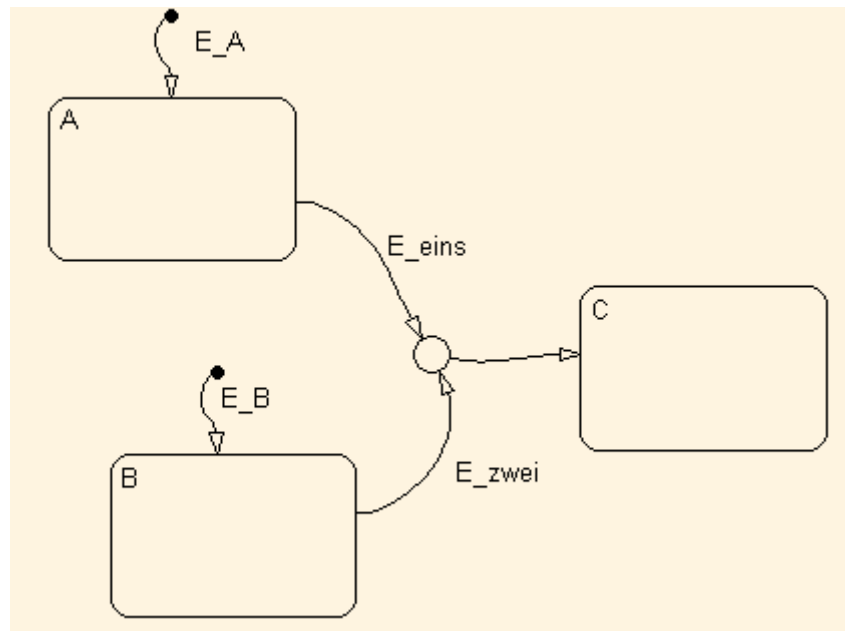
илэрхийлдэг. Үүний тулд Connective Junction Tool-г сонгож, маузны зүүн товчлуураар үүсгэдэг. Доорхи үндсэн шилжүүлгийг хэрэглэдэг. Үүнд:

Нэг үүсгүүрээс хоёр зорилгод хүрэх шилжилт:



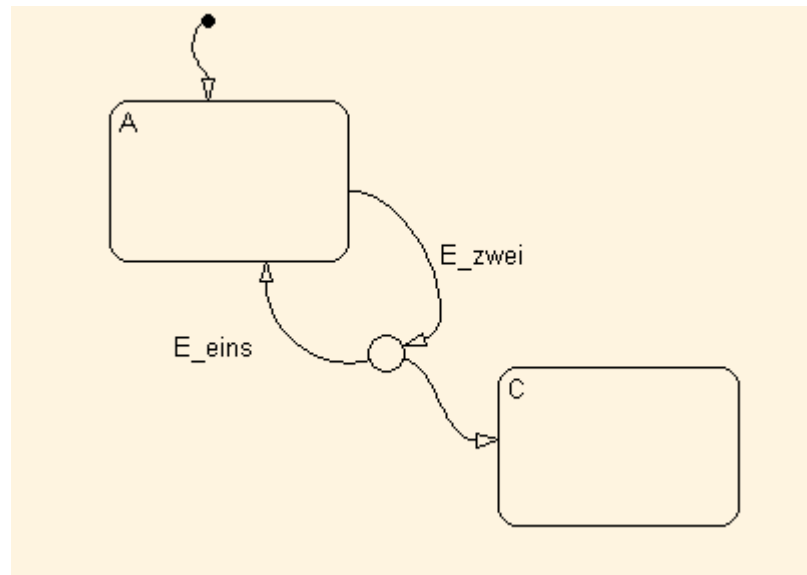
Зураг 3.9 Нэг үүсгүүрээс хоёр зорилгод хүрэх шилжилт

Хоёр үүсгүүрээс нэг зорилгод хүрэх шилжилт :



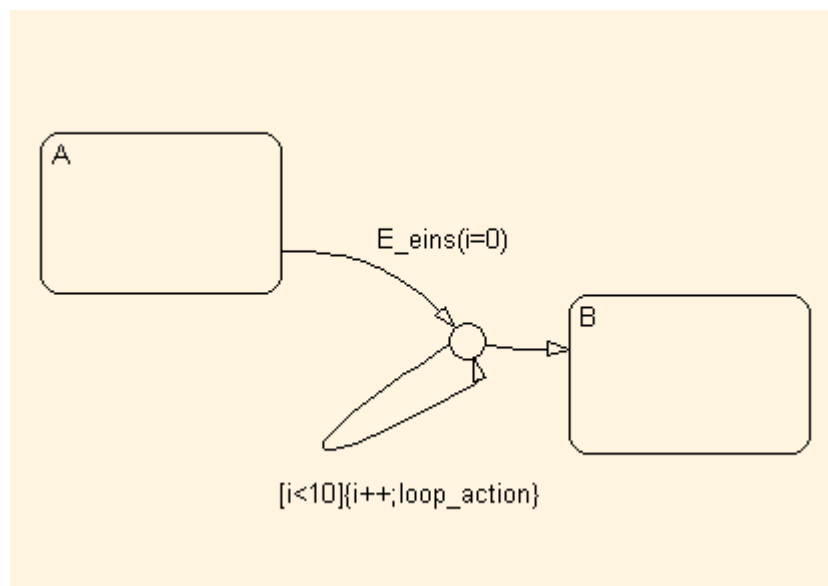
Зураг 3.10 Хоёр үүсгүүрээс нэг зорилгод хүрэх шилжилт

Self Loop (өөрөө давтах) шилжилт :



Зураг 3.11 Self Loop шилжилт

For –давталт:



Зураг 3.12 for –давталттай холболтын цэг

If –асуулга:

Matlab-ын if командыг ашиглаж

```

if a>b
    if a>c
        action1
    else
        action2
    end
else

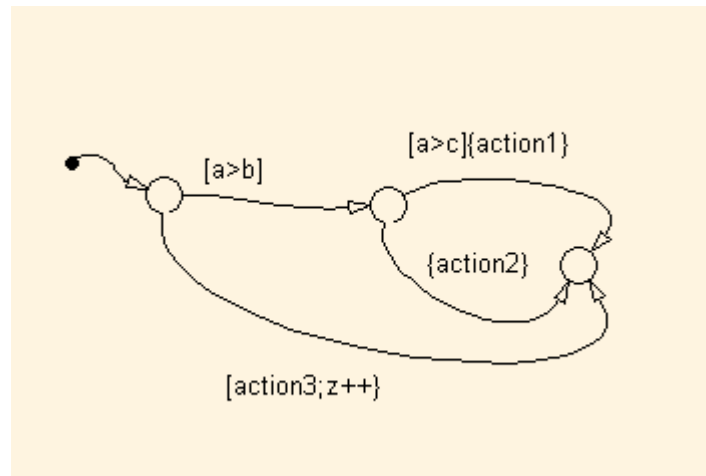
```

```

    action3
    z=z+1
end

```

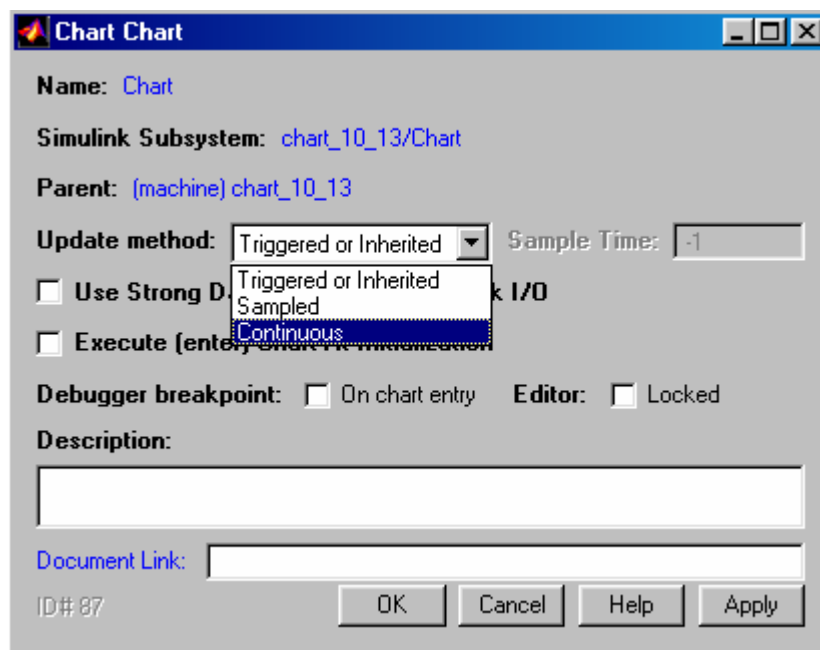
программын хэсгийг зураг 3.13-д үзүүлсэн chart-аар сольж болдог.



Зураг 3.13: Stateflow дахь if –асуулгын урсгалын диаграмм

3.4 Chart –ын шинж чанар ба тригерлэх арга

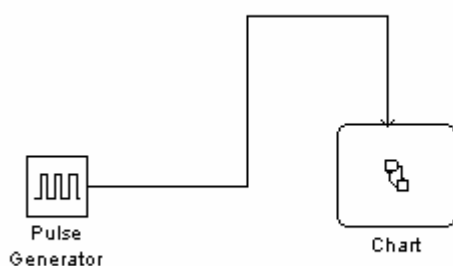
Stateflow chart нь үйл явц (event) тохиолдох үед л ажилладаг. Үйл явцыг дотоод , тойруу (жишээ нь нэг төлөв идэвхижих буюу нэг төлөвөөс гарахад) буюу гадна талаас үүсгэж болдог. Загварчлал ажиллаж эхлэхэд chart нь тайван байдалд байх (State Machine бүхэлдээ) ба ажилуулахын тулд гаднын үйл явц идэвхижих (“сэрээх” гэж ярьдаг) ёстой. Chart –г идэвхижүүлэх Triggered, Inherited, Sampled ба Continuous гэсэн 4 янзын арга байдаг. Эдгээр аргуудыг File/Chart Properties цэснээс сонгодог ба Triggered /Inherited нь стандарт арга (Standart -Update) юм (Зураг 3.14).



Зураг 3.14: Chart –Properties дахь Update –Method –ын тааруулга

Триггерлэх (triggered) арга

Энэ аргад хэрэв гадна талаас нь яг тодорхой триггерлэх дохиог өгвөл Chart идэвхиждэг. Үүний тулд Data Dictionary-д дор хаяж нэг үйл явц (event) SIMULINK-ийн оролт мэт тодорхойлогдсон байх ёстой (Зураг 3.15).



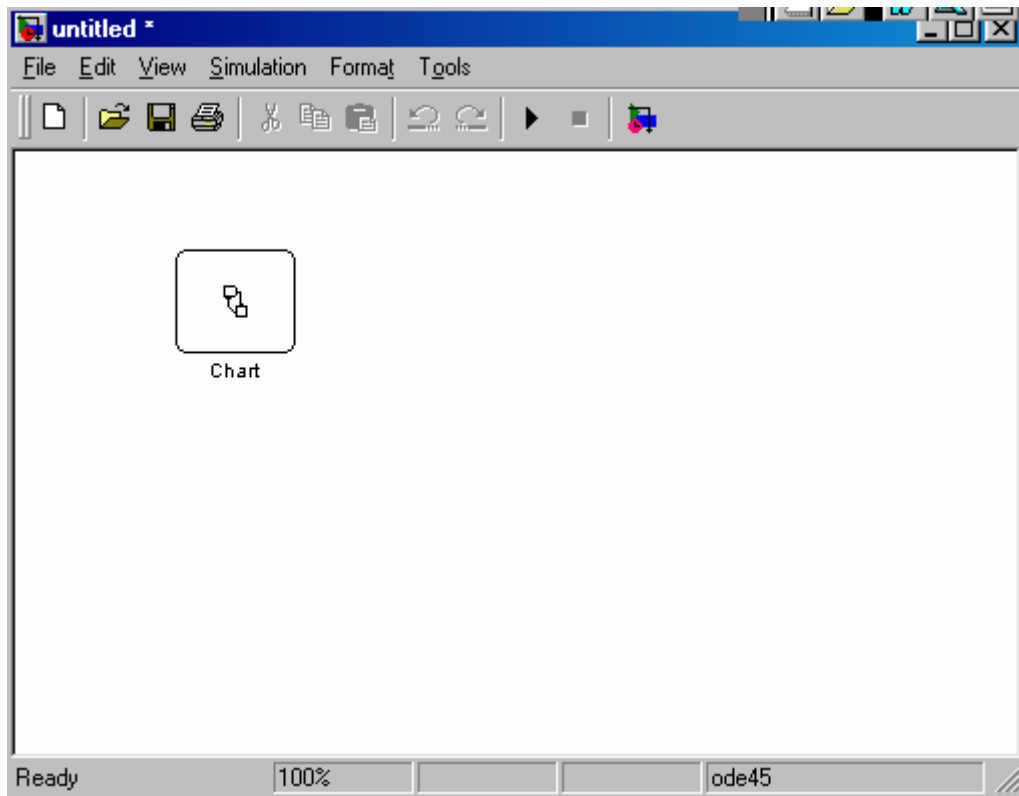
Зураг 3.15 Stateflow Chart –г триггерлэх

3.5 Stateflow- ыг ашиглах жишээ

Хэрэв гадна агаарын температур $30\text{ }^{\circ}\text{C}$ хэмээс бага байвал сэнс салгаатай байдалд (off) байна. Ийм биш бол On байдалд шилжиж сэнсийг залгах процессыг Stateflow – ын диаграммаар илэрхийлж загварчлах

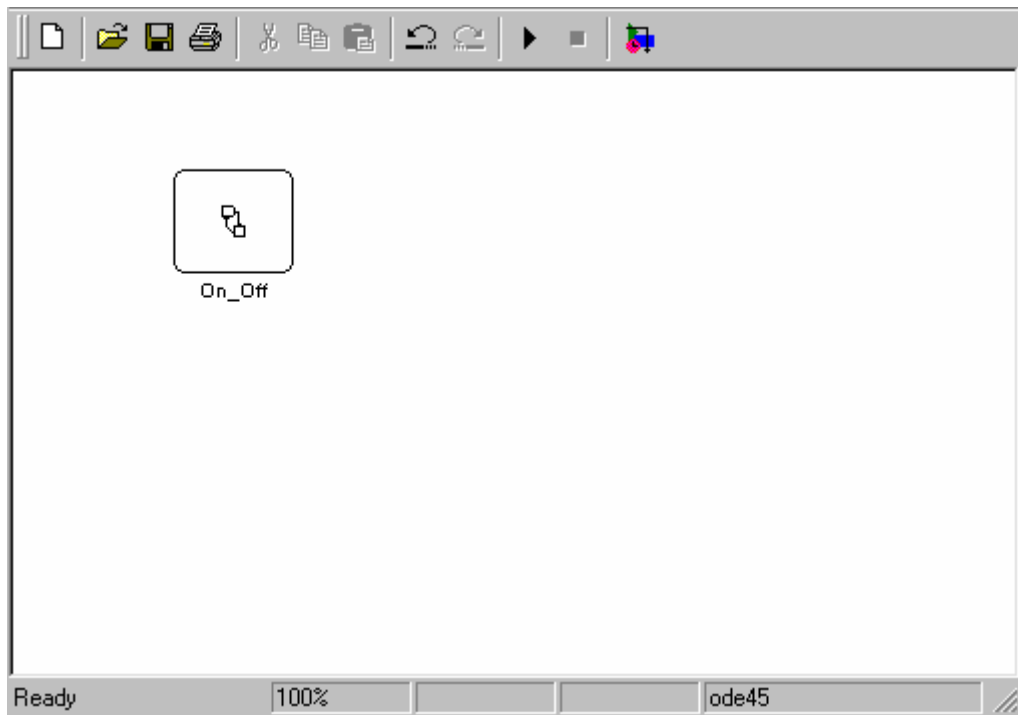
Үүний тулд ямар дараалалтай үйлдэл гүйцэтгэхийг үзье. Үүнд: Simulink -ийн загварыг үүсгэнэ.

1. Matlab- ын промптын(>>) дараа sfnew (эсвэл sf) гэж оруулна. Ингэхэд доорхи цонх гарна (Зураг 3.16).



Зураг 3.16

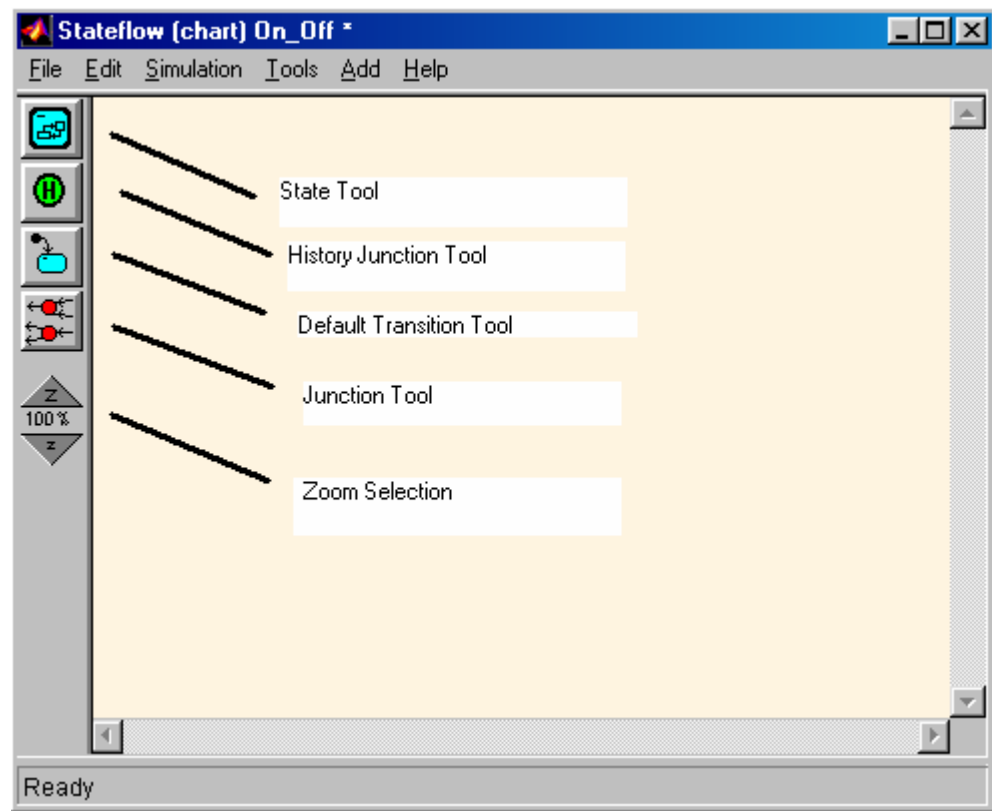
2. Stateflow - ын Untitled (Chart) тэмдэгийг маузоор тогшиж On_Off гэж тэмдэгийн нэрийг сольё (Зураг 3.17). Ямар нэр өгөхийг хэрэглэгч өөрөө сонгоно гэдгийг анхаарна уу?
3. File цэсээс Save As –ыг сонгож Simulink- ийн загварыг хадгална.



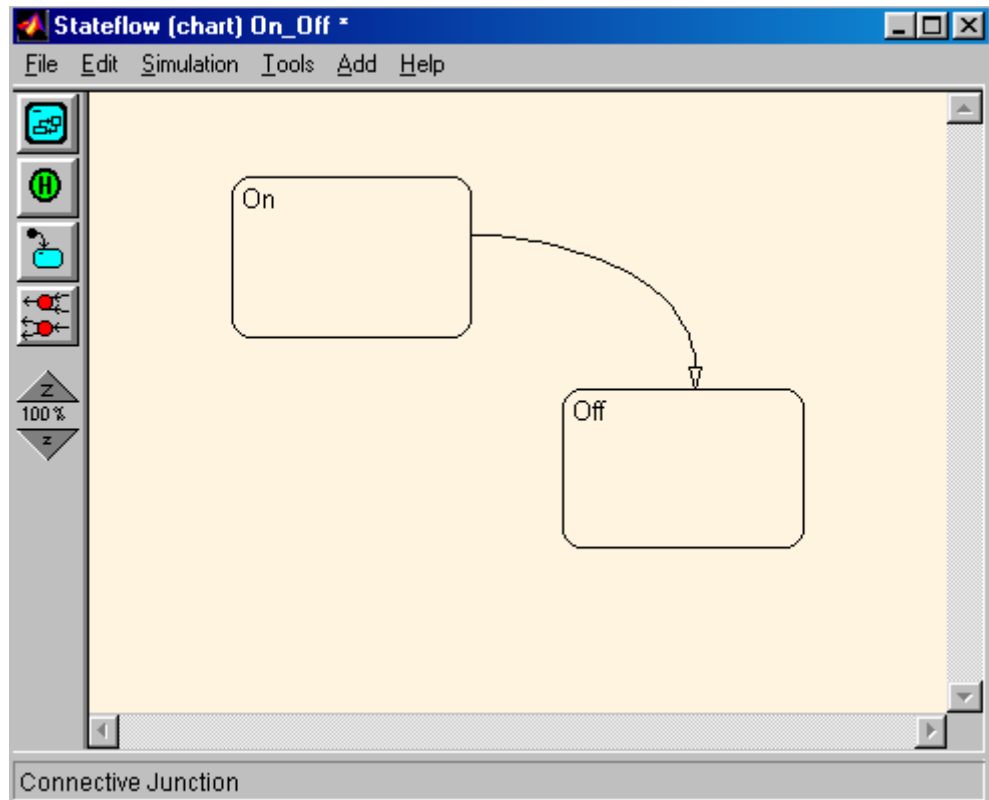
Зураг 3.17

3.5.1 Stateflow -ын диаграмм үүсгэх

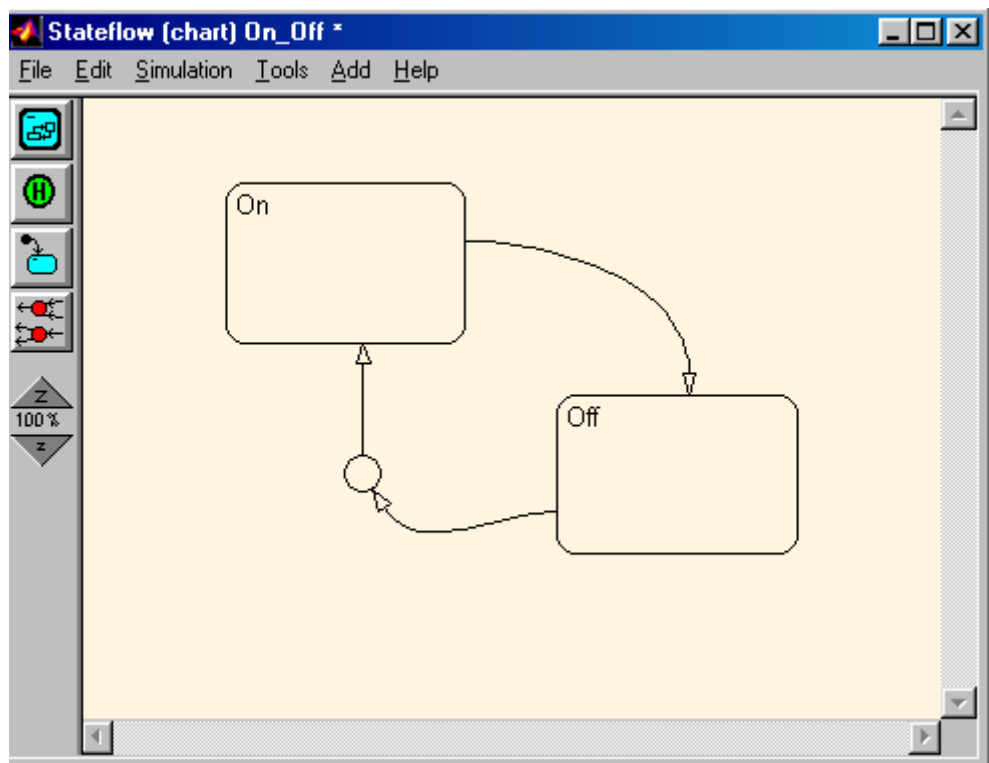
1. Simulink дэх Stateflow –ын загварын цонход маузаар хоёр удаа тогшино. Ингэж графикийн редакторын цонх гардаг. Энэ цонхны зүүн талд зураг зурахад шаардлагатай дүрсний хэрэгсэл бүхий багана гардаг (Зураг 3.18).
2. Зургийн хэрэгслийн баганаас **State** -г сонгож, маузын заагчаар төлөв байдлын дүрсийг шилжүүлж, редакторын цонхонд байрлуулна. Төлөвийн дүрс доторхи ‘?’ тэмдэгийн нэрийг On ба Off гэж сольё. Төлөвийг идэвхижилтгүй болгохдоо төлөвийн гадна талд тогших буюу **Esc** товчийг дарна (Зураг 3.19).
3. Төлөв хоорондын шилжилтийг (transition) маузын зүүн курсороор зурдаг (Зураг 3.19). Харин холболт хийхийн тулд **Junction** товчлуурыг маузаар тогшиж, зангилааны цэгийг графикийн редактор руу зөөдөг (Зураг 3.20).
4. On төлөвөөс Off төлөвт шилжих шилжилтийн тэмдэгийн нэрийг off_switch, Off төлөвөөс зангилааны цэгрүү шилжих шилжилтийн тэмдэгийн нэрийг on_switch гэж тус тус нэрлэе. Зангилааны цэгээс Off төлөврүү шилжих шилжилтийн тэмдэгт [temp<=30] гэсэн нөхцөл олгоё. Off төлөврүү шилжих стандарт шилжилтийг зургийн хэрэгсэлийн **Default Transition** -г сонгож Off төлөврүү холбоно (Зураг 3.21).



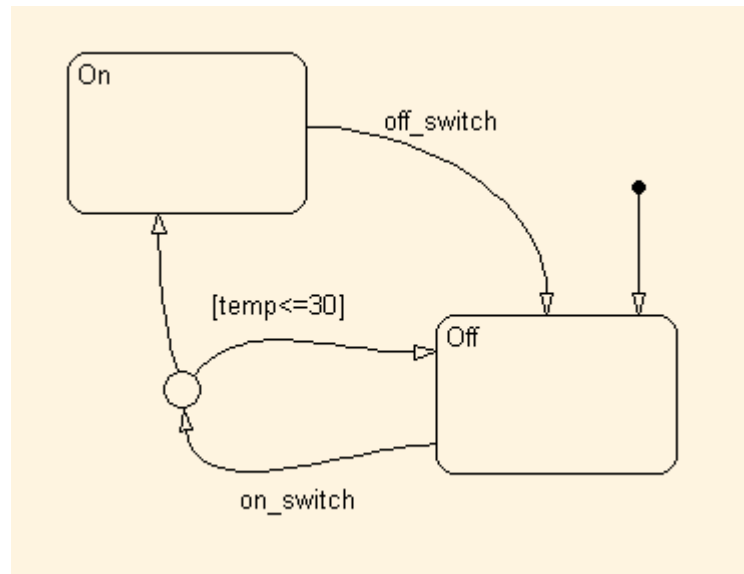
Зураг 3.18



Зураг 3.19



Зураг 3.20

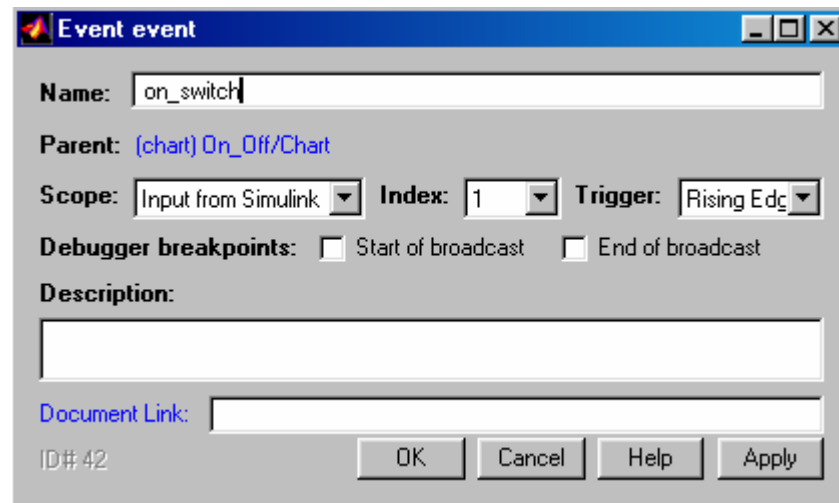


Зураг 3.21

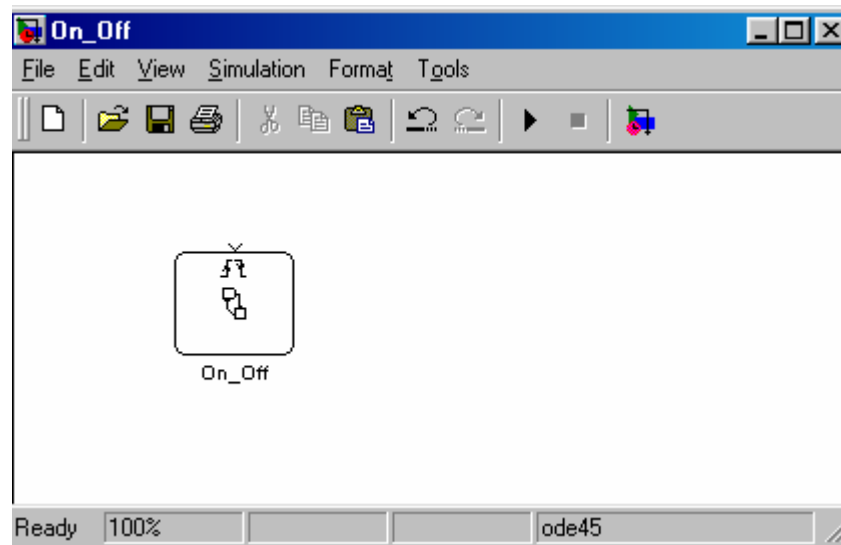
3.5.2 Үйл явцын (Event) оролтыг тодорхойлох

Загварчлалыг гүйцэтгэхээс өмнө On ба Off төлөвийн хоорондын шилжилт гүйцэтгэх on_switch ба off_switch үйл явцыг тодорхойлох шаардлагатай билээ. Үйл явц нь Stateflow диаграммын ажиллагааг удирддаг. Жишээлбэл, On төлөв идэвхитэй байхад on_switch үйл явц үүсвэл On төлөвөөс Off-төлөврүү шилжилт үүсч, Off төлөв идэвхитэй болдог. Иймд үүсгэсэн диаграммын оролтын үйл явцыг тодорхойлохын тулд :

1. Редакторын **Add** цэсээс **Event** дэд цэсийг сонгоно. Дэлгэцэнд гарах дэд цэсээс **Input from Simulink** -г сонгоно (Зураг 3.22) . Ингэж шинэ үйл явцын диалог үүснэ.
2. **Event properties** –ын **Name** талбарт on_switch гэж оруулна. Үүний дараа **Trigger**–г **Rising Edge** (өсөх фронт) гэж сонгож тааруулна. **Ok** товч дарж өөрчлөлтийг баталгаажуулна.
3. 1 ба 2 –р алхмыг off_switch үйл явцын хувьд давтан тодорхойлж, **Trigger** –г **Falling Edge** (буурах фронт) болгож өөрчлөнө. Ингэж бид Stateflow-ын on_switch ба off_switch үйл явцыг тодорхойлох ба Simulink загвар дахь Stateflow блок нь Зураг 3.23- т үзүүлсэн байдалтай дүрслэгдэнэ.



Зураг 3.22



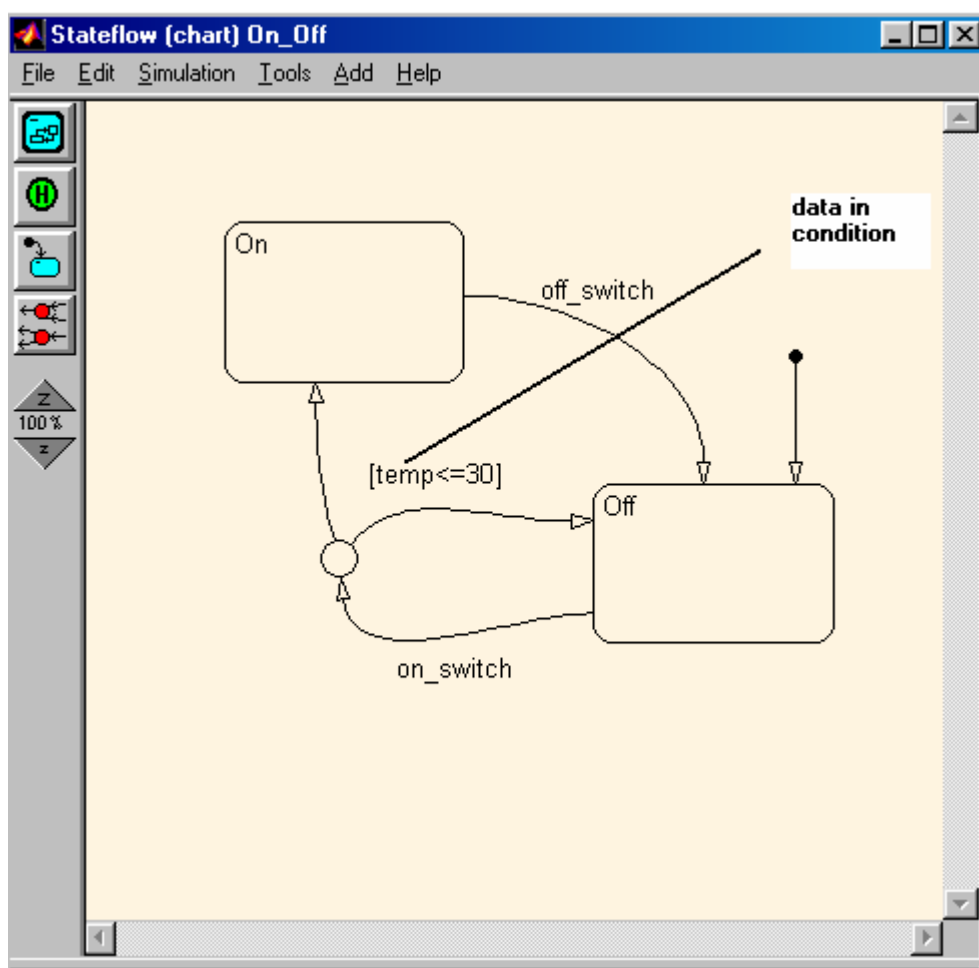
Зураг 3.23

3.5.3 Оролтын өгөгдлийг тодорхойлох

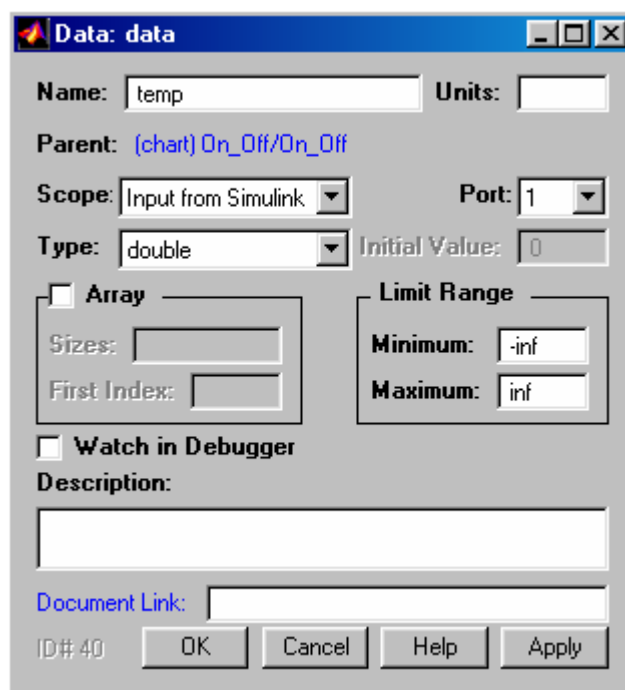
Зангилааны цэг ба off төлөв байдлын хоорондын шилжилт дээр Stateflow-ын **temp** хувьсахад тулгуурласан нөхцөл байгаа тул үүнийг диаграммд тодорхойлох ёстой билээ (Зураг 3.24). Temp-ийн оролтын өгөгдлийг тодорхойлохын тулд:

1. График редакторын **Add** цэсээс **Data**-г сонгоно. Үүссэн дэд цэсэнд **Input from Simulink** гэж сонголт хийнэ.

2. **Name** талбарт temp гэж оруулна. Бусад талбаруудыг стандарт утгатай нь үлдээнэ. **OK**– г сонгож өөрчлөлтийг баталгаажуулах ба цонхыг хаана (Зураг 3.25).



Зураг 3.24



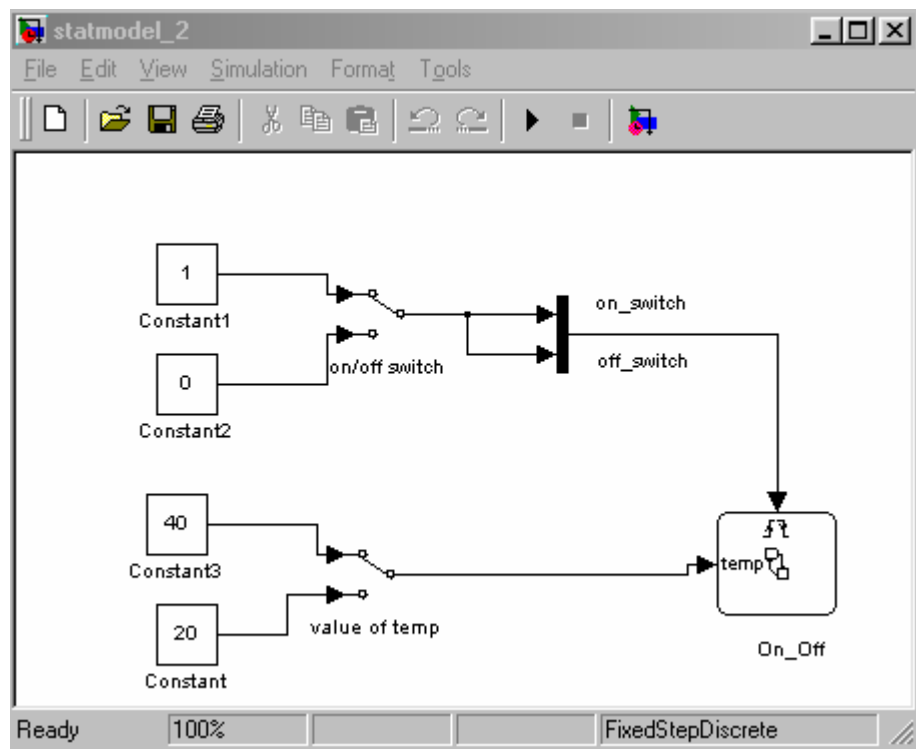
Зураг 3.25

3.5.4 Stateflow -ын холболтыг тодорхойлох

Stateflow – ын диаграммд үйл явц ба өгөгдлийг Input from Simulink гэж тодорхойлсон тул Simulink -ийн загвар , блок болон Stateflow–ын блокийн хоорондын үйл явц ба өгөгдлийн үүсгүүрийг тодорхойлох хэрэгтэй. Үүний тулд өмнө үүсгэсэн Stateflow -ын Chart-ыг ашиглан Simulink -ийн загварыг Зураг 3.26 -д заасан байдлаар үүсгэнэ. Энэ загварт temp хувьсахын авах утгыг гараар сэлгэн залгах түлхүүр ашиглан өөрчлөх тул Constant3=40 ба Constant=20 гэсэн утга бүхий тогтмол утгын хоёр үүсгүүр ашиглав. Мөн гараар сэлгэн залгах түлхүүрийг Switch Events болгон ашигласан тул түлхүүрийн гаралтыг зэрэгцээгээр Mux оролттой холбоно. Mux нь хоёр оролттой тул оролтын тоог мөн тодорхойлно. Mux–ын гаралтыг Stateflow Chart -ын On_Off блокийн оролттой холбоно. Загварчлал гүйцэтгэхийн тулд загварчлалын параметрыг Simulink- д хэрхэн тодорхойлохыг өмнө үзсэн билээ.

Stateflow диаграммыг шалгах

Stateflow диаграммын тэмдэг ба тэмдэглэгээг зөв болохыг график редакторын **Tools** цэсээс **Parse Diagram**-ыг сонгож шалгадаг. Ингэхэд гарсан мэдээлэл нь MATLAB Command Window -д гардаг. Диаграммд алдаа байвал улаан өнгөтэй товчлуур дүрсний дараа мэдээлэл гарах ба улаан өнгөтэй товчлуур гараагүй бол алдаагүй байгааг илэрхийлнэ.



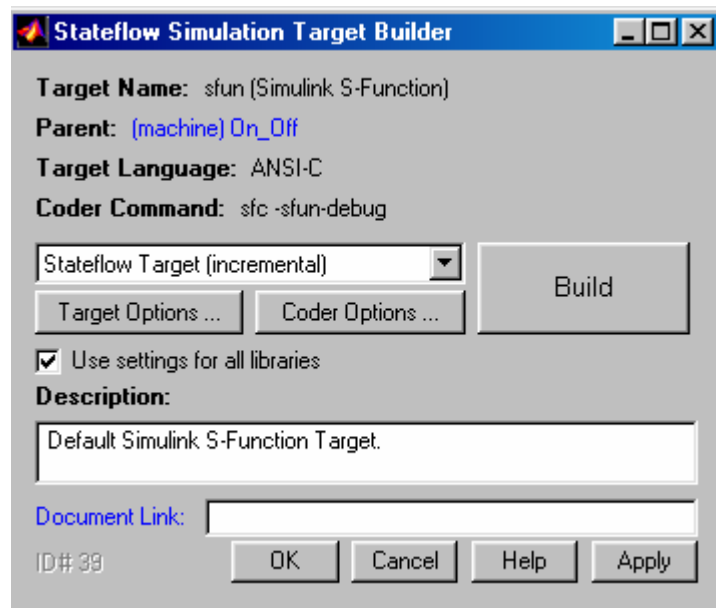
Зураг 3.26

Загварчлалын бэлтгэл хийх ба загварчлах

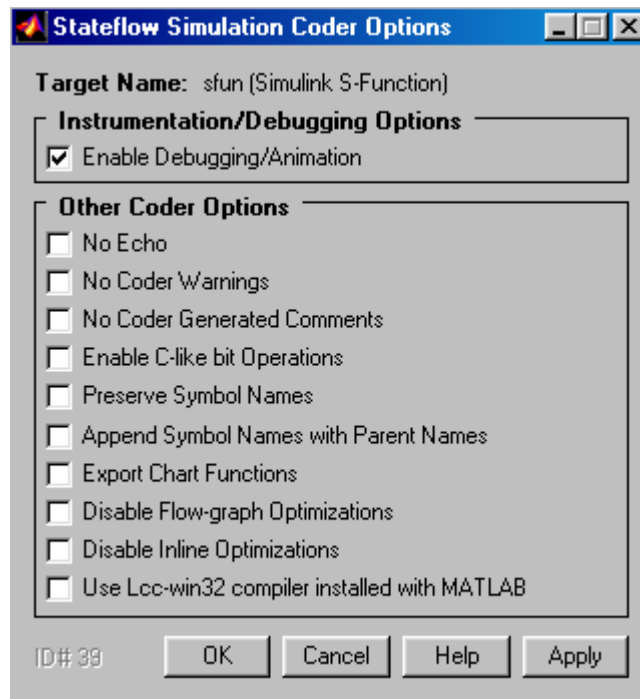
Stateflow-ын On_Off блокийг хоёр тогшиж Stateflow-ын диаграммыг гаргана. График редакторын **Tools** -оос **Open Simulation Target** -ыг сонгоно. Ингэхэд **Simulation Target Builder** гэсэн харилцах хайрцаг гардаг (Зураг 3.27). Энэ хайрцагт гарсан **Coder Options**-ыг сонгоход **Simulation Coder Options** (Зураг 3.28) хэмээх харилцах хайрцаг гарах ба энд **Enable Debugging/Animation** -ыг идэвхижүүлж, **OK** товчоор өөрчлөлтийг батламжлана. Үүний дараа **Simulation Target Builder** харилцах хайрцагийг хаана.

График редакторын **Tools** -оос **Debug** -ийг сонгоно. Энд **Animation** секцэд **Enabled** товчлуур сонгосон эсэхийг шалгах хэрэгтэй (Зураг 3.29). Ингээд **Close** -оор **Debug** – ийн цонхыг хаана. Үүгээр загварчлал хийх бэлтгэл хангагдлаа гэж үзэж болно.

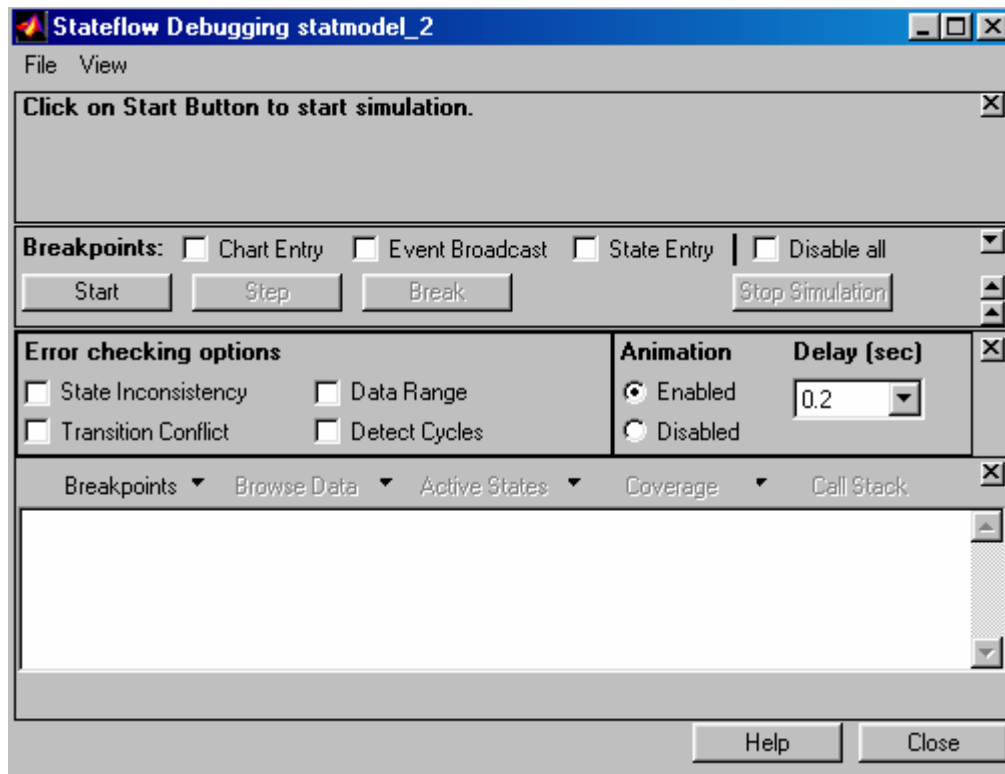
Загварчлалыг эхлүүлэхийн тулд диаграммын редакторын **Simulation** цэснээс **Start** -ыг сонгоно. Загварчлалын зөвхөн нэг үр дүнг Зураг 3.30 –д үзүүллээ. Зураг 3.26 –д үзүүлсэний дагуу on/off switch ба value of temp



Зураг 3.27

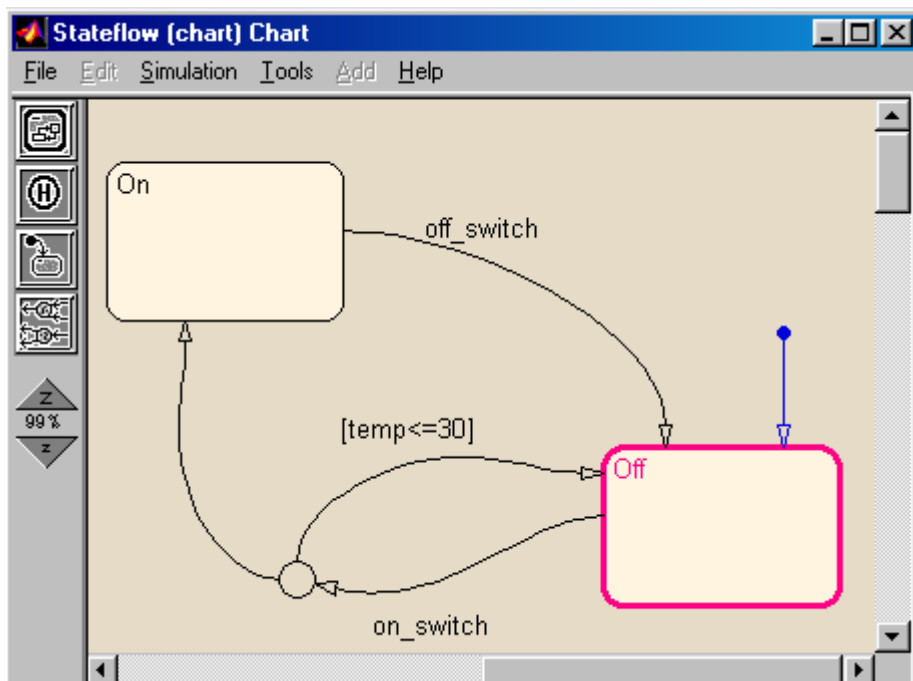


Зураг 3.28



Зураг 3.29

сэлгэн залгах түлхүүрийг янз бүрийн байрлалд тавьж (давхар тогших) загварчлалыг гүйцэтгэнэ. Жишээлбэл Зураг 3.30-д үзүүлсэн загварчлалын үр дүнгээс үзвэл төлөв байдал нь off (тод өргөн шулуунаар дүрслэгдсэн) байдалд оршиж байна.



Зураг 3.30

4 Matlab- д программ зохиох жишээ, дасгал

Дасгал №1

Матриц ба скалярын тухай бодлого

$$5x_1 + 2x_2 = 2$$

$$3x_1 - 2x_2 = 7$$

тэгшитгэлийг Matlab ашиглан тодорхойлох

Дасгал №2

Matlab-ын тусламжтай доорхи функцийн графикийг байгуулах:

$$x = A \cdot \sin(\sqrt{\omega^2 - \gamma^2} t + \alpha) \cdot e^{-\gamma t}$$

$$\omega = 10; \gamma = 0:1:10; A = 5; 0 \leq t \leq 3$$

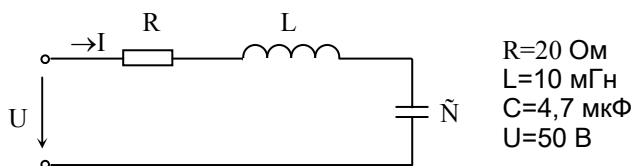
$$\alpha = \arccos(\gamma/\omega)$$

Дасгал №3

$n = 1:1:64$ байхад Z_1^n ба Z_2^n -г $Z_1 = 1.05e^{j2\pi/32}$ ба $Z_2 = 0.95e^{j2\pi/32}$ утгын хувьд зурах программ зохиох

Дасгал №4

Доорхи схемийн хувьд I гүйдлийн өөрчлөлтийг ω давтамжаас хамааруулж гаргах



Дасгал №5

$Z = e^{-(x^2+y^2)/2}$ функцийг тооцоолж, график гаргах программ зохиох

Дасгал №6

$$P(S) = a_n S^n + a_{n-1} S^{n-1} + \dots + a_2 S^2 + a_1 S + a_0$$

полиномын коэффициентыг

$$P = [a_n \ a_{n-1} \ \dots \ a_2 \ a_1 \ a_0];$$

хэлбэрээр бичдэг.

Өгөгдсөн нь:

$$P1 = S^4 + 3S^3 + 5S^2 + 10S + 6$$

$$P2 = S^2 + 8S + 3$$

Олох нь:

- $P1(s)$ ба $P2(s)$ -ийн нойл утга (шийдийг) олох
- $P1$ ба $P2$ -ийн утгыг $s=2$ байхад олох
- $P3 = P1 * P2$ полиномыг тооцоолох

Дасгал 7

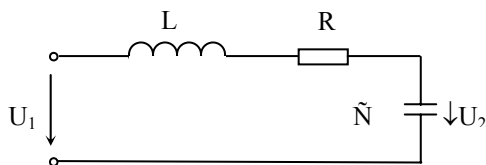
$N(z) = z^2 + k * z + 0.25$ полиномын тэг цэгийг (шийд) k параметраас хамааруулж тодорхойлох

Дасгал №8

RLC –хэлбэлзлийн хүрээний дамжуулалтын функц

$$G(S) = \frac{U_2(S)}{U_1(S)} = \frac{1}{S^2 LC + S \cdot RC + 1} \quad \text{бол шилжилтийн тодорхойломж ба давтамжийн}$$

тодорхойломжийг тооцоолох
 Үүнд: $R=0.8$ Ом, $L=1$ мкГн, $C=1$ мкФ



Гэхдээ хугацаа, давтамжийн хязгаар ба диаграммын хуваарийг программд шууд тодорхойлно.

Дасгал №9

Дээрхи (Дасгал NN№8) дасгалд хугацаа, давтамжийн хязгаар ба диаграммын хуваарийг хэрэглэгчээс (программ зохиогчоос) оруулахаар программыг өөрчлөх

Дасгал №10

Дамжуулалтын z-функцийг хугацааны хувьд тасралтгүй s –дамжуулалтын функц шиг өндөр зэрэгтэй гишүүн z –ээр эхэлсэн мэт полиномоор тодорхойлжээ. Matlab нь хугацааны тасралтгүй ба тасралттай дамжуулалтын функцийг ялгаж чаддаггүй тул шилжилтийн тодорхойломжийн step функцийн оронд dstep функц, импульсийн тодорхойломжид dimpulse, оролтын дурын сигналын мэдрэмжийг (тодорхойломж) dsim, давтамжийн тодорхойломжийг dbode функцээр тус тус тодорхойлдог. Дискрет хугацааны шаталсан функц plot-ын оронд stairs функцийг хэрэглэдэг (тохируулагч үйлчлэлийг stairs, тохируулагдах хэмжигдхүүнд stairs буюу plot-г хэрэглэдэг). Тэгвэл хугацааны тасралттай $G(Z) = \frac{0.1}{Z - 0.9}$ системийн шилжилтийн

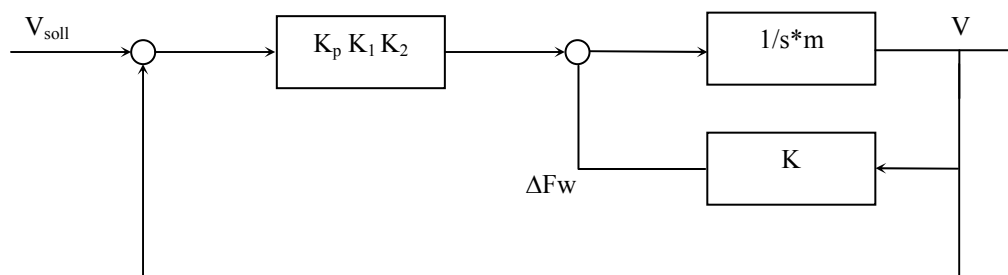
тодорхойломжийг гаргах

Анхаар:

Z ба S –дамжуулалтын функцийг нэг программд хольж хэрэглэж болохгүй

Дасгал №11

Хөдөлгөөний хурдыг тохируулах шугамчилсан блок-схем доорхи байдлаар өгөгджээ.



Үүнд:

$$m=1100\text{кг}, K_p \cdot K_1 \cdot K_2 = \frac{55\text{кВт}}{170\text{км/цаг} \cdot 40\text{км/цаг}} = 105 \frac{\text{Вт}}{(\text{м/сек})^2}$$

$$K=2 \cdot C \cdot V_0 = 2 \cdot \frac{55\text{кВт}}{(170\text{км/цаг})^3} \cdot 80\text{км/цаг} = 23 \frac{\text{Вт}}{(\text{м/сек})^2}$$

Анхны хурд $V_0=60$ км/цаг –аас эхлэн $V_{\text{sold}}=100$ км/цаг хүрэх хурдатгалын процессыг тодорхойлох. Үүнд:

Дамжуулалтын функц

$$G(C) = \frac{V(S)}{V_{sold}(S)} = \frac{K_p K_1 K_2}{K + K_p K_1 K_2} \cdot \frac{1}{1 + S \cdot \frac{m}{K + K_p K_1 K_2}}$$

гэж өгөгджээ. Feedback, series,

сloop ба parallel функцийг хэрэглэнэ үү?

4.1 Matlab – д программ зохиох жишээ , дасгалын хариу

Matlab – ын жишээ дасгалуудын шийд программ , графикийг доорхи хэсэгт үзүүлээ. Эдгээр программыг анхааралтай судалж , туршиж үзвэл Matlab- д программ бичиж сурахад тус нэмэр болно гэж үзэж байна.

Хэрэв танд программд бичсэн команд, оператор ойлгомжгүй байвал Matlab- ын Help-ийг ашиглаарай. help <команд буюу оператор> гэж бичнэ.

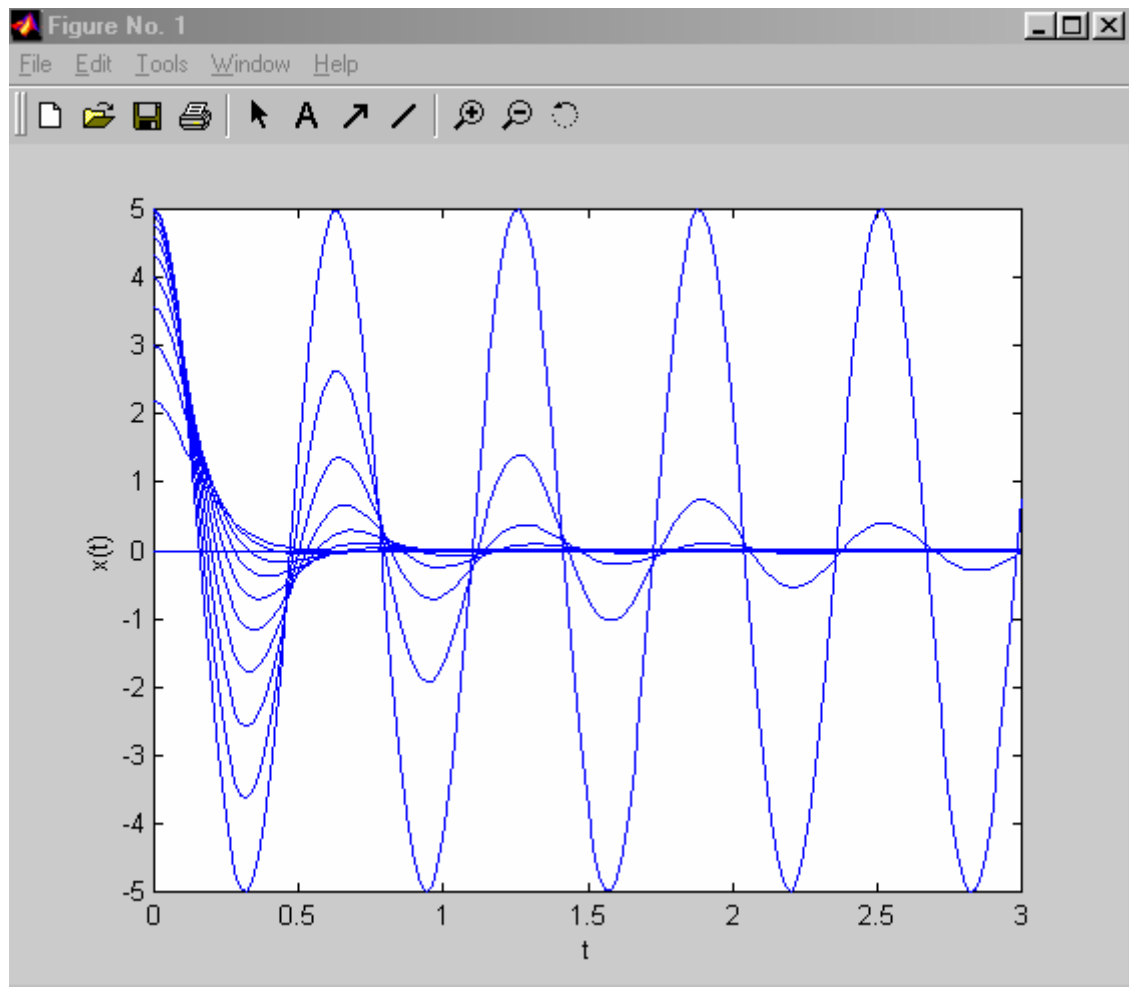
Дасгал №1

```
% dasgal1.m
% Matrix and Skalars
% 5x1+ 2x2=2
% 3x1-2x2=7
% A*X=B

A=[5 2 ;3 -2];
B=[2;7];
%X=inv(A) * B inv(A) ni A matriziin urvuu matrix
X=inv(A)*B
```

Дасгал №2

```
%dasgal_2
A=5;
omega=10;
t=0:0.02:3
for gamma =0:1:10
    alpha=acos(gamma/omega);
    x=A*sin(sqrt(omega^2-gamma^2)*t+alpha).*exp(-gamma*t);
    plot(t,x)
    hold on
end
xlabel('t')
ylabel('x(t)')
```



Зураг 1: Дасгал 2-ын хариу график

Дасгал №3

```
% dasgal_3.m
```

```
n=1:1:64;
```

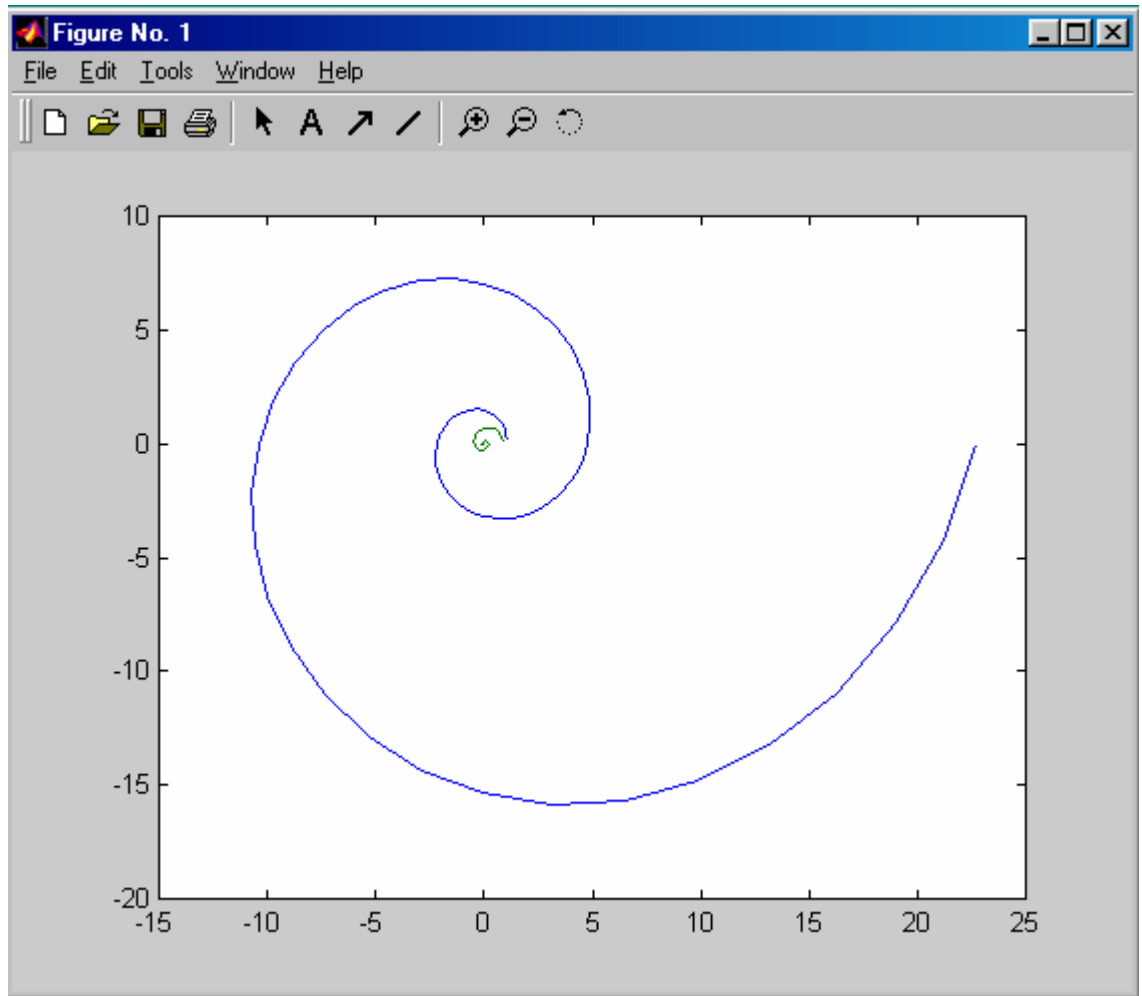
```
z1=1.05*exp(j*2*pi/32);
```

```
z2=0.95*exp(j*2*pi/32);
```

```
y1=z1.^n;
```

```
y2=power(z2,n);
```

```
plot(real(y1),imag(y1),real(y2),imag(y2))
```

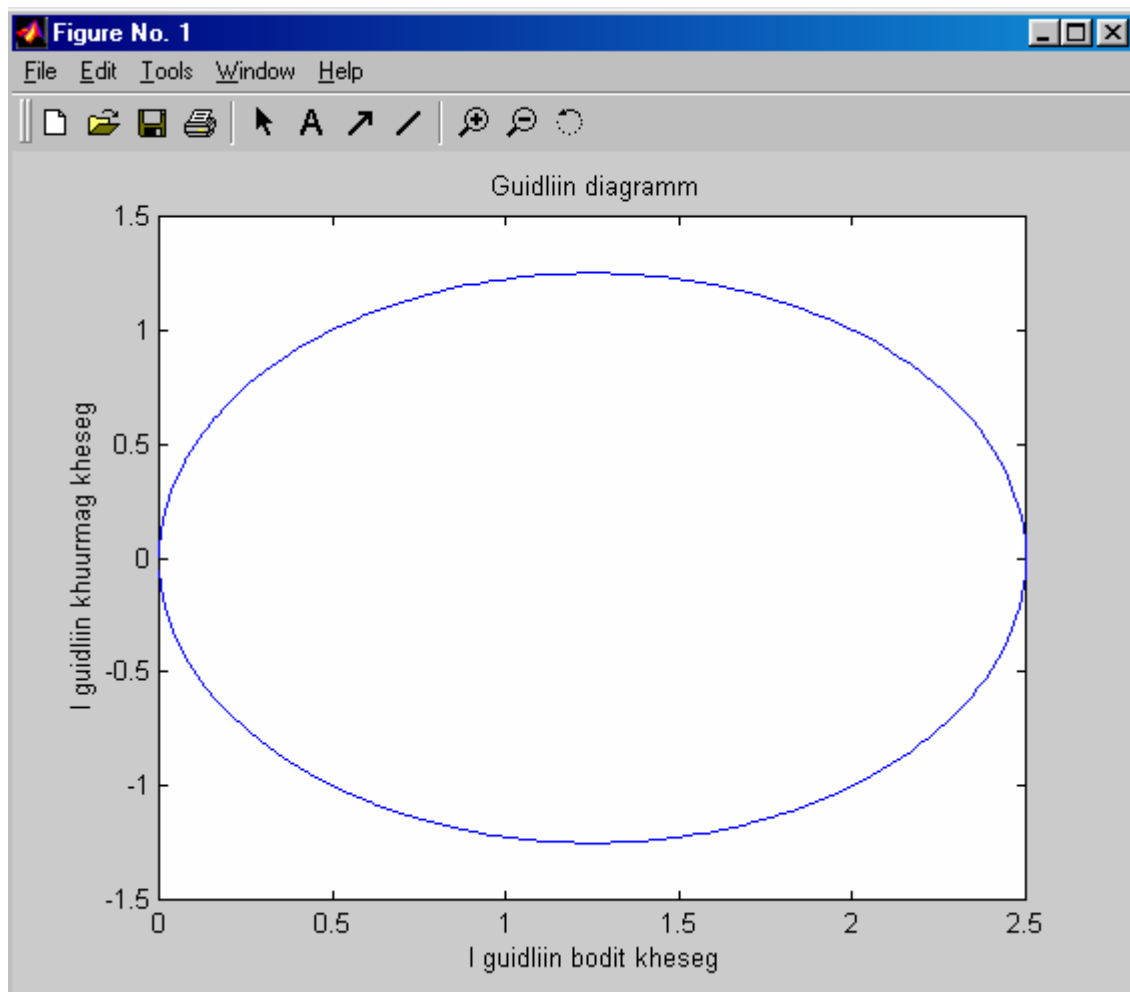


Зураг 2 : Дасгал 3-ын хариу

Дасгал №4

```
% dasgal_4
```

```
U=50; R=20; L=0.01; C=4.7e-6;  
omega=1e-6:20:1e5;  
I=U./(R+j*omega.*L+1./(j*omega.*C));  
plot(I);  
xlabel(' I guidliin bodit kheseg');  
ylabel(' I guidliin khuurmag kheseg')  
title(' Guidliin diagramm')
```



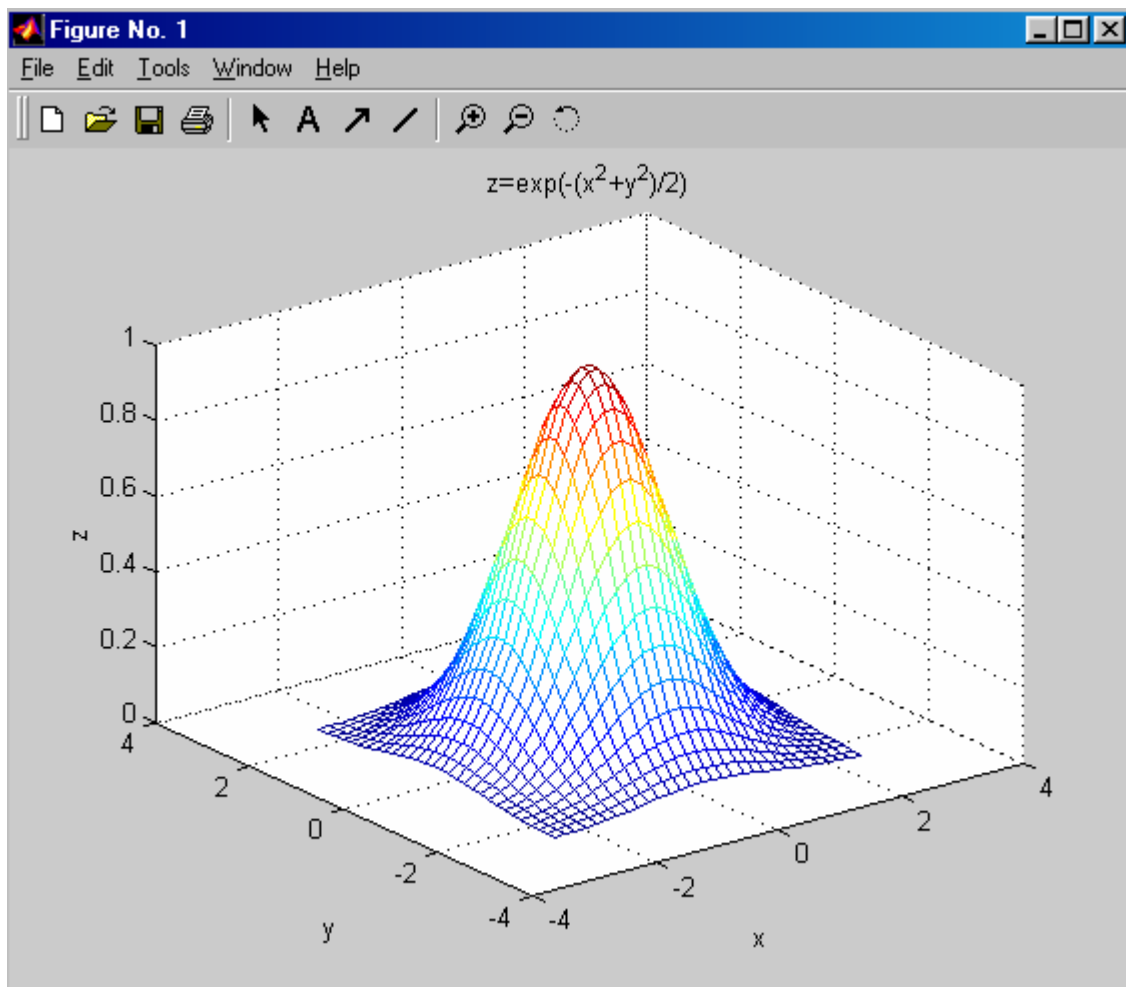
Зураг 3 : Дасгал 4-ын хариу

Дасгал №5`% dasgal_5.m`

```

x=linspace(-2.5,2.5,30);
y=linspace(-2.5,2.5,30);
[X,Y]=meshgrid(x,y);
z=exp(-(X.^2+Y.^2)/2);
mesh(X,Y,z)
title('z=exp(-(x^2+y^2)/2)');
xlabel('x')
ylabel('y')
zlabel('z')

```



Зураг 4 : Дасгал 5-ын хариу

Дасгал №6`% dasgal_6`

```

p1=[1 3 5 10 6];p2=[1 8 3];
n1= roots(p1) %p1- iin shiid olokh
n2=roots(p2) %p2-iin shiid olokh
p1=polyval(p1,2) %s=2 baikh p1-iin utga
p2=polyval(p2,2) %s=2 baikh p2-iin utga
p3=conv(p1,p2) % p3 -g p1*p2 gej urjij gargana

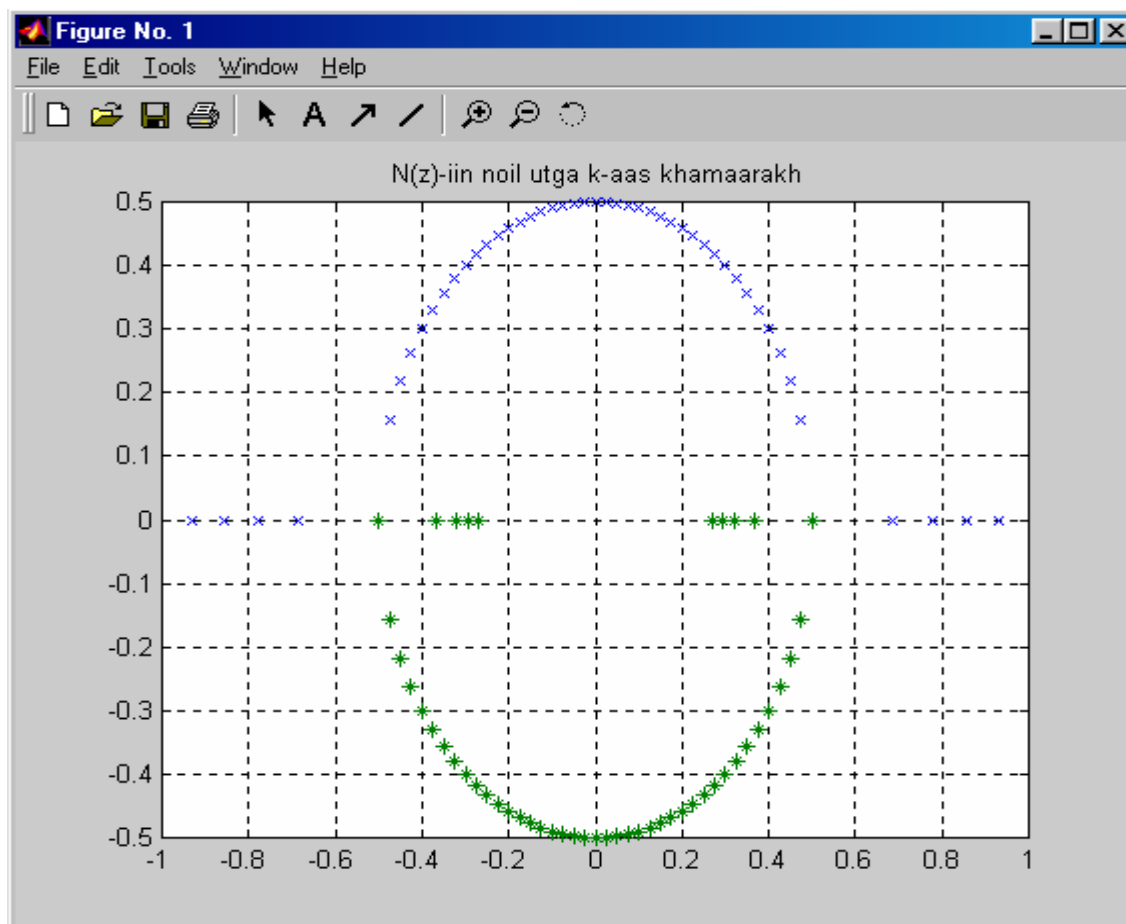
```

Дасгал №7

```

%dasgal_7
for k=-1.2:0.05:1.2
    N= [1 k 0.25];
    n=roots(N)
plot(real(n(1)),imag(n(1)),'x',real(n(2)),imag(n(2)),'*')
hold on
grid
title(' N(z)-iin noil utga k-aas khamaarakh')
end

```



Зураг 5: Дасгал 7 –ийн хариу

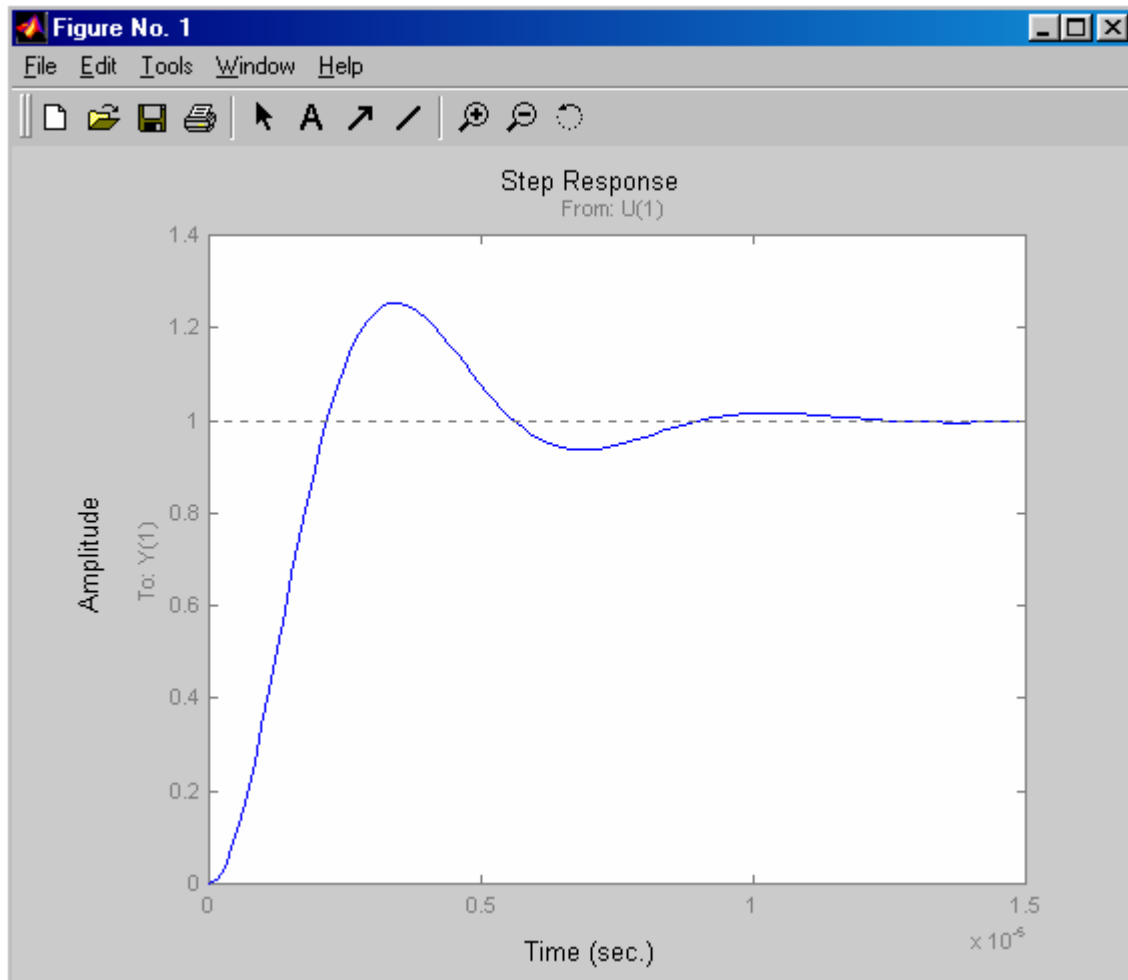
Дасгал №8

```

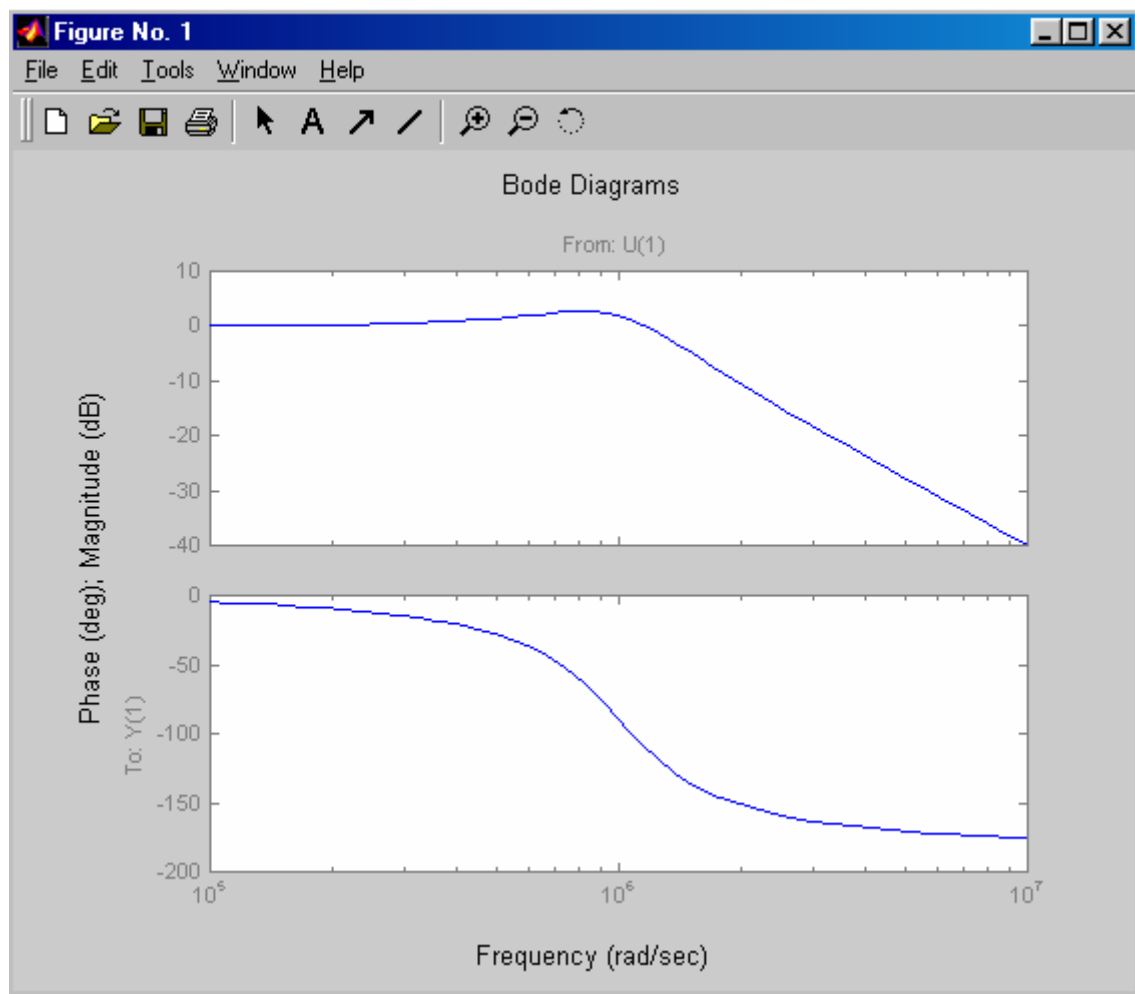
%dasgal_8
%rlc Khelbelzliin khuree
R=0.8;
L=1e-6;
C=1e-6;
Z=[1];
N=[L*C R*C 1];

```

```
step(Z,N);  
pause;  
bode(Z,N);
```



Зураг 6: Дасгал 8-ын хариу - Хугацааны тодорхойломж буюу огцом мэдрэмж



Зураг 7: Дасгал 8 – Амплитуд ба фазын тодорхойломж

Дасгал №9

```
% dasgal_9
```

```
%rlc2 : RLC-Khelbelzliin khuree
```

```
R=0.8;
```

```
L=1e-6;
```

```
C=1e-6;
```

```
Z=[1];
```

```
N=[L*C R*C 1];
```

```
Tmax=input('Tmax=');%Khugazaanii mujiig Keyboard-aas oruulakh (20 mksec)
```

```
t=0:sqrt(L*C)/100:Tmax;
```

```
y=step(Z,N,t);
```

```
figure(1);
```

```
plot(t,y);
```

```
axis([0 Tmax 0 1.5]);
```

```
figure(2);
```

```
analyse(y,t);
```

```
w=logspace(4,8);
```

```
[Betrag,Phase]=bode(Z,N,w);
```

```
figure(3);
```

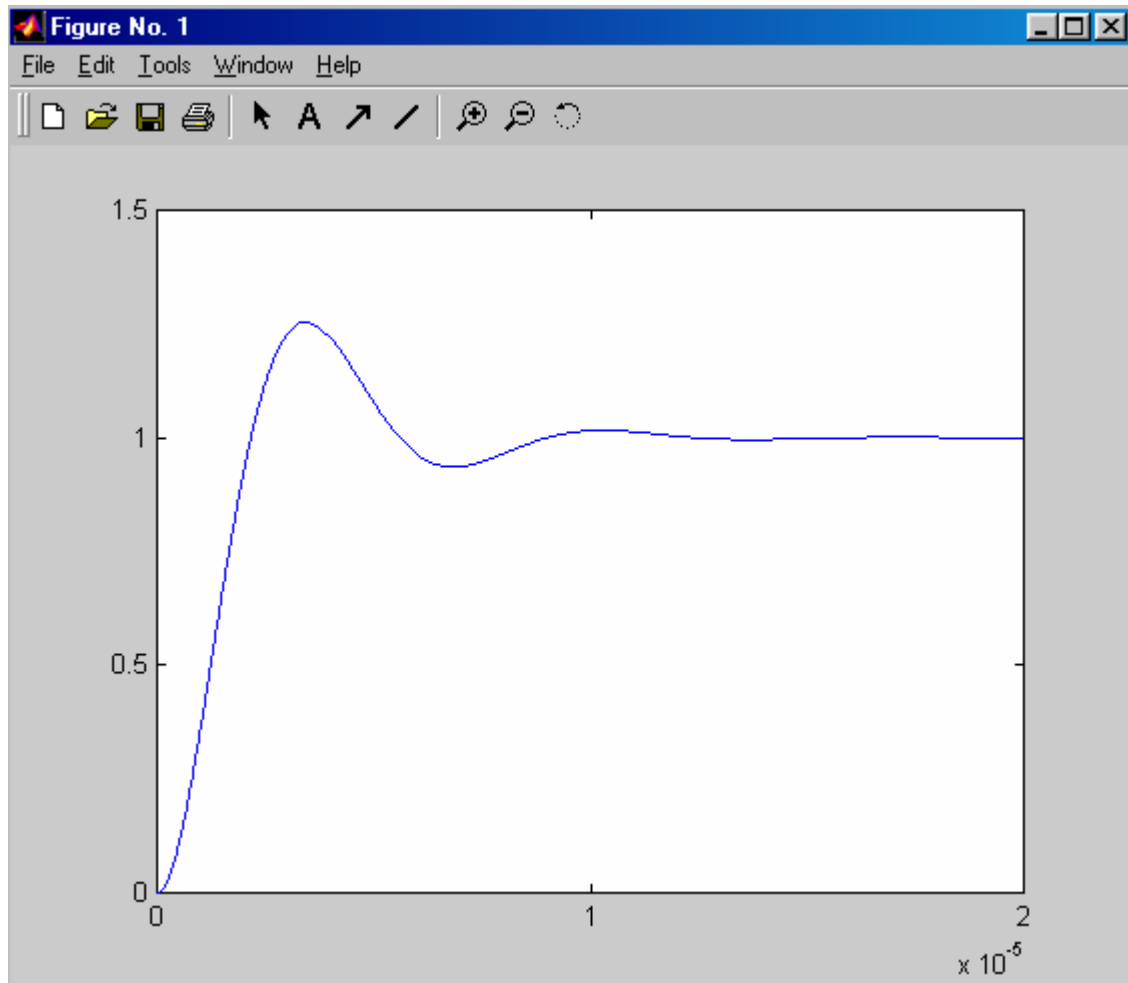
```
subplot(2,1,1);
```

```
loglog(w,Betrag);
```

```
grid;
```

```
subplot(2,1,2);
```

```
semilogx(w,Phase);
grid;
```

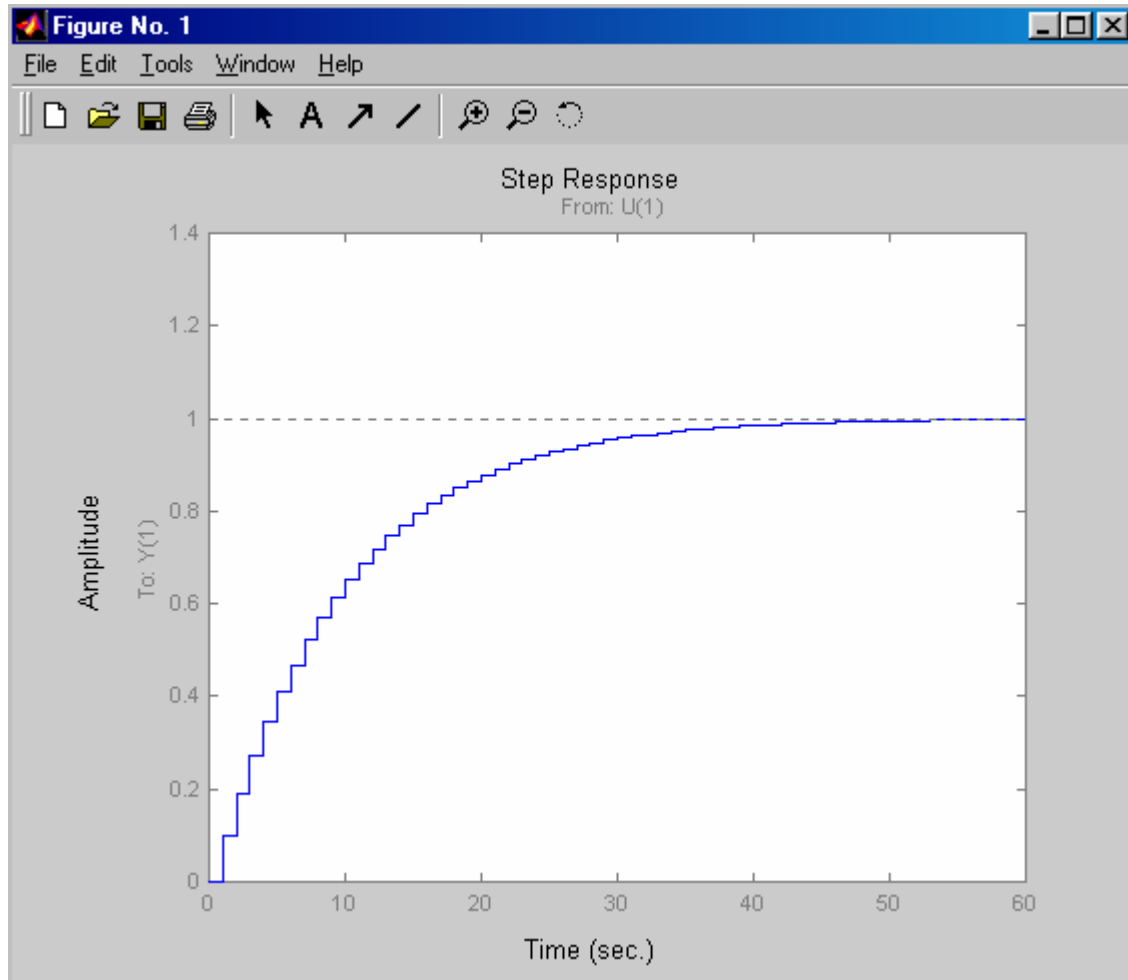


Зураг 8: Дасгал 9-ын хариу

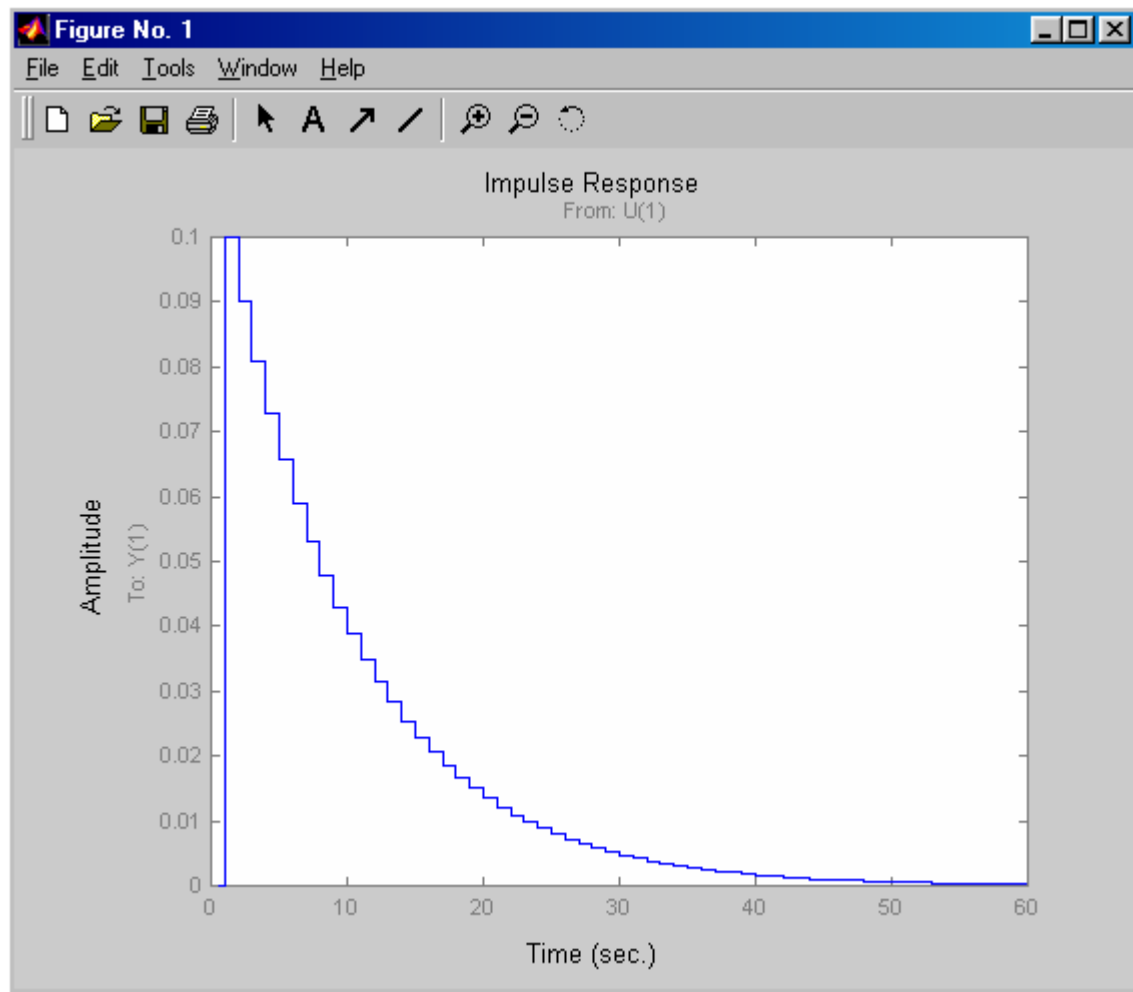
Дасгал №10

```
% dasgal_10
```

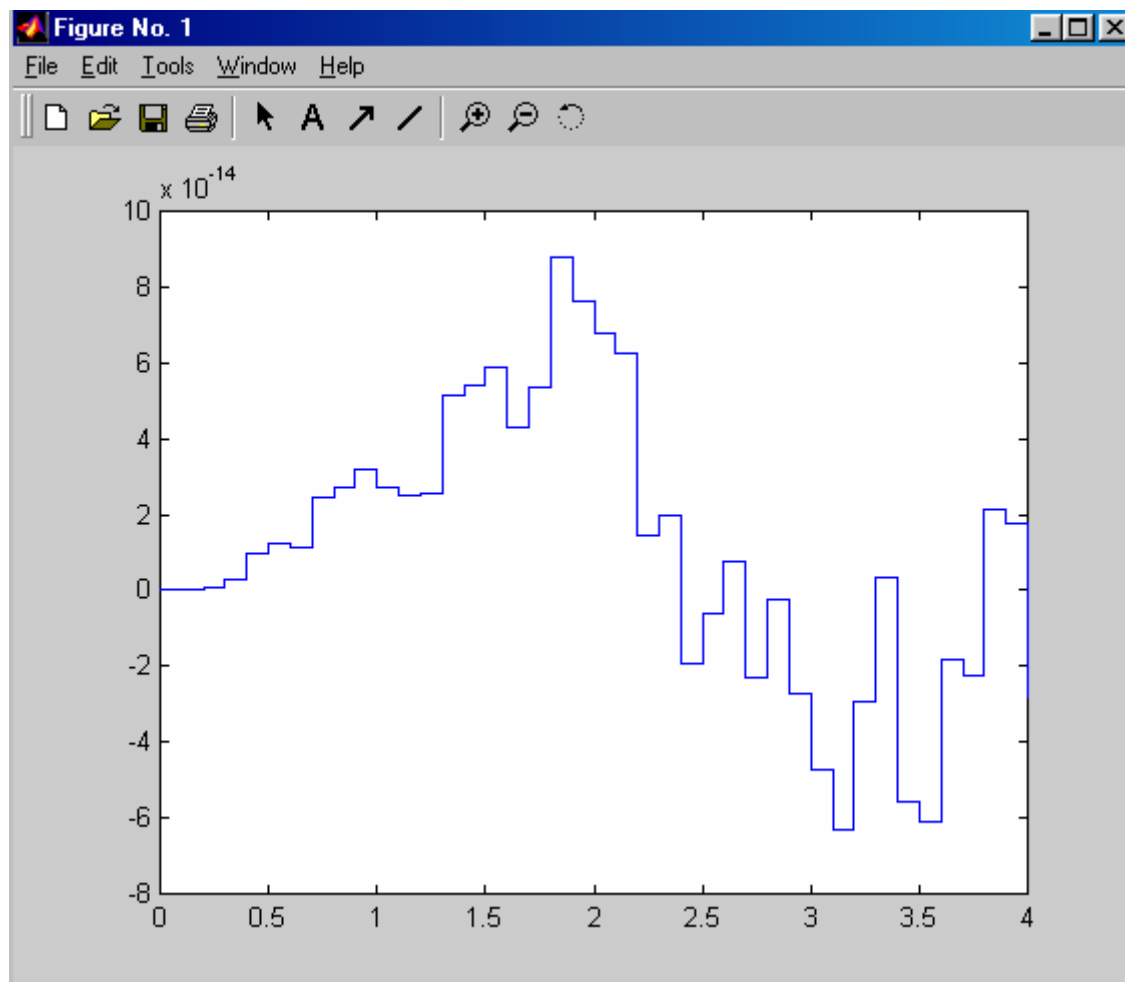
```
Z=[0 0.1];
N=[1 -0.9];
dstep(Z,N);
pause;
dimpulse(Z,N);
pause;
f=200;
T=0.1;
t=0:T:4;
u=sin(2*pi*f*t);
y=dlsim(Z,N,u);
stairs(t,y);
pause;
dbode(Z,N,T);
```



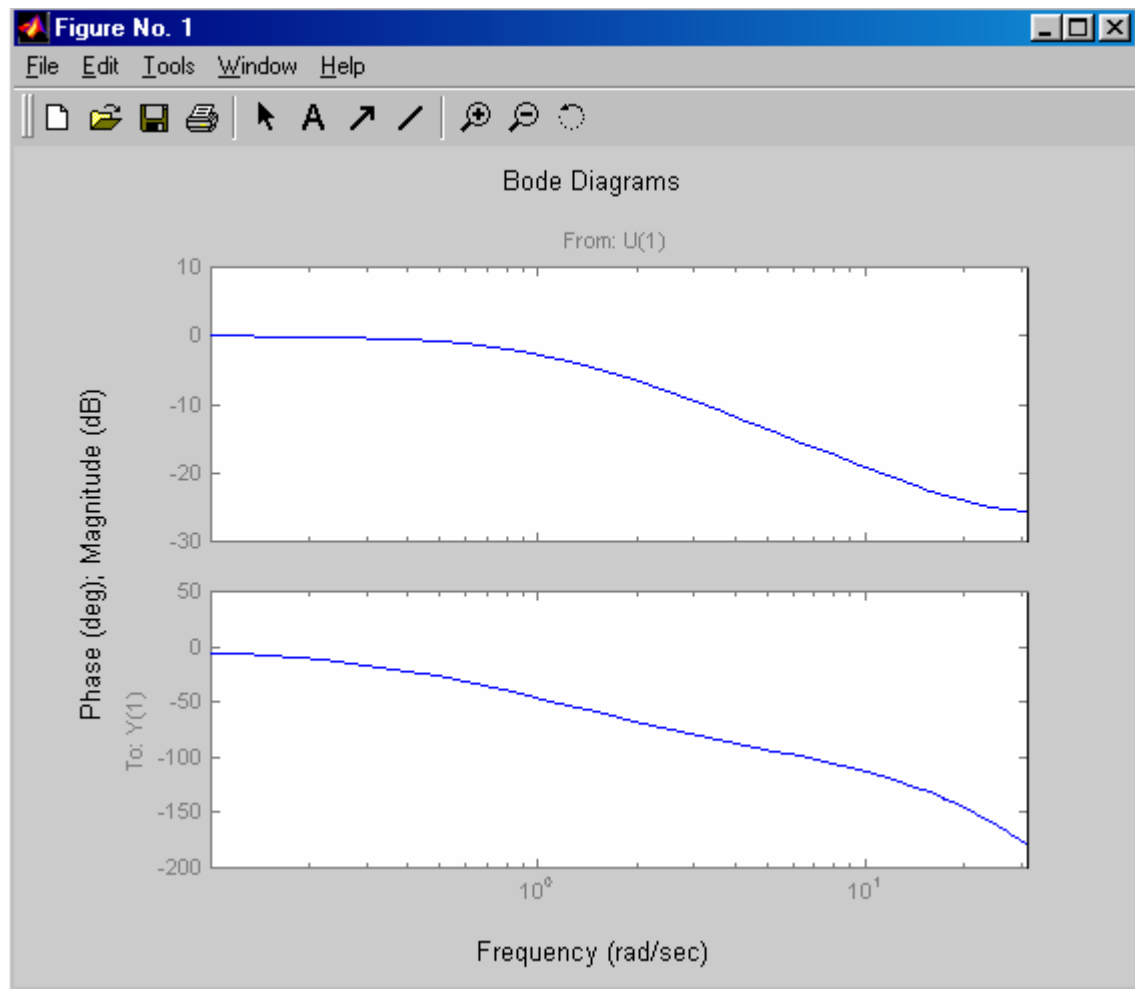
Зураг 9: Дасгал 10 -ын хариу : Хугацааны тодорхойломж



Зураг 10: Дасгал 10 –ын хариу: Импульсийн тодорхойломж



Зураг 11: Дасгал 10 –ын хариу: Хугацааны дискрет функц



Зураг 12: Дасгал 10 –ын хариу: **Z** - дамжуулалтын функцийн давтамжийн тодорхойломж

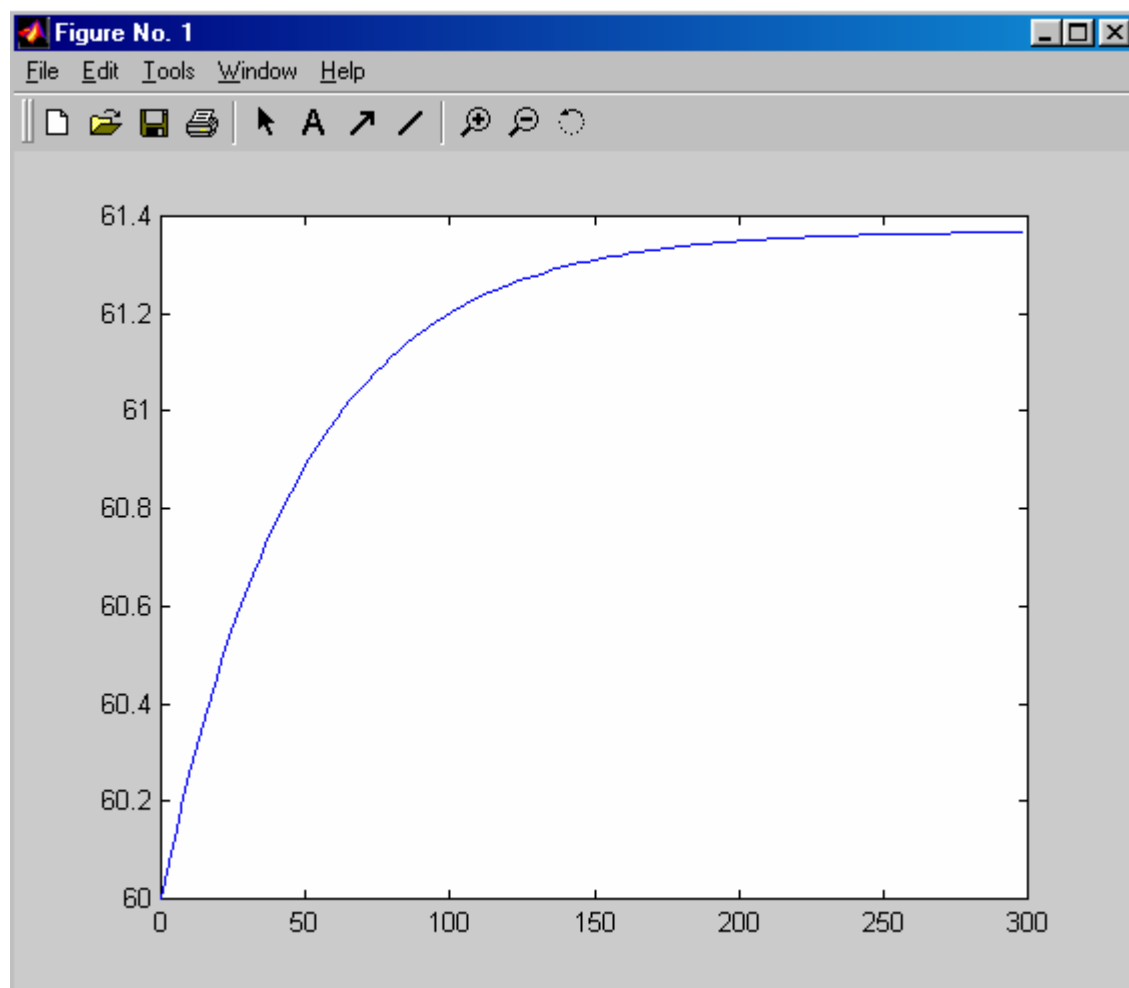
Дасгал №11

`%dasgal_11`

```

Z1=[105]; N1=[1];%P-element
Z2=[1];      %I-element
N2=[1 100 0];
Z3=[23]; N3=[1];
[Z0,N0]= feedback(Z2,N2,Z3,N3,-1)
[Z0, N0]=series(Z1,N1,Z0,N0);
[Z,N]=cloop(Z0,N0,-1);
[y,x,t]=step(Z,N0);
dV=0.3;
v0=60;
plot(t,y*dV+v0);

```



Зураг 13: Дасгал 11-ын хариу:

5 Simulink-ийг хэрэглэх жишээ

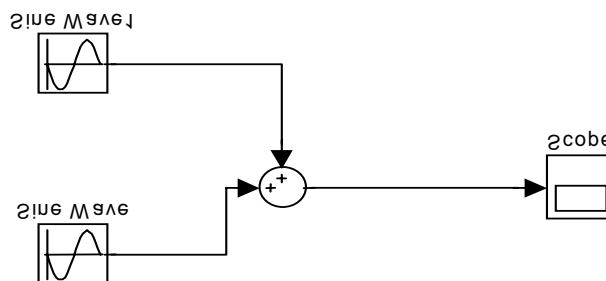
Жишээ 1

Синус хэлбэрийн хоёр сигналыг нэмж, янз бүрийн давтамжинд туршиж, осциллографт үр дүнг гаргах

Бүтэц:

Шаардлагатай блокийг блокийн сангаас маузаар зөөнө.

- Sources -оос синус сигнал үүсгэх Sine блокийг 2 –ыг зөөнө
 - Math - оос оролтыг нийлбэрлэх Sum блокийг зөөнө
 - Sinks - оос сигналын графикийг харуулах Scope блокийг зөөнө
- Оролт ба гаралтыг доорхи байдлаар холбоно (Зураг 1.1).



Хэлдэлзэл

Зураг 1.1

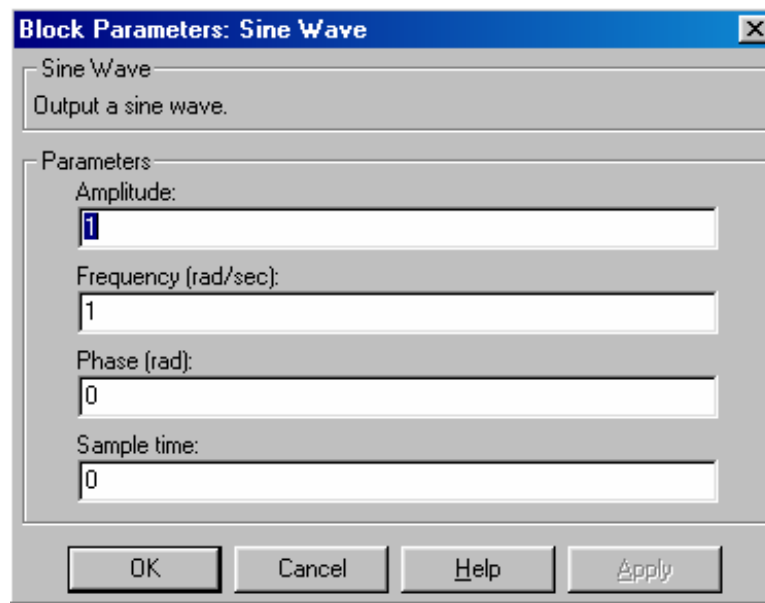
Хэрэв Sine Wave блок дээр хоёр дахин тогшвол (click) доор дүрсэлсэн параметрын цонх гарна (Зураг 1. 2). Энд Frequency -г 1,1 болгож өөрчлөх ба Ok -д тогшиж параметрын утгыг өгнө.

Мөн Scope блок дээр хоёр дахин тогшиж осциллографын цонхыг нээнэ (Зураг 1. 3).

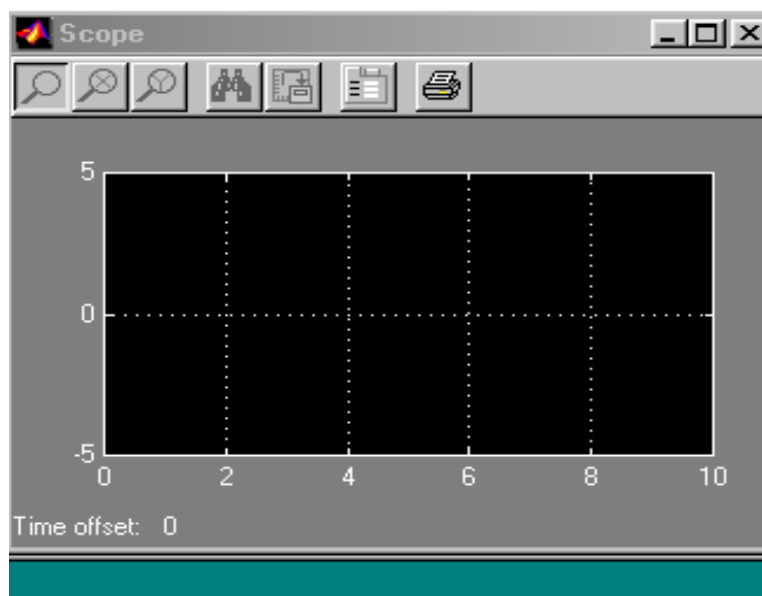
ЗАГВАРЧЛАХ:

Загварчлалыг Simulation / Start мөнүгээр ажиллуулна. Загварчлал нь 10 сек-ын дараа өөрөө зогсоно. Үр дүнг осциллограф дээр гаргана (Зураг 1.4).

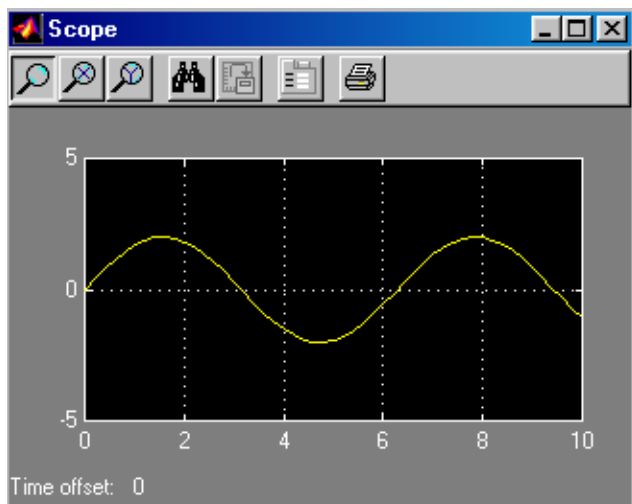
Загварчлалын хугацааг уртасгаж, 100 сек болгож (Зураг 1.5-ын Stop time) Ok гэж батламжлан загварчлалыг дахин ажиллуулах (Start даргах)



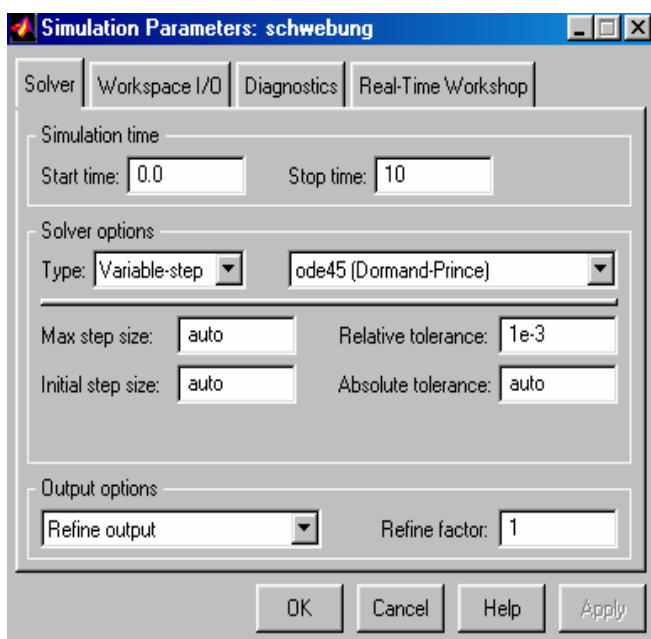
Зураг 1.2



Зураг 1.3



Зураг 1.4



Зураг 1.5

Ингэж загварчлахад хурц үзүүртэй хэлбэлзэл гарч байна (Зураг 1.6).

Шалтгаан нь:

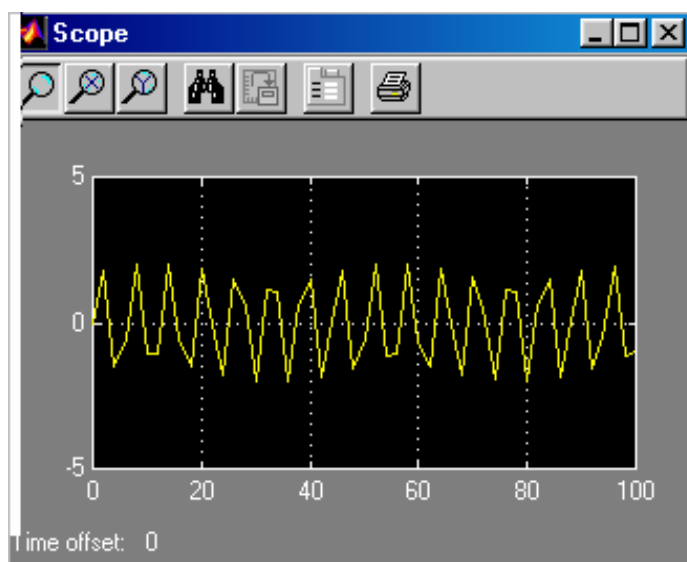
Жигд дүрс гаргахад гаралтын утга хэтэрхий бага байна.

Сайжруулахын тулд Simulation Parameters дахь Refine Factor -ыг 10 болгож ихэсгэнэ (Зураг 1.7).

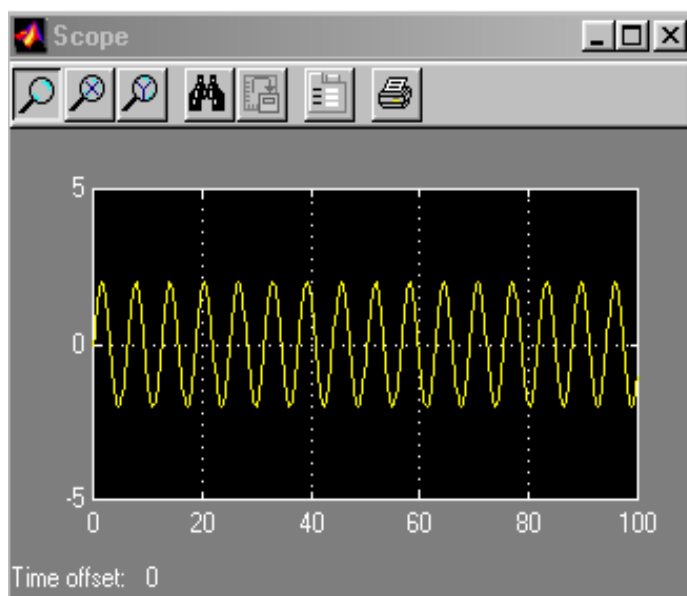


Осциллографын хуваарийг оновчтой зохицуулахын тулд түүн дээр дарна (Зураг 1.8).

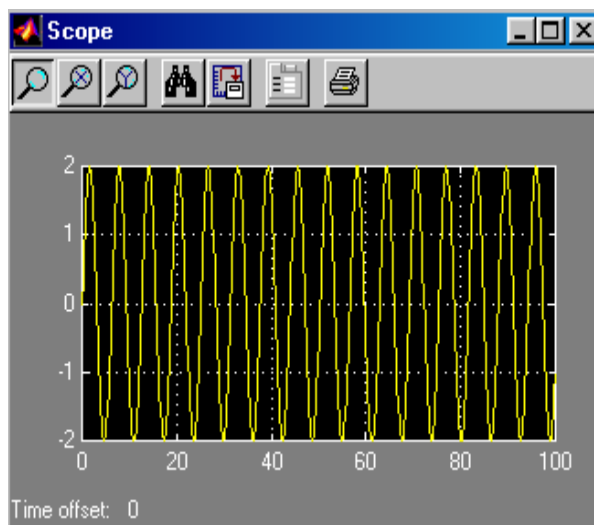
-ыг ашиглаж ,



Зураг 1. 6



Зураг 1. 7



Зураг 1.8

Үр дүнг гаргах:

1. Osci – блокод



Үүний тулд блокт параметрыг тогшиж параметрын цонхыг нээнэ. Дараа нь зургийн карт Data history-г сонгоно (Зураг 1.9).

Save data to workspace –г тогшино. Загварчлалыг ажиллуулахад муруйн график нь осциллографын өгөгдөл хэлбэрээр Matlab-д агуулагдана (Зураг 1.10).

2. Matlab-ын Workspace –д хадгалах

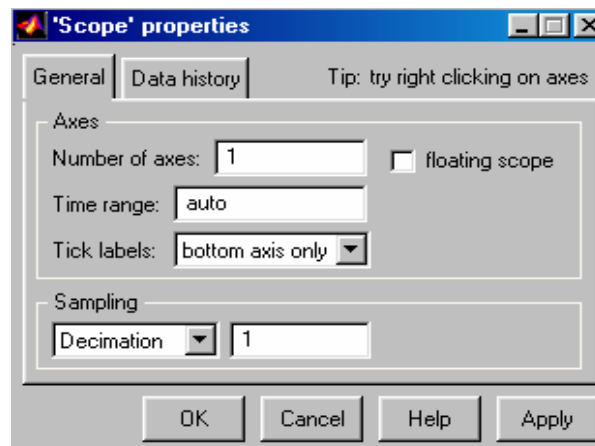
Sinks-ээс simout блокийг авна. simout блокийн гаралттай түүний оролтыг холбоно. Үүний тулд маузын баруун талын товчийг дарж байгаад холболтын салаалалт хийнэ. Загварчлалыг ажиллуулахад simout хэлбэрээр Matlab –д хадгалагдана.

(Зураг 1.12)

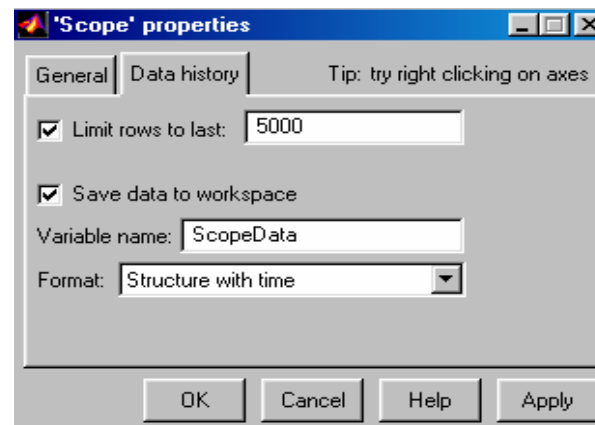
3. Файлд хадгалах

Sink – ийн To File –г Sum блокийн гаралттай холбоно (Зураг 1.13). Ингээд параметрийн цонх дээр хоёр дахиж тогшино. Файлын нэрээ schwebung.mat болгож өөрчлөнө (Зураг 1.11). Блокийн хэмжээг татах замаар буланд тааруулж болно. Дахин загварчлал хийхэд өгөгдлүүд нь schwebung.mat нэртэй файлд хадгалагдана.

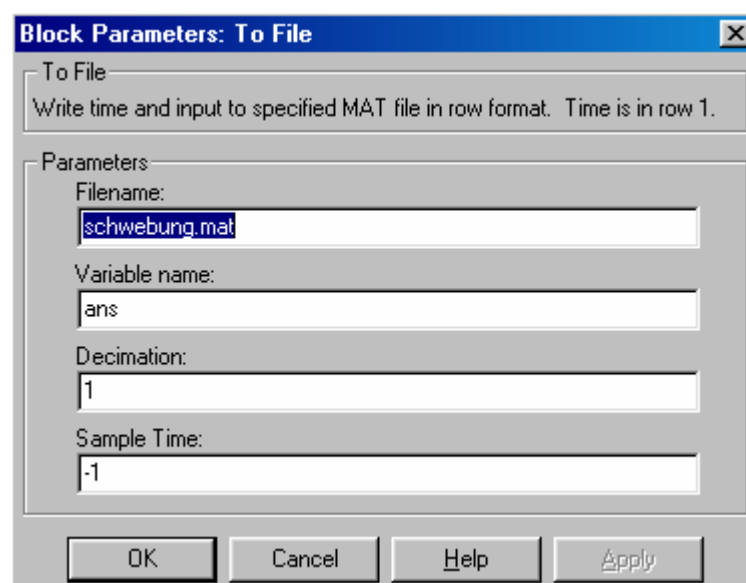
Ингэж ажиллахад бэлэн загвар үүсгэнэ.



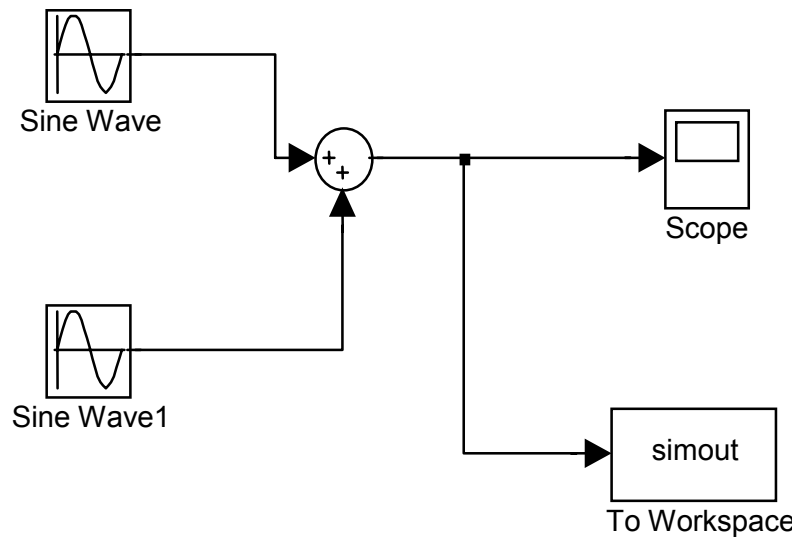
Зураг 1.9



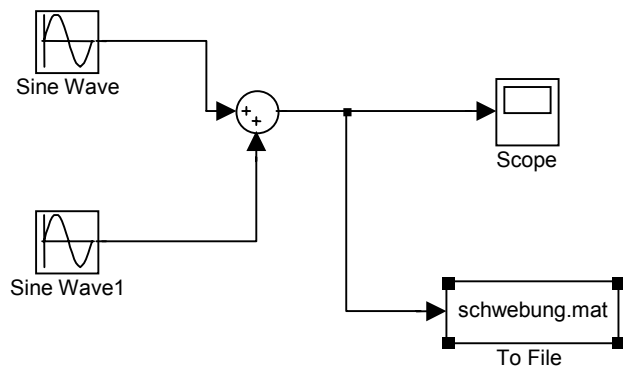
Зураг 1.10



Зураг 1.11



Зураг 1.12



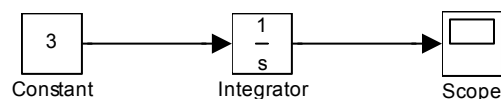
Зураг 1.13

Жишээ 2

Хугацаанаас хамаарах функцийн интегралыг тодорхойлох. Доорхи загварыг авч үзье.

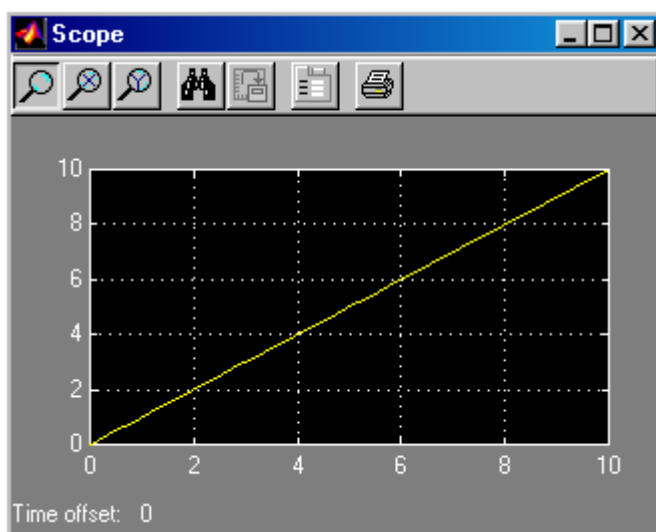
Constant блокийг Source -ээс ,Integrator блокийг Continuous -ээс тус тус авна.

Integrieren



Зураг 2.1

Загварчлалыг ажиллуулж Osci –ийн Autoscale хийсний дараа доорхи график гарна.



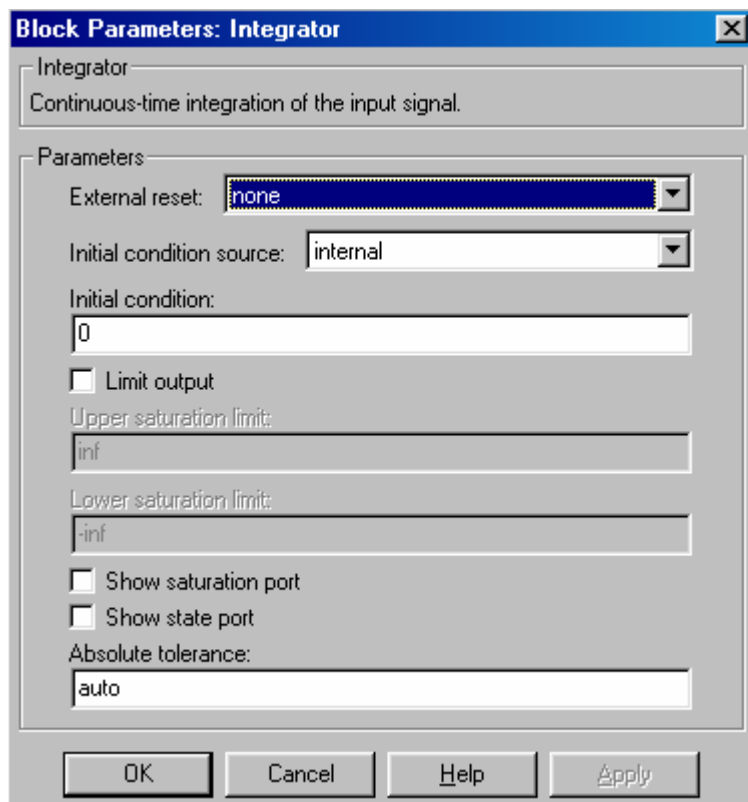
Зураг 2.2

Ажиллах зарчмыг дээрхи графикаас тодорхойлбол:

Интегратор нь оролтын функцийг интегралчилна. Үр дүн нь өсөлтийн коэффициент нь 1 -тэй тэнцүү шугаман өсөлттэй функц гарав.

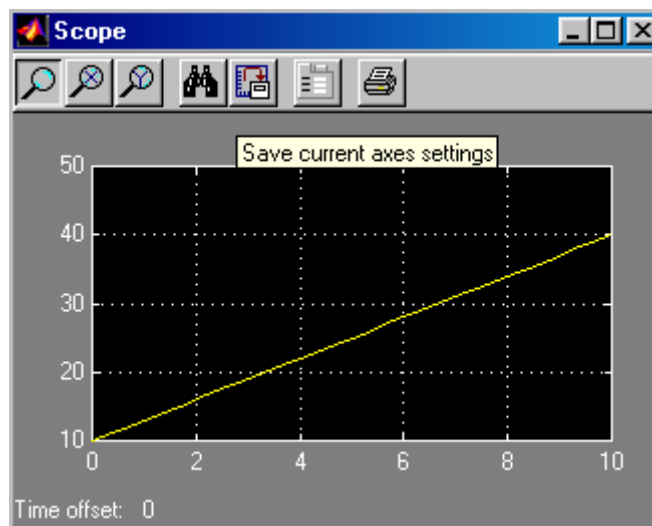
Параметрын цонхонд (Зураг 2.3) тогтмолыг өөрчилбөл:

- Интегралчлалын тогтмолыг 0 гэж параметрын цонхонд өөрчлөн,



Зураг 2.3

харин тогтмолыг (Constant) 3 , анхны нөхцөлийг 10 гэж өөрчлөхөд доорхи үр дүн гарч байна (Зураг 2.4).



Зураг 2.4

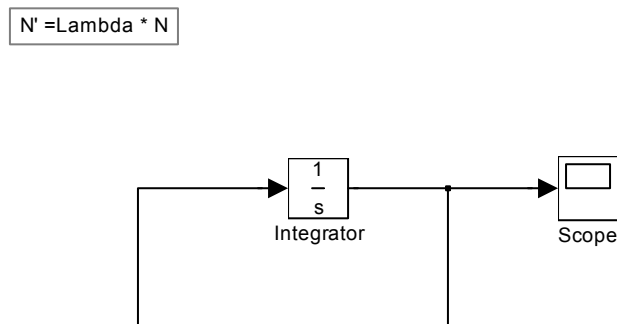
Жишээ 2.1

Одоо логистик дифференциал тэгшитгэлийн шийдийг тодорхойлох тухай үзье. Үүнд:

- Төлөвийн хэмжигдэхүүн нь бодгалийн тоо $N(t)$
- Өсөлтийн зэрэг нь бодгалийн тоотой пропорционал тул

$$N' = \lambda N$$

тэгшитгэл нь хязгааргүй нөөцтэй байх үеийн өсөлтийг тодорхойлдог. Симуляцийн загварт $\lambda = 1$ гэж үзье.



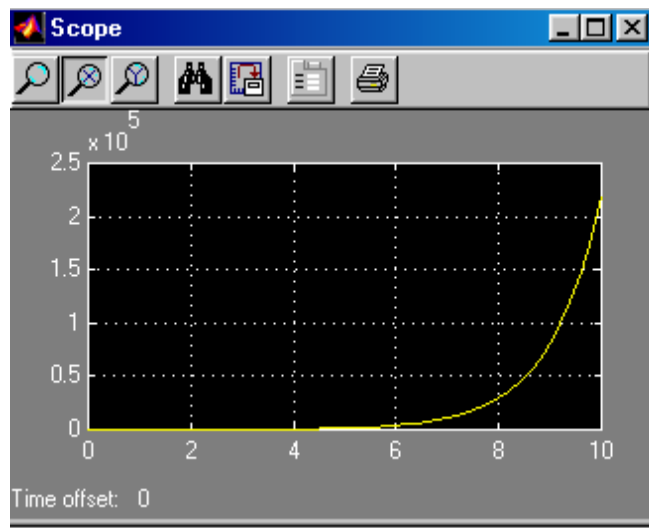
Зураг 2.5

Ажиллах зарчим нь:

- Интеграторын оролтонд N' -ийг өгнө
- Гаралтанд N гарна (N' -ийн интеграл мэт)
- Битүү гогцоо нь $N = N'$ гэж ажиллахыг албадаж байна.

Загварчлалыг ажиллуулахад анхны утга 0 учир үр дүн нь 0 гарч байна.

Анхны утгыг $N(0)=10$ гэж интеграторт тааруулбал экспонентоор өөрчлөгдөх өсөлт гарна (Зураг 2.6).



Зураг 2.6

Жишээ 2.2

$$x'' + \omega^2 x = 0$$

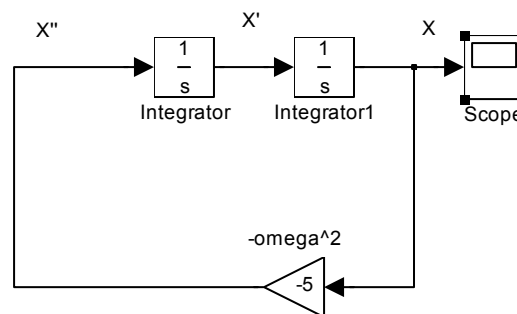
тэгшитгэлээр тодорхойлогдсон гармоник генераторыг загварчлах

Үүний тулд:

- x сигналыг тодорхойлно.
- x -ын өмнөх интеграторын оролтонд x' -ийг өгөх ба мөн x' -ийн өмнөх интеграторын оролтонд x'' -ийг өгнө. Гэдрэг холбоо үүсгэж дифференциал тэгшитгэлийг үүсгэнэ.

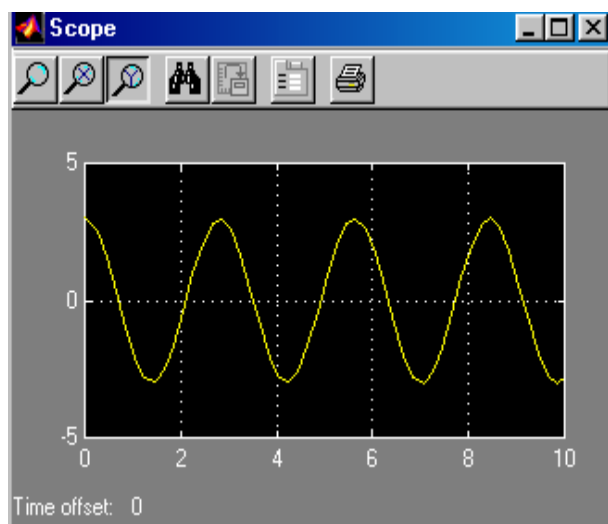
Bewegungsgleichung

$$x'' + \omega^2 x = 0$$



Зураг 2.7

$x_0 = 0$ байх үеийн үр дүнг үзүүлбэл (Зураг 2.8):



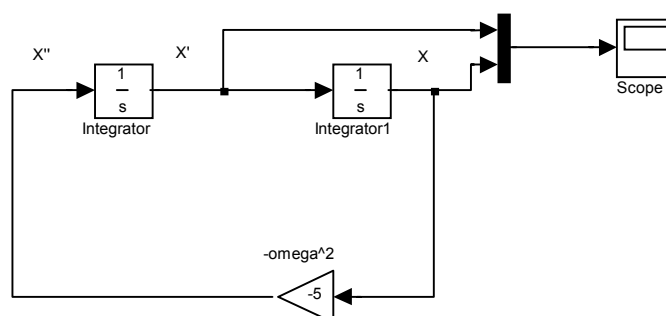
Зураг 2.8

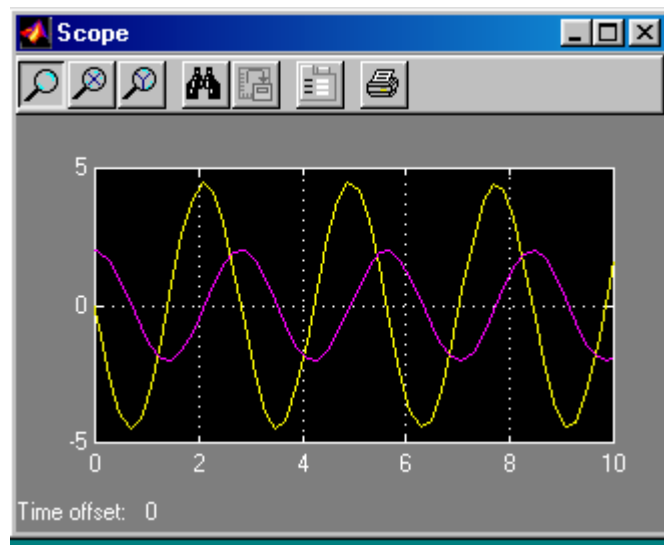
Жишээ 3**Олон хэмжээст системийн тухай ойлголт**

Сигналын дамжуулагч нь хэд хэдэн сигналыг (Вектор) бүхэлд нь агуулсан байж болдог. Үүнийг векторын дамжуулагч гэдэг.

Жишээ болгож, шилжилт ба хурдыг заах гармоник сигналыг загварчлах тухай үзье. Үүнийг хэд хэдэн дамжуулагчийг Signals&Systems - ийн Mux блокод өгч вектор сигналтай ажиллах боломж гардаг. Osci блокод сигналууд нь янз бүрийн өнгөөр гардаг (Зураг 3.2).

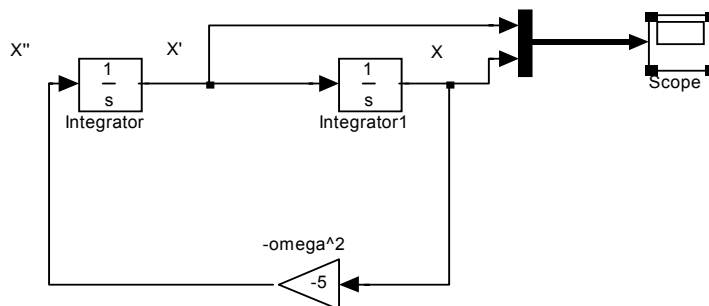
ШИЙД: (Зураг 3.1)





Зураг 3.2

Format/Wide nonscalar lines – аар векторын (олон) дамжуулагчийг зузаан шугамаар тодотгож болдог (Зураг 3.3).



Зураг 3.3

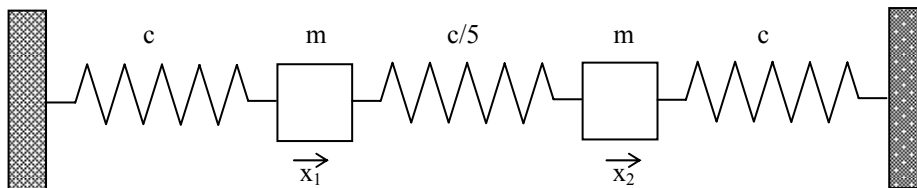
Одоо бид хоёр хэмжээст хэлбэлзлийн гинж болох пүрш- массын системийн тухай үзье: (Зураг 3.4)

Уг систем нь доорхи дифференциал тэгшитгэлээр илэрхийлэгддэг, Үүнд:

$$x_1 + \frac{6}{5} \omega^2 x_1 - \frac{1}{5} \omega^2 x_2 = 0$$

$$x_2'' - \frac{1}{5} \omega^2 x_1 + \frac{6}{5} \omega^2 x_2 = 0$$

$\omega = \sqrt{c/m}$



Зураг 3.4

Загварчлалын анхны нөхцөл: $x_1(0)=1, x_2(0)=0$

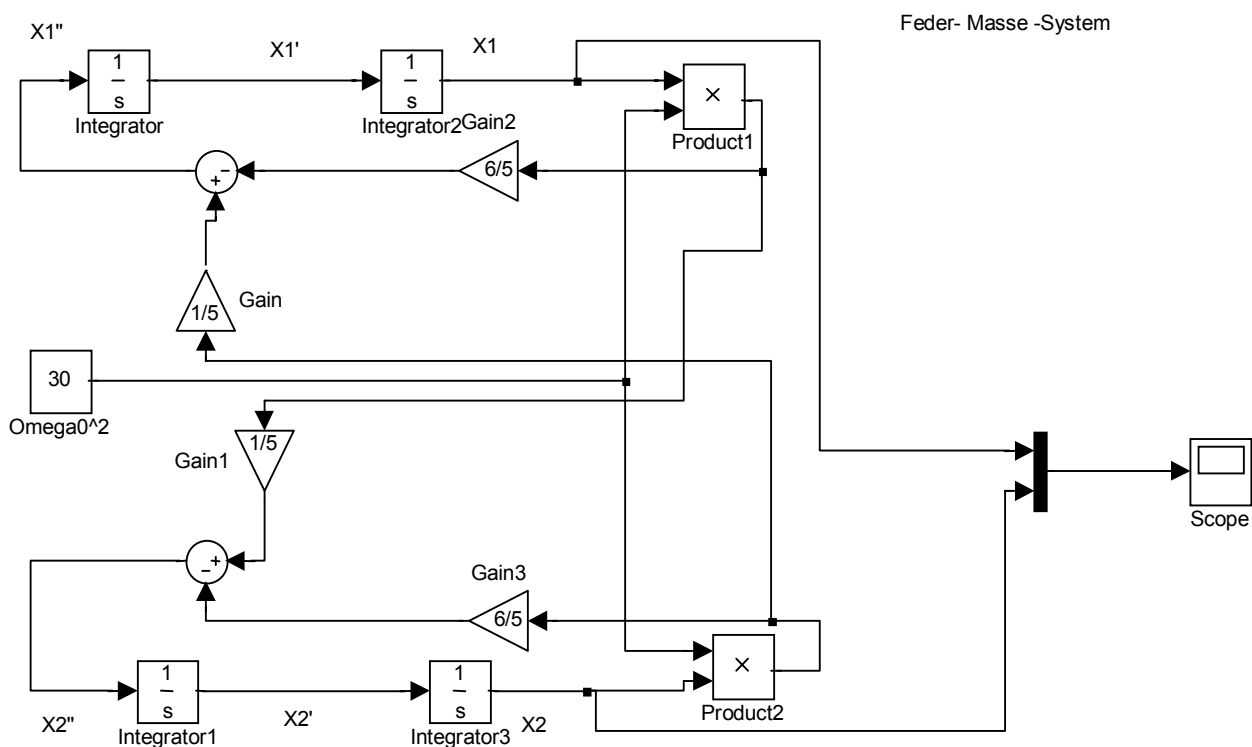
$v_1(0)=0, v_2(0)=0$

Загвар:

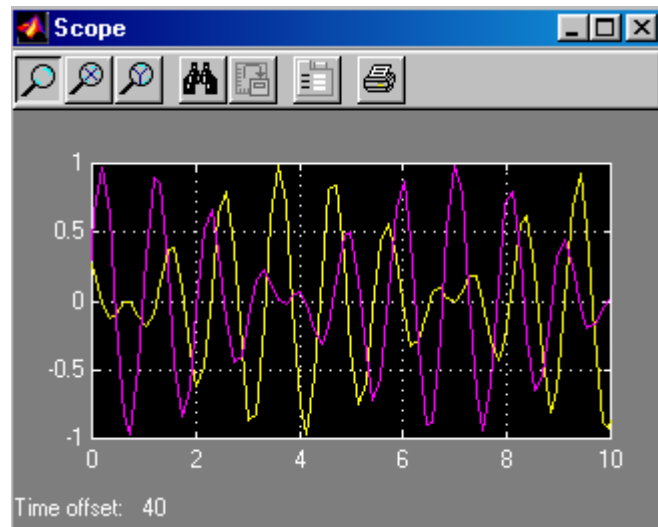
Зарим блокийг Format/Flip буюу Format/Rotate Block ашиглаж эргүүлэх шаардлага гардаг.

Sum блокийн параметрыг List of signs -ээр (+-) - д тааруулдаг.

Зураг 3.5



Загварчлалын үр дүнг гаргавал:



Зураг 3.6

Хэлбэлзэлийн гинжийг матрицийн хэлбэрт бичвэл координатууд нь вектор хэлбэрт бичигдэнэ:

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

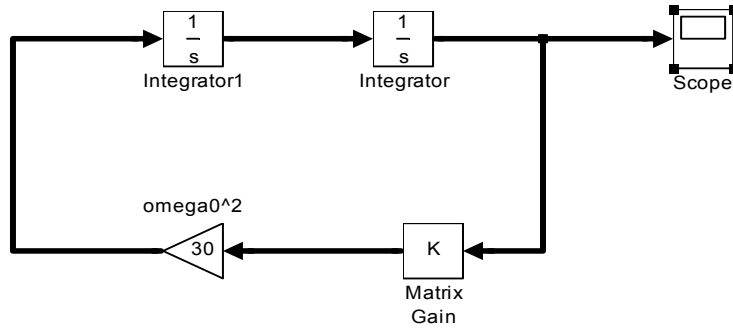
x_1 , x_2 -ын коэффициентийг 2×2 хэмжээст матрицаар бичвэл:

$$S = \begin{pmatrix} 6/5 & -1/5 \\ -1/5 & 6/5 \end{pmatrix} \omega^2$$

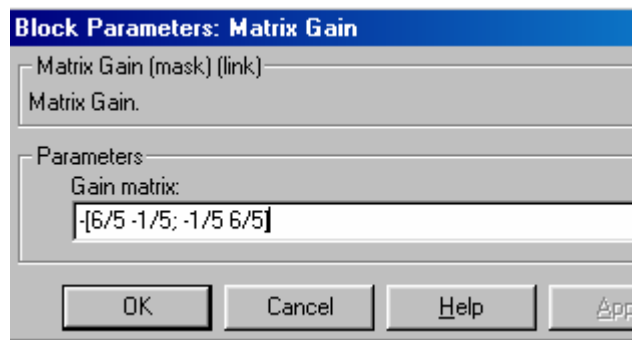
болж, дифференциал тэгшитгэлийг

$$x'' + S x = 0$$

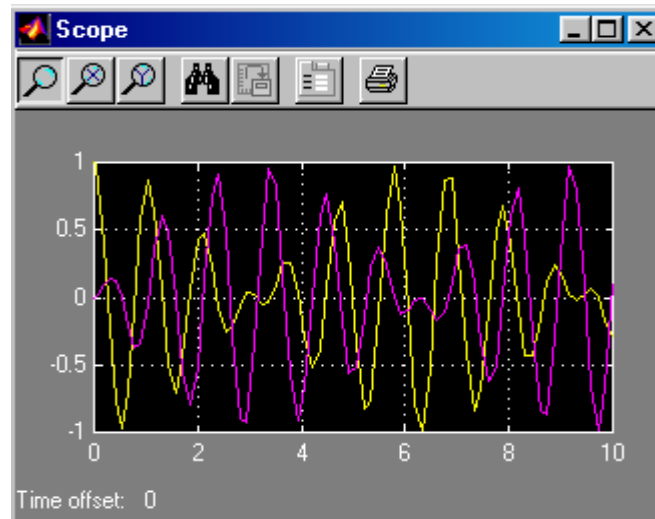
гэж бичиж болно. Ингэснээр Simulink загварыг хялбархан гаргаж болно. Ингэж үүссэн загвар нь матрицтай ажиллах тул, бүдүүн шугамаар холбогддог. Вектортой ажиллаж байгаа тул интеграторын анхны нөхцлийг $[1 \ 0]$ ($x_0 = \text{integrator1}$) ба $[0 \ 0]$ ($v_0 = \text{integrator}$) гэж бичнэ. Matrix Gain нь тогтмол матрицтай үржигдэж байна. Матрицийг оруулахдаа параметрийн цонхыг ашиглана (Зураг 3.8). Загварчлалын үр дүн дээрхитэй ижил байна (Зураг 3.9).



Зураг 3.7



Зураг 3.8



Зураг 3.9

Жишээ 4**Дискрет систем**

Дунд сургуулийн сурагчдын тоо нь 8, 9, 10-р ангид суралцаж буй сурагчдын тоо x_8, x_9, x_{10} -аас бүрдэнэ гэж үзье. Тэгвэл жил бүрийн сурагчдын тоо нь:

- 8-р ангид дэвшин суралцагч сурагчдын тоо X_{in}
- улиран суралцагчдын тоо (W_8 хувьтай), суралцахаа болсон сурагчдын тоо, (A_8 хувьтай)
- сүүлийн жил анги дэвшсэн сурагчдын тоо зэргээс бүрдэнэ.

Энэ нь доорхи тэгшитгэлээр илэрхийлэгдэнэ.

Үүнд:

$$\begin{cases} x_8(k) = X_{in} + W_8 x_8(k-1) \\ x_9(k) = (1 - W_8 - A_8) x_8(k-1) + W_9 x_9(k-1) \\ x_{10}(k) = (1 - W_9 - A_9) x_9(k-1) + W_{10} x_{10}(k-1) \end{cases}$$

Төгссөн сурагчдын тоог

$$x_{abi}(k) = (1 - W_{10} - A_{10}) x_{10}(k-1)$$

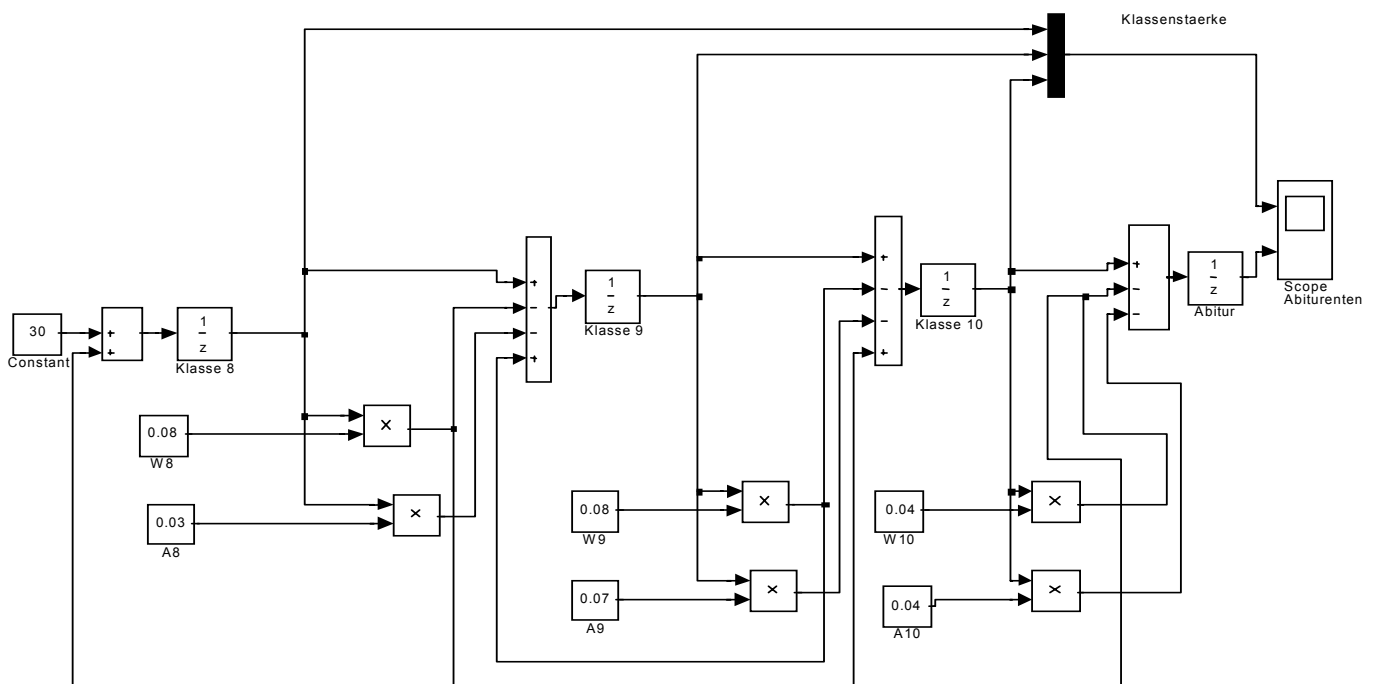
гэж тодорхойлно.

Үндсэн блок: Discrete/Unit Delay

Загварчлалын дискрет алхам бүрт утга авна.

Параметр Sample Time= 1 (нэг жил)

Загвар: Зураг 4.1



Загварчлалын параметр:

- Solver discrete (no continuos states)
- Алхамын хэмжээ (fixed-step) тогтмол
- Үр дүн (Зураг 4.3)
- **Бие даасан блокыг үүсгэх:**
 - анги бүрт блокын бүтэц давтагдаж байна
 - нэг ангийг илэрхийлэх блокыг дангаар нь гаргавал сайн
 - бие даасан блок нь оролт ба гаралттай (Inport, outport блокыг хэрэглэх)

Оролт:

- шинэ сурагчдын тоо

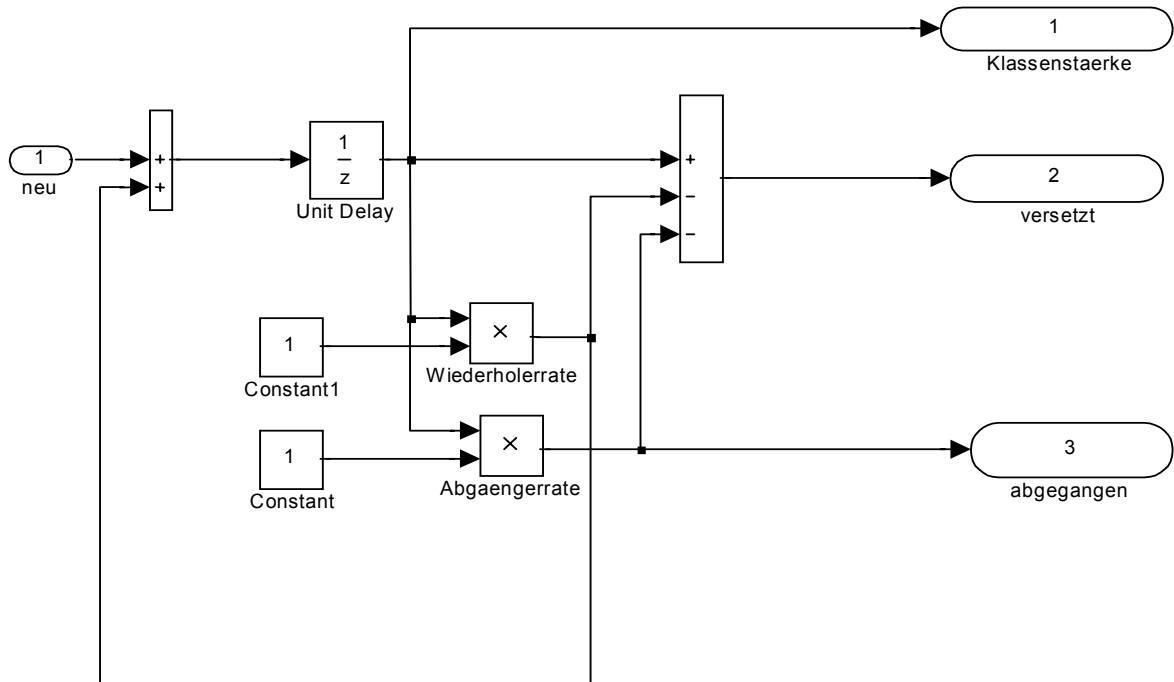
Гаралт:

- ангид байгаа сурагчдын тоо
- анги дэвшсэн сурагчдын тоо
- сургууль завсардсан сурагчдын (гарсан)тоо

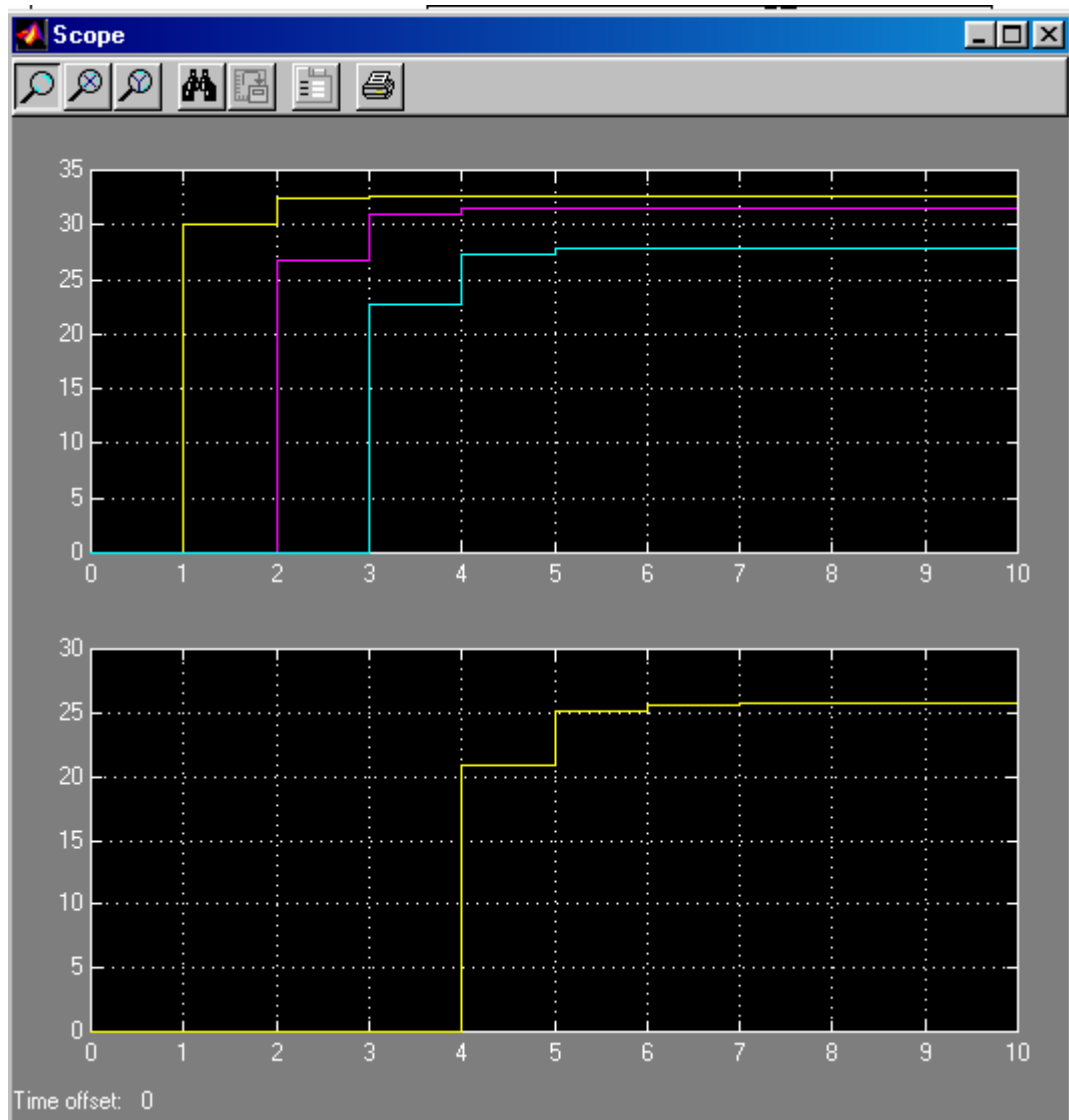
- Анги хэмээх дэд загварыг шинэ цонхонд үүсгэх
- Бие даасан блокод шилжүүлж хувиргах

- Бүх блокийг сонгох хүрээнд хийж тэмдэглэх
- Edit /Create Subsystem-ээр шинэ дэд блок үүсгэх
- Copy ба Paste-ээр шинэ модельд хуулах

Ингэж анги (Klasse) хэмээх дэд блокийг үүсгэлээ (Зураг 4.2).

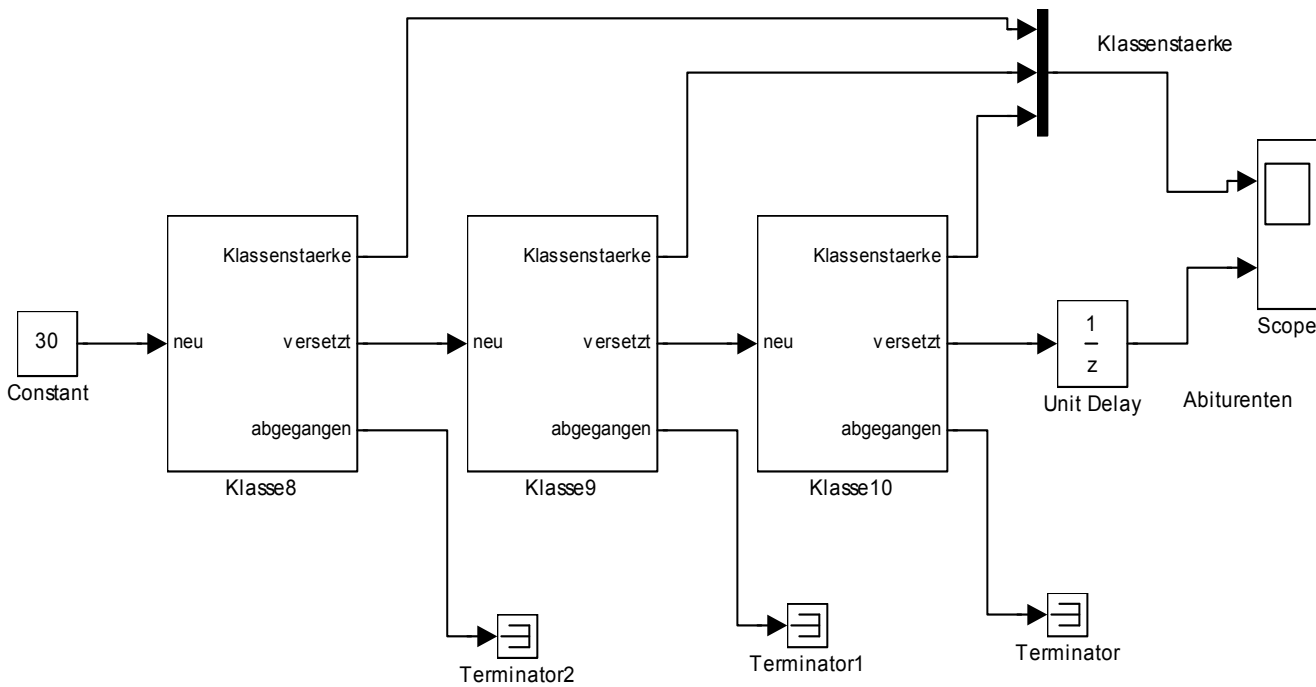


Зураг 4.2



Зураг 4.3

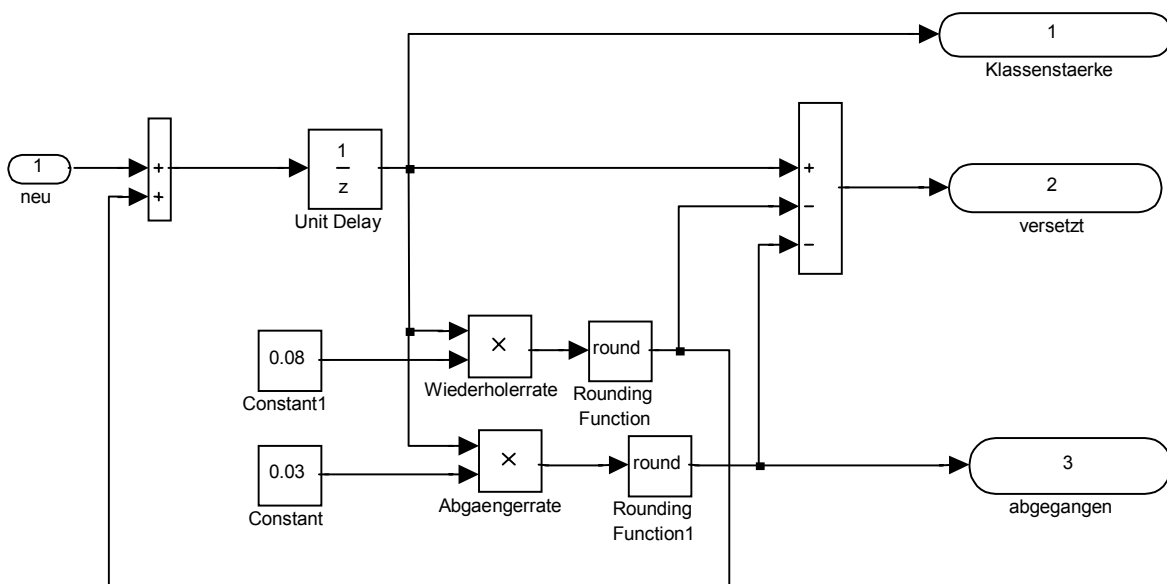
- Ерөнхий загварт сургууль завсардсан сурагчдын тоог хэрэглэхгүй. Учир нь:
 - Matlab цонхонд алдааны санамж үүснэ. Харин үүнийг Terminator-блокттой холбоно.



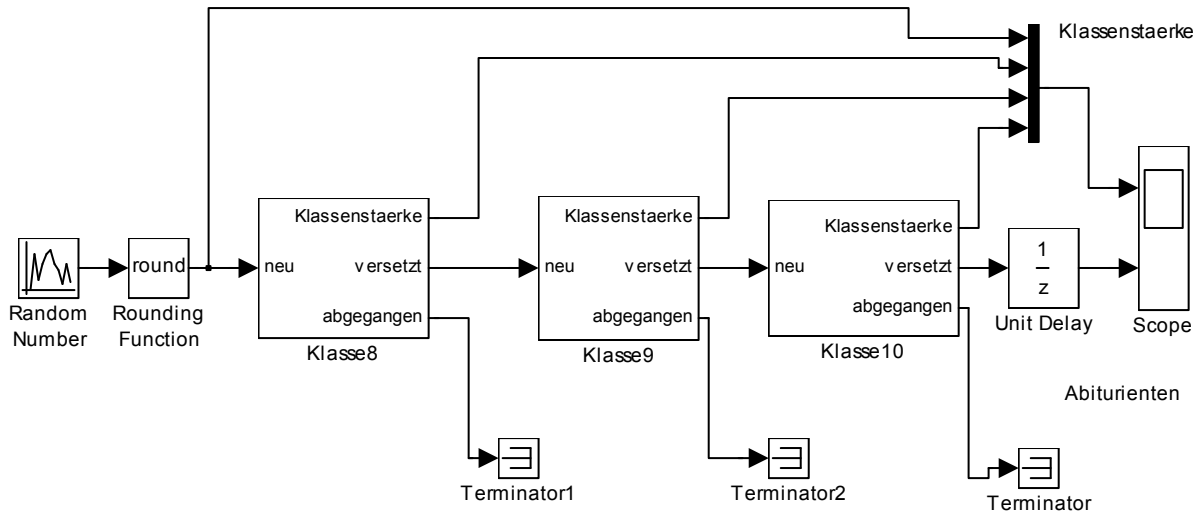
Зураг 4.4

Одоо тухайн сургуулийн загварыг өргөжүүлье. Үүний тулд: Санамсаргүй тооны генераторыг (Random Number блокийг ашиглах) сурагчдын тоог үүсгэгч болгож хэрэглэе. Санамсаргүй тооны генератороос ирэх тоон утга нь бутархайн хэсэгтэй байх тул (Rounding Function) блокоор бүхэл тоо болгож хувиргана (Зураг 4.6). Учир нь сурагчдын тоо бүхэл байх ёстой.

Өргөжүүлж ,сайжруулсан ангийн загвар: Зураг 4.5

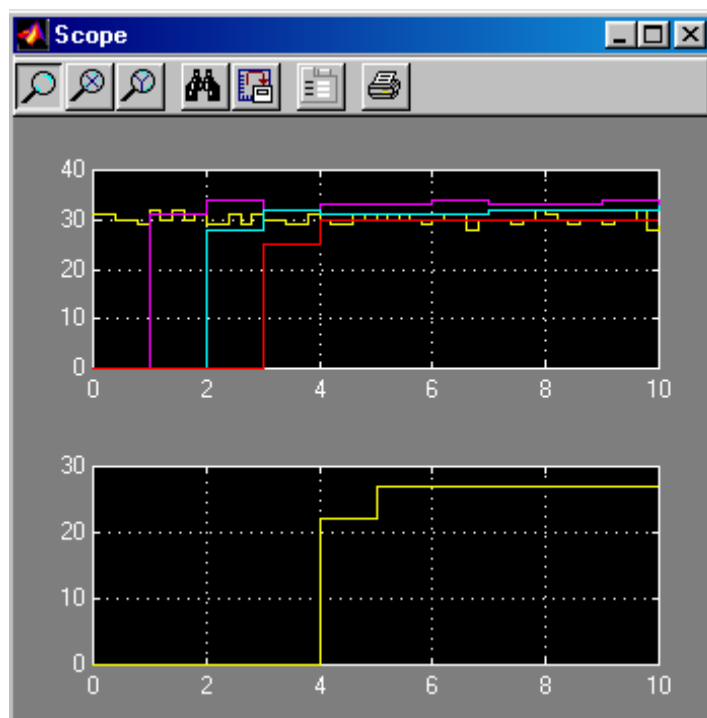


Зураг 4.5



Зураг 4.6 :Өргөжүүлж , сайжруулсан сургуулийн загвар

Өргөжүүлж, сайжруулсан сургуулийн загварын симуляцийн үр дүнг Зураг 4.7-д үзүүлээ.



Зураг 4.7

Цаашид сайжруулж болох санамж:

- W_i ба A_i “тогтмолуудыг” статистик өөрчлөлттэй болгох
- Оролтын загварыг сайжруулах

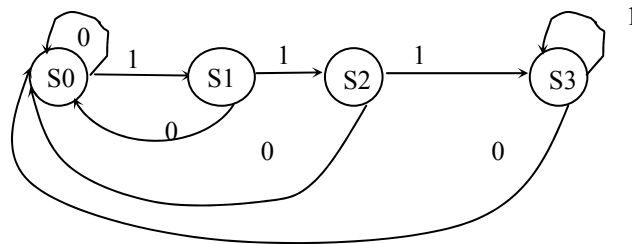
Жишээ 5

Дискрет систем

Оролтын сигналыг таних

Даалгаварын нөхцөл:

- Оролтын утга нь \emptyset ба 1 гэсэн дурын дараалал үүсгэнэ
- Программ нь дэс дараалсан гурван 1-ийг таних ёстой. Үүний тулд дөрвөн төлөв байдалтай төгсгөлтэй автоматаар илэрхийлбэл:



Зураг 5.1

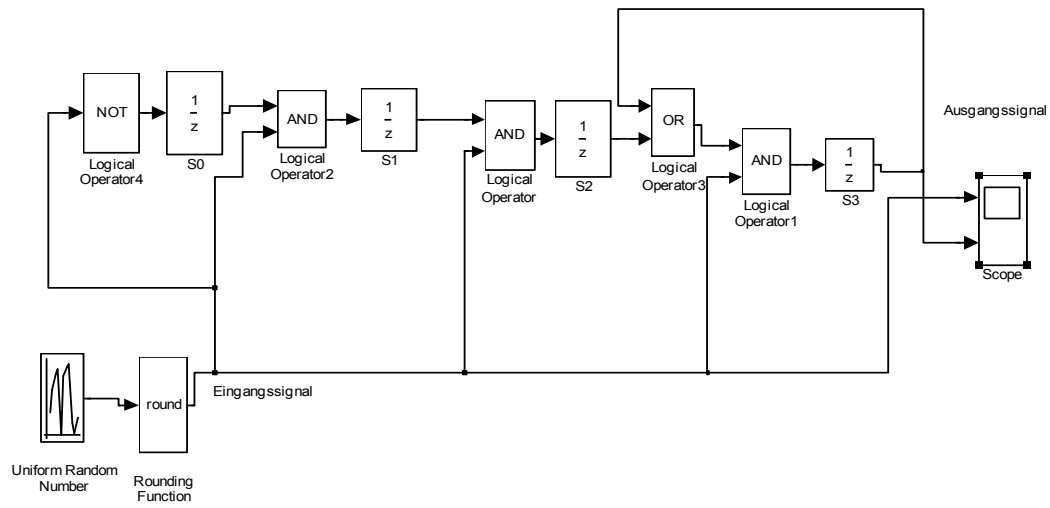
Төлөв байдал бүрт нэг Unit Delay-г харгалзуулах (Төлөв байдлыг тэмдэглэх)

- S0 идэвхитэй (өөрөөр хэлбэл төлөв 1-тэй тэнцүү) \Leftrightarrow 1 орж ирээгүй
- S2 идэвхитэй \Leftrightarrow шууд (11) орж ирсэн
- S3 идэвхитэй \Leftrightarrow шууд (111) орж ирсэн

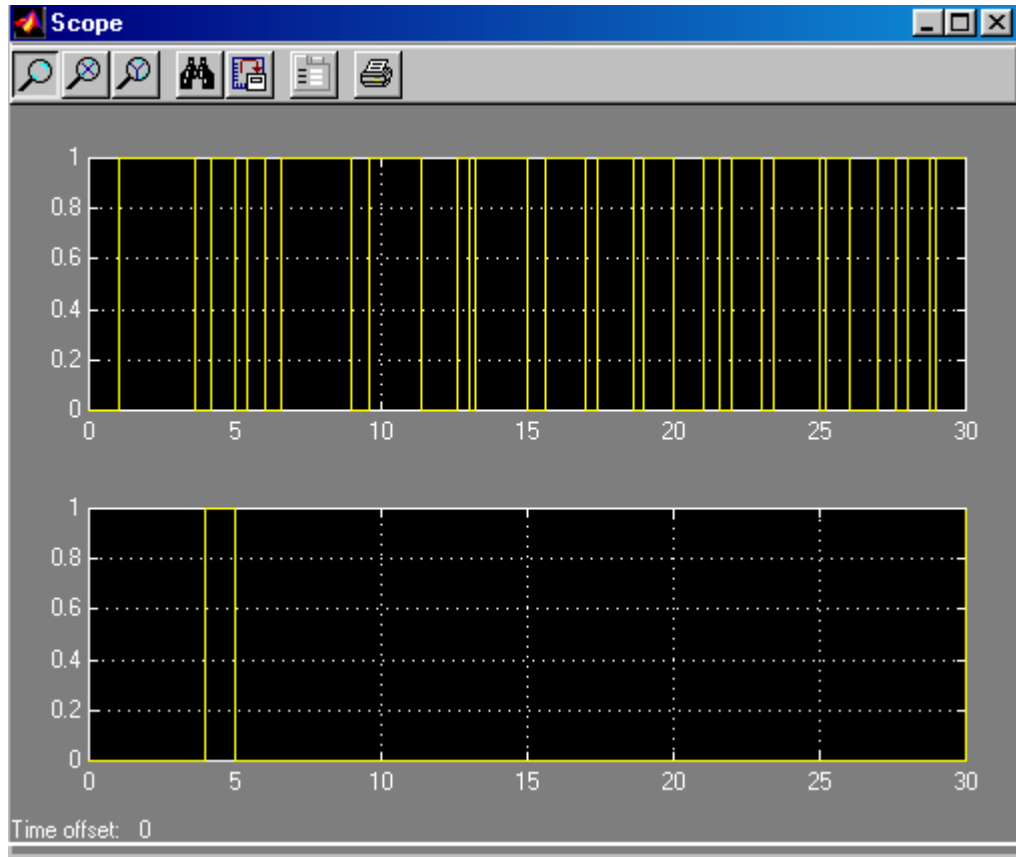
Шилжилтийг (нэг төлөвөөс нөгөө төлөвт) AND/OR схемээр орлуулах, тухайлбал

- S1 идэвхитэй байвал \Leftrightarrow S0 идэвхитэй ба 1 орж ирнэ
 - S3 идэвхитэй байвал \Leftrightarrow (S2 идэвхитэй буюу S3 идэвхитэй) ба 1 орж ирнэ.
- Анхны төлөв байдал нь 1 гэсэн анхны утгатай ба бусад нь \emptyset утгатай байна.

Загвар: Зураг 5.2



Загварчлалын үр дүн:



Зураг 5.3

Жишээ 6**Жишээ 6.1****Шугаман бүс систем**

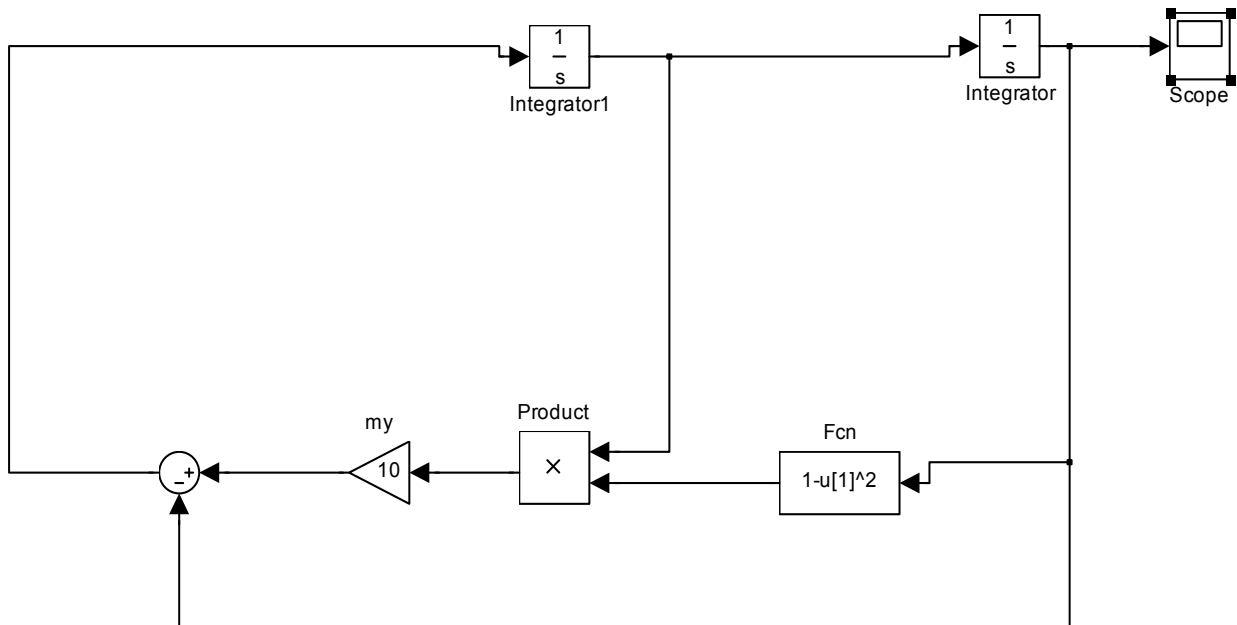
Van-der-Pool – ын генератор:

Электрон генераторын загварыг авч үзье.

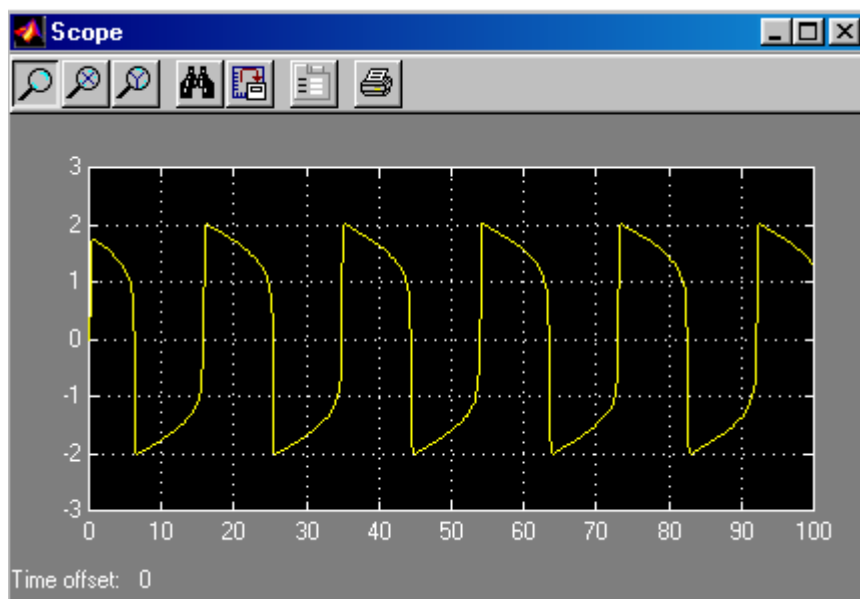
Үндсэн тэгштгэл нь

$$x'' - \mu(1-x^2)x' + x = 0$$

Загвар : Зураг 6.1

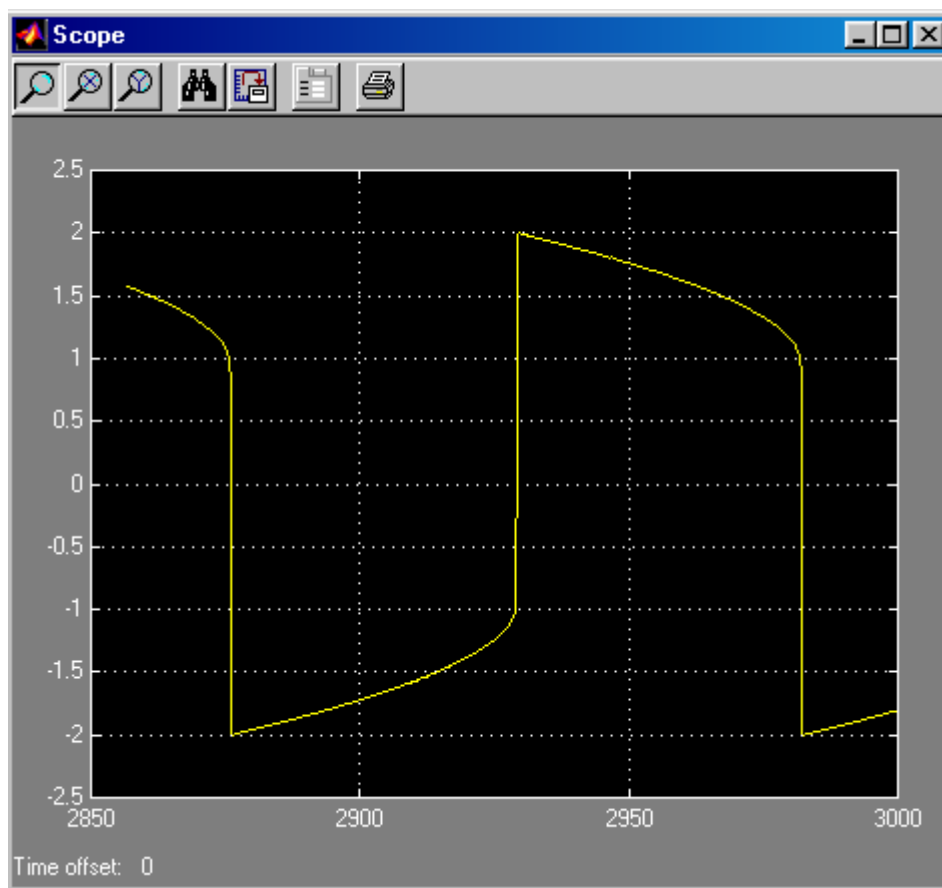


$\mu=10$ байхад гарах үр дүн:



Зураг 6.2

$\mu=650$ байхад гарах үр дүн:



Зураг 6.3

- Тооцооны хугацаа μ -ээс хэрхэн хамаарахыг ажиглах
- Хэлбэлзлийн үеийг тодорхойлох
- Фронтын өсөлтийг гаргах

Жишээ 6.2

Гулсах үрэлттэй пүршийг авч үзье

- Үрэлтийн хүчийг үйлчлэх хүч F_N -ээр тодорхойлбол:

$$FR = \begin{cases} -\eta F_N & |x'| > \emptyset \\ \eta F_N & |x'| < \emptyset \end{cases}$$

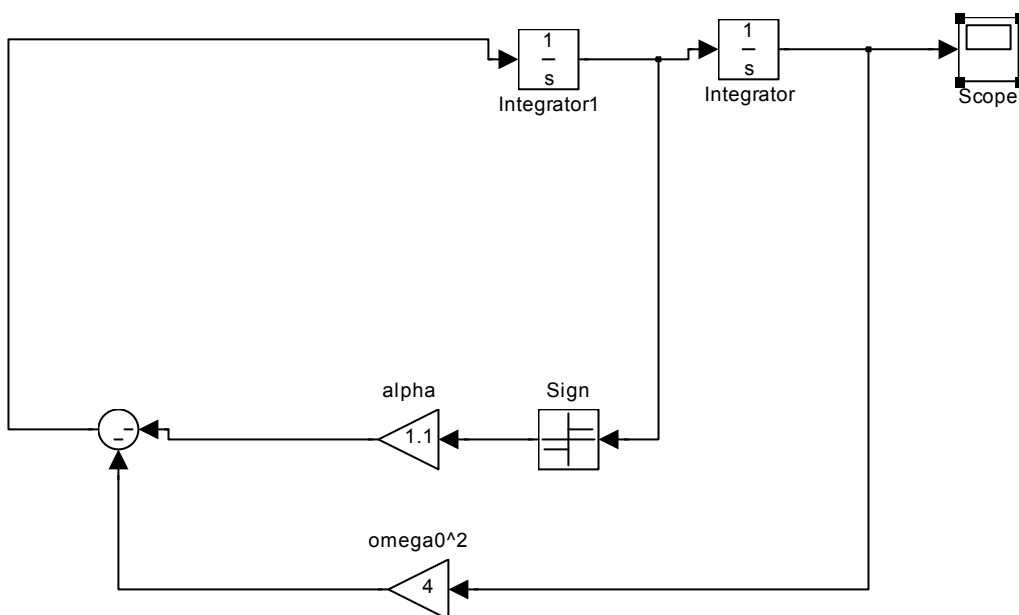
Цэгцтэй байдлаар бичвэл

$$F_R = -\mu F_N \text{sing}(V)$$

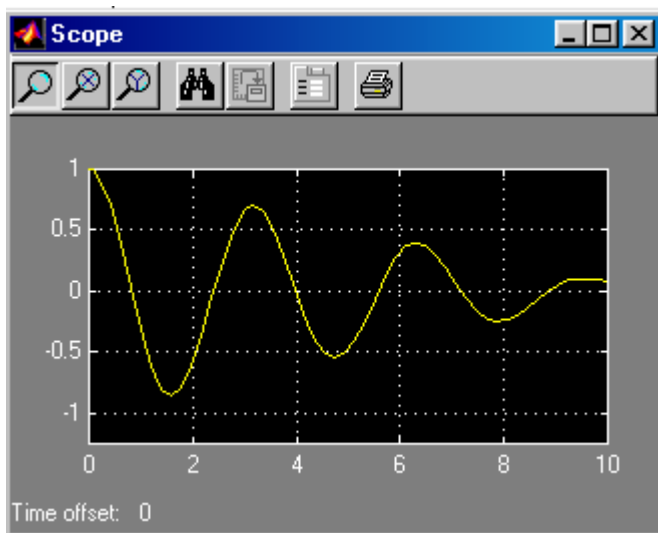
- Хөдөлгөөний тэгштгэлийг бичвэл

$$m x'' + \mu F_N \text{sign}(x') + cx = \emptyset$$

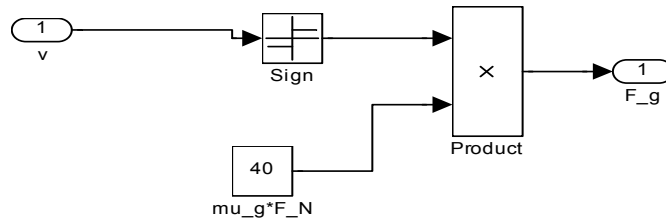
Загвар: Зураг 6.4



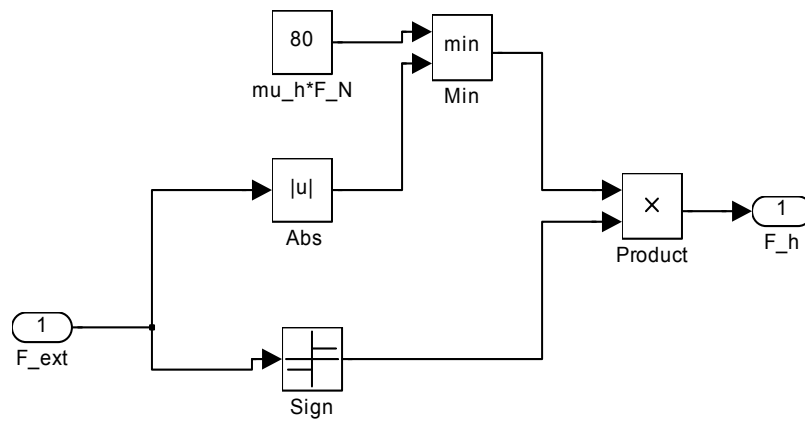
Үр дүн:



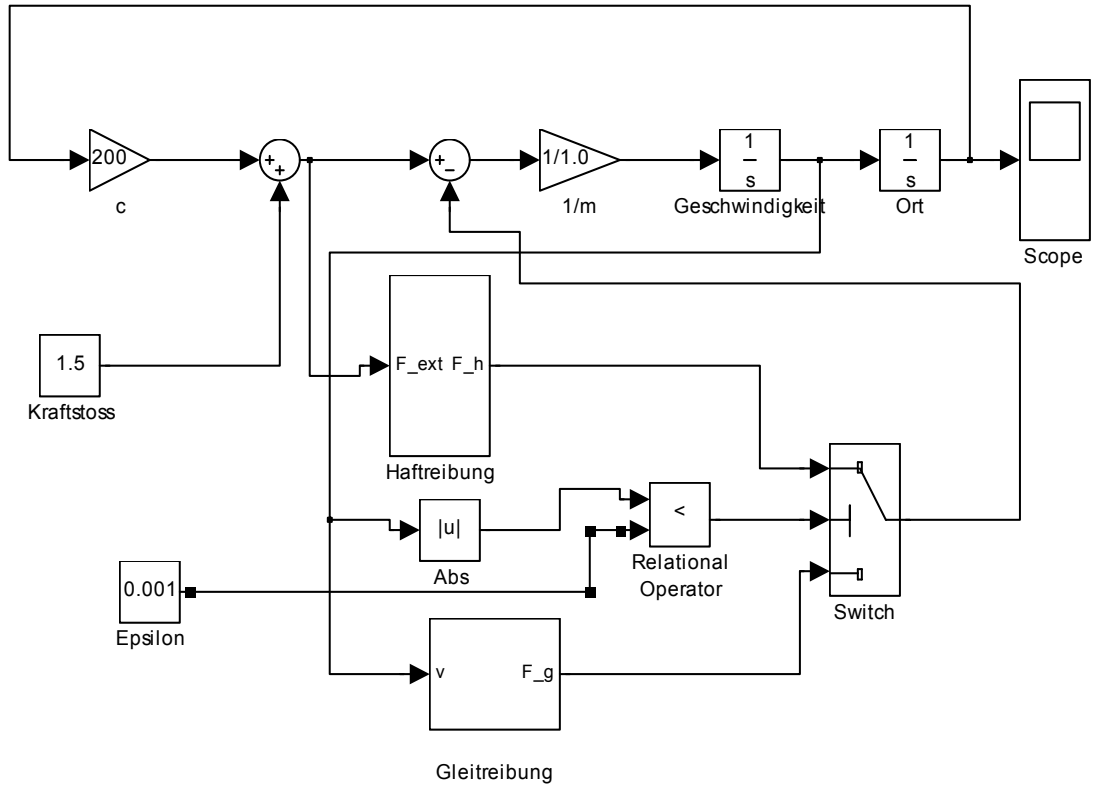
Зураг 6.5

Гулсах үрэлтийн дэд загвар: Зураг 6.6**Зуурах үрэлтийн дэд загвар:**

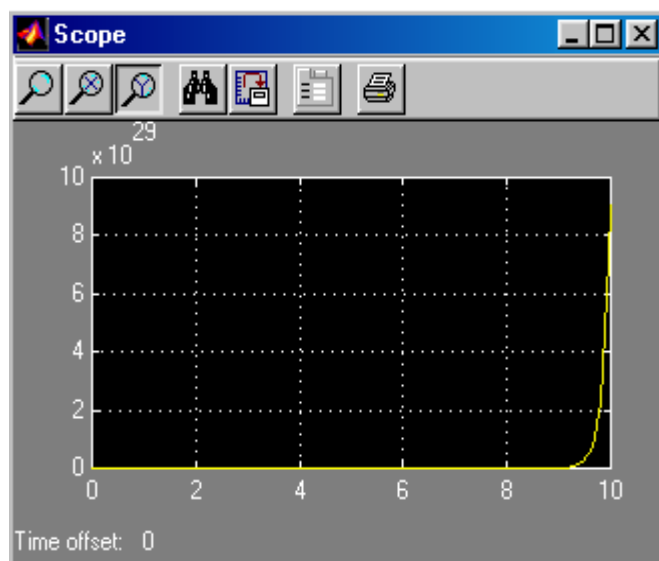
Зураг 6.7



Үндсэн загвар: Зураг 6.8



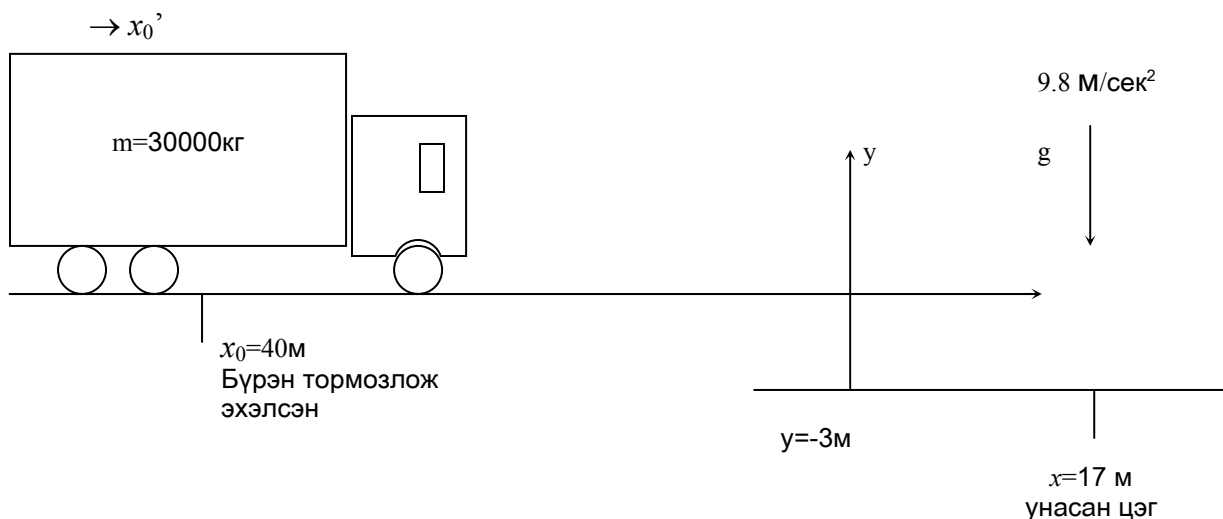
Загварчлалын үр дүн: Зураг 6.9



Жишээ 7

1. Ачааны автомашин

Ачааны автомашин (жин 30000кг) хэтэрхий орой тормозолсоны улмаас 3 м-ийн гүн ангалруу харайн оржээ. Осол гарсан газарт ангалын ирмэг хүртэл 40 м тормозны мөр ба ачааны машин үсрэн буусан цэг нь 17м –т байхыг шинжээч илрүүлжээ. Ачааны машины жолооч 80 км/цаг –ийн хурдтай явсан гэх ба ангалыг мэдмэгцээ бүрэн тормозолсон гэж мэдүүлээд, тэрээр ямар ч буруугүй, учир нь ийм осолтой газар ямарч анхааруулах тэмдэг тавиагүй байсан гэжээ (Зураг 7.1).



Зураг 7.1: Ачааны машины осол болсон байдлын тойм зураг

Иймд Та жолоочийн мэдүүлэг үнэн болохыг батлах ёстой. Хэрэв Та жолоочийн мэдүүлэгийг баталсан бол ачааны автомашин бүрэн тормозлож эхэлсэн үеийн хурдыг (ойролцоогоор) тооцоолно уу? Даалгаварын нөхцлийг баримтална уу.

Заавар: Та тооцоог ямагт СИ системд (м, кг сек, К, А) хийх ёстой. Хэрэв физик хэмжигдэхүүний хэмжих нэгжийг өөрөөр авбал эдгээрийг мөн ижил нэгжид шилжүүлэх хэрэгтэй (жишээ нь: км/цаг –ийг м/сек-д)

1.1 Дифференциал тэгшитгэл зохиох

Тойм зураг дээр ачааны автомашиныг хүч ба чиглэлийн вектортой цэгийн масс мэт дүрслэнэ. Ачааны автомашин цэгийн массын хувьд x – ба y -чиглэл дэх хөдөлгөөний тэгшитгэлийг гаргах хэрэгтэй. x -чиглэл дэх хүч нь тормозны хүч (дугуй нь газартай байх туршид), мөн агаарын эсэргүүцэл үйлчилнэ. y -чиглэлд газрын хөрсний контактын хүч, хүндийн хүч ба чөлөөтэй унах үед мөн агаарын эсэргүүцэл үйлчилнэ.

Тормозны хүчний тэгшитгэл

$$F_{\text{brems}} = m \cdot g \cdot \mu \left(1 - \frac{\dot{x}}{100}\right)$$

$\mu=0.6$ ба энэ нь резин/асфальтын хоорондын үрэлтийн коэффициент юм.

Агаарын эсэргүүцлийн тэгшитгэл нь

$$F_{luft} = C_W \cdot A \cdot \dot{x}^2 \quad \text{ба} \quad C_W = 1.1 \text{ сек}^2/\text{м}^4 \text{ юм.}$$

Х - чиглэлийн эсэргүүцлийн талбай $A_x = 7\text{м}^2$ ба у - чиглэлд $A_y = 30\text{м}^2$ $F_{luft} = C_W \cdot A \cdot \dot{x}^2$ тэгшитгэлийг FCN-д $1.1 \cdot 7 \cdot u^2$ (х - чиглэлд $A_x = 7\text{м}^2$) гэж бичнэ. Харин Y- чиглэлд $1.1 \cdot 30 \cdot u^2$ гэж бичнэ (Зураг 7.4).

Х- чиглэлийн тэгшитгэлийг

$$m x'' = -F_B(x') - F_L(x')$$

гэж бичвэл

$$x'' = 1/m (-F_B(x') - F_L(x'))$$

болно.

Х - чиглэлийн блокийг гаргахад тормозны хүчний блокийн гаралтыг нийлбэрлэх блокой шууд холбохгүй. Учир нь нисэн унаж байхад тормоз ямарч үйлчлэлгүй тул эхлээд нөхцөл шалгах ёстой. Энэ нь:

$x > 0$ бол $F_{brems} = 0!$ байх юм.

Тэгвэл $x > 0$ нөхцөлийг шалгаж, үүгээрээ тормозны хүчийг үйлчлэх эсвэл үйлчлэхгүй болгох блок (relational operator) шаардлагатай.

Х - чиглэлийн блокод $k = 1/30000$

Хурдны анхны утга $X_0^1 = 80 \cdot 1000/3600$

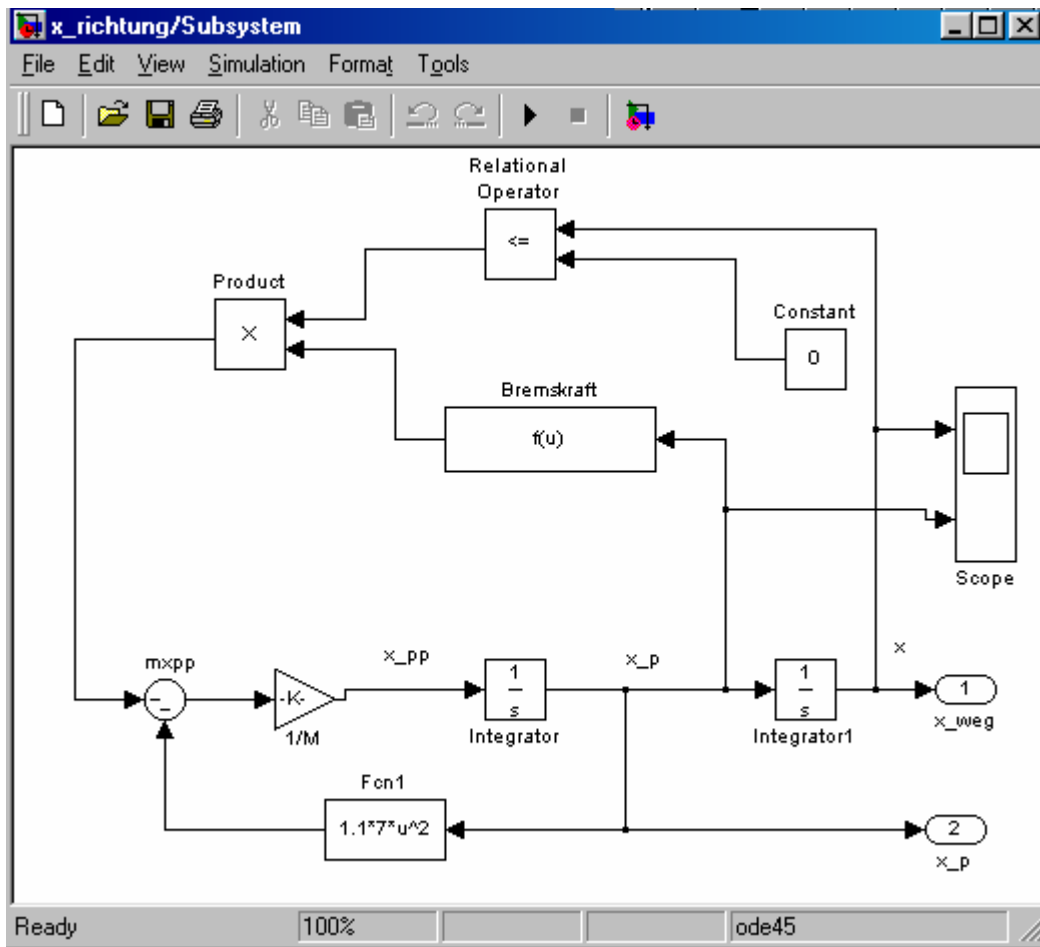
Тормозны хүчний тэгшитгэл $f_B(u) = 30000 \cdot 9.8 \cdot 0.6(1-u/100)$

Агаарын эсэргүүцлийн хүч $f_L(u) = 1.1 \cdot 7 \cdot u^2$

гэж тус тус тодорхойлогдоно (Зураг 7.2).

Y - чиглэлийн блок од хүндийн хүч $F_{Schwerkraft} = gm$ байдаг тул

$F_{Schwerkraft} = 9.8 \cdot 30000$ ХУ- чиглэлийн блокийн тогтмолын блокод (G) бичигдэнэ (Зураг 7.4).

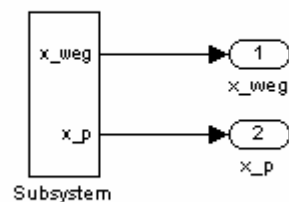


Зураг 7.2: X- чиглэлийн тэгшитгэлийн блок

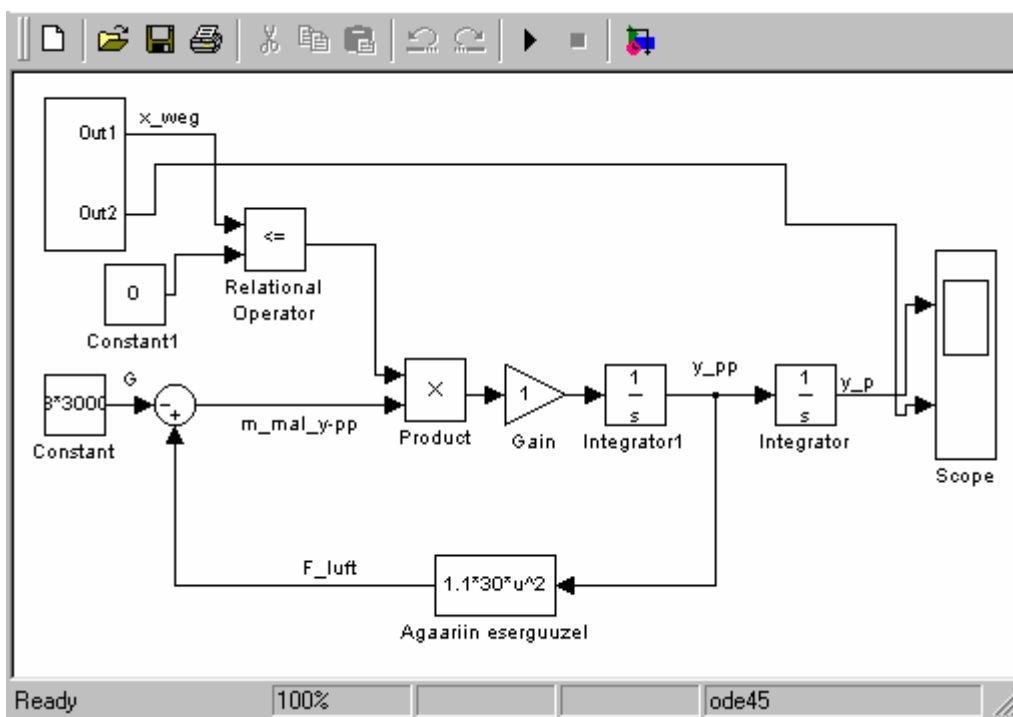
X - чиглэлийн блок-схемийг дэд систем болгох

Үүнд:

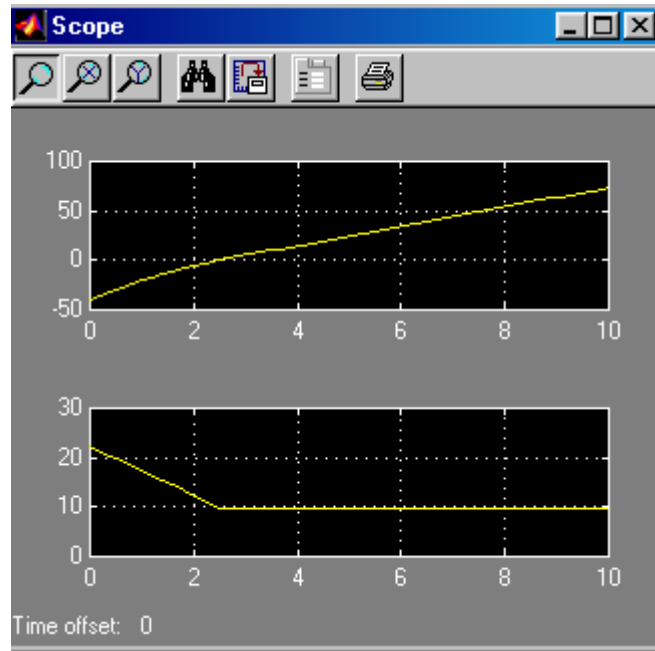
- Блокийг тэмдэглэж EDIT→CREATE SUBSYSTEM сонгох
- VIEW→BROWSER-оор иерархи үүсгэнэ (Зураг 7.3)



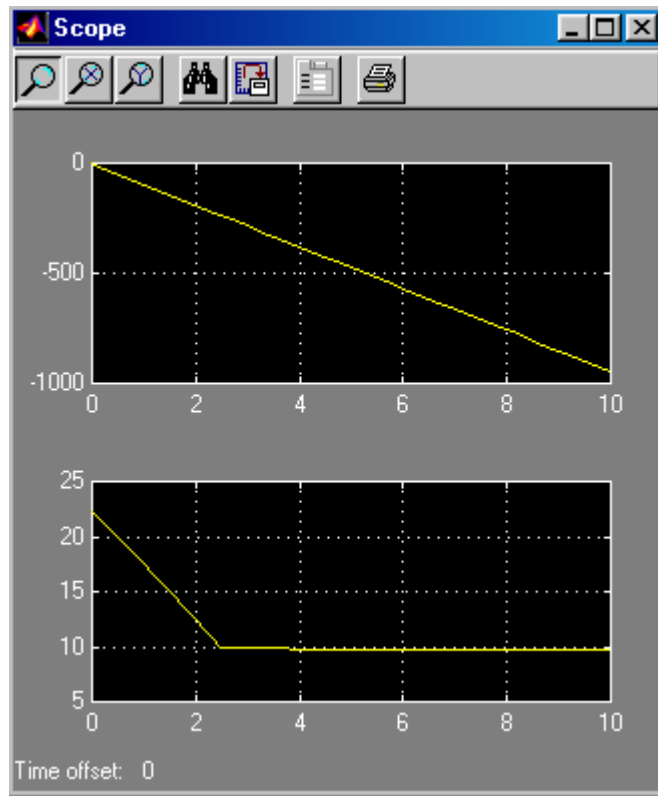
Зураг 7.3: X-чиглэлийн блокийн дэд систем



Зураг 7. 4: ХҮ -чиглэлийн блок



Зураг 7.5: X- чиглэлийн загварчлалын үр дүн



Зураг 7.6 : XY- чиглэлийн загварчлалын үр дүн

6 MATLAB/Simulink – ийн дадлагын ажлын даалгавар ба шийд

Даалгавар 1

- Хэрэв синус хэлбэлзлийг x – ба y -чиглэлд үйлчлэхээр залгавал гарах муруйг судлах. Хэлбэлзлэлийн янз бүрийн давтамж ба фазын ялгаатай байхад судалж үзэх.
- Заавар:
 - Sinks-ээс XY Graph блокийг хэрэглэх
 - Simulation –Parameters – д хамгийн бага алхмыг тааруулах

Даалгавар 2

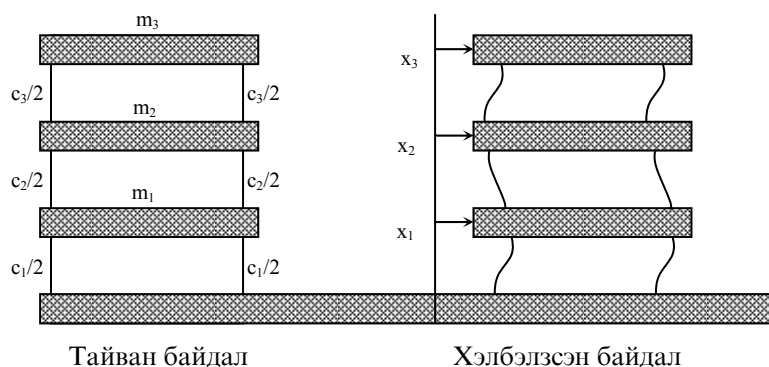
- Доорхи функцийн интегралыг тодорхойлох загвар (модель) үүсгэх. Үүнд:
 $f(t) = \sin(3t) \cdot \exp(-t)$
- Тооцоо хийж $t=6$ болтол графикаар дүрслэх

Даалгавар 3

- Доорхи тэгшитгэлээр өгөгдсөн унтрах хэлбэлзлийн хөдөлгөөнийг загварчлах
 $X'' + 3\delta X' + \omega_0^2 X = 0$
- Уг системийг ω_0 ба δ -ийн янз бүрийн утга болон янз бүрийн анхны байрлал ба хурданд судлах

Даалгавар 4

- Гурван давхар байшингийн чичиргээг судлах



- Хөдөлгөөний тэгшитгэлийг доорхи байдлаар илэрхийлдэг. Үүнд:

$$m_1 x_1'' + (c_1 + c_2)x_1 - c_2 x_2 = 0$$

$$m_2 x_2'' - c_2 x_1 + (c_2 + c_3)x_2 - c_3 x_3 = 0$$

$$m_3 x_3'' - c_3 x_2 + c_3 x_3 = 0$$

- Масс ба хатуулаг чанарын утгууд нь

- $m_1 = m_2 = 6$ т, $m_3 = 1$ т
- $c_1 = c_2 = 3c$, $c_3 = c = 10^6$ Н/м

1. Хэрэв дээд давхарын анхны хурд нь $V_3(0) = 10$ м/сек байвал, давхар бүрийн хэлбэлзлийг тодорхойлох

2. Газрын чичиргээнээс болж барилгын доод давхарт гаднын өдөөх хүч $F_3(t)$ үйлчилдэг. Ингэснээр хөдөлгөөний нэг дэх тэгшитгэл өөрчлөгдөж $m_1 x_1'' + (c_1 + c_2)x_1 - c_2 x_2 = F_1(t)$ болдог. Өдөөлт нь энгийн синус хэбэлзэл байна гэж үзэж, өдөөлтийн янз бүрийн үелзэлийн хувьд барилгын хэлбэлзлийн хэлбэрийг судлах.
3. Эцэст нь бодит өдөөлтийн хэлбэрийг, жишээ нь унтралтай, бага зэрэг шуугиантай газар хөдлөлийн долгионыг авч үзэх

Даалгавар 5

- Кола зарах автомат 10 ба 50 цент ба 1 долларын төмөр мөнгийг хүлээн авдаг. Хэрэв оруулсан мөнгөний нийлбэр 2.20 доллараас хэтрэхэд нэг лааз ундаа ба хариулт мөнгийг гаргаж өгнө.
- Мөн мөнгө автоматруу хийж буй худалдан авагчийн загварыг нэмж, энэ бүгдийг бүхэлд нь загварчлах

Даалгавар 6

Пүршний шугаман хүчний хуулийг

$F(x) = -x^3 + kx$ –ээр орлуулбал доорхи дифференциал тэгшитгэлээр илэрхийлэгдэх генератор гардаг. Үүнд:

$$x'' + \nu x' + x^3 + kx = A \cos(\omega t)$$

- Энэ системийн шинж чанарыг янз бүрийн параметрийн хувьд байрлал ба фазын орон зайд судлах

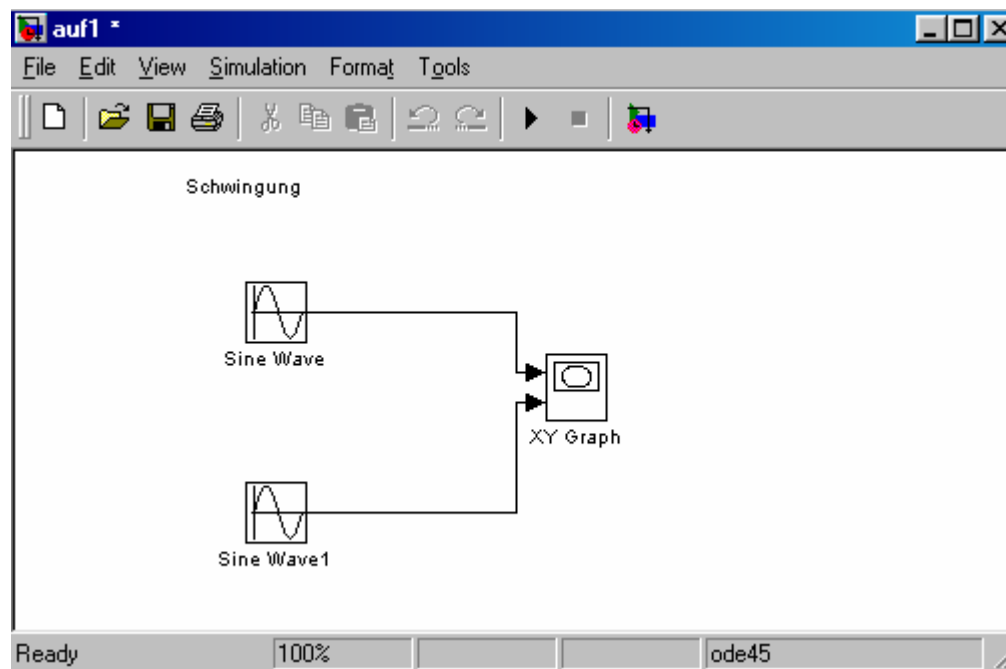
Даалгавар 7

- Лоренцийн (Lorenz) загвар нь цаг уурын атмосферийн үндсэн бүтэц үзэгдлийг энгийн дифференциал тэгшитгэлээр илэрхийлдэг. Үүнд:
 $x' = a(y - x)$
 $y' = -xz + bx - y$
 $z' = xy - cz$
- x хэмжигдэхүүн нь халсан агаарын хурдны чиглэлийг, y ба z нь температурын тархалтыг илэрхийлдэг.
- Янз бүрийн параметрийн утганд түүний шинж чанарыг судлах.
 $a=10, b=28, c=2.667$
- $x(t), y(t), z(t)$ -ийн гурван хэмжээст хөдөлгөөнийг дүрслэх

Даалгавар 1

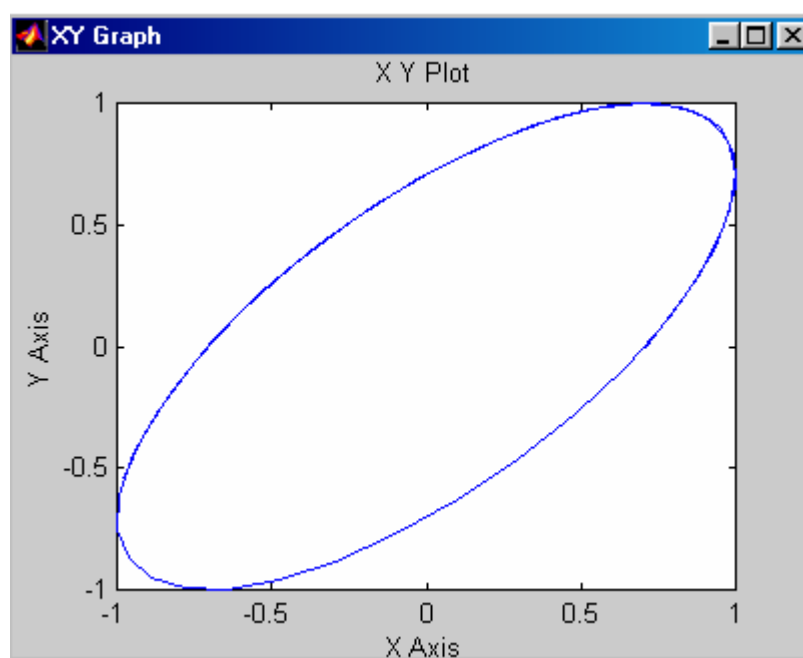
Шийд:

Загвар:

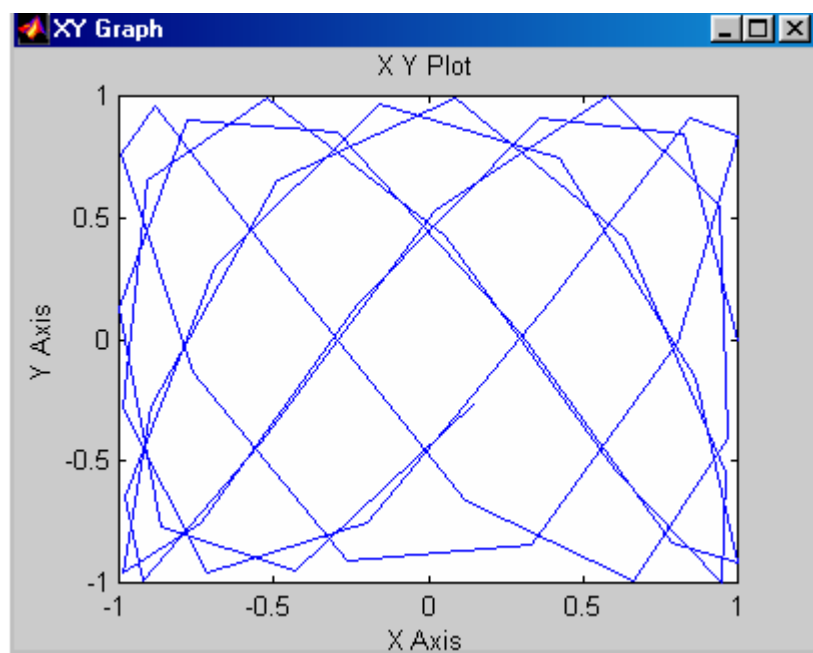


Загварчлалын үр дүн:

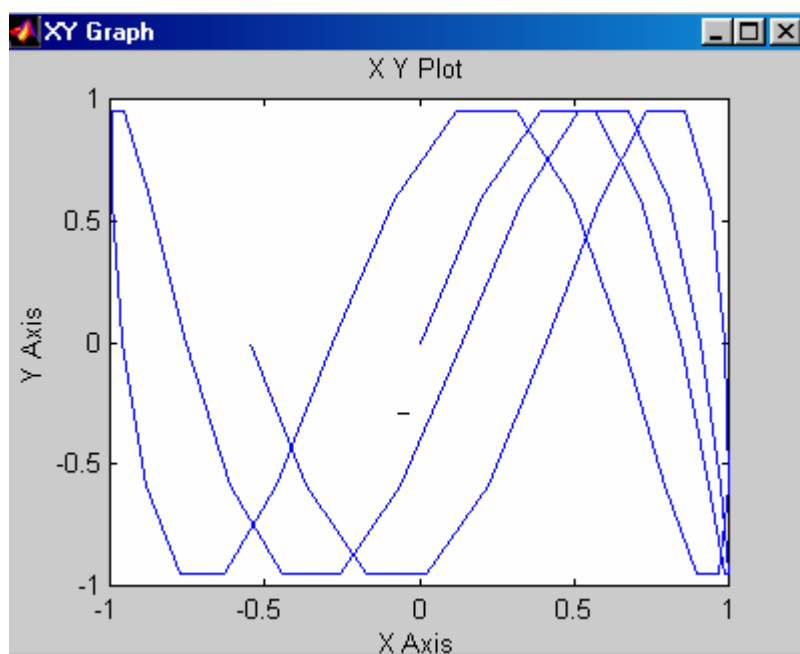
Давтамжийн харьцаа 1:1, фазын зөрүү $\pi/4$ байхад



Давтамжийн харьцаа 3:5, фазын зөрүү $\pi/2$ байхад



Давтамжийн харьцаа 1:3,1415926, фазын зөрүү 0 байхад

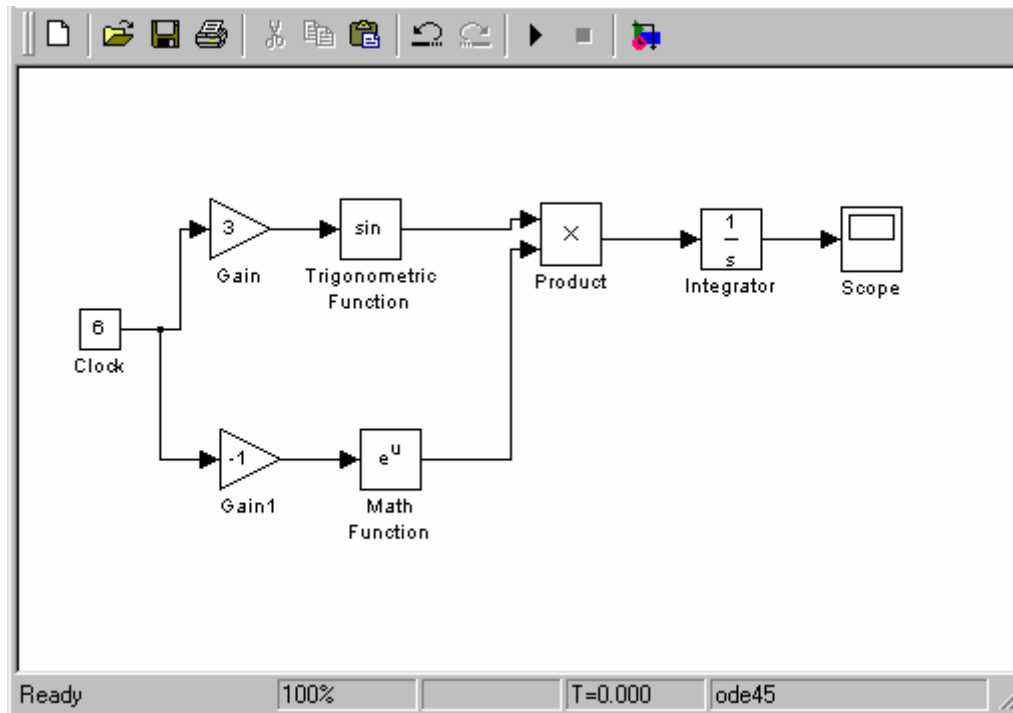


Даалгавар 2

Шийд:

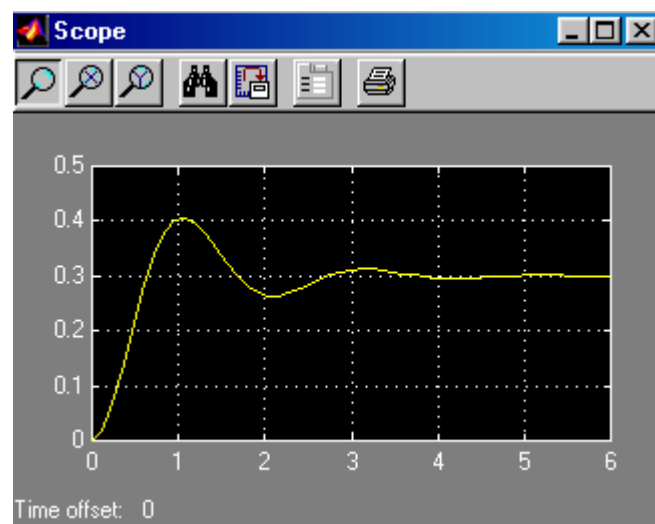
Хугацааны хамааралтай $f(t)$ сигналыг үүсгэхийн тулд $f_1(t)=t$ функц шаардлагатай.

Sources блокоос Clock буюу Ramp блокийг хэрэглэх



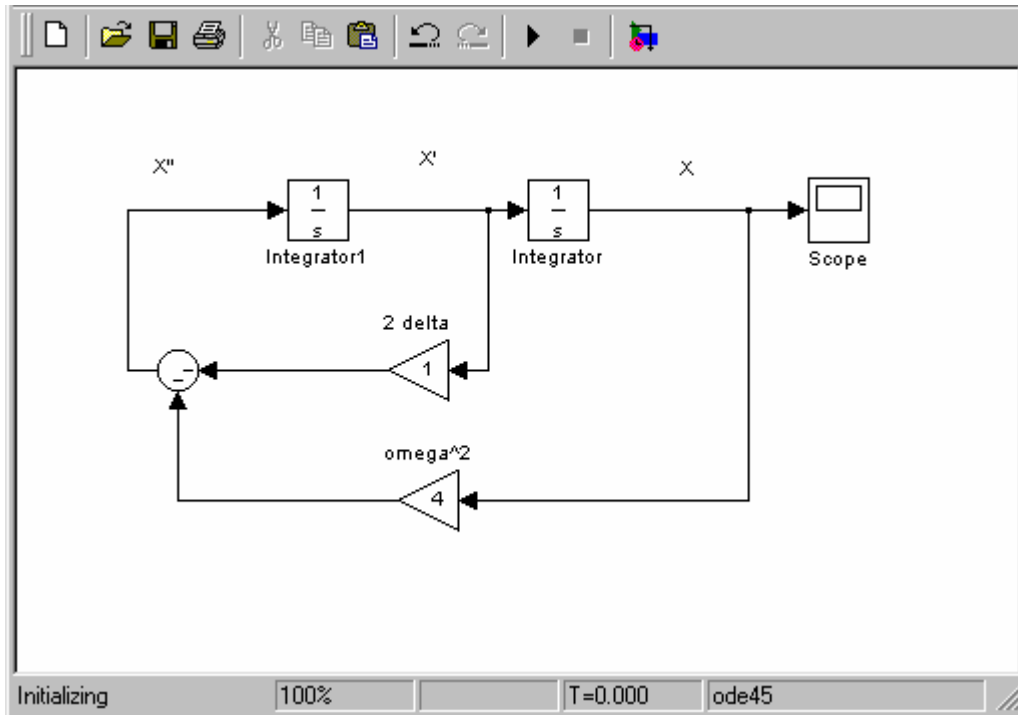
Интегралчлалын анхны нөхцөл 0

Симуляци хийх үеийн Simulation Parameters – ийн утга 6.0

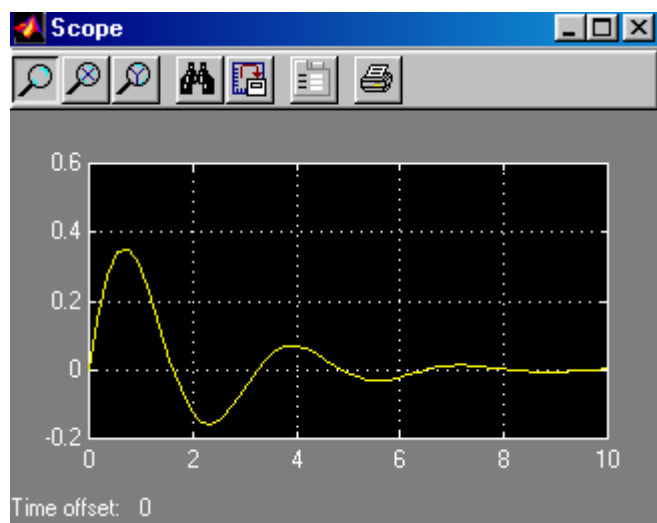


Даалгавар 3

Шийд: Загвар



Загварчлалын үр дүн:

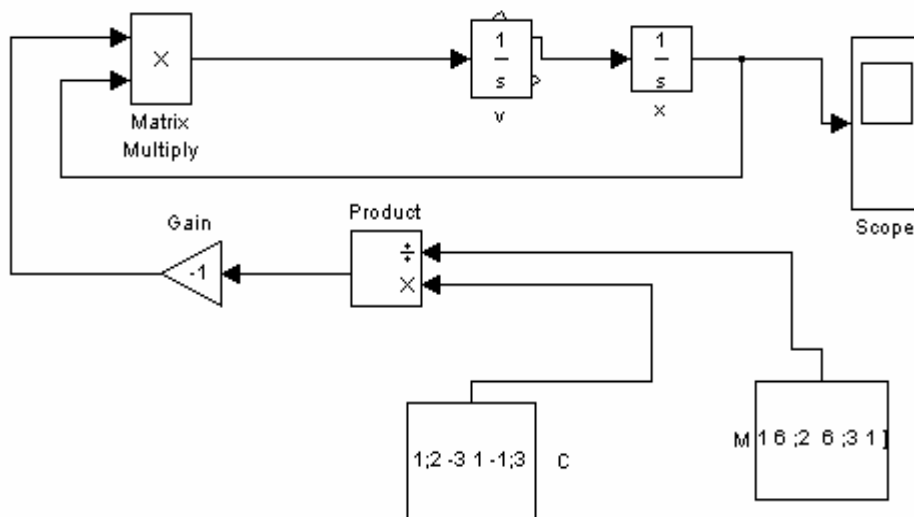


Даалгавар 4

1. Анхны утга өгсөн үеийн чөлөөт хэлбэлзэлийг матрицийн хэлбэрт бичвэл:

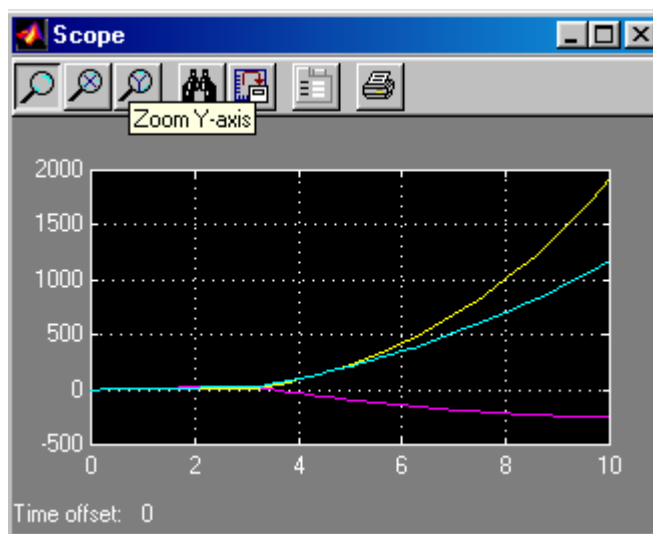
$M \dot{x}'' + C \dot{x} = 0$ эндээс шийд нь $x'' = - M^{-1} C \dot{x}$ ба

Загвар нь:



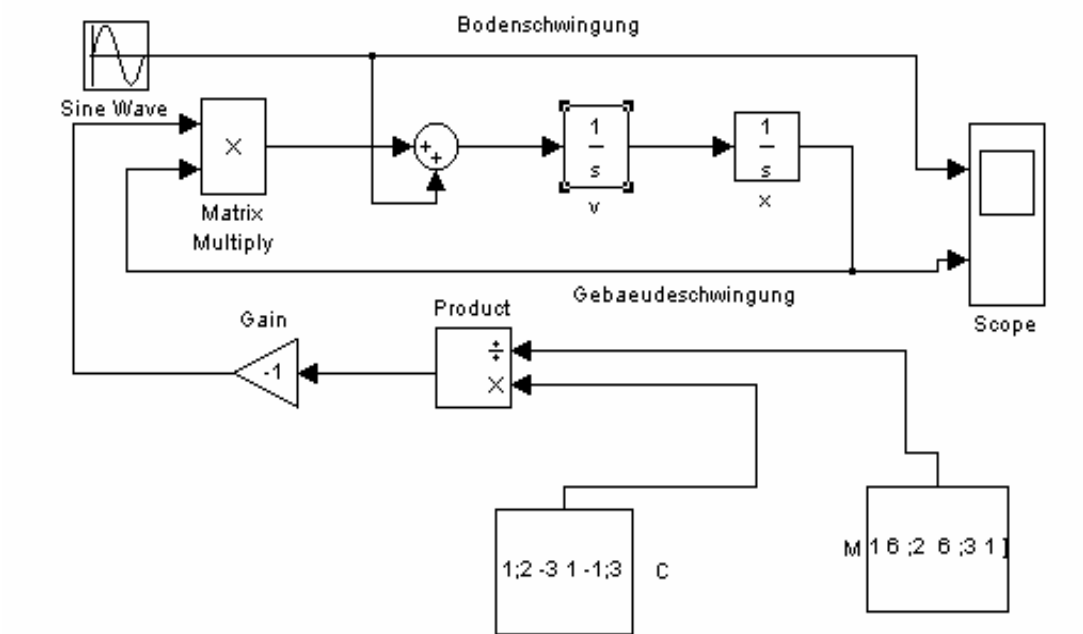
Product блокйг $/*$ параметртэй болгож $M^{-1} C$ – г тооцоолно.

Загварчлалын үр дүн:

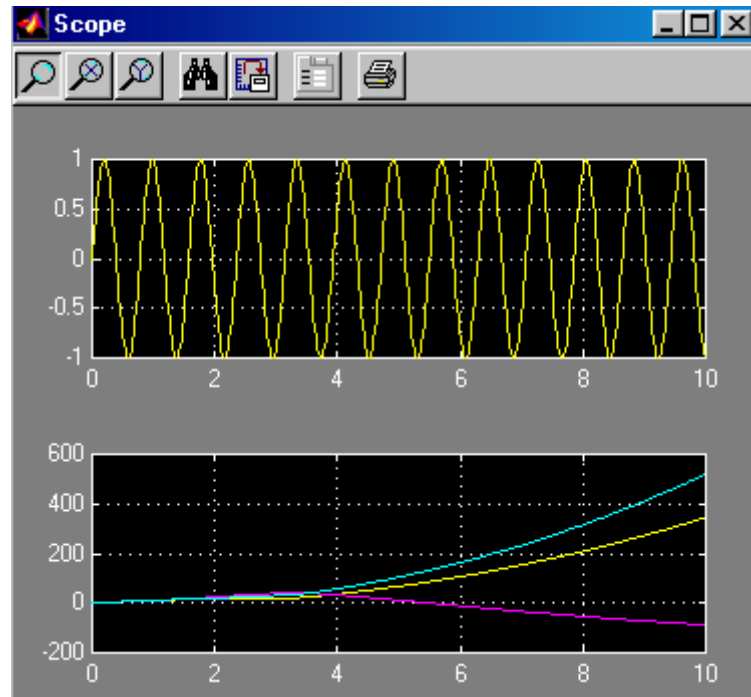


2. Синус сигналаар өдөөгдсөн хэлбэлзэлийн загварыг доор үзүүлэв.

Доор үзүүлсэн загварыг $\omega=1.2$ 1/сек, $\omega=12$ 1/сек, $\omega=120$ 1/сек байхад тус тус симуляци хийж үр дүнг гаргана уу?

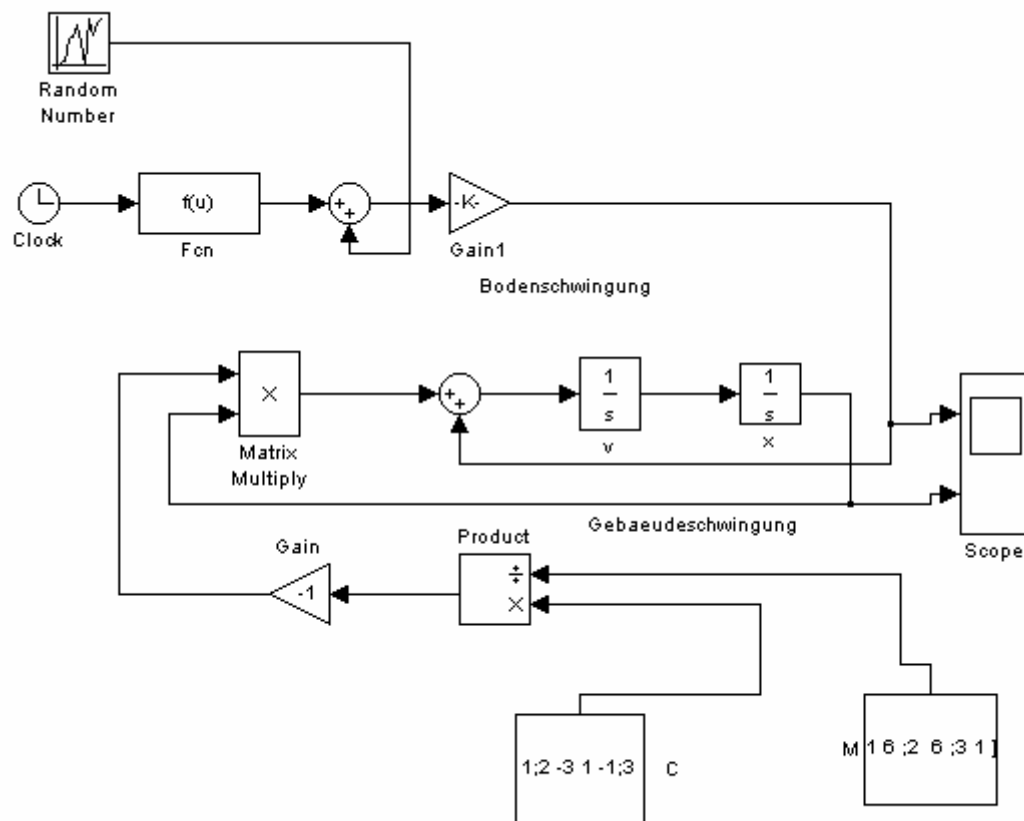


Загварчлалын үр дүн:



3. Бодит өдөөлт

Загвар:



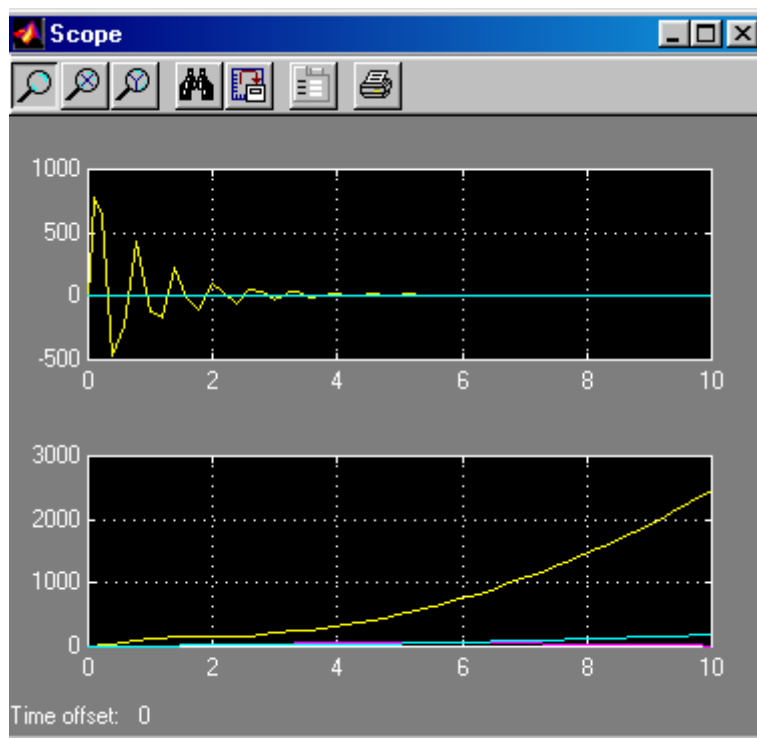
Бодит өдөөлт гаргахын тулд буурах хэлбэлзэлийг шуугиантай хольж загварчлана.

Буурах хэлбэлзэлийг функцийн блокоор гаргана. Үүнд:

$$f(u) = 1e3 * \sin(10 * u[1]) * \exp(-1.1 * u[1])$$

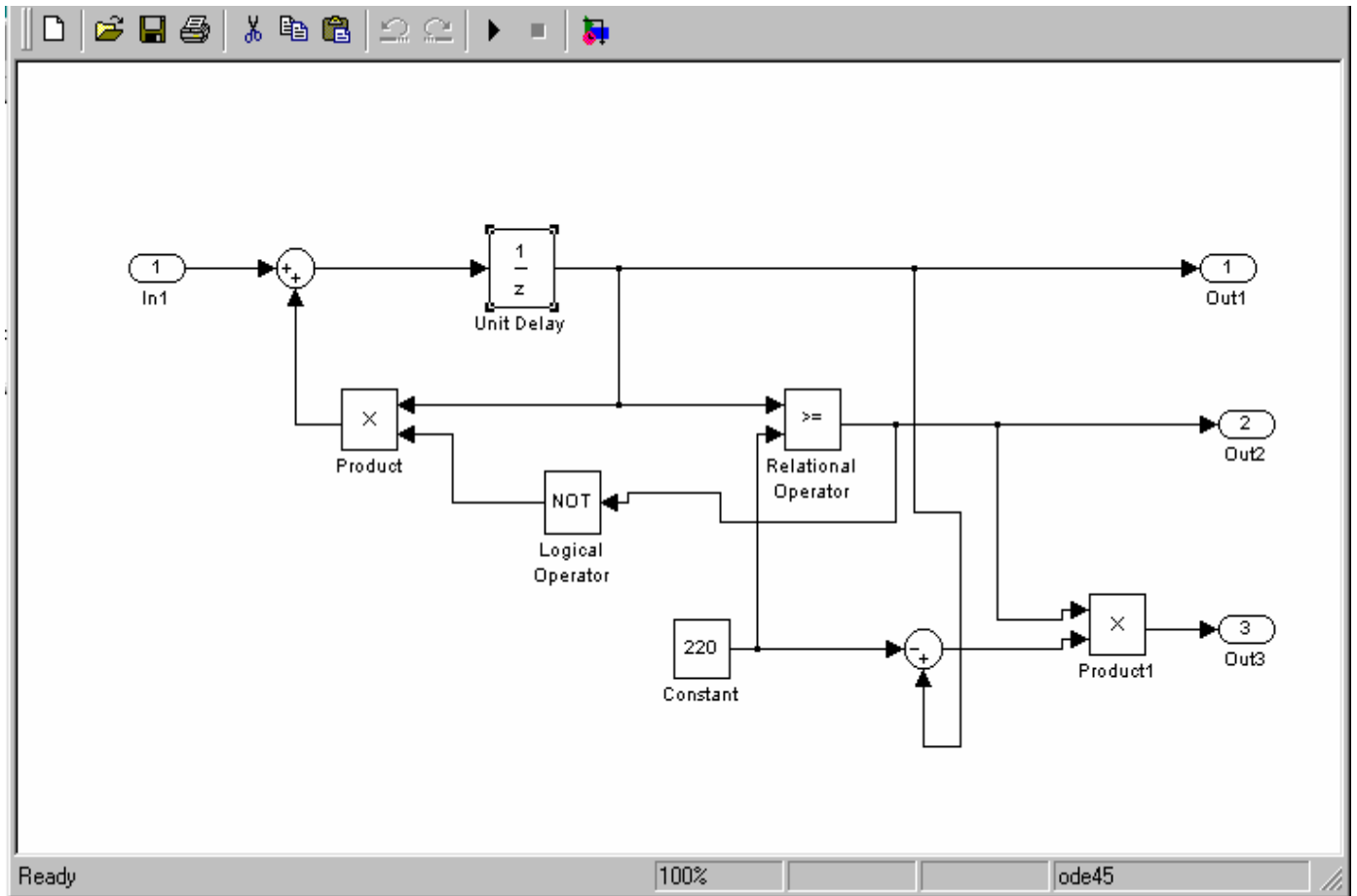
K векторыг [1 0 0] -ээр үржиж хувиргана.

Загварчлалын үр дүнг удаах графикт үзүүлээ.

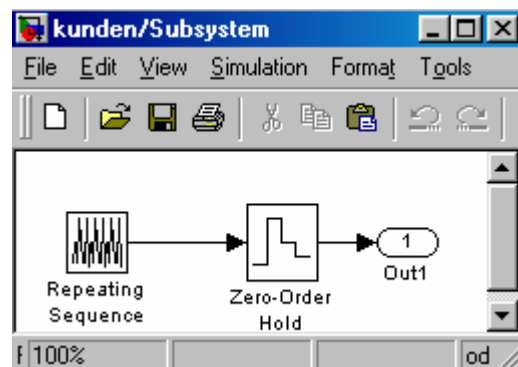


Даалгавар 5

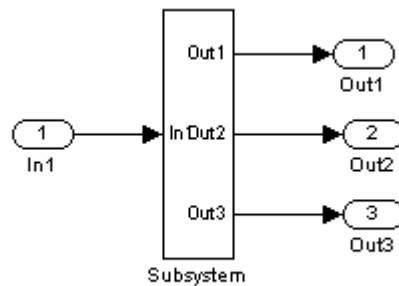
Автоматын загвар:



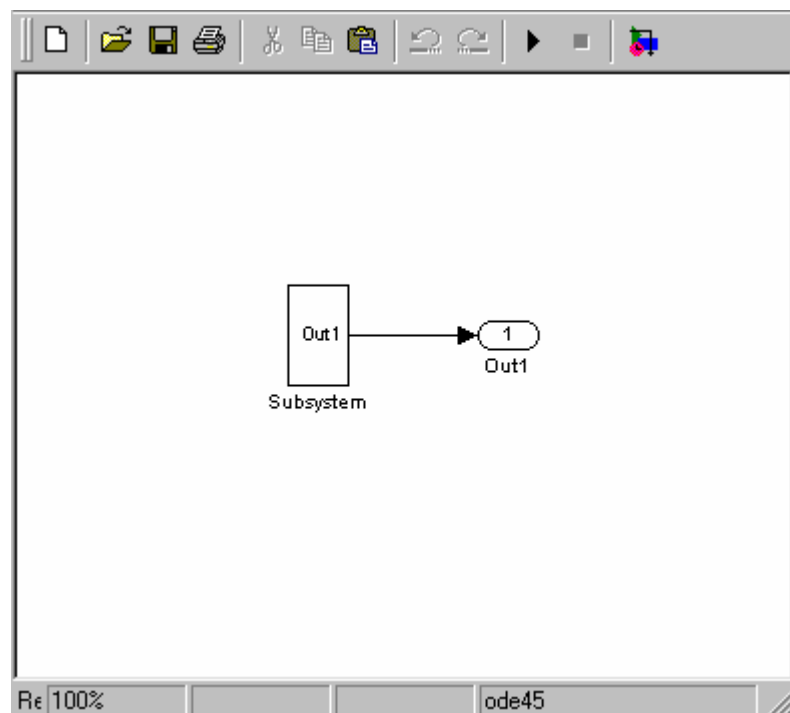
Худалдан авагчийн загвар:



Автоматын дэд загвар:



Худалдан авагчийн дэд загвар:

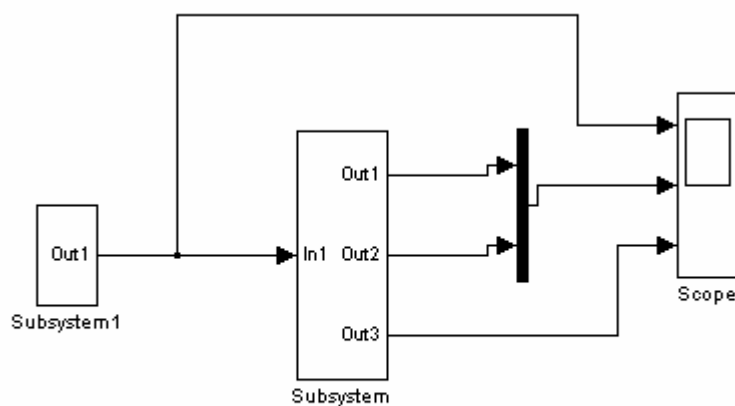


Тайлбар:

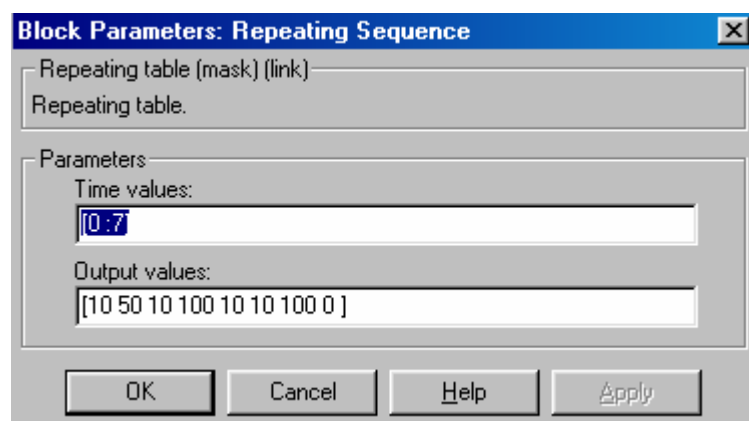
Repeating Sequence блок нь тодорхой хугацаанд давтагдах утгуудын дарааллыг үүсгэдэг.

Гэхдээ завсрын хугацаа гарвал шугаман интерполяци хийдэг.

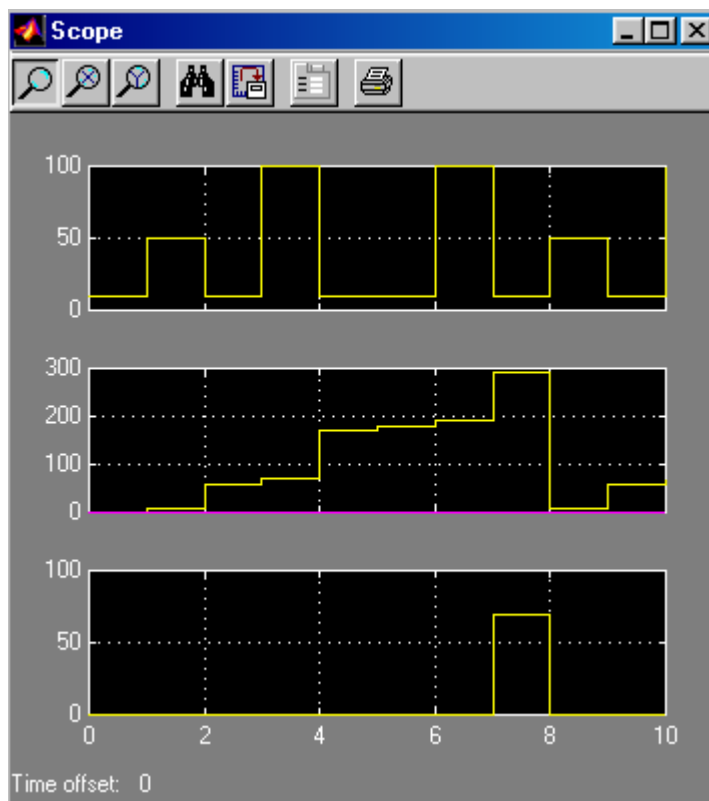
Zero –Order Hold блок нь шугаман интерполяци хийсэн утгын дарааллаас дискрет хугацааны утгын дарааллыг гаргадаг.

Ерөнхий загвар:

Repeating Sequence блоктын параметрыг тодорхойлох жишээ:

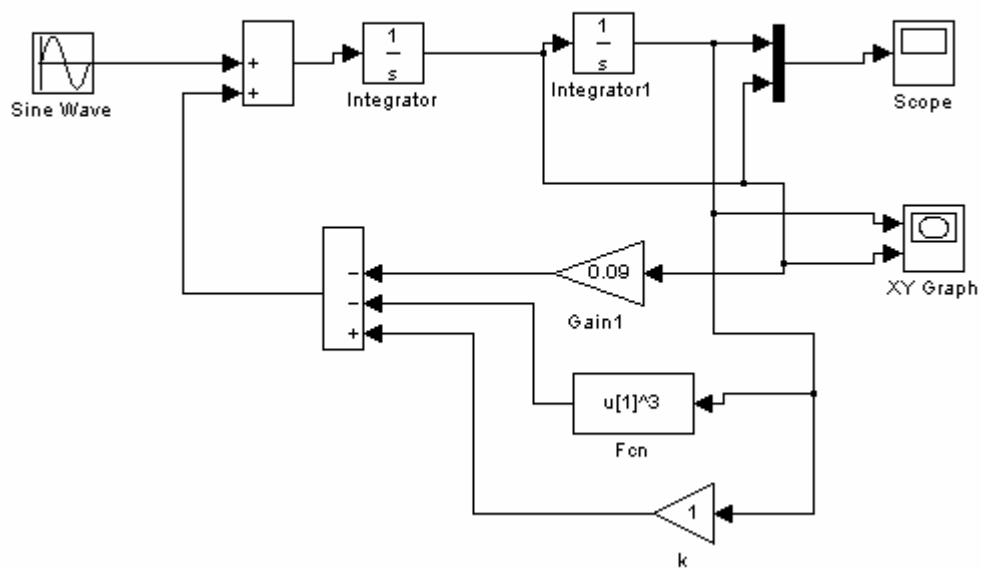


Загварчлалын үр дүн:

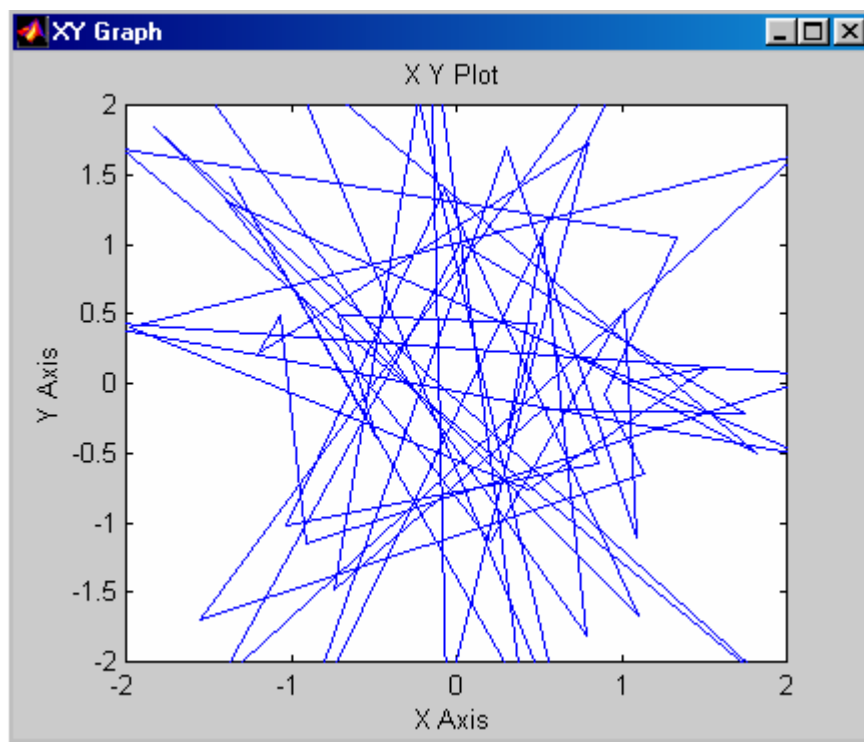


Даалгавар 6:

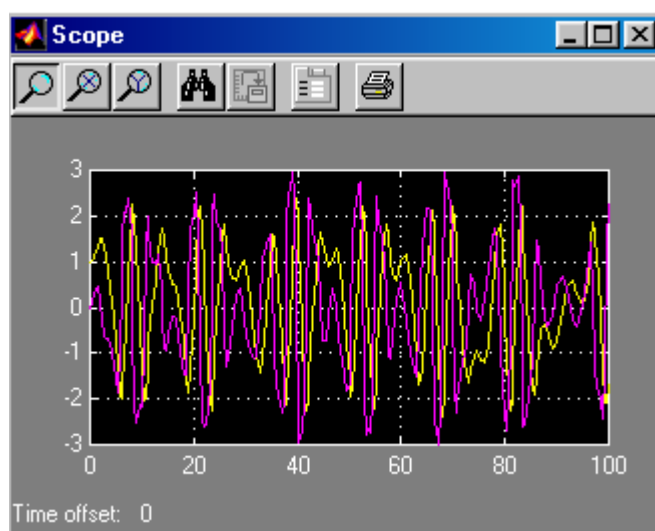
Загвар:

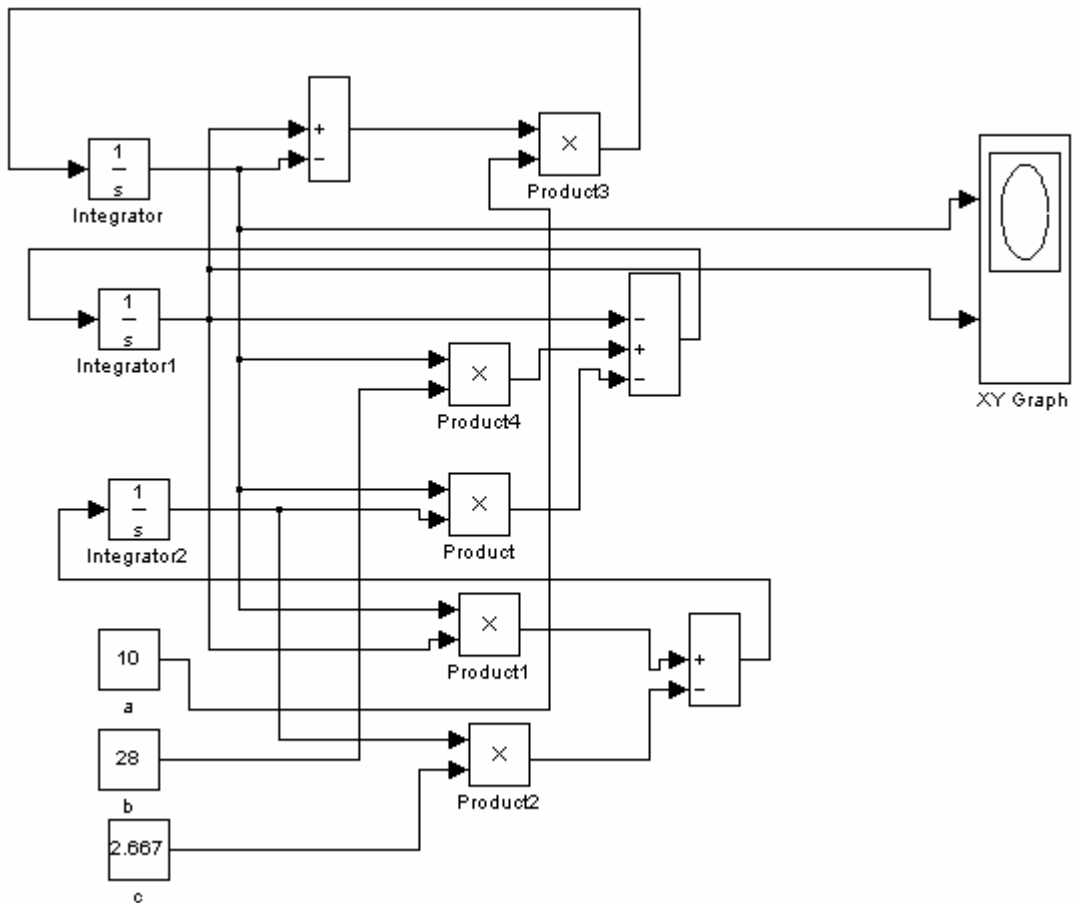


Фазын орон зай дахь хэлбэлзэл

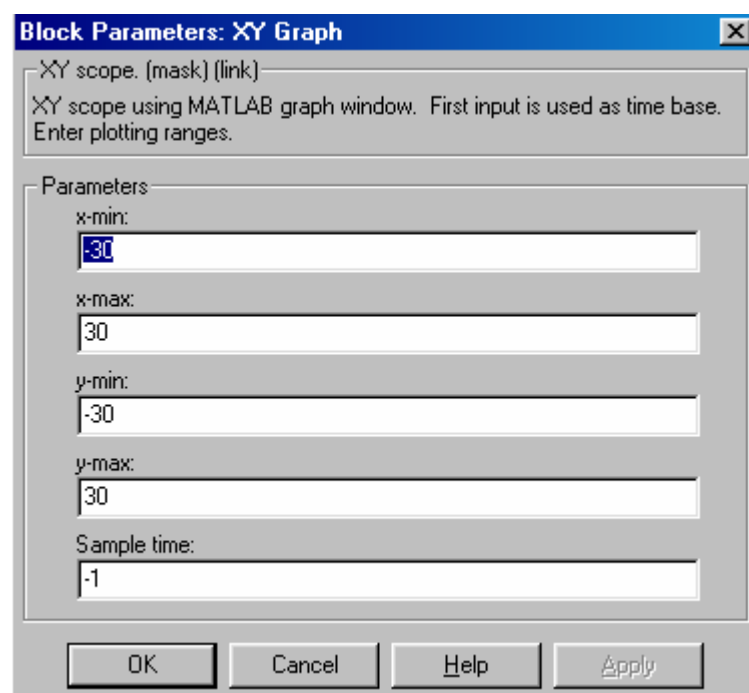
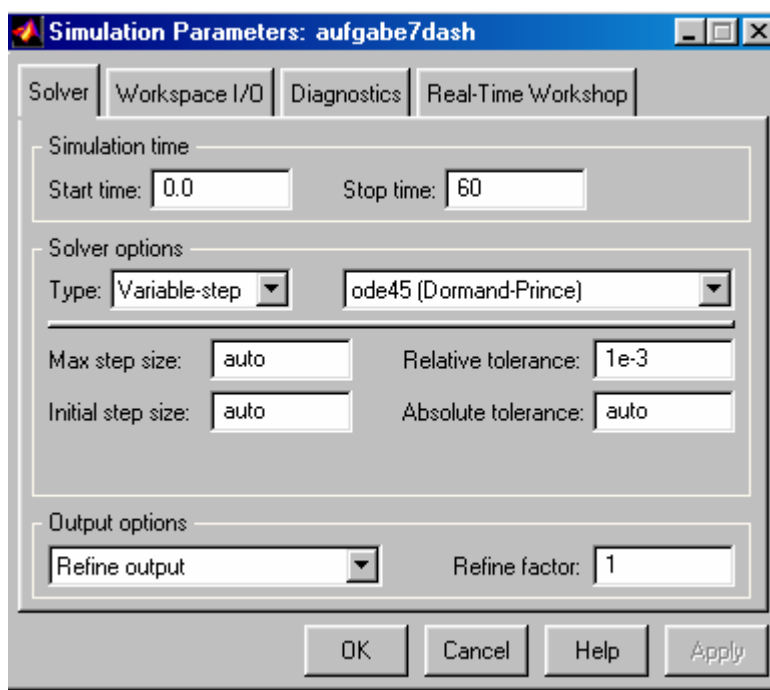


Хугацааны орон зай дахь хэлбэлзэл

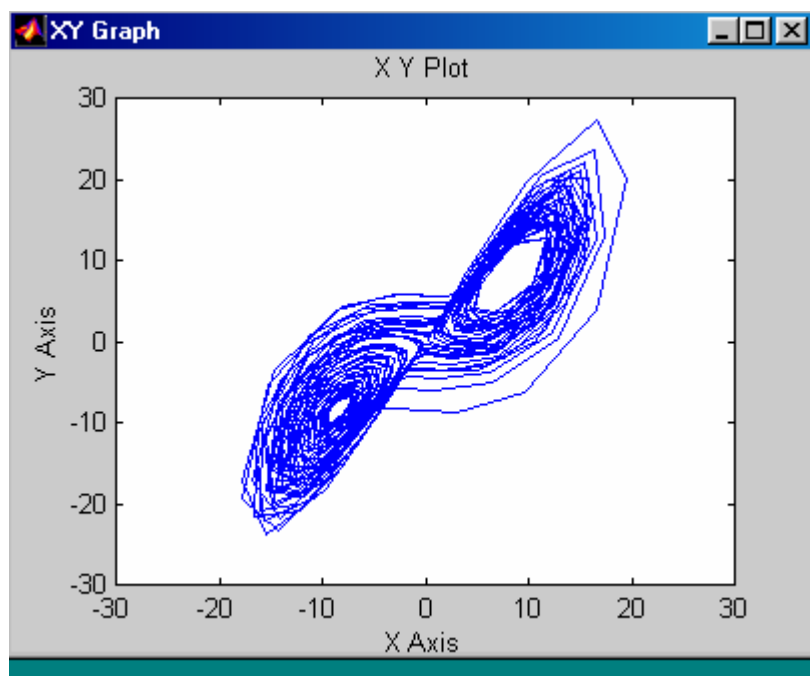


Даалгавар 7**Загвар:****Заавар:**

Simulation Parameters-ын Stop time -г 60 сек, XYGraph-ийн Block Parameters $x\text{-min}=-30$, $x\text{-max}=3$, $y\text{-min}=-30$, $y\text{-max}=30$ гэж тохируулсны дараа загварчлалыг ажиллуулна (Доорхи зургийг үзнэ үү?).

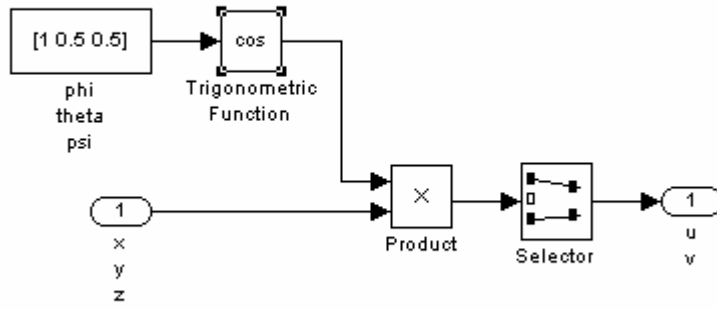


Дээр тохируулсан параметрт гаргах загварчлалын үр дүн:

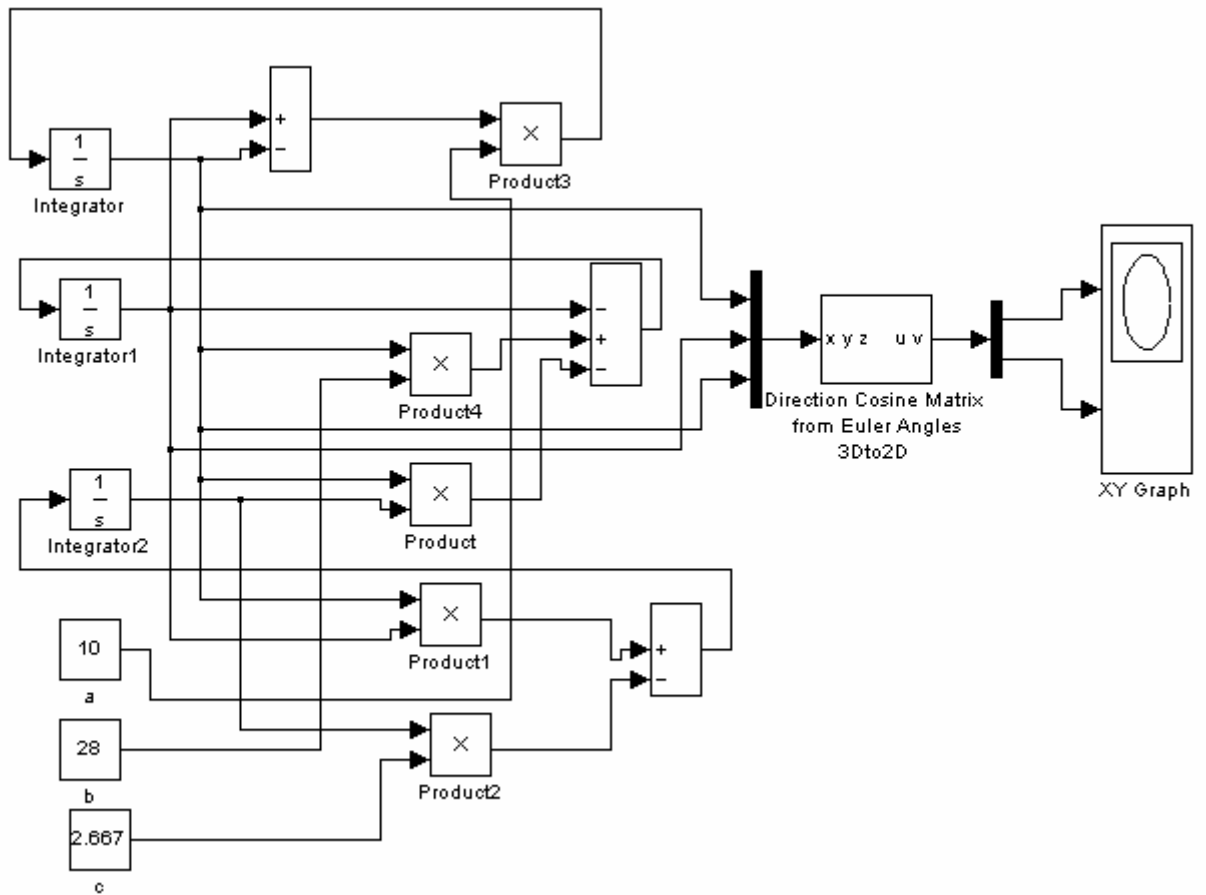


Загварын сайжруулахын тулд:

- Проекцийн 3d-координатын системийг эргүүлэх
- Үүний тулд эргэлтийн матрицийг гурван өнцгөөр эргүүлэх (Ейлерийн өнцөг)
- Мөн дэд системийг гаргах



Ерөнхий загвар:



Загварчлалын үр дүн:

