

Пояснительная записка

Тема:

Автоматизированная стреляющая установка

Краткая аннотация:

Проект представляет собой автоматизированную катапульту, управляемую с помощью платы Arduino, с тремя сервоприводами для регулировки угла наклона, натяжения и спускового механизма. Конструкция выполнена из дерева, а для запуска используются канцелярские резинки, обеспечивающие натяжение. Катапульта дополнительно оснащена датчиком расстояния (до 4 м), который позволяет устройству автоматически прицеливаться в заданную цель, рассчитывая оптимальный угол наклона. Программное обеспечение на Arduino координирует работу всех компонентов, обеспечивая точное управление, автоматизацию процессов и повторяемость действий. Этот проект не только демонстрирует основы робототехники, программирования и механического дизайна, но и служит отличной учебной моделью для изучения физических законов в практике.

Основные законы физики:

1. Кинематика движения тела

Описывает движение снаряда, учитывая начальную скорость, угол запуска, время полёта и гравитацию.

2. Сила упругости (закон Гука)

Натяжение резинок подчиняется закону Гука, где сила упругости пропорциональна деформации.

3. Закон сохранения энергии

Потенциальная энергия резинок преобразуется в кинетическую энергию снаряда.

4. Принципы баллистики

Рассматривают дальность полёта, углы наклона и влияние гравитации на траекторию.

5. Момент силы и рычажный принцип

Работа рычага катапульты описывается моментом силы, который определяется длиной плеча и приложенной силой.

Вывод формул и уравнений:

1. Траектория движения снаряда:

$$x = v_0 \cdot \cos(\theta) \cdot t$$
$$y = v_0 \cdot \sin(\theta) \cdot t - \frac{1}{2} g t^2$$

где x, y — координаты снаряда, v_0 — начальная скорость, θ — угол наклона, g — ускорение свободного падения, t — время полёта.

2. Дальность полёта снаряда:

$$R = \frac{v_0^2 \sin(2\theta)}{g}$$

где:

- R — заданная дальность полёта,
- v_0 — начальная скорость снаряда,
- θ — угол вылета,
- g — ускорение свободного падения (9.81 м/с^2),

3. Сила упругости (закон Гука):

$$F = k \cdot x$$

где F — сила упругости, k — коэффициент упругости резинок ($\approx 135 \text{ Н/м}$), x — растяжение резинок.

4. Потенциальная энергия резинок:

$$E_{\text{п}} = \frac{1}{2} kx^2$$

5. Кинетическая энергия снаряда:

$$E_{\text{к}} = \frac{1}{2} m v_0^2$$

6. Сопротивление воздуха зависит от скорости v :

$$F_d = \frac{1}{2} \rho C_d A v^2,$$

Дано:

1. Масса деревянного снаряда — 1.58 г.
2. Длина вертикального рычага — 25 см.
3. Расстояние от оси вращения до точки крепления резинки — 9.5 см.
4. Длина от основания до крышечки со снарядом — 23 см.
5. Длина блокиратора (мы устанавливаем его угол) — 19 см.

6. Расстояние между основанием и карандашом (об который ударяется рычаг) — 16 см.
7. Длина резинки в состоянии покоя — 9 см, в натянутом состоянии — 19 см.
8. Коэффициент упругости системы из трёх резинок — 93.1 Н/м (подвесили литр молока, померили).

Желаемая дальность полёта — переменное значение, которое определяется датчиком расстояния. Блокиратор вращается вокруг оси вращения, задаем желаемый угол его положения. Трение и потери энергии в расчётах не учитываются.

1. Формула для дальности полёта снаряда (R):

Начальная скорость определяется энергией натяжения резинки, которая преобразуется в кинетическую энергию снаряда:

$$E_{\text{натяжения}} = \frac{1}{2} kx^2, \quad E_{\text{кинетическая}} = \frac{1}{2} mv_0^2.$$

Из равенства энергий:

$$v_0 = \sqrt{\frac{kx^2}{m}}.$$

2. Уравнения движения с учетом сопротивления воздуха

Сопротивление воздуха зависит от скорости v :

$$F_d = \frac{1}{2} \rho C_d A v^2,$$

где:

- ρ : плотность воздуха (1.225 кг/м³),
- C_d : коэффициент сопротивления воздуха (например, 1.05 для кубика),
- A : площадь поперечного сечения,
- v : скорость снаряда.

3. Уравнения движения

1. Горизонтальная составляющая:

$$m \frac{dv_x}{dt} = -F_d \frac{v_x}{v},$$

где v_x : горизонтальная скорость, v : общая скорость.

2. Вертикальная составляющая:

$$m \frac{dv_y}{dt} = -mg - F_d \frac{v_y}{v},$$

где v_y : вертикальная скорость.

Эти уравнения определяют ускорения a_x и a_y :

$$a_x = -\frac{F_d}{m} \cdot \frac{v_x}{v}, \quad a_y = -g - \frac{F_d}{m} \cdot \frac{v_y}{v}.$$

4. Численное интегрирование

Для решения уравнений движения используются численные методы. На каждом шаге времени (Δt):

1. Обновляем скорости:

$$v_x \leftarrow v_x + a_x \Delta t, \quad v_y \leftarrow v_y + a_y \Delta t.$$

2. Обновляем положение снаряда:

$$x \leftarrow x + v_x \Delta t, \quad y \leftarrow y + v_y \Delta t.$$

Процесс повторяется, пока снаряд не коснётся земли ($y = 0$).

5. Нахождение угла вылета (θ):

Для достижения заданной дальности (R), нужно найти угол вылета θ , при котором $x = R$ в момент приземления. Это достигается через численный метод подбора, например, с помощью `fsolve`:

$$x_{\text{final}}(\theta) - R = 0.$$

6. Угол блокиратора (ϕ):

После нахождения угла вылета θ , угол блокиратора определяется из геометрической зависимости:

$$\phi = \arcsin\left(\frac{d_r}{L_r} \sin(\theta)\right),$$

где:

- d_r : расстояние от оси вращения до точки крепления резинки,
- L_r : длина рычага.

Код для вычисления угла и построения графика полета:

```
1. import numpy as np
2. import matplotlib.pyplot as plt
3. from scipy.optimize import fsolve
4. import math
5.
6. # Константы
7. g = 9.81 # Ускорение свободного падения (м/с^2)
8. rho = 1.225 # Плотность воздуха (кг/м^3)
```

```

9. Cd = 1.05 # Коэффициент аэродинамического сопротивления для кубика
10. A = 0.02 * 0.01 # Площадь поперечного сечения кубика (м^2)
11. m = 0.00158 # Масса снаряда (кг)
12. k = 93.1 # Коэффициент упругости (Н/м)
13. x = 0.1 # Натяжение резинки (м)
14. d_r = 0.095 # Расстояние от оси до точки крепления резинки (м)
15. L_r = 0.25 # Длина рычага (м)
16.
17. # Начальная скорость, вычисляемая из энергии натяжения резинки
18. v0 = np.sqrt(k * x**2 / m)
19.
20. # Функция силы сопротивления воздуха
21. def drag_force(v):
22.     return 0.5 * rho * Cd * A * v**2
23.
24. # Симуляция движения и возврат траектории
25. def simulate_motion_with_trajectory(v0, theta, dt=0.001, max_time=10):
26.     vx = v0 * np.cos(theta)
27.     vy = v0 * np.sin(theta)
28.     x, y = 0, 0
29.
30.     x_positions, y_positions = [x], [y]
31.
32.     while y >= 0:
33.         v = np.sqrt(vx**2 + vy**2)
34.         Fd = drag_force(v)
35.
36.         ax = -Fd * (vx / v) / m
37.         ay = -g - (Fd * (vy / v) / m)
38.
39.         vx += ax * dt
40.         vy += ay * dt
41.
42.         x += vx * dt
43.         y += vy * dt
44.
45.         x_positions.append(x)
46.         y_positions.append(y)
47.
48.         if x > 100 or y < -100:
49.             break
50.
51.     return x_positions, y_positions
52.
53. # Поиск угла вылета для заданной дальности
54. def find_theta_for_range(R, v0, tolerance=0.001):
55.     def difference(theta):
56.         x, _ = simulate_motion_with_trajectory(v0, theta)
57.         return x[-1] - R
58.
59.     theta = fsolve(difference, np.radians(45))[0]
60.     return theta
61.
62. # Построение траектории для заданной дальности
63. def plot_trajectory_for_range(R):
64.     theta = find_theta_for_range(R, v0)
65.     x_positions, y_positions = simulate_motion_with_trajectory(v0, theta)
66.
67.     plt.figure(figsize=(10, 6))
68.     plt.plot(x_positions, y_positions, label=f"Траектория для R={R} м")
69.     plt.xlabel("Горизонтальное расстояние (м)", fontsize=12)
70.     plt.ylabel("Вертикальное расстояние (м)", fontsize=12)
71.     plt.title("Движение снаряда с учетом сопротивления воздуха", fontsize=14)
72.     plt.grid(True)
73.     plt.legend()
74.     plt.show()
75.
76.     return theta
77.

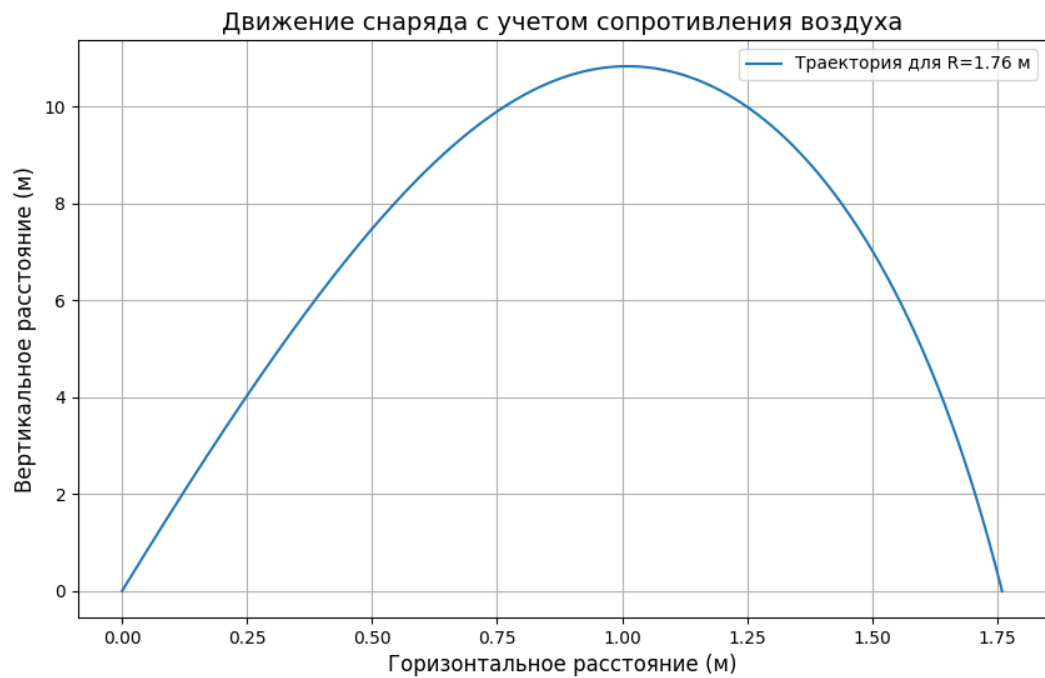
```

```

78. # Вычисление угла блокиратора
79. def calculate_phi(R):
80.     theta = find_theta_for_range(R, v0)
81.     phi = math.asin((d_r / L_r) * math.sin(theta))
82.     return math.degrees(phi)
83.
84. # Пример использования
85. R = float(input("Введите дальность: ")) # Дальность в метрах
86. theta = plot_trajectory_for_range(R)
87. phi = calculate_phi(R)
88. print(f"Угол блокиратора (phi) для R={R} м: {phi:.2f}°")
89. print(f"Угол вылета (theta) для R={R} м: {np.degrees(theta):.2f}°")

```

Так например для дальности 176 см (полученной с датчика расстояния):



Угол блокиратора (ϕ) для $R=1.76$ м: 22.29°

Угол вылета (θ) для $R=1.76$ м: 86.61°

Замеры конструкции:

