МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(национальный исследовательский университет)»

ОТЧЕТ ПО КОНТРОЛЬНОЙ РАБОТЕ

по курсу «Виртуальные динамические модели и цифровые двойники»

Выполнил:

Студент группы М80-201М-21

Фейзуллин Кирилл Маратович

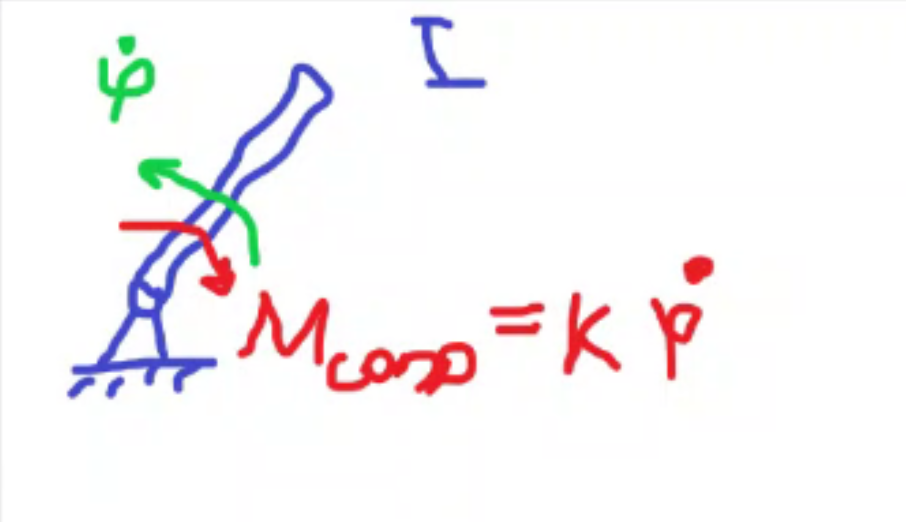
Вариант - 13

**Контрольная работа**

**Задание 1**

По заданной схеме смоделировать динамику системы плоских тел с учетом коэффициента затухания опоры – ржавая опора.

**Вариант 1**



**Результат работы**

Код программы:

package Controsha

model FreeEnd

ForceOutput FO;

equation

FO.X = 0;

FO.Y = 0;

FO.Fx = 0;

FO.Fy = 0;

FO.M = 0;

end FreeEnd;

model PalkaNaOpore

parameter Modelica.Units.SI.Length L1 = 2;

parameter Modelica.Units.SI.Length L2 = 4;

TwoPortRod2D Palka1(L = L1, Color = {0, 255, 80}, m = 2);

//TwoPortRod2D Palka2(L = L2, Color = {255, 255, 0}, m=4);

**Support2D Opora(Xp = 0, Yp = 0, Xt = -L1 / 2, Yt = 0, K = 5);**

//Joint2D Sharnir(Xt1 = L1 / 2, Yt1 = 0, Xt2 = -L1 / 2, Yt2 = 0);

FreeEnd Hvost;

equation

connect(Palka1.Body\_Out, Opora.Body\_In);

// connect(Palka1.Body\_Out, Sharnir.Body\_In1);

//connect(Palka2.Body\_Out, Sharnir.Body\_In2);

connect(Palka1.F\_A, Opora.FO);

// connect(Palka1.F\_B, Sharnir.FO1);

// connect(Palka2.F\_A, Sharnir.FO2);

connect(Palka1.F\_B, Hvost.FO);

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end PalkaNaOpore;

model Body2D

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

parameter Real Color[3] = {0, 0, 255};

Modelica.Units.SI.Length X;

Modelica.Units.SI.Length Y;

Modelica.Units.SI.Angle Phi;

KinematicOutput Body\_Out;

equation

Body\_Out.X = X;

Body\_Out.Y = Y;

Body\_Out.Phi = Phi;

end Body2D;

connector KinematicInput

input Modelica.Units.SI.Length X;

input Modelica.Units.SI.Length Y;

input Modelica.Units.SI.Angle Phi;

end KinematicInput;

connector KinematicOutput

output Modelica.Units.SI.Length X;

output Modelica.Units.SI.Length Y;

output Modelica.Units.SI.Angle Phi;

end KinematicOutput;

model Rod2D

extends Fejzullin\_Lab\_5\_6.Controsha.Body2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

equation

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Rod2D;

model FreeRod

parameter Modelica.Units.SI.Length L = 2;

Rod2D Palka(L = L, Color = {0, 255, 80});

equation

Palka.X = L / 2 \* sin(time);

Palka.Y = L / 2 \* cos(time);

Palka.Phi = time;

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end FreeRod;

model Wheel2D

extends Fejzullin\_Lab\_5\_6.Controsha.Body2D;

parameter Modelica.Units.SI.Length R = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape WheelShape(shapeType = "cylinder", length = 0.1, width = 2 \* R, height = 2 \* R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape BoxShape(shapeType = "box", length = 0.2, width = R, height = R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = 0.7 \* Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

equation

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-06, Interval = 0.002));

end Wheel2D;

model FreeWheel

parameter Modelica.Units.SI.Length R = 2;

Wheel2D Wheel(R = R, Color = {0, 255, 0});

equation

Wheel.X = R / 2 \* sin(time);

Wheel.Y = R / 2 \* cos(time);

Wheel.Phi = time;

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end FreeWheel;

model Support2D

parameter Modelica.Units.SI.Length Xp = 0;

parameter Modelica.Units.SI.Length Yp = 0;

parameter Modelica.Units.SI.Length Xt = 0;

parameter Modelica.Units.SI.Length Yt = 0;

parameter Real K = 0;

parameter Real Color[3] = {0, 0, 0};

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

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape SupportShape(shapeType = "cylinder", length = 0.5, width = 0.2, height = 0.2, widthDirection = {1, 0, 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xp, Yp, -0.2}, R = orientation, r\_shape = {0, 0, 0});

// Additional for Dynamica Modeling

Modelica.Units.SI.Force Rx;

Modelica.Units.SI.Force Ry;

ForceOutput FO;

// End of Additional

KinematicInput Body\_In;

equation

Xp = Body\_In.X + Xt \* cos(Body\_In.Phi) - Yt \* sin(Body\_In.Phi);

Yp = Body\_In.Y + Xt \* sin(Body\_In.Phi) + Yt \* sin(Body\_In.Phi);

// Additional for Dynamica Modeling

FO.X = Xp;

FO.Y = Yp;

FO.Fx = Rx;

FO.Fy = Ry;

**FO.M = 0 - K \* der(Body\_In.Phi);**

// End of Additional

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Support2D;

model Joint2D

Modelica.Units.SI.Length Xsh;

Modelica.Units.SI.Length Ysh;

parameter Modelica.Units.SI.Length Xt1 = 0;

parameter Modelica.Units.SI.Length Yt1 = 0;

parameter Modelica.Units.SI.Length Xt2 = 0;

parameter Modelica.Units.SI.Length Yt2 = 0;

parameter Real Color[3] = {0, 0, 200};

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

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape SupportShape(shapeType = "cylinder", length = 0.5, width = 0.2, height = 0.2, widthDirection = {1, 0, 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xsh, Ysh, -0.2}, R = orientation, r\_shape = {0, 0, 0});

KinematicInput Body\_In1;

KinematicInput Body\_In2;

// Additional for Dynamica Modeling

Modelica.Units.SI.Force Rx;

Modelica.Units.SI.Force Ry;

ForceOutput FO1;

ForceOutput FO2;

// End of Additional

equation

Xsh = Body\_In1.X + Xt1 \* cos(Body\_In1.Phi) - Yt1 \* sin(Body\_In1.Phi);

Ysh = Body\_In1.Y + Xt1 \* sin(Body\_In1.Phi) + Yt1 \* cos(Body\_In1.Phi);

Xsh = Body\_In2.X + Xt2 \* cos(Body\_In2.Phi) - Yt2 \* sin(Body\_In2.Phi);

Ysh = Body\_In2.Y + Xt2 \* sin(Body\_In2.Phi) + Yt2 \* cos(Body\_In2.Phi);

// Additional for Dynamica Modeling

FO1.X = Xsh;

FO1.Y = Ysh;

FO1.Fx = Rx;

FO1.Fy = Ry;

FO1.M = 0;

FO2.X = Xsh;

FO2.Y = Ysh;

FO2.Fx = -Rx;

FO2.Fy = -Ry;

FO2.M = 0;

// End of Additional

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Joint2D;

model Slider2D

parameter Modelica.Units.SI.Length Xp = 0;

parameter Modelica.Units.SI.Length Yp = 0;

parameter Modelica.Units.SI.Angle Phip = 0;

parameter Modelica.Units.SI.Length Xt = 0;

parameter Modelica.Units.SI.Length Yt = 0;

Modelica.Units.SI.Length Xpol;

Modelica.Units.SI.Length Ypol;

Modelica.Units.SI.Length S;

parameter Real Color[3] = {0, 255, 255};

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

parameter Modelica.Units.SI.Length l = 0.3;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape SliderShape(shapeType = "box", length = l, width = 0.2, height = 0.2, lengthDirection = {cos(Phip), sin(Phip), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xpol - l / 2 \* cos(Phip), Ypol - l / 2 \* sin(Phip), 0}, R = orientation);

KinematicInput Body\_In;

equation

Xpol = Body\_In.X + Xt \* cos(Body\_In.Phi) - Yt \* sin(Body\_In.Phi);

Ypol = Body\_In.Y + Xt \* sin(Body\_In.Phi) + Yt \* sin(Body\_In.Phi);

Xpol = Xp + S \* cos(Phip);

Ypol = Yp + S \* sin(Phip);

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Slider2D;

model RollCircleOnBottom

parameter Modelica.Units.SI.Length Xp = 0;

parameter Modelica.Units.SI.Length Yp = 0;

parameter Modelica.Units.SI.Angle Phip = 0;

parameter Modelica.Units.SI.Length R = 1;

Modelica.Units.SI.Length Xk;

Modelica.Units.SI.Length Yk;

Modelica.Units.SI.Length S;

parameter Real Color[3] = {0, 0, 0};

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

parameter Modelica.Units.SI.Length l = 3.5 \* R;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape FlatShape(shapeType = "box", length = l, width = 0.2, height = 0.2, lengthDirection = {cos(Phip), sin(Phip), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xp - l / 2 \* cos(Phip) + 0.1 \* sin(Phip), Yp - l / 2 \* sin(Phip) - 0.1 \* cos(Phip), 0}, R = orientation);

KinematicInput Body\_In;

// Additional for Dynamica Modeling

Modelica.Units.SI.Force N;

Modelica.Units.SI.Force Ftr;

ForceOutput FO;

// End of Additional

equation

Xk = Body\_In.X + R \* sin(Phip);

Yk = Body\_In.Y - R \* cos(Phip);

Xk = Xp + S \* cos(Phip);

Yk = Yp + S \* sin(Phip);

der(S) = -der(Body\_In.Phi) \* R;

// Additional for Dynamica Modeling

FO.X = Xk;

FO.Y = Yk;

FO.Fx = (-N \* sin(Phip)) - Ftr \* cos(Phip);

FO.Fy = N \* cos(Phip) - Ftr \* sin(Phip);

FO.M = 0;

// End of Additional

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end RollCircleOnBottom;

// New Obljects for Dynamica Modeling

connector ForceInput

input Modelica.Units.SI.Length X;

input Modelica.Units.SI.Length Y;

input Modelica.Units.SI.Force Fx;

input Modelica.Units.SI.Force Fy;

input Modelica.Units.SI.MomentOfForce M;

end ForceInput;

connector ForceOutput

output Modelica.Units.SI.Length X;

output Modelica.Units.SI.Length Y;

output Modelica.Units.SI.Force Fx;

output Modelica.Units.SI.Force Fy;

output Modelica.Units.SI.MomentOfForce M;

end ForceOutput;

model TwoPortBody2D

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

parameter Real Color[3] = {0, 0, 255};

parameter Modelica.Units.SI.Acceleration g = 9.81;

parameter Modelica.Units.SI.Mass m = 1;

Modelica.Units.SI.Length X;

Modelica.Units.SI.Length Y;

Modelica.Units.SI.Angle Phi;

Modelica.Units.SI.Velocity Vx;

Modelica.Units.SI.Velocity Vy;

Modelica.Units.SI.AngularVelocity Omega;

Modelica.Units.SI.Length CA[2];

Modelica.Units.SI.Length CB[2];

KinematicOutput Body\_Out;

ForceInput F\_A;

ForceInput F\_B;

equation

Body\_Out.X = X;

Body\_Out.Y = Y;

Body\_Out.Phi = Phi;

der(X) = Vx;

der(Y) = Vy;

der(Phi) = Omega;

CA = {F\_A.X - X, F\_A.Y - Y};

CB = {F\_B.X - X, F\_B.Y - Y};

m \* der(Vx) = F\_A.Fx + F\_B.Fx;

m \* der(Vy) = F\_A.Fy + F\_B.Fy - m \* g;

m \* der(Omega) = CA[1] \* F\_A.Fy - CA[2] \* F\_A.Fx + F\_A.M + CB[1] \* F\_B.Fy - CB[2] \* F\_B.Fx + F\_B.M;

end TwoPortBody2D;

model TwoPortRod2D

extends Fejzullin\_Lab\_5\_6.Controsha.TwoPortBody2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

parameter Modelica.Units.SI.MomentOfInertia J = m \* L ^ 2 / 12;

equation

end TwoPortRod2D;

model TwoPortWheel2D

extends Fejzullin\_Lab\_5\_6.Controsha.TwoPortBody2D;

parameter Modelica.Units.SI.Length R = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape WheelShape(shapeType = "cylinder", length = 0.1, width = 2 \* R, height = 2 \* R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape BoxShape(shapeType = "box", length = 0.2, width = R, height = R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = 0.7 \* Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

parameter Modelica.Units.SI.MomentOfInertia J = m \* R ^ 2 / 2;

equation

end TwoPortWheel2D;

model ThreePortBody2D

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

parameter Real Color[3] = {0, 0, 255};

parameter Modelica.Units.SI.Acceleration g = 9.81;

parameter Modelica.Units.SI.Mass m = 1;

Modelica.Units.SI.Length X;

Modelica.Units.SI.Length Y;

Modelica.Units.SI.Angle Phi;

Modelica.Units.SI.Velocity Vx;

Modelica.Units.SI.Velocity Vy;

Modelica.Units.SI.AngularVelocity Omega;

Modelica.Units.SI.Length CA[2];

Modelica.Units.SI.Length CB[2];

Modelica.Units.SI.Length CC[2];

KinematicOutput Body\_Out;

ForceInput F\_A;

ForceInput F\_B;

ForceInput F\_C;

equation

Body\_Out.X = X;

Body\_Out.Y = Y;

Body\_Out.Phi = Phi;

der(X) = Vx;

der(Y) = Vy;

der(Phi) = Omega;

CA = {F\_A.X - X, F\_A.Y - Y};

CB = {F\_B.X - X, F\_B.Y - Y};

CC = {F\_C.X - X, F\_C.Y - Y};

m \* der(Vx) = F\_A.Fx + F\_B.Fx + F\_C.Fx;

m \* der(Vy) = F\_A.Fy + F\_B.Fy + F\_C.Fy - m \* g;

m \* der(Omega) = CA[1] \* F\_A.Fy - CA[2] \* F\_A.Fx + F\_A.M + CB[1] \* F\_B.Fy - CB[2] \* F\_B.Fx + F\_B.M + CC[1] \* F\_C.Fy - CC[2] \* F\_C.Fx + F\_C.M;

end ThreePortBody2D;

model ThreePortWheel2D

extends Fejzullin\_Lab\_5\_6.Controsha.ThreePortBody2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

parameter Modelica.Units.SI.MomentOfInertia J = m \* L ^ 2 / 12;

equation

end ThreePortWheel2D;

model ThreePortRod2D

extends Fejzullin\_Lab\_5\_6.Controsha.ThreePortBody2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

parameter Modelica.Units.SI.MomentOfInertia J = m \* L ^ 2 / 12;

equation

end ThreePortRod2D;

// End of New Objects

end Controsha;

Графики:

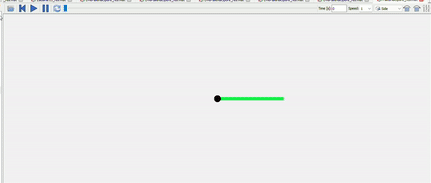


Рисунок 1. Работа затухания

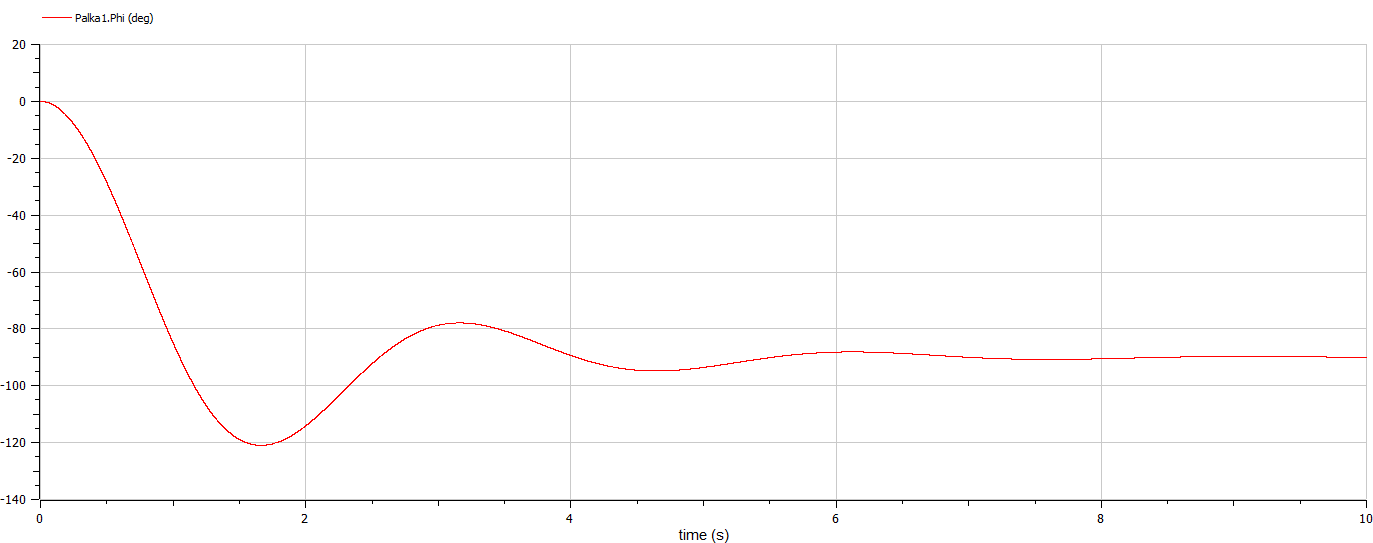


Рисунок 2. Зависимость угла от времени

**Задание 2**

По заданной схеме смоделировать динамику системы плоских.

**Вариант 1**



**Результат работы**

Код программы:

package Controsha2

model FreeEnd

ForceOutput FO;

equation

FO.X = 0;

FO.Y = 0;

FO.Fx = 0;

FO.Fy = 0;

FO.M = 0;

end FreeEnd;

model KolesoSOporoi

parameter Modelica.Units.SI.Length R1 = 2;

parameter Modelica.Units.SI.Length R2 = 1;

parameter Modelica.Units.SI.AngularVelocity Omega\_0 = 2;

parameter Modelica.Units.SI.AngularVelocity Omega\_1 = 2;

TwoPortWheel2D Koleso1(R = R1, Color = {0, 255, 0}, m=2, Omega(start = Omega\_0));

TwoPortWheel2D Koleso2(R = R2, Color = {0, 255, 0}, m=2, Omega(start = -Omega\_1));

Support2D Opora1(Xp = 0, Yp = 0, Xt = 0 , Yt = 0);

Support2D Opora2(Xp = -R1 - R2, Yp = 0, Xt = 0 , Yt = 0);

FreeEnd Hvost1();

FreeEnd Hvost2();

equation

connect(Koleso1.Body\_Out, Opora1.Body\_In);

connect(Koleso1.F\_A, Opora1.FO);

connect(Koleso2.Body\_Out, Opora2.Body\_In);

//connect(Koleso2.F\_B, Koleso1.F\_B);

connect(Koleso2.F\_B, Opora2.FO);

connect(Koleso1.F\_B, Hvost1.FO);

connect(Koleso2.F\_A, Hvost2.FO);

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end KolesoSOporoi;

model PalkaNaOpore

parameter Modelica.Units.SI.Length L1 = 2;

parameter Modelica.Units.SI.Length L2 = 4;

TwoPortRod2D Palka1(L = L1, Color = {0, 255, 80}, m = 2);

//TwoPortRod2D Palka2(L = L2, Color = {255, 255, 0}, m=4);

Support2D Opora(Xp = 0, Yp = 0, Xt = -L1 / 2, Yt = 0, K = 5);

//Joint2D Sharnir(Xt1 = L1 / 2, Yt1 = 0, Xt2 = -L1 / 2, Yt2 = 0);

FreeEnd Hvost;

equation

connect(Palka1.Body\_Out, Opora.Body\_In);

// connect(Palka1.Body\_Out, Sharnir.Body\_In1);

//connect(Palka2.Body\_Out, Sharnir.Body\_In2);

connect(Palka1.F\_A, Opora.FO);

// connect(Palka1.F\_B, Sharnir.FO1);

// connect(Palka2.F\_A, Sharnir.FO2);

connect(Palka1.F\_B, Hvost.FO);

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end PalkaNaOpore;

model Body2D

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

parameter Real Color[3] = {0, 0, 255};

Modelica.Units.SI.Length X;

Modelica.Units.SI.Length Y;

Modelica.Units.SI.Angle Phi;

KinematicOutput Body\_Out;

equation

Body\_Out.X = X;

Body\_Out.Y = Y;

Body\_Out.Phi = Phi;

end Body2D;

connector KinematicInput

input Modelica.Units.SI.Length X;

input Modelica.Units.SI.Length Y;

input Modelica.Units.SI.Angle Phi;

end KinematicInput;

connector KinematicOutput

output Modelica.Units.SI.Length X;

output Modelica.Units.SI.Length Y;

output Modelica.Units.SI.Angle Phi;

end KinematicOutput;

model Rod2D

extends Fejzullin\_Lab\_5\_6.Controsha.Body2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

equation

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Rod2D;

model FreeRod

parameter Modelica.Units.SI.Length L = 2;

Rod2D Palka(L = L, Color = {0, 255, 80});

equation

Palka.X = L / 2 \* sin(time);

Palka.Y = L / 2 \* cos(time);

Palka.Phi = time;

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end FreeRod;

model Wheel2D

extends Fejzullin\_Lab\_5\_6.Controsha.Body2D;

parameter Modelica.Units.SI.Length R = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape WheelShape(shapeType = "cylinder", length = 0.1, width = 2 \* R, height = 2 \* R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape BoxShape(shapeType = "box", length = 0.2, width = R, height = R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = 0.7 \* Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

equation

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-06, Interval = 0.002));

end Wheel2D;

model FreeWheel

parameter Modelica.Units.SI.Length R = 2;

Wheel2D Wheel(R = R, Color = {0, 255, 0});

equation

Wheel.X = R / 2 \* sin(time);

Wheel.Y = R / 2 \* cos(time);

Wheel.Phi = time;

annotation(

experiment(StartTime = 0, StopTime = 10, Tolerance = 1e-06, Interval = 0.0002));

end FreeWheel;

model Support2D

parameter Modelica.Units.SI.Length Xp = 0;

parameter Modelica.Units.SI.Length Yp = 0;

parameter Modelica.Units.SI.Length Xt = 0;

parameter Modelica.Units.SI.Length Yt = 0;

parameter Real K = 0;

parameter Real Color[3] = {0, 0, 0};

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

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape SupportShape(shapeType = "cylinder", length = 0.5, width = 0.2, height = 0.2, widthDirection = {1, 0, 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xp, Yp, -0.2}, R = orientation, r\_shape = {0, 0, 0});

// Additional for Dynamica Modeling

Modelica.Units.SI.Force Rx;

Modelica.Units.SI.Force Ry;

ForceOutput FO;

// End of Additional

KinematicInput Body\_In;

equation

Xp = Body\_In.X + Xt \* cos(Body\_In.Phi) - Yt \* sin(Body\_In.Phi);

Yp = Body\_In.Y + Xt \* sin(Body\_In.Phi) + Yt \* sin(Body\_In.Phi);

// Additional for Dynamica Modeling

FO.X = Xp;

FO.Y = Yp;

FO.Fx = Rx;

FO.Fy = Ry;

FO.M = 0 - K \* der(Body\_In.Phi);

// End of Additional

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Support2D;

model Joint2D

Modelica.Units.SI.Length Xsh;

Modelica.Units.SI.Length Ysh;

parameter Modelica.Units.SI.Length Xt1 = 0;

parameter Modelica.Units.SI.Length Yt1 = 0;

parameter Modelica.Units.SI.Length Xt2 = 0;

parameter Modelica.Units.SI.Length Yt2 = 0;

parameter Real Color[3] = {0, 0, 200};

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

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape SupportShape(shapeType = "cylinder", length = 0.5, width = 0.2, height = 0.2, widthDirection = {1, 0, 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xsh, Ysh, -0.2}, R = orientation, r\_shape = {0, 0, 0});

KinematicInput Body\_In1;

KinematicInput Body\_In2;

// Additional for Dynamica Modeling

Modelica.Units.SI.Force Rx;

Modelica.Units.SI.Force Ry;

ForceOutput FO1;

ForceOutput FO2;

// End of Additional

equation

Xsh = Body\_In1.X + Xt1 \* cos(Body\_In1.Phi) - Yt1 \* sin(Body\_In1.Phi);

Ysh = Body\_In1.Y + Xt1 \* sin(Body\_In1.Phi) + Yt1 \* cos(Body\_In1.Phi);

Xsh = Body\_In2.X + Xt2 \* cos(Body\_In2.Phi) - Yt2 \* sin(Body\_In2.Phi);

Ysh = Body\_In2.Y + Xt2 \* sin(Body\_In2.Phi) + Yt2 \* cos(Body\_In2.Phi);

// Additional for Dynamica Modeling

FO1.X = Xsh;

FO1.Y = Ysh;

FO1.Fx = Rx;

FO1.Fy = Ry;

FO1.M = 0;

FO2.X = Xsh;

FO2.Y = Ysh;

FO2.Fx = -Rx;

FO2.Fy = -Ry;

FO2.M = 0;

// End of Additional

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Joint2D;

model Slider2D

parameter Modelica.Units.SI.Length Xp = 0;

parameter Modelica.Units.SI.Length Yp = 0;

parameter Modelica.Units.SI.Angle Phip = 0;

parameter Modelica.Units.SI.Length Xt = 0;

parameter Modelica.Units.SI.Length Yt = 0;

Modelica.Units.SI.Length Xpol;

Modelica.Units.SI.Length Ypol;

Modelica.Units.SI.Length S;

parameter Real Color[3] = {0, 255, 255};

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

parameter Modelica.Units.SI.Length l = 0.3;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape SliderShape(shapeType = "box", length = l, width = 0.2, height = 0.2, lengthDirection = {cos(Phip), sin(Phip), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xpol - l / 2 \* cos(Phip), Ypol - l / 2 \* sin(Phip), 0}, R = orientation);

KinematicInput Body\_In;

equation

Xpol = Body\_In.X + Xt \* cos(Body\_In.Phi) - Yt \* sin(Body\_In.Phi);

Ypol = Body\_In.Y + Xt \* sin(Body\_In.Phi) + Yt \* sin(Body\_In.Phi);

Xpol = Xp + S \* cos(Phip);

Ypol = Yp + S \* sin(Phip);

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end Slider2D;

model RollCircleOnBottom

parameter Modelica.Units.SI.Length Xp = 0;

parameter Modelica.Units.SI.Length Yp = 0;

parameter Modelica.Units.SI.Angle Phip = 0;

parameter Modelica.Units.SI.Length R = 1;

Modelica.Units.SI.Length Xk;

Modelica.Units.SI.Length Yk;

Modelica.Units.SI.Length S;

parameter Real Color[3] = {0, 0, 0};

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

parameter Modelica.Units.SI.Length l = 3.5 \* R;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape FlatShape(shapeType = "box", length = l, width = 0.2, height = 0.2, lengthDirection = {cos(Phip), sin(Phip), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {Xp - l / 2 \* cos(Phip) + 0.1 \* sin(Phip), Yp - l / 2 \* sin(Phip) - 0.1 \* cos(Phip), 0}, R = orientation);

KinematicInput Body\_In;

// Additional for Dynamica Modeling

Modelica.Units.SI.Force N;

Modelica.Units.SI.Force Ftr;

ForceOutput FO;

// End of Additional

equation

Xk = Body\_In.X + R \* sin(Phip);

Yk = Body\_In.Y - R \* cos(Phip);

Xk = Xp + S \* cos(Phip);

Yk = Yp + S \* sin(Phip);

der(S) = -der(Body\_In.Phi) \* R;

// Additional for Dynamica Modeling

FO.X = Xk;

FO.Y = Yk;

FO.Fx = (-N \* sin(Phip)) - Ftr \* cos(Phip);

FO.Fy = N \* cos(Phip) - Ftr \* sin(Phip);

FO.M = 0;

// End of Additional

annotation(

experiment(StartTime = 0, StopTime = 1, Tolerance = 1e-6, Interval = 0.002));

end RollCircleOnBottom;

// New Obljects for Dynamica Modeling

connector ForceInput

input Modelica.Units.SI.Length X;

input Modelica.Units.SI.Length Y;

input Modelica.Units.SI.Force Fx;

input Modelica.Units.SI.Force Fy;

input Modelica.Units.SI.MomentOfForce M;

end ForceInput;

connector ForceOutput

output Modelica.Units.SI.Length X;

output Modelica.Units.SI.Length Y;

output Modelica.Units.SI.Force Fx;

output Modelica.Units.SI.Force Fy;

output Modelica.Units.SI.MomentOfForce M;

end ForceOutput;

model TwoPortBody2D

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

parameter Real Color[3] = {0, 0, 255};

parameter Modelica.Units.SI.Acceleration g = 9.81;

parameter Modelica.Units.SI.Mass m = 1;

Modelica.Units.SI.Length X;

Modelica.Units.SI.Length Y;

Modelica.Units.SI.Angle Phi;

Modelica.Units.SI.Velocity Vx;

Modelica.Units.SI.Velocity Vy;

Modelica.Units.SI.AngularVelocity Omega;

Modelica.Units.SI.Length CA[2];

Modelica.Units.SI.Length CB[2];

KinematicOutput Body\_Out;

ForceInput F\_A;

ForceInput F\_B;

equation

Body\_Out.X = X;

Body\_Out.Y = Y;

Body\_Out.Phi = Phi;

der(X) = Vx;

der(Y) = Vy;

der(Phi) = Omega;

CA = {F\_A.X - X, F\_A.Y - Y};

CB = {F\_B.X - X, F\_B.Y - Y};

m \* der(Vx) = F\_A.Fx + F\_B.Fx;

m \* der(Vy) = F\_A.Fy + F\_B.Fy - m \* g;

m \* der(Omega) = CA[1] \* F\_A.Fy - CA[2] \* F\_A.Fx + F\_A.M + CB[1] \* F\_B.Fy - CB[2] \* F\_B.Fx + F\_B.M;

end TwoPortBody2D;

model TwoPortRod2D

extends Fejzullin\_Lab\_5\_6.Controsha.TwoPortBody2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

parameter Modelica.Units.SI.MomentOfInertia J = m \* L ^ 2 / 12;

equation

end TwoPortRod2D;

model TwoPortWheel2D

extends Fejzullin\_Lab\_5\_6.Controsha2.TwoPortBody2D;

parameter Modelica.Units.SI.Length R = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape WheelShape(shapeType = "cylinder", length = 0.1, width = 2 \* R, height = 2 \* R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape BoxShape(shapeType = "box", length = 0.2, width = R, height = R, widthDirection = {cos(Phi), sin(Phi), 0}, lengthDirection = {0, 0, 1}, color = 0.7 \* Color, specularCoefficient = 0.5, r = {X, Y, 0}, R = orientation, r\_shape = {0, 0, 0});

parameter Modelica.Units.SI.MomentOfInertia J = m \* R ^ 2 / 2;

equation

end TwoPortWheel2D;

model ThreePortBody2D

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

parameter Real Color[3] = {0, 0, 255};

parameter Modelica.Units.SI.Acceleration g = 9.81;

parameter Modelica.Units.SI.Mass m = 1;

Modelica.Units.SI.Length X;

Modelica.Units.SI.Length Y;

Modelica.Units.SI.Angle Phi;

Modelica.Units.SI.Velocity Vx;

Modelica.Units.SI.Velocity Vy;

Modelica.Units.SI.AngularVelocity Omega;

Modelica.Units.SI.Length CA[2];

Modelica.Units.SI.Length CB[2];

Modelica.Units.SI.Length CC[2];

KinematicOutput Body\_Out;

ForceInput F\_A;

ForceInput F\_B;

ForceInput F\_C;

equation

Body\_Out.X = X;

Body\_Out.Y = Y;

Body\_Out.Phi = Phi;

der(X) = Vx;

der(Y) = Vy;

der(Phi) = Omega;

CA = {F\_A.X - X, F\_A.Y - Y};

CB = {F\_B.X - X, F\_B.Y - Y};

CC = {F\_C.X - X, F\_C.Y - Y};

m \* der(Vx) = F\_A.Fx + F\_B.Fx + F\_C.Fx;

m \* der(Vy) = F\_A.Fy + F\_B.Fy + F\_C.Fy - m \* g;

m \* der(Omega) = CA[1] \* F\_A.Fy - CA[2] \* F\_A.Fx + F\_A.M + CB[1] \* F\_B.Fy - CB[2] \* F\_B.Fx + F\_B.M + CC[1] \* F\_C.Fy - CC[2] \* F\_C.Fx + F\_C.M;

end ThreePortBody2D;

model ThreePortWheel2D

extends Fejzullin\_Lab\_5\_6.Controsha2.ThreePortBody2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

parameter Modelica.Units.SI.MomentOfInertia J = m \* L ^ 2 / 12;

equation

end ThreePortWheel2D;

model ThreePortRod2D

extends Fejzullin\_Lab\_5\_6.Controsha.ThreePortBody2D;

parameter Modelica.Units.SI.Length L = 1;

Modelica.Mechanics.MultiBody.Visualizers.Advanced.Shape RodShape(shapeType = "box", length = L, width = 0.1, height = 0.1, lengthDirection = {cos(Phi), sin(Phi), 0}, widthDirection = {0, 0, 1}, color = Color, specularCoefficient = 0.5, r = {X - L / 2 \* cos(Phi), Y - L / 2 \* sin(Phi), 0}, R = orientation);

parameter Modelica.Units.SI.MomentOfInertia J = m \* L ^ 2 / 12;

equation

end ThreePortRod2D;

// End of New Objects

end Controsha2;

Графики:

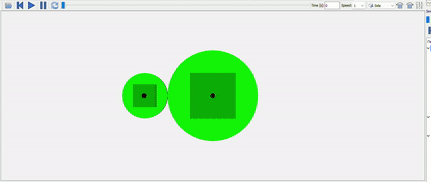


Рисунок 3. Совместная работа двух окружностей

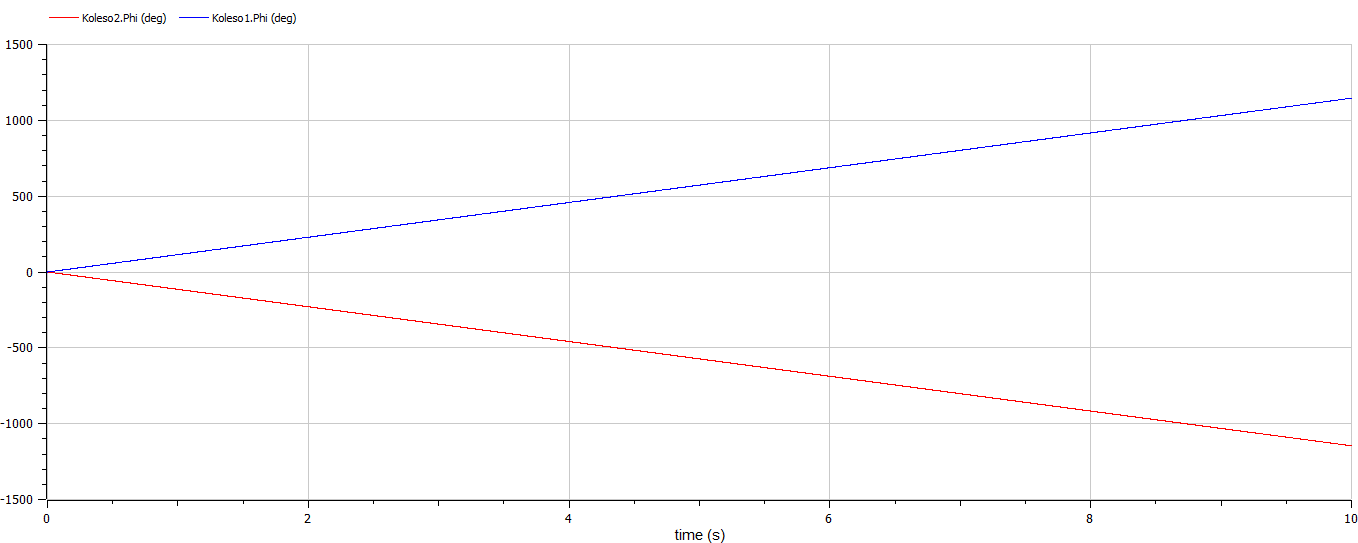


Рисунок 2. Зависимость угла от времени