

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Институт машиностроения

(наименование института полностью)

Кафедра

«Промышленная электроника»

(наименование)

11.03.04 Электроника и микроэлектроника

(код и наименование направления подготовки/специальности)

Электроника и робототехника

(направленность (профиль)/специальности)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему Модель термокамеры с регулированием температуры

Обучающийся

А.А. Малько

(Инициалы Фамилия)

(личная подпись)

Руководитель

канд. техн. наук, доцент М.В. Позднов

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультант

О.А. Головач

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2024

Аннотация

Название дипломной работы: «Модель термокамеры с регулированием температуры».

Выпускная работа состоит из введения, восьми глав, заключения, таблиц, списка литературы, включая зарубежные источники, и графической части на 6 листах формата А1.

Ключевым вопросом дипломной работы является проектирование модели термокамеры с возможностью регулирования температуры. Мы затрагиваем проблему улучшения качества жизни людей.

Целью работы является разработка мобильной термокамеры с возможностью регулирования температуры и ее системы управления.

Дипломная работа может быть разделена на следующие логически взаимосвязанные части: постановка целей и задач, анализ аналогичных устройств на рынке; анализ элемента Пельтье, а также его доступных моделей, выбор термопасты для мобильной термокамеры; разработка электрической схемы и печатной платы, проведение ряда экспериментов на элементе Пельтье; анализ наиболее эффективного метода беспроводной связи для управления термокамерой, а также разработка приложения на мобильное устройство.

В конце исследования мы представляем работу, результатом которой является практическая реализация модели термокамеры с возможностью регулирования температуры с помощью мобильного приложения. Термокамера может использоваться для различных бытовых задач.

Подводя итоги, мы бы хотели подчеркнуть, что данная работа актуальна для различных бытовых задач. Эта работа была проведена для того, чтобы предложить пользователю наиболее дешевый вариант устройства с наибольшим количеством функций.

Abstract

The title of the graduation work is «Model of a thermal camera with temperature control».

The senior paper consists of an introduction, four parts, a conclusion, tables, list of references and the graphic part on 6A1 sheets.

The key issue of the thesis is the design of model of a thermal camera with the possibility of temperature control. We are addressing the issue of improving people's quality of life.

The aim of the work is to develop a mobile thermal camera with the possibility of temperature control and its control system.

The graduation work may be divided into several logically connected parts which are: setting goals and objectives, analyzing similar devices on the market; analyzing the Peltier element, as well as its available models, choosing thermal paste for a mobile thermal camera; developing an electrical circuit and a printed circuit board, conducting a number of experiments on the Peltier element; analyzing the most effective method of wireless communication for control a thermal camera, as well as the development of an application for a mobile device.

Finally, we present the work, the result of which is the practical implementation of a thermal camera model with the possibility of temperature control using a mobile application. The thermal chamber can be used for various household tasks.

In conclusion we'd like to stress this work is relevant for various household tasks. This work was carried out in order to offer the user the cheapest version of the device with the largest number of functions.

Содержание

1	Постановка задачи	9
1.1	Поиск и анализ технических параметров аналогичных устройств	9
1.2	Поиск и анализ технических параметров аналогичных устройств	10
1.3	Критерии для разработки	11
1.4	Принцип управления температурой в термокамере.....	11
2	Анализ элемента Пельтье	14
2.1	Элемент Пельтье	14
2.2	Выбор элемента Пельтье	15
2.3	Выбор термопасты для элемента Пельтье.....	24
2.3.1	Проблема выбора	24
2.3.2	Анализ термопаст и выбор самой лучшей и экономически выгодной модели.....	25
3	Проектирование устройства.....	27
3.1	Структура устройства.....	27
3.2	Разработка импульсного понижающего преобразователя напряжения.....	27
3.3	Схема импульсного преобразователя	34
3.4	Разработка печатной платы.....	35
3.5	Экспериментальное исследование тепловых процессов	37
3.5.1	Первый опыт по определению эффективного метода подачи напряжения и тока	37
3.5.2	Второй опыт по определению эффективного метода подачи напряжения.....	45
3.5.3	Аппроксимация полученных функций.....	47

4Управление термокамерой	50
4.1 Проблема исследования	50
4.2 Выбор способа беспроводной связи	50
4.3 Выбор программы для реализации приложения	52
4.4 Программный код для мобильного приложения	54
4.5 Тестирование и отладка.....	58
Заключение	60
Список используемых источников.....	61
Приложение А Программный код первой части первого опыта	64
Приложение Б Программный код второй части первого опыта	66
Приложение В Программный код для работы мобильного приложения на смартфон	68

Список терминов и определений

Аппроксимация – замена одних математических объектов другими, в том или ином смысле близкими к исходным

Маркетплейс – торговая площадка в интернет-ресурсе

ПИД-регулятор – устройство в управляющем контуре с обратной связью

Рекомбинация – исчезновение пары свободных носителей заряда противоположных знаков (электронов и дырок) в среде с выделением энергии.

Термокамера – камера для термической обработки

Элемент Пельтье – термоэлектрический преобразователь, основанный на эффекте Пельтье

ArduinoNano – микроконтроллер, разработанный торговой маркой Arduino

C++ – компилируемый типизированный язык программирования общего назначения

SPICE-симулятор – симулятор электронных схем общего назначения с открытым исходным кодом

Перечень сокращений и обозначений

ИППН – импульсный понижающий преобразователь напряжения

ПИД – пропорционально-интегрально-дифференцирующий

САПР – система автоматизированного проектирования

ШИМ – широтно-импульсная модуляция

ЭП – элемент Пельтье

Введение

В современном мире все чаще прослеживается тенденция улучшения, оптимизации и повышения качества нашей жизни в области коммуникаций, медицины, энергетики, транспорта и других сфер, благодаря изобретению новых технологий.

Одной из причин, почему инновации стали настолько важными, связана с технологическим прогрессом. Быстрые изменения в области технологий требует модернизации уже существующих, а также введению новых, созданных с нуля процессов.

Термоэлектрическое явление передачи энергии при протекании электрического тока в месте контакта разнородных проводников было открыто французским физиком Жак Пельтье в начале 19 века. Возможности и способы применения элементов, которые используют данное физическое явление, безграничны. Использование традиционных компрессорных установок в таких ситуациях затруднительно из-за крупных габаритов, наличия движущихся механических частей, наличия охлаждающих жидкостей и компрессоров.

Именно поэтому полезным изобретением является разработка мобильной термокамеры, которая поможет человеку нагреть или охладить необходимые продукты. Кроме того, такие термокамеры могут быть использованы для выращивания кисло – молочных продуктов за счет поддержания постоянной температуры. Также такое устройство можно использовать в медицинской сфере для хранения препаратов, где также необходимо поддержание постоянной температуры.

1 Постановка задачи

1.1 Поиск и анализ технических параметров аналогичных устройств

Объектами исследования являются явление термоэлектрического эффекта и принцип регулирования температуры.

Предметом исследования в выпускной квалификационной работе является мобильная термокамера на основе термоэлектрического эффекта Пельтье. Разработка камеры и ее структурных элементов позволяет глубже раскрыть тему термоэлектрического эффекта, а в частности процесса охлаждения, нагрева и регулирования температуры.

Главной целью работы является конструирование универсальной термокамеры для хранения различных продуктов, а также системы управления для нее. Камера должна быть многофункциональной и иметь гибкие настройки регулирования температуры.

При разработке были поставлены основные задачи:

- изучить физические особенности термоэлектрического модуля, установить основные зависимости между величинами, выявить основные закономерности при работе с модулем;
- спроектировать регулятор температуры и изготовить его печатную плату;
- разработать конструкцию термокамеры;
- разработать программное обеспечение для управления температурой в камере.

При работе с модулем было сделано множество предположений касательно физических процессов, происходящих внутри него. Следует ожидать, что выбор формы питающего напряжения модуля напрямую влияет на его производительность и эффективность. Также предполагается, что графические зависимости для холодопроизводительности модуля, представленные производителями, не соответствуют реальной перекачивающей тепловой способности модуля.

В ходе проведения работы были использованы различные методы исследования. В качестве практического метода использовался метод измерения, измерительное устройство в виде датчика определяло температуру внутри камеры, а осциллограф фиксировал значения напряжений и токов в схеме.

Научная новизна работы заключается в усовершенствовании программной части устройства с точки зрения добавления смарт функций, которые позволяют создать более точное и гибкое регулирование. Функции позволяют менять направление процессов, продукт можно как охлаждать по заданному диапазону времени, так и нагревать. Были проведены исследования, определяющие реальную производительность элемента, были предприняты попытки рассмотреть физику тепловых процессов, происходящих внутри элемента.

1.2 Поиск и анализ технических параметров аналогичных устройств

В современном мире представлено достаточное количество компактных холодильных камер в различном ценовом диапазоне. Необходимо произвести поиск аналогичных устройств на маркетплейсах и их анализ. Полученные данные занесем в таблицу 1.

Таблица 1 – Технические параметры аналогичных устройств

Марка прибора	AVS CC-27WBC	Starwind CF-132	Meuver AF-K30	Разработная термокамера
Объем, л	27	32	30	5
Габариты, м	0,35×0,48×0,38	0,45×0,3×0,45	0,61×0,37×0,37	0,15×0,15×0,4
Терморегулирование	нагрев, охлаждение, поддержание	нагрев, охлаждение	охлаждение, заморозка	нагрев, охлаждение, поддержание
Диапазон температур, °C	-2...+65	+18...+65	-20...+20	-20...+20
Уровень шума, дБ	50	50	65	50
Напряжение питания, В	17	12	12	12
Наличие дисплея	нет	да	да	нет
Наличие приложения	нет	нет	нет	да
Стоимость, руб.	24680	7590	18295	5000

Проанализировав технические характеристики аналогичных устройств можно сделать несколько выводов. Большинство устройств, представленных на рынке, не имеют мобильного приложения для управления, более того, некоторые модели не имеют даже дисплея. Также следует отметить, не все модели имеют разнообразное терморегулирование. В нашей работе предлагаются возможности нагрева, охлаждения и регулирования температуры, а также управление термокамерой с помощью приложения на мобильное устройство.

1.3 Критерии для разработки

Разрабатываемая модель термокамеры с регулированием температуры предназначена для хранения и приготовления различного вида сырья, устройство может работать в нескольких режимах: нагрев, охлаждение, поддержание постоянной температуры.

Устройство выполнено в виде двух элементов, термокамеры и блока управления. Корпус термокамеры состоит из пенополистирола, на внешней стороне закреплен радиатор, к которому прикреплен ЭП, для улучшения теплопроводностью между этих двух элементов нанесена термопаста. Кроме того, для увеличения теплообмена на радиатор с внешней стороны закреплен кулер для охлаждения.

Блок управления состоит из печатной платы, в основе которой лежит отладочная плата небольшого размера – ArduinoNano. Также, программный алгоритм, загруженный в плату, позволяет с помощью мобильного приложения изменять и сохранять температуру в полости термокамеры.

1.4 Принцип управления температурой в термокамере

В основу управления температурой внутри термокамеры легло ПИД – регулирование. Оно позволяет обеспечить точное и быстрое регулирование, что необходимо для качественной работы термокамеры. «Принцип пропорционально-интегрально-дифференциального (ПИД) регулирования основан на формировании управляющего сигнала, являющегося суммой трёх слагаемых (составляющих): пропорциональный (P); интегральный

(I); дифференциальный (D). Данные составляющие формируются соответствующими блоками программного модуля ПИД-регулятора.

ПИД регулятор на базе микропроцессорных устройств, как и любая система автоматического регулирования с обратной связью, предусматривает наличие двух входов и одного выхода: вход сигнала обратной связи с датчика технологического параметра и вход сигнала задания требуемой величины технологического параметра. Выходной сигнал ПИД-регулятора формируется на программном уровне в результате математических вычислений с тремя слагаемыми (P, I, D):»[7]

$$u(t) = P + I + D \quad (1)$$

Объектом регулирования и управления является температура внутри термокамеры, при использовании датчика температуры осуществляется принцип обратной связи, программный код получает и анализирует полученные данные и вносит поправки в величину управляющего воздействия.

Для управления величиной напряжения преобразователя микроконтроллер использует корректируемый алгоритмом ШИМ (широтно-импульсно модулированный) сигнал. На рисунке 1 представлена структурная схема регулятора.



Рисунок 1 – Структурная схема регулятора

На рисунке 1 величина $y(t)$ является температурой объекта управления, $y_1(t)$ величина температуры объекта, которая фиксируется датчиком температуры. Величина $f(t)$ является возмущающим воздействием, то есть окружающей средой, при изменениях температуры окружающей среды изменяются показания датчика температуры.

2 Анализ элемента Пельтье

2.1 Элемент Пельтье

Большая часть мобильных термокамер создана на базе элемента Пельтье. Элемент Пельтье (ЭП) — это устройство, работающее на принципе термоэлектрического эффекта, при котором происходит преобразование электрической энергии в тепловую и наоборот. Элементы Пельтье используются для охлаждения или нагрева объектов на основе эффекта Пельтье. ЭП является компактным и надежным элементом и одной из задач разработчика является выбор подходящего для этого элемента Пельтье, то есть ЭП должен быть экономически выгодным и выдавать наибольший перепад температур, используя меньшее количество потребляемых мощностей. Фотография ЭП представлена на рисунке 2.



Рисунок 2 – Фотография элемента Пельтье

Эффект Пельтье – термоэлектрическое явление переноса энергии при прохождении электрического тока в месте контакта(спая) двух разнородных проводников, от одного проводника к другому.

«Объяснение эффекта. Причина возникновения эффекта Пельтье на контакте полупроводников с одинаковым видом носителей тока (два полупроводника n-типа или два полупроводника p-типа) такая же, как и в случае контакта двух металлических проводников. Эффект Пельтье, как и все термоэлектрические явления, выражен особенно сильно в цепях,

составленных из электронных (n-тип) и дырочных (p-тип) полупроводников. Рассмотрим ситуацию, когда ток в контакте идёт от дырочного полупроводника к электронному (p-n). При этом электроны и дырки движутся навстречу друг другу, и, встретившись, рекомбинируют. В результате рекомбинации освобождается энергия, которая выделяется в виде тепла» [1]. «При постоянном электрическом токе носители возбуждаются, это охлаждает одну сторону (электрод) элемента и греет другой электрод» [18]. «Полупроводниковые элементы расположены на двух керамических пластинах, соединенных между собой так, чтобы нагревающиеся контакты выходили на одну сторону пластины, а охлаждающиеся на другую» [10].

2.2 Выбор элемента Пельтье

Производители ЭП часто предлагают в качестве основной расчетной характеристики тепловую характеристику - зависимость теплового потока теплового насоса $Q_{ЭП}$ от перепада температур на его сторонах dT (холодной и горячей) при заданном токе через ЭП. Эта характеристика сама по себе без учета параметров камеры и потерь в ЭП не дает информации о максимально достижимом перепаде температуры. При отсутствии дополнительных расчетов оценка по тепловым характеристикам дает завышенный перепад по температуре.

Для выбора ЭП и учета вышеуказанных факторов предлагается использовать тепловую схему замещения ЭП с учетом влияния параметров термокамеры (рисунок 3) [1].

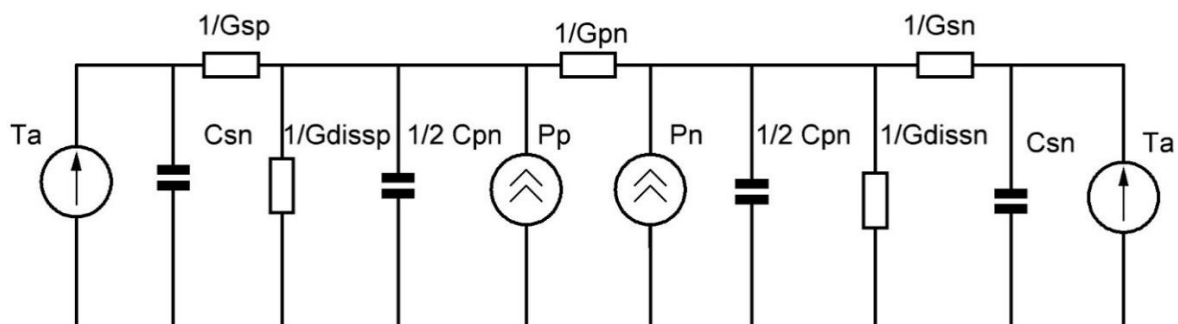


Рисунок 3 – Схема замещения элемента Пельтье

На схеме замещения имеются тепловые сопротивления, отмеченные

символами $1/G$ (обратные тепловым проводимостям), тепловые емкости - отмеченные символами C , источники теплового потока с символами P . Слева и справа к ЭП подключаются радиатор и термокамера, соответственно. По левому и правому краю схемы установлено два одинаковых источника ЭДС, определяющих одинаковые и заданные температуры окружающей среды T_a . В режиме охлаждения поток тепловой мощности направлен в ЭП справа налево. Температура горячей стороны ЭП - T_p , а холодной - T_n .

Рассчитать схему из двух ЭДС и двух источников тока можно методом наложения. Сначала нужно рассчитать систему только с источниками ЭДС без источников тока. Далее рассчитать с закоротками вместо источников ЭДС, но с источниками тока. Потом надо результаты расчетов относительно потоков в ветвях сложить. Очевидно, что система на рисунке 3 только с двумя источниками ЭДС решается тривиально – потоки от действия одинаковых ЭДС отсутствуют, следовательно, можно исключить в дальнейшем цепи с ЭДС и места их ввода закоротить.

Для оценочного выбора ЭП необходимо воспользоваться упрощенной тепловой схемой, которую в последующем надо решить исходя из указанного выше принципа. Для ее построения примем несколько допущений:

- Работа схемы рассматривается в установившемся режиме работы, следовательно, все емкости C исключаются из схемы замещения.

- Холодная сторона работает на холодильник ($G_{sn}=0$), идеально изолирующий камеру от внешней среды. Следовательно, со схемы правее источника P_n удаляются все элементы.

- Горячая сторона подключена к идеальному радиатору с нулевым тепловым сопротивлением (G_{sp} стремится к бесконечности). Следовательно, левее источника P_p располагается закоротка на общий вывод.

С учетом этих допущений из схемы по рисунку 3 получим схему замещения (рисунок 4).

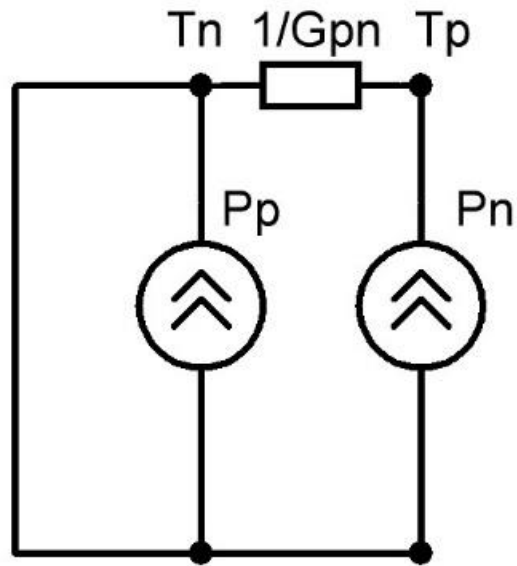


Рисунок 4 – Схема замещения с учетом допущений

В этой схеме замещения $P_n = Q_{дж} - Q_{ЭП}$ представляет разницу Джоулевых потерь $Q_{дж}$ ЭП и теплового потока теплового насоса $Q_{ЭП}$, а $P_p = Q_{дж} + Q_{ЭП}$ является их суммой [3].

Для определения потерь $Q_{дж}$, нужно использовать электротепловые характеристики, представляющие зависимости $U(dT)$ при заданном токе i , представлены на рисунке 5.

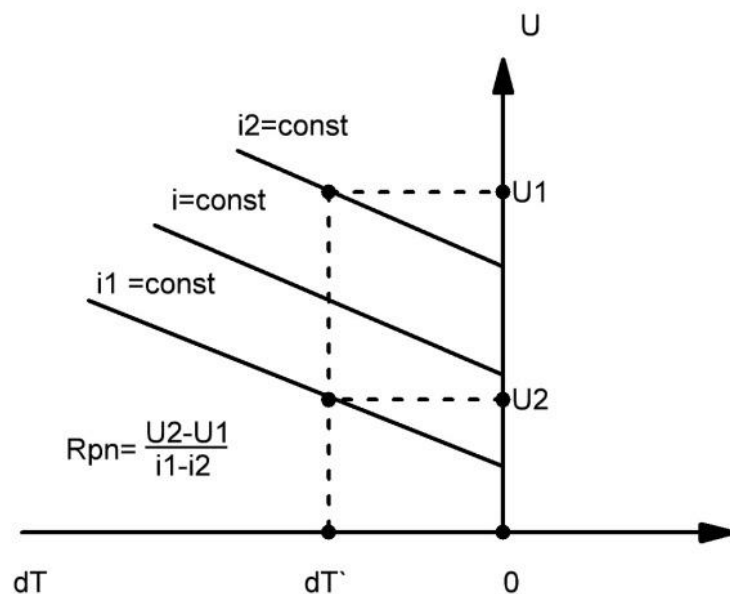


Рисунок 5 – Электротепловые характеристики

По ним определяется динамическое сопротивление R_{pn} в окрестности

заданного тока i . Для этого нужно выбрать ближайшие сверху и снизу характеристики с постоянными токами i_2 и i_1 . Определить для произвольного dT напряжение на верхней на нижней характеристике U_1 и U_2 и определить $R_{pn} = (U_2 - U_1)/(I_2 - I_1)$.

Далее по формуле определить потери $Q_{дж} = 1/2 \times I_2 \times R_{pn}$. Нужно отметить, что это половина потерь общих потерь в ЭП. Одна половина потерь выделяется на холодной стороне, а вторая на горячей.

В тепловой характеристике, которую предоставляет производитель отражается зависимость только компоненты $Q_{ЭП}$ от dT . Сама компонента $Q_{ЭП}$ отсутствует в схеме замещения (рисунке 3 и 4) и является частью источников P_p и P_n . Поэтому схему замещения, которая представлена на рисунке 4 можно преобразовать в схему с источниками по рисунке 6а, выделив из источников P_n и P_p источник $Q_{ЭП}$, и далее перегруппировать схему в рисунок 6б.

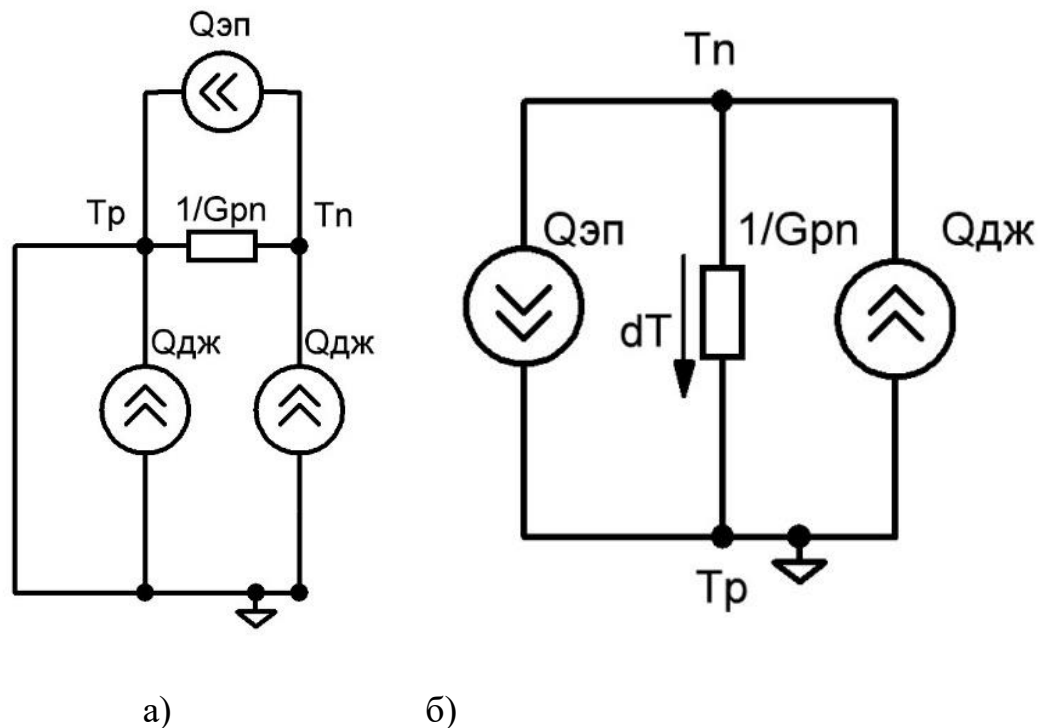


Рисунок 6 – Последовательное преобразование тепловой схемы с допущениями

По полученной схеме видно, что источник тока $Q_{дж}$ нагружен на тепловую характеристику ЭП $Q_{ЭП}(dT)$. Падение температур на ЭП от протекания этого источника определяет падение температур между камерой и внешней средой $dT = T_n - T_p$.

Графически рассчитать эту схему можно проводя горизонтальную линию с уровня $Q_{дж}$ до пересечения с тепловой характеристикой при заданном токе i . (рисунок 7)

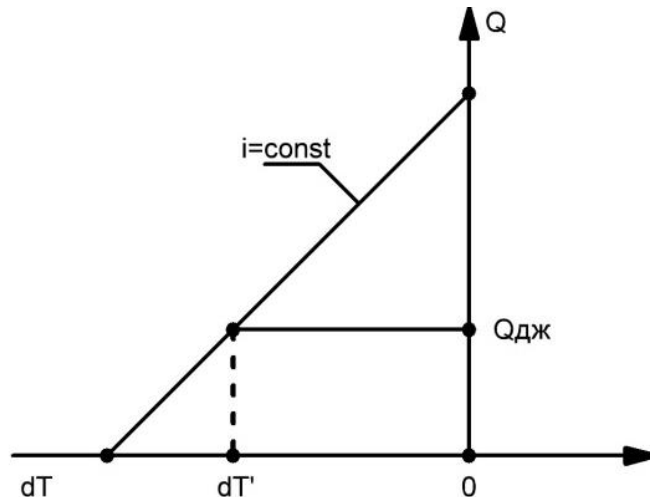


Рисунок 7 – Пример расчета

Проведенный расчет дает перепад температур $dT' = T_n - T_p$ на ЭП при идеальных условиях – идеальном радиаторе и абсолютно изолированной холодильной камере. Учитывая, что горячая сторона из-за идеального радиатора имеет температуру $T_p = T_a$, то перепад dT' совпадает с перепадом температур между камерой холодильника и внешней средой. Пример построения графиков для одного из элемента (ТЕС1-12706) на основе графических характеристик, которые дает производитель, представлен на рисунке 8.

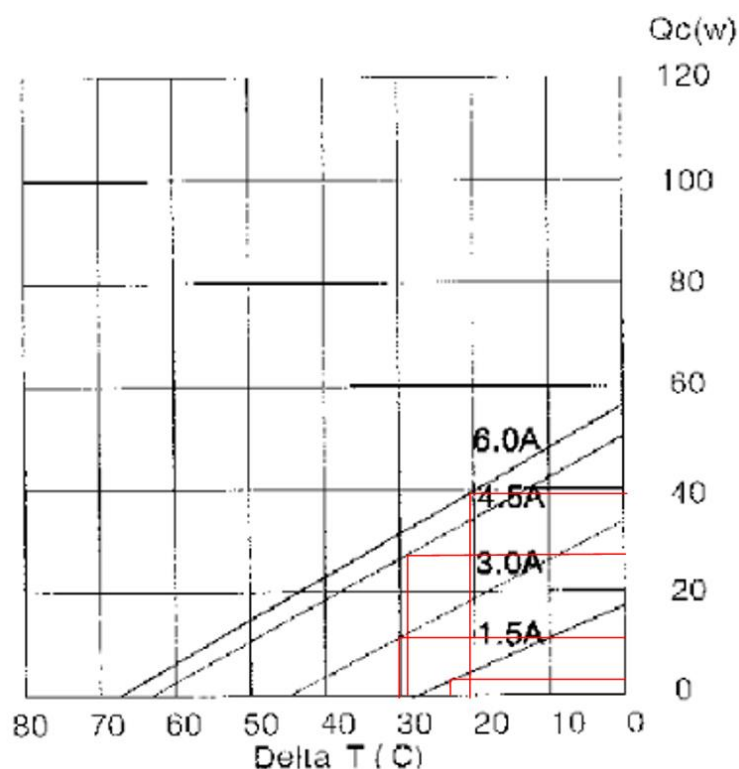


Рисунок 8 – Пример расчета для TEC1-12706

На основе расчета была составлена таблица 2, в ней отражен максимальный перепад температур dT и при каком токе он достигается.

Таблица 2 – Расчетные данные TEC1-12706

$dT_{max}, ^\circ C$	23.6	30	29	20.5
I, A	1.5	3	4.5	6

Полученные данные занесли в САПР «MathCad», для того чтобы получить наглядный пример графика изменения температуры при заданном токе, представленный на рисунке 9.

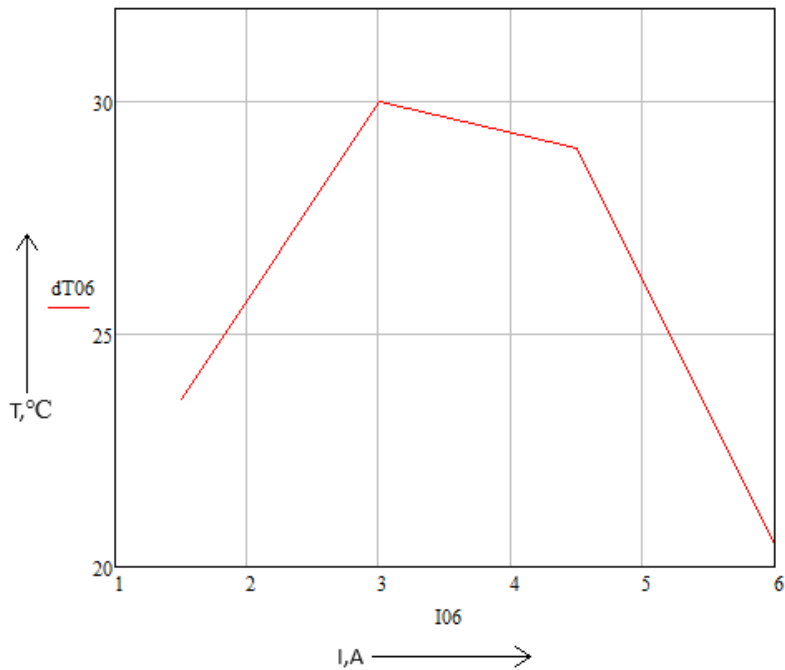


Рисунок 9 – График зависимости dT от I модели ТЕС1-12706

Указанный расчет дает максимально возможный перепад температур в ЭП, но не отражает реальный перепад температур, при работе на термокамеру с ее тепловым сопротивлением R_{sn} . Для уточнения результата расчета нужно изменить условие 2 в допущениях, например, принять сопротивление термокамеры не бесконечным, а равным $R_{sn} = 1/G_{sn}$. Тогда схема замещения, преобразованная из схемы по рисунок 6а с учетом элемента $1/G_{sn}$ на рисунке 3 будет выглядеть как изображенная на рисунке 10а, а после преобразования, как на рисунке 10б.

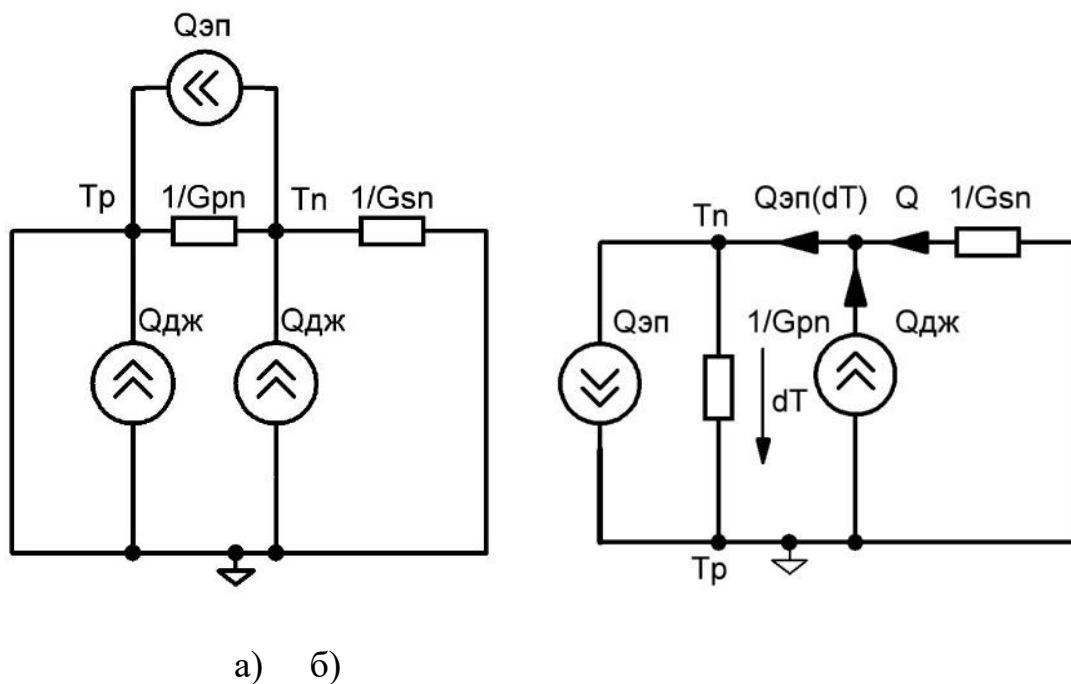


Рисунок 10 – Последовательное преобразование схемы с учетом конечного $R_{сн}$.

Для расчета этой схемы, состоящей из параллельного соединения трех элементов: ЭП, источника $Q_{дж}$ и сопротивления $1/G_{сн}$ нужно характеристику ЭП $Q_{эп}(dT)$ перенести вниз по вертикальной оси на $Q_{дж}$ и пересечь с линией, соответствующей сопротивлению $R_{сн} = 1/G_{сн}$ (рисунок 11).

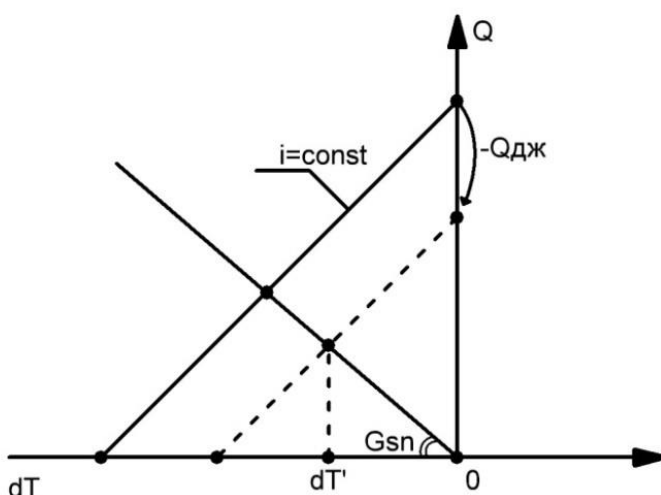


Рисунок 11 – Пример расчета

Первым преобразованием определяется поток через сопротивление $R_{сн} = 1/G_{сн}$: $Q = Q_{эп}(dT) - Q_{дж}$ исходя из первого закона Кирхгофа для

электрической цепи, вторым - приравниванием характеристики R_{sn} находится решение схемы и искомый перепад температур dT' . Это $dT' = T_n - T_p$ при идеальном радиаторе, как и в предыдущем случае совпадает с перепадом температур между камерой холодильника и внешней средой, т.е. является искомым перепадом. На рисунке 12 приведен пример расчета для модели TEC1-12706.

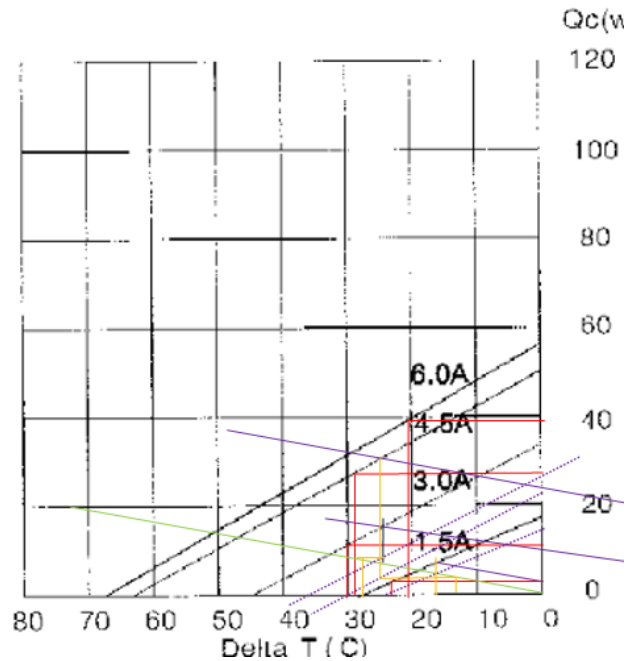


Рисунок 12 – Пример расчета для TEC1-12706 с учетом реальных условий

В таблице 3 представлены реальный рассчитанный перепад температур при заданном токе I .

Таблица 3 – Расчетные данные для TEC1-12706 с учетом реальных условий

dT реальное, °C	17.5	27.6	13
I , A	1.5	3	4.5

На основе анализа ЭП были проведены расчеты по схеме рисунке 6 при $R_{sn}=0,11$ град/Вт соответствующей реальной термокамере из пенополистирола размерами 150x150x400 мм, а также составлена таблица для разных ЭП. Дополнительно проделывая расчеты для разных токов, были определены токи I_{max} на которых достигается максимальный перепад

температур dT_{max} , а также был подсчитана удельная температурная стоимость ЭП. Это экономический показатель, влияющий на выбор ЭП. Чем он ниже, тем дешевле стоимость достижения одного и того же перепада температур при заданном тепловом сопротивлении термокамеры. (взята минимальная и максимальная цена в долларах, курс доллара принят за 91 рубль)

Таблица 4 - Сравнительная характеристика моделей элемента Пельтье

Электронный компонент	Максимальный ток(А)	Максимальная разница температур $dT(^{\circ}C)$	Ценовой диапазон (\$)	I_{max} , А	dT_{max}	Руб/ dT
TEC1-12703	3	68	4-13.8	1.6	26.3	13.8-47.7
TEC1-12705	5.3	66	3.4-14.5	2.5	29.2	11.6-45.2
TEC1-12706	6.4	66	3-11.5	3	30	9.1-34.9
TEC1-12708	8.5	66	5-19.5	4	27.3	16.6-65
TEC1-12710	10.5	66	17-32	4	31.4	49.3-92.7
TEC1-12715	15	70	6.8-141	9	44.5	13.9-288.3
TEC1-12730	30.5	68	73.8-122	12	30	223,7-370.1

Таким образом, благодаря анализу и графическим вычислениям, можно наглядно увидеть, что самым лучшим и наиболее выгодным, с экономической точки зрения, является элемент Пельтье модели TEC1-12706 [22]. Именно он лег в основу модели мобильной термокамеры.

2.3 Выбор термопасты для элемента Пельтье

2.3.1 Проблема выбора

У элемента Пельтье одна сторона работает на нагрев, а вторая на охлаждение. Для того, чтобы охладить нагревающую сторону, необходимо присоединение к элементу радиатора. Чтобы заменить воздух между соприкасающимися поверхностями, необходимо использовать термопасту, которая не только соединяет ЭП с радиатором в единое целое, но и обладает высокой теплопроводностью, а значит тепло будет передаваться с достаточной скоростью, чтобы обеспечить нормальное охлаждение. Кроме того, термопаста обладает в десятки раз большей теплопроводностью, чем воздух.

Термопаста – это многокомпонентный теплопроводящий состав, который наносится между нагреваемым элементом(горячая сторона элемента Пельтье) и отводящим тепло радиатором [11].

2.3.2 Анализ термопаст и выбор самой лучшей и экономически выгодной модели

В современном мире, рынок термопаст очень велик, поэтому необходимо было провести анализ самых популярных моделей, сравнить их характеристики и найти наиболее экономически выгодный и с самыми лучшими характеристиками.

Для того, чтобы более наглядно увидеть технические характеристики и минимальные цены моделей термопаст, была составлена таблица 5 [16].

Таблица 5 – Сравнительная характеристика моделей термопаст

Модель	Цвет	Теплопроводность, Вт/мК	Рабочая температура, °С	Вес, гр	Цена за 1гр, руб
Arctic MX-2 2019	Серый	5.6	От -50 до 150	4	81
Thermalright TF7	Серый	12.8	От -150 до 250	2	179
Zalman ZM-STC8	Серый	8.3	От -40 до 200	1.5	290
ID – Cooling FROST X25	Серый	10.5	От -40 до 180	2	291
AeroCool BARAF	Серый	5.15	От -30 до 280	1	385
STEEL Frost Cuprum	Белый с рыжим оттенком	9.8	От -50 до 150	3	105
AeroCool Cog	Серый	8.5	От -30 до 280	2	310
GD900-1	Серый	6.8	От -50 до 120	7	38

*Цены взяты из маркетплейса «ЯндексМаркет».

*Чем выше коэффициент теплопроводности, тем лучше термопаста проводит тепло.

Таким образом, проведя анализ термопаст, был сделан вывод использовать пасту GD900-1, она обладает теплопроводностью 6.8 Вт/мК, имеет рабочую температура от -50 до 120 °С, а также имеет низкую стоимость на маркетплейсах. Данная паста обладает рядом преимуществ:

- Легкость нанесения и удаления

- Низкая стоимость
- Высокая производительность
- Хорошую плотность

Для улучшения контакта, ЭП будет промазываться термопастой со стороны, прилегающей к радиатору и со стороны, которая прилегает к алюминиевой внутренней поверхности мобильной термокамеры.

3 Проектирование устройства

3.1 Структура устройства

Структура устройства состоит из нескольких основных узлов:

- Термоэлектрический модуль
- Датчик температуры
- Понижающий импульсный преобразователь
- Микроконтроллер ArduinoNano
- Мобильное приложение

Структурная схема устройства изображена на рисунке 13.

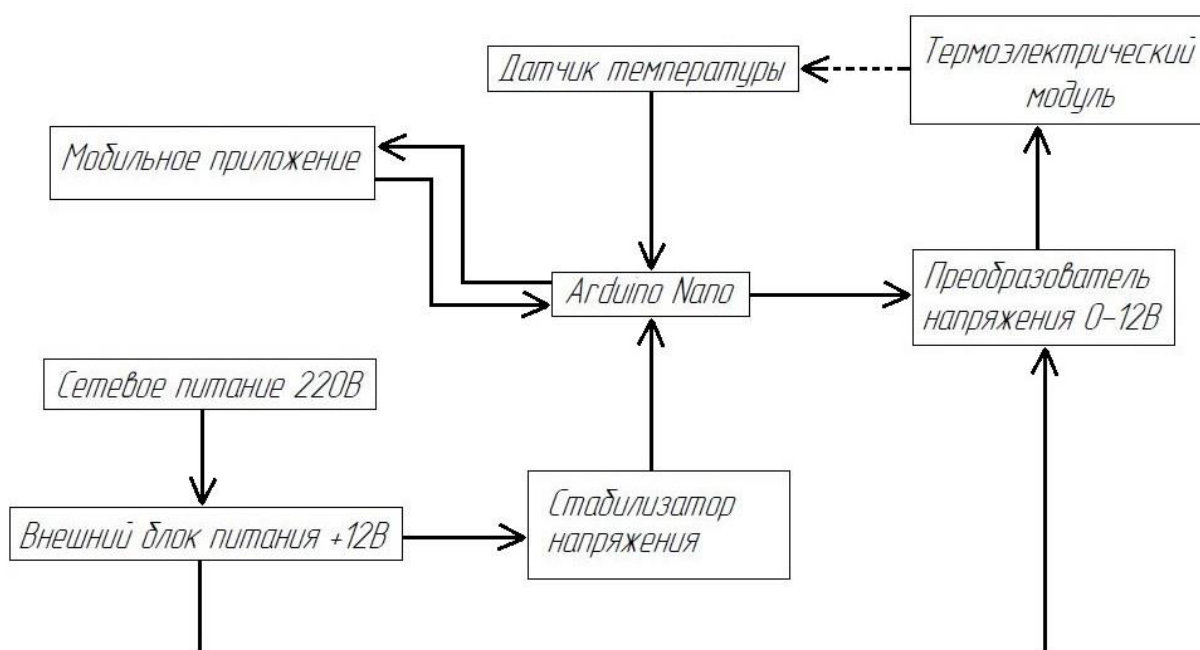


Рисунок 13 – Структурная схема устройства

Электрическая часть устройства состоит из силовой и управляющей части, в первую входит блок питания на +12В и импульсный преобразователь напряжения 0-12В. В управляющей части находится мобильное приложение на смартфон, микроконтроллер ArduinoNano, который генерирует ШИМ сигнал для управления импульсным преобразователем, а также датчик температуры.

3.2 Разработка импульсного понижающего преобразователя напряжения

«Основное отличие импульсных источников питания от линейных (или непрерывного действия) заключается в том, что регулирующий элемент, чаще всего транзистор, работает в ключевом режиме (режиме переключений). То есть большую часть периода работы находится в области отсечки или насыщения, а в активной зоне находится только в момент переключения. Следовательно, средняя за период мощность на рассеиваемая транзисторе будет значительно меньше, чем при работе в линейном режиме. Таким образом, импульсные источники питания по сравнению с линейными имеют более высокий КПД и меньшую массу, и размеры» [4].

Разработка необходима для того, чтобы регулировать работу ЭП. На рисунке 14 представлена схема ИППН 1 рода, в левой части которого находится ШИМ модулятор, а в правой части непосредственно ИППН. Моделирование производилось в SPICE-симуляторе «TINA-TI».

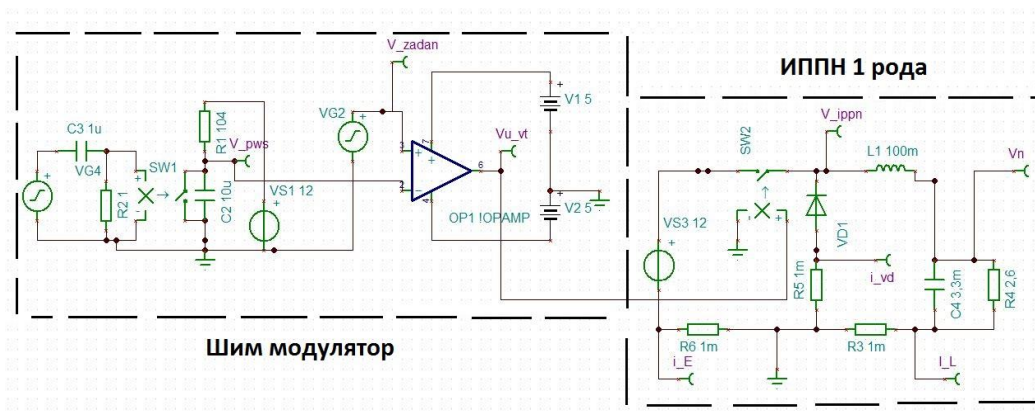


Рисунок 14 – Изображение схемы ИППН в симуляторе «TINA-TI»

Схема ИППН 1 рода состоит из ключа SW2, выполняющий роль, эквивалентную транзистору, двух шунтирующих резистора R6 и R3, нагрузки R4, источника напряжения VS3, диода VD1, катушки индуктивности L1 и конденсатор C4, которые необходимы для уменьшения пульсаций, которые негативно влияют на работу ЭП.

Операционный усилитель OP1 генерирует ШИМ сигнал. При замыкании ключа, ток нарастает и передается в нагрузку и на увеличение энергии в дросселе L1. При размыкании ключа, в дросселе возникает

напряжение равное ЭДС и энергия, накопленная в дросселе, частично поступает в нагрузку, создавая выходное напряжение V_n .

На рисунке 15 представлено изображение осциллограммы из симулятора «TINA-TI». На ней представлена зависимость напряжений. Когда сигнал V_zadan выше, чем V_pws , напряжение Vu_vt , которое подается на ключ $SW2$ повышается, следовательно, когда ниже, напряжение становится равным нулю, создавая ШИМ сигнал. В момент повышения ключ замыкается, в момент понижения открывается.

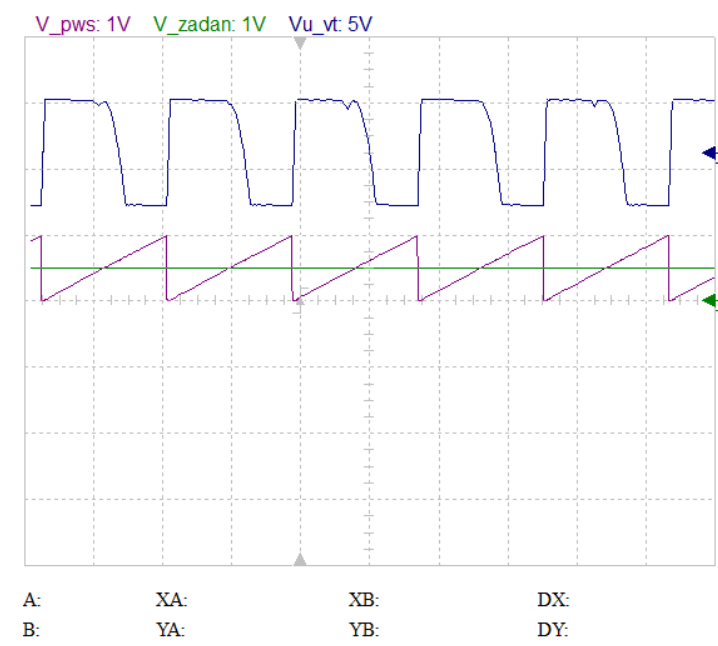


Рисунок 15 – Зависимость напряжений V_zadan , V_pws , Vu_vt

Для улучшения работы ЭП и предотвращения его перегрева, необходимо подобрать правильный дроссель $L1$ и катушку $C4$. Кроме того, необходимо выбрать правильную нагрузку $R4$, сопротивление будет равно сопротивлению ЭП. Оно рассчитывается с помощью характеристик, которые дает производитель, которые представлены на рисунке 16.

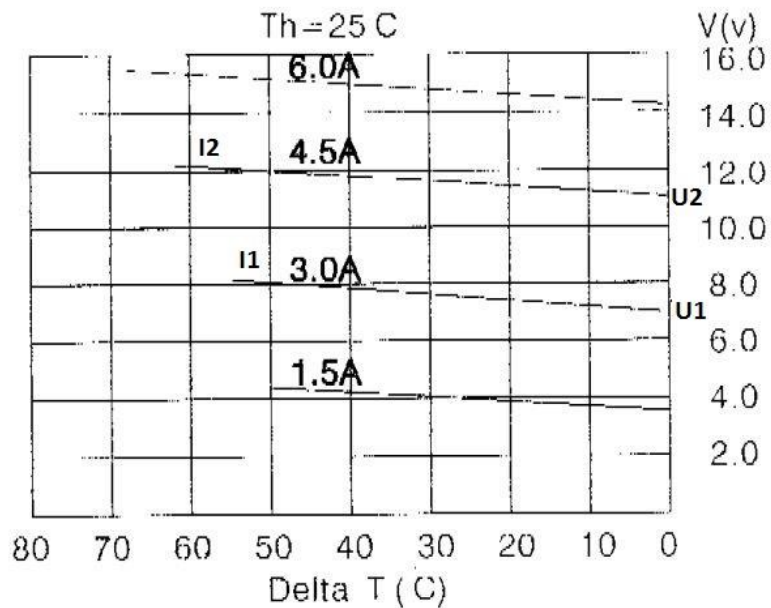


Рисунок 16 – Характеристики ЭП модели TEC1-12706

Зная, что максимальный перепад температур будет достигаться при 3А, высчитаем сопротивление ЭП с помощью закона Ома (формула 2).

$$R = \frac{U_2 - U_1}{I_2 - I_1} = \frac{4}{1,5} = 2.66 \text{ Ом} \quad (2)$$

Номинал катушки L1 подбирался по осциллограмме тока дросселя, чтобы при всем диапазоне регулирования ШИМ пиковые значения ток дросселя не превышал максимальный ток ЭП при максимальном заполнении импульсов ШИМ (рисунок 17, 18, 19). Дополнительно был подобран конденсатор C2, чтобы получать приемлемый уровень пульсации напряжения на ЭП – много меньший, чем среднее значение напряжения на ЭП. Выбор остановился на L1 = 100мГн, а C2 = 3,3Ф.

С увеличением заполнения импульса генератора напряжения VG2 ток на ЭП увеличивается. Чтобы это доказать необходимо проверить это на осциллограмме в симуляторе «TINA-TI». Будем постепенно увеличивать напряжение. Первое значение будет 250мВ. Изображение осциллограммы представлена на рисунке 17.

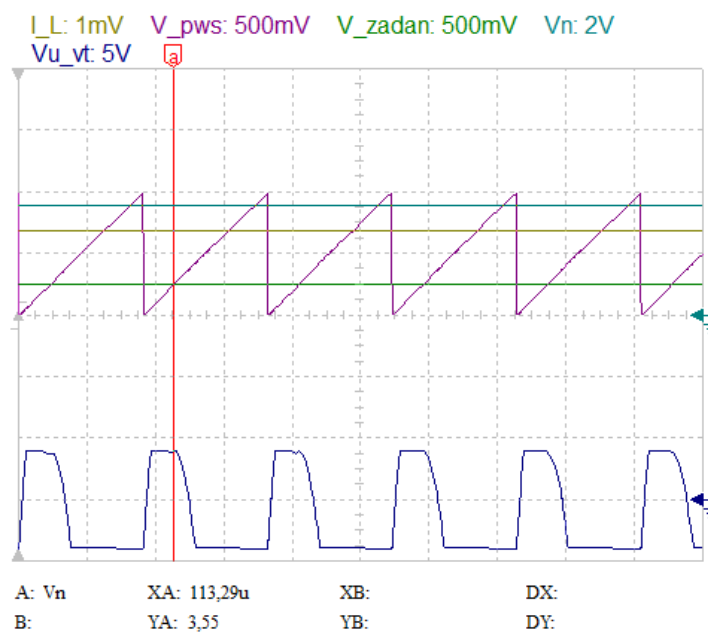


Рисунок 17 – Изображение осциллограммы при $V_{G2} = 250\text{mV}$

Таким образом, был получен результат $V_n = 3,55\text{В}$ при $V_{G2} = 250\text{mV}$, следовательно по закону Ома максимальный ток будет равен $1,37\text{А}$ (формула 3).

$$I_{250} = \frac{U}{R} = \frac{3,55}{2,6} = 1,37 \quad (3)$$

Следующим шагом будем подавать $V_{G2} = 500\text{mV}$. Изображение осциллограммы представлена на рисунке 18.

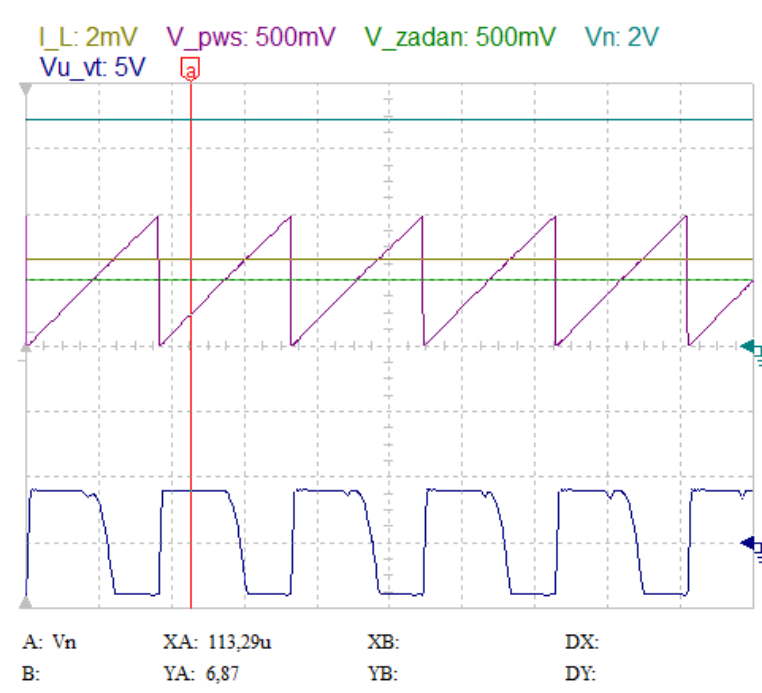


Рисунок 18 - Изображение осциллограммы при $V_{G2} = 500\text{mV}$

Исходя из данных осциллограммы, напряжение $V_n = 6,87В$. Аналогично первому случаю, используя формулы 3, было рассчитано, что максимальный ток достигает $2,64А$.

Следующим шагом, увеличим V_{G2} до $750мВ$. Изображение осциллограммы представлено на рисунке 19.

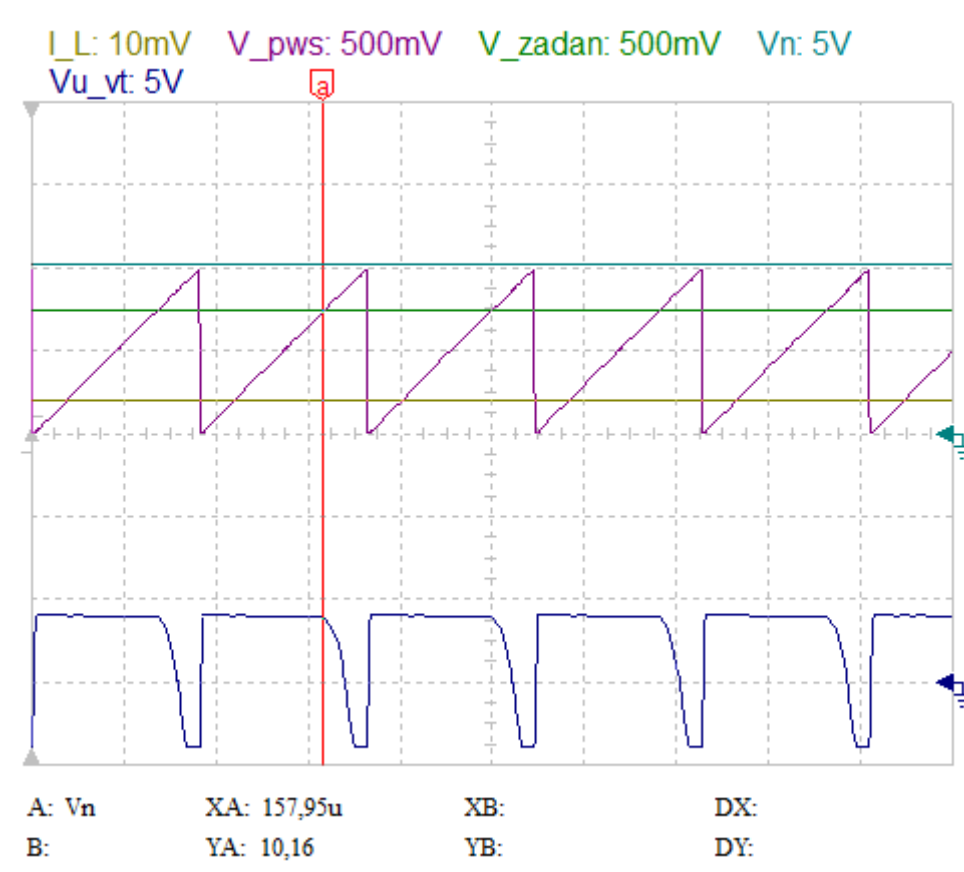


Рисунок 19 - Изображение осциллограммы при $V_{G2} = 750мВ$

Напряжение $V_n = 10,16В$, следовательно, по формуле 3 максимальный ток будет равен $3,9А$.

Завершающим повышением будет $V_{G2} = 1В$, в этот момент ключ будет постоянно полностью открыт, поэтому ШИМ сигнал будет отсутствовать. Максимальный ток в этот момент будет равен $4,6А$. Изображение осциллограммы приведено на рисунке 20.

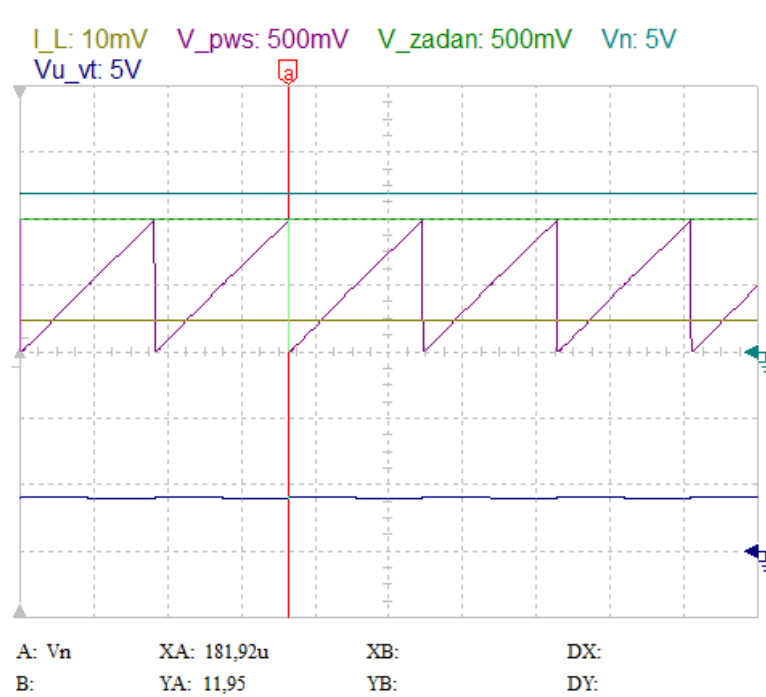


Рисунок 20 - Изображение осциллограммы при $V_{G2} = 1В$

На рисунке 21 приведены ток дросселя I_L и ток ключа I_E , изображение показывает минимальные пульсации, которые не будут влиять на работу ЭП.

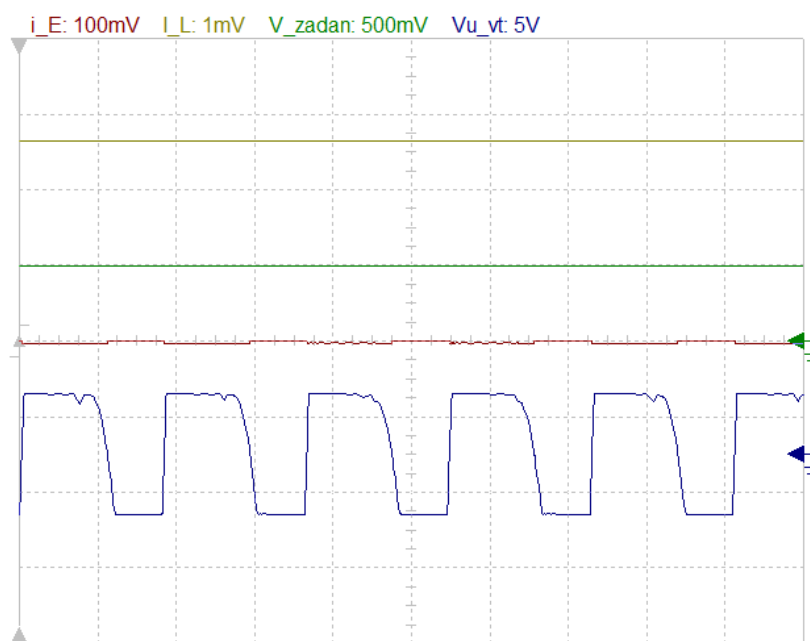


Рисунок 21 - Изображение осциллограммы, демонстрирующее отсутствие пульсаций

Таким образом, была выполнена разработка ИППН 1 разряда, в котором практически отсутствуют пульсации и был выявлен максимальный ток, который равен 4,6А.

3.3 Схема импульсного преобразователя

Прежде чем собирать мобильную термокамеру и тестировать ее, необходимо разработать электрическую схему, позволяющую регулировать работу ЭП. Создание схемы и печатной платы будет производиться в среде автоматизированного проектирования «EasyEda»[19].

Проанализировав несколько вариантов схем, было решено выбрать схему импульсного преобразователя с функцией регулирования напряжения, которая представлена на рисунке 22.

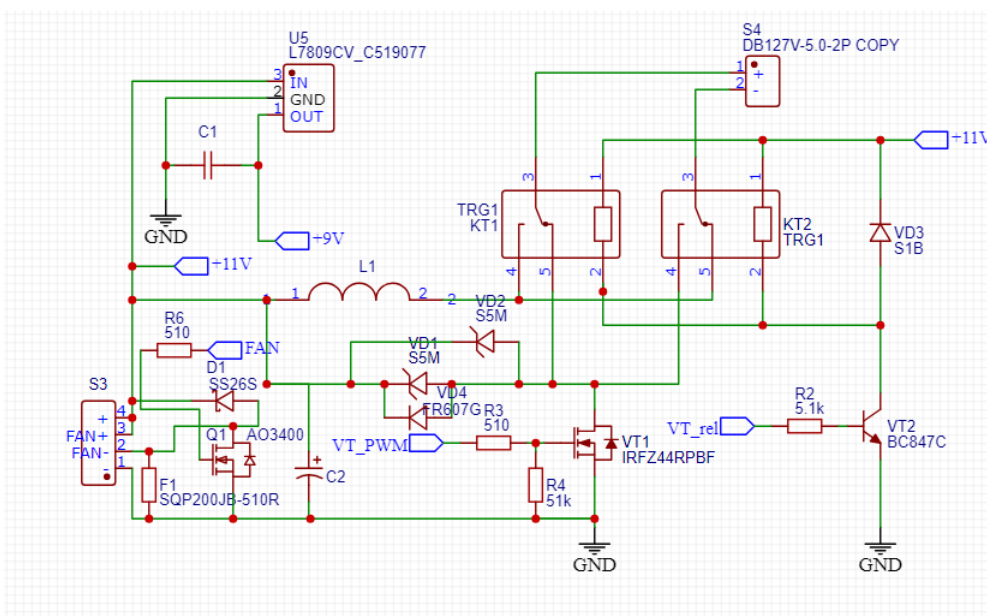


Рисунок 22 – Изображение схемы импульсного преобразователя в среде «EasyEda»

Схема состоит из двух реле KT1 и KT2, силового ключа, ключа VT2, катушки индуктивности L1, обратных диодов: VD1,VD2,VD3. Кроме того, схема включает в себя ряд дополнительных элементов, таких как узел управления группой реле, узел регулирования частоты оборота вентилятора. Реле синхронно переключаются транзистором, который управляется микроконтроллером AtMega328P и меняют полярность тока на ЭП, в следствие чего можно менять режим работы ЭП: нагревать или охлаждать.

3.4 Разработка печатной платы

Электрическая схема, для управления термокамерой представлена на рисунке 23.

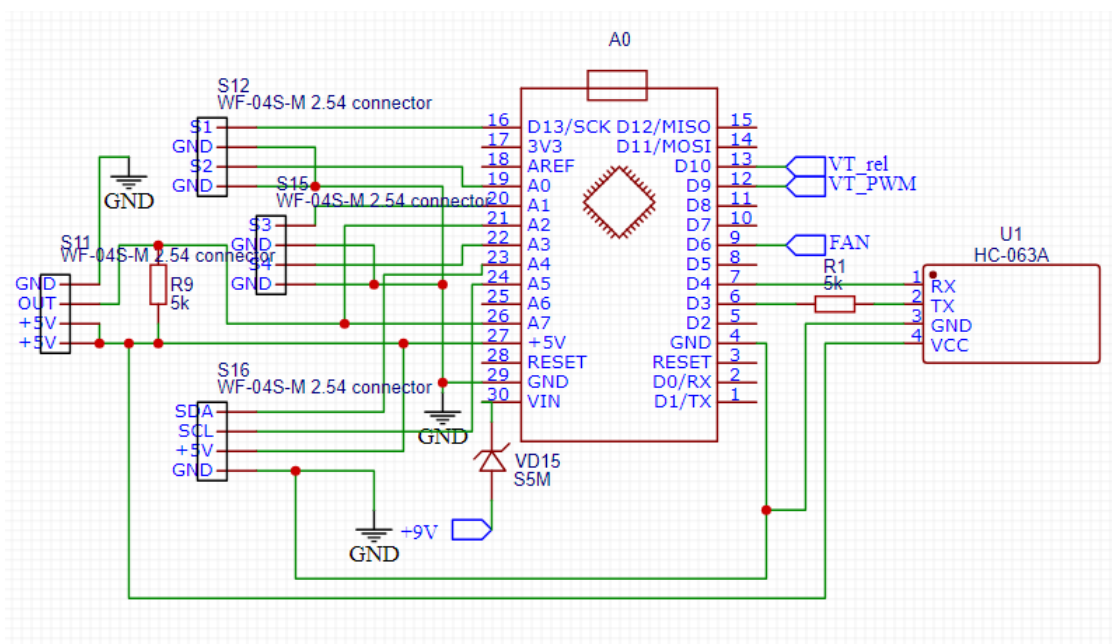


Рисунок 23 – Изображение электрической схемы соединений в среде «EasyEda»

В нее выходит 4 разъема питания WF-04S-M-2.54, плата ArduinoNano и Bluetooth– модуль HC-06.

Среда «EasyEda» позволяет практически автоматически построить 3D– модель печатной платы, исходя из данных нарисованной электрической схемы соединений.

После сборки электрической схемы, необходимо выбрать в графическом интерфейсе вкладку «Дизайн» и нажать «Преобразовать схему в печатную плату». После этого откроется окно трассировки, где можно настроить размер печатной платы, разместить элементы наиболее лучшим образом на плате и выполнить трассировку. Габариты платы 90×90 мм, расстояние между отверстиями 70 мм, отверстия диаметром 2мм, толщина платы 1,6мм. Результат показан на рисунке 24.

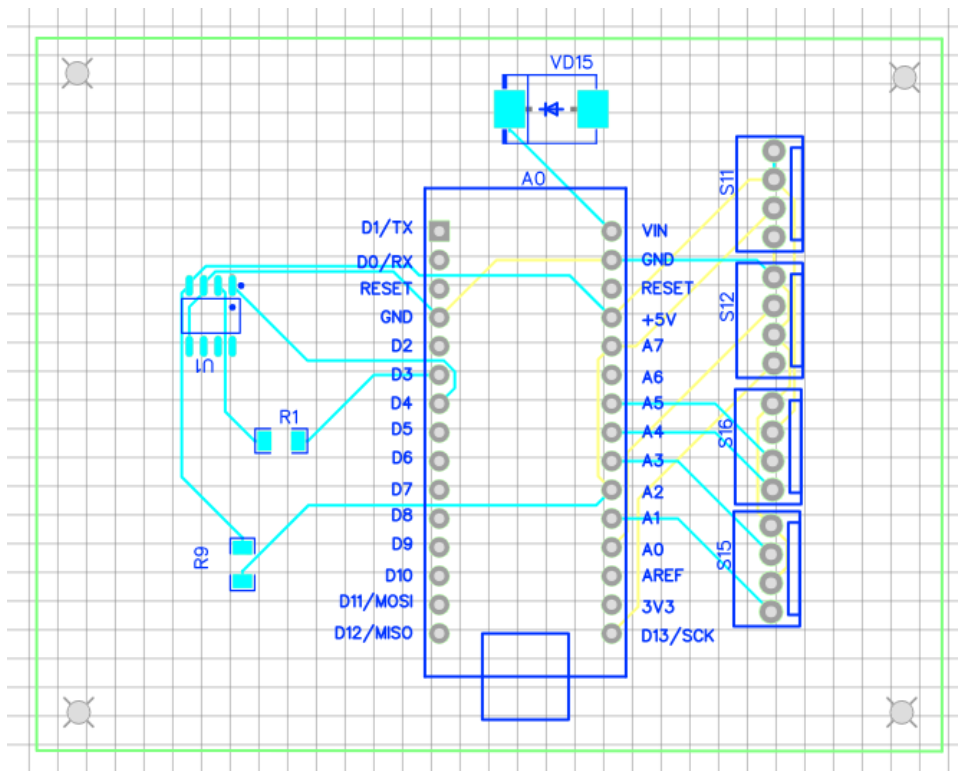


Рисунок 24 – Результат трассировки

Следующим шагом необходимо в окне «Вид» нажать «3D» и 3D-модель нашей платы автоматически построится, согласно результатам трассировки. Результат представлен на рисунке 25.

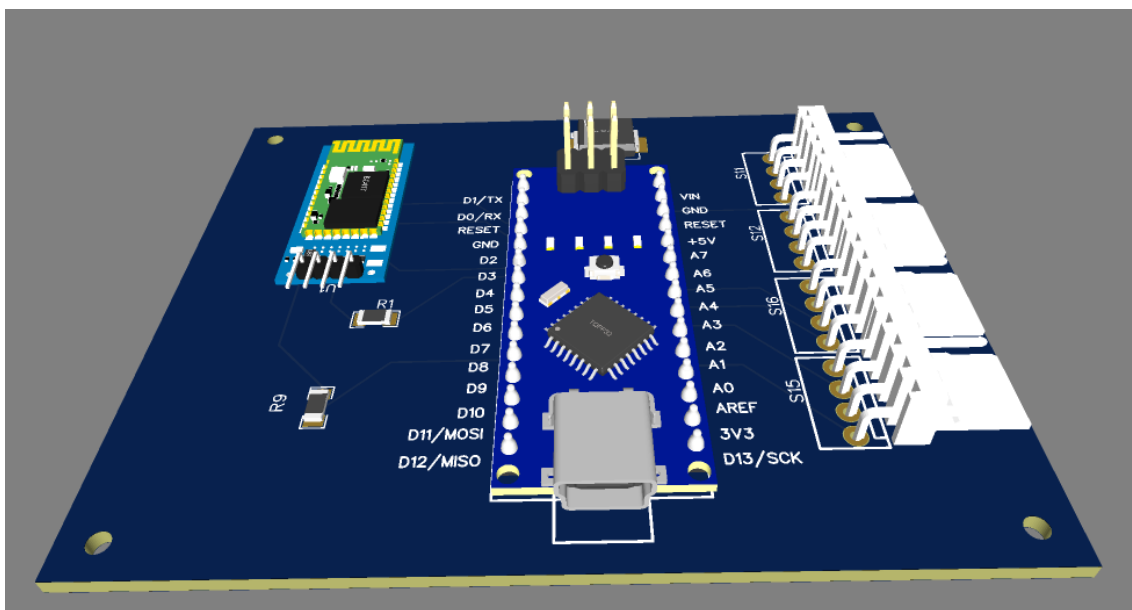


Рисунок 25 – 3D-модель печатной платы в среде «EasyEda»

Таким образом разработана 3D-модель печатной платы, которую можно сразу заказать на ресурсе «EasyEda» с фабричного производства.

3.5 Экспериментальное исследование тепловых процессов

3.5.1 Первый опыт по определению эффективного метода подачи напряжения и тока

Для того, чтобы получить максимальный перепад температур dT , необходимо выяснить какой метод подачи тока и напряжения является наиболее эффективным – скачкообразная подача или сглаженная.

Первым опытом было решено сгладить напряжение путем добавления конденсатора $C3$ и катушки индуктивности $L1$ и подавать ток скачкообразно. Итоговое изображение схемы из «EasyEda» представлена на рисунке 26.

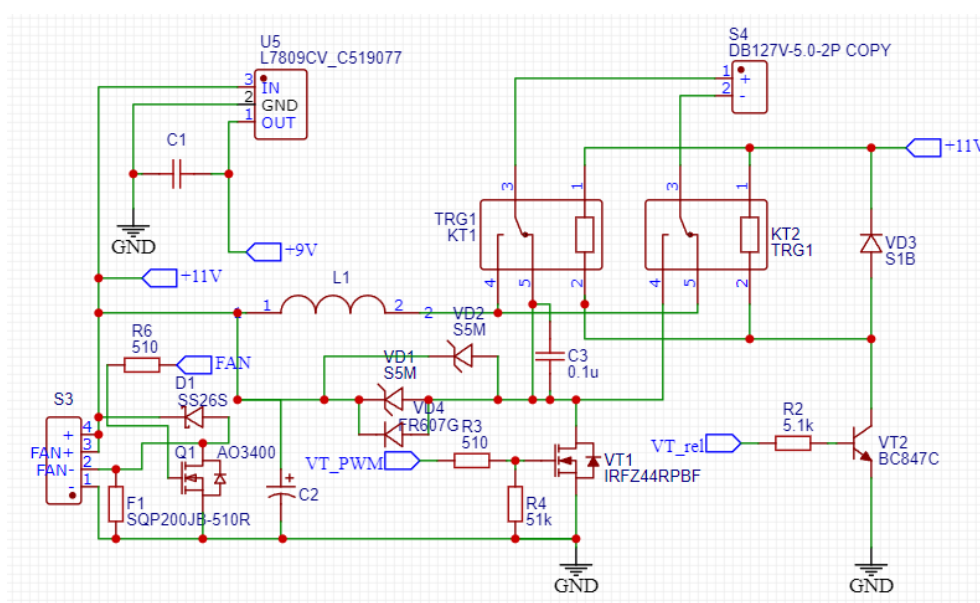


Рисунок 26 – Изображение модернизированной схемы из «EasyEda»

Первым делом необходимо написать программный код для платы ArduinoNano на языке программирования C++. «Наиболее распространенным подходом к программированию является использование среды разработки Arduino IDE, в которой используется язык программирования C» [17]. Объявим необходимые библиотеки, которые приведем в листинге 1:

Листинг 1

```
#include<TimerOne.h> - библиотека для управления таймером
#include<OneWire.h> - библиотека для работы с шиной OneWire
#include<DallasTemperature.h> - библиотека для работы с датчиками температуры
DallasDS18B20
#include<SoftwareSerial.h> - библиотека для создания программного
последовательного порта
#define ciklvalue 10 – количество точек для снятия данных
```

Следующим шагом необходимо инициализировать объекты и переменные (листинг 2):

Листинг 2

```
OneWire oneWire(A2); // создаем объект oneWire для работы с датчиком,
подключенным к порту A2
DallasTemperature ds(&oneWire); // создаем объект ds для работы с датчиком
температуры
byte num; // переменная для хранения количества подключенных датчиков
long T = 0; // переменная для хранения времени в секундах, прошедшего с начала
работы
```

Затем необходимо объявить переменные для PID – регулятора и других расчетов (листинг 3):

Листинг 3

```
int tz = 25; // заданная температура
float kP = 200, kI = 0, kD = 0; // коэффициенты PID-регулятора
int constrI = 1000; // ограничение интегральной составляющей
float t1 = 0; // температура датчика 1
float t2 = 0; // температура датчика 2
float I = 0; // ток
float delta_t = 0; // разница температур
float Q, t; // переменные для расчетов
float U; // напряжение
float PWMParam; // глобальная переменная для заполнения ШИМ
int pws = 50; // начальное значение ШИМ
int dT = 1; // период расчета PID-регулятора
int pid, PID = 0; // расчетный и ограниченный уровень заполнения нагревом
float err = 0, errLast = 0; // ошибка и предыдущая ошибка для PID
float dif = 0; // дифференциальная составляющая
float integral = 0; // интегральная составляющая
float myTimer1, myTimer2, myTimer3; // таймеры
int mode = 2; // режим работы
const int pwmPin = 9; // пин для ШИМ
```

После этого вызывается функция `setup()`, в которой будет определяться режим работы выводов, запуск используемых библиотек, инициализация переменных (листинг 4):

Листинг 4

```
void setup() {
  Serial.begin(9600); // инициализация последовательного порта со скоростью 9600
  бод
  TCCR1A = 0b00000011; // настройка таймера 1: 10-битный режим ШИМ
  TCCR1B = 0b00001001; // настройка таймера 1: быстрый ШИМ с делителем 1
```

```

ds.begin(); // инициализация датчиков температуры
num = ds.getDeviceCount(); // получение количества подключенных датчиков
Serial.print("Sensortempnum="); Serial.println(num); // вывод количества датчиков в
монитор порта
pinMode(3, OUTPUT); // настройка пина 3 как выход
pinMode(10, OUTPUT); // настройка пина 10 как выход
pinMode(9, OUTPUT); // настройка пина 9 как выход
pinMode(pwmPin, OUTPUT); // настройка пина для ШИМ как выход
ds.requestTemperatures(); // запрос температуры с датчиков
t1 = ds.getTempCByIndex(0); // получение температуры с первого датчика
t2 = ds.getTempCByIndex(1); // получение температуры со второго датчика
digitalWrite(10, LOW); // выключение нагрева (охлаждение)
// Увеличение значения ШИМ с выводом промежуточных значений
for (int pwmValue = 0; pwmValue<= 1023; pwmValue += 102) {
analogWrite(pwmPin, pwmValue); // установка значения ШИМ
delay(3000); // задержка 3000 миллисекунд
printIntermediateValues(); // вывод промежуточных значений
}
myTimer1 = millis(); // инициализация таймера 1
myTimer2 = millis(); // инициализация таймера 2
myTimer3 = millis(); // инициализация таймера 3
}

```

Далее необходимо вызвать функцию `loop()`, для того, чтобы выполнять определенный цикл кода несколько раз (листинг 5):

Листинг 5

```

voidloop() {
ds.requestTemperatures(); // запрос температуры с датчиков
I = analogRead(A4); // чтение аналогового значения тока
if (millis() - myTimer3 >= 7200000) { // проверка, прошло ли 2 часа
myTimer3 = millis(); // сброс таймера 3
if (pws< 1024) pws += 100; else pws = 1023; // увеличение значения ШИМ
analogWrite(9, pws); // установка значения ШИМ
Serial.print("pws="); Serial.println(pws); // вывод значения ШИМ
printIntermediateValues(); // вывод промежуточных значений
}
I = (((5 * I / 1024) - (2.47 - 2.28) - 2.3) / 0.188) - 0.05; // пересчет тока
if (millis() - myTimer1 >= 200) { // проверка, прошло ли 200 миллисекунд
myTimer1 = millis(); // сброс таймера 1
delta_t = t1 - t2; // расчет разницы температур
t1 = t1 * 0.2 + ds.getTempCByIndex(0) * 0.8; // фильтрация температуры горячей
стороны
t2 = t2 * 0.2 + ds.getTempCByIndex(1) * 0.8; // фильтрация температуры холодной
стороны
}
if (millis() - myTimer2 >= 3000) { // проверка, прошло ли 3000 миллисекунд
myTimer2 = millis(); // сброс таймера 2
}
}

```

```

printIntermediateValues(); // вывод промежуточных значений
Serial.print("$"); // начальный символ для строки данных
Serial.print(t1); Serial.print(" "); // температура датчика 1
Serial.print(t2); Serial.print(" "); // температура датчика 2
Serial.print(delta_t); Serial.print(" "); // разница температур
Serial.print(I); Serial.print(" "); // ток
Serial.print(";"); // конечный символ
}
}

```

Также необходима функция для вывода промежуточных значений (листинг 6):

Листинг 6

```

void printIntermediateValues() {
Serial.print("$"); // начальный символ
Serial.print(t1); Serial.print(" "); // температура датчика 1
Serial.print(t2); Serial.print(" "); // температура датчика 2
Serial.print(delta_t); Serial.print(" "); // разница температур
Serial.print(I); Serial.print(" "); // ток
Serial.print(";"); // конечный символ
}

```

Блок-схема программы первого опыта представлена на рисунке 27.

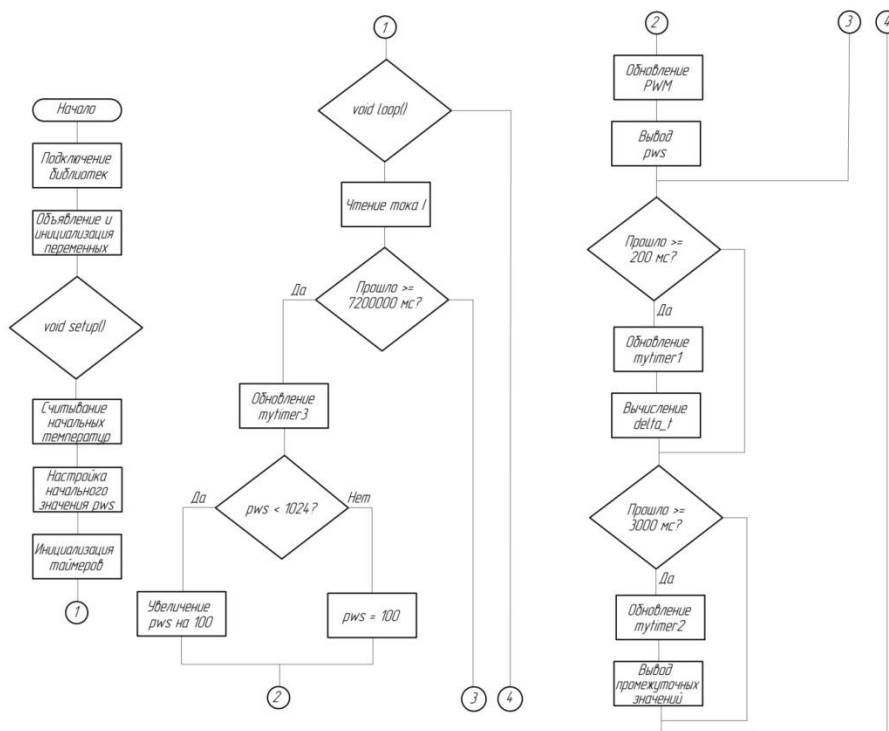


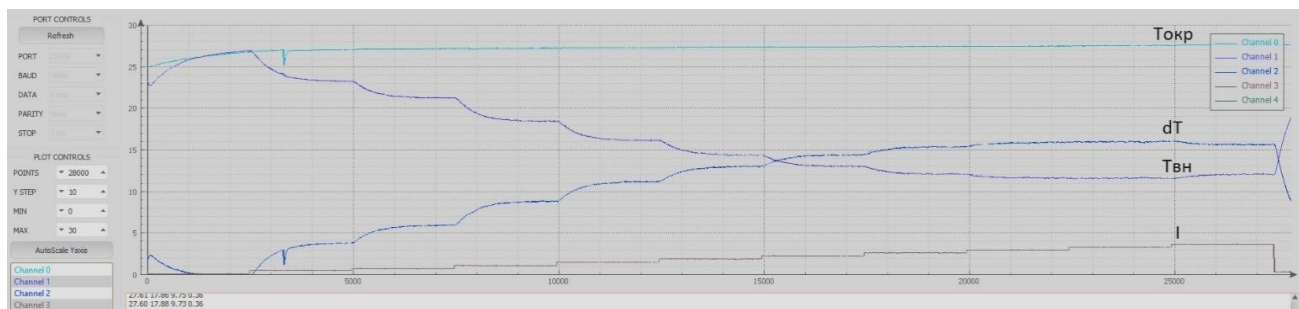
Рисунок 27 – Блок-схема программы для первого опыта

Данный программный код каждые 90 минут менял значение ШИМ сигнала на +100 единиц из 1024, за это время разница температур должна достигать максимальных значений. Так как при подаче постоянного тока разница температур между внешней и внутри термокамеры меняется не скачком, а плавно увеличивается. Полная программы приведен в приложении А.

Для вывода полученных данных в графическое изображение, необходимо воспользоваться программным обеспечением «Serial Port Plotter». С помощью него данные будут сразу строиться в виде графиков.

Загрузив написанный код в среду разработки ArduinoIDE, выбрав используемую плату, необходимо дождаться завершения полного цикла работы программы.

В конечном итоге, было снято 11 точек. Каждые 90 минут программа повышала скачком ток. Каждые 2 секунды замерялась температура внутри термокамеры и окружающей среды. Из опыта было получено семейство графиков первого эксперимента, представленный на рисунке 28.



Токр – график температуры окружающей среды

dT – график разницы температур между внутренней и окружающей

Твн – график температуры внутри термокамеры

I – график изменения постоянного тока

Рисунок 28 – Семейство графиков первого опыта

Кроме того, используя полученные данные, была построена зависимость dT от I в САПР MathCad. График принимает ступенчатый вид,

так как горизонтальное изменение это скачок подаваемого тока, а вертикальное это увеличение dT при заданном токе (рисунок 29).

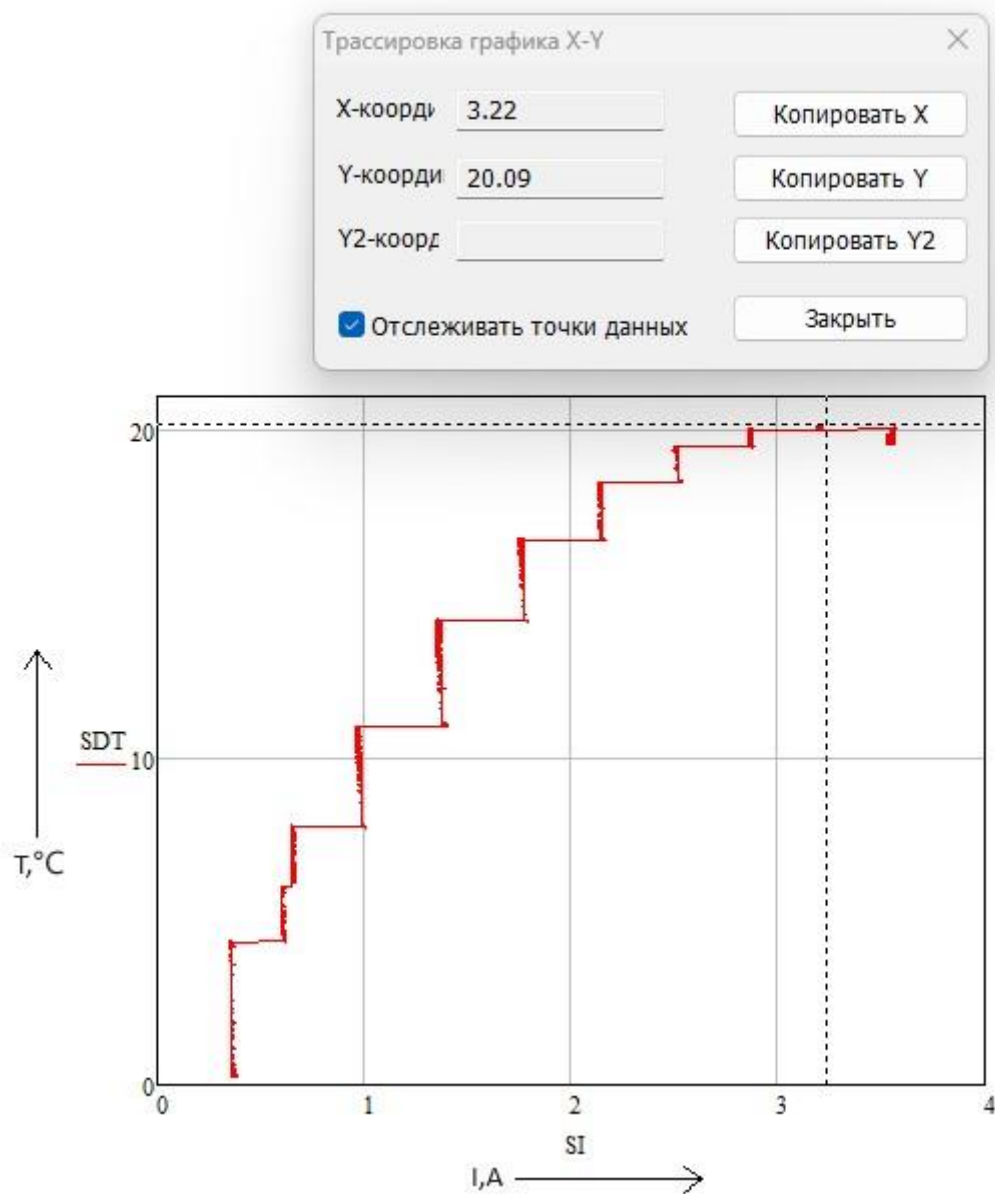


Рисунок 29 – Результат первого опыта

Воспользовавшись инструментом «Трассировка», было выявлено, что максимальный перепад температур достигает $20,09^{\circ}\text{C}$ при токе равном $3,22\text{A}$.

Следующим шагом было решено проверить, совпадут ли результаты, если подавать ток существенно меньшим шагом. Каждые 50 секунд увеличивался ШИМ на +1 значение из 1024, каждые 2 секунды считывалась температура внутренней и окружающей среды. Часть кода, в которую были

внесены изменения представлена в листинге 7. Полный код программы представлен в приложении Б.

Листинг 7

```

voidloop() {
ds.requestTemperatures();
I=analogRead(A4);
if (millis() - myTimer1 >= 50000) {
    myTimer1 = millis();
if(pws<1024 && stop==0) pws=pws+1; else {pws=0; stop=1;}
analogWrite(9, pws);
}
}
    
```

Блок-схема программы для второго случая первого опыта представлена на рисунке 30.

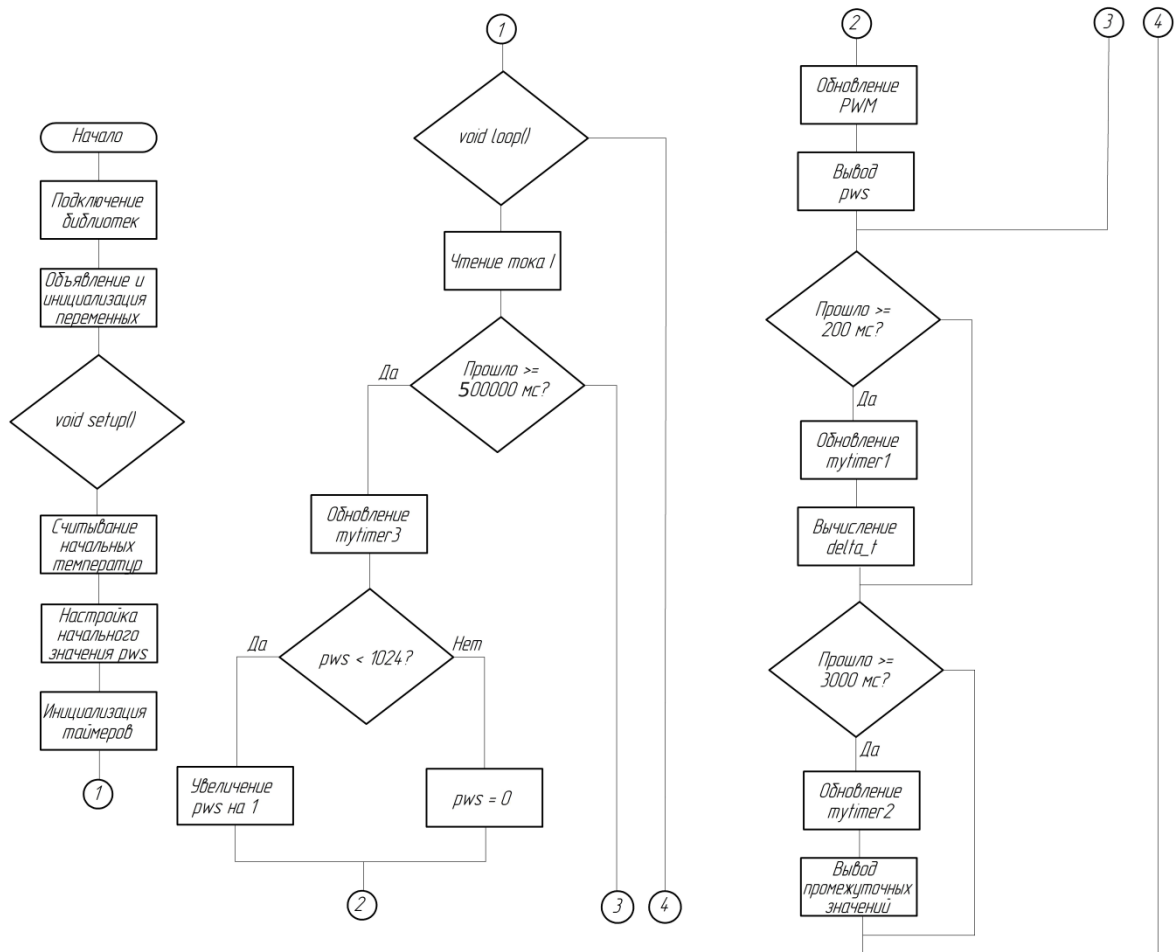


Рисунок 30 – Блок-схема программы для второго случая первого опыта

Аналогично первому случаю, данные автоматически вносили в «Serial Port Plotter», полученные семейства графиков представлены на рисунке 31.

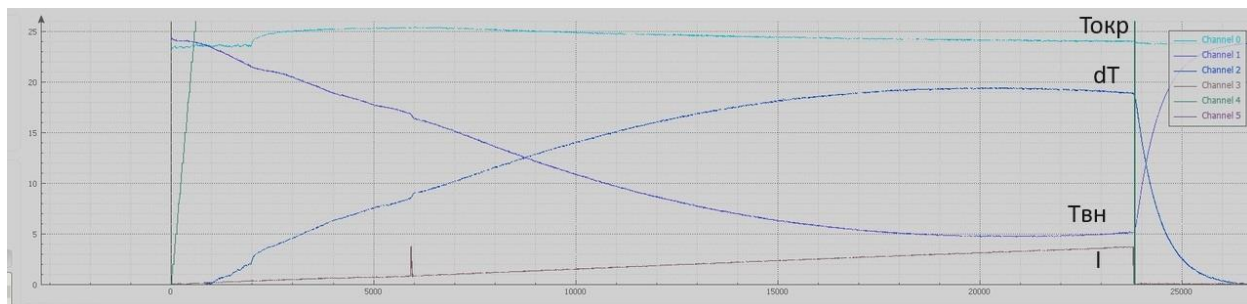


Рисунок 31 – Семейство графиков второго случая

Токр – график температуры окружающей среды

dT – график разницы температур между внутренней и окружающей

Твн – график температуры внутри термокамеры

I – график изменения постоянного тока

Занеся полученные данные в САПР MathCad, также построили зависимость dT от I (рисунок 32).

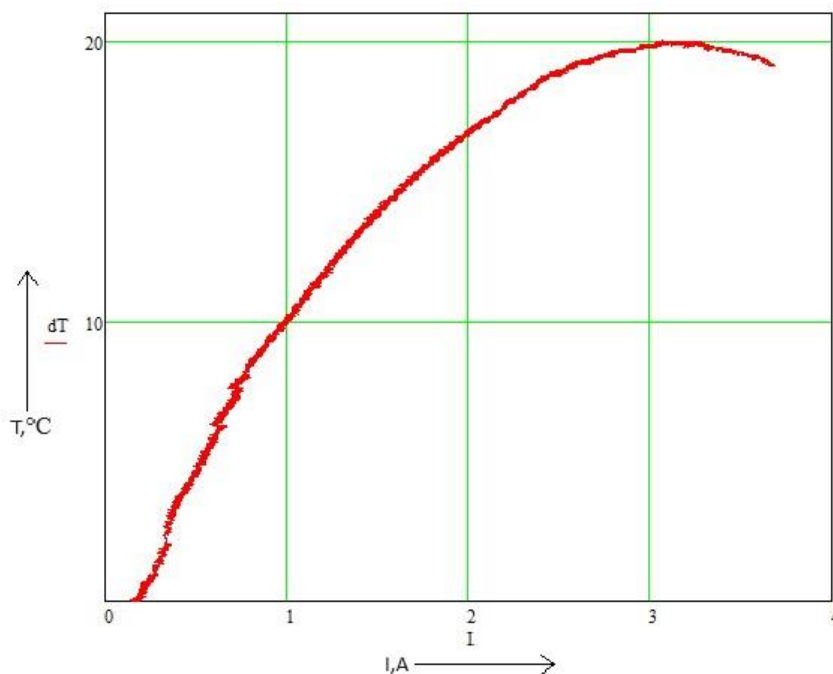


Рисунок 32 – Зависимость dT от I во втором случае

Используя «Трассировку», аналогично первому случаю, а также поместив две кривые на один график (рисунок 33) определили, что максимальная $dT = 20,08^{\circ}\text{C}$ достигается при $I = 3,22\text{A}$.

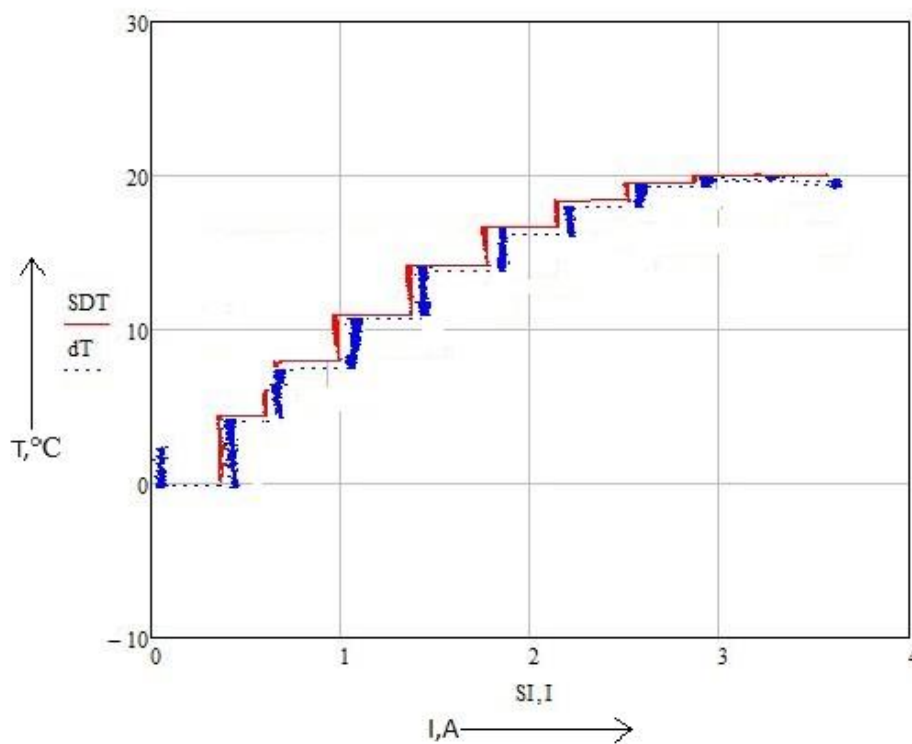


Рисунок 33 – Зависимость dT от I в обоих случаях первого опыта

Следовательно метод подачи тока никак не влияет на полученный результат, в обоих случаях максимальный перепад температур $dT=20,09^{\circ}\text{C}$, который достигается при токе равном $3,22\text{A}$.

3.5.2 Второй опыт по определению эффективного метода подачи напряжения

Вторым опытом было решено определить влияние подачи напряжения ШИМ формы, а не сглаженного как в первом опыте. Напряжение подавалось ШИМ сигналом, за счет удаления из схемы конденсатора $C3$ и добавления вместо катушки индуктивности $L1$ закоротки. Измененное изображение схемы из редактора «EasyEda» представлено на рисунке 34.

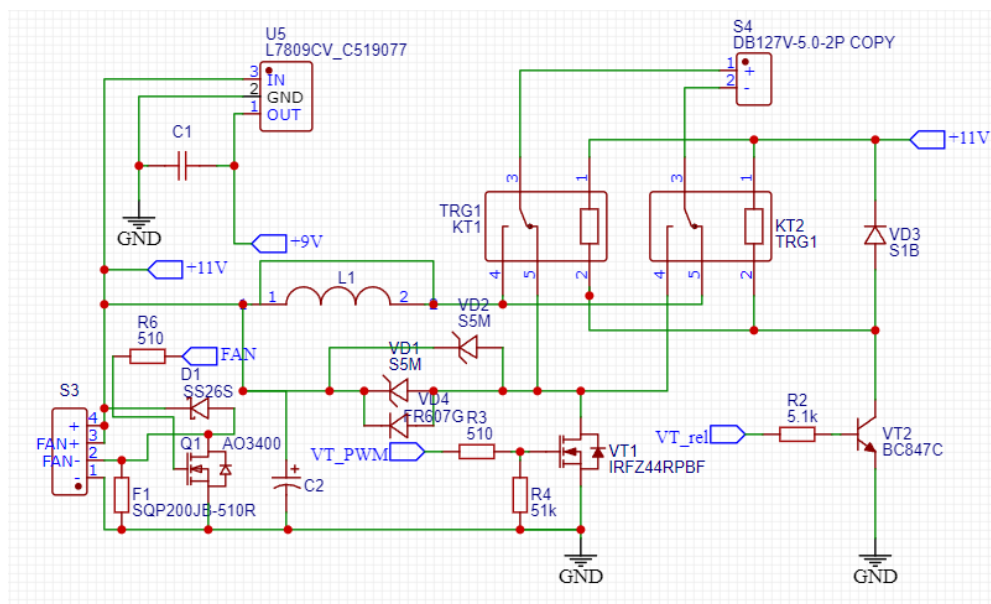


Рисунок 34 – Изображение измененной схемы второго опыта

После изменения схемы, была запущена программа из первого случая первого опыта – метод подачи тока скачкообразно. Данные снимались также с помощью приложения «Serial Port Plotter» и были занесены в САПР «MathCad» для построения зависимости $dT_{\text{от } I}$, которая представлена на рисунке 35.

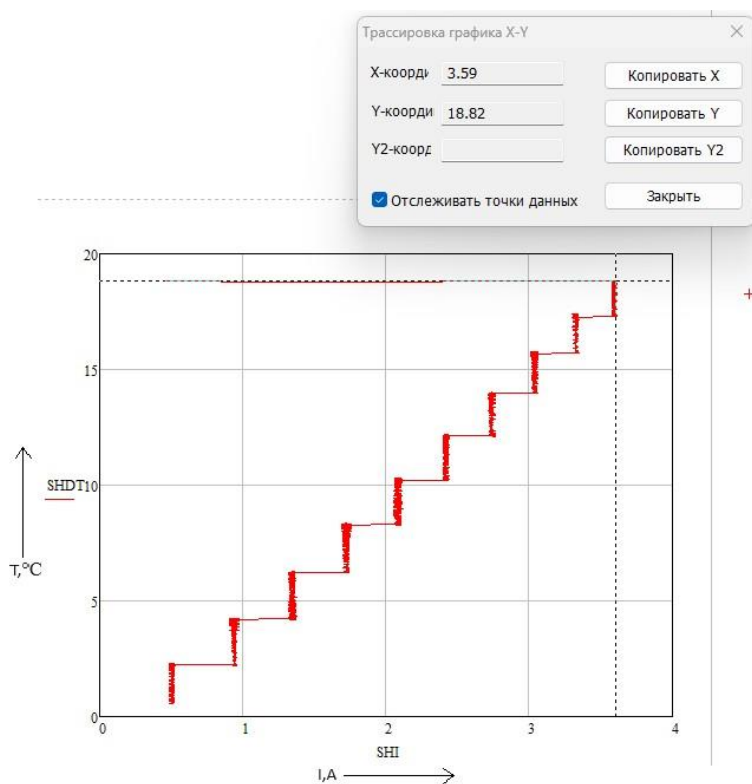


Рисунок 35 – Зависимость dT от I во втором опыте

Воспользовавшись инструментом «Трассировка», было выявлено, что максимальный перепад температур достигает 18,82°C при токе равном 3,59А.

3.5.3 Аппроксимация полученных функций

Следующим шагом было решено осуществить аппроксимацию, полученных после проведения опыта, функций. Это необходимо сделать для того, чтобы понять максимальную разницу между сглаженной и скачкообразной подачей напряжения.

Для этого необходимо задать аппроксимирующую функцию с помощью формулы. Формула аппроксимирующей функции для первого опыта представлена в формуле 4.

$$dT_1(I) = (-1,962 \times I^2 + 12,058 \times I) \times 1,081 \quad (4)$$

Таким образом, наложение аппроксимирующей функции на кривую из первого опыта представлено на рисунке 36.

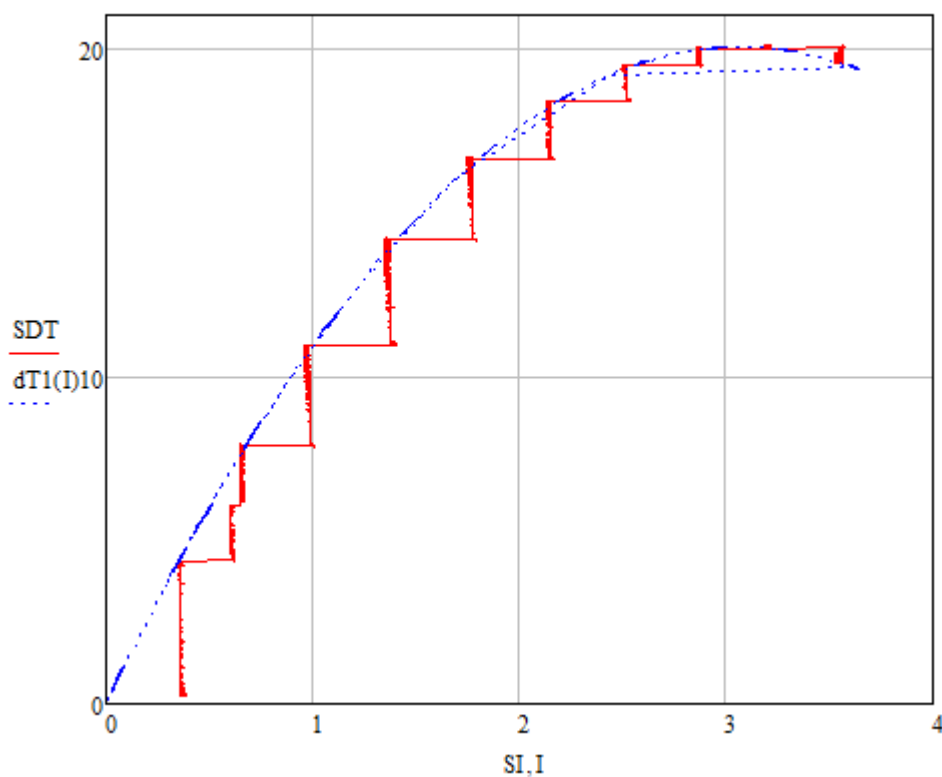


Рисунок 36 – Наложение аппроксимирующей функции на кривую из первого опыта

Аналогично для второго опыта приведем формулу аппроксимирующей кривой, представленную в формуле 5.

$$dT2(I) = (1,75 \times I^2 + 26,08 \times I) \times 0,165 \quad (5)$$

Таким образом, наложение аппроксимирующей функции на кривую из второго опыта представлена на рисунке 37.

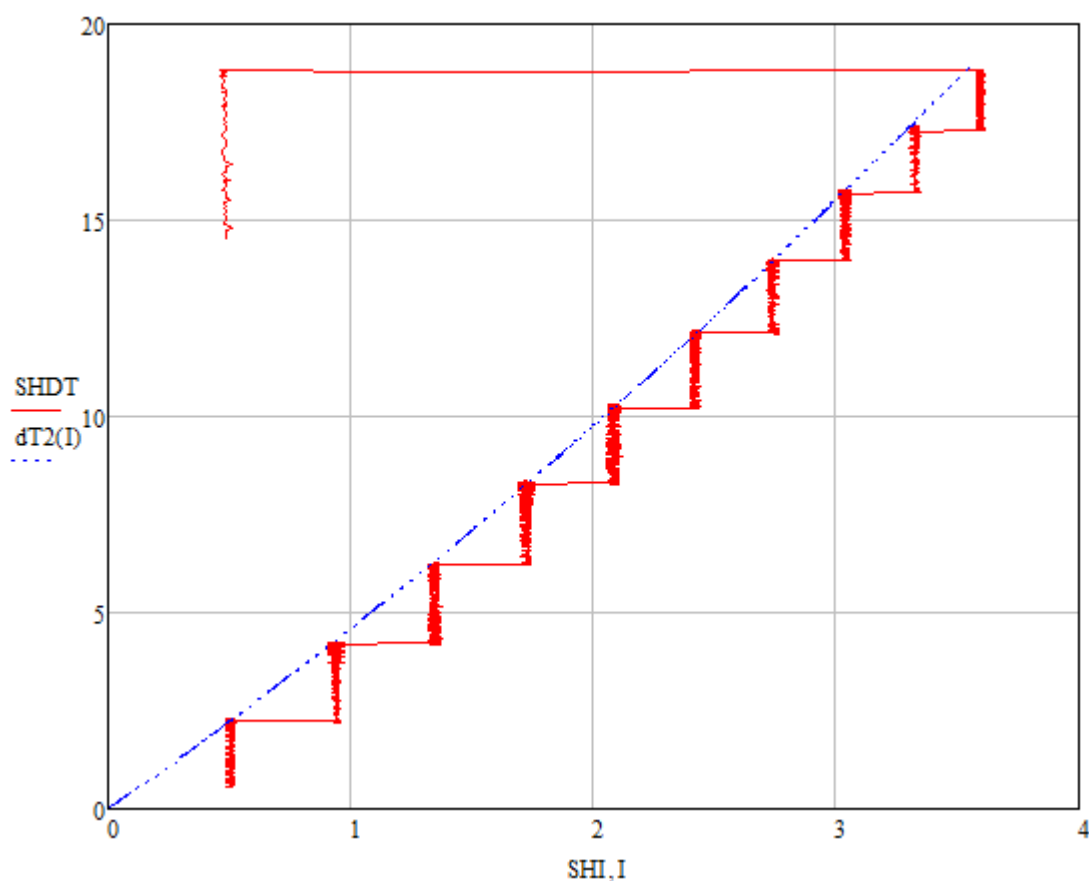


Рисунок 37 – Наложение аппроксимирующей функции на кривую из второго опыта

Отобразим две аппроксимирующие кривые на одной графике (рисунок 38).

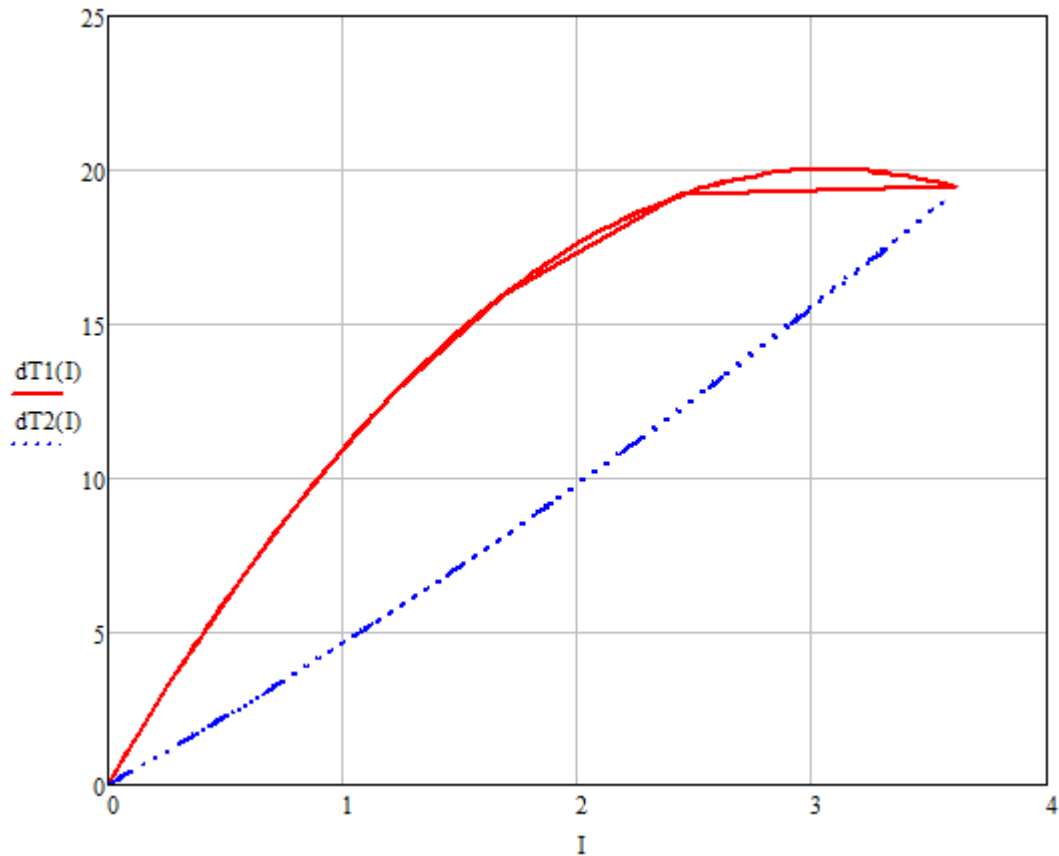


Рисунок 38 – Аппроксимирующие кривые двух опытов

Таким образом, поместив обе аппроксимирующие кривые на один график, можно наглядно увидеть, что плавная подача напряжения позволяет повысить перепад температур, используя меньший ток, что, несомненно, является наиболее эффективным методом.

Кроме того, было определено, что метод подачи тока, не изменяет результат эксперимента, графики полностью совпадают.

4 Управление термокамерой

4.1 Проблема исследования

При эксплуатации не все мобильные термокамеры способны поддерживать постоянную температуру внутри себя, что несомненно является минусом для тех, кому необходимо держать определенное количество градусов, например, для выращивания бифидобактерий, где необходимо установить температуру 37-41С°. Кроме того, не все термокамеры позволяют увидеть график изменения температуры, который необходим определенному количеству пользователей. Для ввода необходимых параметров и их контроля необходимы периферийные устройства. К таким устройствам можно отнести дисплей, клавиатуру, кнопки, или энкодеры. Использование этих элементов затратно с экономической точки зрения и снижает технологичность изготовления устройства. Именно поэтому встал вопрос реализации управления более интересным способом - использование Android-устройства (телефона, планшета и т.д.)

4.2 Выбор способа беспроводной связи

Существует несколько видов беспроводной связи [6]:

- Bluetooth связь. Bluetooth - это стандарт беспроводной связи, обеспечивающий обмен данными между устройствами на основе ультракоротких радиоволн. От других стандартов беспроводной связи он отличается используемыми частотами, правилами передачи и способом шифрования данных. Bluetooth работает на частотах, близких к 2,4 ГГц.

- Wi-Fi модуль. Модули Wi-Fi включают в себя интегрированные аппаратные компоненты, предназначенные для обеспечения функций связи Wi-Fi. Эти модули обычно состоят из коммуникационных чипов Wi-Fi, входных радиочастотных интерфейсов, антенн и вспомогательных цепей. Чипы связи Wi-Fi служат основным компонентом модуля, управляя различными протоколами связи Wi-Fi и передачей данных. ВЧ-интерфейсы усиливают и фильтруют РЧ-сигналы для обеспечения стабильной и

надежной связи Wi-Fi

- GSM модули. GSM-модуль — это устройство, работающее в качестве мобильного телефона или модема, которое используется для связи любого устройства по сети. GSM-модулю для работы требуется SIM-карта, и он работает в диапазоне сети, которую поддерживает он сам и мобильный оператор.

Проанализировав характеристики и ценовой диапазон видов модулей беспроводной связи, было принято решение использовать Bluetooth модуль для обеспечения связи между блоком управления и Android-устройством. Блок управления мобильной термокамерой реализован на отладочной плате ArduinoNano на базе микроконтроллера AtMega328P. Arduino можно программировать на специальном языке Arduino IDE, который основан на C/C++. Bluetooth связь является более удобным способом, по сравнению с другими типами связи, такими как, Wi-Fi или GSM модуля, с наличием или без выхода интернет. Указанные типы связи в основном используется для управления удаленными объектами, такими как “умный дом” и требуют дополнительных интерфейсных устройств и платных сторонних услуг (роутер, сим-карта, выход в интернет).

Преимущества использования Bluetooth для управления термокамерой:

- Простота настройки и использования, на уровне встроенного в модуль связи и Android протокола. Подключение устройства по Bluetooth обычно не требует сложной настройки и паролей.

- Универсальная совместимость. Большинство современных устройств поддерживают стандарт Bluetooth, что позволит легко подключать устройство к термокамере.

- Надежность. Bluetooth обеспечивает стабильное соединение и отсутствие помех.

- Низкая стоимость. Bluetooth модуль обойдется дешевле, чем Wi-Fi модуль или GSM модуль.

4.3 Выбор программы для реализации приложения

Для согласования модуля с Android-устройством требуется разработка полноценного приложения, что требует большого количества времени и определенных навыков программирования. Более простым и удобным вариантом решения проблемы представляется использование специализированных приложений с графическим интерфейсом, который может включать стандартные органы управления и считывания информации (кнопки, переключатели, линейки, графики и т.д.) с настраиваемыми сценариями выполнения и графическим интерфейсом.

После анализа нескольких вариантов для разработки был выбран продукт, размещенный на сайте «RemoteXY.com». Он позволяет создать графический интерфейс для управления нашим устройством [3].

Сам сайт реализовывает дистанционное управление Arduino, ESP32, STM32, nRF. Поддерживает такие операционные системы как Android и IOS. Позволяет реализовать связь с помощью Bluetooth, Wi-Fi, Ethernet, USB, Internet.

В самом редакторе «RemoteXY» создается графический интерфейс, который позволяет разместить на экране смартфона кнопки для управления, сенсоры, индикаторы, оформление. Кроме того, настроить свойства интерфейса: конфигурация, настройки, вид, форматы экрана, страницы. В редакторе можно выбрать непосредственно плату, тип связи, среду разработки, а также разместить несколько страниц приложения. Конкретно для нашего случая было выбрано: тип связи – Bluetooth, плата – ArduinoNano, модуль Bluetooth–HC05, среда разработки – ArduinoIDE. В настройках последовательного порта: интерфейс подключения – SoftwareSerial, контакт RX – 4, контакт TX– 5, скорость обмена – 9600. (рисунок 39)

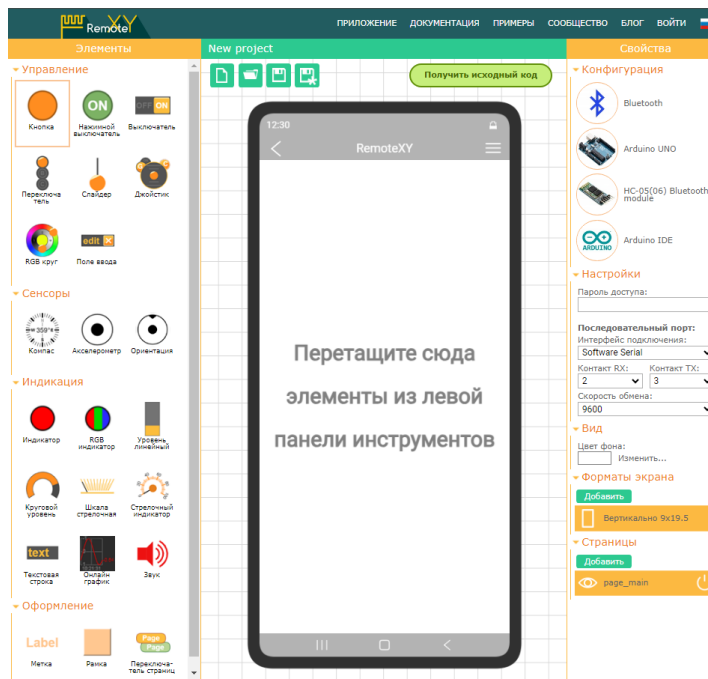


Рисунок 39 – Редактор «RemoteXY»

Кроме того, сайт «RemoteXY» имеет отдельную вкладку с примерами, где представлены проекты пользователей с подробным описанием действий, схем подключения и кодами для работы их приложения.

С учетом технического задания был создан интерфейс, изображенный на рисунке 40.

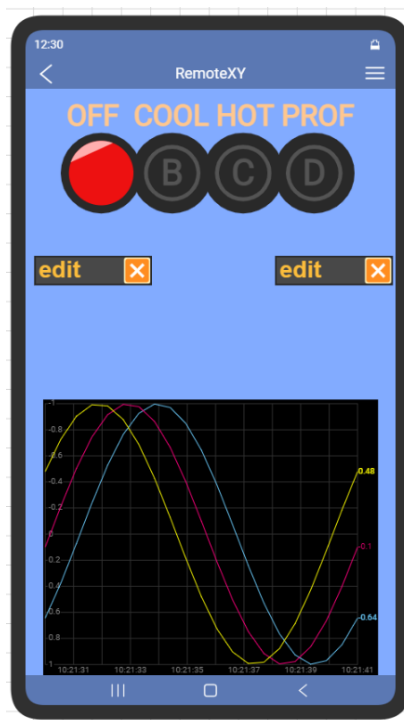


Рисунок 40 – Графический интерфейс для мобильного приложения

С помощью кнопок можно выбирать необходимый режим:

OFF – термокамера выключена;

COOL – термокамера работает в режиме охлаждения;

HOT – термокамера работает в режиме нагрева;

PROF – регулирование температуры по предустановленному термопрофилю (нагрев до заданной температуры в течение заданного времени с последующим охлаждением).

Левое поле ввода (edit) устанавливает температуру режима COOL, правое в режиме HOT. Дополнительно имеется окно с построением временной зависимости температур в процессе работы камеры, на котором будет размещено 3 графика – график окружающей среды, график внутри термокамеры и график разницы температур между окружающей и внутренней средой.

После создания необходимого интерфейса, необходимо нажать кнопку «Получить исходный код» и сайт «RemoteXY» выдаст код графического интерфейса для выбранной среды разработки, в нашем случае ArduinoIDE.

Данная функция, несомненно, упрощает реализацию графического интерфейса. Но для того, чтобы интерфейс работал, необходимо установить библиотеку «RemoteXY» в среду разработки, это можно сделать, непосредственно скачав библиотеку на самом сайте «RemoteXY» во вкладке «Библиотеки» или в самой среде разработки во вкладке «Управление библиотеками» написав в поиске «RemoteXY».

4.4 Программный код для мобильного приложения

Как было сказано выше, среда разработки «RemoteXY» предоставляет пользователю код на языке C++ только для описания графического интерфейса, выданная программа представлена в листинге 8.

Листинг 8

```
/*  
-- \u0422\u0411 by grmnw --
```

```
This source code of graphical user interface  
has been generated automatically by RemoteXY editor.
```

To compile this code using RemoteXY library 3.1.10 or later version

download by link <http://remotexy.com/en/library/>

To connect using RemoteXY mobile app by link <http://remotexy.com/en/download/>

- for ANDROID 4.13.11 or later version;

- for iOS 1.10.3 or later version;

This source code is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

*/

```
////////////////////////////////////
```

```
// RemoteXYincludelibrary //
```

```
////////////////////////////////////
```

```
// можете включить вывод отладочной информации в Serial на 115200
```

```
//#define REMOTEXY__DEBUGLOG
```

```
// определение режима соединения и подключение библиотеки RemoteXY
```

```
#define REMOTEXY_MODE__SOFTSERIAL
```

```
#include <SoftwareSerial.h>
```

```
// настройки соединения
```

```
#define REMOTEXY_SERIAL_RX 4
```

```
#define REMOTEXY_SERIAL_TX 5
```

```
#define REMOTEXY_SERIAL_SPEED 9600
```

```
#include <RemoteXY.h>
```

```
// конфигурация интерфейса RemoteXY
```

```
#pragma pack(push, 1)
```

```
uint8_t RemoteXY_CONF[] = // 82 bytes
```

```
{ 255,23,0,12,0,75,0,17,0,0,0,193,1,126,200,1,1,5,0,7,
```

```
3,57,40,10,36,2,26,2,11,68,6,106,114,94,19,24,179,246,94,7,
```

```
85,57,40,10,36,2,26,2,11,3,12,15,100,28,132,1,8,129,14,5,
```

```
99,10,17,79,70,70,32,32,67,79,79,76,32,72,79,84,32,80,82,79,
```

```
70,0 };
```

```
// структура определяет все переменные и события вашего интерфейса управления
```

```
struct {
```

```
    // input variables
```

```
    char temp_cool[11]; // string UTF8 end zero
```

```
    char Temp_hot[11]; // string UTF8 end zero
```

```
    uint8_t select_01; // =0 если переключатель в положении А, =1 если в положении В,  
=2 если в положении С, ...
```

```
    // output variables
```

```

float onlineGraph_1_var1;
float onlineGraph_1_var2;
float onlineGraph_1_var3;

// other variable
uint8_t connect_flag; // =1 if wire connected, else =0

} RemoteXY;
#pragma pack(pop)

////////////////////////////////////
//      END RemoteXY include      //
////////////////////////////////////

void setup()
{
RemoteXY_Init ();

// TODO you setup code

}

void loop()
{
RemoteXY_Handler ();

// TODO you loop code
// используйте структуру RemoteXY для передачи данных
// не используйте функцию delay(), вместо нее используйте RemoteXY_delay()
}

```

Для того, чтобы программа функционировала, необходимо, самостоятельно дописать программный код, в котором кнопка «OFF» должна выключать термокамеру, а также при открытии мобильного приложения термокамера также была выключена. При нажатии кнопки «COOL» термокамера переходила в режим охлаждения, а на кнопке «HOT» в режим нагрева, а при нажатии на кнопку «PROF» в режим регулирования по предусмотренному термопрофилю. В полях ввода необходимо реализовать считывание введенного числа, чтобы программа распознавала введенное число и давала установку на нагрев или охлаждения до заданного значения. Кроме того, необходимо выполнить привязку и считывания данных с

датчиков температур в графики.

Для написания кода были получены навыки в описании ПИД – регулятора [7], [21], а также в использовании функции millis() [5], [20]. Были подключены все необходимые библиотеки, а также по примерам, которые дает сайт RemoteXY, было изучено как реализовать работу графического интерфейса [9]. Итоговый код представлен в приложении В. Блок-схема алгоритма работы программы представлена на рисунке 41.

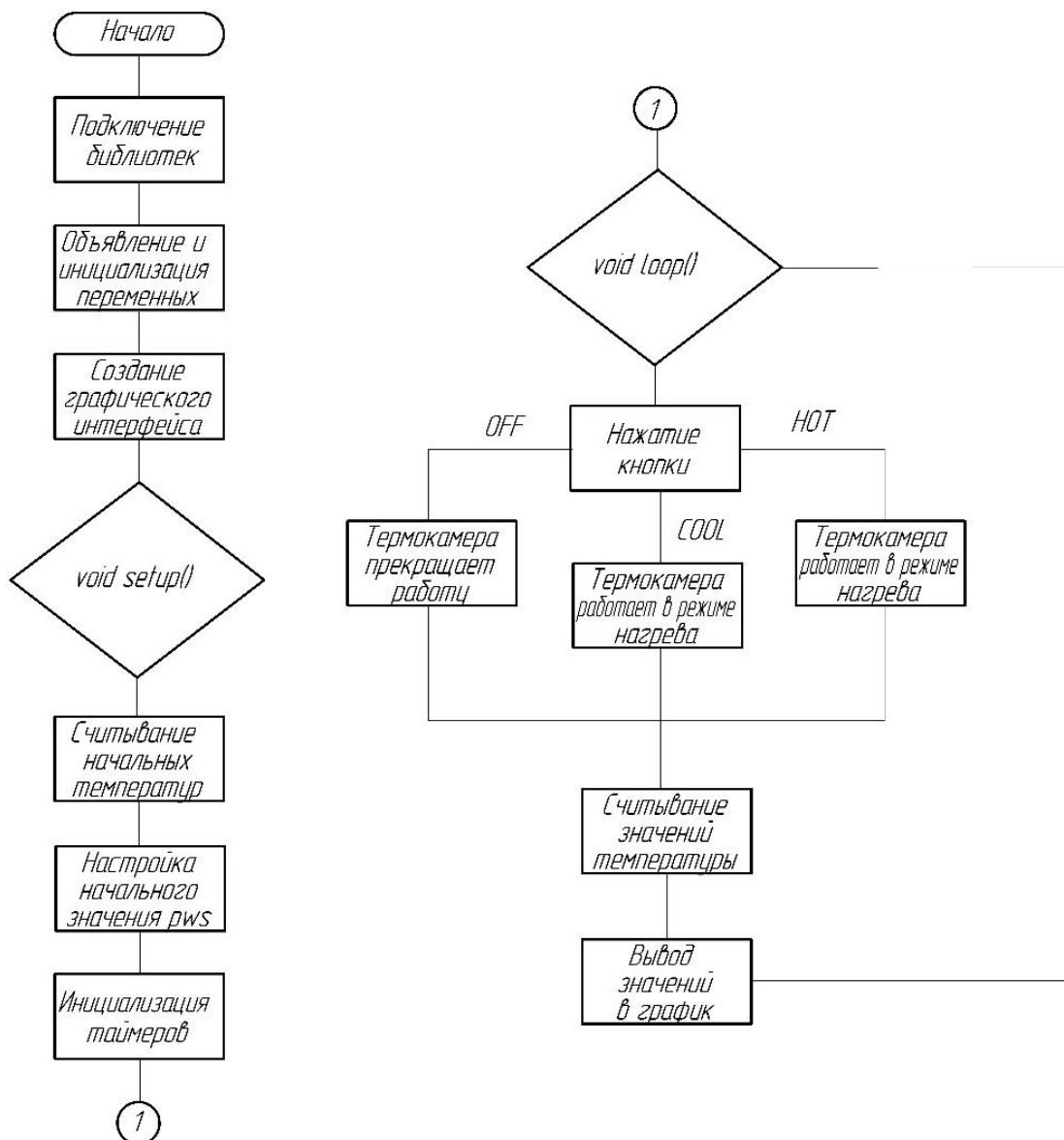


Рисунок 41 – Блок-схема алгоритма работы программы

4.5 Тестирование и отладка

После выполнения всех действий и написания кода, необходимо приступить к тестированию и отладке мобильного приложения. Для начала необходимо скачать с магазина само приложение «RemoteXY», представленное на рисунке 42.

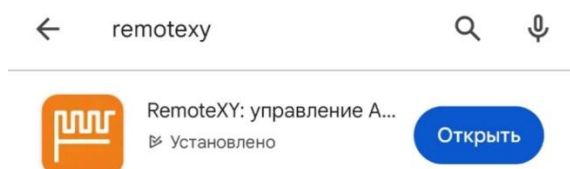


Рисунок 42 – Приложение «RemoteXY»

Следующим шагом необходимо загрузить программу в ArduinoNano. Далее открыть приложение и выполнить привязку нашего BluetoothмодуляHC-06со смартфоном. После соединения приложение выведет на экран кнопку нашего устройства, представленная на рисунке 43.

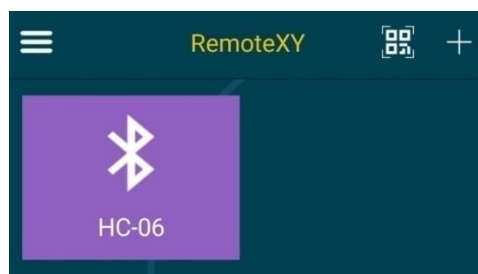


Рисунок 43– Кнопка приложения для работы со смартфоном

Затем откроется наш графический интерфейс и можно приступить к работе. Нажав кнопку «COOL» и введя значение 18 в левое поле, включили режим охлаждения до 18 C°, после охлаждения и удержания необходимой температуры, переходим в режим нагрева, нажав кнопку «HOT» и введя значение 40, включили режим нагрева полости термокамеры до 40 C°. Также получаем данные онлайн в виде графиком, где красный график – график температуры внутри термокамеры, голубой – график температуры окружающей среды, желтый – график разницы температур между окружающей и внутренней средой. Примеры работы приложения в режиме

охлаждения и нагрева представлены на рисунке 44.

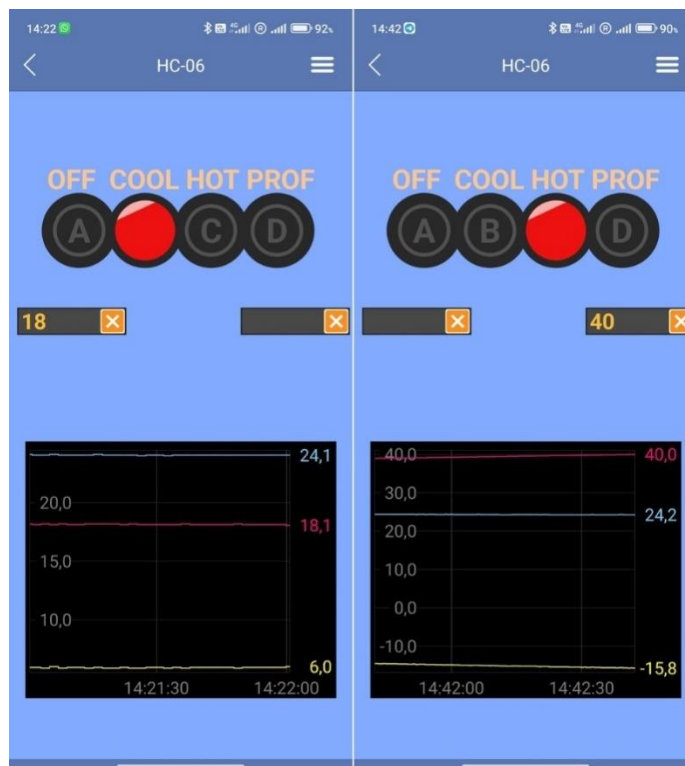


Рисунок 44 – Пример работы мобильного приложения в двух режимах

Таким образом, было проведено многочасовое апробирование работы приложения, и установлены особенности работы:

1. Имеется задержка в несколько тактов от момента смены режима работы камеры с помощью кнопок до изменения режима работы камеры, что вероятно обуславливается невысокой скоростью обмена данных с приложением (9600 бод)

2. Перезагрузка приложения не обнуляет данные на графике. После перезагрузки построение графика продолжается в соответствии со временем, то есть данные отражаются с привязкой к абсолютному времени.

Таким образом, разработанное Android – приложение удовлетворяет техническому заданию и создано за минимальное время.

Заключение

По результатам выпускной квалификационной работы была разработана мобильная термокамера с регулированием температуры, с помощью приложения на смартфон. Разработанное устройство полностью соответствует заявленным характеристикам и выполняет заданные функции.

По результатам графических расчетов, было доказано, что модель элемента Пельтье TEC1-12706 является наиболее экономически выгодным и дает наибольшую разницу температур за наименьшее количество потраченных денежных средств.

Была разработана печатная плата устройства, которая легла в основу создания модели термокамер

По результатам экспериментов было выявлено, какой метод подачи напряжения является наиболее эффективным и позволяет достичь наибольшего перепада температур. Она составила $20,09^{\circ}\text{C}$ при токе равном $3,22\text{A}$.

Кроме того, по результатам опытов, было выявлено, что метод подачи тока не влияет на разницу температур. Следовательно, этот критерий не влияет на работу разработанной термокамеры.

Было разработано мобильное приложение, которое позволяет управлять термокамерой с личного смартфона с помощью Bluetooth. Кроме того, в приложении можно задавать нужную температуру внутри термокамеры, а также в режиме онлайн наблюдать за изменением температуры с помощью графиков.

При достигнутых результатах разработка мобильного приложения для устройства будет продолжаться. Планируется добавление пользовательского меню с выбором уже заранее заготовленных настроек приготовления продуктов.

На текущий момент планируемые цели и задачи выпускной квалификационной работы успешно достигнуты.

Список используемых источников

1. Апарцев Олег Роленович Термоэлектрические кулеры и тепловые процессы в терминах SPICE-моделирования // Электроника и электротехника 2017. - 2017. - №2. - С. 1-11. URL: <https://cyberleninka.ru/article/n/termoelektricheskie-kulery-i-teplovye-protssessy-v-terminah-spice-modelirovaniya>. (Дата обращения 15.02.2024)
2. Белозерцев В.Н. Исследование основных характеристик термоэлектрического охладителя и генератора: лаб. практикум / [В.Н. Белозерцев и др.]. – Самара: Изд-во СГАУ, 2015. – 76 с.: ил. ISBN 978-5-7883-1063-3.
3. Дистанционное управление Arduino [Электронный ресурс]. URL: <https://remotexu.com/ru>. (Дата обращения: 27.04.2024).
4. Импульсный понижающий преобразователь напряжения [Электронный ресурс]. URL: <https://clck.ru/3B9mVu>. (Дата обращения: 07.02.2024)
5. Многозадачность в Arduino [Электронный ресурс]. URL: <https://alexgyver.ru/lessons/how-to-sketch/>. (Дата обращения: 20.04.2024)
6. Обзор современных технологий беспроводной передачи данных в частотных диапазонах ISM (Bluetooth, ZigBee, Wi-Fi) и 434/868 МГц [Электронный ресурс]. URL: <https://wireless-e.ru/standarty/short-range-rf/>. (Дата обращения: 27.04.2024)
7. ПИД регулятор [Электронный ресурс]. URL: <https://alexgyver.ru/lessons/pid/>. (Дата обращения 15.02.2024)
8. ПИД-регуляторы в частотных преобразователях Веспер — высокая точность процесса автоматического регулирования [Электронный ресурс]. URL: <https://www.vesper.ru/presscenter/articles/pid-regulatory-v-chastotnykh-preobrazovatelaykh-vesper-vysokaya-tochnost-protssessa-avtomaticheskogo/>. (Дата обращения 15.02.2024).
9. Примеры проектов Arduino [Электронный ресурс]. URL:

<https://remotexy.com/ru/examples/>. (Дата обращения 21.04.2024)

10. Торопыгина, И. В. Исследование элемента Пельтье и его практическое применение / И. В. Торопыгина, Т. В. Янковская // Сборник материалов IX Всероссийской научно-практической конференции молодых ученых с международным участием "Россия молодая" : Конференция проходит при поддержке Российского фонда фундаментальных исследований, Кемерово, 18–21 апреля 2017 года / Ответственный редактор Костюк Светлана Георгиевна. – Кемерово: Кузбасский государственный технический университет имени Т.Ф. Горбачева, 2017. – С. 13009. – EDN ZQVTUD.

11. Что такое термопаста и как ее наносить? [Электронный ресурс]. URL: <https://clck.ru/3B9m3P>. (Дата обращения: 01.03.2024)

12. ШИМ сигнал [Электронный ресурс] URL: <https://alexgyver.ru/lessons/pwm-signal/> (Дата обращения 20.02.2024)

13. Эффект Пельтье [Электронный ресурс]. URL: <https://clck.ru/3A8qoN>. (Дата обращения 6.02.2024)

14. Элемент Пельтье [Электронный ресурс]. URL: <https://clck.ru/39EERU>. (Дата обращения: 06.02.2024)

15. Элемент Пельтье TEC1-12706 12В 6А 40 х 40 мм [Электронный ресурс]. URL: <https://www.chipdip.ru/product0/8008032923>. (Дата обращения: 25.02.2024).

16.17 лучших термопаст [Электронный ресурс]. URL: <https://mysku.club/blog/ratings/72432.html#t11>. (Дата обращения 01.03.2024)

17. Arduino S. A. Arduino // Arduino LLC. – 2015. – Т. 372.

18. Design of Peltier Element Based on Semiconductors with Hopping Electron Transfer via Defects / N. A. Poklonski, S. A. Vyrko, A. I. Kovalev [et al.] // Devices and Methods of Measurements. – 2021. – Vol. 12, No. 1. – P. 13-22. – DOI 10.21122/2220-9506-2021-12-1-13-22. – EDN VXMVWF.

19. EasyEDAStdTutorial [Электронный ресурс]. URL: <https://docs.easyeda.com/en/FAQ/Editor/index.html>. (Дата обращения 02.03.2024)

20. millis() – ArduinoReference [Электронный ресурс]. URL: <https://arduino.cc/reference/en/language/functions/time/millis/>. (Дата обращения 10.04.2024)

21. PID – ArduinoReference [Электронный ресурс]. URL: <https://www.arduino.cc/reference/en/libraries/pid/>. (Дата обращения 10.04.2024)

22. TEC1-12706 [Электронный ресурс]. URL: <https://clck.ru/3B3FQ5>. (Дата обращения 25.02.2024)

Приложение А

Программный код для первой части первого опыта

```
#include <TimerOne.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SoftwareSerial.h>
#define ciklvalue 10
OneWireoneWire(A2);
DallasTemperature ds(&oneWire);
byte num;
int VS_PIN = 10;
long T = 0;

int tz = 25; float kP = 200, kI = 0, kD = 0;
int constrI = 1000;
float t1 = 0;
float t2 = 0;
float I = 0;
float delta_t=0;
float Q,t;
float U;
float PWMparam;
int pws=50;
int dT = 1;
int pid, PID = 0;
float err = 0, errLast = 0
float dif = 0;
float integral = 0;
float myTimer1, myTimer2, myTimer3;
int mode=2;
const int pwmPin = 9;

void setup() {
  Serial.begin(9600);
  TCCR1A = 0b00000011;
  TCCR1B = 0b00001001;
  ds.begin();
  num = ds.getDeviceCount();
  Serial.print("Sensor temper num="); Serial.println(num);
  pinMode(3, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(pwmPin, OUTPUT);
  ds.requestTemperatures();
  t1=ds.getTempCByIndex(0);
  t2=ds.getTempCByIndex(1);
  digitalWrite(10,LOW);

  for (int pwmValue = 0; pwmValue<= 1023; pwmValue += 102) {
    analogWrite(pwmPin, pwmValue);
    delay(3000);
```



```

printIntermediateValues();
}
myTimer1 = millis();
myTimer2 = millis();
myTimer3 = millis();
}

void loop() {
ds.requestTemperatures();
I=analogRead(A4);
if (millis() - myTimer3 >= 7200000)
{
myTimer3 = millis();
if(pws<1024) pws=pws+100; else pws=1023;
analogWrite(9, pws);
Serial.print("pws=");Serial.println(pws);
printIntermediateValues();
}
I=(((5*I/1024)-(2.47-2.28)-2.3)/0.188)-0.05;
if (millis() - myTimer1 >= 200)
{
myTimer1 = millis();

delta_t=t1-t2;
t1 = t1 * 0.2 + ds.getTempCByIndex(0)*0.8;
t2 = t2 * 0.2 + ds.getTempCByIndex(1)*0.8;
}
if (millis() - myTimer2 >= 3000)
{
myTimer2 = millis();
printIntermediateValues();

Serial.print("$");
Serial.print(t1);Serial.print(" ");
Serial.print(t2);Serial.print(" ");
Serial.print(delta_t);Serial.print(" ");
Serial.print(I);Serial.print(" ");
Serial.print(","); //конечный символ
}
}

void printIntermediateValues() {
Serial.print("$");
Serial.print(t1);Serial.print(" ");
Serial.print(t2);Serial.print(" ");
Serial.print(delta_t);Serial.print(" ");
Serial.print(I);Serial.print(" ");
Serial.print(",");
}
}

```

Приложение Б

Программный код второй части первого опыта

```
#include <TimerOne.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SoftwareSerial.h>
#define ciklvalue 10
OneWireoneWire(A2);
DallasTemperature ds(&oneWire);
byte num;
int VS_PIN = 10;
long T = 0;

int tz = 25; float kP = 200, kI = 0, kD = 0;
int constrI = 1000;
float t1 = 0;
float t2 = 0;
float I, Isr;
float delta_t=0;
float Q,t;
float U;
float PWMparam;
int pws=0;
int dT = 1;
int pwmValue;
int pid, PID = 0;
float err = 0, errLast = 0;
float dif = 0;
float integral = 0;
float myTimer1, myTimer2, myTimer3;
int mode=2;
byte stop=0;
const int pwmPin = 9;

void setup() {
  Serial.begin(9600);
  TCCR1A = 0b00000011;
  TCCR1B = 0b00001001;
  ds.begin();
  num = ds.getDeviceCount();
  Serial.print("Sensor temper num="); Serial.println(num);
  pinMode(3, OUTPUT);
  pinMode(10, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(A4, INPUT);
  pinMode(pwmPin, OUTPUT);
  ds.requestTemperatures();
  t1=ds.getTempCByIndex(0);
  t2=ds.getTempCByIndex(1);
  digitalWrite(10,LOW);
```

```

myTimer1 = millis();
myTimer2 = millis();
myTimer3 = millis();

}

void loop() {
ds.requestTemperatures();

I=analogRead(A4);
if (millis() - myTimer1 >= 50000)
{
myTimer1 = millis();
if(pws<1024 && stop==0) pws=pws+1; else {pws=0; stop=1;}
analogWrite(9, pws);
}

if (millis() - myTimer2 >= 500)
{
myTimer2 = millis();
I=((5*I/1024)-(2.47-2.28)-2.3)/0.188)-0.05;
delta_t=t1-t2;
t1 = t1 * 0.8 + ds.getTempCByIndex(0)*0.2;
t2 = t2 * 0.8 + ds.getTempCByIndex(1)*0.2;
Isr = Isr * 0.9 + I*0.1;
}
if (millis() - myTimer3 >= 2000)
{
myTimer3 = millis();

Serial.print("$"); //начальныйСИМВОЛ
Serial.print(t1);Serial.print(" ");
Serial.print(t2);Serial.print(" ");
Serial.print(delta_t);Serial.print(" ");
Serial.print(Isr);Serial.print(" ");
Serial.print(pws);Serial.print(" ");
Serial.print(";"); //конечныйСИМВОЛ
}
}

```

Приложение В

Программный код для работы мобильного приложения на смартфон

```
#include <TimerOne.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <SoftwareSerial.h>
#define REMOTEXY_MODE__SOFTSERIAL
#include <SoftwareSerial.h>
#include <RemoteXY.h>
#define REMOTEXY_SERIAL_RX 4
#define REMOTEXY_SERIAL_TX 5
#define REMOTEXY_SERIAL_SPEED 9600
OneWire oneWire(A2); // порт подключения датчиков (A2)
DallasTemperature ds(&oneWire);
float t1 = 0;
float t2 = 0;
float delta_t=0;
byte T=2; // период обновления ШИМ в сек
byte num;
int pid, PID = 0; // расчетный и ограниченный 0..100 % уровень заполнения нагревом
float err = 0, errLast = 0; // разница между заданным и реальным значением
float dif = 0; // дифференциальная составляющая разницы
float integral = 0; // интегральная составляющая
int constrI = 1023; // ограничение интеграла сверху
float kP = 500, kI = 0.2, kD = 0;
unsigned long myTimer1=0; // таймер ШИМ
int Tz_cool, Tz_hot;
#pragma pack(push, 1)

uint8_t RemoteXY_CONF[] =
{ 255,23,0,12,0,75,0,17,0,0,0,193,1,126,200,1,1,5,0,7,
  3,57,40,10,36,2,26,2,11,68,6,106,114,94,19,24,179,246,94,7,
  85,57,40,10,36,2,26,2,11,3,12,15,100,28,132,1,8,129,14,5,
  99,10,17,79,70,70,32,32,67,79,79,76,32,72,79,84,32,80,82,79,
  70,0 };

struct {

  char temp_cool[11];
  char temp_hot[11];
  uint8_t select_01; // =0 если переключатель в положении А, =1 если в положении В,
=2 если в положении С, ...

  float onlineGraph_1_var1;
  float onlineGraph_1_var2;
  float onlineGraph_1_var3;
```

```

    uint8_t connect_flag;
} RemoteXY;
#pragma pack(pop)

void setup()
{
    RemoteXY_Init ();
    Serial.begin(9600);
    // Пины D9 и D10 - 15.6 кГц 10bit
    TCCR1A = 0b00000011; // 10bit
    TCCR1B = 0b00001001; // x1 fast pwm
    ds.begin(); // инициализация датчика ds18b20
    num = ds.getDeviceCount(); // узнаем количество датчиков
    Serial.print("Sensor temper num=");Serial.println(num);
    pinMode(10, OUTPUT); //D10 пин управления реле
    pinMode(9, OUTPUT); //D9 пин управления шим
    analogWrite(9, 0); // начальное вкл ЭП 0 - ШИМ
}

void loop()
{
    if (RemoteXY.select_01==0) {digitalWrite(10,LOW); analogWrite(9,
0);Serial.println("выкл");} //выкл
    if (RemoteXY.select_01==1) {
        while (millis() - myTimer1 <= 2000) {}; //расчет параметров управления
регулятора ПИД каждую секунду
        myTimer1 = millis();
        int Mode=-1;
        digitalWrite(10,LOW);
        int Reg = Regulator( t2, Tz_cool, dT, Mode, kP, kI, kD);
        analogWrite(9, Reg);
        Serial.println("охл.");} //охлажд
        if(RemoteXY.select_01==2) {digitalWrite(10,HIGH); analogWrite(9,
1023);Serial.println("нагрев");} //нагрев

    RemoteXY_Handler ();
    Tz_cool = atoi (RemoteXY.temp_cool);
    Tz_hot = atoi (RemoteXY.temp_hot);
    RemoteXY.onlineGraph_1_var1=t1;
    RemoteXY.onlineGraph_1_var2=t2;
    RemoteXY.onlineGraph_1_var3=delta_t;
    ds.requestTemperatures(); //считывание
    t1=ds.getTempCByIndex(0);
    t2=ds.getTempCByIndex(1);
    delta_t=t1-t2;
    Serial.println(Tz_cool);
    Serial.println(Tz_hot);
}

```

```

int Regulator( float t, float tz, int dT, int Mode, float kP, float kI, float kD) //
температура камеры t изадания tz, время дискрета ШИМ dT, kP kI kD коэффПИД
{
  err = (tz - t)*Mode; //ошибка по температуре
  integral = integral + err*dT;
  integral = constrain (integral, 0, constrI); //ограничим значение интеграла
  floatdeltaErr = (err - errLast)/dT; //изменение ошибки по сравнению с предыдущим
периодом
  errLast = err;
  dif = dif * 0.8 + deltaErr * 0.2; // фильтрованная (с тау = 5 циклов) диф.
составляющая
  pid =kP * err + kI * integral + kD * dif;
  PID = constrain (pid, 0, 1023);
  Serial.print(t);Serial.print(" ");
  Serial.print(tz);Serial.print(" ");
  Serial.print(pid);Serial.print(" ");
  Serial.print(PID);Serial.println(" ");
  return PID;
}

```