МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)

Кафедра программного обеспечения вычислительной техники и
автоматизированных систем

РГЗ

по дисциплине: Системное моделирование

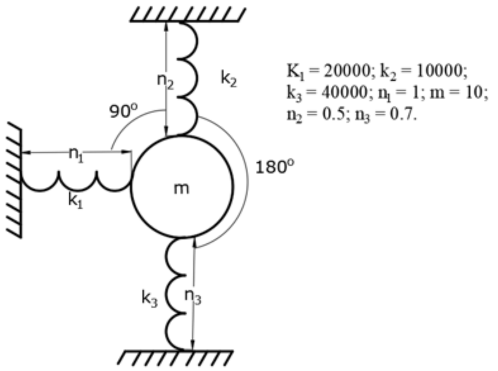
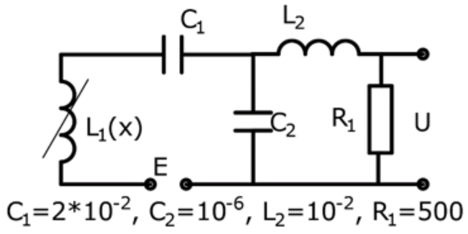
тема: Математическое моделирование работы электронно-механической
измерительной системы

Выполнил: ст. группы ПВ-223
Игнатьев Артур Олегович
Проверил:
Полунин Александр Иванович

Белгород 2024 г.

Цель работы:
1. Формулировка задачи.
2. Математическая постановка задачи: вывод необходимых формул, выбор и запись расчётных методов и алгоритмов.
3. Блок-схема программы.
4. Результаты расчётов — графики.

Вариант 3

3		

Выполнение работы:

Для механической системы:

Уравнение Лагранжа второго рода имеет вид:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\varphi}_j}\right) - \frac{\partial T}{\partial \varphi_j} = Q_j$$

Для данной системы система будет выглядеть как

$$\begin{cases} \dfrac{d}{dt}\left(\dfrac{\partial T}{\partial y`}\right) - \dfrac{\partial T}{\partial y} = -\dfrac{\partial \Pi}{\partial y} \\ \dfrac{d}{dt}\left(\dfrac{\partial T}{\partial x`}\right) - \dfrac{\partial T}{\partial x} = -\dfrac{\partial \Pi}{\partial x} \end{cases}$$

Найдём кинетическую энергию:

$$T = T_1 + T_2$$

$$T_1 = \frac{m(x`)^2}{2}$$

$$T_1 = \frac{m(y`)^2}{2}$$

$$T = \frac{m(x`)^2}{2} + \frac{m(y`)^2}{2}$$

Найдём потенциальную энергию

$$\Pi = \Pi_1 + \Pi_2 + \Pi_3 + \Pi_g$$

$$\Pi_g = mgy$$

$$\Pi_1 = \frac{k_1 x^2}{2}$$

$$\Pi_2 = \frac{k_2 y^2}{2}$$

$$\Pi_3 = \frac{k_3 y^2}{2}$$

$$\Pi = mgy + \frac{k_1 x^2}{2} + \frac{k_2 y^2}{2} + \frac{k_3 y^2}{2},$$

Далее найдём необходимые производные для составления уравнения:

Для кинетической энергии:

$$\frac{\partial T}{\partial y`} = \frac{\partial}{\partial y`}\left(\frac{m(y`)^2}{2} + \frac{m(x`)^2}{2}\right) = my`$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial y`}\right) = \frac{d}{dt}(my`) = my``$$

$$\frac{\partial T}{\partial y} = 0$$

Аналогично,

$$\frac{\partial T}{\partial x`} = \frac{\partial}{\partial x`}\left(\frac{m(y`)^2}{2} + \frac{m(x`)^2}{2}\right) = mx`$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial x`}\right) = \frac{d}{dt}(mx`) = mx``$$

$$\frac{\partial T}{\partial x} = 0$$

Для потенциальной энергии:

$$\frac{\partial \Pi}{\partial x} = k_1 x$$

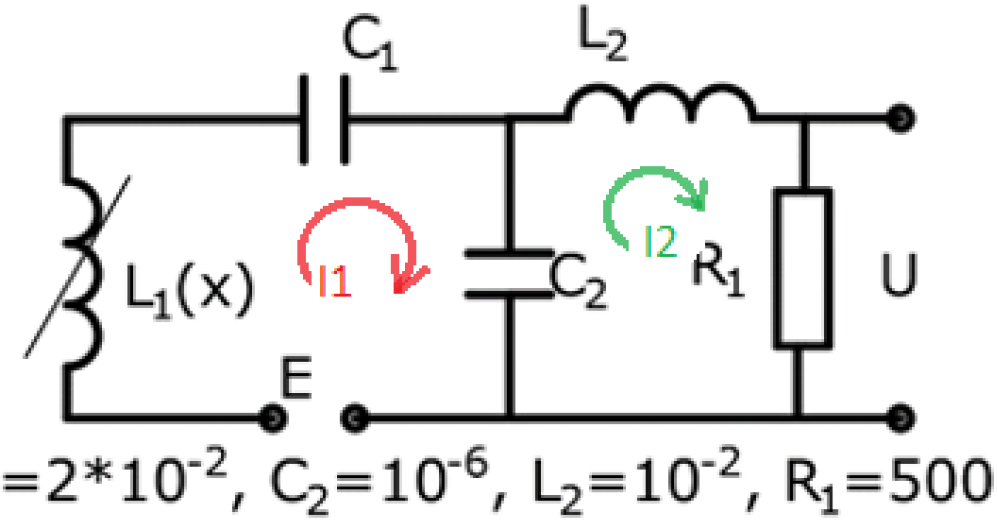
$$\frac{\partial \Pi}{\partial y} = mg + k_2 y + k_3 y$$

Получим систему:

$$\begin{cases} x`` = \dfrac{-k_1 x}{m} \\ y`` = -\dfrac{mg + k_2 y + k_3 y}{m} \end{cases}$$

$$\begin{cases} \dfrac{dV_x}{dt} = \dfrac{-k_1 x}{m} \\ \dfrac{dx}{dt} = V_x \\ \dfrac{dV_y}{dt} = -\dfrac{mg + k_2 y + k_3 y}{m} \\ \dfrac{dy}{dt} = V_y \end{cases}$$

Для электрической системы:



Составим Уравнения согласно второму закону Кирхгофа:

$$\begin{cases} \dfrac{dL_1}{dx}\dfrac{dx}{dt}(I_0 + I_1) + L_1(x)\dfrac{dI_1}{dt} + \dfrac{1}{C_1}\displaystyle\int_0^t (I_0 + I_1)dt + \dfrac{1}{C_2}\int_0^t (I_0 + I_1)dt - \dfrac{1}{C_2}\int_0^t (I_0 + I_2)dt = E \\ L_2\dfrac{dI_2}{dt} + (I_0 + I_2)R_1 + \dfrac{1}{C_2}\displaystyle\int_0^t (I_0 + I_2)dt - \dfrac{1}{C_2}\int_0^t (I_0 + I_1)dt = 0 \end{cases}$$

$$L = L_0 - \mu|x|$$

$$I_0 = \frac{E}{R_1}$$

Подставим: $Q_1 = \int_0^t I_1 dt, \quad Q_2 = \int_0^t I_2 dt,$

$$\begin{cases} \dfrac{dI_1}{dt} = \dfrac{E - \dfrac{Q_1}{C_1} - \dfrac{Q_1}{C_2} + \dfrac{Q_2}{C_2} - \dfrac{dL_1}{dx} a(I_0 + I_1)}{L_1(x)} \\ \dfrac{dL_1(x)}{dt} = \begin{cases} \mu, x < 0; \\ -\mu, x \geq 0; \end{cases} \\ \dfrac{dI_2}{dt} = \dfrac{-(I_0 + I_2)R_1 - \dfrac{Q_2}{C_2} + \dfrac{Q_1}{C_2}}{L_2} \\ \dfrac{dQ_1}{dt} = I_1 + I_0 \\ \dfrac{dQ_2}{dt} = I_2 + I_0 \end{cases}$$

Код программы:

```
import math
from scipy.integrate import odeint
import numpy as np
import matplotlib.pyplot as plt

# MS
k1 = 20000
k2 = 10000
k3 = 40000
n1 = 1
n2 = 0.5
n3 = 0.7
m = 10
g = 9.81
# ES
C1 = 2e-2
C2 = 1e-6
L2 = 1e-2
R1 = 500
E = 1
mu = 1
```

```
L0 = 5e-2
coef = 0.1


def models(solve, time):
    global k1
    global k2
    global k3
    global m
    global n1
    global n2
    global n3
    global g
    V1, y, V2, x = solve[:4]

    V2_ = -(k1 * x) / m
    x_ = V2
    V1_ = -(m*g + k2 * y + k3 * y) / m
    y_ = V1

    global C1
    global C2
    global L2
    global R1
    global E
    global mu
    global L0
    global coef
    I1, I2, Q1, Q2 = solve[4:]
    I0 = E / R1
    mu_ = mu
    if (x >= 0):
        mu_ = -mu


    def L1(x):
        if (abs(x) <= coef):
            return L0 - mu * abs(x)
        else:
            return R1 * 2
    I1_ = (E - Q1 / C1 - Q1 / C2 + Q2 / C2 - mu_ * (V1_ + V2_) * (I0
+ I1)) / L1(x)
```

```
    I2_ = (Q1 / C2 - Q2 / C2 - R1 * (I0 + I2)) / L2
    Q1_ = I0 + I1
    Q2_ = I0 + I2

    return [V1_, y_ , V2_, x_, I1_, I2_, Q1_, Q2_]


if __name__ == '__main__':
    time_ = 1.75
    t = np.linspace(0, time_, 1000)
    solve0 = [0, 0.03, 0, 0.03, 0, 0, 0, 0]  # Vy, y, Vx, x, I1, I2, 
Q1, Q2
    solve = odeint(models, solve0, t)
    Vy = solve[:, 0]
    y = solve[:, 1]
    Vx = solve[:, 2]
    x = solve[:, 3]
    I1 = solve[:, 4]
    I2 = solve[:, 5]
    Q1 = solve[:, 6]
    Q2 = solve[:, 7]
    U_in_R = np.array([I2[_] * R1 for _ in range(len(I2))])
    fig, axes = plt.subplots(2,1)
    line_width = 2
    color1 = '#F27E28'  # orange
    color2 = '#53D11C'  # green
    color3 = '#9F28F2'  # purple
    f1 = axes[0]
    f1.plot(t, y, label='y(time)', color=color1, 
linewidth=line_width)
    f1.plot(t, x, label='x(time)', color=color2, 
linewidth=line_width)
    # f1.plot(t, I2, label='I2(time)', color=color3, 
linewidth=line_width)
    f1.set_xlabel('time')
    f1.set_ylabel('м')
    f1.set_title('График смещения x и y', fontsize=8)
    f1.legend()
    f1.grid()
    f2 = axes[1]
    f2.plot(t, U_in_R, label='U(time)', color=color3, 
linewidth=line_width)
```

```
    f2.set_xlabel('time')
    f2.set_ylabel('B')
    f2.legend()
    f2.grid()
    plt.show()
```

Результат работы программы: