**Моделирование по электричеству: "Движение частицы в поле плоского конденсатора"**

**Петренко Людмила и Кусайкина Елизавета, М3104**

**Формулировка задачи:**

Частица влетает в поле конденсатора параллельно пластинам вдоль оси конденсатора. Под действием поля конденсатора частица отклоняется к одной из пластин и к концу пути попадает на неё. Координату, где упала частица можно измерить.

Создать модель поведения частицы с возможностью изучения её траектории в зависимости от изменяемых параметров системы.

Определить закон, по которому изменяется траектория движения частицы. Учесть влияние силы тяжести на частицу и визуализировать траекторию движения частицы в виде графика.

Входными данными являются:

1. масса частицы – m

2. заряд частицы - q

3. расстояние между пластинами конденсатора – distance (d)

4. напряжение на конденсаторе - U

5. начальная скорость частицы – v\_0 ()

6. длина конденсатора(опционально) – length (l)

7. начальная координата по вертикали – y\_enter ()

**Теория:**

На частицу в поле плоского конденатора действует сила Кулона и сила тяжести. По второму закону Ньютона:

Знак заряда определяет направление силы Кулона относительно направления поля (направления вектора напряженности). Будем считать, что если заданная разность потенциалов положительная, то верхняя обкладка будет заряжена отрицательно, а нижняя положительно. Таким образом, вектор напряженности будет сонаправлен с осью y.

Разделим выражение на массу частицу и спроецируем на оси:

Используем кинематические формулы для перемещения в проекции на оси:

– начальное положение нулевое, ускорение тоже

– начальная скорость направлена вдоль оси конденсатора, а значит равна нулю в проекции на вертикальную ось

Рассчитаем время падения частицы на одну из обкладок конденсатора:

В случае положительного ускорения координата по вертикали будет увеличиваться и, следовательно, частица упадет на верхнюю обкладку – перед d должен стоять ‘+’. В противном случае координата будет уменьшаться и d нужно взять со знаком ‘-’

Учитывая, что конденсатору можно задать конечную длину, частица не во всех случаях сможет упасть на обкладки. Если длина конденсатора меньше, чем предполагаемая горизонтальная координата падения частицы, будем считать:

– время вылета частицы из конденсатора

Опишем движение частицы после вылета из конденсатора:

Единственная сила, которая на нее действует – сила тяжести. Частица свободно подает. Ее движение можно представить как свободное падение под углом к горизонту.

– начальное положение в конце конденсатора, ускорение нулевое

Величина проекции начальной скорости не поменялась, поскольку ускорения по оси х не было.

– начальное положение в конце конденсатора.

Начальную скорость находим из уравнения*,* спроецированного на ось y, как конечную скорость при движении внутри конденсатора.

**Ввод данных:**

try:

    m = int(input()) # масса частицы

except ValueError:

    m = 9.109 \* 10\*\*(-31)

try:

    q = int(input()) # заряд частицы

except ValueError:

    q = -1.60217\*10\*\*(-19)

try:

    distance = int(input()) # расстояние между обкладками

except ValueError:

    distance = 0.5

try:

    U = int(input()) # напряжение на конденсаторе

except ValueError:

    U = 5000

try:

    v\_0 = int(input()) # начальная скорость частицы

except ValueError:

    v\_0 = 5000000

try:

    length = int(input()) # длина конденсатора

except ValueError:

    length = 10\*\*9

try:

    y\_enter = int(input()) # начальное положение относительно обкладок

    if (abs(y\_enter) > abs(distance/2)):

        y\_enter = 0

except ValueError:

    y\_enter = 0

g = 9.8

**Код программы:**

Подключаем модули:

%matplotlib inline

import numpy as np

import matplotlib.pyplot as plt

import matplotlib.patches as patches

import math

Записываем уравнения движения, полученные в блоке с теорией:

a = -g + q\*U/(m\*distance)

def x(t):

    return (v\_0)\*t

def y(t):

    return y\_enter + (a)\*(t\*t)/2

if a > 0:

    t\_end = math.sqrt((distance - 2\*y\_enter)/a)

else:

    t\_end = math.sqrt((-distance - 2\*y\_enter)/a)

t\_after\_end = np.linspace(0, 0)

if x(t\_end) > length:

    t\_end = length/v\_0

    t\_after\_end = np.linspace(0, distance\*10)

t = np.linspace(0, t\_end)

if length == 10\*\*9:

    length = x(t\_end) + distance/10

def y\_after\_end(t):

    return y(t\_end) + (a\*t\_end)\*t + (-g)\*(t\*t)/2

def x\_after\_end(t):

    return x(t\_end) + (v\_0)\*t

Нарисуем график и обкладки конденсатора:

fig = plt.figure(figsize=(15,5))

ax = plt.subplot(1,1,1)

ax.set\_xlim((-distance/10, x(t\_end) + distance\*10))

ax.set\_ylim((-distance/2 - distance\*10, distance/2 + distance\*10))

ax.minorticks\_on()

ax.grid(True, which='both')

ax.grid(which='minor', color='gray', linestyle=':')

ax.add\_patch(patches.Rectangle((0, distance/2), length, distance/10, edgecolor = 'red', facecolor = 'orange',fill = True))

ax.add\_patch(patches.Rectangle((0, -distance/2 - distance/10), length, distance/10, edgecolor = 'red', facecolor = 'orange',fill = True))

line, = ax.plot(x(t), y(t), lw=3, color = "green")

if t\_after\_end.any() != np.linspace(0, 0).any():

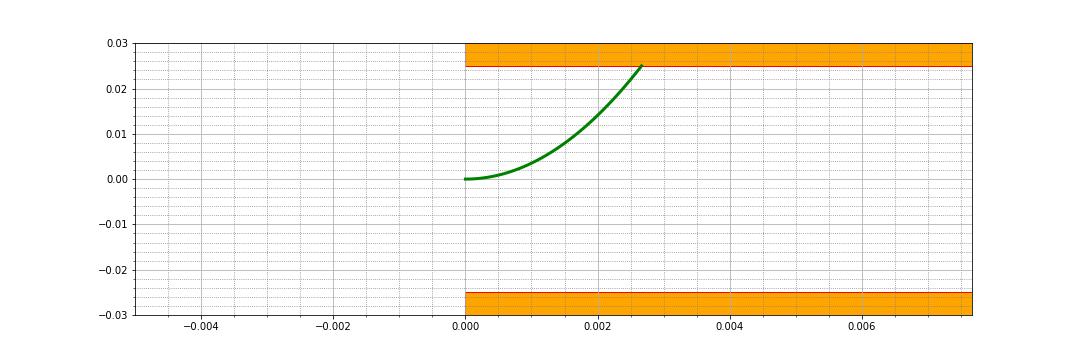
    line\_after\_end, = ax.plot(x\_after\_end(t\_after\_end), y\_after\_end(t\_after\_end), lw=3, color = "green")

**Результаты:**

Рассмотрим некоторые основные графики, которые можно получить с помощью нашего кода, изменяя параметры системы.

* Вариант, который рассматривается по умолчанию – электрон, поле в конденсаторе направлено вниз

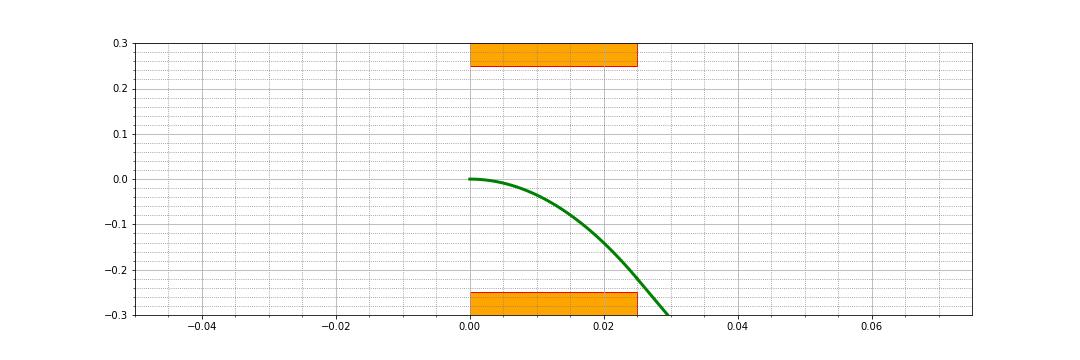
Численные параметры системы:



Поскольку частица заряжена отрицательно, она будет смещаться в сторону, противоположную направлению поля, пока не упадет на обкладку конденсатора.

* А теперь возьмем положительно заряженную частицу с таким же полем, и увеличим расстояние между обкладками.

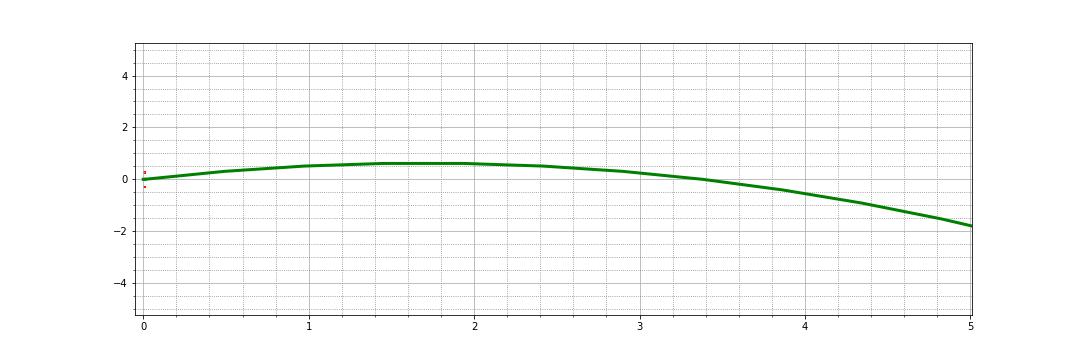
Численные параметры системы:



Кроме того, что знак заряда частицы поменял направление ее смещения, увеличение расстояния между обкладками ослабило поле, и частица уже не достигла обкладки конденсатора, а вылетела из него. После вылета из конденсатора частица продолжила свободно подать по параболе.

* Чтобы это продемонстрировать, рассмотрим случай, когда рассматриваемый объект обладает большой, по сравнению с частицей, массой.

Численные параметры системы:



На графике видно, как поле в конденсаторе разгоняет частицу, которая, вылетая из него, продолжает двигаться по параболе, свободно падая.

* Среди параметров системы также можно указать в качестве начальной координаты по вертикальной оси не центр конденсатора, т.е. сместить точку, в которой частица влетит, к одной из обкладок.

Численные параметры системы:

