

Министерство науки и высшего образования Российской Федерации  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных технологий

Кафедра программного обеспечения вычислительной техники и  
автоматизированных систем

**Отчёт**  
по расчётно-графической работе

по дисциплине «Основы теории управления»

**Компьютерное моделирование систем управления**  
ОГУ 09.03.01.4023.837 О

Руководитель  
кандидат техн. наук, профессор  
\_\_\_\_\_ А.М. Семенов  
« \_\_\_\_ » \_\_\_\_\_ 2023 г.

Студент группы название  
\_\_\_\_\_ А.В. Лаптиеv  
« \_\_\_\_ » \_\_\_\_\_ 2023 г.

Оренбург 2023

«УТВЕРЖДАЮ»  
Заведующий кафедрой  
Д.В. Горбачев  
«\_\_» \_\_\_\_\_ 2023 г.

## ЗАДАНИЕ НА РГЗ

Исполнитель Лаптиеv Артур Витальевич

Группа 20ИВТ(б)ПОВТ

Руководитель Семенов Анатолий Михайлович

Тема: «Компьютерное моделирование систем управления».

Цель работы:

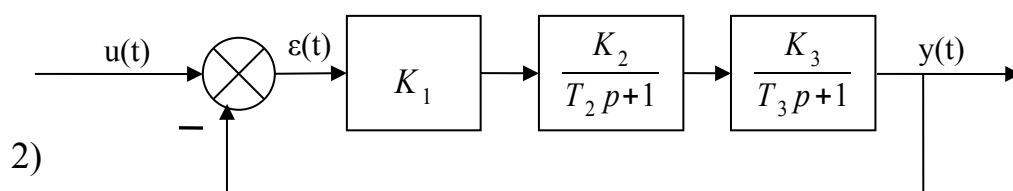
1. Освоить отдельные понятия и вопросы теории управления (регуляторы, корректирующие элементы, структурно-неустойчивые системы, жесткие и гибкие обратные связи в качестве корректирующих элементов, введение в закон регулирования производных и интегралов, применение астатических звеньев в качестве корректирующих элементов; оптимальный переходный процесс; критерий оптимальности; интегральные методы оценки качества систем).

2. Закрепить навыки работы с ПК «МВТУ»; методику определения устойчивости систем управления (СУ) с помощью временных характеристик;

3. Получить практические навыки программной реализации алгоритмов управления.

Исходные данные для работы (согласно варианту 22):

1) структурная схема исходной системы управления:



исходные параметры:

- нулевые начальные условия;
- ступенчатое входное воздействие  $U(t) = 0.8(t)$ ;
- коэффициент усиления усилителя  $K_1 = 1$ ;
- коэффициент усиления 1-го апериодического звена  $K_2 = 10$ ;
- постоянная времени 1-го апериодического звена  $T_2 = 2.5$  сек.;
- коэффициент усиления 2-го апериодического звена  $K_3 = 3$ ;
- постоянная времени 2-го апериодического звена  $T_3 = 4$  сек.;

3) требования к системе управления:

- ПД закон регулирования;
- заданное значение отклонения  $\Delta$  регулируемой величины  $Y$  не более 5 %, т. е.  $\Delta = \pm 5 \cdot 0.05 = \pm 0.25$  (при определении  $t_p$ );
- варьируемые параметры: параметры корректирующего устройства;
- нулевая ошибка  $\epsilon(t)$ ;
- время регулирования  $t_p$  не более 10 сек.;
- перерегулирование  $\sigma$  не более 30%.

Материалы, представляемые к защите:

- 1) расчетно-пояснительная записка, набранная и распечатанная в текстовом редакторе Microsoft Word;
- 2) интерфейс программы;
- 3) листинг.

Дата выдачи задания: «\_\_» \_\_\_\_\_ 2023 г.

Руководитель \_\_\_\_\_  
(подпись)

Срок сдачи законченной работы: «\_\_» \_\_\_\_\_ 2023 г.

Задание принял к исполнению: «\_\_» \_\_\_\_\_ 2023 г.

\_\_\_\_\_  
(подпись)

**Цель работы:** исследование и программная реализация типовых алгоритмов управления.

**Задачи работы:** получить практические навыки в построении цифровых моделей систем управления.

## **1 Общие сведения**

### **1.1 Основные понятия синтеза СУ**

Синтез САУ – процедура определения структуры и параметров системы по заданным показателям качества управления. Как любая обратная задача, задача синтеза в отличие от задачи анализа имеет множество решений для одних и тех же заданных условий.

Все математические задачи, решаемые в теории управления, можно объединить в два больших класса: задачи анализа; задачи синтеза [1].

Задачи синтеза можно рассматривать как обратные задачам анализа: в них требуется выбрать принцип управления, определить структуру и рассчитать параметры системы по заданным показателям качества управления. Простейшими задачами синтеза являются, например, задачи определения передаточного коэффициента разомкнутой САУ по заданной ошибке или условию минимума интегральной оценки.

Синтез является важнейшим этапом проектирования систем, основным и наиболее важным приложением результатов, полученных теорией автоматического управления, которая сегодня позволяет создать систему управления практически для любого объекта. Однако спроектировать высококачественную систему управления этим объектом можно лишь при наличии достаточно точной математической модели. Причем, согласно принципу Эшби, сложность управляющего устройства должна быть не ниже сложности объекта управления.

Цели синтеза системы управления [2]:

- создание новой системы управления на основе новых достижения науки и техники;

- совершенствование существующей системы управления на основе выявленных недостатков, а также появления новых задач и требований.

В общем виде задачи синтеза систем управления заключаются в определении структуры и параметров системы исходя из заданных требований к значениям показателей эффективности ее функционирования, а также способов обеспечения целей функционирования системы [4].

Простейшими задачами синтеза являются, например, задачи определения передаточного коэффициента разомкнутой СУ по заданной ошибке или условию минимума интегральной оценки.

Синтез представляет собой многошаговый итеративный процесс, включающий последовательное решение следующих основных задач [9]:

- формирование замысла и цели создания системы управления;

- формирование вариантов новой системы;

- приведение описаний вариантов системы во взаимное соответствие;

- оценка эффективности вариантов и принятия решения о выборе варианта новой системы;
- разработка требований к системе управления;
- разработка программ реализации требований к системе управления;
- реализация разработанных требований к системе управления.

В общем случае при проектировании системы необходимо определить алгоритмическую и функциональную структуры системы, т. е. решить задачу полного синтеза. Определение алгоритмической структуры (теоретический синтез) производится с помощью математических методов и на основании требований, записанных в четкой математической форме.[9]

Определение функциональной структуры (технический синтез) заключается в выборе конкретных физических элементов и согласования их между собой по статическим и энергетическим характеристикам. Эта процедура не имеет пока строгой математической основы (т. е. не формализована) и поэтому относится к области инженерного творчества.

Так как не всякий элемент полученной в результате расчета алгоритмической структуры может иметь отображение в виде физического блока функциональной структуры (не может быть физически реализован), задачу синтеза в большинстве случаев невозможно решить, определяя сначала алгоритмическую структуру САУ, а затем по ней – функциональную структуру. Поэтому задачу синтеза в большинстве случаев решают следующим образом.

Сначала, исходя из известности объекта управления ОУ, требований к назначению и условиям работы СУ, по каталогам серийного оборудования выбирают функционально необходимые элементы системы: регулирующий орган РО; исполнительное устройство ИУ; датчики Д.

Эти элементы СУ вместе с объектом управления ОУ представляют неизменяемую часть функциональной структуры системы (рисунок 1).

Затем, на основании требований к статическим и динамическим свойствам СУ определяют изменяемую часть функциональной структуры системы, в которую входят: усилительно-преобразующий блок УПБ; корректирующие устройства КУ[12].

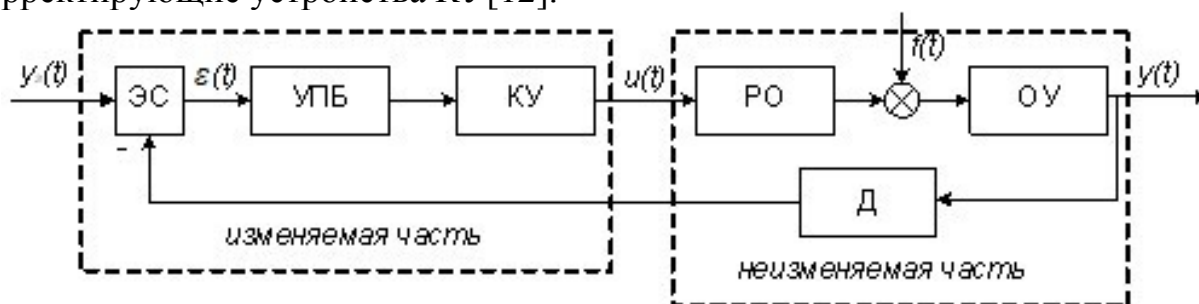


Рисунок 1 – Функциональная структура системы

Таким образом, процедуры определения алгоритмической и функциональной структур тесно переплетаются друг с другом. Окончательное решение о структуре СУ принимается на основе компромисса между качеством управления, с одной стороны, и простотой и надежностью, с другой. Заключительным этапом проектирования СУ является параметрическая

оптимизация - определение настроечных параметров выбранного регулятора. После решения задачи синтеза обычно выполняют анализ синтезированной системы, т. е. проверяют, обладает ли система необходимыми показателями устойчивости и качества управления. Применение на всех этапах синтеза и анализа СУ цифровых вычислительных машин позволяет рассмотреть большое количество вариантов структур и параметров и тем самым существенно ускорить решение задачи синтеза [16].

### 1.2 Общие принципы синтеза алгоритмической структуры СУ

Для решения задачи синтеза, например, алгоритмической структуры типовой одноконтурной СУ должны быть известны:

- передаточная функция объекта управления  $W_o(p)$ ;
- возмущение, действующее на выходе объекта  $f(p)$  рисунок 2.

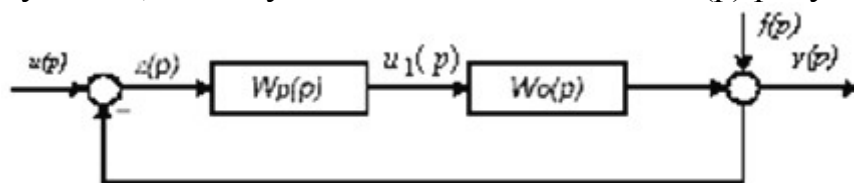


Рисунок 2 – Алгоритмическая структура типовой одноконтурной СУ

Результатом решения задачи синтеза алгоритмической структуры должна быть передаточная функция регулятора  $W_p(p)$ .

#### **Пропорциональный алгоритм регулирования (П-алгоритм)**

Зачастую при создании СУ на первом этапе проектирования в алгоритм ее функционирования закладывают простейший закон регулирования, согласно которому связь между сигналом рассогласования  $\varepsilon(p)$  и сигналом  $u_1(p)$  следующая:

$$u_1 = k_n \varepsilon, \quad (1)$$

где  $k_n$  – передаточный коэффициент.

Такой закон регулирования называют **пропорциональным** (П-закон регулирования). Ему соответствует передаточная функция:

$$W_p(p) = k_n. \quad (2)$$

Это простейший алгоритм, который реализуется при помощи безинерционного звена. Преимущества П-регулятора – простота и быстродействие. Недостатки – ограниченная точность (особенно при управлении объектами с большой инерционностью и запаздыванием).

#### **Интегральный алгоритм регулирования (И-алгоритм)**

Этот алгоритм реализуется при помощи интегрального звена с передаточной функцией:

$$W_u(p) = W_p(p) = \frac{1}{T_u p},$$

(3)

где  $T_u$  – постоянная времени,  $k_u = \frac{1}{T_u}$  – коэффициент передачи звена.

Тогда в закон регулирования вводят интеграл от сигнала

рассогласования:

$$u_1 = \frac{1}{T_u} \int_0^t \varepsilon dt. \quad (4)$$

При интегральном алгоритме регулирования управляющее воздействие  $u_1$  в каждый момент времени пропорционально интегралу от сигнала ошибки. Преимущества И-регулятора - лучшая (по сравнению с П-регулятором) точность в установившихся режимах, недостатки - худшие свойства в переходных режимах (меньшее быстродействие и более высокая колебательность).

### **Пропорционально-интегральный алгоритм регулирования (ПИ-алгоритм)**

Этот алгоритм реализуется при помощи регулятора с передаточной функцией:

$$W_{mu}(p) = W_p(p) = k_n(p) + \frac{k_u}{p} = \frac{k_n p + k_u}{p}, \quad (5)$$

где  $k_n, k_u$  – передаточные коэффициенты.

В этом случае закон регулирования, называемый пропорционально-интегральным (ПИ-законом регулирования), будет описываться следующим уравнением:

$$u_1 = k_n \varepsilon + k_u \int_0^t \varepsilon dt. \quad (6)$$

Введение интеграла в закон регулирования исключает статическую ошибку СУ, превращая ее в астатическую систему, но при этом одновременно уменьшаются запас устойчивости и быстродействие системы.

Благодаря наличию интегральной составляющей в алгоритме, ПИ-регулятор обеспечивает высокую точность в установившихся режимах, а при определенном соотношении коэффициентов  $k_n, k_u$  обеспечивает хорошие показатели и в переходных режимах. Поэтому он получил наибольшее распространение в промышленной автоматике.

### **Пропорционально-дифференциальный алгоритм регулирования (ПД-алгоритм)**

Этот алгоритм реализуется при помощи пропорционального регулятора и введением в закон регулирования производной от сигнала рассогласования. Передаточная функция алгоритма:

$$W_{nd}(p) = W_p(p) = (k_d p + k_n) = k \left( \frac{k_d}{k_n} p + 1 \right), \quad (7)$$

где  $k_d = T_d$  – коэффициент дифференцирующего звена.

Закон регулирования имеет вид:

$$u_1 = k_n \varepsilon + k_d \frac{d\varepsilon}{dt}. \quad (8)$$

Регулятор, реализующий ПД-закон регулирования, реагирует не только на значение отклонения регулируемой величины в данный момент времени, но и на скорость изменения отклонения. Следовательно, регулятор работает с опережением, улучшая качество переходного процесса за счет учета тенденции

последующего его развития, т. е. увеличивает быстродействие системы и запас устойчивости. Недостатком пропорционально – дифференциального закона регулирования является ограниченная точность. [6-10].

### **Пропорционально-интегрально-дифференциальный алгоритм регулирования (ПИД-алгоритм)**

Этот алгоритм реализуется при помощи регулятора с передаточной функцией:

$$W_p(p) = (k_n + \frac{k_u}{p} + k_d p), \quad (9)$$

где  $k_d, k_n, k_u$  – постоянные коэффициенты.

ПИД-закон регулирования, реализует алгоритм:

$$u_1 = k_n \varepsilon + k_d \frac{d}{dt} \varepsilon + k_u \int \varepsilon dt. \quad (10)$$

Совместное введение производной и интеграла обеспечивает желаемое быстродействие, необходимый запас устойчивости и отсутствие статической ошибки СУ. ПИД-алгоритм – наиболее гибкий алгоритм регулирования (в классе линейных алгоритмов). Он сочетает в себе преимущества более простых выше рассмотренных алгоритмов. Коэффициенты  $k_d, k_n, k_u$  входящие в передаточные функции типовых регуляторов, подлежат настройке при наладке СУ и поэтому называются настроечными параметрами. Они имеют наименования: – коэффициенты пропорциональной, интегральной и дифференциальной частей СУ.

## **2 Моделирование СУ на ЭВМ**

Моделирование СУ на ЭВМ базируется на цифровых моделях типовых динамических звеньев (ТДЗ). Основу цифрового моделирования представляет преобразование дифференциального уравнения (ДУ) в разностное. Одним из самых распространенных методов решения дифференциального уравнения является метод Эйлера.

### **2.1 Сущность метода Эйлера**

ДУ представляется в форме Коши:

$$\frac{dy}{dt} = f(t, y, x), \quad (11)$$

где  $y$  – искомая функция;  $u$  – задающее воздействие;  $t$  – независимая переменная (время).

Сущность метода заключается в том, что если известно значение функции в какой-то  $(k-1)$ -й момент, то пользуясь рекуррентным соотношением, можно рассчитать значение функции в  $k$ -й момент (рисунок 5).



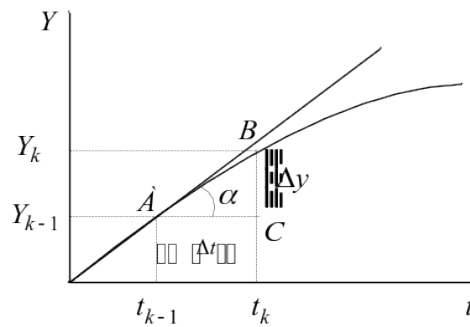


Рисунок 5 – Графическая интерпретация метода

Как видно из рисунка 5:

$$Y_k = Y_{k-1} + \Delta Y \quad (12)$$

Из (12) следует, что для нахождения  $Y_k$  при известном  $Y_{k-1}$ , необходимо определить  $\Delta Y$ . Как следует из решения треугольника ABC:

$$\Delta Y \approx \Delta t \cdot \operatorname{tg} \alpha, \quad (13)$$

где  $t$  – шаг решения;

$\alpha$  – угол наклона касательной к графику  $Y(t)$  в точке  $t_{k-1}$ .

Из определения производной следует, что производная – это тангенс угла наклона касательной к графику функции:

$$\operatorname{tg} \alpha = \frac{dY}{dt}. \quad (14)$$

Как следует из (13,14) можно записать:

$$\operatorname{tg} \alpha = f(t_{k-1}, Y_{k-1}, U_{k-1}). \quad (15)$$

Подставив (15) в (13) получим:

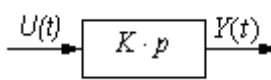
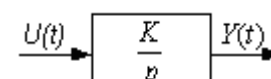
$$\Delta Y \approx \Delta t \cdot f(t_{k-1}, Y_{k-1}, U_{k-1}). \quad (16)$$

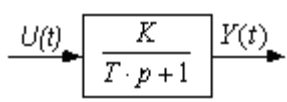
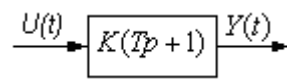
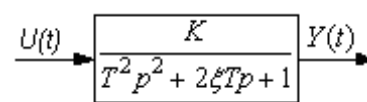
Тогда уравнение Эйлера для решения ДУ в общем виде запишется следующим образом:

$$Y_k = Y_{k-1} + \Delta t \cdot f(t_{k-1}, Y_{k-1}, U_{k-1}). \quad (17)$$

В таблице 2 приведены цифровые модели (ЦМ) ТДЗ.

Таблица 2 – Цифровые модели ТДЗ

Операторный коэффициент передачи звена	Дифференциальное уравнение	Цифровая модель
Дифференцирующее звено 	$Y(t) = K \frac{dU}{dt}$	$Y_k = \frac{K}{\Delta t} (U_{k+1} - U_k)$
Интегрирующее звено 	$\frac{dY}{dt} = K \cdot U(t)$	$Y_k = Y_{k-1} + \Delta t \cdot K \cdot U_{k-1}$

<p>Апериодическое звено</p> 	$T \frac{dY}{dt} + Y(t) = K \cdot U(t)$	$Y_k = Y_{k-1} + \Delta t \left( \frac{K}{T} U_{k-1} - \frac{Y_{k-1}}{T} \right)$
<p>Форсирующее звено</p> 	$Y(t) = K \cdot \left[ T \frac{dU}{dt} + U(t) \right]$	$Y_k = \frac{KT}{\Delta t} U_{k+1} + K \left( 1 - \frac{T}{\Delta t} \right) U_k$
<p>Колебательное звено</p> 	$T^2 \frac{d^2 Y}{dt^2} + 2\xi T \frac{dY}{dt} + Y(t) = K \cdot U(t)$	$Z_k = Z_{k-1} + \Delta t \cdot \left( \frac{K}{T^2} U_{k-1} - \frac{2Z_{k-1}}{T} + \frac{Y_{k-1}}{T^2} \right)$ $Y_k = Y_{k-1} + \Delta t \cdot Z_{k-1}$

## 2.2 Компьютерная методика моделирования СУ

Цифровое моделирование включает в себя следующие этапы:

1. Определение ЦМ элементов, входящих в состав СУ.
2. Составление блок-схемы алгоритма ЦМ СУ.
3. Написание программы.
4. Отладка программы.
5. Исследование системы по ЦМ.

Согласно изложенной методике моделирования, используя таблицу 2, определим ЦМ элементов, входящих в систему:

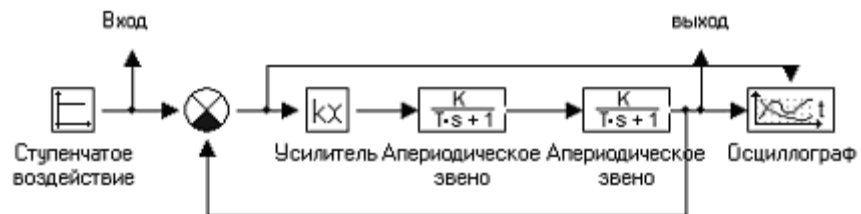


Рисунок 6 – схема объекта управления

Таблица 3 – Цифровые модели ТДЗ входящие в схему

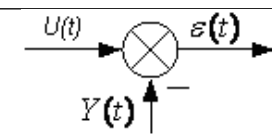
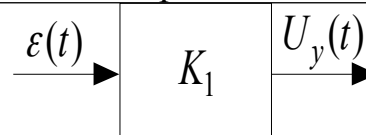
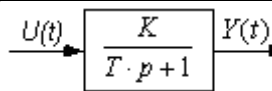
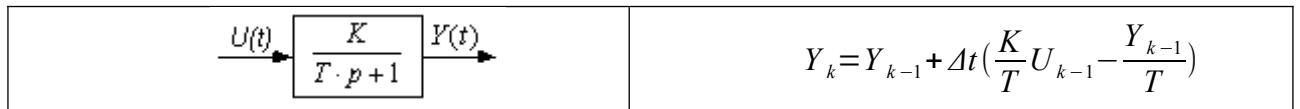
Элемент	ЦМ
	$\varepsilon_{k-1} = U_{k-1} - Y_{k-1}$

Таблица 3 – Продолжение

	$U_{yk-1} = K_1 \cdot \varepsilon_{k-1}$
	$Y_k = Y_{k-1} + \Delta t \left( \frac{K}{T} U_{k-1} - \frac{Y_{k-1}}{T} \right)$



Эти элементы являются объектом управления, добавим к нему ПИ регулятор. Рисунок 6

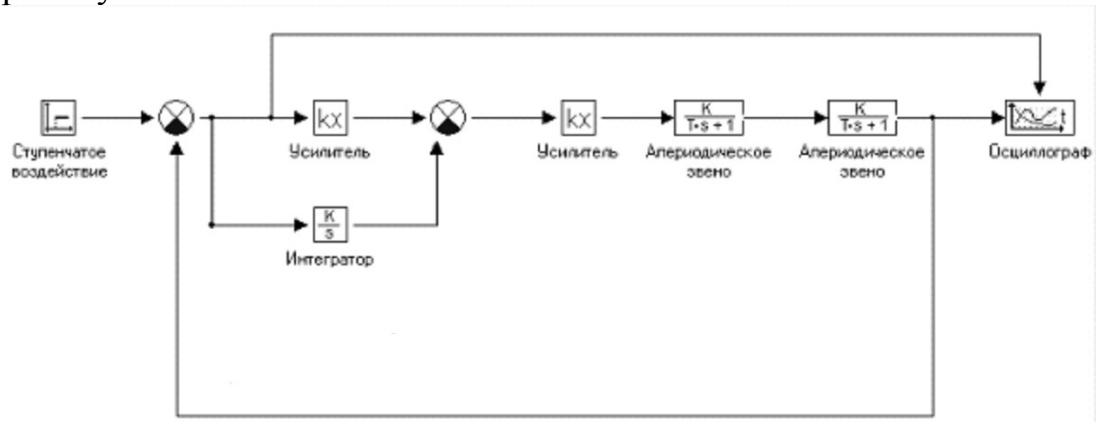


Рисунок 6 - Схема СУ с ПИ регулятором.

Контроллер	$k_n$	$T_u$	$T_o$
П	$k_n/2$	—	—
ПИ	$k_n/2, 2$	$P/1, 2$	—
ПИД	$k_n/1, 7$	$P/2$	$P/8$

$$k_n = \frac{4}{2} = 2; k_o = 0, 4;$$

Блок-схема алгоритма цифрового моделирования СУ представлена на рисунке 7.

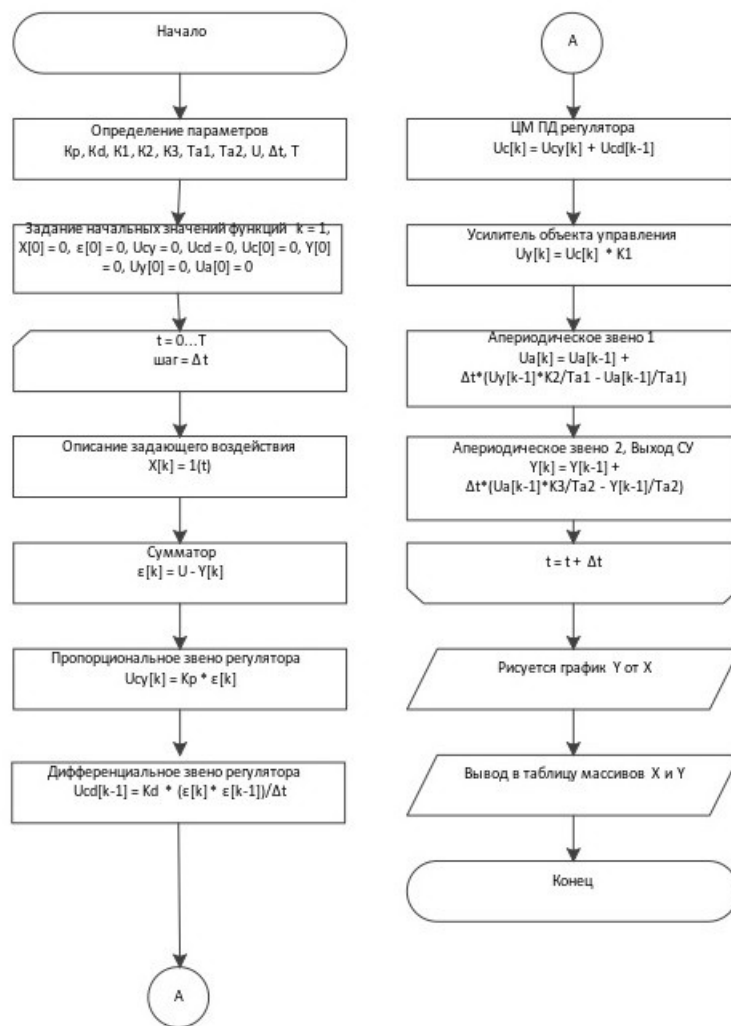


Рисунок 7 - Блок-схема алгоритма цифрового моделирования СУ с ПД регулятором

На рисунке 8 представлен интерфейс программы, позволяющий анализировать влияние параметров СУ на качество процесса управления с параметрами, заданными по варианту и подобранными значениями регулятора.

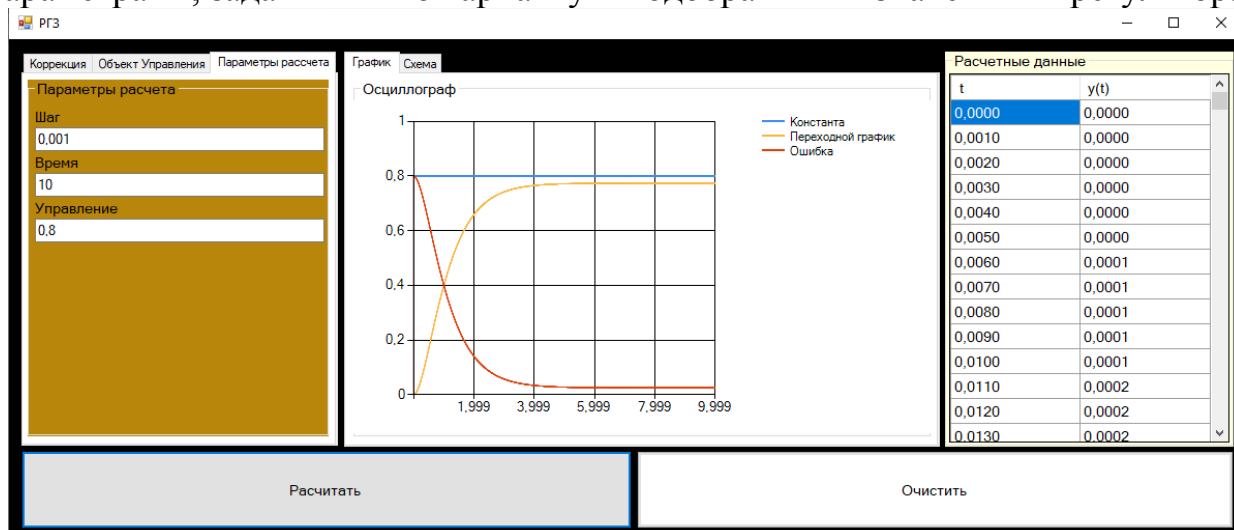


Рисунок 8-интерфейс программы анализа СУ.

На рисунке 9 представлен график, полученный в программном комплексе “МВТУ” с исходными параметрами и параметрами управления, в точности соответствующими установленным в программном средстве. Из графика можно сделать вывод, что результаты программного средства полностью соответствуют результатам “ПК МВТУ”. Время регулирования составляет  $\approx 9$  секунд, что соответствует допустимому результату по варианту.

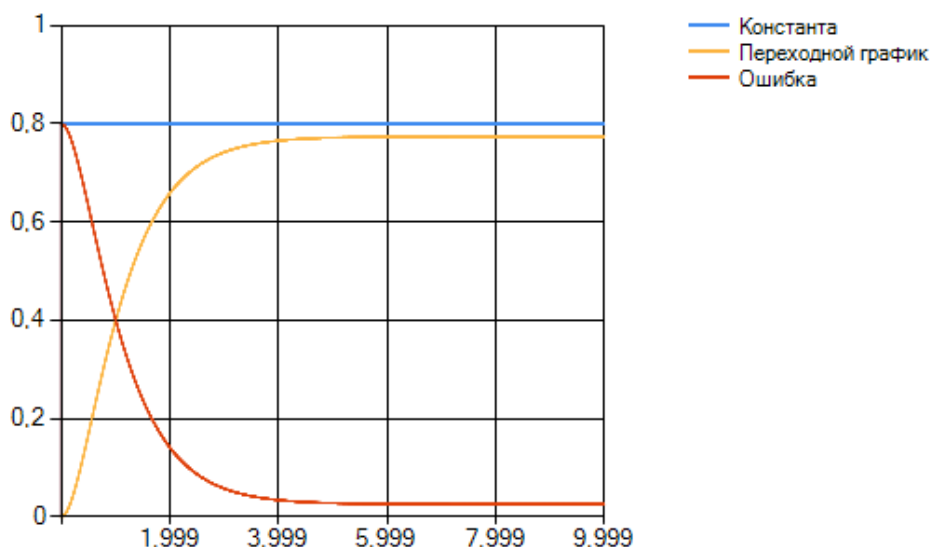


Рисунок 9 – график “ПК МВТУ”

## Вывод

В ходе выполнения расчётно-графического задания были освоены отдельные понятия и вопросы теории управления (регуляторы, корректирующие элементы, структурно-неустойчивые системы, жесткие и гибкие обратные связи в качестве корректирующих элементов, введение в закон регулирования производных и интегралов, применение астатических звеньев в качестве корректирующих элементов; оптимальный переходный процесс; критерий оптимальности; интегральные методы оценки качества систем).

Получены практические навыки программной реализации алгоритмов управления в том числе навыки работы с регуляторами.

В ходе экспериментальных исследований можно сформулировать следующие выводы влияния настроечных параметров на показатели качества регулирования:

1. Увеличение коэффициента  $k_n$  пропорциональной части регулятора приводит к увеличению перерегулирования  $\sigma$ , времени  $t_p$  переходного процесса и уменьшению степени затухания.

2. Увеличение коэффициента  $k_d$  дифференциальной части регулятора приводит к уменьшению времени  $t_p$  переходного процесса и уменьшению перерегулирования.

В пропорционально-дифференцируемом алгоритме регулирования (ПД-алгоритм), регулятор работает с опережением, улучшая качество переходного процесса за счет учета тенденции последующего его развития, т. е. увеличивает быстродействие системы и запас устойчивости. Недостатком пропорционально – дифференциального закона регулирования является ограниченная точность.

## Список использованных источников

- 1 Семенов, А. М. Электронное гиперссылочное учебное пособие по дисциплине «Основы теории управления» / А. М. Семенов, И. Б. Крылов. – Оренбург. гос. ун-т, Университетский фонд алгоритмов и программ. – № 521 ; зарегистр. 06.11.09. Мирошник, И. В. Теория автоматического управления. Линейные системы. Учебное пособие для вузов / И.В. Мирошник. – СПб.: Питер, 2005. – 336с.
- 2 Семенов, В. В. Математическая теория управления в примерах и задачах: Учебное пособие для студентов вузов / В. В. Семенов, А. В. Пантелеев, А. С. Бортаковский. – М. : Изд-во МАИ, 1997. – 261 с.
- 3 Семенов, А. М. Исследование типовой САУ. Методические рекомендации по выполнению курсовой работы / А. М. Семенов. – Оренбург: ФВУ, 1999. – 34 с.
- 4 Семенов, А. М. Работоспособность САУ: Учебное пособие / А. М. Семенов, В. В. Паничев. – Оренбург : ФВУ, 2002. – 70 с.
- 5 «МВТУ» – программный комплекс для моделирования и исследования систем и объектов. – Режим доступа: сайт <http://energy.power.bmstu.ru/mvtu/>
- 6 Дядик, В. Ф. Теория автоматического управления: учебное пособие / В. Ф. Дядик, С. А. Байдали, Н. С. Криницын. – Томск: Изд-во Томского политехнического университета, 2011. – 196 с.
- 7 Пупков, К. А. Методы классической и современной теории автоматического управления : учебник. В 5 т. Т. 1. Математические модели, динамические характеристики и анализ систем автоматического управления / К. А. Пупков, Н. Д. Егупов. – 2-е изд., перераб. и доп. – М. : МГТУ им. Н.Э. Баумана, 2004. – 656 с.
- 8 Цыпкин, Я. З. Основы теории автоматических систем : учеб. пособие для вузов / Я. З. Цыпкин. – М. : Наука, 1977. – 559 с.
- 9 Востриков, А. С. Теория автоматического регулирования : учеб. пособие для вузов / А. С. Востриков, Г. А. Французова. – М. : Высш. шк., 2004. – 365 с.
- 10 Анхимюк, В. Л. Теория автоматического управления: Учеб. пособие для вузов / В. Л. Анхимюк, О. Ф. Опейко, Н. Н. Михеев. – 2-е изд., испр. – Минск: Дизайн ПРО, 2002. – 352 с.
- 11 Юревич, Е. И. Теория автоматического управления / Е. И. Юревич. – СПб. : ВHV, 2016. – 560 с.
- 12 Бесекерский, В.А. Теория систем автоматического управления / В. А. Бесекерский, Е. П. Попов. - 4-е изд., перераб. и доп.- СПб.: Профессия, 2004. – 752 с.
- 13 Куропаткин, П. В. Теория автоматического управления: Учеб. пособие для вузов / П. В. Куропаткин ; под ред. Д. В. Васильева. – М. : Высш. школа, 1973. – 528 с.
- 14 Теория автоматического управления : Учеб. пособие для вузов / Под ред. А.С. Щаталова. – М.: Высш. шк., 1977. – 448 с.: ил.

## Приложение А

*(обязательное)*

### Код программы

```
try
    {
        {
            double kp = Convert.ToDouble(KP.Text),
                kd = Convert.ToDouble(KD.Text),
                k1 = Convert.ToDouble(K1.Text),
                k2 = Convert.ToDouble(K2.Text),
                k3 = Convert.ToDouble(K3.Text),
                t1 = Convert.ToDouble(T1.Text),
                t2 = Convert.ToDouble(T2.Text),
                step = Convert.ToDouble(Step.Text),
                u = Convert.ToDouble(U.Text),
                out_step = 0.01;
            int st = 1,
                time = Convert.ToInt32(Convert.ToDouble(Time.Text) / Convert.ToDouble(Step.Text));
            List<double> x = new List<double>() { 0 };
            List<double> un = new List<double>() { 0 };
            double
                cd_prev = 0,
                cd_next = 0;
            List<double> uc = new List<double>() { 0 };
            List<double> uy = new List<double>() { 0 };
            List<double> ua1 = new List<double>() { 0 };
            List<double> ua2 = new List<double>() { 0 };
            List<double> y = new List<double>() { 0 };
            List<double> err = new List<double>() { 0 };
            int k = 1;
            for (int t = 0 + st; t <= time; t += st)
            {
                x.Add(t * step);
                un.Add(u);
                err.Add(un[k] - y[k - 1]);
                if (k > 1)
                    cd_next = -(y[k - 1] - y[k - 2]) / step;
                else
                    cd_next = -y[k - 1] / step;
                uc.Add(err[k] * kp +
                    cd_prev * kd
                );
                uy.Add(uc[k] * k1);
                ua1.Add(ua1[k - 1] + step * (k2 / t1 * uy[k] - ua1[k - 1]
                / t1));
                ua2.Add(ua2[k - 1] + step * (k3 / t2 * ua1[k] - ua2[k - 1]
                / t2));
            }
        }
    }
}
```



```
        y.Add(ua2[k]);
        k++;
        cd_prev = cd_next;
    }
    chart_graph("Константа", x, un, SeriesChartType.Line);
    chart_graph("Переходной график", x, y, SeriesChartType.Line);
    chart_graph("Ошибка", x, err, SeriesChartType.Line);
    int tab_step = 1;
    if (st < out_step)
        tab_step = Convert.ToInt32(st / out_step);
    Tab.RowCount = time / tab_step + 1;
    for (int i = 0; i <= time; i += tab_step)
    {
        Tab[0, i / tab_step].Value = x[i];
        Tab[1, i / tab_step].Value = y[i];
    }
}
catch (Exception error)
{
    MessageBox.Show($"Ошибка: {error.Message}");
}
```