

Федеральное государственное автономное
образовательное учреждение
высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра «Системы автоматики, автоматизированное управление и
проектирование»

УТВЕРЖДАЮ
Заведующий кафедрой
_____ С.В. Ченцов

« ____ » _____ 2020 г.

БАКАЛАВРСКАЯ РАБОТА

27.03.04 – Управление в технических системах

ПОСТРОЕНИЕ СИСТЕМЫ УПРАВЛЕНИЯ КУРСОМ САМОЛЕТА В СРЕДЕ МОДЕЛИРОВАНИЯ SIMINTECH

Руководитель _____ «____» 06. 2020 г. доц., канд.техн.наук
А.В. Чубарь

Выпускник _____ «____» 06. 2020 г. Ю.А. Мацкевич

Нормоконтролер _____ «____» 06. 2020 г. Т.А. Грудинова

Красноярск 2020

РЕФЕРАТ

Выпускная квалификационная работа по теме «Построение системы управления курсом самолета в среде моделирования SimInTech» содержит 49 страницы текстового документа, 31 иллюстраций, 18 использованных источников.

УПРАВЛЯЕМОСТЬ, ЛЕТАТЕЛЬНЫЙ АППАРАТ, SIMINTECH

Развитие летательных аппаратов предъявляет возрастающие требования к процессу управления, обеспечивающему высокую безопасность и регулярность полетов.

Цель работы заключается в: построении системы управления курсом самолета с использованием компьютерной среды моделирования.

Для достижения поставленной цели решены следующие задачи:

- 1 Реализовать 3d модель визуализацию самолёта в среде разработки;
- 2 Построить систему ручного управления скоростью самолёта;
- 3 Построить систему ручного управления углом тангенса самолета;
- 4 Построить систему ручного управления крена самолета.
- 5 Построить систему автоматической стабилизации положения самолета в пространстве.

В результате выполнения выпускной квалификационной работы проведена оптимизация параметром, построена система ручного управления скоростью самолета, углом тангенса и крена, а также реализована 3d модель самолета на основании проведенной работы сформирована выпускная квалификационная работа.

СОДЕРЖАНИЕ

Введение.....	4
1 Обзор предметной области	6
1.1 Описание объекта автоматизации	6
1.2 Общие сведения.....	8
1.3 Силы, действующие на летательный аппарат.....	9
2 Построение системы управления курсом самолета	12
2.1 Описание метода проектирования	13
2.2 Среда моделирования	14
2.3 Оптимизация передаточной функции скорости самолета.....	15
2.4 Оптимизация передаточных функций крена и тангажа самолета.....	21
3 Формирование проекта системы автоматического регулирования	27
3.1 Создание базы данных проекта	27
3.2 Моделирование схем общего вида	28
3.3 Создание пульта управления оператора	29
3.4 Разработка 3D визуализации проекта	32
3.5 Запуск проекта.....	35
Заключение	37
Список использованных источников	38
Приложение А	40
Приложение В.....	47

ВВЕДЕНИЕ

Динамические свойства летательных аппаратов зачастую не являются удовлетворительными, а попытки усовершенствовать их путем изменения формы приводят к ухудшению аэродинамических форм [1]. Вследствие этого появляется задача усовершенствования устойчивости и управляемости летательных аппаратов средствами автоматики без ухудшения их аэродинамических характеристик.

Устойчивость и управляемость это самые важные свойствами самолета, определяющие возможность и безопасность полета [2].

Во время движения летательного аппарата на него воздействуют возмущающие силы и моменты [3,4], вызывающие перегрузки, тепловые напряжения и т. д. Поскольку прочность конструкции ограничена, то действующие на летательный аппарат возмущения также должны быть ограничены. Конечно добиться этой цели можно, если система будет противодействовать возмущениям, не допуская в то же время резких перемещений управляющих поверхностей.

Данная работа посвящена проектированию системы управления курсом и скоростью самолёта.

Актуальность данной работы представлена в том, что система автоматического регулирования скорости и угла крена самолета необходимая для измерения и индикации экипажу самолетов, соответствующих положению летательного аппарата относительно горизонтальной плоскости, называемая авиагоризонтом необходима лётчику для контроля положения в пространстве при выводе самолёта из сложного положения, при выполнении фигур пилотажа, а также при выполнении полётов по маршруту днём и ночью. Поэтому, способ индикации пространственного положения на авиагоризонте, обеспечивающего эффективность действий при выводе самолёта из сложного положения в облаках и при выполнении пилотажа в настоящее время является достаточно актуальной проблемой.

Цель работы: построить систему управления курсом самолета с использованием компьютерной среды моделирования.

Задачи:

1. Провести обзор предметной области;
2. Построить систему ручного управления скоростью самолёта;
3. Построить систему ручного управления углом тангажа самолета;
4. Построить систему ручного управления крена самолета;
5. Построить систему автоматической стабилизации положения самолета в пространстве;
6. Реализовать 3d модель визуализацию самолёта в среде разработки.

1 Обзор предметной области

1.1 Описание объекта автоматизации

В наши дни в авиационной промышленности применяются системы автоматического управления. Особенностью современных летательных аппаратов представляет значительное изменение пилотажных характеристик в эксплуатационной области их применения.

Летательный аппарат представляет собой замкнутую систему автоматического регулирования, в которой самолет — объект, а автопилот — регулятор. Регулируемая величина в процессе регулирования может изменяться по определенному закону (режим управления) или же поддерживаться постоянной (режим стабилизации). Регулятор осуществляет измерение разности между заданными и действительными значениями регулируемого параметра. И после, в соотношении от знака и величины имеющейся разницы, оказывает воздействие на объект, в итоге которого измеряемое значение параметра становится соответствующим заданному.

Управляемость самолета — это его свойство отвечать соответствующими линейными и угловыми перемещениями в пространстве на команды управления. Под отличной управляемостью подразумевают реакция летательного аппарата на перемещения рычагов управления, с приложением небольших усилий, без особо сильного запаздывания и колебательности[5].

Управляемость самолета тесно связана с устойчивостью его возмущенного движения. Летательный аппарат с большим запасом устойчивости требует не сложных движений рычагами управления. Поэтому часто, характеризуя летательный аппарат, используют термины маневренности, устойчивости и управляемости, не отделяя эти понятия друг от друга. Характеристики управляемости можно получить, используя методы теории автоматического управления.

В режиме автоматического управления самолетом субъектами

управления являются система пилотажно-навигационный комплекс и автоматического управления. Для повышения показателей системы автоматического управления работают во всех режимах. Таким образом, на сравнении положений самолета основан принцип автоматического управления положением [6]. И если появляется нестыковка, то система автоматического управления создает воздействующий сигнал на рычаги управления.

Отклонения самолета от заданного положения воспринимают устройства измерения и являются чувствительными элементами автопилота.

В качестве измерительных устройств в автопилотах широко применяются гироскопические приборы, которые реагируют на угловые отклонения самолета и их производные.

Помимо приборов в автопилотах широко используются дифференцирующие и интегрирующие устройства, а для измерения линейных отклонений, например, высоты, применяется барометрическое устройство.

Суммирующее устройство служит для алгебраического сложения сигналов, которые поступают от устройств обратной связи, измерительных устройств и датчика режима полета.

В качестве суммирующего устройства в современных автопилотах широко используются потенциометрические и мостовые схемы, а также магнитные усилители, которые, помимо суммирования входных сигналов, усиливают их.

Усилитель служит для усиления сигналов до величины, способной управлять рулевой машиной [7].

Рулевая машина является исполнительным органом автопилота и служит для привода руля самолета в соответствие с заданным законом управления.

В современных автопилотах применяются электромеханические и электрогидравлические рулевые машины.

Датчик режима полета и устройство обратной связи, в большинстве автопилотов, выполнены в виде потенциометрических датчиков, преобразующих угол отклонения руля в электрическое напряжение, и угол поворота рукоятки управления.

Широкая автоматизация процесса управления самолётом не исключает лётчика из системы управления, а оставляет за ним функции включения выключения, настройки автопилота, а также контроль процесса пилотирования самолёта. Поэтому задача разработчика заключается в рациональном распределении и сочетании в рамках единой системы управления, функций пилота и САУ [8].

1.2 Общие сведения

Как известно, типичным примером, когда используются обратные связи по нескольким переменным является система автоматического управления положением самолета. Регулирование положением самолета меняется с помощью элеронов, рулей высоты и руля направления, как показано на рисунке 1.

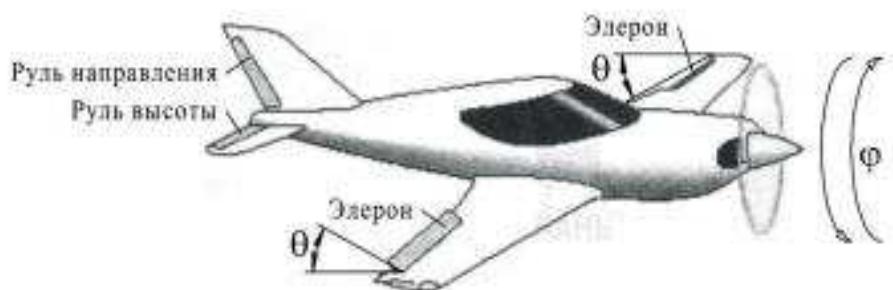


Рисунок 1 - Самолет и его органы управления

Пилот устанавливает нужную траекторию полета, манипулируя органами управления.

На рисунке видим, что данная система управления регулирует угол крена ϕ путем отклонения элеронов. А благодаря давлению воздуха на

поверхность крыльев приводит к возникновению вращательного момента за счет отклонения элеронов на угол θ . Следовательно, в связи с этим происходит вращение самолета относительно горизонтальной оси. Элероны отклоняются с помощью гидравлического исполнительного механизма, динамические свойства которого описываются передаточной функцией $1/s$.

Для автоматического регулирования скорости и угла крена самолета во время полета используется авиаоризонт — бортовой гирокомпасный прибор, который используется для определения поперечного и продольного углов наклона летательного аппарата (тангажа и крена), то есть углов ориентации относительно истинной вертикали. Летчики используют прибор для стабилизации и управлением летательным аппаратом в воздушном пространстве. [9]

Величины и направления сил и моментов, действующие на самолет, регулируются за счет отклонений органов управления — элеронов и руля направления тем самым, изменяя параметры движения самолета в желаемую сторону.

Созданное системой автоматического управления или пилотом равновесие, может быть легко нарушено воздействием какого-либо возмущающего фактора, например, турбулентностью атмосферы или порывами ветра. Поэтому требуется обеспечить устойчивость.

1.3 Силы, действующие на летательный аппарат

Существенное упрощение системы уравнений управляемого движения летательного аппарата достигается, когда удается разделить ее на три группы уравнений, описывающих движение аппарата в трех взаимно перпендикулярных плоскостях. Возможность такого разделения обусловлена наличием симметрии летательного аппарата в связанной системе координат.

Общее движение летательного аппарата складывается из продольного

бокового и вертикального движений, причем между этими движениями имеет место взаимное влияние. Продольное и вертикальное движения могут существовать самостоятельно как движения в определённой плоскости. В то время как боковое движение может существовать лишь совместно с ними [10].

Силы, действующие по трем осям в связанной системе координат обозначены как T , P , Q (рисунок 2). Земная система координат обозначена как O_3UVW , с началом в точке старта самолёта O_3 .

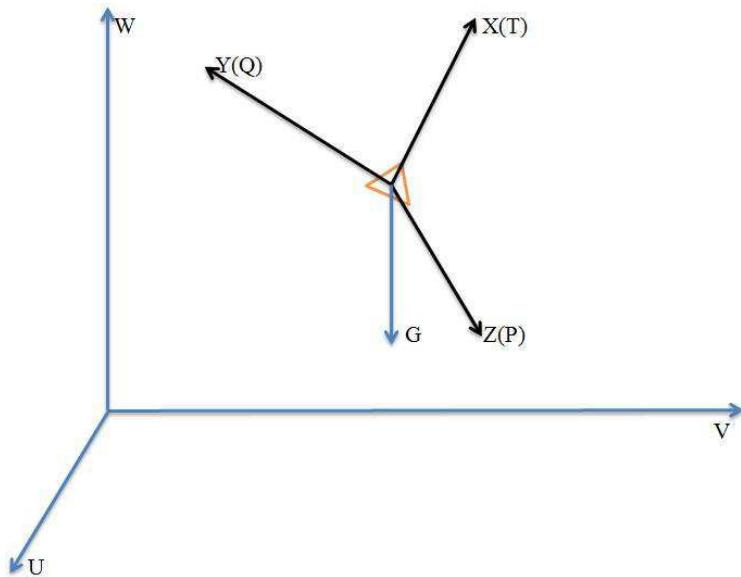


Рисунок 2 – Силы, действующие на летательный аппарат

На самолет действует сила тяжести G , которую возможно рассчитать по формуле:

$$G = m \cdot g, \quad (1)$$

где g – ускорение свободного падения, равное $9,81 \text{ м/с}^2$;

m – масса самолёта.

В системе координат $OXZY$ вдоль продольной оси аппарата OX действует сила:

$$T = F_{\text{тяги}} \cdot \cos \theta - G - F_{\text{con}}, \quad (2)$$

где θ – угол атаки;

$F_{\text{тяги}}$ – сила тяги двигателя;

F_{con} – сила лобового сопротивления воздуха.

На самолет воздействует скоростной напор воздушного потока, который возможно рассчитать по формуле:

$$q = \frac{V^2 \cdot \rho}{2}, \quad (3)$$

где ρ – плотность воздуха;

V – скорость самолёта.

Вдоль оси OZ на самолёт действует сила Р:

$$P = V_p \cdot \sin \psi - k_p \cdot q_p \quad (4)$$

где V_p – вектор скорости, направленный по оси P

k_p – коэффициент бокового сопротивления;

q_p – скоростной напор, действующий по оси P .

Подъемная сила крыла Q возникает при появлении набегающего потока и рассчитывается как:

$$Q = c_Q \cdot S_w \cdot q \quad (5)$$

где c_Q - коэффициент подъемной силы;

S_w - площадь крыла самолёта.

2 Построение системы управления курсом самолета

Действительный угол крена измеряется гироскопическим датчиком ($k_1 = 1$) и сравнивается с заданным значением ϕ_3 . Разность между ϕ_3 и ϕ усиливается посредством усилителя и подается на вход исполнительного механизма, который управляет отклонением элеронов. В качестве корректирующего звена по скорости ($d\phi/dt$) использован также гироскопический датчик.

Схема системы показана на рисунке 3 и 4.

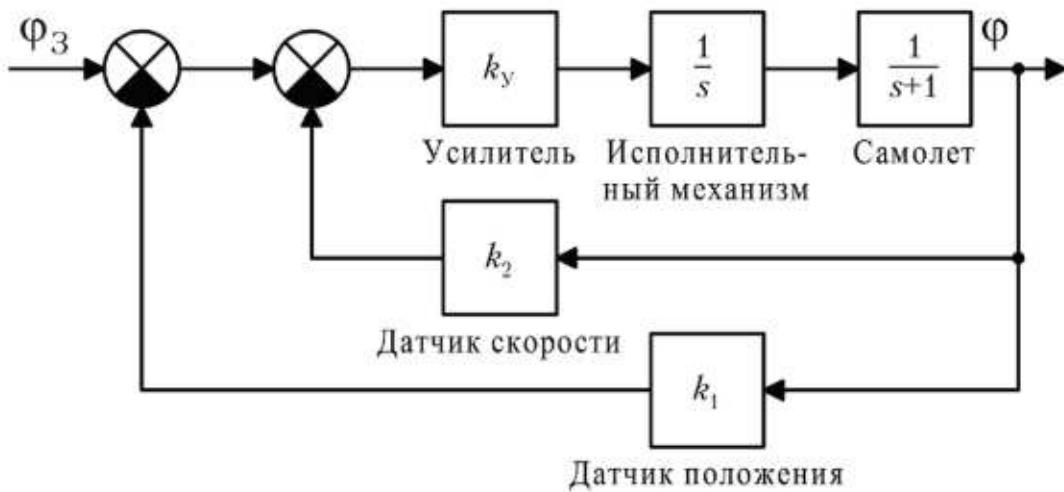


Рисунок 3 – Структурная схема САР угла самолета

Необходимо, используя параметрическую оптимизацию в среде SimInTech, определить параметры k_y и k_2 , при которых переходная характеристика при $\phi_3(t) = 1(t)$ имела перерегулирование не более 10%, а время регулирования (для «трубки» 2%) – не более 9 с.

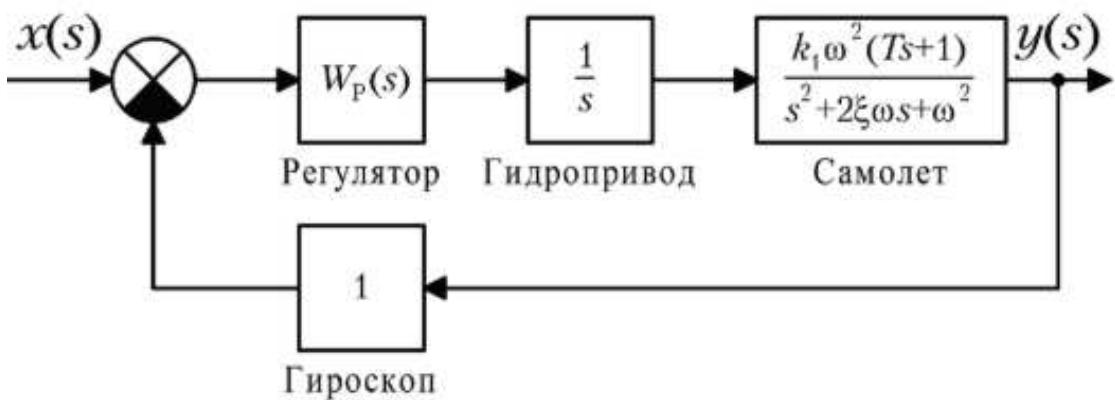


Рисунок 4 – Структурная схема САР скорости самолета

Требуется методом компьютерного моделирования в среде Simintech синтезировать регулятор, реализующий ПИД-закон регулирования с передаточной функцией так чтобы реакция системы на ступенчатый входной сигнал $x(1) = 1(0)$ имела перерегулирование не более 5% и время регулирования (при «трубке» 2%) менее 5 с.

2.1 Описание метода проектирования

Качество работы современных автоматизированных систем управления промышленными технологическими процессами в значительной степени зависит от того, насколько правильно выбраны настройки регуляторов, входящих в состав этих систем. Проблеме правильного выбора настроек регуляторов усугубляется еще и тем, что в процессе эксплуатации автоматической системы параметры объекта управления изменяются, и для обеспечения соответствия между этими параметрами и настраиваемыми параметрами регуляторов требуется их перенастройка. Решением проблемы может быть использование регуляторов с оптимальной настройкой их параметров. Для расчетов и оптимизации передаточных функций объекта будет использована среда динамического моделирования SimInTech. В качестве объектов автоматизации используются две передаточные функции: управление скоростью полета и управление креном и тангажом самолета.

2.2 Среда моделирования

Одна из наиболее главных стадий в жизненном цикле изделия это проектирование. Использование систем проектирования технических устройств позволяет существенно повысить качество изделий и увеличить срок эксплуатации.

Для выполнения выпускной квалификационной работы, в качестве среды разработки, была выбрана среда динамического моделирования технических систем SimInTech исходя из того, что обладает необходимыми элементами и модулями для выполнения задания проекта.

Данная система проектирования была выбрана в качестве среды разработки текущего проекта.

SimInTech осуществляет моделирование технологических процессов, протекающих в различных отраслях и моделирует системы управления, а также позволяет повысить качество проектирования систем управления за счет проверки принимаемых решений на любой стадии проекта [11].

Для детального исследования и анализа нестационарных процессов в ядерных и тепловых энергостанциях, в системах автоматического управления, в следящих приводах и роботах, и в любых технических системах, описание динамики которых может быть представлено в виде системы - алгебраических уравнений и реализовано методами моделирования.

В качестве языка программирования в SimInTech используется язык Си, с помощью которого выполняются сложные операции. В языке программирования поддерживается все основные конструкции языков высокого уровня, а также содержится большое набор математических функций, которые постоянно расширяется [12].

На данный момент, в SimInTech существует возможность создавать модели на базе специализированных расчетных кодов и внедрять их в комплексные модели, для этого и созданы модули расширения.

Помимо всего прочего, SimInTech позволяет пользователю создавать модели технических устройств, используя структурные блок-схемы. А для связи между различными проектами используется база данных сигналов, реализованных в виде отдельных проектов в единый пакет [13].

2.3 Оптимизация передаточной функции скорости самолета

В программном комплексе SimInTech была собрана структурная схема регулирования скорости, изображенная на рисунке 5.

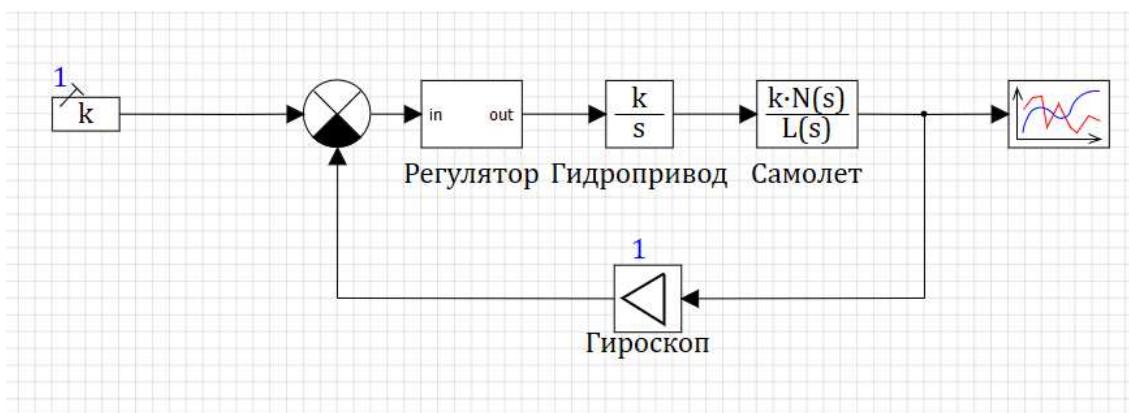


Рисунок 5 – Структурная схема регулирования скорости

Для обеспечения безопасного перемещения самолета в воздухе при регулировании скорости требуется поддерживать высокую точность в установившемся режиме. Для этого необходима нулевая статическая ошибка стабилизации скорости. С этой целью в системе для коррекции использован ПИД-регулятор (рисунок 6), коэффициенты которого будут использованы для оптимизации параметров системы.[14]

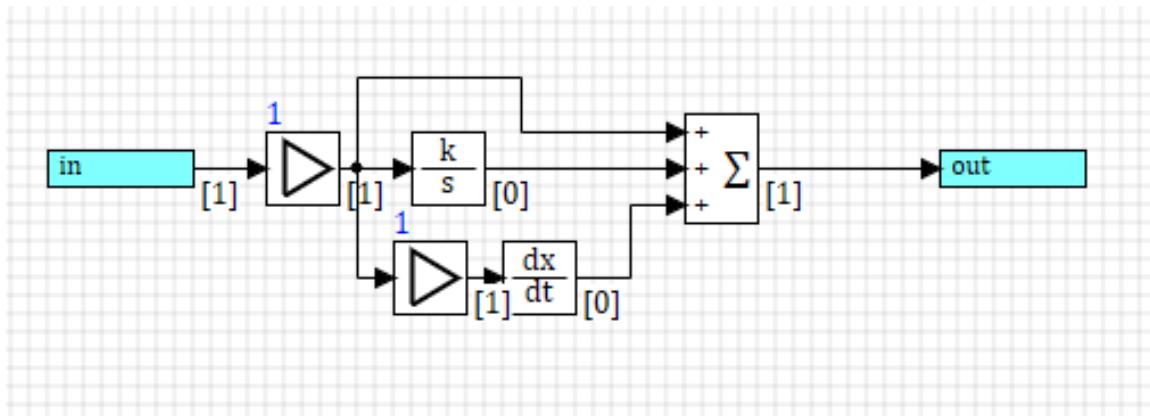


Рисунок 6 – Субмодель ПИД регулятора

Используя неоптимизированные параметры системы, кривая регулирования будет иметь вид, представленный на рисунке 7.

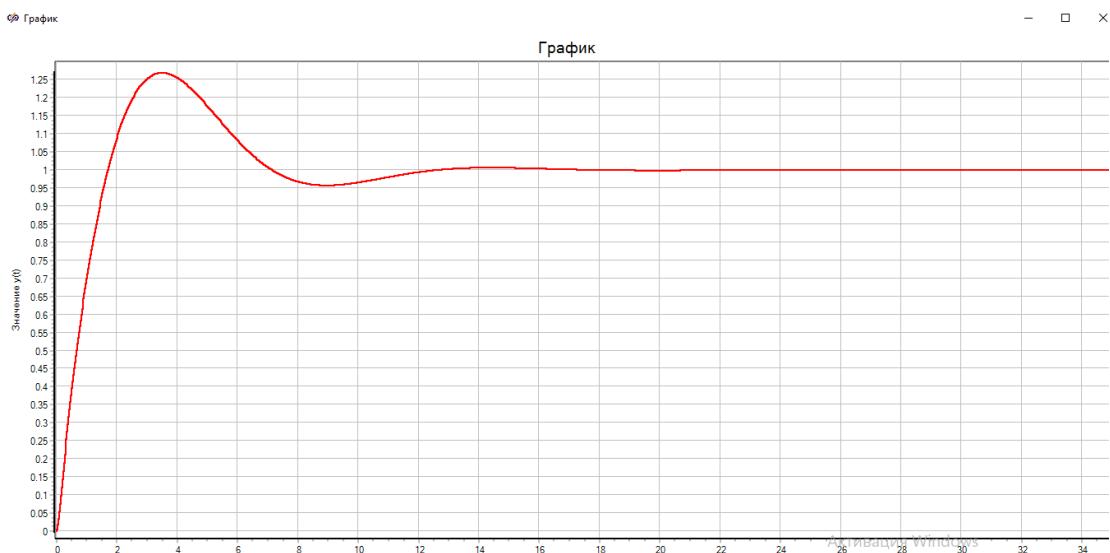


Рисунок 7 – Кривая регулирования

По данной кривой регулирования можно определить, что система является устойчивой. Выполнен расчет следующих параметров:

$$t_{\text{рег}} = 6,3 \text{ с},$$

где $t_{\text{рег}}$ – время регулирования сигнала в пятипроцентной зоне.

$$\sigma = \left| \frac{Y_{max1} - Y_{уст}}{Y_{уст}} \right| = 27\% ,$$

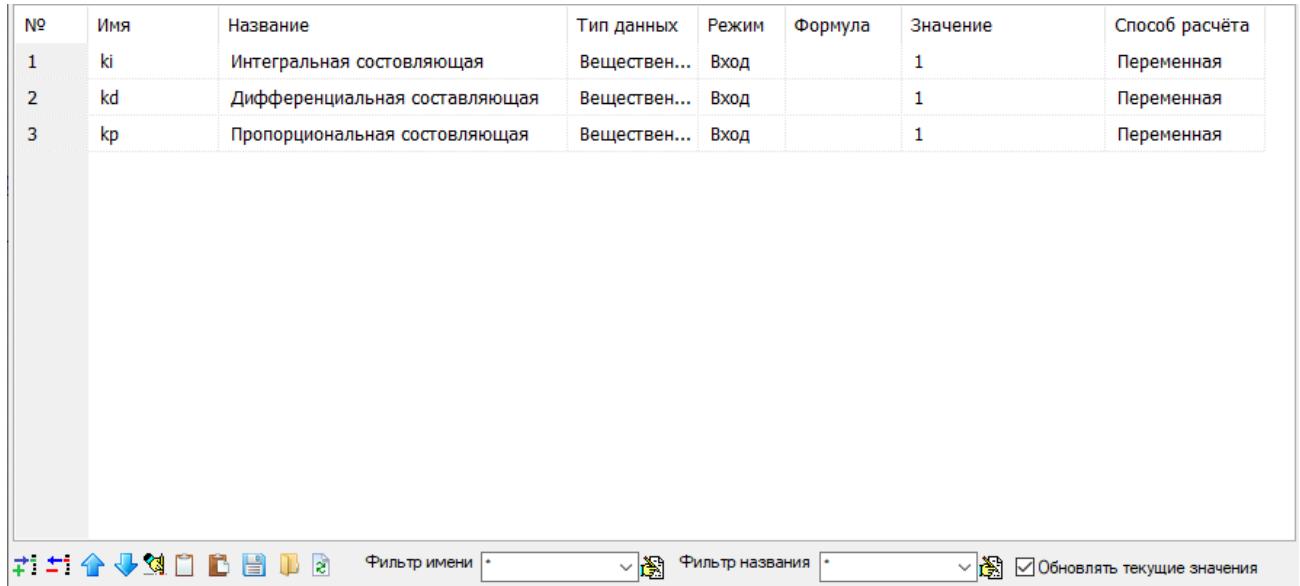
где σ -перерегулирование,

Y_{max1} -первое максимальное отклонение от установившегося значения,

$Y_{уст}$ – установившееся значение системы.

Для оптимизации системы необходимо добавить все регулируемые параметры в список сигналов проекта (рисунок 8)

№	Имя	Название	Тип данных	Режим	Формула	Значение	Способ расчёта
1	ki	Интегральная составляющая	Веществен...	Вход		1	Переменная
2	kd	Дифференциальная составляющая	Веществен...	Вход		1	Переменная
3	kp	Пропорциональная составляющая	Веществен...	Вход		1	Переменная



The screenshot shows a software interface for managing system parameters. At the top is a table with columns: №, Имя (Name), Название (Description), Тип данных (Data Type), Режим (Mode), Формула (Formula), Значение (Value), and Способ расчёта (Calculation Method). Three rows are listed, each corresponding to a PID controller component (ki, kd, kp) with values set to 1 and mode set to 'Переменная' (Variable). Below the table is a toolbar with various icons for file operations like save, open, and print. To the right of the toolbar are two filter fields labeled 'Фильтр имени' (Filter by name) and 'Фильтр названия' (Filter by description), both containing an asterisk (*). A checkbox labeled 'Обновлять текущие значения' (Update current values) is checked. The entire interface is contained within a light gray frame.

Рисунок 8 – Параметры системы

В роли оптимизатора был использован блок оптимизации SimInTech, сформированы критерии оптимизации: время регулирования, перерегулирование и установившееся значение. Задан диапазон значений критериев в таблице 1 и выбран метод оптимизации (рисунок 9).

Таблица 1- Диапазон значений критериев

Критерий	Минимальное значение	Максимальное значение
$t_{\text{пер}}$	0 с.	6 с.
σ	0%	5%
$Y_{\text{уст}}$	0,995	1,005

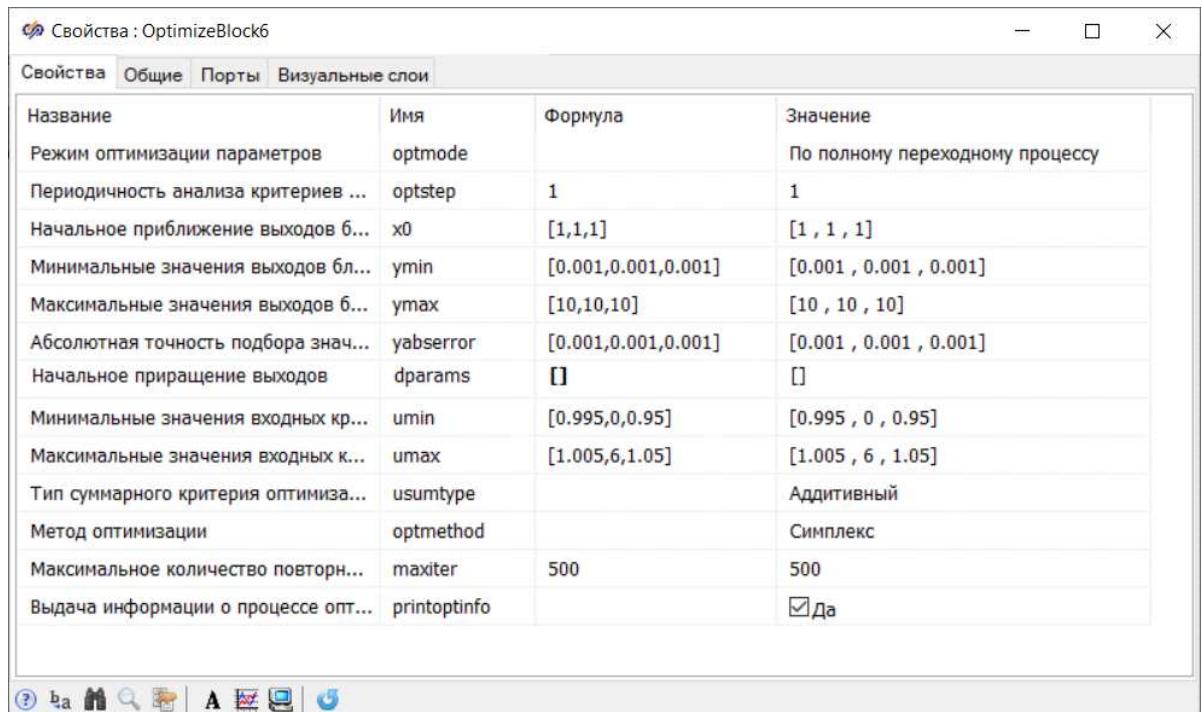


Рисунок 9 – Параметры блока оптимизации

Сформирована структурная схема для оптимизации по заданным критериям (рисунок 10). Использован измеритель времени переходного процесса, определение максимального отклонения осуществляется с помощью вычисления максимального абсолютного значения системы, установившееся значение определяется с помощью ступеньки и ключа.

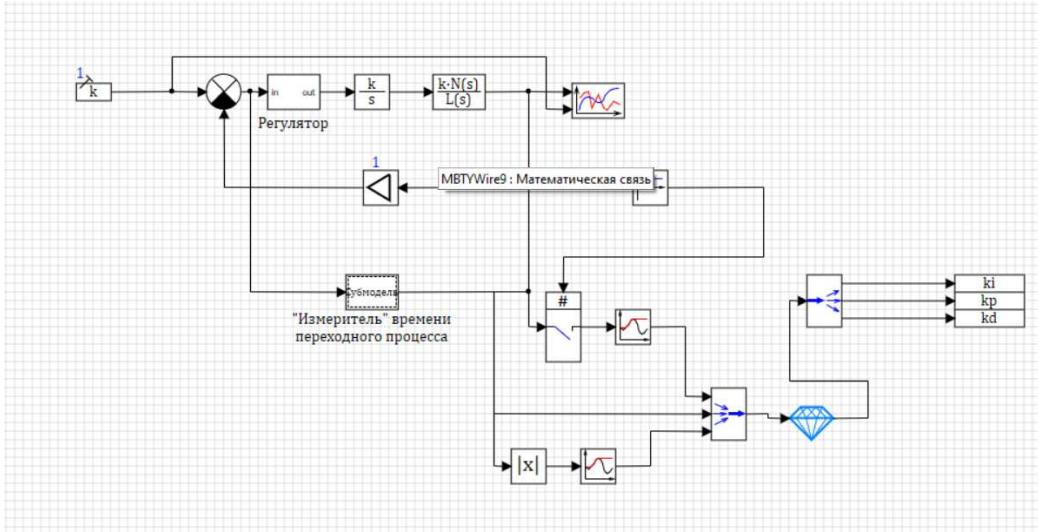


Рисунок 10 – Структурная схема для оптимизации по заданным критериям

После оптимизации были получены следующие параметры системы (рисунок 11)

Список сигналов проекта: C:\Users\л\Desktop\Юленька\Диплом\ДИПЛОМ_скорость\Скорость оптимизация.prt							
№	Имя	Название	Тип данных	Режим	Формула	Значение	Способ расчёта
1	ki	Сигнал1	Веществен...	Вход		0.001	Переменная
2	kd	Сигнал1	Веществен...	Вход		1.1920345	Переменная
3	kp	Сигнал2	Веществен...	Вход		10	Переменная

Рисунок 11 – Оптимизированные параметры системы

По новой кривой регулирования на рисунке 12 можно получить следующие параметры:

$$t_{\text{пер}} = 5.8 \text{ с}$$

где $t_{\text{рег}}$ – время регулирования сигнала в пятипроцентной зоне.

$$\sigma = \left| \frac{Y_{max1} - Y_{yct}}{Y_{yct}} \right| = 0\%$$

$$Y_{\text{yct}} = 1$$

где σ -перерегулирование,

Y_{max1} -первое максимальное отклонение от установившегося значения,

$Y_{\text{уст}}$ – установившееся значение системы.

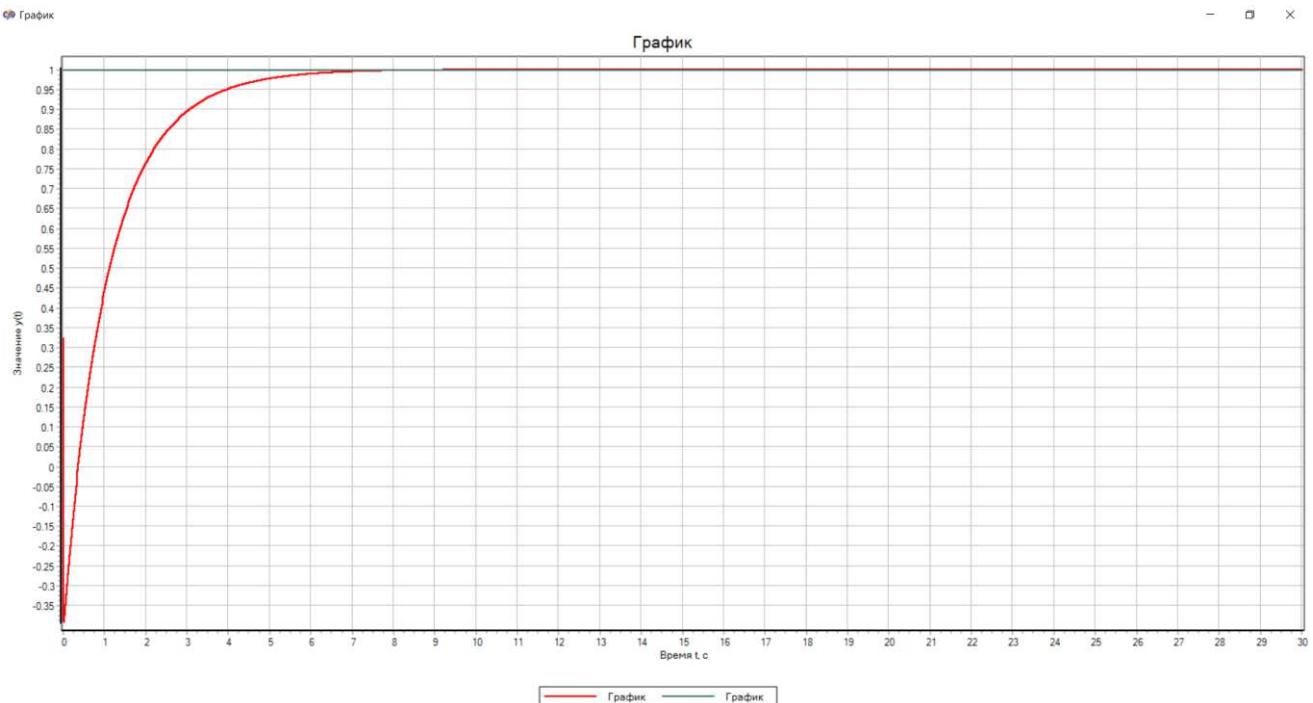


Рисунок 12 – Оптимизированная кривая регулирования

Сравнение параметров системы до и после оптимизации представлено в таблице 2.

Таблица 2- Сравнение параметров системы

Критерий	Значение до оптимизации	Значение после оптимизации
$t_{\text{рег}}$	6,3 с	6 с
σ	27%	0%
$Y_{\text{уст}}$	1	1

В результате оптимизации удалось исключить перерегулирование из системы, а также уменьшить ее время регулирования.

2.4 Оптимизация передаточной функций крена и тангажа самолета

Структурная схема регулирования крена собрана в программном комплексе SimInTech изображена на рисунке 13.

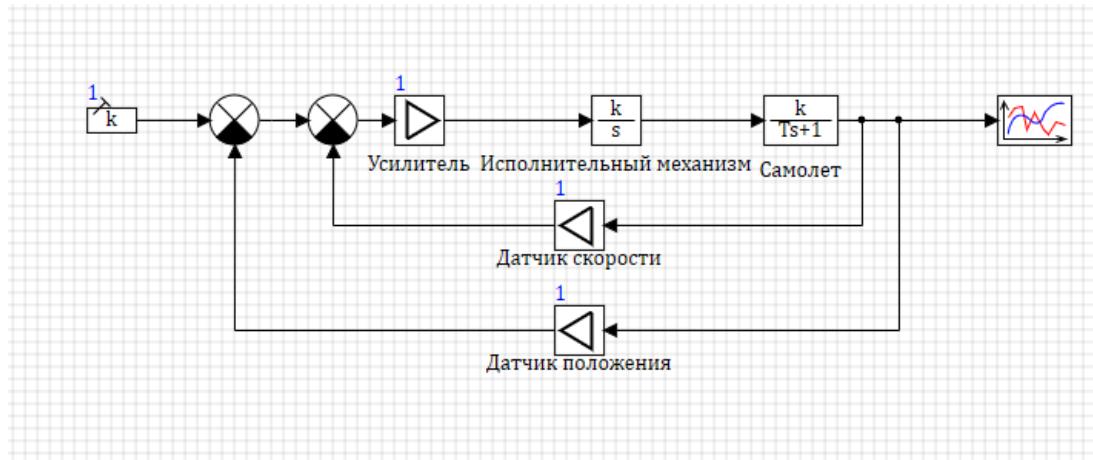


Рисунок 13– Структурная схема регулирования крена

В данной схеме для регуляции параметров системы будет использоваться регуляция коэффициентов усилителя и датчика скорости.

Кривая регулирования будет иметь вид, представленный на рисунке 14.

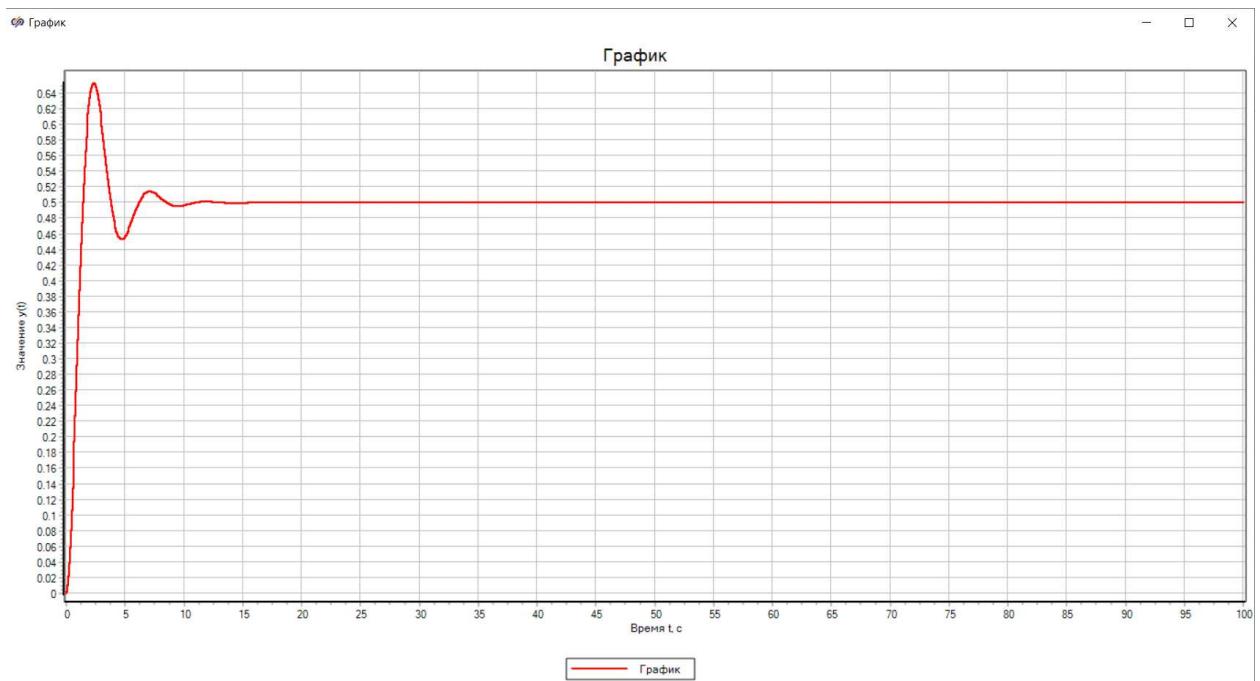


Рисунок 14 – Кривая регулирования

По данной кривой регулирования можно определить, что система является устойчивой. Выполнен расчет следующих параметров:

$$t_{\text{пер}} = 5 \text{ с},$$

где $t_{\text{пер}}$ – время регулирования сигнала в пятитысячной зоне.

$$\sigma = \left| \frac{Y_{\max 1} - Y_{\text{уст}}}{Y_{\text{уст}}} \right| = 30\%,$$

$$Y_{\text{уст}} = 0,5,$$

где σ -перерегулирование,

$Y_{\max 1}$ -первое максимальное отклонение от установившегося значения,

$Y_{\text{уст}}$ – установившееся значение системы.

По аналогии с передаточной функцией регуляции скорости добавлены коэффициенты, для оптимизации(рисунок 15)

№	Имя	Название	Тип данных	Режим	Формула	Значение	Способ расчёта
1	k2	коэффициент датчика скорости	Веществен...	Вход		1	Переменная
2	ky	коэффициент усилителя	Веществен...	Вход		1	Переменная

Фильтр имени: [] Фильтр названия: [] Обновлять текущие значения

Рисунок 15 – Кривая регулирования

Сформированы критерии оптимизации: время регулирования, перерегулирование и установившееся значение. Задан диапазон значений критериев в таблице 3 и выбран метод оптимизации (рисунок 16).

Таблица 3- Диапазон значений критериев

Критерий	Минимальное значение	Максимальное значение
$t_{\text{рег}}$	0 с	6 с
σ	0%	10%
$Y_{\text{уст}}$	0,9	1,1

Свойства : OptimizeBlock6

Свойства	Общие	Порты	Визуальные слои
Название	Имя	Формула	Значение
Режим оптимизации параметров	optmode		По полному переходному процессу
Периодичность анализа критериев оптимизации при расчёте в динамике, сек	optstep	1	1
Начальное приближение выходов блока	x0	[1,1]	[1 , 1]
Минимальные значения выходов блока	ymin	[0.001,0.001]	[0.001 , 0.001]
Максимальные значения выходов блока	ymax	[100,100]	[100 , 100]
Абсолютная точность подбора значений выходов	yabserror	[0.001,0.001]	[0.001 , 0.001]
Начальное приращение выходов	dparams	[]	[]
Минимальные значения входных критериев оптимизации	umin	[0.95,0,0.9]	[0.95 , 0 , 0.9]
Максимальные значения входных критериев оптимизации	umax	[1.05,6,1.1]	[1.05 , 6 , 1.1]
Тип суммарного критерия оптимизации	usumtype		Аддитивный
Метод оптимизации	optmethod		Симплекс
Максимальное количество повторных моделирований при расчёте по полно...	maxiter	500	500
Выдача информации о процессе оптимизации	printoptinfo		<input checked="" type="checkbox"/> Да

Рисунок 16 – Параметры блока оптимизации

Сформирована структурная схема для оптимизации по заданным критериям (рисунок 17). Использован измеритель времени переходного процесса, определение максимального отклонения осуществляется с помощью вычисления максимального абсолютного значения системы, установившееся значение определяется с помощью ступеньки и ключа.

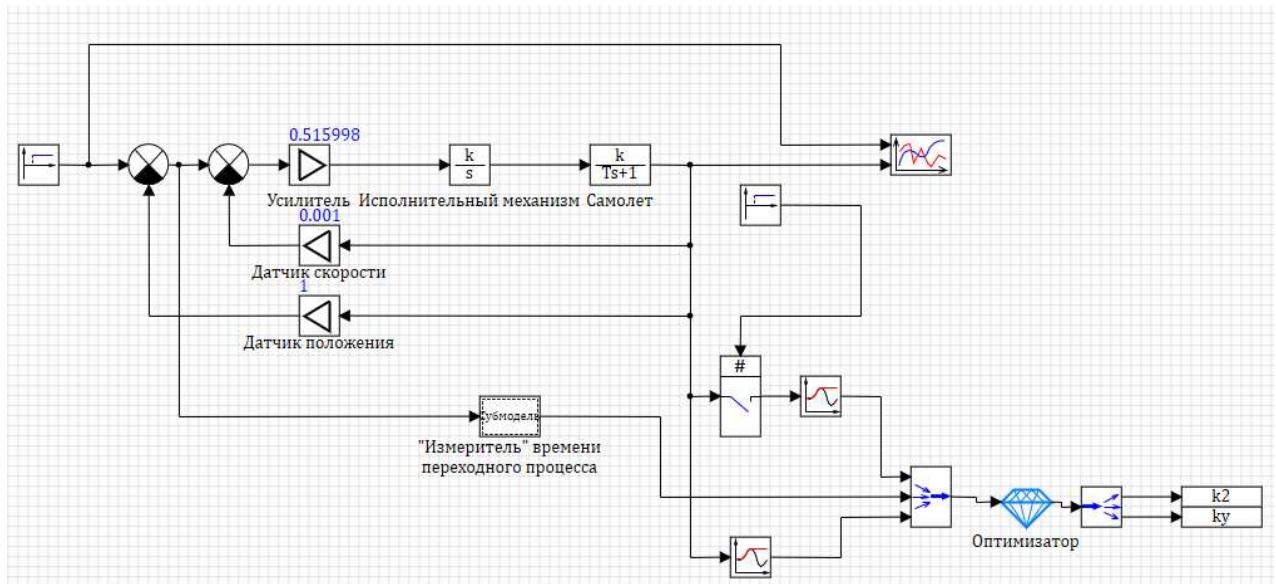


Рисунок 17– Структурная схема для оптимизации

После оптимизации были получены следующие параметры системы (рисунок 18)

Список сигналов проекта: C:\Users\l\Desktop\Юленька\Диплом\ДИПЛОМ_крен\Крен оптимизация.prt							
№	Имя	Название	Тип данных	Режим	Формула	Значение	Способ расчёта
1	tpp	Сигнал1	Вещественное	Вход		0	Переменная
2	dy	Сигнал1	Вещественное	Вход		0	Переменная
3	ky	Сигнал1	Вещественное	Вход		0.51599808	Переменная
4	k2	Сигнал1	Вещественное	Вход		0.001	Переменная

Рисунок 18 – Оптимизированные параметры системы

По новой кривой регулирования на рисунке 19 можно получить следующие параметры:

$$t_{\text{пер}} = 4 \text{ с}$$

где $t_{\text{пер}}$ – время регулирования сигнала в пятипроцентной зоне.

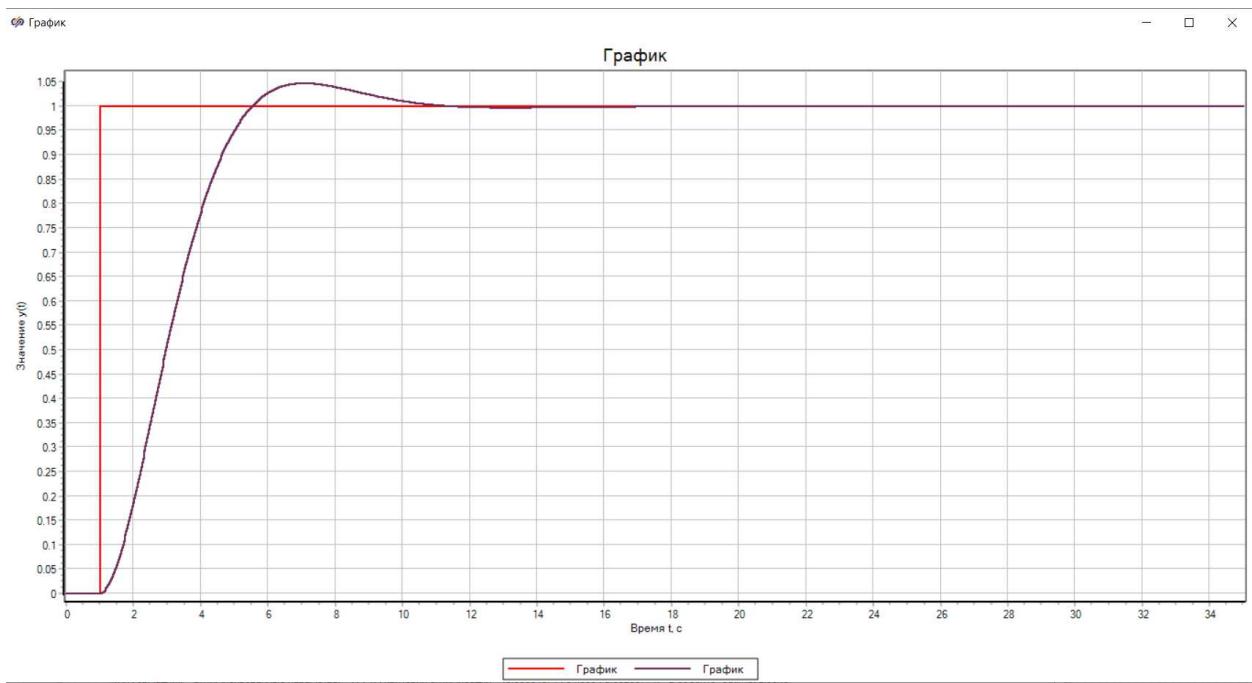
$$\sigma = \left| \frac{Y_{\max 1} - Y_{\text{уст}}}{Y_{\text{уст}}} \right| = 5\%$$

$$Y_{\text{уст}} = 1$$

где σ -перерегулирование,

$Y_{\max 1}$ -первое максимальное отклонение от установившегося значения,

$Y_{\text{уст}}$ – установившееся значение системы.



3 Формирование проекта системы автоматического регулирования

3.1 Создание базы данных проекта

В SimInTech есть возможность подключать внешнюю базу данных. Она подключается в виде модуля расширения, как динамически подключаемая библиотека (dll).

База сигналов проекта – это структурированная файловая база данных, содержащая в себе константы и переменные (сигналы), используемые в одном или сразу нескольких проектах.

Таким образом, для работы проекта системы необходима связь между схемами общего вида, для этого и была создана база данных (рисунок 20).

База данных включает в себя одну категорию, состоящую из трех групп сигналов: входные, выходные и сигналы для симуляции процесса. Входные сигналы состоят из следующих значений: управляющее воздействие скоростью самолета, управляющее воздействие горизонтальной и вертикальной составляющих. Аналогично входным сигналам сформированы выходные: значение скорость, угол вертикальный, угол горизонтальный. В сигналы для симуляции процесса входят: положение самолета в пространстве (X, Y, Z), вертикальный и горизонтальный наклон самолета.

Редактор базы данных сигналов: C:\Users\...\Desktop\ДИПЛОМ\ДИПЛОМ_БД\database.db							
Категории		Группы сигналов		Сводная			
Nº	1	Nº	1	Название	Тип данных	Формула	Значение
0	1	2	Input	V...	Веществен...	-0.37189085	Перемен...
		3	Output	Hor	Веществен...	-22.845405	Перемен...
		sim		V	Веществен...	1	Перемен...

Фильтр имени групп сигналов
По умолчанию (*)

Фильтр названия групп
Название группы

Глобальный фильтр

Фильтр категорий

Фильтр имени сигналов
По умолчанию (*)

Фильтр названия сигналов
По умолчанию (*)

Обновлять текущие значения

Справка Ok

Рисунок 20 – База данных проекта

3.2 Моделирование схем общего вида

При создании проекта управления самолета были добавлены следующие модули: модуль управления скоростью, модуль управления наклоном, модуль управления креном.

Для модуля управления скорости (риунок 21) взята оптимизированная передаточная функция регулирования скорости самолета. Константе присвоено значение управляющего воздействия скорости, запись значения скорости происходит после расчета передаточных функций.

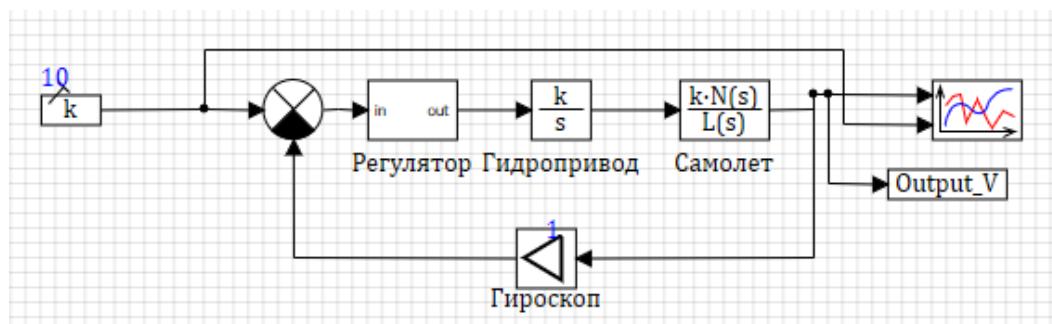


Рисунок 21 – Модуль управления скорости

Для модуля управления креном взята оптимизированная передаточная функция регулирования крена самолета (рисунок 22). Константе присвоено значение управляющего воздействия крена, запись значения крена происходит после расчета передаточных функций.

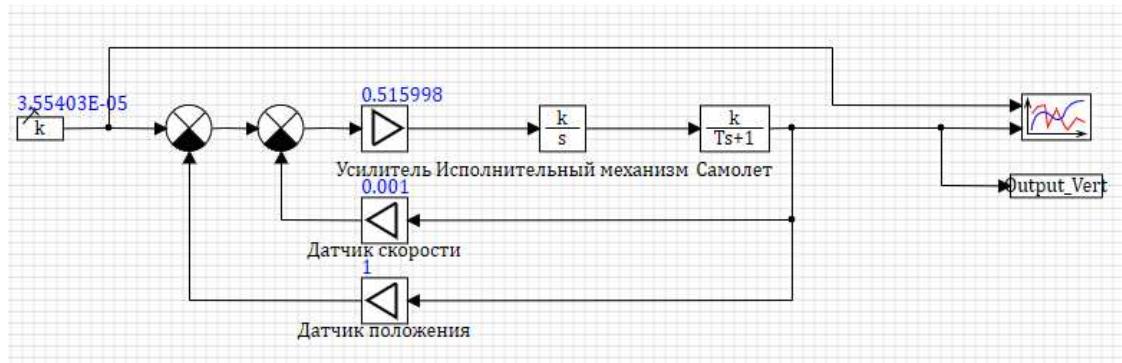


Рисунок 22 – Модуль регулирования крена самолета

Модуль управления тангажом был описан такой же передаточной функцией. После формирования всех модулей для объединения их в общий проект был создан пакет модулей (рисунок 23).

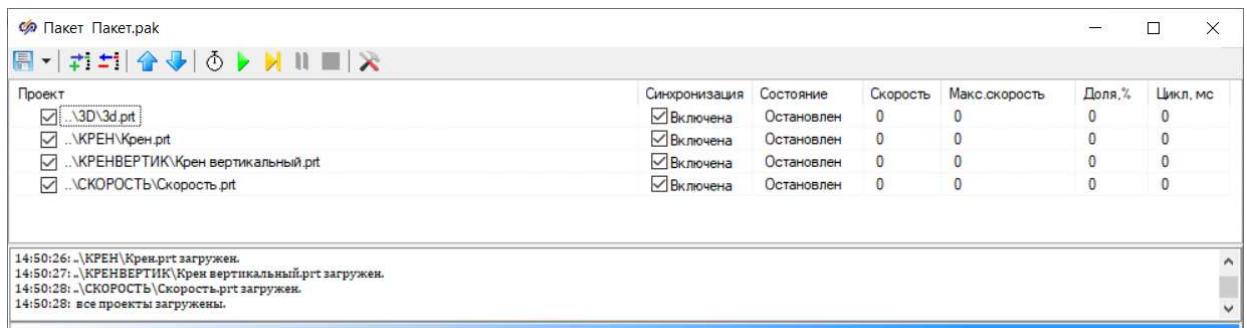


Рисунок 23 – Пакет проекта

3.3 Создание пульта управления оператора

Для управления системой был создан пульт управления, который позволяет задавать управляющие воздействия по всем модулям системы и при необходимости обеспечивает стабилизацию его горизонтального и вертикального положения, стабилизация происходит при нажатии соответствующей кнопки и сопровождается сигналом индикатора (рисунок 24).

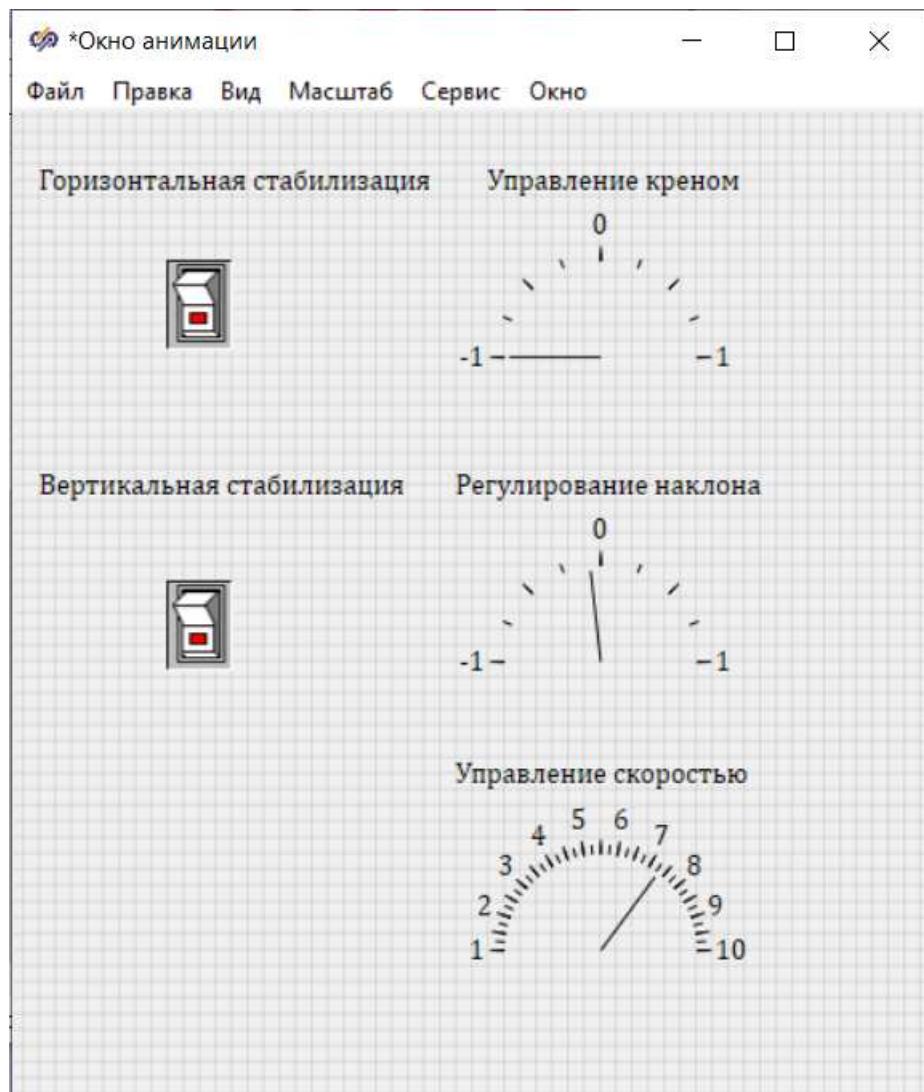


Рисунок 24 – Пульт управления оператора

Связь пульта управления и проекта осуществляется через скрипт пульта, в котором отслеживаются все манипуляции пользователя и передаются в базу данных, а из нее на модули (рисунок 25).

The screenshot shows a software interface titled "Скрипт анимации: Окно анимации". The menu bar includes "Файл", "Правка", "Поиск", "Расчёт", and "Справка". Below the menu is a toolbar with various icons. The main area displays a script in a hierarchical tree view:

```
1 initialization
2   Voltmeter.Value=1;
3   Voltmeter1.Value=0;
4   Voltmeter2.Value=0;
5   Button.Down=false;
6   Button.Value=0;
7   Button1.Down=false;
8   Button1.Value=0;
9   end
10  Input_V=roundstep(Voltmeter.Value,1);
11  if (Button2.Down=true) then
12    begin
13      Sim_Button=1;
14    end
15  else
16    begin
17      Sim_Button=0;
18    end
19    if (Button.Value=1) then
20      begin
21        Voltmeter1.DirectMode=false;
22      end
```

Рисунок 25 – Скрипт пульта управления

Стабилизация скрипта работает следующим образом: проверяется, нажата ли кнопка, после вычисляется абсолютное значение текущего угла наклона и умножается на определенный процент, после чего происходит проверка на положительное и отрицательное число, в первом случае задается отрицательное, во втором- положительное. Управление стрелочным прибором отключается, а отображение его значения соответствует текущему воздействию. Вертикальная стабилизация использует в качестве задающего

воздействия 2,5% от текущего значения угла наклона, а горизонтальная - 5% от текущего угла крена.

3.4 Разработка 3D визуализации проекта

Для создания визуализации в пакет проекта был добавлен новый модуль – динамическая отрисовка (рисунок 26).

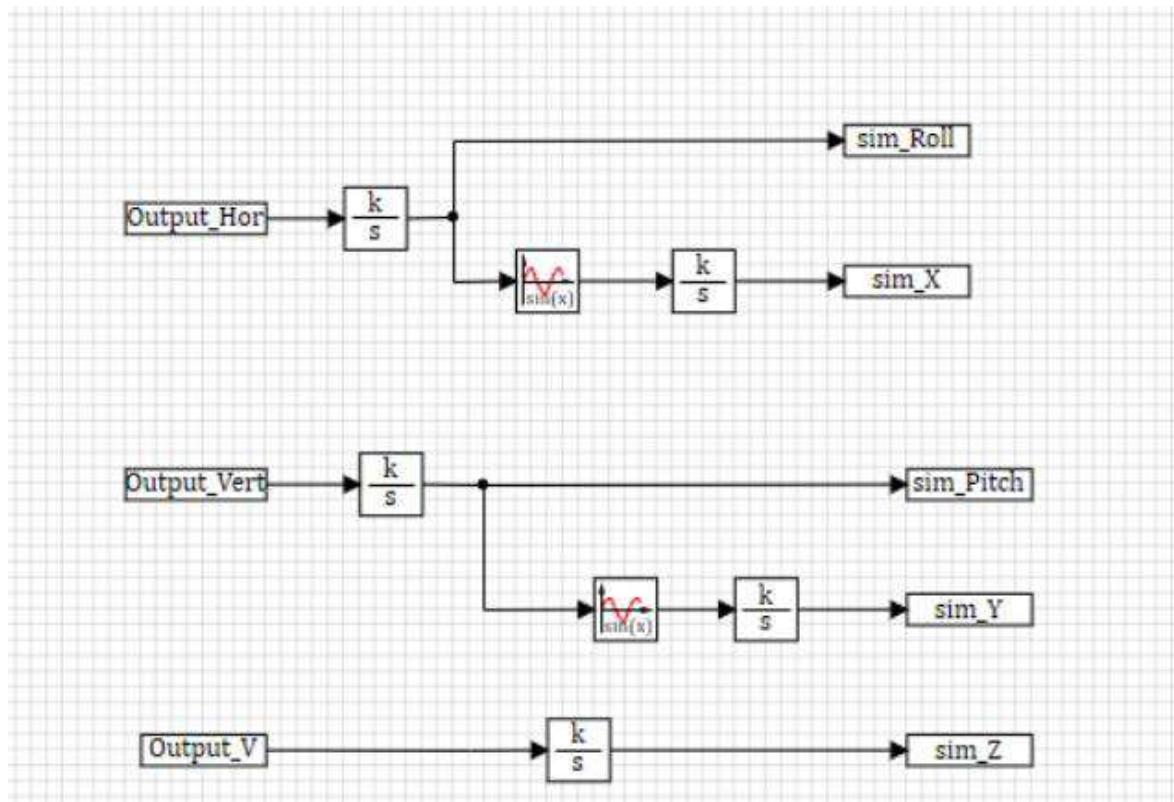


Рисунок 26 – Модуль динамической отрисовки

Данный модуль позволяет накапливать значение выходных характеристик системы и преобразовывать его в координаты 3D пространства, а также наклонять его горизонтально и вертикально, что и требуется в демонстрации этого проекта. В роли пространства выступает модуль 3D визуализации SimInTech.

Для подключения модуля визуализации использовался скрипт модуля динамической отрисовки (рисунок 27).

```
□ initialization
//функция выполняет создание окна визуализации
viewerwindow_1 = Viewer3DCreate;
viewerwindow_2 = Viewer3DCreate;
// процедура задания видимости сетки (плоскость Y=0)
Viewer3DSetGridVisibility(viewerwindow_1, False);
// процедура задания видимости координатных осей
Viewer3DSetAxesVisibility(viewerwindow_2, False);
// процедура задания видимости координатных осей
Viewer3DSetGridVisibility(viewerwindow_2, False);
// процедура установки размеров окна
Viewer3DSetWindowSize(viewerwindow_1, 750, 600);
// процедура установки положения окна
Viewer3DSetWindowPosition(viewerwindow_1, 600, 130);
// процедура установки размеров окна
Viewer3DSetWindowSize(viewerwindow_2, 1400, 600);
// процедура установки положения окна
Viewer3DSetWindowPosition(viewerwindow_2, 600, 730);
// функция устанавливает фон для окна 3D визуализации
Viewer3DSetBackGround(viewerwindow_1, "", getbgracolor(177, 222, 11, 1));
// функция устанавливает фон для окна 3D визуализации
Viewer3DSetBackGround(viewerwindow_2, "", getbgracolor(177, 222, 11, 1));
// поворот X
Sim_Pitch = 0;
// поворот Z
Sim_Roll = 0;
// поворот Y
Sim_Turn=180;
Sim_X = 0;
Sim_Y = 0;
Sim_Z = 0;
if(sim_button=0)then
  ...
```

Рисунок 27 – Скрипт модуля динамической отрисовки

Скрипт создает окно 3D визуализации, происходит отключение сетки, и меняется размер окна, указывается цвет заднего фона и происходит обнуление всех переменных. В данное окно загружается модель формата *.obj, переменные базы данных привязываются к координатам модели и наклону самолета. Для объединения модуля с проектом значения из базы данных группы симуляции привязываются к значениям модуля (рисунок 28).

Редактор базы данных сигналов: C:\Users\m\Desktop\Мацкевич Диплом\ДИПЛОМ_БД\database.db

Редактор Настройки Состояние сети

№	Категории
0	Группы сигналов
1	Input
2	Output
3	sim

Группа	Сводная					
№	Имя	Название	Тип данных	Формула	Значение	Способ расчёта
1	X	Свойство1	Веществен...		0	Переменная
2	Y	Свойство1	Веществен...		0	Переменная
3	Z	Свойство1	Веществен...		3.1866068	Переменная
4	Roll	Свойство1	Веществен...		0	Переменная
5	Pitch	Свойство1	Веществен...		0	Переменная
6	Button	Свойство1	Веществен...		0	Переменная

Фильтр имени групп сигналов
По умолчанию (*)

Фильтр имени сигналов
По умолчанию (*)

Фильтр названия сигналов
По умолчанию (*)

Глобальный фильтр

Фильтр категории

Справка Ok

Рисунок 28 – База данных проекта

После привязки значений, для синхронизации запуска и работы модулей визуализация была добавлен в пакет модулей проекта (рисунок 29).

The screenshot shows a software interface for managing project packages. The title bar reads "Пакет проекта.pak". The toolbar includes icons for file operations like Open, Save, and Print, as well as project-related functions. The main area is divided into two sections: "Проект" (Project) on the left and a table on the right.

	Синхронизация	Состояние	Скорость	Макс.скорость	Доля,%	Ц
<input checked="" type="checkbox"/> ..\ДИПЛОМ_крен\Крен.prt	<input checked="" type="checkbox"/> Включена	Остановлен	0	0	0	0
<input checked="" type="checkbox"/> ..\ДИПЛОМ_крен_верт\Крен вертикальный.prt	<input checked="" type="checkbox"/> Включена	Остановлен	0	0	0	0
<input checked="" type="checkbox"/> ..\ДИПЛОМ_отрисовка\Динамическая отрисовка.prt	<input checked="" type="checkbox"/> Включена	Остановлен	0	0	0	0
<input checked="" type="checkbox"/> ..\ДИПЛОМ_скорость\Скорость.prt	<input checked="" type="checkbox"/> Включена	Остановлен	0	0	0	0

At the bottom, a log window displays the following messages:

```
20:10:57: ..\ДИПЛОМ_крен\Крен.prt загружен.  
20:10:57: ..\ДИПЛОМ_крен_верт\Крен вертикальный.prt загружен.  
20:10:58: ..\ДИПЛОМ_отрисовка\Динамическая отрисовка.prt загружен.  
20:10:58: ..\ДИПЛОМ_скорость\Скорость.prt загружен.  
20:10:58: все проекты загружены.
```

Рисунок 29 – Внешний вид пакета проектов SimInTech

В результате была получена динамическая 3D визуализация, в которой модель самолета отображает воздействия на систему модулей проекта (рисунок 30).

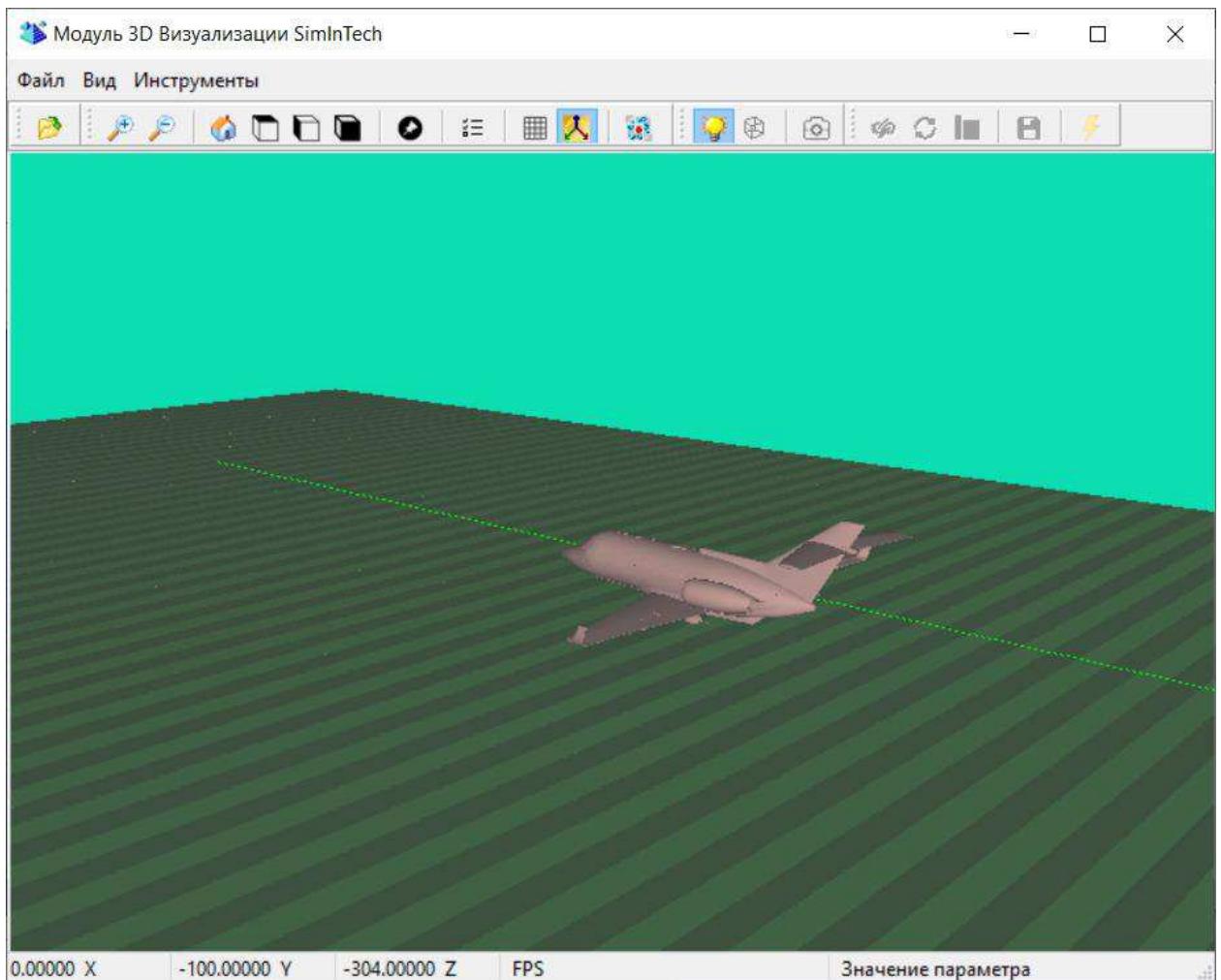


Рисунок 30 – 3D визуализация проекта

3.5 Запуск проекта

Для запуска системы, был запущен пакет проекта. С помощью вольтметра «Регулирование наклона» передвигая стрелку вольтметра влево – самолет будет наклоняться вниз, при передвижении вправо – самолет будет наклоняться вверх. Чтобы изменять скорость полета самолета в проекции, необходимо передвигать стрелку вправо или влево на вольтметре с подписью «Управление скоростью». При большой скорости самолету будет тяжелее

стабилизироваться. Вольтметр с подписью «Управление креном» отвечает за повороты самолета вправо и влево. Если повернуть стрелку вольтметра влево – самолет будет делать оборот вокруг оси в левую сторону, при повороте вправо - самолет будет делать оборот вокруг оси в правую сторону.

При нажатии на кнопку горизонтальной стабилизации самолет выравнивается ровно по горизонту, а при нажатии на кнопку вертикальной стабилизации – держит положение вертикально земле. Проект работает и при нажатии обоих кнопок. В этом случае самолет полностью стабилизируется и вертикально и горизонтально.

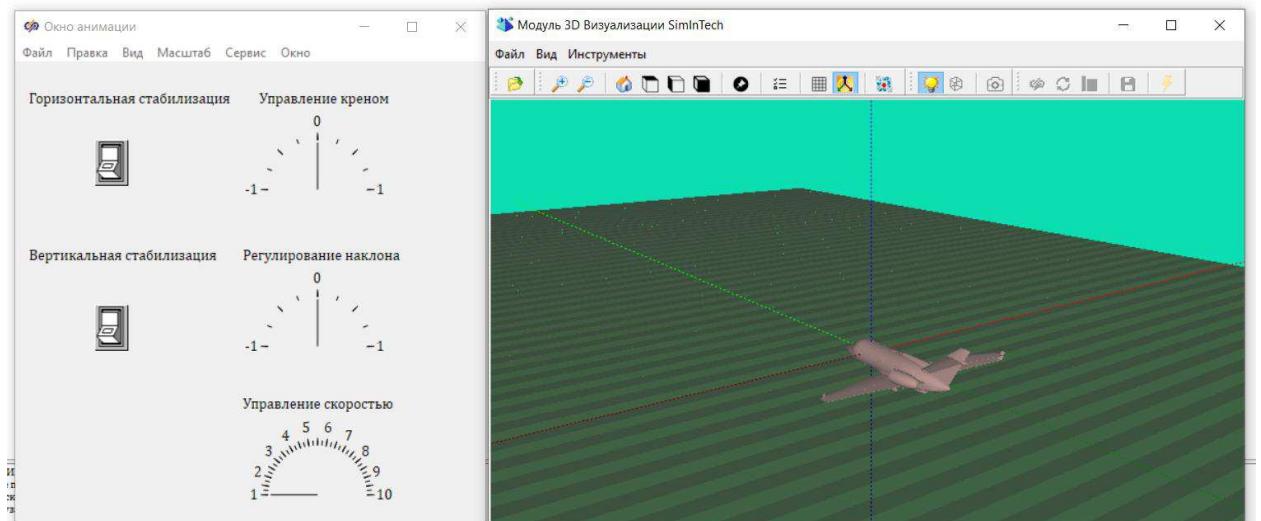


Рисунок 31 – Работа системы

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной работы был выполнен обзор развития воздушного судна. Подробно рассмотрено устройство самолета, а также дипломная работа содержит общую характеристику процесса управления самолетом.

Выбрана среды разработки текущего проекта. Описано использование таких средств среды разработки как база данных и язык программирования.

Была построена система управления скоростью, углом тангажа и крена самолета, а также построена система автоматической стабилизации положения самолета в пространстве в виде схемы в программе SimInTech.

Для обеспечения безопасного перемещения самолета требуется поддерживать высокую точность в установившемся режиме. С этой целью в системе для коррекции был реализован ПИД-регулятор и оптимизированы его характеристики для наилучшей работы.

Разработана имитационная модель системы управления курсом самолета, подключен визуальный 3D модуль и создана общая база данных для передачи параметров между проектами.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Кузовков, Н. Т. Динамика систем автоматического управления: учебник / Н.Т. Кузовков. – Москва: Машиностроение, 1968. – 428 с.
- 2 Балакин, В. Л., Динамика полета самолета. Устойчивость и управляемость продольного движения: учебное пособие / В.Л. Балакин, Ю.Н. Лазарев. – Самара. гос. аэрокосм. ун-т им. С. П. Королева, 2011. – 66 с.
- 3 Козлов, В. И. Системы автоматического управления летательными аппаратами: учебник / В.И.Козлов. – Москва: Машиностроение, 1979. – 216 с.
- 4 Катков, М. С. Непрерывные системы адаптивного управления с идентификаторами: науч. Изд / М.С. Катков. – Москва: Мир книги, 1992. – 386 с.
- 5 Красовский, А.А. Системы автоматического управления полетам и их аналитическое конструирование: учебное пособие / А.А. Красовский. – Москва: Наука, 1973. – 560 с.
- 6 Карташов, Б.А., Козлов О.С.Среда динамического моделирования технических систем SimInTech: практикум / Б.А.Карташов, О.С. Козлов. – Санкт – Петербург: ДМК Пресс, 2017. – 424 с.
- 7 Хабаров, С. П., Шилкина М.Л. Построение распределенных моделей в системе SimInTech: методические указания / С.П Хабаров, М.Л. Шилкина, – Санкт- Петербург: СПбГЛТУ, 2018.–122с.
- 8 Воронин, А.В. Теория автоматического управления. Основы построения и анализа систем автоматического регулирования: учебно-методическое пособие / Томск: Издательство Томского политехнического университета, 2013. – 51 с.
- 9 Гуськов, Ю.П., Управление полетом самолетов: учеб. пособие для вузов / Ю.П. Гуськов, Г.И. Загайнов. – Москва: Машиностроение, 1991. – 268с.
- 10 Гусев, А.Н. Системы автоматического управления самолетом: Учебное пособие / А.Н. Гусев. – Самар, гос. аэрокосм. ун-т. Самара, 2004. – 138 с.

11 SimInTech / ООО "ЗВ Сервис" <https://project797652.turbo.site>
2018-2020 [Электронный ресурс]. – Режим доступа:
<https://project797652.turbo.site/page363830#93279bd3-2068-4a20-a89e-b4c17ae80365>

12 Википедия. Свободная энциклопедия [Электронный ресурс]: Simintech, Режим доступа: <https://ru.wikipedia.org/wiki/Simintech>

13 Воробьев, В. Г., Кузнецов, С. В. Автоматическое управление полетом самолетов: учебное пособие / В.Г. Воробьев, С.В. Кузнецов. – Москва: Высшая школа, 1995. – 488 с.

14 Кейн, В.М. О методах расчета и оптимизации авиационных систем автоматического управления; учебное пособие/ В.М. Кейн. – Санкт-Петербург – Труды ВАУ ГА, 1969. – 80 с.

15 Справочная система SimInTech [Электронный ресурс] – Режим доступа: https://help.simintech.ru/#o_simintech/o_simintech.html

16 Цыпкин, Я.З. Адаптация и обучение в автоматических системах: учебное пособие / Я.З. Цыпкин. – Москва – Наука, 1968. – 399 с.

17 Михалев, И.А., Окоемов , Б.Н. Системы автоматического управления самолетами: учебник / И.А. Михалев, Б.Н. Окоемов. – Москва: Машиностроение, 1974. – 232 с.

18 СТО 4.2 07 2014. Система менеджмента качества. Общие требования к построению, изложению и оформлению документов учебной деятельности. – Взамен СТО 4.2 07 2012 ; дата введ. 09.01.2014. – Красноярск, 2014. – 60 с.

ПРИЛОЖЕНИЕ А

Скрипт проекта 3D визуализации

initialization

//функция выполняет создание окна визуализации

viewerwindow_1 = Viewer3DCreate;

viewerwindow_2 = Viewer3DCreate;

// процедура задания видимости сетки (плоскость Y=0)

Viewer3DSetGridVisibility(viewerwindow_1, False);

// процедура задания видимости координатных осей

Viewer3DSetAxesVisibility(viewerwindow_2, False);

// процедура задания видимости координатных осей

Viewer3DSetGridVisibility(viewerwindow_2, False);

// процедура установки размеров окна

Viewer3DSetWindowSize(viewerwindow_1, 750, 600);

// процедура установки положения окна

Viewer3DSetWindowPosition(viewerwindow_1, 600, 130);

// процедура установки размеров окна

Viewer3DSetWindowSize(viewerwindow_2, 1400, 600);

// процедура установки положения окна

Viewer3DSetWindowPosition(viewerwindow_2, 600, 730);

// функция устанавливает фон для окна 3D визуализации

Viewer3DSetBackGround(viewerwindow_1, "", getbgracolor(177, 222, 11, 1));

Продолжение Приложение А

```
// функция устанавливает фон для окна 3D визуализации  
Viewer3DSetBackGround(viewerwindow_2, "", getbgracolor(177,  
222, 11, 1));  
  
// поворот X  
  
Sim_Pitch = 0;  
  
// поворот Z  
  
Sim_Roll = 0;  
  
// поворот Y  
  
Sim_Turn=180;  
  
Sim_X = 0;  
  
Sim_Y = 0;  
  
Sim_Z = 0;  
  
if(sim_button=0)then  
  
begin  
  
// функция выполняет создание объекта, расположенного в файле с  
адресом  
  
object_1 = Viewer3DPlotFromFile(viewerwindow_1,  
"obj\samolet.obj", 0, 0, 0, [0, 0, 0]);  
  
// функция выполняет создание объекта, расположенного в файле с  
адресом  
  
object_2 = Viewer3DPlotFromFile(viewerwindow_2,  
"obj\samolet.obj", 0, 0, 0, [0, 0, 0]);  
  
end
```

Продолжение Приложение А

```
// функция выполняет масштабирование 3D-объекта в окне согласно  
заданному масштабу по осям в процентах  
  
Viewer3DSetScale(viewerwindow_1, object_1, 2, 2, 2);  
  
//функция устанавливает цвет для 3D-объекта в окне  
  
Viewer3DSetColor(viewerwindow_1, object_1, getbgracolor(97, 102,  
143, 1));  
  
// функция выполняет задание новых координат для объекта в окне  
  
Viewer3DSetPosition(viewerwindow_1, object_1, Sim_X, Sim_Y, -  
Sim_Z);  
  
// функция выполняет поворот объекта в окне вокруг его оси X на  
заданный угол  
  
Viewer3DSetPitch(viewerwindow_1, object_1, Sim_Pitch);  
  
// функция выполняет поворот объекта в окне вокруг его оси Z на  
заданный угол  
  
Viewer3DSetRoll(viewerwindow_1, object_1, Sim_Roll);  
  
//функция выполняет поворот объекта в окне вокруг его оси Y на  
заданный угол  
  
Viewer3DSetTurn(viewerwindow_1, object_1, Sim_Turn);  
  
// функция выполняет масштабирование 3D-объекта в окне согласно  
заданному масштабу по осям в процентах  
  
Viewer3DSetScale(viewerwindow_2, object_2, 2, 2, 2);  
  
//функция устанавливает цвет для 3D-объекта в окне
```

Продолжение Приложение А

```
Viewer3DSetColor(viewerwindow_2, object_2, getbgracolor(97, 102, 143,
1));
// функция выполняет задание новых координат для объекта в окне

Viewer3DSetPosition(viewerwindow_2, object_2, Sim_X, Sim_Y, -
Sim_Z);
// функция выполняет поворот объекта в окне вокруг его оси X на
заданный угол

Viewer3DSetPitch(viewerwindow_2, object_2, Sim_Pitch);

// функция выполняет поворот объекта в окне вокруг его оси Z на
заданный угол

Viewer3DSetRoll(viewerwindow_2, object_2, Sim_Roll);

// функция выполняет поворот объекта в окне вокруг его оси Y на
заданный угол

Viewer3DSetTurn(viewerwindow_2, object_2, Sim_Turn);

// процедура задает координаты базовой точки для камеры

Viewer3DSetCameraPosition(viewerwindow_1, 50, 20, -50);

// процедура задания 3D-примитива в качестве цели камеры

Viewer3DSetCameraTarget(viewerwindow_1, object_1);

// процедура задает координаты базовой точки для камеры

Viewer3DSetCameraPosition(viewerwindow_1, Sim_X, Sim_Y, -
Sim_Z);
for (i = 0, 100) begin
```

Продолжение Приложение А

```
// функция выполняет создание параллелепипеда с размерами  
wid×len×hei в точке (X,Y,Z)  
  
object_id = Viewer3DPlotParallelepiped(viewerwindow_1, 1000, 1, 8,  
0, -100, -i*8);  
  
if i mod 2 = 0 then  
  
//функция устанавливает цвет для 3D-объекта в окне  
  
Viewer3DSetColor(viewerwindow_1, object_id,  
getbgracolor(48, 186, 30,1))  
  
else  
  
//функция устанавливает цвет для 3D-объекта в окне  
  
Viewer3DSetColor(viewerwindow_1, object_id,  
getbgracolor(20, 112, 8,1));  
  
// функция выполняет создание параллелепипеда с размерами  
wid×len×hei в точке (X,Y,Z)  
  
object_id =  
Viewer3DPlotParallelepiped(viewerwindow_1, 1000, 1, 8, 0, -100, i*8);  
  
if i mod 2 = 0 then  
  
//функция устанавливает цвет для 3D-объекта в окне  
  
Viewer3DSetColor(viewerwindow_1, object_id,  
getbgracolor(48, 186, 30,1))  
  
else  
  
//функция устанавливает цвет для 3D-объекта в окне
```

Продолжение Приложение А

```
Viewer3DSetColor(viewerwindow_1,          object_id,  
getbgracolor(20, 112, 8,1));  
  
// функция выполняет создание параллелепипеда с размерами  
wid×len×hei в точке (X,Y,Z)  
  
object_id = Viewer3DPlotParallelepiped(viewerwindow_2,  
1000, 1, 8, 0, -100, -i*10);  
  
if i mod 2 = 0 then  
  
//функция устанавливает цвет для 3D-объекта в окне  
  
Viewer3DSetColor(viewerwindow_2,          object_id,  
getbgracolor(48, 186, 30,1))  
  
else  
  
//функция устанавливает цвет для 3D-объекта в окне  
  
Viewer3DSetColor(viewerwindow_2,          object_id,  
getbgracolor(20, 112, 8,1));  
  
end;  
  
end;  
  
// функция выполняет задание новых координат для объекта в окне  
  
Viewer3DSetPosition(viewerwindow_1, object_1, Sim_X, Sim_Y, -Sim_Z);  
  
// функция выполняет задание новых координат для объекта в окне  
  
Viewer3DSetPosition(viewerwindow_2, object_2, Sim_X, Sim_Y, -Sim_Z);  
  
// функция выполняет поворот объекта в окне вокруг его оси X на  
заданный угол  
  
Viewer3DSetPitch(viewerwindow_1, object_1, Sim_Pitch);
```

Продолжение Приложение А

// функция выполняет поворот объекта в окне вокруг его оси Z на заданный угол

Viewer3DSetRoll(viewerwindow_1, object_1, Sim_Roll);

// функция выполняет поворот объекта в окне вокруг его оси Y на заданный угол

Viewer3DSetTurn(viewerwindow_1, object_1, Sim_Turn);

// функция выполняет поворот объекта в окне вокруг его оси X на заданный угол

Viewer3DSetPitch(viewerwindow_2, object_2, Sim_Pitch);

// функция выполняет поворот объекта в окне вокруг его оси z на заданный угол

Viewer3DSetRoll(viewerwindow_2, object_2, Sim_Roll);

// функция выполняет поворот объекта в окне вокруг его оси у на заданный угол

Viewer3DSetTurn(viewerwindow_2, object_2, Sim_Turn);

//// процедура задает координаты базовой точки для камеры

Viewer3DSetCameraPosition(viewerwindow_1,Sim_X ,Sim_Y+20,-
Sim_Z+50)

ПРИЛОЖЕНИЕ В

Скрипт проекта пульта управления

initialization

Voltmeter.Value=1;

Voltmeter1.Value=0;

Voltmeter2.Value=0;

Button.Down=false;

Button.Value=0;

Button1.Down=false;

Button1.Value=0;

end

Input_V=roundstep(Voltmeter.Value,1);

if (Button2.Down=true) then

begin

Sim_Button=1;

end

else

begin

Sim_Button=0;

end

if (Button.Value=1) then

begin

Voltmeter1.DirectMode=false;

Продолжение Приложение Б

```
FillCircle.Color=getbgracolor(0,255,0,1);

b=abs(sim_Roll)*0.05;

if(sim_Roll>0) then

begin

Input_Hor=-b;

end

if(sim_Roll<0) then

begin

Input_Hor=b;

end

Voltmeter1.Value=Input_Hor;

end

else

begin

Voltmeter1.DirectMode=true;

FillCircle.Color=getbgracolor(0,0,255,1);

Input_Hor=Voltmeter1.Value;

end

if (Button1.Value=1) then

begin
```

Продолжение Приложение Б

```
FillCircle1.Color=getbgracolor(0,255,0,1);

Voltmeter2.DirectMode=false;

b=abs(sim_Pitch)*0.025;

if(sim_Pitch>0) then

begin

Input_Vert=-b;

end

if(sim_Pitch<0) then

begin

Input_Vert=b;

end

Voltmeter2.Value=Input_Vert;

end

else

begin

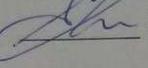
Voltmeter2.DirectMode=true;

FillCircle1.Color=getbgracolor(0,0,255,1);

Input_Vert=Voltmeter2.Value;

end
```

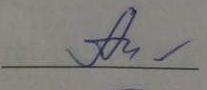
Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Кафедра «Системы автоматики, автоматизированное управление и
проектирование»

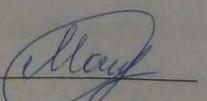
УТВЕРЖДАЮ
Заведующий кафедрой

С.В. Ченцов
«26» 06 2020 г.

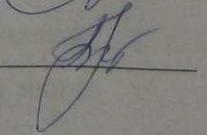
БАКАЛАВРСКАЯ РАБОТА

27.03.04 – Управление в технических системах

ПОСТРОЕНИЕ СИСТЕМЫ УПРАВЛЕНИЯ КУРСОМ САМОЛЕТА В СРЕДЕ МОДЕЛИРОВАНИЯ SIMINTECH

Руководитель  «26» 06. 2020 г. доц., канд.техн.наук
А.В. Чубарь

Выпускник  «26» 06. 2020 г. Ю.А. Мацкевич

Нормоконтролер  «26» 06. 2020 г. Т.А. Грудинова

Красноярск 2020