

*Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»*

МГТУ им. Н.Э. Баумана
Факультет “Энергомашиностроение”
Кафедра Э5 “Вакуумная и компрессорная техника”

ДОМАШНЕЕ ЗАДАНИЕ № 1 и № 2
по курсу “Пневмосистемы и их элементы”

«Расчёт пневматической системы. Этапы I и II»

Вариант 10-3

Студент: Вульф М.Д.
Группа: Э5-81

Преподаватель: Чернышёв А.В.

Москва
2018/2019 учебный год

Введение

Неотъемлемой частью большинства современных сложных технических систем наземного применения и летательных аппаратов является одна или несколько пневматических систем. Чаще всего пневматические системы являются составной частью более общей пневмомеханической или пневмогидравлической системы. Примером таких систем может служить система наддува топливных баков. В летательных аппаратах пневмосистемы, прежде всего, применяются в системах ориентации и стабилизации, кондиционирования и пожаротушения. В наземных комплексах пневмосистемы применяются практически во всех областях техники, например, системы искусственного дыхания в медицине, прецизионные системы подготовки газовых смесей в лазерной технологии, а также системы воздухообеспечения машиностроительных, горнодобывающих и горно-обогатительных предприятий. Диапазон применения пневмосистем крайне широк, а их рабочие характеристики оказывают существенное влияние на работоспособность, энергоемкость и надежность комплексов, в состав которых они входят. Поэтому современный инженер, специализирующийся в области разработки и эксплуатации пневматических систем, должен владеть основами расчета процессов, протекающих в них.

Техническое задание

Требуется произвести расчет процессов наполнения и опорожнения емкостей условной пневматической системы (ПС). Расчётные схемы пневматических систем представляют набор емкостей постоянного объема, соединенных между собой. На рис. 1 приведена расчётная схема рассматриваемого варианта задания (согласно схеме № 10). В таблице 1 содержатся исходные данные для расчёта (согласно варианту №3).

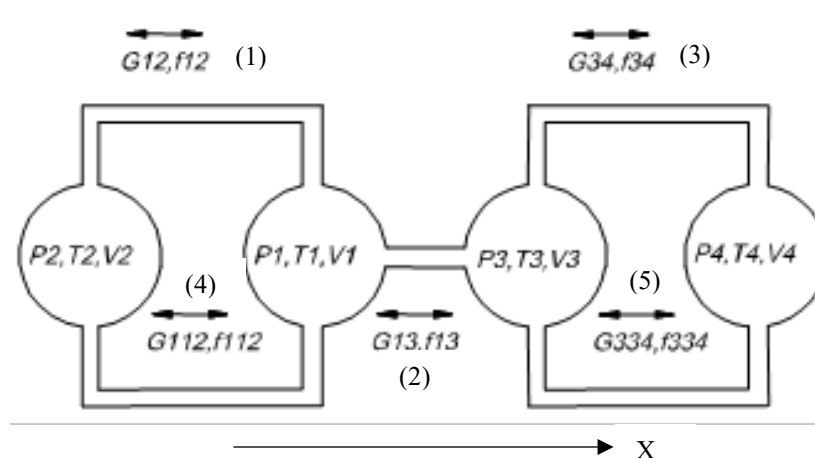


Рис. 1. Расчётная схема №10

Таблица 1. Исходные данные

$P_1 \times 10^{-5}$, Па	2	μ_{12}	0,75	$V_1 \times 10^3$, м ³	4
$P_2 \times 10^{-5}$, Па	5	μ_{13}	0,75	$V_2 \times 10^3$, м ³	5
$P_3 \times 10^{-5}$, Па	8	μ_{34}	0,8	$V_3 \times 10^3$, м ³	3
$P_4 \times 10^{-5}$, Па	12	μ_{112}	0,9	$V_4 \times 10^3$, м ³	6
$d_{12} \times 10^3$, м	0,5	μ_{334}	0,9		
$d_{13} \times 10^3$, м	0,8	T_1 , К	300		
$d_{34} \times 10^3$, м	0,9	T_2 , К	300		
$d_{112} \times 10^3$, м	0,7	T_3 , К	300		
$d_{334} \times 10^3$, м	0,6	T_4 , К	300		

Математическая модель

На первом этапе расчет ПС проводился при следующих допущениях:

1. При моделировании процессов, протекающих в системе, за рабочее тело принят идеальный газ, подчиняющийся термическому уравнению состояния;
2. Изменение состояния рабочего тела в полостях системы квазистатическое;
3. Теплообмен между рабочим телом и корпусными элементами системы отсутствует на первом этапе расчёта и учитывается на втором;
4. Процесс истечения рабочего тела из полостей системы через условные сопротивления принимается одномерным, установившимся, протекающим без трения и теплообмена. Потери при истечении рабочего тела, обусловленные принятыми допущениями, учитываются коэффициентами расхода.

В соответствии с принятыми допущениями, математическая модель расчёта ПС для рассматриваемого варианта на первом этапе выполнения домашнего задания представляет собой следующее.

Начальные условия для расчета ПС описывают состояние системы в момент времени $t=0$ (таблица 1):

$$P_1=P_{1н}, P_2=P_{2н}, P_3=P_{3н}, P_4=P_{4н}, T_1=T_2=T_3=T_4=T_{ос}.$$

Для воздуха приняты показатель адиабаты и газовая постоянная:

$$k=1,41$$

$$R = 287,2 \frac{\text{Дж}}{\text{кг} \cdot \text{К}}$$

Коэффициент теплоотдачи принят:

$$\alpha = 5 \frac{\text{Вт}}{\text{м}^2 \cdot \text{К}}$$

Уравнения математической модели:

- Уравнения расхода рабочего тела через условные сопротивления по формуле Сен-Венана и Ванцеля:

$$G_i = \pm(\mu * f)_i * P_A * \sqrt{\frac{2k}{(k-1)RT_A} \left[\left(\frac{P_B}{P_A} \right)^{\frac{2}{k}} - \left(\frac{P_B}{P_A} \right)^{\frac{k+1}{k}} \right]} \text{ при } \beta \geq \beta_{кр}$$

$$G_i = \pm(\mu * f)_i * P_A * \sqrt{\frac{2k}{(k-1)RT_A} \left[(\beta_{кр})^{\frac{2}{k}} - (\beta_{кр})^{\frac{k+1}{k}} \right]} \text{ при } \beta < \beta_{кр}$$

где $i = 12, 13, 34, 112, 334$ – номер условного сопротивления согласно рис. 1; P_A и P_B – наибольшее и наименьшее давления сосудов соответственно, которые сообщаются посредством соответствующего условного сопротивления; T_A – температура в сосуде с наибольшим давлением; $\beta = \frac{P_B}{P_A}$ – отношение давлений; $\beta_{кр} = 0,528$ – критическое отношение давлений для воздуха; μ – коэффициент расхода; f – площадь поперечного сечения условного сопротивления. В данной модели расход считается положительным, если направление истечения газа совпадает с выбранным направлением оси X (рис. 1).

- Дифференциальные уравнения, описывающее изменение давления рабочего тела в полостях системы, с учётом отсутствия изменения объёма сосудов:

$$\begin{aligned} \frac{dP_1}{dt} &= \frac{k}{V_1} \cdot (R(T_{A1}(G_{12} + G_{112}) - T_{A2}G_{13}) + \frac{dQ_1}{dt} * (k - 1)) \\ \frac{dP_2}{dt} &= \frac{k}{V_2} \cdot (-RT_{A1}(G_{12} + G_{112}) + \frac{dQ_2}{dt} * (k - 1)) \\ \frac{dP_3}{dt} &= \frac{k}{V_3} \cdot (R(T_{A2}G_{13} - T_{A3} * ((G_{34} + G_{334}))) + \frac{dQ_3}{dt} * (k - 1)) \\ \frac{dP_4}{dt} &= \frac{k}{V_4} \cdot (RT_{A3}(G_{34} + G_{334}) + \frac{dQ_4}{dt} * (k - 1)) \end{aligned}$$

где T_{Ai} – температура сосуда, из которого течёт газ, V – объём сосуда.

- Здесь изменение теплового потока через поверхность теплообмена, представляющую поверхность шара определяется следующим образом:

$$\frac{dQ_i}{dt} = \alpha * S_{Ti} * (T_{Si} - T_i)$$

$$S_{Ti} = \pi * \left(\frac{6 * V_i}{\pi}\right)^{2/3}$$

- Дифференциальные уравнения, описывающее изменение температуры рабочего тела в полостях системы, с учётом отсутствия изменения объёма сосудов:

$$\frac{dT_1}{dt} = \frac{T_1}{p_1 * V_1} \cdot \left[V_1 * \frac{dP_1}{dt} - R * T_1 (G_{12} + G_{112} - G_{13}) + \frac{dQ_1}{dt} * (k - 1) \right]$$

$$\frac{dT_2}{dt} = \frac{T_2}{p_2 * V_2} \cdot \left[V_2 * \frac{dP_2}{dt} + R * T_2 (G_{12} + G_{112}) + \frac{dQ_2}{dt} * (k - 1) \right]$$

$$\frac{dT_3}{dt} = \frac{T_3}{p_3 * V_3} \cdot \left[V_3 * \frac{dP_3}{dt} - R * T_3 (G_{13} - (G_{34} + G_{334})) + \frac{dQ_3}{dt} * (k - 1) \right]$$

$$\frac{dT_4}{dt} = \frac{T_4}{p_4 * V_4} \cdot \left[V_4 * \frac{dP_4}{dt} - R * T_4 (G_{34} + G_{334}) + \frac{dQ_4}{dt} * (k - 1) \right]$$

Конечные условия: $P_{\min} \geq 0,95P_{\max}$, где P_{\min} и P_{\max} – минимальное из давлений в сосудах соответственно. Таким образом, расчет прекращается в момент времени, когда параметры системы устанавливаются таким образом, что минимальное давление в полостях системы будет отличаться от максимального не более, чем на 5 %.

Программа расчёта на ЭВМ

Программа расчёта была написана на языке C# в среде Visual Studio Community. Программа расчёта имеет графический интерфейс (рис. 2) и использует методы объектно-ориентированного программирования. Программа позволяет вводить данные пневматической системы, отличные от заданного варианта и шаг времени для численного решения задачи. При нажатии на кнопку «Рассчитать» Программа выполняет численный расчёт пневматической системы и по окончании расчёта выводит данные в Excel, также строится графическая зависимость параметров пневматической системы от времени.

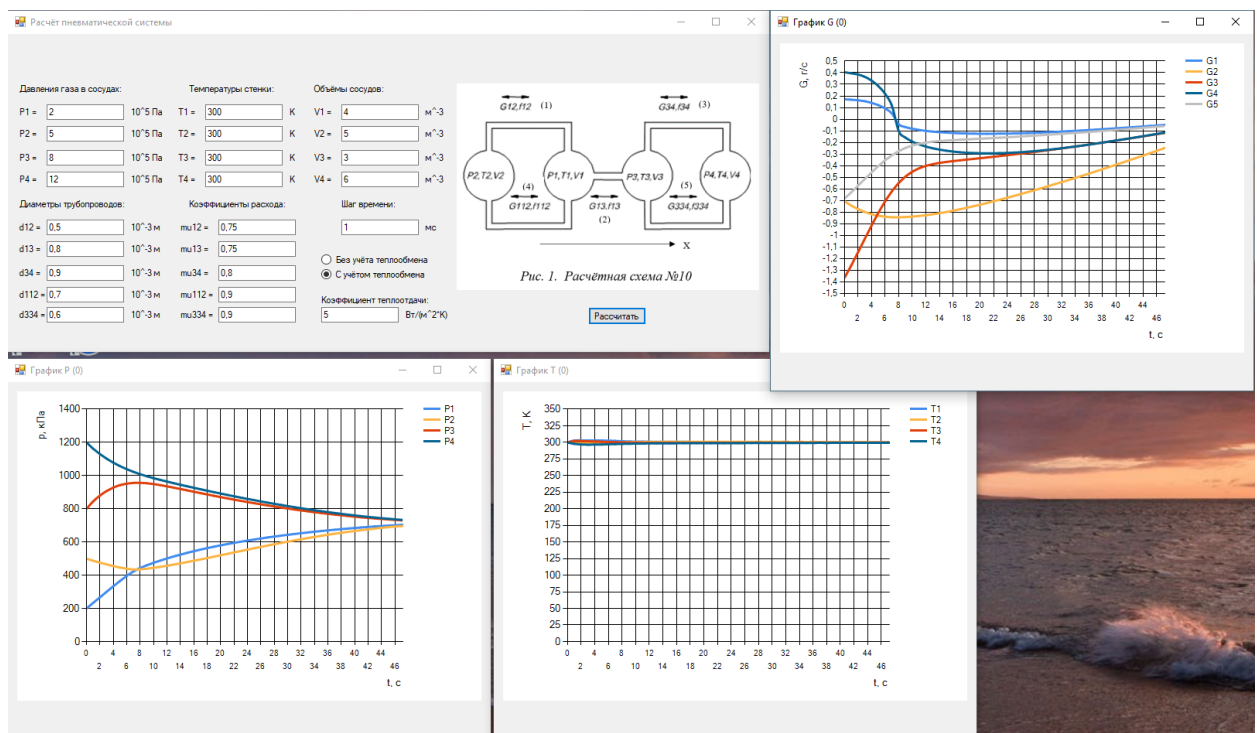


Рис. 2. Графический интерфейс программы расчёта

Код программы:

Класс PneumoCalc:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PnumSysCalc
{
```

```

public class PneumoCalc
{
    //Газовая постоянная воздуха, показатель адиабаты и критическое отношение
    давлений
    public readonly double R = 287.2;
    public readonly double K = 1.41;
    public readonly double BK = 0.528;
    /// <summary>
    /// Определение эффективной площади поперечного сечения условного сопротивления
    /// </summary>
    /// <param name="Mu">Коэффициент расхода</param>
    /// <param name="D">Диаметр условного сопротивления</param>
    /// <returns></returns>
    public double Section(double Mu, double D)
    {
        return Mu * Math.PI * D * D / 4;
    }
    public double Surface(double V)
    {
        return Math.PI*Math.Pow(6*V/Math.PI,2/3);
    }
    /// <summary>
    /// Определение направления расхода
    /// </summary>
    /// <param name="pL">Давление левого сосуда</param>
    /// <param name="pR">Давление правого сосуда</param>
    /// <returns></returns>
    public int FlowDirection(double pL, double pR)
    {
        if (pL > pR) return 1;
        else return -1;
    }
    /// <summary>
    /// Определение расхода
    /// </summary>
    /// <param name="F">Площадь эффективного поперечного сечения условного
    сопротивления</param>
    /// <param name="pL">Давление "левого" сосуда</param>
    /// <param name="pR">Давление "правого" сосуда</param>
    /// <param name="TpA">Температура сосуда с наибольшим давлением</param>
    /// <returns></returns>
    public double Flow(double F, double pL, double pR, double TpA)
    {
        try
        {
            //Определение максимальных давлений
            double pA = Math.Max(pL, pR);
            double pB = Math.Min(pL, pR);

            //Расчёт расхода с учётом направления истечения
            double G = FlowDirection(pL, pR) * F * pA * Math.Sqrt(2 * K / ((K - 1) *
R * TpA)) * Math.Sqrt(Math.Pow(Math.Max(pB / pA, BK), 2 / K) - Math.Pow(Math.Max(pB / pA,
BK), (K + 1) / K));

            return G;
        }
        catch (Exception)
        {
            MessageBox.Show("Ошибка Определения расхода!");
            return 0;
        }
    }
}
}

```


Класс главной формы:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Diagnostics;

namespace PnumSysCalc
{
    public partial class MainForm : Form
    {
        double[] P = new double[5];
        List<double> PCheck = new List<double>(); //Динамический массив
        double[] PLast = new double[4];
        double[] dPdt = new double[5];
        double[] Ts = new double[5];
        double[] T = new double[5];
        double[] dTdt = new double[5];
        double[] dQdt = new double[5];
        double[] V = new double[5];
        double[] D = new double[6];
        double[] Mu = new double[6];
        int[] A = new int[6];
        double[] G = new double[6];
        double[] TPmax = new double[4];
        double[,] PRes = new double[5, 10000000];
        double[,] TRes = new double[5, 10000000];
        double[,] GRes = new double[6, 10000000];
        double DT = new double();
        double Alpha = new double();
        bool flag = true;
        bool[] flagArr = new bool[4];
        int pY = new int();

        //Создание объектов классов для расчёта
        PnumoCalc Calc = new PnumoCalc();

        public MainForm()
        {
            InitializeComponent(); //Инициализация компонентов на форме окна Windows
        }
        /// <summary>
        /// Перевод текста в элементы массива
        /// </summary>
        public void ReadData()
        {
            try
            {
                //Перевод данных в double
                P[1] = double.Parse(textP1.Text) * Math.Pow(10,5);
                P[2] = double.Parse(textP2.Text) * Math.Pow(10, 5);
                P[3] = double.Parse(textP3.Text) * Math.Pow(10, 5);
                P[4] = double.Parse(textP4.Text) * Math.Pow(10, 5);

                T[1] = double.Parse(textT1.Text);
                T[2] = double.Parse(textT2.Text);
            }
        }
    }
}
```

```

T[3] = double.Parse(textT3.Text);
T[4] = double.Parse(textT4.Text);

Ts[1] = double.Parse(textT1.Text);
Ts[2] = double.Parse(textT2.Text);
Ts[3] = double.Parse(textT3.Text);
Ts[4] = double.Parse(textT4.Text);

V[1] = double.Parse(textV1.Text) * Math.Pow(10, -3);
V[2] = double.Parse(textV2.Text) * Math.Pow(10, -3);
V[3] = double.Parse(textV3.Text) * Math.Pow(10, -3);
V[4] = double.Parse(textV4.Text) * Math.Pow(10, -3);

D[1] = double.Parse(textD12.Text) * Math.Pow(10, -3);
D[2] = double.Parse(textD13.Text) * Math.Pow(10, -3);
D[3] = double.Parse(textD34.Text) * Math.Pow(10, -3);
D[4] = double.Parse(textD112.Text) * Math.Pow(10, -3);
D[5] = double.Parse(textD334.Text) * Math.Pow(10, -3);

Mu[1] = double.Parse(textMu12.Text);
Mu[2] = double.Parse(textMu13.Text);
Mu[3] = double.Parse(textMu34.Text);
Mu[4] = double.Parse(textMu112.Text);
Mu[5] = double.Parse(textMu334.Text);

DT = double.Parse(textStep.Text)/1000;
if (radioTerm.Checked == true)
    Alpha = double.Parse(textAlpha.Text);
else Alpha = 0;
}
catch (Exception)
{
    MessageBox.Show("Ошибка ввода!");
}
}

/// <summary>
/// Нажатие на кнопку расчёт
/// </summary>
/// <param name="sender"></param>
/// <param name="e"></param>
private void bCalc_Click(object sender, EventArgs e)
{
    //Удаление лишних процессов Excel
    try
    {
        foreach (Process proc in Process.GetProcessesByName("Microsoft Excel"))
        {
            proc.Kill();
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }

    Excel myExcel = new Excel();
    myExcel.NewDocument();//Создание нового документа Excel

    //Создание объектов класса Графическое окно и их настройка
    ChartForm PChart = new ChartForm
    {

```

```

        Location = new Point(this.Location.X, this.Location.Y + 450 + pY),
        Text = "График P (" + (pY / 50).ToString() + ")"
    };

    //Инициализация графика
    PChart.Show();
    PChart.chart.ChartAreas[0].AxisY.Title = "p, кПа";
    PChart.chart.ChartAreas[0].AxisY.Interval = 200;

    ChartForm TChart = new ChartForm
    {
        StartPosition = FormStartPosition.Manual,
        Location = new Point(PChart.Location.X + 650, PChart.Location.Y),
        Text = "График T (" + (pY / 50).ToString() + ")"
    };

    TChart.Show();
    TChart.chart.ChartAreas[0].AxisY.Title = "T, K";
    TChart.chart.ChartAreas[0].AxisY.Interval = 25;

    ChartForm GChart = new ChartForm
    {
        StartPosition = FormStartPosition.Manual,
        Location = new Point(TChart.Location.X + 650, PChart.Location.Y),
        Text = "График G (" + (pY / 50).ToString() + ")"
    };

    GChart.Show();
    GChart.chart.ChartAreas[0].AxisY.Title = "G, г/с";
    GChart.chart.ChartAreas[0].AxisY.Interval = 0.10;

    pY = pY + 50;

    //Настройка графиков
    for (int i = 0; i < 4; i++)
    {
        PChart.chart.Series.Add("P" + (i + 1).ToString());
        PChart.chart.Series[i].ChartType =
System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Spline;
        PChart.chart.Series[i].BorderWidth = 3;

        TChart.chart.Series.Add("T" + (i + 1).ToString());
        TChart.chart.Series[i].ChartType =
System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Spline;
        TChart.chart.Series[i].BorderWidth = 3;
    }

    for (int i = 0; i < 5; i++)
    {
        GChart.chart.Series.Add("G" + (i + 1).ToString());
        GChart.chart.Series[i].ChartType =
System.Windows.Forms.DataVisualization.Charting.SeriesChartType.Spline;
        GChart.chart.Series[i].BorderWidth = 3;
    }

    //Получение данных газовой постоянной и показателя адиабаты
    double R = Calc.R;
    double K = Calc.K;

    double Pmax = new double();
    double Pmin = new double();

```

```

//Определение данных
ReadData();

double[] F = new double[6]; //Эффективное сечение

//Определение Эффективных сечений
for(int i=1; i<6; i++)
{
    F[i] = Calc.Section(Mu[i], D[i]);
}

double[] S = new double[5]; //Площадь поверхности

for (int i = 1; i < 5; i++)
{
    S[i] = Calc.Surface(V[1]);
}

int n = 0;
double tt = 0;
int gt = 1000; //шаг вывода на графики
string lbl = ""; //строка вывода

//Вывод первой строки заголовка в Excel
myExcel.SetValue("A1", "t,c");

myExcel.SetValue("B1", "P1,Па");
myExcel.SetValue("C1", "P2,Па");
myExcel.SetValue("D1", "P3,Па");
myExcel.SetValue("E1", "P4,Па");

myExcel.SetValue("F1", "T1,K");
myExcel.SetValue("G1", "T2,K");
myExcel.SetValue("H1", "T3,K");
myExcel.SetValue("I1", "T4,K");

myExcel.SetValue("J1", "G12,г/с");
myExcel.SetValue("K1", "G13,г/с");
myExcel.SetValue("L1", "G34,г/с");
myExcel.SetValue("M1", "G112,г/с");
myExcel.SetValue("N1", "G334,г/с");

do //ЦИКЛ
{

    //Определение температур исходящих потоков
    if (P[1] > P[2]) TPmax[1] = T[1];
    else TPmax[1] = T[2];

    if (P[3] > P[1]) TPmax[2] = T[3];
    else TPmax[2] = T[1];

    if (P[4] > P[3]) TPmax[3] = T[4];
    else TPmax[3] = T[3];

    //Определение расходов
    G[1] = Calc.Flow(F[1], P[2], P[1], TPmax[1]);
    G[4] = Calc.Flow(F[4], P[2], P[1], TPmax[1]);

    G[2] = Calc.Flow(F[2], P[1], P[3], TPmax[2]);

    G[3] = Calc.Flow(F[3], P[3], P[4], TPmax[3]);
    G[5] = Calc.Flow(F[5], P[3], P[4], TPmax[3]);
}

```

```

//Определение тепловых потоков
for (int i =1; i<5; i++)
{
    dQdt[i] = Alpha * S[i] * (Ts[i] - T[i]);
}

//Вывод текущих данных в матрицу
for (int i = 0; i < 4; i++)
{
    PRes[i,n] = P[i+1];
    TRes[i,n] = T[i+1];
    GRes[i,n] = G[i+1];
}
GRes[4,n] = G[5];

if ((n % gt == 0) | (n == 0)) {
    //Вывод данных в Excel
    lbl = ((int)(n / gt) + 2).ToString();//переменная номера ячейки excel

    myExcel.SetValue("A" + lbl, (Math.Round(tt,3)).ToString());

    myExcel.SetValue("B" + lbl, (Math.Round(PRes[0, n])).ToString());
    myExcel.SetValue("C" + lbl, (Math.Round(PRes[1, n])).ToString());
    myExcel.SetValue("D" + lbl, (Math.Round(PRes[2, n])).ToString());
    myExcel.SetValue("E" + lbl, (Math.Round(PRes[3, n])).ToString());

    myExcel.SetValue("F" + lbl, (Math.Round(TRes[0, n],1)).ToString());
    myExcel.SetValue("G" + lbl, (Math.Round(TRes[1, n],1)).ToString());
    myExcel.SetValue("H" + lbl, (Math.Round(TRes[2, n],1)).ToString());
    myExcel.SetValue("I" + lbl, (Math.Round(TRes[3, n],1)).ToString());

    myExcel.SetValue("J" + lbl, (Math.Round(GRes[0, n] * 1000,
2)).ToString());
    myExcel.SetValue("K" + lbl, (Math.Round(GRes[1, n] * 1000,
2)).ToString());
    myExcel.SetValue("L" + lbl, (Math.Round(GRes[2, n] * 1000,
2)).ToString());
    myExcel.SetValue("M" + lbl, (Math.Round(GRes[3, n] * 1000,
2)).ToString());
    myExcel.SetValue("N" + lbl, (Math.Round(GRes[4, n] * 1000,
2)).ToString());

    //Вывод данных на график
    for (int i = 0; i < 4; i++)
    {
        PChart.chart.Series[i].Points.AddXY(Math.Round(tt, 3), PRes[i, n]
/ 1000);
        TChart.chart.Series[i].Points.AddXY(Math.Round(tt, 3), TRes[i,
n]);
    }
    for (int i = 0; i < 5; i++)
    {
        GChart.chart.Series[i].Points.AddXY(Math.Round(tt, 3), GRes[i, n]
* 1000);
    }
}

n = n+1;
tt = tt + DT;

```

```

//Определение изменений давления и температуры
dPdt[2] = -1 * K * R / V[2] * TPmax[1] * (G[1]+ G[4]) + K/ V[2] * (K -
1)* dQdt[2];
dTdt[2] = T[2] / (P[2] * V[2]) * (V[2] * dPdt[2] - (-1)*R* T[2]*(G[1] +
G[4]) + (K - 1) * dQdt[2]);

dPdt[1] = K * R / V[1] * (TPmax[1] * (G[1] + G[4]) - TPmax[2]* G[2]) + K /
V[1] * (K - 1) * dQdt[1];
dTdt[1] = T[1] / (P[1] * V[1]) * (V[1] * dPdt[1] - R * T[1] * (G[1] +
G[4] - G[2]) + (K - 1) * dQdt[1]);

dPdt[3] = K * R / V[3] * (TPmax[2] * G[2] - TPmax[3] * (G[3]+G[5])) + K /
V[3] * (K - 1) * dQdt[3];
dTdt[3] = T[3] / (P[3] * V[3]) * (V[3] * dPdt[3] - R * T[3] * (G[2] -
(G[3] + G[5])) + (K - 1) * dQdt[3]);

dPdt[4] = K * R / V[4] * (TPmax[3] * (G[3] + G[5])) + K / V[4] * (K - 1)
* dQdt[4];
dTdt[4] = T[4] / (P[4] * V[4]) * (V[4] * dPdt[4] - R * T[4] * (G[3] +
G[5]) + (K - 1) * dQdt[4]);

PCheck.Clear(); //Очистка динамического массива
//Пересчёт новых давлений
for (int i = 1; i < 5; i++)
{
    flagArr[i - 1] = true;

    PLast[i - 1] = P[i];

    P[i] = P[i] + dPdt[i] * DT;
    T[i] = T[i] + dTdt[i] * DT;

    if (P[i] != PLast[i - 1]) //Если давление изменяется, то добавить к
сравнению
        PCheck.Add(P[i]);

    //Проверка расхождения
    if (Math.Abs((P[i] - PLast[i - 1]) / PLast[i - 1]) < 0.0000001) {
flagArr[i - 1] = false; }
        else flagArr[i - 1] = true;
    }

    if (flagArr.Max() == false) flag = false;

    //Определение максимального и минимального давлений

    if (PCheck.LongCount() == 0) MessageBox.Show("Давления не изменятся!!!");
    else {
        Pmax = PCheck.Max();
        Pmin = PCheck.Min();
    }

} while ((Pmin<=0.95* Pmax)&(flag)); //Условие окончания счёта - разница
между max и min не больше 5%

lbl = ((int)(n / gt) + 3).ToString();
n--;

//Вывод последних значений следующих величин: расходов, давлений, температур
myExcel.SetValue("A" + lbl, (Math.Round(tt, 2)).ToString());

myExcel.SetValue("B" + lbl, (Math.Round(PRes[0, n])).ToString());
myExcel.SetValue("C" + lbl, (Math.Round(PRes[1, n])).ToString());
myExcel.SetValue("D" + lbl, (Math.Round(PRes[2, n])).ToString());

```

```

myExcel.SetValue("E" + lbl, (Math.Round(PRes[3, n])).ToString());

myExcel.SetValue("F" + lbl, (Math.Round(TRes[0, n], 1)).ToString());
myExcel.SetValue("G" + lbl, (Math.Round(TRes[1, n], 1)).ToString());
myExcel.SetValue("H" + lbl, (Math.Round(TRes[2, n], 1)).ToString());
myExcel.SetValue("I" + lbl, (Math.Round(TRes[3, n], 1)).ToString());

myExcel.SetValue("J" + lbl, (Math.Round(GRes[0, n] * 1000, 2)).ToString());
myExcel.SetValue("K" + lbl, (Math.Round(GRes[1, n] * 1000, 2)).ToString());
myExcel.SetValue("L" + lbl, (Math.Round(GRes[2, n] * 1000, 2)).ToString());
myExcel.SetValue("M" + lbl, (Math.Round(GRes[3, n] * 1000, 2)).ToString());
myExcel.SetValue("N" + lbl, (Math.Round(GRes[4, n] * 1000, 2)).ToString());

for (int i = 0; i < 4; i++)
{
    PChart.chart.Series[i].Points.AddXY(Math.Round(tt, 3), PRes[i, n] /
1000);
    TChart.chart.Series[i].Points.AddXY(Math.Round(tt, 3), TRes[i, n]);
}

for (int i = 0; i < 5; i++)
{
    GChart.chart.Series[i].Points.AddXY(Math.Round(tt, 3), GRes[i, n] *
1000);
}

//Развёртывание окна excel
myExcel.Visible = true;

//ПЕРЕКЛИКОВАНИЕ ВСЕГО
P = new double[5];
PCheck = new List<double>(); //Динамический массив
Plast = new double[4];
dPdt = new double[5];
Ts = new double[5];
T = new double[5];
dTdt = new double[5];
dQdt = new double[5];
V = new double[5];
D = new double[6];
Mu = new double[6];
A = new int[6];
G = new double[6];
TPmax = new double[4];
PRes = new double[5, 10000000];
TRes = new double[5, 10000000];
GRes = new double[6, 10000000];

DT = new double();
Alpha = new double();

flag = true;
flagArr = new bool[4];

pY = new int();

//Создание объектов классов для расчёта
Calc = new PneumoCalc();
}
}
}

```

Точка входа в программу:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace PnumSysCalc
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new MainForm()); //Запуск главного окна
        }
    }
}
```


Результаты расчёта

Таблица 2. Результаты расчёта без теплообмена

t, c	P1,кПа	P2,кПа	P3,кПа	P4,кПа	T1,К	T2,К	T3,К	T4,К	G12,г/с	G13,г/с	G34,г/с	G112,г/с	G334,г/с
0	200	500	800	1200	300	300	300	300	0,17	-0,71	-1,36	0,4	-0,68
1	239,454	486,199	849,571	1160,413	315,2	297,6	305	297,1	0,17	-0,74	-1,25	0,4	-0,63
2	279,695	472,859	889,305	1124,836	327,5	295,2	308,4	294,4	0,16	-0,77	-1,12	0,38	-0,56
3	319,978	460,325	919,186	1093,484	337,5	292,9	310,6	292	0,15	-0,8	-0,99	0,35	-0,49
4	359,22	449,224	939,695	1066,32	345,5	290,8	311,7	289,9	0,13	-0,81	-0,86	0,3	-0,43
5	396,184	440,32	951,757	1043,065	351,9	289,1	311,9	288	0,1	-0,82	-0,74	0,22	-0,37
6	429,075	434,857	956,669	1023,234	357	288,1	311,6	286,4	0,04	-0,83	-0,63	0,08	-0,32
7	452,04	437,36	955,965	1006,191	360,1	288,8	310,9	285	-0,05	-0,83	-0,55	-0,12	-0,28
8	470,403	443,45	951,219	991,247	361,9	290,5	309,8	283,8	-0,07	-0,83	-0,49	-0,17	-0,25
9	486,752	450,978	943,829	977,769	363,2	292,6	308,5	282,7	-0,08	-0,82	-0,45	-0,19	-0,23
10	501,647	459,445	934,874	965,261	363,9	295	307,1	281,6	-0,09	-0,81	-0,43	-0,21	-0,21
11	515,373	468,578	925,105	953,384	364,4	297,4	305,7	280,6	-0,1	-0,81	-0,41	-0,22	-0,2
12	528,1	478,193	915,006	941,936	364,6	300	304,2	279,6	-0,1	-0,8	-0,4	-0,23	-0,2
13	539,957	488,153	904,852	930,808	364,6	302,6	302,8	278,6	-0,1	-0,79	-0,39	-0,24	-0,19
14	551,045	498,35	894,797	919,946	364,4	305,1	301,4	277,7	-0,1	-0,77	-0,38	-0,24	-0,19
15	561,448	508,696	884,923	909,326	364,1	307,7	300	276,8	-0,11	-0,76	-0,37	-0,25	-0,19
16	571,24	519,114	875,276	898,941	363,7	310,2	298,6	275,8	-0,11	-0,75	-0,37	-0,25	-0,18
17	580,483	529,539	865,878	888,79	363,2	312,6	297,3	274,9	-0,11	-0,73	-0,36	-0,25	-0,18
18	589,231	539,916	856,745	878,877	362,6	314,9	295,9	274	-0,1	-0,72	-0,35	-0,25	-0,18
19	597,533	550,196	847,884	869,206	362	317,2	294,7	273,1	-0,1	-0,7	-0,35	-0,24	-0,17
20	605,43	560,334	839,301	859,784	361,4	319,4	293,4	272,3	-0,1	-0,68	-0,34	-0,24	-0,17
21	612,957	570,295	830,998	850,617	360,7	321,5	292,2	271,4	-0,1	-0,67	-0,33	-0,24	-0,16
22	620,146	580,044	822,98	841,71	360	323,5	291,1	270,6	-0,1	-0,65	-0,32	-0,23	-0,16
23	627,021	589,553	815,247	833,069	359,4	325,4	290	269,8	-0,1	-0,63	-0,31	-0,23	-0,16
24	633,605	598,797	807,802	824,698	358,7	327,2	288,9	269	-0,09	-0,61	-0,3	-0,22	-0,15
25	639,914	607,756	800,647	816,604	358	328,9	287,8	268,2	-0,09	-0,59	-0,29	-0,21	-0,15
26	645,962	616,412	793,782	808,791	357,4	330,5	286,8	267,5	-0,09	-0,57	-0,28	-0,21	-0,14
27	651,759	624,75	787,21	801,264	356,7	332	285,8	266,8	-0,08	-0,55	-0,27	-0,2	-0,14
28	657,313	632,757	780,931	794,028	356,1	333,5	284,9	266,1	-0,08	-0,53	-0,26	-0,19	-0,13
29	662,629	640,424	774,947	787,088	355,5	334,8	284	265,4	-0,08	-0,51	-0,25	-0,18	-0,13
30	667,708	647,741	769,258	780,448	355	336,1	283,2	264,7	-0,07	-0,48	-0,24	-0,17	-0,12
31	672,552	654,704	763,864	774,114	354,4	337,3	282,4	264,1	-0,07	-0,46	-0,23	-0,17	-0,12
32	677,159	661,307	758,765	768,089	353,9	338,3	281,6	263,5	-0,07	-0,44	-0,22	-0,16	-0,11
33	681,529	667,545	753,962	762,379	353,4	339,4	280,9	262,9	-0,06	-0,42	-0,21	-0,15	-0,1
34	685,658	673,417	749,455	756,987	352,9	340,3	280,2	262,4	-0,06	-0,39	-0,2	-0,14	-0,1
35	689,544	678,919	745,242	751,917	352,5	341,2	279,5	261,9	-0,06	-0,37	-0,19	-0,13	-0,09
36	693,182	684,051	741,323	747,175	352,1	342	278,9	261,4	-0,05	-0,34	-0,17	-0,12	-0,09
37	696,57	688,81	737,697	742,763	351,7	342,7	278,4	260,9	-0,05	-0,32	-0,16	-0,11	-0,08
38	699,704	693,196	734,363	738,685	351,3	343,4	277,9	260,5	-0,04	-0,29	-0,15	-0,1	-0,07
39	702,581	697,208	731,32	734,945	351	344	277,4	260,1	-0,04	-0,27	-0,14	-0,09	-0,07
39,14	702,958	697,732	730,923	734,456	351	344,1	277,4	260,1	-0,04	-0,26	-0,13	-0,09	-0,07

Таблица 2. Результаты расчёта без теплообмена

t, с	P1,кПа	P2,кПа	P3,кПа	P4,кПа	T1,К	T2,К	T3,К	T4,К	G12,г/с	G13,г/с	G34,г/с	G112,г/с	G334,г/с
0	200	500	800	1200	300	300	300	300	0,17	-0,71	-1,36	0,4	-0,68
1	233,959	487,467	845,233	1162,118	302,9	298,9	302,2	297,9	0,17	-0,74	-1,26	0,4	-0,63
2	267,338	476,148	879,954	1129,97	303	298,8	302,1	296,9	0,16	-0,77	-1,15	0,39	-0,57
3	300,868	465,394	907,422	1102,215	303	298,8	301,7	296,6	0,16	-0,8	-1,03	0,37	-0,51
4	333,964	455,44	928,277	1078,1	302,9	298,9	301,3	296,7	0,14	-0,82	-0,91	0,33	-0,46
5	365,976	446,667	942,979	1057,127	302,8	299	301	296,9	0,12	-0,83	-0,81	0,29	-0,4
6	396,155	439,559	952,128	1038,887	302,6	299,2	300,6	297,2	0,09	-0,84	-0,71	0,22	-0,35
7	423,445	434,863	956,484	1022,976	302,4	299,5	300,3	297,5	0,05	-0,84	-0,62	0,12	-0,31
8	444,095	435,516	956,91	1008,973	301,8	300,1	300	297,8	-0,04	-0,84	-0,55	-0,1	-0,27
9	460,308	439,652	954,3	996,462	301,4	300,4	299,8	298	-0,07	-0,84	-0,49	-0,16	-0,25
10	475,008	444,872	949,482	985,056	301,3	300,6	299,7	298,2	-0,08	-0,84	-0,45	-0,19	-0,23
11	488,588	450,866	943,154	974,435	301,2	300,7	299,6	298,4	-0,09	-0,83	-0,42	-0,22	-0,21
12	501,27	457,448	935,851	964,354	301,1	300,7	299,5	298,5	-0,1	-0,83	-0,4	-0,24	-0,2
13	513,184	464,49	927,968	954,648	301	300,8	299,5	298,6	-0,11	-0,82	-0,39	-0,25	-0,19
14	524,414	471,899	919,782	945,211	301	300,8	299,4	298,7	-0,11	-0,81	-0,38	-0,26	-0,19
15	535,022	479,601	911,472	935,982	300,9	300,9	299,4	298,7	-0,12	-0,8	-0,37	-0,27	-0,18
16	545,061	487,536	903,151	926,928	300,9	300,9	299,4	298,7	-0,12	-0,79	-0,36	-0,28	-0,18
17	554,573	495,653	894,891	918,031	300,8	300,9	299,4	298,8	-0,12	-0,77	-0,35	-0,28	-0,18
18	563,597	503,908	886,737	909,284	300,8	300,9	299,4	298,8	-0,12	-0,76	-0,35	-0,29	-0,17
19	572,167	512,262	878,719	900,685	300,8	300,9	299,4	298,8	-0,12	-0,75	-0,34	-0,29	-0,17
20	580,313	520,679	870,853	892,236	300,7	300,9	299,5	298,8	-0,12	-0,73	-0,33	-0,29	-0,17
21	588,063	529,127	863,154	883,939	300,7	301	299,5	298,9	-0,12	-0,72	-0,33	-0,29	-0,16
22	595,443	537,577	855,628	875,799	300,7	301	299,5	298,9	-0,12	-0,71	-0,32	-0,29	-0,16
23	602,477	546	848,282	867,819	300,6	300,9	299,5	298,9	-0,12	-0,69	-0,32	-0,29	-0,16
24	609,188	554,372	841,119	860,003	300,6	300,9	299,5	298,9	-0,12	-0,67	-0,31	-0,29	-0,15
25	615,596	562,67	834,142	852,356	300,6	300,9	299,5	298,9	-0,12	-0,66	-0,3	-0,29	-0,15
26	621,72	570,872	827,355	844,883	300,5	300,9	299,5	299	-0,12	-0,64	-0,3	-0,28	-0,15
27	627,578	578,957	820,759	837,586	300,5	300,9	299,5	299	-0,12	-0,63	-0,29	-0,28	-0,14
28	633,185	586,908	814,355	830,47	300,5	300,9	299,6	299	-0,12	-0,61	-0,28	-0,27	-0,14
29	638,557	594,706	808,146	823,539	300,5	300,9	299,6	299	-0,11	-0,59	-0,27	-0,27	-0,14
30	643,707	602,337	802,132	816,796	300,5	300,9	299,6	299,1	-0,11	-0,58	-0,27	-0,26	-0,13
31	648,646	609,785	796,316	810,245	300,4	300,8	299,6	299,1	-0,11	-0,56	-0,26	-0,26	-0,13
32	653,385	617,038	790,698	803,889	300,4	300,8	299,6	299,1	-0,11	-0,54	-0,25	-0,25	-0,13
33	657,932	624,084	785,28	797,732	300,4	300,8	299,6	299,1	-0,1	-0,52	-0,24	-0,24	-0,12
34	662,295	630,912	780,063	791,776	300,4	300,8	299,6	299,2	-0,1	-0,5	-0,23	-0,23	-0,12
35	666,481	637,513	775,048	786,025	300,4	300,8	299,7	299,2	-0,1	-0,49	-0,23	-0,23	-0,11
...													
44	696,672	685,585	739,147	744	300,2	300,5	299,8	299,5	-0,06	-0,31	-0,15	-0,15	-0,07
45	699,206	689,589	736,195	740,466	300,2	300,5	299,8	299,5	-0,06	-0,29	-0,14	-0,14	-0,07
46	701,574	693,316	733,451	737,169	300,2	300,4	299,8	299,5	-0,05	-0,27	-0,13	-0,13	-0,06
47	703,772	696,765	730,915	734,111	300,2	300,4	299,8	299,6	-0,05	-0,25	-0,12	-0,12	-0,06
47,11	703,993	697,111	730,661	733,804	300,2	300,4	299,8	299,6	-0,05	-0,25	-0,12	-0,12	-0,06

Результаты расчёта для рассматриваемого варианта сведены в таблицу 2 и 3.

Графики результатов представлены на рис. 3-5.

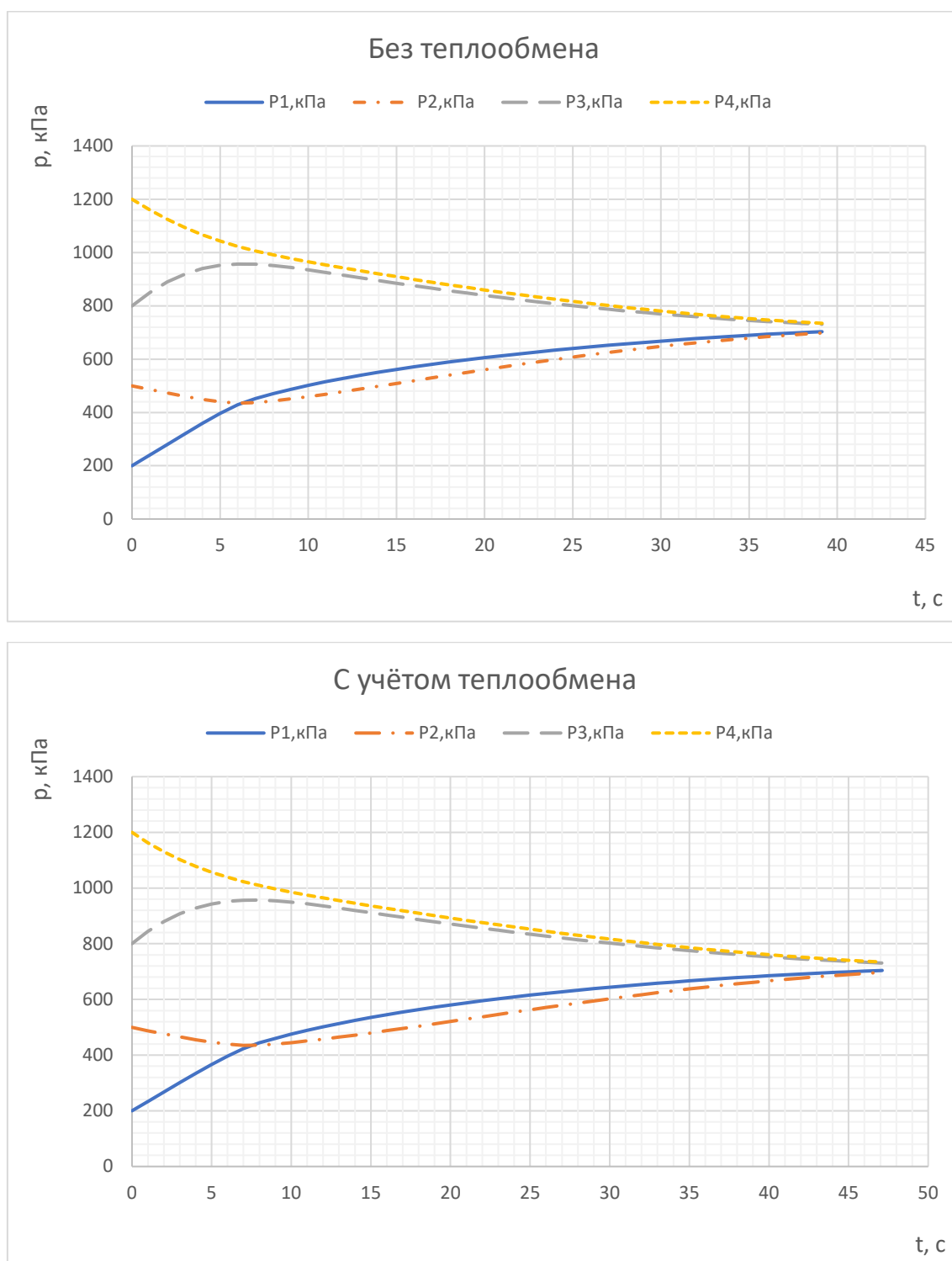


Рис. 3. График зависимости давления в сосудах от времени

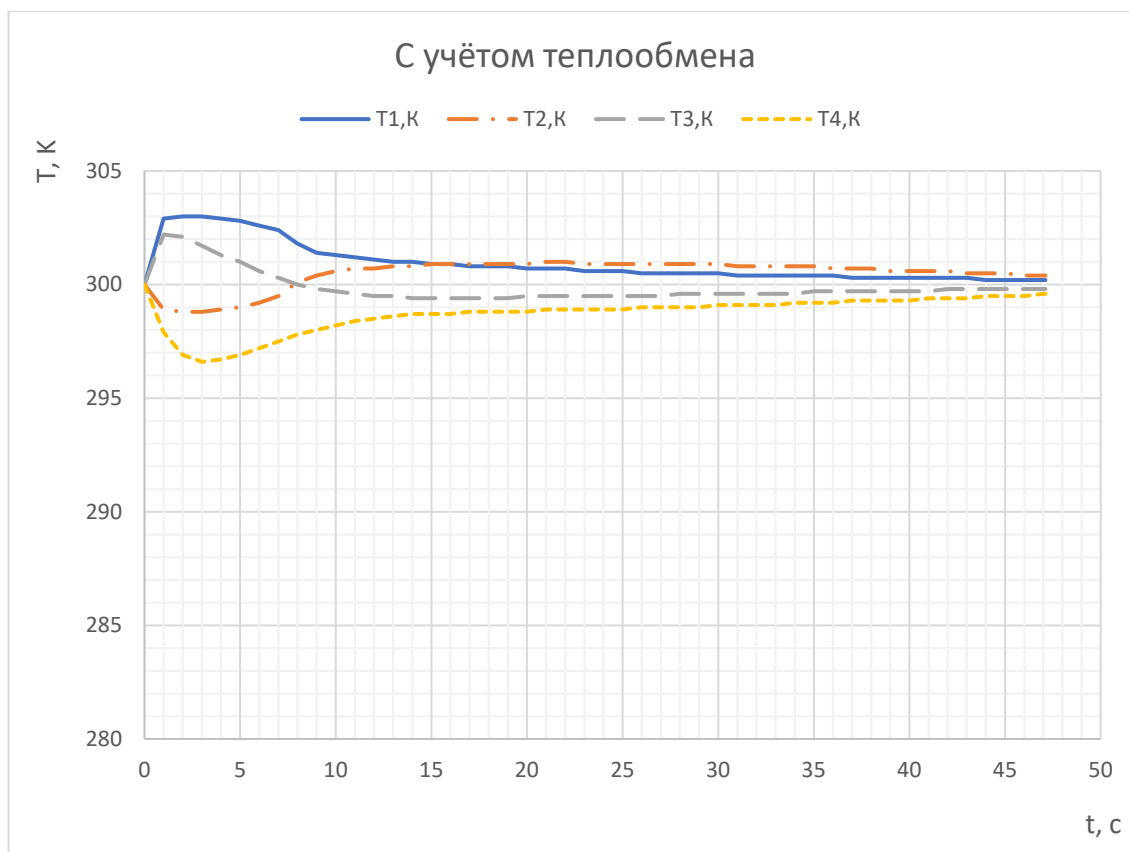
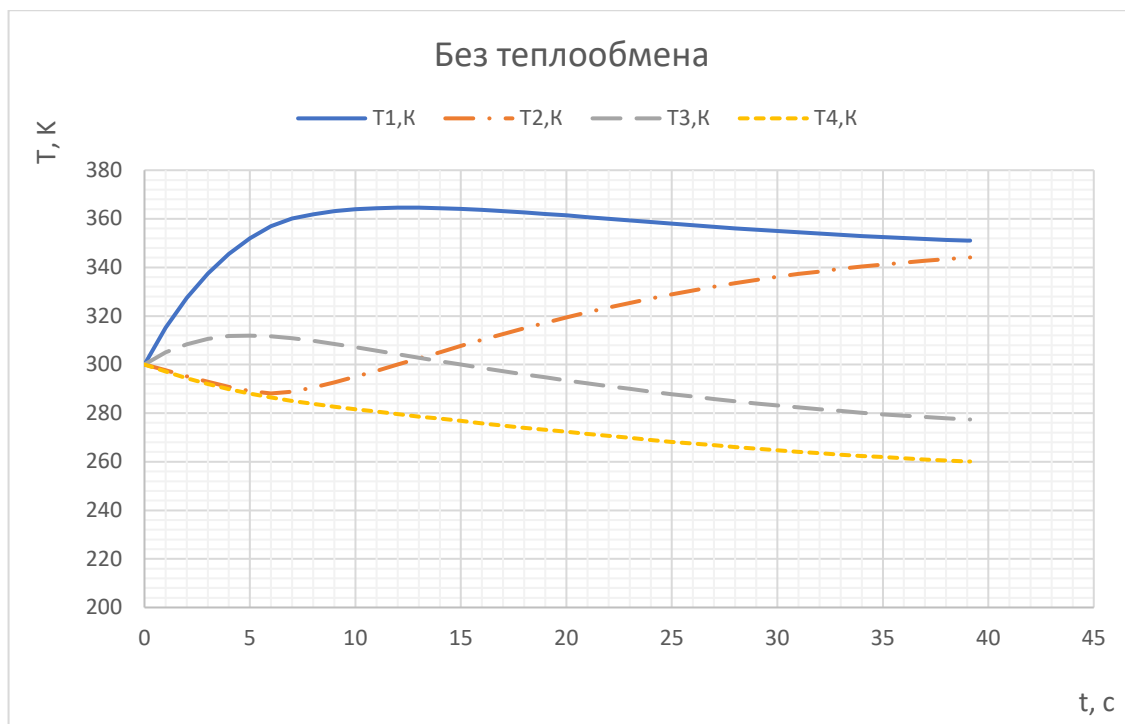


Рис. 4. График зависимости температуры в сосудах от времени

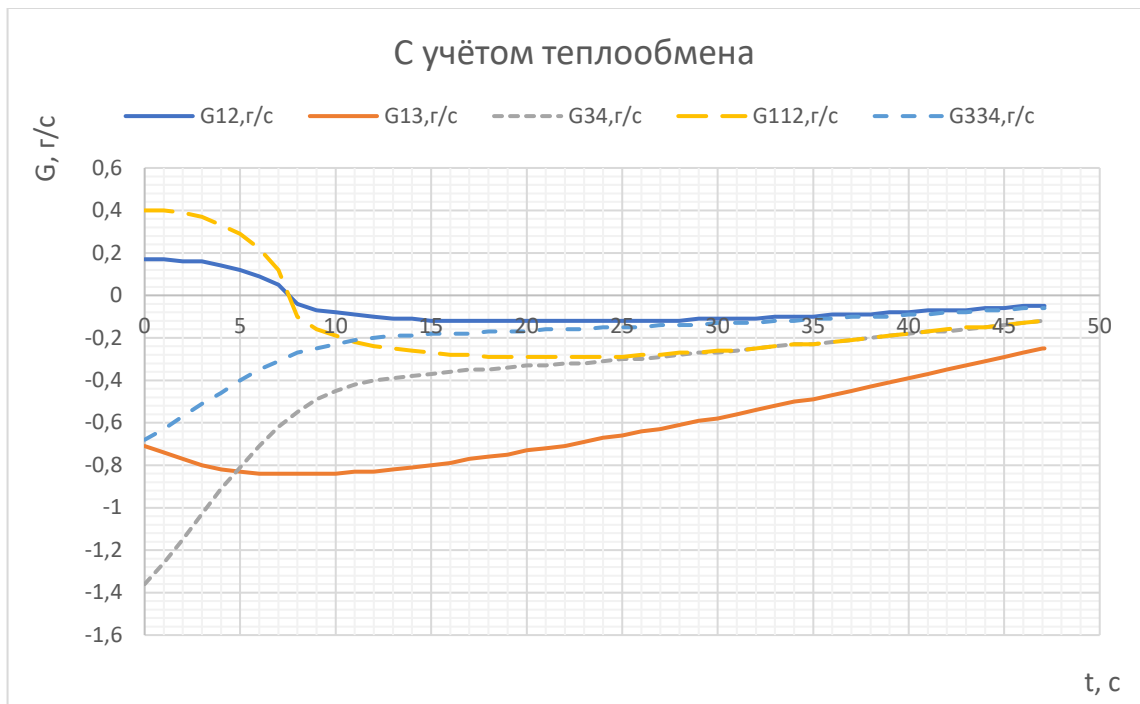
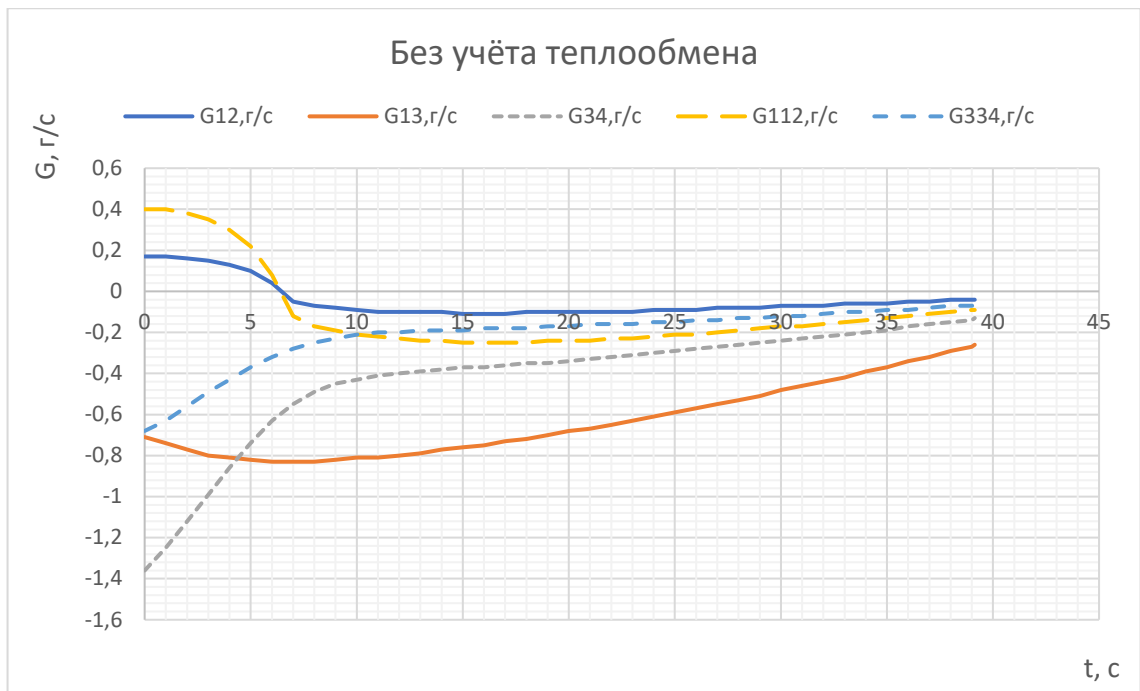


Рис. 5. График зависимости расходов через условные сопротивления от времени

Анализ полученных результатов

Таким образом, можно сделать вывод о том, что время процесса распределения газа в сосудах пневмосистемы, при котором минимальное и максимальное давление в сосудах отличается не более, чем на 5%, составляет 39,14 с без учёта теплообмена и 47,11 с учётом теплообмена.

Отсутствие учёта теплообмена между рабочим телом и корпусными элементами отдаляет картину протекающих процессов от реальности (так, минимальная температура газа в сосуде достигла значения 260 К). В связи с чем данный теплообмен был учтён, и температура не опустилась ни в одном сосуде ниже 296 К.