Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчёт

по лабораторной работе № 2

Название: Программно-алгоритмическая реализация метода Рунге-Кутта 4го порядка точности при решении системы ОДУ в задаче Коши.

Дисциплина: Моделирование

Студент	ИУ7-65Б		Д.О. Склифасовский
	(Группа)	(Подпись, дата)	(И.О. Фамилия)
Преподователь			В.М. Градов
		(Подпись, дата)	(И.О. Фамилия)

Москва, 2020

Цель работы: Получение навыков разработки алгоритмов решения задачи Коши при реализации моделей, построенных на системе ОДУ, с использованием метода Рунге-Кутта 4-го порядка точности.

Исходные данные

Задана система электротехнических уравнений, описывающих разрядный контур, включающий постоянное активное сопротивление R_k, нелинейное сопротивление $R_p(I)$, зависящее от тока I, индуктивность L_k и емкость C_k.

$$\begin{cases} \frac{dI}{dT} = \frac{U-(R_k+R_p(I))I}{L_k} \\ \frac{dU}{dt} = -\frac{I}{C_k} \end{cases}$$

Начальные условия: $t = 0, I = I_0, U = U_0$.

Здесь I, U - ток и напряжение на конденсаторе.

Сопротивление R_p рассчитать по формуле

$$R_p = \frac{l_p}{2\pi R^2 \int_0^1 \sigma(T(z))zdz}$$

Для функции $T(z)$ применить выражение $T(z) = T_0 + (T_w - T_0)z^m$.

Параметры T_0, m находятся интерполяцией из табл.1 при известном токе I.

Коэффициент электропроводности $\sigma(T)$ зависит от T и рассчитывается интерполяцией из табл.2.

Таблица 1

I, A	To, K	m
0.5	6730	0.50
1	6790	0.55
5	7150	1.7
10	7270	3
50	8010	11
200	9185	32
400	10010	40
800	11140	41
1200	12010	39

Таблица 2.

T, K	σ, 1/Ом см
4000	0.031
5000	0.27
6000	2.05
7000	6.06
8000	12.0
9000	19.9
10000	29.6
11000	41.1
12000	54.1
13000	67.7
14000	81.5

Параметры разряного контура: $R = 0.35$ см,

$L_e = 12$ см,

$L_k = 187 * 10^{-6}$ Гн,

$C_k = 268 * 10^{-6}$ Ф,

$R_k = 0.25$ Ом,

$U_c = 1400$ В,

$I_0 = 0..3$ А,

$T_w = 2000$ К

Теоритические сведения

Метод Рунге-Кутты 4-го порядка точности.

$$y_{n+1} = y_n + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6},$$

$$z_{n+1} = z_n + \frac{q_1 + 2q_2 + 2q_3 + q_4}{6}$$

$$k_1 = h_n f(y_n, z_n), \quad q_1 = h_n \varphi(y_n)$$

$$k_2 = h_n f(y_n + \frac{k_1}{2}, z_n + \frac{q_1}{2}), \quad q_2 = h_n \varphi(y_n + \frac{k_1}{2})$$

$$k_3 = h_n f(y_n + \frac{k_2}{2}, z_n + \frac{q_2}{2}), \quad q_3 = h_n \varphi(y_n + \frac{k_2}{2})$$

$$k_4 = h_n f(y_n + k_3, z_n + q_3, \quad q_4 = h_n \varphi(y_n + k_3)$$

Реализация

Листинг 1 – Интерполяция

```
double Interpolation(double Y, List<double> tableY, List<double> table)
{
    int iMax = 0,
    iMin = 0;
    for (int i = 0; i < tableY.Count; i++)
    {
        if (Y > tableY[i])
            iMax = i;
        else
        {
            iMax = i;
            break;
        }
    }
    if (iMax == 0)
        iMax = 1;
    iMin = iMax - 1;
    return table[iMin] + (table[iMax] - table[iMin]) / (tableY[iMax] - tableY[iMin]) * (Y - tableY[iMin]);
}
```

Листинг 2 – Интегрирование

```
double GetInt(double I, double arg)
{
    double tZero = Interpolation(I, arrI, arrTZero);
    TZeroForGraph = tZero;
    double m = Interpolation(I, arrI, arrM);
    double t = tZero + (tw - tZero) * (Math.Pow(arg, m));
    double sigma = Interpolation(t, arrT, arrSigma);
    return sigma * arg;
```

```
}

double CalculateIntegral(double I)
{
    double a = 0,
        b = 1,
        n = 100,
        h = (b - a) / n,
        result = (GetInt(I, a) + GetInt(I, b)) / 2,
        curr = 0;

    for (int k = 0; k < n - 1; k++)
    {
        curr += h;
        result += GetInt(I, curr);
    }
    return result * h;
}
```

Листинг 3 – Нахождение сопротивления

```
double GetRp(double Le, double R, double I)
{
    double res = Le / (2 * Math.PI * Math.Pow(R, 2) *
      CalculateIntegral(I));
    return res;
}
```

Листинг 4 – Решение системы методом Рунге-Кутта

```
double f(double I, double U, double Le, double R, double Lk,
   double Rk)
{
    RpForGraph = GetRp(Le, R, Math.Abs(I));
    return (U - (Rk + RpForGraph) * I) / Lk;
}

```

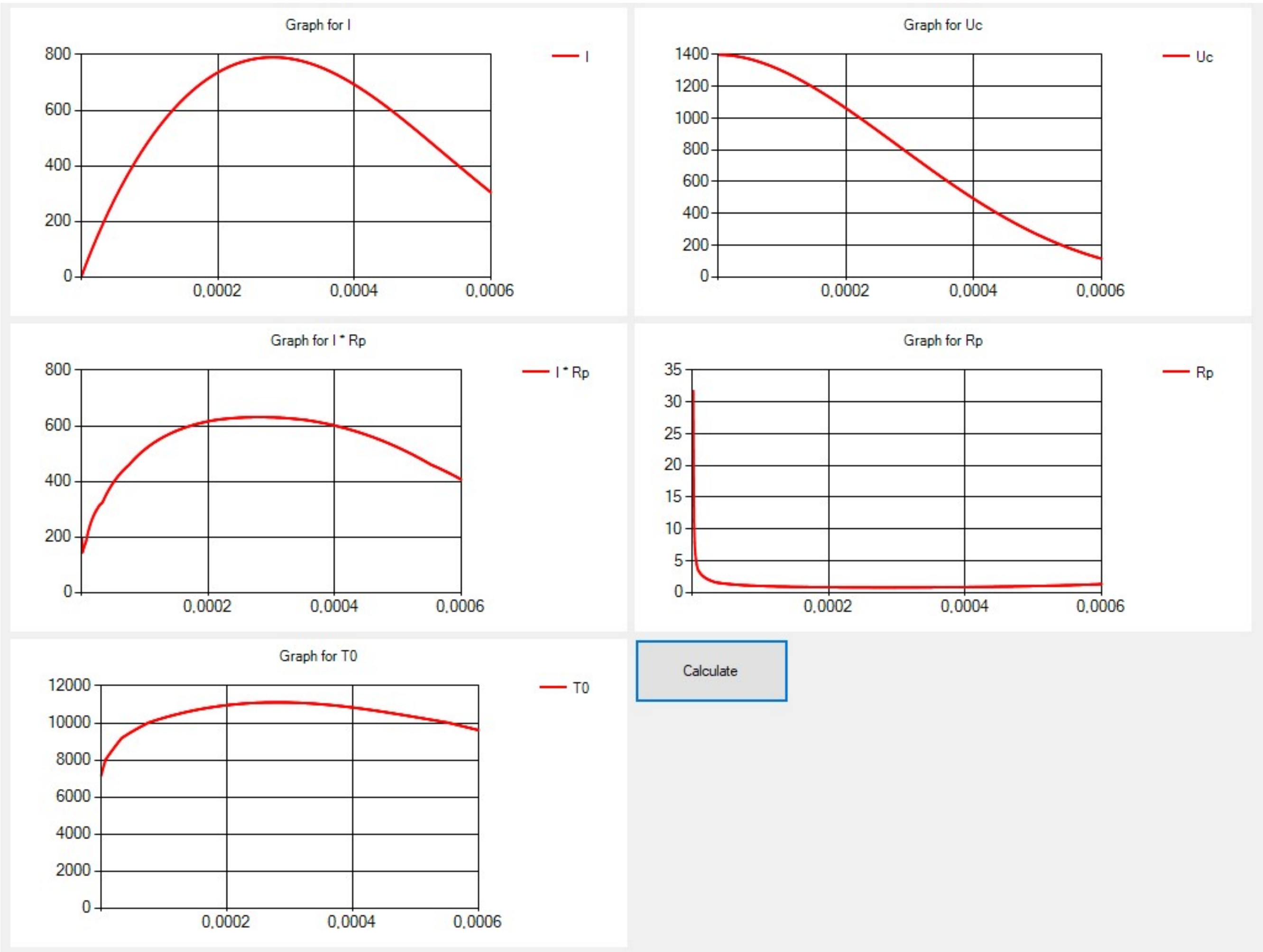
```
double g(double I, double Ck)
{
    return -I / Ck;
}

List<double> GetCoefs(double I, double U, double Le, double R, double Lk, double hn, double Rk, double Ck)
{
    double k1 = f(I, U, Le, R, Lk, Rk),
        q1 = g(I, Ck),
        k2 = f(I + hn * k1 / 2, U + hn * q1 / 2, Le, R, Lk, Rk),
        q2 = g(I + hn * k1 / 2, Ck),
        k3 = f(I + hn * k2 / 2, U + hn * q2 / 2, Le, R, Lk, Rk),
        q3 = g(I + hn * k2 / 2, Ck),
        k4 = f(I + hn * k3, U + hn * q3, Le, R, Lk, Rk),
        q4 = g(I + hn * k3, Ck);
    return new List<double>() { k1, k2, k3, k4, q1, q2, q3, q4 };
}

List<double> GetIAndU(double I, double U, double Le, double R, double Lk, double hn, double Rk, double Ck)
{
    List<double> coefs = GetCoefs(I, U, Le, R, Lk, hn, Rk, Ck);
    List<double> result = new List<double>() {
        I + hn * (coefs[0] + 2 * coefs[1] + 2 * coefs[2] + coefs[3]) / 6,
        U + hn * (coefs[4] + 2 * coefs[5] + 2 * coefs[6] + coefs[7]) / 6
    };
    return result;
}
```

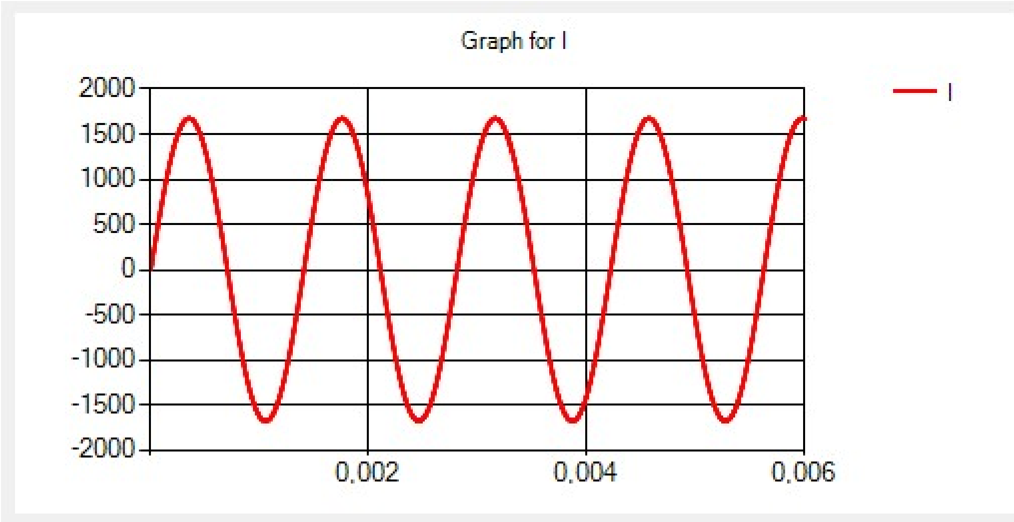
Результаты работы программы

1. Графики зависимости от времени импульса t: $I(t), U(t), R_p(t)$, произведения $I(t)R_p(t), T_0(t)$ при заданных выше параметрах. Указать шаг сетки.

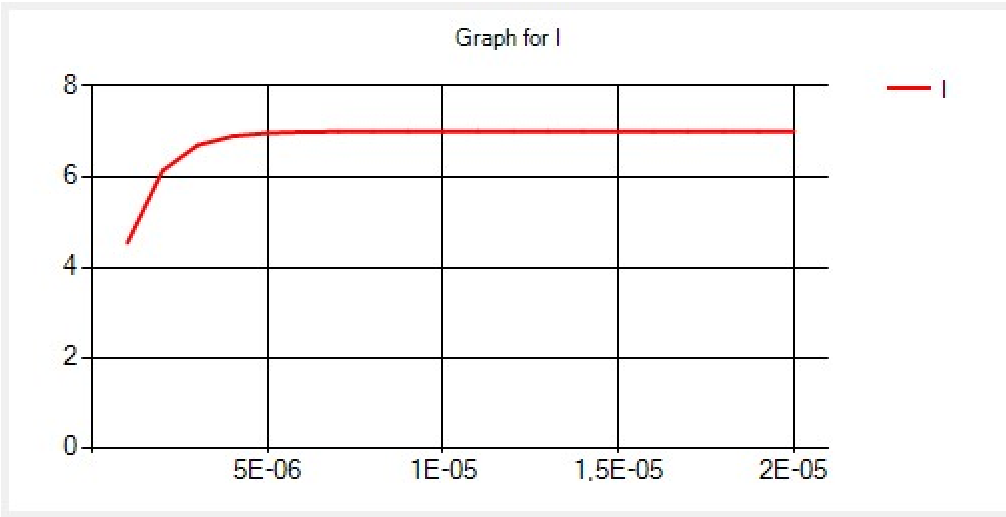


Шаг сетки - 1e-6

2. График зависимости $I(t)$ при $R_k + R_p = 0$. Обратить внимание на то, что в этом случае колебания тока будут незатухающими.



3. График зависимости $I(t)$ при $R_k + R_p = \text{const} = 200$ Ом в интервале значений 0 - 20 мкс.



4. Результаты исследования влияния параметров контура C_k, L_k, R_k на длительность импульса $t_{\text{имп}}$ апериодической формы. Длительность импульса определяется по кривой зависимости тока от времени на высоте $0.35 I_{\max}$, $I_{\max}$ - значение тока в максимуме.

   График при начальных значениях:

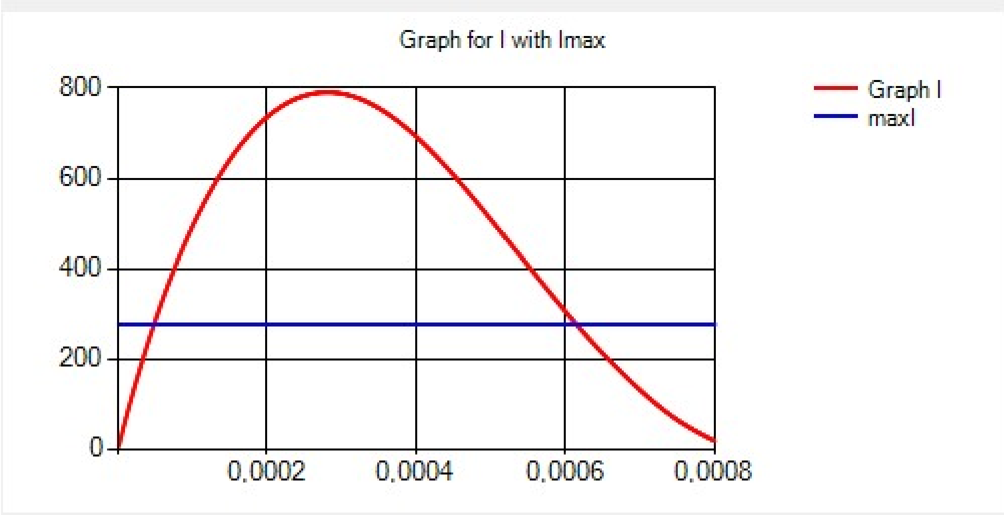
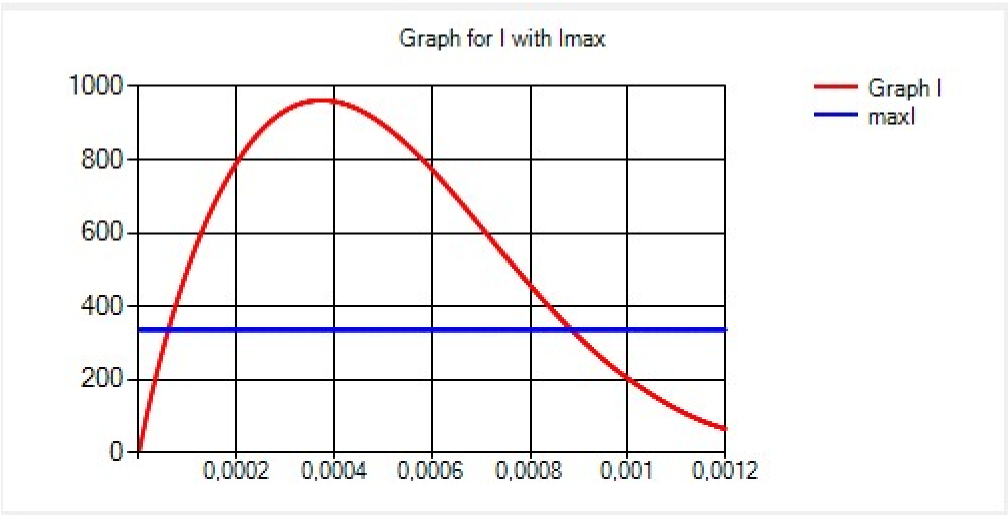
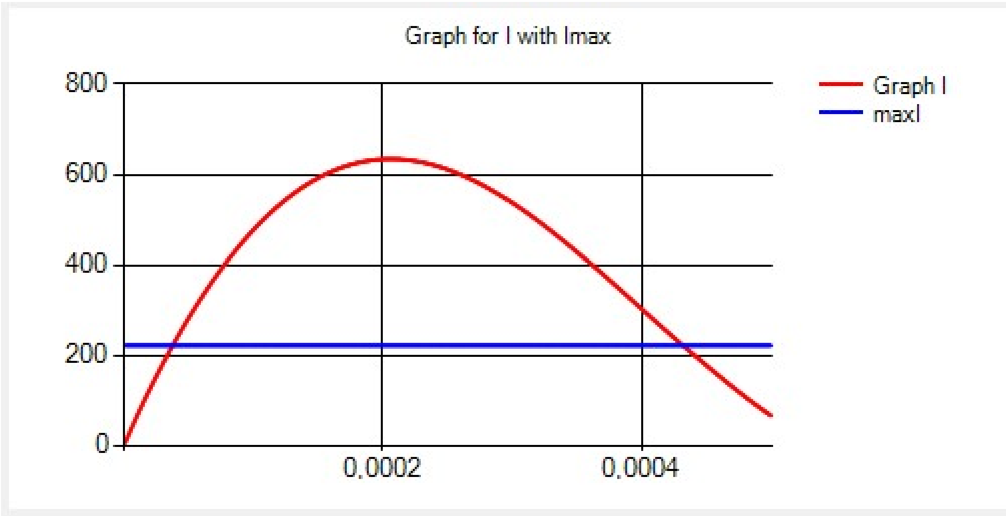


График при увеличении C_k в 2 раза:



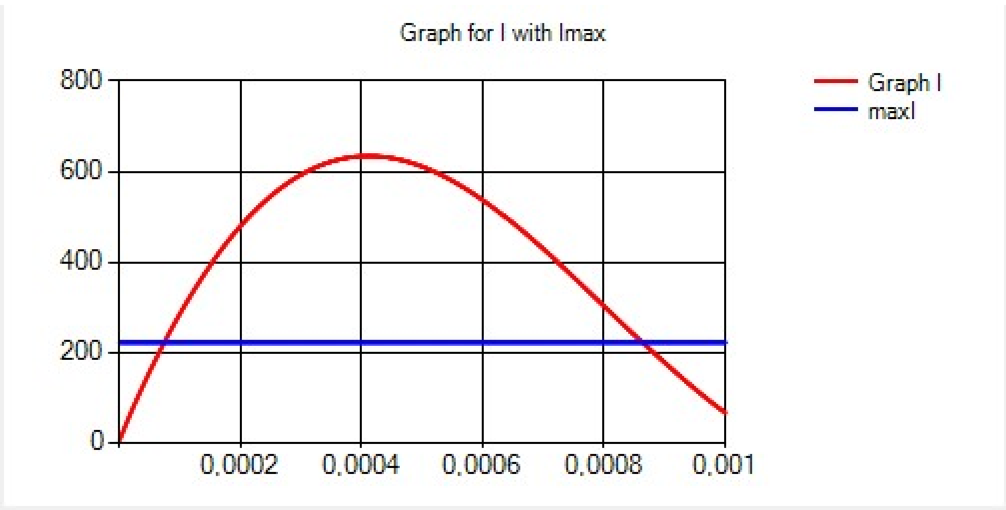
При увеличении C_k, $t_{\text{имп}}$ увеличивается.

График при уменьшении C_k в 2 раза:



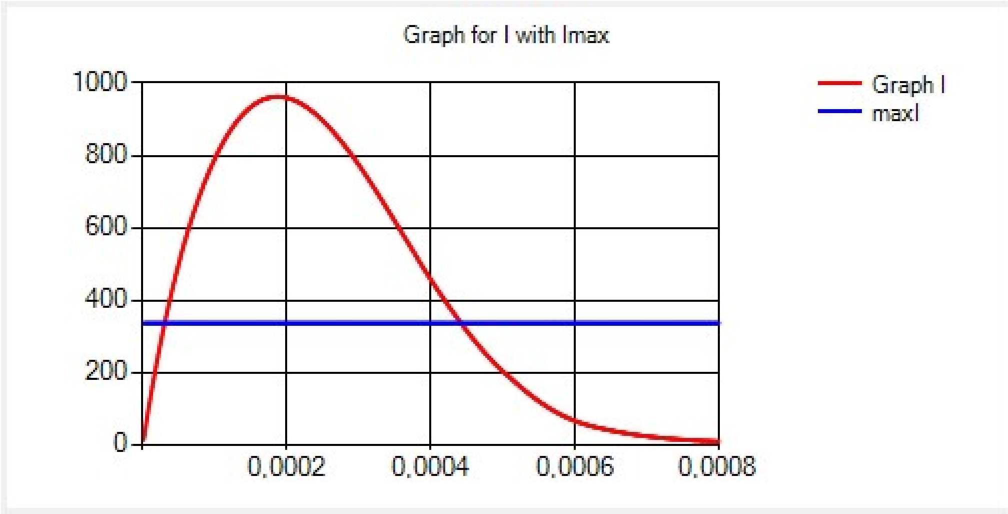
При уменьшении C_k, $t_{\text{имп}}$ уменьшается.

График при увеличении L_k в 2 раза:



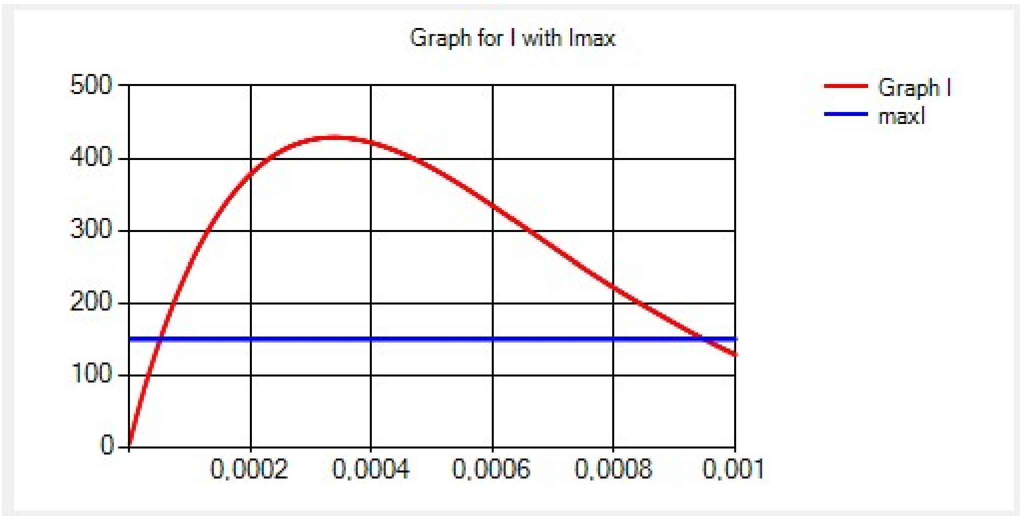
При увеличении L_k, $t_{\text{имп}}$ увеличивается.

График при уменьшении L_k в 2 раза:



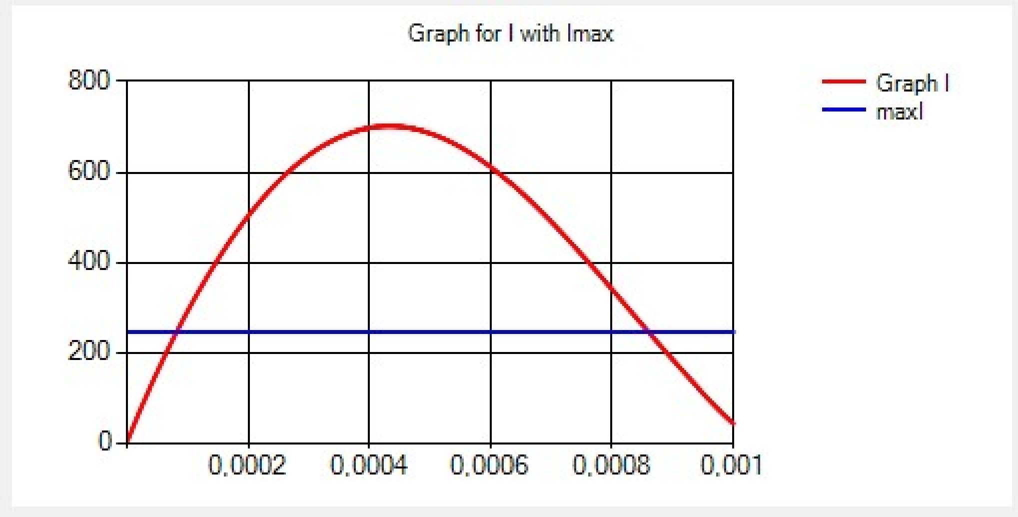
При уменьшении L_k, $t_{\text{имп}}$ уменьшается.

График при увеличении R_k в 5 раза:



При увеличении R_k, $t_{\text{имп}}$ увеличивается.

График при уменьшении R_k в 5 раза:



При уменьшении R_k, $t_{имп}$ уменьшается.

Ответы на вопросы

1. Какие способы тестирования программы, кроме указанного в п.2, можете предложить ещё?

   Можно убрать лампу, тогда при большом значении параметра R_k будет апериодическое затухание, а при небольшом - затухающие колебания.

2. Получите систему разностных уравнений для решения сформулированной задачи неявным методом трапеций. Опишите алгоритм реализации полученных уравнений.

   $u_{n+1} = u_n + \int_{x_n}^{x_{n+1}} f(x, u(x))dx$

   $u_{n+1} = u_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, u_{n+1})] + O(h^2)$

   $$\begin{cases} \frac{dI}{dT} = \frac{U-(R_k+R_p(I))I}{L_k} \\ \frac{dU}{dT} = \frac{-I}{C_k} \end{cases}$$

   $I_{n+1} = I_n + \frac{h}{2}[\frac{U_n-(R_k+R_p(I_n))I_n}{L_k} + \frac{U_{n+1}-(R_k+R_p(I_{n+1}))I_{n+1}}{L_k}]$

   $U_{n+1} = U_n + \frac{h}{2}[-\frac{I_n}{C_k} - \frac{I_{n+1}}{C_k}] = U_n - \frac{h}{2}[\frac{I_n+I_{n+1}}{C_k}]$

Подставляя U_{n+1} в выражение для I_{n+1}

$I_{n+1} = I_n + \frac{h}{2L_k}[2U_n - (R_k + R_p(I_n) + \frac{h}{2C_k})I_n - (R_k + R_p(I_{n+1}) + \frac{h}{2C_k})I_{n+1}]$

Получим уравнение вида

$x = f(x)$

Его можно решить методом простых итераций или методом Ньютона, после этого определить U_{n+1}.

3. Из каких соображений проводится выбор численного метода того или иного порядка точности, учитывая, что чем выше порядок точности метода, тем он более сложен и требует, как правило, больших ресурсов вычислительной системы?

   Оценка для частного случая вида правой части дифференциального уравнения $\phi(x, u) \equiv \phi(x)$

   Если правая часть непрерывна и ограничена, и её четвёртые производные тоже, то использование метода Рунге-Кутта четвёртого порядка имеет смысл.

   Иначе, предельный порядок схемы Рунге-Кутта не может быть достигнут, и стоит использовать более простые схемы.