Московский авиационный институт

(национальный исследовательский университет)

**Институт информационных технологий и прикладной математики**

Кафедра мехатроники и теоретической механики

**Лабораторные работы № 1-3 по курсу «МЕТОДЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ ДИНАМИКИ СИСТЕМ ТЕЛ: ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД» в среде OpenModelica**

Работу выполнил(а): Шиляева Н. С.

Группа: М8О-304Б-18

Преподаватель: Беличенко М. В.

Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_

Мосвка, 2020

**Лабораторная работа № 1**

**Задание:**

1. **Построение фазового портрета системы дифференциальных уравнений второго порядка**

Целью первой ознакомительной лабораторной работы является численное интегрирование системы дифференциальных уравнений с помощью средств OpenModelica, и построение графиков её решений, в частности, создание фазового портрета системы.

Пусть дано дифференциальное уравнение второго порядка:

Представим его в нормальной форме Коши, введя дополнительную переменную :

Код программы

model lr\_1

Real x[5];

Real y[5];

initial equation

for i in 1:5 loop

x[i]=2-2/5\*(i-1);

y[i]=0;

end for;

equation

for i in 1:5 loop

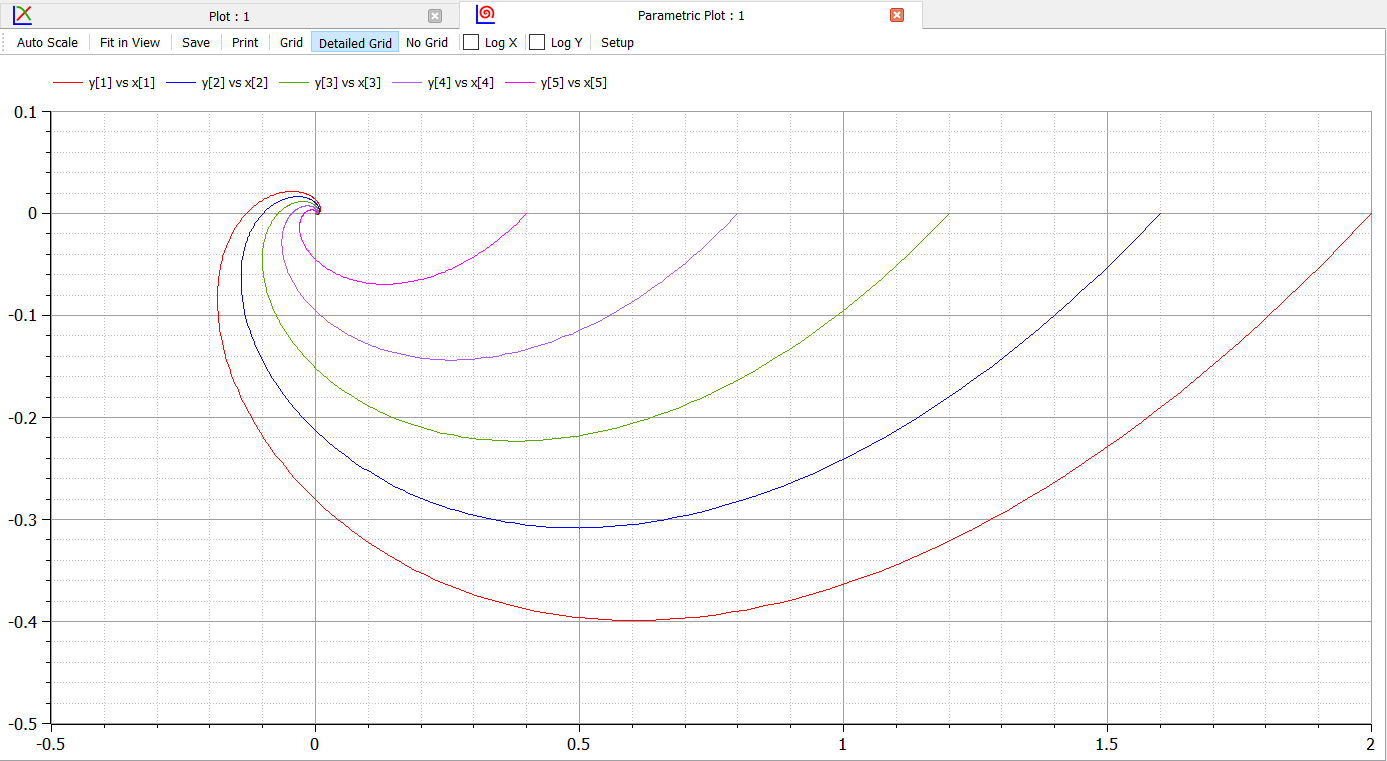
der(x[i])=-9\*x[i] + 20\*y[i];

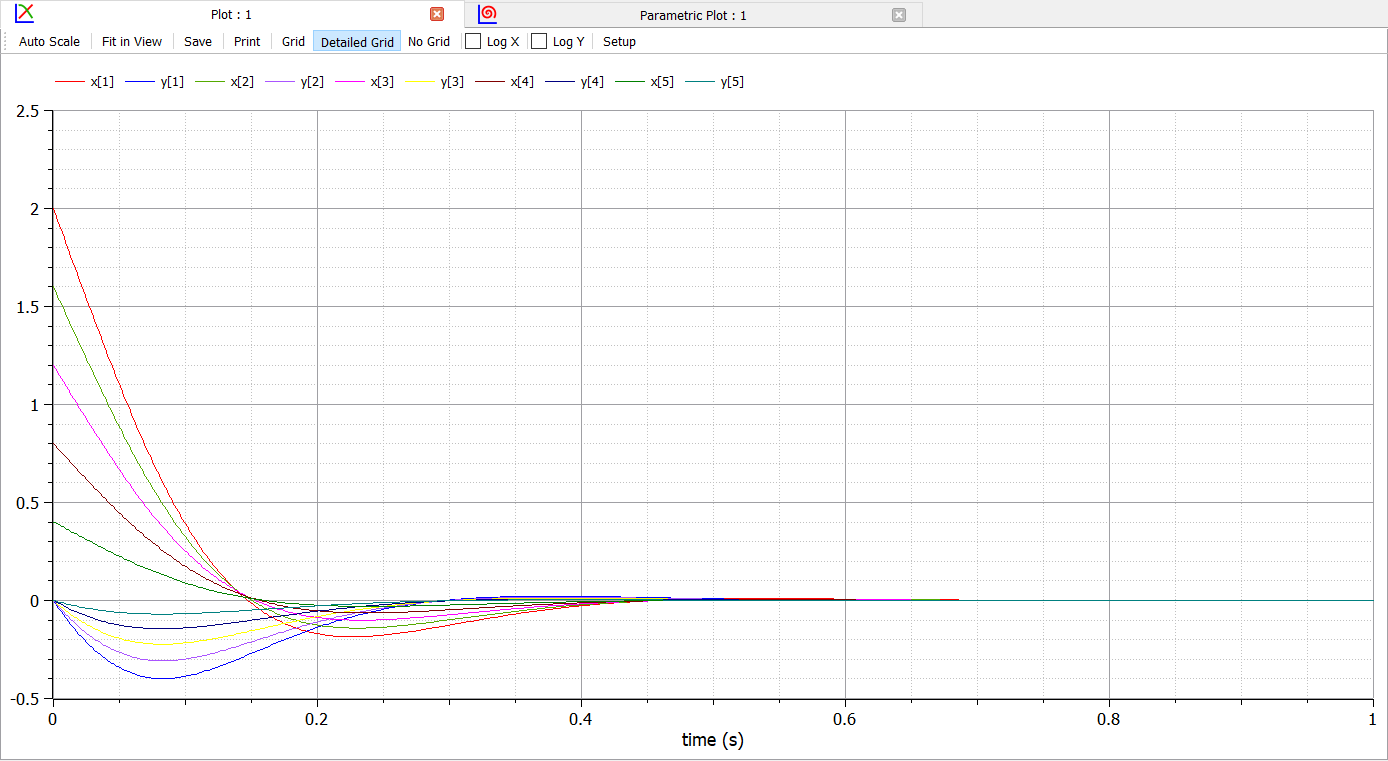
der(y[i])=4\*x[i]\*y[i] - 5\*x[i] -10\*y[i];

end for;

end lr\_1;

Графики





Код программы

model lr\_2

import Modelica.SIunits;

Modelica.SIunits.Angle phi (start=10);

Modelica.SIunits.AngularVelocity omega (start=0);

parameter Modelica.SIunits.Mass m=2;

parameter Modelica.SIunits.Length l=20;

parameter Modelica.SIunits.Acceleration g=9.81;

parameter Modelica.Mechanics.MultiBody.Frames.Orientation O= Modelica.Mechanics.MultiBody.Frames.axesRotations({1,2,3},{0,0,0},{0,0,0});

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape Palka (shapeType="cylinder", length=l, width=l/50, height=l/50, lengthDirection={sin(phi),-cos(phi),0}, widthDirection={0,0,1}, color={255,0,0}, specularCoefficient=0.5, r={0,0,0}, R=O, r\_shape={0,0,0});

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape Sphere (shapeType="sphere", length=l, width=l/3, height=l/3, lengthDirection={sin(phi),-cos(phi),0}, widthDirection={0,0,1}, color={0,255,0}, specularCoefficient=0.5, r={l\*sin(phi),l\*(-cos(phi)),0}, R=O, r\_shape={0,0,0});

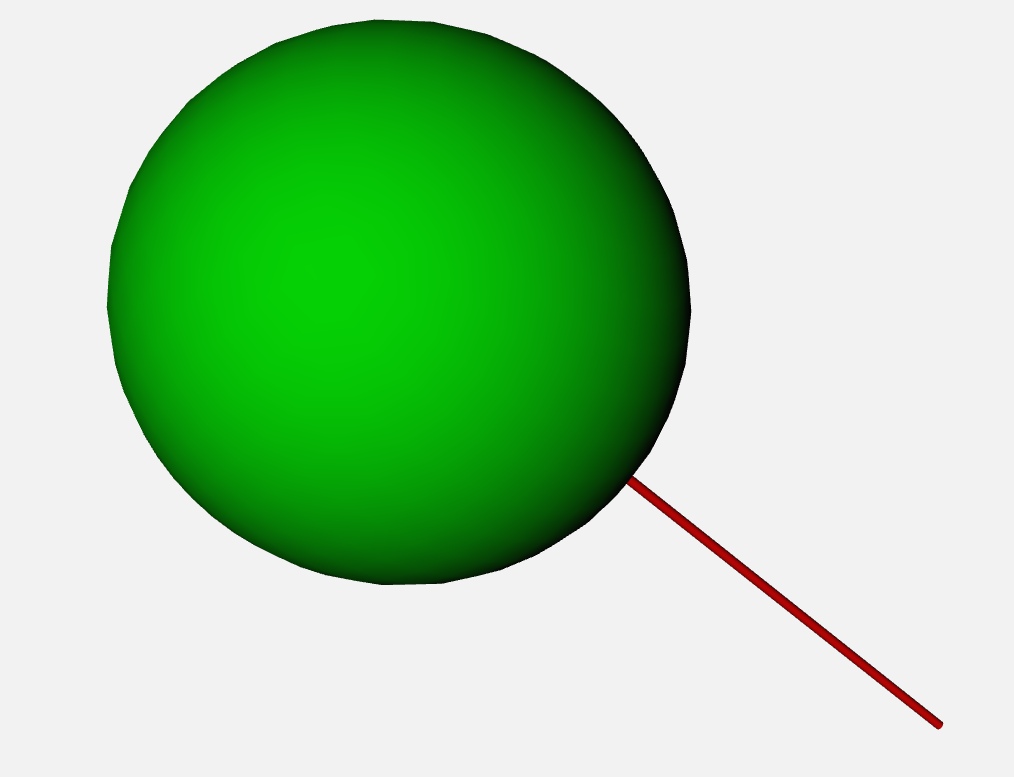
equation

omega = der(phi);

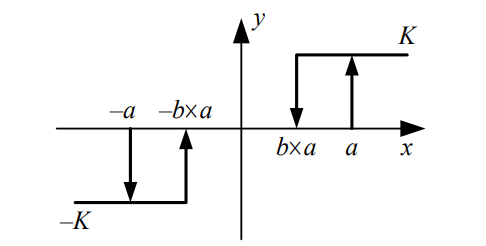
m\*(l^2)\*der(omega) = -m\*g\*l\*sin(phi);

end lr\_2;

График



**Лабораторная работа № 2**



Уравнения

Код программы

model func\_4

Real x1(start = 10);

Real v1(start = 0);

Real x2(start = 10);

Real v2(start = 0);

Real F1;

Real F2;

parameter Real K = 10;

parameter Real k2 = 2;

parameter Real k3 = 20;

parameter Real k4 = 1;

parameter Real a = 2;

parameter Real b = 0.5;

equation

if x1>(b\*a) then F1=K;

elseif x1==(b\*a) then F1=K;

elseif ((x1<(b\*a)) and (x1>(-a))) then F1=0;

elseif x1==(-a) then F1=(-K);

else F1=(-K);

end if;

if x2<(-b\*a) then F2=(-K);

elseif x2==(-b\*a) then F2=(-K);

elseif ((x2>(-b\*a)) and (x2<a)) then F2=0;

elseif x2==a then F2=K;

else F2=K;

end if;

der(v1) = (-F2) - k2 \* x1 + k3 \* sin(k4 \* time);

der(v2) = (-F1) - k2 \* x2 + k3 \* sin(k4 \* time);

der(x1) = v1;

der(x2) = v2;

end func\_4;

Графики x(t) и F(t)

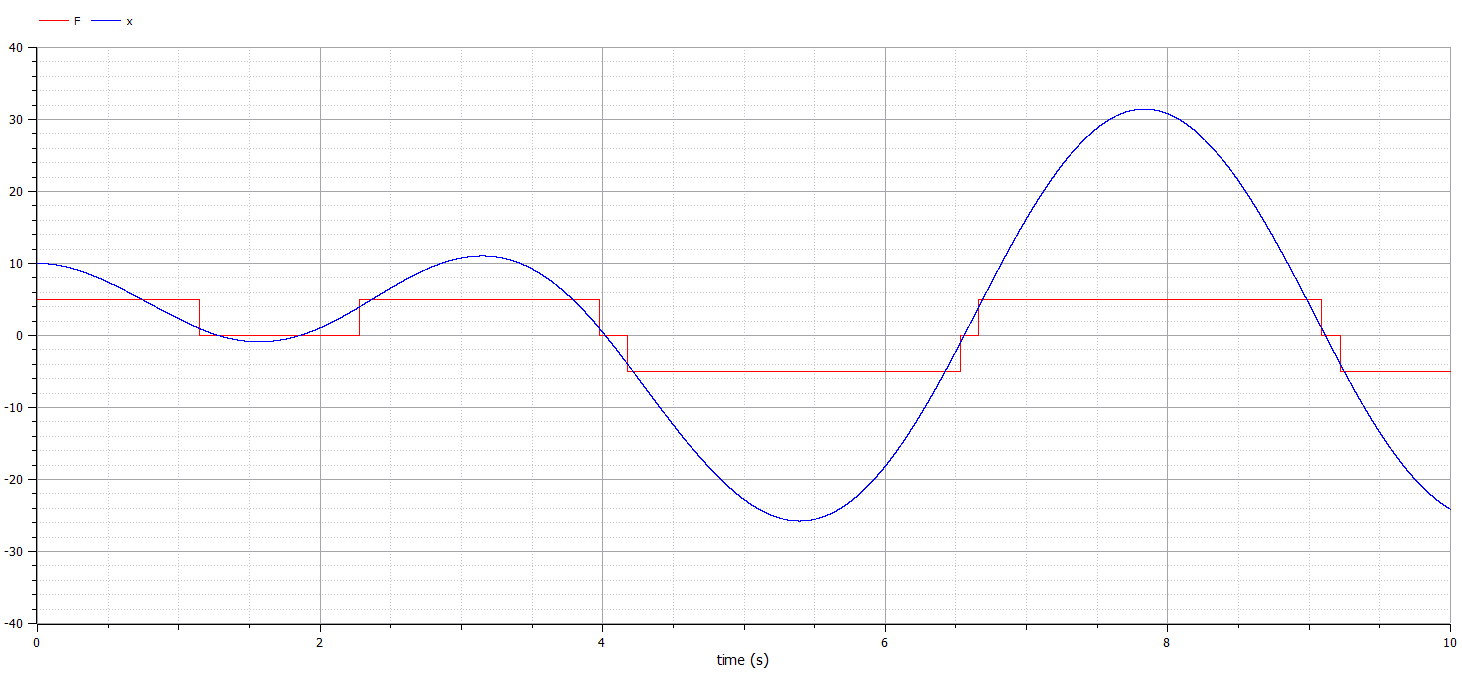
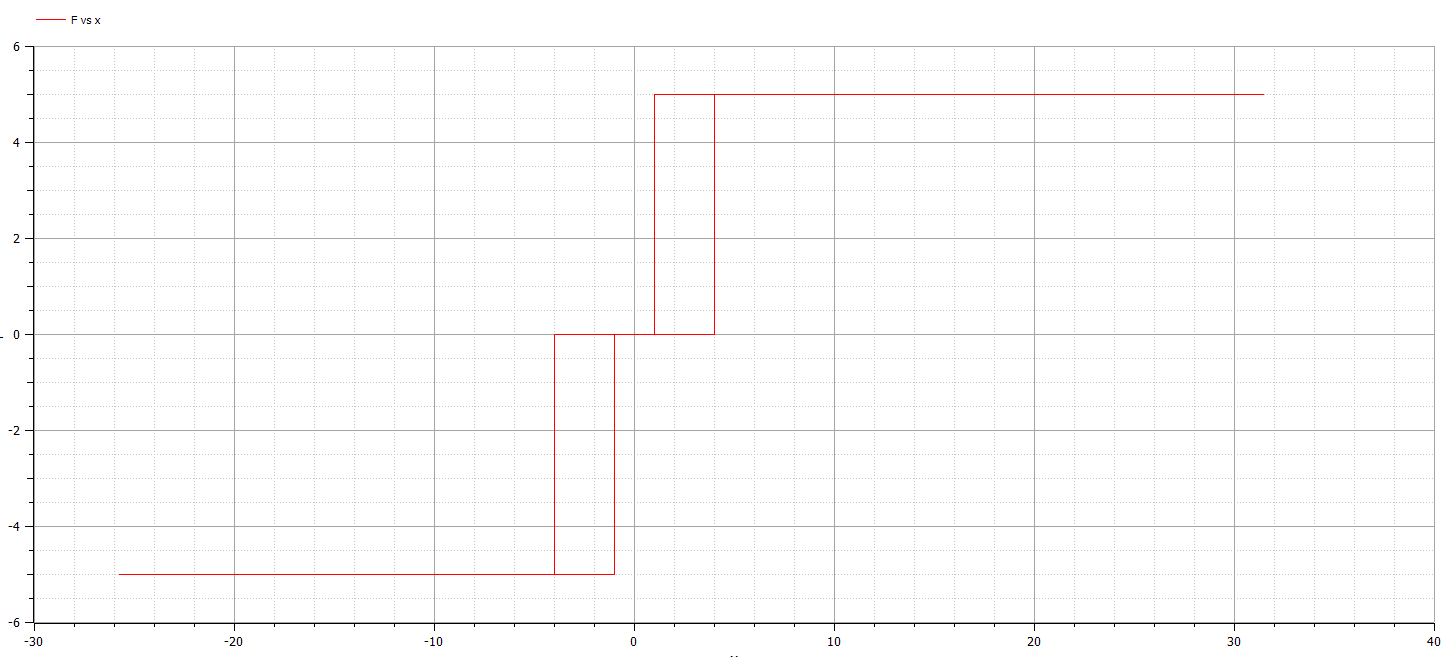
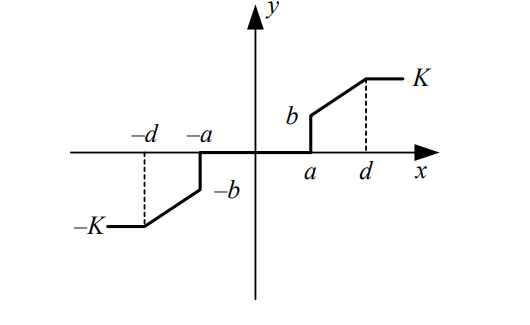


График F(x)





Код программы

model func\_3

Real x(start = 10);

Real v(start = 0);

Real F;

parameter Real K = 10;

parameter Real k2 = 2;

parameter Real k3 = 20;

parameter Real k4 = 1;

parameter Real a = 2;

parameter Real b = 2;

parameter Real d = 4;

equation

if x<=(-d) then

F=-K;

elseif x==(-a) then

F=(-b);

elseif x==a then

F=b;

elseif ((x>a) and (x<d)) or ((x<(-a)) and (x>(-d))) then

F=(K/d)\*x;

elseif (x>=d) then

F=K;

else

F=0;

end if;

der(v) = (-F) - k2 \* x + k3 \* sin(k4 \* time);

der(x) = v;

end func\_3;

Графики F(t) и x(t)

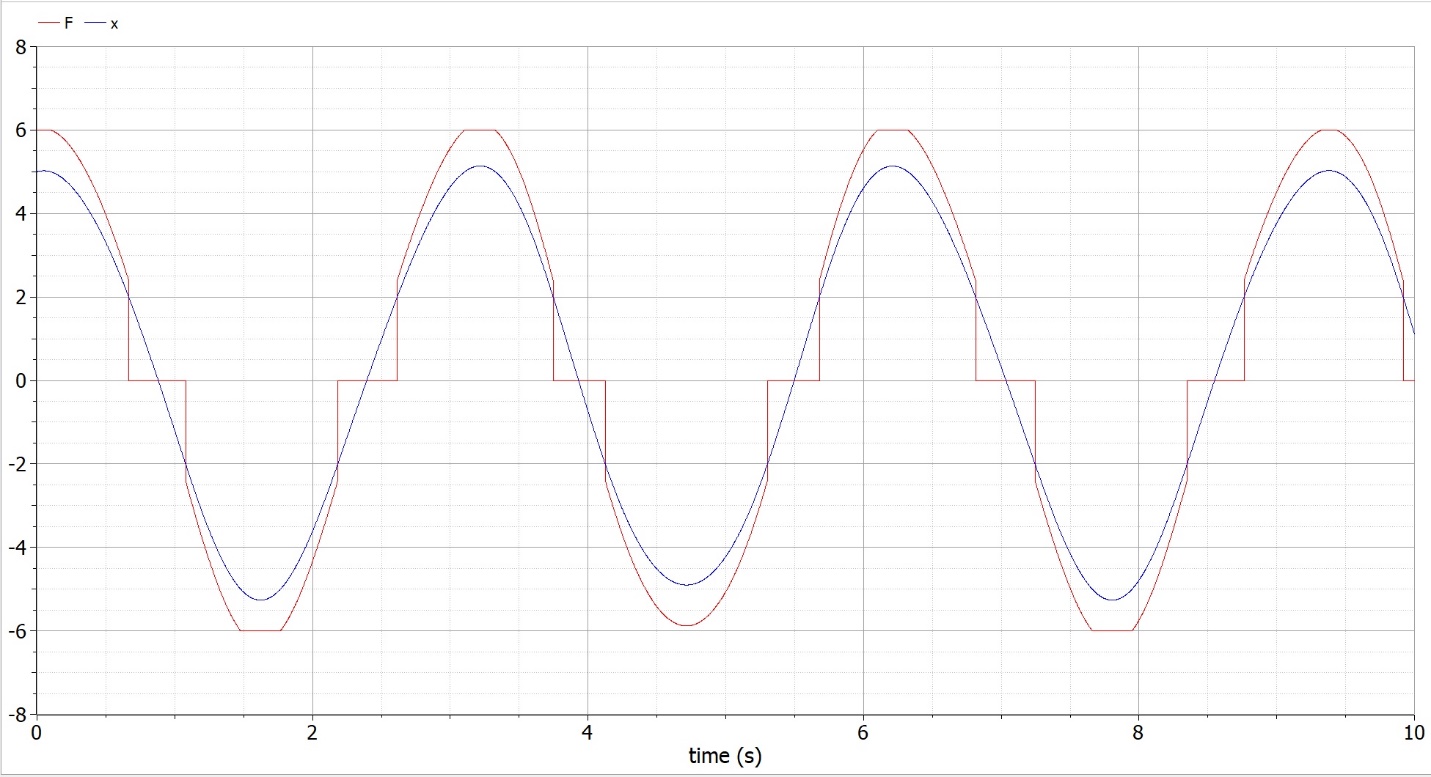
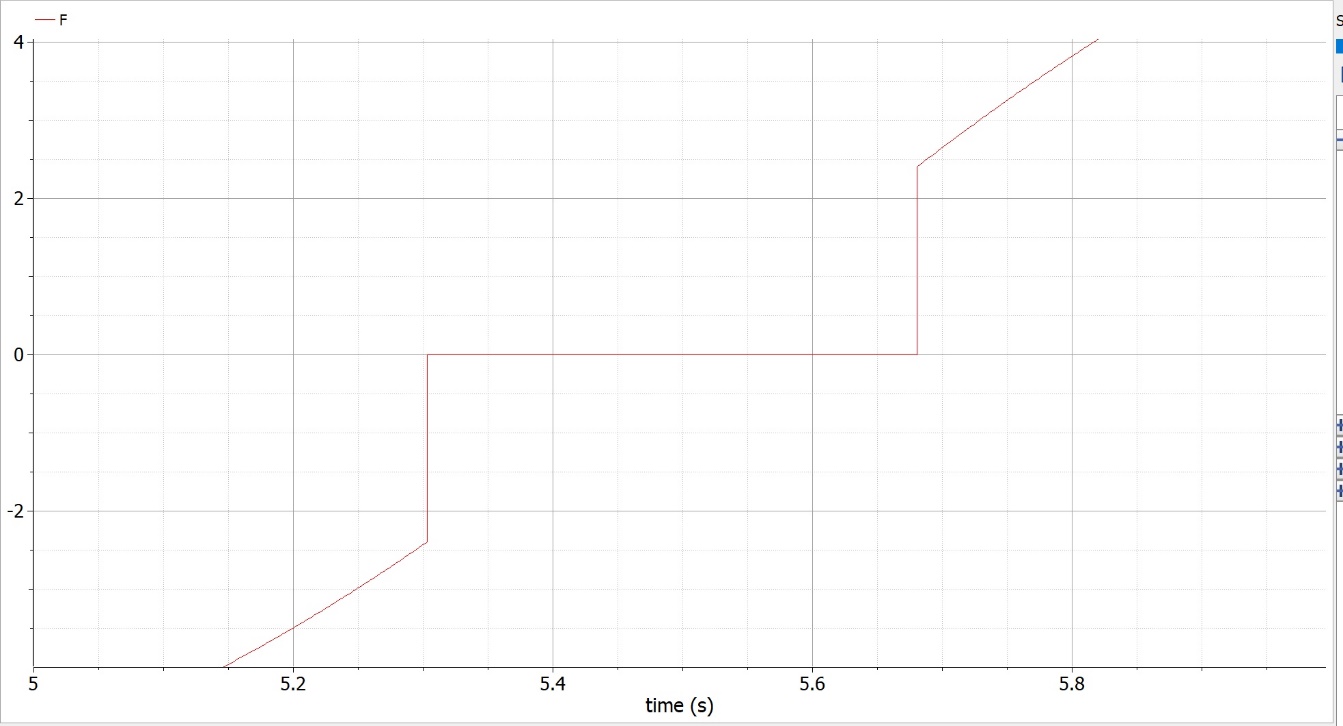


График F(x)



**Лабораторная работа № 3**

Модифицированный класс резистора:

model my\_resist

parameter Modelica.SIunits.Resistance R(start=1)

"Resistance at temperature T\_ref";

parameter Modelica.SIunits.Temperature T\_ref=300.15 "Reference temperature";

parameter Real alpha = 0;

parameter Modelica.SIunits.Current Io=1;

extends Modelica.Electrical.Analog.Interfaces.OnePort;

extends Modelica.Electrical.Analog.Interfaces.ConditionalHeatPort( T = T\_ref);

Modelica.SIunits.Resistance R\_actual;

equation

assert((1 + alpha\*(T\_heatPort - T\_ref)) >= Modelica.Constants.eps, "Temperature outside scope of model!");

R\_actual = R\*(1 + alpha\*(1/Io));

v = R\_actual\*i;

LossPower = v\*i;

annotation (),

Icon(coordinateSystem(

preserveAspectRatio=true,

extent={{-100,-100},{100,100}},

grid={2,2}), graphics={

Rectangle(

extent={{-70,30},{70,-30}},

lineColor={0,0,255},

fillColor={255,255,255},

fillPattern=FillPattern.Solid),

Line(points={{-90,0},{-70,0}}, color={0,0,255}),

Line(points={{70,0},{90,0}}, color={0,0,255}),

Text(

extent={{-144,-40},{142,-72}},

lineColor={0,0,0},

textString="R=%R"),

Line(

visible=useHeatPort,

points={{0,-100},{0,-30}},

color={127,0,0},

smooth=Smooth.None,

pattern=LinePattern.Dot),

Text(

extent={{-152,87},{148,47}},

textString="%name",

lineColor={0,0,255})}),

Diagram(coordinateSystem(

preserveAspectRatio=true,

extent={{-100,-100},{100,100}},

grid={2,2}), graphics={

Rectangle(extent={{-70,30},{70,-30}}, lineColor={0,0,255}),

Line(points={{-96,0},{-70,0}}, color={0,0,255}),

Line(points={{70,0},{96,0}}, color={0,0,255})}));

end my\_resist;

