**Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики**

Изображение выглядит как текст, Шрифт, белый, Графика

Автоматически созданное описание

**УЧЕБНЫЙ ЦЕНТР ОБЩЕЙ ФИЗИКИ ФТФ**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Группа** | *M3219* | | | **К работе допущен** | |  | |
| **Студент** | | *Сливкин Артём Сергеевич* | | **Работа выполнена** | | |  |
| **Преподаватель** | | | | **Отчет принят** | | | |
|  |  |  |  |  |  |  |  |

**Рабочий протокол и отчет по моделированию**

**Задание:**

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как снимок экрана, текст, линия, дизайн

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

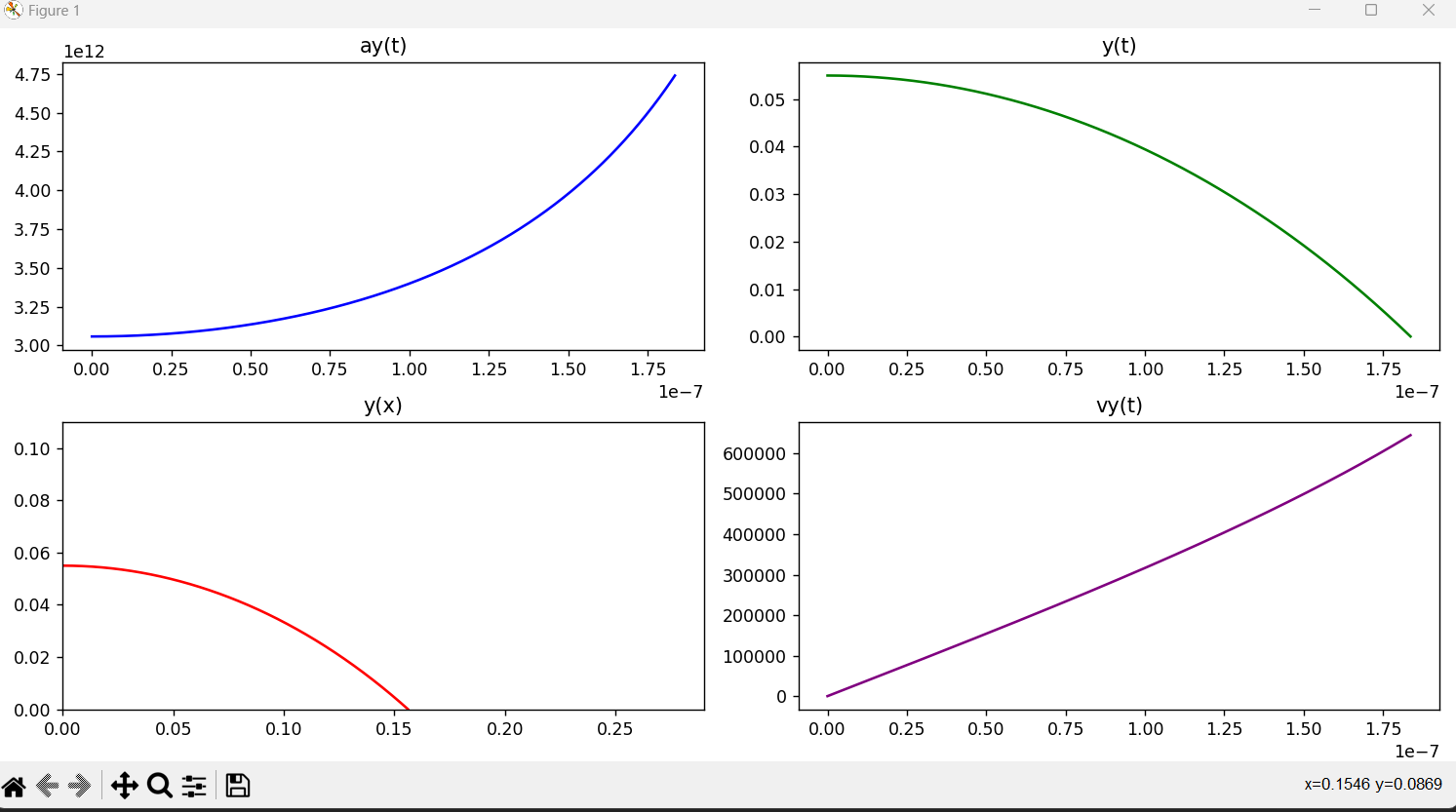
Изображение выглядит как текст, число

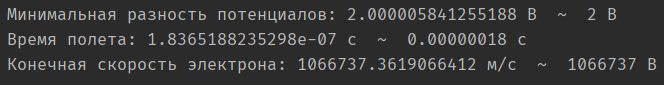
Автоматически созданное описание

Изображение выглядит как текст, число, Шрифт, кроссворд

Автоматически созданное описание

import math  
import matplotlib.pyplot as plt  
  
time\_steps = 10 \*\* 5  
q\_electron = 1.6 \* 10 \*\* (-19)  
r1 = 0.1  
r2 = 0.21  
m = 9.1 \* 10 \*\* (-31)  
v = 8.5 \* 10 \*\* 5  
l = 0.29  
  
  
class Modeling:  
 def \_\_init\_\_(self):  
 self.U = self.bin\_search()  
 self.xc, self.yc, self.tp, self.vy, self.ay = [], [], [], [], []  
 self.d = r2 - r1  
 self.t\_s, self.v\_end = 0, 0  
  
 def simulate\_electron\_movement(self, U):  
 d = r2 - r1  
 y = d / 2  
 r = y + r1  
 calculated\_a = q\_electron \* U / (m \* math.log(r2 / r1))  
 a = calculated\_a / r  
 vy = 0  
 t = 0  
 dt = l / (v \* time\_steps)  
 for i in range(time\_steps):  
 if y - vy \* dt - a \* dt \*\* 2 / 2 > 0:  
 y = y - vy \* dt - a \* dt \*\* 2 / 2  
 vy = vy + a \* dt  
 else:  
 y = 0  
 vy = 0  
 r = y + r1  
 a = calculated\_a / r  
 t += dt  
 if t > (l / v):  
 break  
 return y == 0  
  
 def bin\_search(self):  
 left, right = 2, 100  
 while right - left > 0.00001:  
 mid = (left + right) / 2  
 if self.simulate\_electron\_movement(mid):  
 right = mid  
 else:  
 left = mid  
 return right  
  
 def data(self):  
 calculated\_a = q\_electron \* self.U / (m \* math.log(r2 / r1))  
 y = self.d / 2  
 r = y + r1  
 a = calculated\_a / r  
 vy = 0  
 t = 0  
 dt = l / (v \* time\_steps)  
 for i in range(time\_steps):  
 self.tp.append(t)  
 self.ay.append(a)  
 self.vy.append(vy)  
  
 self.xc.append(v \* t)  
 self.yc.append(y)  
  
 if y - vy \* dt - a \* dt \*\* 2 / 2 > 0:  
 y = y - vy \* dt - a \* dt \*\* 2 / 2  
 vy = vy + a \* dt  
 r = y + r1  
 a = calculated\_a / r  
 else:  
 break  
 t += dt  
 self.t\_s = t  
 if t > (l / v):  
 break  
 self.v\_end = math.sqrt(v \*\* 2 + vy \*\* 2)  
 self.t\_s = t  
 return y == 0  
  
 def provide\_answers(self):  
 print(f"Минимальная разность потенциалов: {self.U:} В ~ {self.U:.0f} В")  
 print(f"Время полета: {self.t\_s:} с ~ {self.t\_s:.8f} с")  
 print(f"Конечная скорость электрона: {self.v\_end:} м/c ~ {self.v\_end:.0f} B")  
  
 def graph(self):  
 plt.figure(figsize=(12, 8))  
  
 plt.subplot(2, 2, 1)  
 plt.plot(self.tp, self.ay, color='blue')  
 plt.title("ay(t)")





Емкость цилиндрического конденсатора C = 2pi\*eps0\*eps\*L/ln(R/r). Емкость конденсатора C q/U = tau\*L/U, линейная плотность заряда tau=q/l. Напряженность поля цилиндрического конденсатора E = q/(2pi\*eps0\*r\*L). Приравниваем формулу мкости конденсатора: tau\*L/U = 2pi\*eps0\*eps\*L/ln(R/r), отсюда линейная плотность заряда tau = 2pi\*eps0\*eps\*U/ln(R/r). Поле внутри цил.конденсатора неоднородно и напряженность убывает с увеличением расстояния от оси системы. Вся система обладает осевой симметрией. Найдем поток поля смещенния по теореме Гаусса: интеграл от Dds = sum(q). D = tau/2pi\*r, D = eps0\*eps\*E. Выражение для нарпяженности поля в указанной точке при r=x: E = tau/(2pi\*eps0\*eps\*x), подставляя tau = 2pi\*eps0\*eps\*U/ln(R/r) получим E = U/(x\*ln(R/r). Вектор E = (вектор F/q) тогда F = q\*E. По второму закону Ньютона F = m\*a тогда a = F/m=q\*U/(m\*x\*ln(R/r). Уравнение движения электрона x(t) = V0\*t + a/2\*t^2. V по y = V0 по y + a\*t. V = sqrt(V0^2 + Vy^2). Чтобы электрон не вылетел он должен пройти расстояние L по горизонтали, двигаясь по вертикали R\*r/2.