Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение

высшего образования

**«Северный (Арктический) федеральный университет имени М.В. Ломоносова**»

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Институт судостроения и морской арктической техники (Севмашвтуз)** | | | | | | | | | | | | | | | | | |  |
|  | (наименование института) | | | | | | | | | | | | | | | | | |  |
|  | Ямутеев Максим Владимирович | | | | | | | | | | | | | | | | | |  |
|  | (ФИО обучающегося) | | | | | | | | | | | | | | | | | |  |
|  |  |
|  | **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)** | | | | | | | | | | | | | | | | | |  |
|  | 27.03.04 Управление и информатика в технических системах | | | | | | | | | | | | | | | | | |  |
|  | (код и наименование направления подготовки / специальности) | | | | | | | | | | | | | | | | | | |
|  | Управление в технических системах | | | | | | | | | | | | | | | | | | |
|  | (наименование направленности образовательной программы (профиля / специальности / название магистерской программы)) | | | | | | | | | | | | | | | | | | |
|  | Цифровая система управления импульсным источником питания повышенной | | | | | | | | | | | | | | | | | | |
|  | (тема ВКР)  надёжности для светодиодных систем освещения | | | | | | | | | | | | | | | | | | |
|  |  | | | | | | | | | | | | | | | | | | |
|  | Утверждена приказом от « 18 » | | | | | | декабря 2019 г. № 10/866 | | | | | | | | | | | | |
|  |  | | | | | | | |  | | | | | | | | | | |
|  | Руководитель ВКР | |  | | | | |  |  | | | | | | | |  | С.М. Сковпень, к.т.н | |
|  | Нормоконтроль | |  | |  | | |  | |  | | | | | | |  | С.М. Сковпень, к.т.н | | | |
|  | Руководитель ОПОП | |  | |  | | |  | |  | | | | | | |  | М.М. Музыка, к.т.н | | | |
|  |  | |  | | (дата) | | |  | | (подпись) | | | | | | |  | (ФИО, должность/ степень/ звание) | | | |
|  | Постановление ГЭК от «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_\_г. | | | | | | | | | | | | | | | | | | | |
|  | Признать, что обучающийся | | | | | М.В. Ямутеев | | | | | | | | | | | | | | |
|  |  | | (инициалы, фамилия) | | | | | | | | | | | | | |  |  | |
|  | выполнил и защитил ВКР с отметкой | | | | | | | | | | |  | | | | | | | | | | |
|  |  | | | | | | | | | | | | | |  | (отметка прописью) | | | |
|  | Председатель ГЭК |  | | |  | | | | | | | |  |  | | | | | | | | | |
|  | (подпись) | | | | | | | | | | (инициалы, фамилия) | | | | | | | | |
|  | Секретарь ГЭК |  | |  | | | | | | | | |  |  | | | | | |
|  | (подпись) (инициалы, фамилия)  Северодвинск 2020 | | | | | | | | | | | | | | | | | | |

|  |  |  |
| --- | --- | --- |
| Министерство науки и высшего образования Российской Федерации | | |
| федеральное государственное автономное образовательное  учреждение высшего образования | | |
|  | **Северный (Арктический) федеральный университет имени М.В. Ломоносова»** | |
|  | |  | | --- | | **Институт судостроения и морской арктической техники (Севмашвтуз)** | | (наименование высшей школы / филиала / института) | |  | |  | | |
|  | **ЗАДАНИЕ ДЛЯ ПОДГОТОВКИ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ** | |
|  | |  | | --- | | 27.03.04 Управление и информатика в технических системах | | код и наименование направления подготовки / специальности | |  |   Тема ВКР: \_\_Цифровая система управления импульсным источником питания повышенной надежности для светодиодных систем освещения\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  | Утверждена  протоколом заседания кафедры от « 6 »  декабря 2019 г. № 4    Обучающемуся:  \_Ямутееву Максиму Владимировичу\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (ФИО)  Курс: \_\_4\_\_\_\_\_\_Группа: \_\_\_521625\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | |
|  | Срок сдачи обучающимся законченной работы «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_ 20\_\_\_\_ г. | |
|  | Исходные данные к работе |  |
|  | Параметры импульсного источника питания:  Входное напряжение *U*вх = 220 В, частота коммутации *f* = 100 кГц, | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | выходное напряжение *U*вых = 18 В, ток нагрузки *I*н = 0,45 А. | | | | | | | |
|  | Основные разделы работы с указанием вопросов, подлежащих рассмотрению\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  1 раздел − Принципы построения источников питания для светодиодных систем освещения  Основные вопросы: обзор силовых цепей импульсных преобразователей, обзор способов стабилизации и регулирования выходного тока\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  2 раздел − Разработка светодиодного светильника повышенной надежности\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Основные вопросы: разработка силовой части импульсного источника питания, разработка системы управления импульсным источником питания, разработка программного обеспечения для микроконтроллера, разработка печатной платы импульсного источника питания\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  3 раздел − Расчет затрат на разработку цифровой системы управления импульсным источником питания\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Основные вопросы: расчет стоимости материалов, расчет основной и дополнительной заработной платы, расчет отчислений на социальные нужды, расчет накладных расходов\_\_\_\_  4 раздел − Техника безопасности при монтаже и наладке источника питания\_  Основные вопросы: правила техники безопасности при монтаже и наладке электронных устройств\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  База проведения исследований \_\_Кафедра судовой электроэнергетики и автоматики\_\_\_\_\_\_\_\_\_  Перечень обязательных приложений к работе\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | | | | | | | |
|  |  | | | | | | | |
|  |  | | | | | | | |
|  |  | | | | | | | |
|  | Перечень графического материала | | | |  | | | |
|  |  | | |  | | | | |
|  |  | | | | | | | |
|  |  | | | | | | | |
|  |  | | | | | | | |
|  |  | | | | | | | |
|  |  | | | | | | | |
|  | Консультанты по работе | |  | | | | | |
|  | по разделу |  | | |  |  |  |  |
|  | по разделу |  | | |  |  |  |  |
|  | по разделу |  | | |  |  |  |  |
|  |  | (дата) | | |  | (подпись) |  | (ФИО, должность) |
|  |  |  | | |  |  |  |  |
|  |  |  | | |  |  |  |  |

Дата выдачи задания «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_\_г.

Руководитель ВКР\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Сковпень С.М.\_\_

(подпись) (инициалы, фамилия)

Задание принял к исполнению «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_\_\_г.

Обучающийся \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_М.В. Ямутеев\_\_\_\_

(подпись) (инициалы, фамилия)

РЕФЕРАТ

Ямутеев М.В. Цифровая система управления импульсным источником питания повышенной надежности для светодиодных систем освещения.

Руководитель ВКР – кандидат технических наук, доцент, Сковпень С.М.

Выпускная квалификационная работа объемом 95 стр., содержит, 64 рисунка, 2 таблицы, 15 источников, 3 приложения, графическую часть на 6 листах.

Ключевые слова: импульсный преобразователь постоянного напряжения, широтно-импульсная модуляция, ПИД-регулятор, стабилизация напряжения, светодиодное освещение.

Цель работы – разработка цифровой системы управления импульсным источником питания повышенной надежности для светодиодных систем освещения.

Структура ВКР: состоит из введения, четырех глав, заключения, списка использованных источников, приложений.

В первой главе рассмотрены принципы построения источников питания для светодиодных систем освещения.

Во второй главе разработана цифровая система управления импульсным источником питания повышенной надежности для светодиодных систем освещения.

В третьей главе представлен расчёт затрат на разработку цифровой системы управления импульсным источником питания повышенной надежности для светодиодных систем освещения.

В четвертой главе описана техника безопасности при монтаже и наладке источника питания.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_М.В. Ямутеев\_\_\_\_

(дата) (подпись) (инициалы, фамилия)

ОГЛАВЛЕНИЕ

[НОРМАТИВНЫЕ ССЫЛКИ 7](#_Toc43762022)

[ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ 8](#_Toc43762023)

[ВВЕДЕНИЕ 9](#_Toc43762024)

[1 ПРИНЦИПЫ ПОСТРОЕНИЯ ИСТОЧНИКОВ ПИТАНИЯ ДЛЯ СВЕТОДИОДНЫХ СИСТЕМ ОСВЕЩЕНИЯ 11](#_Toc43762025)

[1.1 Обзор силовых цепей преобразователей 11](#_Toc43762026)

[1.1.1 Линейный стабилизатор 11](#_Toc43762027)

[1.1.2 Импульсный стабилизатор 12](#_Toc43762028)

[1.1.3 Обратноходовый преобразователь 16](#_Toc43762029)

[1.2 Обзор способов стабилизации и регулирования выходного тока 17](#_Toc43762030)

[1.2.1 Стабилизация тока без обратной связи 17](#_Toc43762031)

[1.2.2 Регулирование тока без обратной связи 18](#_Toc43762032)

[1.2.3 Регулирование тока с обратной связью 19](#_Toc43762033)

[2 РАЗРАБОТКА СВЕТОДИОДНОГО СВЕТИЛЬНИКА ПОВЫШЕННОЙ НАДЕЖНОСТИ 22](#_Toc43762034)

[2.1 Разработка силовой части 22](#_Toc43762035)

[2.1.1 Описание схемы 22](#_Toc43762036)

[2.1.2 Выбор элементов силовой цепи и моделирование источника питания 23](#_Toc43762037)

[2.2 Разработка системы управления 36](#_Toc43762038)

[2.2.1 Описание схемы 36](#_Toc43762039)

[2.2.2 Разработка модели управляемого шим-генератора и моделирование работы иип с обратной связью по току 37](#_Toc43762040)

[2.3 Исследование работы импульсного источника питания с цифровой системой управления при изменении нагрузки 41](#_Toc43762041)

[2.4 Разработка программного обеспечения контроллера 44](#_Toc43762042)

[2.5 Разработка печатной платы импульсного источника питания 53](#_Toc43762043)

[3 РАСЧЕТ ЗАТРАТ НА РАЗРАБОТКУ ЦИФРОВОЙ СИСТЕМЫ УПРАВЛЕНИЯ ИМПУЛЬСНЫМ ИСТОЧНИКОМ ПИТАНИЯ 73](#_Toc43762044)

[4 ТЕХНИКА БЕЗОПАСНОСТИ ПРИ МОНТАЖЕ И НАЛАДКЕ ИСТОЧНИКА ПИТАНИЯ 76](#_Toc43762045)

[ЗАКЛЮЧЕНИЕ 78](#_Toc43762046)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 79](#_Toc43762047)

[ПРИЛОЖЕНИЕ А 81](#_Toc43762048)

[ПРИЛОЖЕНИЕ Б 82](#_Toc43762049)

[ПРИЛОЖЕНИЕ В 83](#_Toc43762050)

# НОРМАТИВНЫЕ ССЫЛКИ

В настоящем текстовом документе использованы ссылки на следующие нормативные документы:

СТО 60-02.2.3-2018 Общие требования к оформлению и изложению документов учебной деятельности обучающихся

ГОСТ Р 52907-2008 «Источники электропитания радиоэлектронной аппаратуры. Термины и определения» Источники электропитания радиоэлектронной аппаратуры. Термины и определения

# ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В настоящем текстовом документе применяются следующие определения, обозначения и сокращения:

Импульсный преобразователь напряжения – стабилизатор напряжения, в котором регулирующий элемент (ключ) работает в импульсном режиме, то есть регулирующий элемент периодически открывается и закрывается.

ШИМ – широтно-импульсная модуляция;

ИИП − импульсный источник питания;

ПО − программное обеспечение;

САПР − система автоматического проектирования;

КПД − коэффициент полезного действия;

МК − микроконтроллер;

АЦП − аналогово-цифровой преобразователь.

# ВВЕДЕНИЕ

Проблема высокого энергопотребления становится все более актуальной в наше время. На одно только освещение уходит порядка 30-35% всей вырабатываемой электроэнергии, а в масштабах крупных городов эта величина в полтора-два раза больше [1].

Светодиодное освещение – это одно из перспективных направлений технологий искусственного освещения, основанное на использовании светодиодов в качестве источника света.

В последние несколько десятков лет технический прогресс в области разработки и изготовления светодиодов идет с невероятной скоростью [2]. Современные светодиоды отличаются миниатюрностью, прочностью, надежностью, хорошими оптическими характеристиками и высоким квантовым выходом излучения. В отличие от многих других источников света, светодиоды могут преобразовывать электрическую энергию в световую с коэффициентом полезного действия близким к единице.

В качестве источников питания светодиодного освещения используются импульсные источники питания, которые генерируют электромагнитные помехи, вследствие чего необходимо обеспечить электромагнитную совместимость, во избежание появления помех в электрической сети. Для обеспечения электромагнитной совместимости применяются различные схемы фильтрации в источниках питания.

При включении в сеть импульсных источников питания появляется реактивная составляющая тока, которую можно исключить с помощью корректора коэффициента мощности.

Целью выпускной квалификационной работы является разработка цифровой системы управления импульсным источником питания повышенной надёжности для светодиодных систем освещения.

Задачи, решаемые в процессе выполнения выпускной квалификационной работы:

* обзор принципов построения источников питания для светодиодных систем освещения;
* разработка цифровой системы управления импульсным источником питания повышенной надёжности;
* изучение техники безопасности при монтаже и наладке источника питания;
* расчёт затрат на разработку источника питания.

## 1 ПРИНЦИПЫ ПОСТРОЕНИЯ ИСТОЧНИКОВ ПИТАНИЯ ДЛЯ СВЕТОДИОДНЫХ СИСТЕМ ОСВЕЩЕНИЯ

## 1.1 Обзор силовых цепей преобразователей

# **1.1.1 Линейный стабилизатор**

Линейный стабилизатор (рисунок 1) является делителем напряжения постоянного тока [3].



Рисунок 1 – Типовая схема линейного стабилизатора тока

Принцип его работы заключается в том, что на вход подаётся нестабильное напряжение (*U*вх), а с коллектора транзистора *VT*1 снимается фиксированное выходное напряжение (*U*вых). Принципиально такого рода стабилизация осуществляется благодаря изменению сопротивления одного из плеч делителя напряжения, в роли которого выступает биполярный транзистор (*VT*1) [13].

При достаточно большом отношении входного и выходного напряжений значительно сокращается КПД стабилизатора напряжения, так как энергия рассеивается в виде тепла на транзисторе стабилизатора. По этим причинам в рассматриваемом устройстве особое внимание необходимо уделить вопросам охлаждения, обеспечивая тем самым безопасную и бесперебойную работу стабилизатора напряжения, увеличивая срок службы и предотвращая регулирующий элемент стабилизатора от перегрева.

Преимущества:

- отсутствие помех в выходном сигнале;

- простое конструктивное исполнение.

Недостатки:

- низкий КПД;

- необходимо охлаждение транзистора, из-за чего увеличиваются габариты устройства;

- питается от силового трансформатора, который также увеличивает габариты источника питания.

# **1.1.2 Импульсный стабилизатор**

В импульсном стабилизаторе напряжения регулирующий элемент работает в ключевом режиме, то есть большую часть времени он находится либо в режиме отсечки, когда его сопротивление максимально, либо в режиме насыщения – с минимальным сопротивлением, а значит, может рассматриваться как ключ. Плавное изменение напряжения происходит благодаря наличию интегрирующего элемента: напряжение повышается по мере накопления им энергии и снижается по мере отдачи ее в нагрузку. Такой режим работы позволяет значительно снизить потери энергии, а также улучшить массогабаритные показатели.

Прямоходовые стабилизаторы образуют большое семейство топологий импульсных источников энергии. На рисунке 2 изображена схема понижающего прямоходового преобразователя.

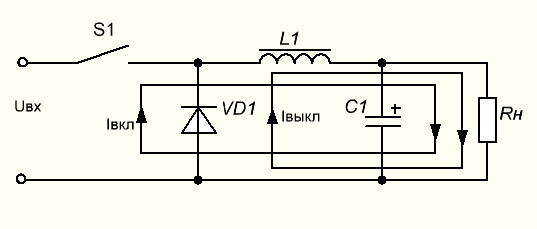


Рисунок 2 – Прямоходовый понижающий преобразователь

Рассмотрим более подробно принцип работы данной схемы. *LC*-фильтр сохраняет энергию между силовыми импульсами драйвера. На вход *LC*-фильтра подается срезанное входное напряжение *U*вх. Фильтр выполняет вольт-временное усреднение колебаний входного напряжения, модулированной по рабочему циклу. Выходное напряжение поддерживается схемой управления путем изменения рабочего цикла. Преобразователь является понижающим, так как выходное напряжение должно быть ниже входного. Диаграммы работы данного преобразователя приведены на рисунке 3.

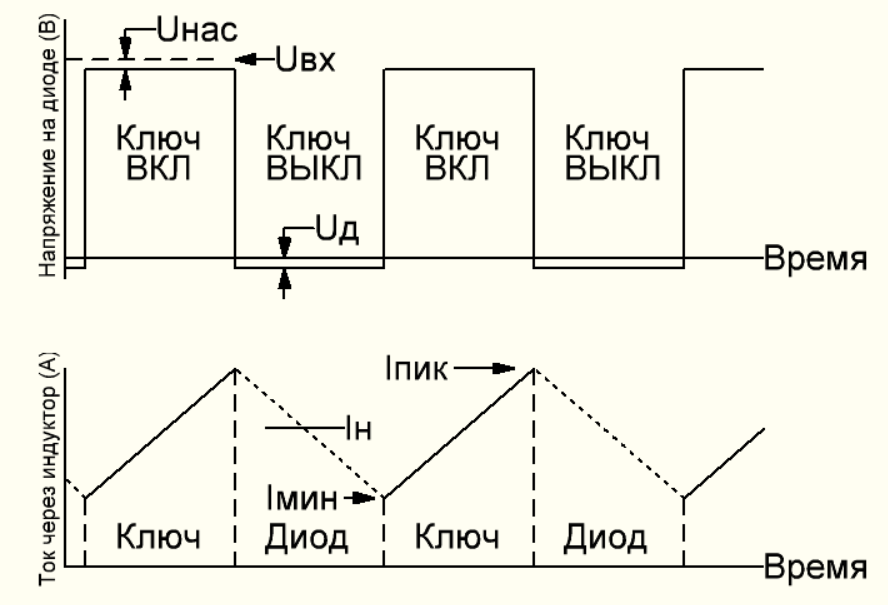
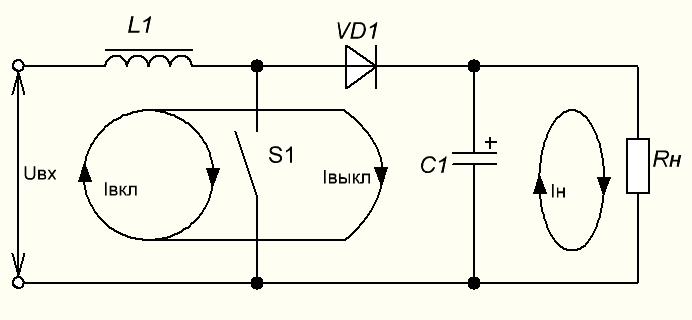


Рисунок 3 – Временные диаграммы напряжения и тока для прямоходового (понижающего) преобразователя

Когда ключ *S*1 замкнут, входное напряжение *U*вх подается на вход *LC*-фильтра. Ток через индуктор линейно повышается. Энергия запасается в индукторе и сохраняется магнитным потоком внутри материала сердечника. Когда ключ *S*1 разомкнут, выходное напряжение стремится упасть ниже уровня нуля, и диод *VD*1, называемый ограничивающим диодом, становится прямосмещённым. Он продолжает проводить ток *I*выкл, который прежде протекал через ключ, и часть сохранённой энергии разряжается на нагрузку. Это формирует локальную токовую петлю, состоящую из диода, индуктора и нагрузки. Когда ключ *S*1 замыкается, диод запирается, и ток протекает через входной источник питания и ключ *S*1.

Другим видом преобразователей является повышающий преобразователь напряжения, схема которого приведена на рисунке 4.

Рисунок 4 – Повышающий преобразователь

Когда ключ *S*1замкнут, создаётся токовая петля, состоящая только из индуктора *L*1, ключа *S*1 и источника входного напряжения *U*вх. В течение этого периода диод *VD*1 смещён в обратном направлении, а кривая тока, протекающего через индуктор *L*1, также имеет положительный линейный наклон. Энергия запасается в магнитном потоке внутри материала сердечника индуктора. Когда ключ *S*1 размыкается, напряжение на индукторе *L*1возвращается к уровню входного напряжения *U*вх. Диод *VD*1 сразу же становится прямо смещённым, когда напряжение индуктора *L*1 превышает выходное напряжение. Затем напряжение индуктора *L*1фиксируется на величине выходного напряжения. Этот уровень напряжения, называемый напряжением обратного хода, равен выходному напряжению плюс падение прямого напряжения на диоде *VD*1. Когда магнитный поток сердечника полностью исчезает до наступления следующего цикла, такой режим работы называется прерывистым. Если же энергия опустошается не полностью, то режим называется непрерывным. На рисунке 5 изображены диаграммы работы повышающего преобразователя.

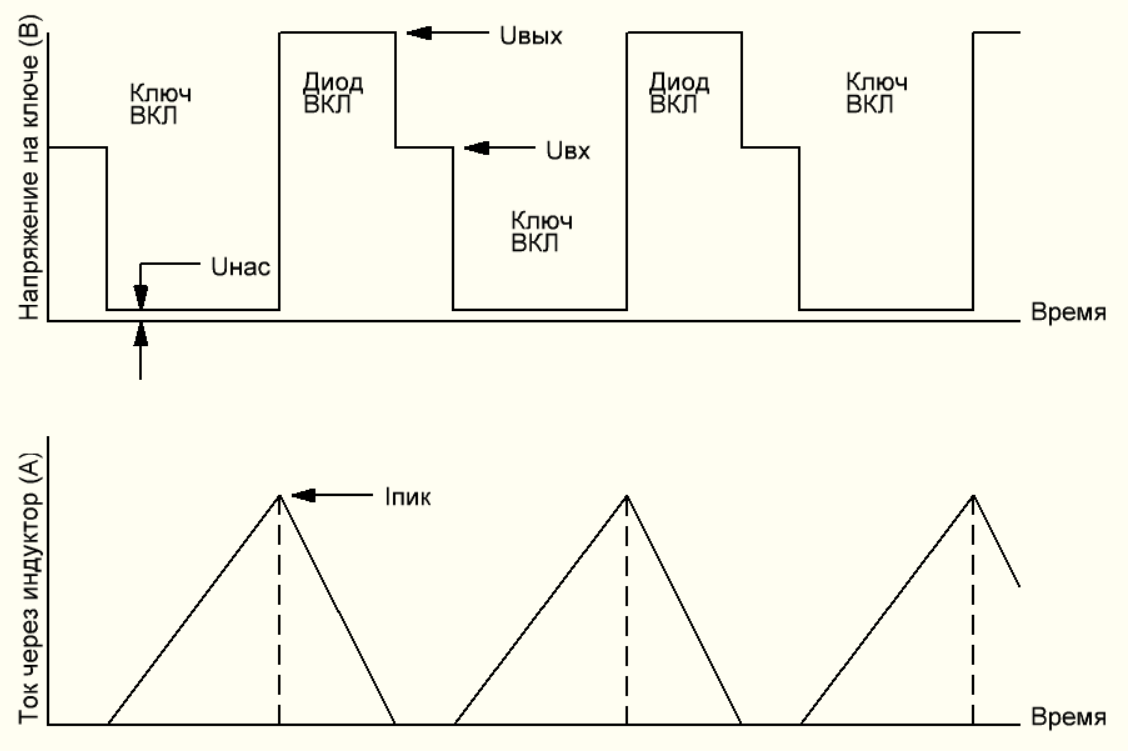


Рисунок 5 – Временные диаграммы напряжения и тока повышающего преобразователя

Преимущества:

* низкие показатели размаха напряжения пульсации;
* более высокие уровни выходной мощности.

Недостатки:

* конструкция более сложная;
* генерирование высокочастотных помех.

# **1.1.3 Обратноходовый преобразователь**

Обратноходовый преобразователь (англ. *Flyback converter*) – это простая схема включения, которая может быть использована для преобразования постоянного напряжения или постоянного тока. Использование двух или более обмоток в индукторе обеспечивает гальваническую развязку [10].

Схема такого преобразователя приведена на рисунке 6.

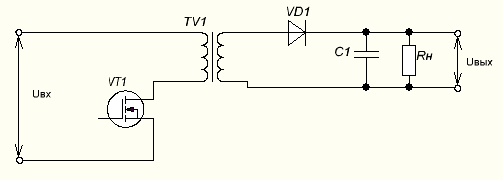


Рисунок 6 – Схема обратноходового преобразователя

Традиционный *flyback* преобразователь использует индуктор как минимум с двумя обмотками (трансформатор) – одна обмотка является основной, соединена с входной цепью источника питания, и подключается к общему проводу, а другая – подключается к нагрузке. Схема устроена так, чтобы магнитная энергия накапливалась в индукторе в течение времени, когда ключ включен, тогда ток увеличивается в первичной обмотке (энергия запасается). Когда ключ разомкнут, магнитная энергия передаётся на нагрузку.

Преимущества:

- возможность регулирования выходного напряжения в широких пределах;

- сохранение требуемого выходного напряжения при изменении напряжения питающей сети.

Недостатки:

- высокий уровень электромагнитных помех, которые создаются как в питающей сети, так и в нагрузке.

# **1.2 Обзор способов стабилизации и регулирования выходного тока**

# **1.2.1 Стабилизация тока без обратной связи**

1. Стабилизаторы тока на транзисторах

На рисунке 7 представлена схема, работа которой основана на так называемом эмиттерном повторителе. Транзистор, включенный таким образом, стремится поддерживать напряжение на эмиттере в точности таким же, как и на базе (разница будет только в падении напряжения на переходе база-эмиттер). Таким образом, зафиксировав напряжение базы с помощью стабилитрона, мы получаем фиксированное напряжение на резисторе *R*1.

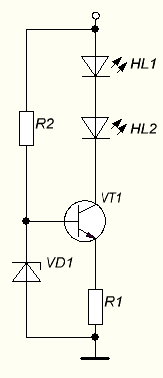


Рисунок 7 – Схема стабилизации тока светодиода с помощью транзистора

Ток, протекающий через светодиоды, задаётся подбором резистора *R*2. Резистор *R*1 подбирается таким образом, чтобы выйти на линейный участок ВАХ диодов (с учётом тока базы транзистора). Напряжение питания всей схемы должно быть не менее суммарного напряжения на всех светодиодах плюс 2-2,5 В для устойчивой работы транзистора.

1. Стабилизаторы тока на интегральных микросхемах

Микросхемы позволяют добиться гораздо более высоких характеристик, чем транзисторы [4]. Чаще всего для сборки стабилизатора тока для светодиодов используют прецизионные термостабильные источники опорного напряжения (*TL*431, *LM*317 и другие). На рисунке 8 представлена типовая схема стабилизатора тока на *TL*431.

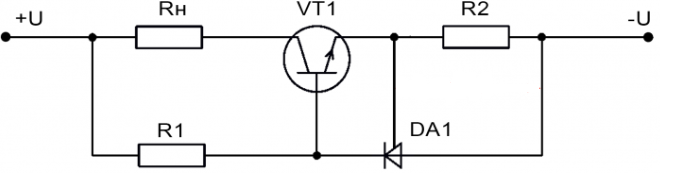


Рисунок 8 – Типовая схема стабилизатора тока на *TL*431

Так как микросхема ведёт себя так, чтобы поддерживать на резисторе *R*2 фиксированное напряжение 2,5 В, то ток через этот резистор всегда будет равен 2,5/*R*2. Если пренебречь током базы, то можно считать, что . При этом, чем выше будет коэффициент усиления транзистора , тем больше эти токи будут совпадать.

Резистор *R*1 подбирается таким образом, чтобы обеспечить минимальный рабочий ток микросхемы – 1 мА.

# **1.2.2 Регулирование тока без обратной связи**

Стабилизатор тока можно также реализовать с помощью ШИМ-регулятора [4]. Схема типового импульсного стабилизатора тока представлена на рисунке 9.

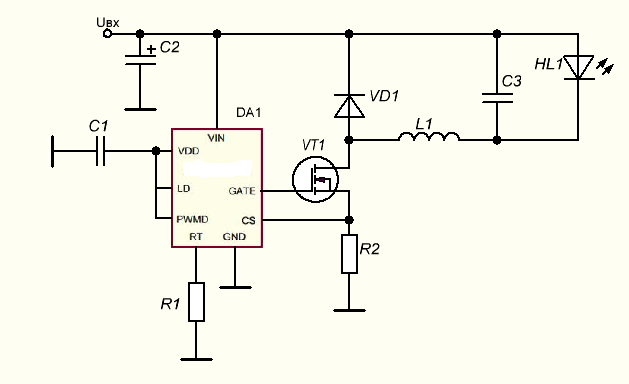


Рисунок 9 – Импульсный стабилизатор тока

Данный стабилизатор работает следующим образом. Микросхема *DA*1 при подаче питания открывает ключ *VT*1, через светодиоды, дроссель *L*1 и резистор *R*2 начинает протекать ток. Когда падение напряжения на *R*2 достигает определённого значения, микросхема закрывает ключ, и ток под действием энергии запасённой в дросселе начинает течь через диод *VD*1. Далее процесс повторяется циклически, управляемый внутренним генератором, частота которого задаётся резистором *R*1. В данной схеме отсутствует обратная связь по току, так как нет датчика тока, который измеряет значение на выходе импульсного источника питания.

# **1.2.3 Регулирование тока с обратной связью**

В качестве схемы регулирования тока с обратной связью можно рассматривать обратноходовый импульсный источник питания, с датчиком тока в выходной цепи. Пример схемы данного стабилизатора представлен на рисунке 10 [10].

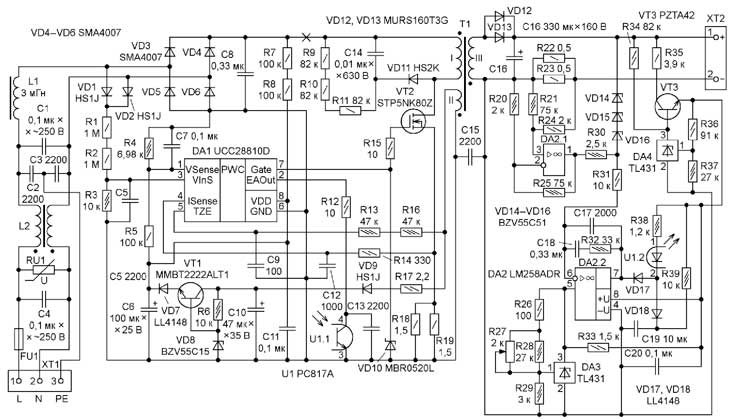


Рисунок 10 − Обратноходовый источник питания с обратной связью по току

На базе данной схемы можно рассмотреть основные узлы и принцип работы устройства. Во вторичной цепи установлен датчик тока – резисторы *R*22, *R*23. Он подключён к входам дифференциального усилителя *DA*2.1. Далее усиленный сигнал подаётся на инвертирующий вход ОУ *DA*2.2. На его неинвертирующий вход поступает образцовое напряжение с регулируемого источника на параллельном стабилизаторе *DA*3. ОУ *DA*2.2выполняет функцию компаратора. Как только напряжение на инвертирующем входе превысит образцовый уровень (на неинвертирующем входе), напряжение на выходе *DA*2.2снизится до нуля и оптопара *U*1 откроется. В результате микросхема *DA*1 уменьшит время открытого состояния транзистора *VT*2 и ток через нагрузку снизится до установленного значения. С помощью переменного резистора *R*27 можно регулировать образцовое напряжение на неинвертирующем входе компаратора *DA*2.2и соответственно ток через нагрузку (светодиоды). Например, при токе нагрузки 350 мА, напряжение на неинвертирующем входе *DA*2.2 – около 3,5 В (примерно среднее положение движка резистора *R*27). При превышении напряжения на выходе 125...128 В, например, в режиме холостого хода, откроется составной стабилитрон *VD*14*-VD*16, и компаратор *DA*2.2 также откроет оптопару *U*1, а микросхема *DА*1 уменьшит время открытого состояния транзистора *VT*2.

На транзисторе *VT*3 и регулируемом источнике образцового напряжения *DА*4 собран стабилизированный источник питания ОУ и оптопары.

Питание микросхемы *DА*1 в момент включения осуществляется через резисторы *R*7*, R*8. В установившемся режиме микросхема питается от дополнительной обмотки трансформатора *T*1 через стабилизатор на транзисторе *VT*1. Эта же обмотка через резисторы *R*13*, R*16подключена к входу *TZE* (вывод 5) *DА*1, который служит для контроля момента нулевой энергии трансформатора *Т*1, что необходимо для определения момента очередного открытия транзистора *VT*2.

## 2 РАЗРАБОТКА СВЕТОДИОДНОГО СВЕТИЛЬНИКА ПОВЫШЕННОЙ НАДЁЖНОСТИ

# **2.1 Разработка силовой части**

## 2.1.1 Описание схемы

В качестве схемы силовой части была выбрана схема обратноходового преобразователя (рисунок 11) [5], так как он обладает следующими преимуществами:

- нечувствительность обратноходового преобразователя к короткому замыканию нагрузки;

- возможность регулирования выходного напряжения в широких пределах, а также поддержание требуемого выходного напряжения в условиях изменения напряжения питающей сети.

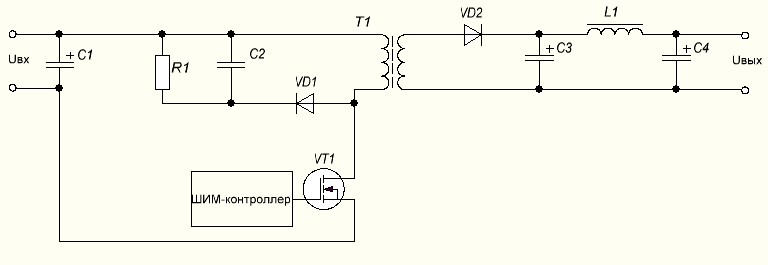


Рисунок 11 – Типовая схема обратноходового преобразователя

Основным элементом обратноходового преобразователя является многообмоточный дроссель *Т*1. Благодаря данному дросселю работа схемы разделяется на два этапа: передача энергии из сети питания в дроссель и передача накопленной энергии во вторичную цепь.

При замыкании ключа к первичной обмотке дросселя *Т*1 прикладывается напряжение источника питания *U*вх. В дросселе *Т*1 начинают почти линейно нарастать ток в первичной обмотке и магнитный поток в магнитопроводе, а следовательно накапливается энергия. В качестве ключа в данной схеме выступает полевой транзистор *VT*1. Наведённая на вторичной обмотке ЭДС запирает диод *VD*2, и ток во вторичной обмотке отсутствует. При размыкании ключа ток в первичной обмотке пропадает, но магнитный поток в дросселе *Т*1 не может измениться мгновенно, поэтому во вторичной обмотке начинает протекать почти линейно спадающий ток, отпирающий диод *VD*2. Этот ток заряжает конденсатор *С*3 и поступает в нагрузку. На первом этапе нагрузка питается только за счет энергии, полученной конденсатором *С*3 во время второго этапа. Частота повторения этапов обычно находится в интервале от 20 кГц до 1 МГц в зависимости от выбранного ШИМ-контроллера. Цепь *R*1, *C*2 и *VD*1 является цепью подавления выброса от индуктивности рассеяния.

# **2.1.1 Выбор элементов силовой цепи и моделирование источника питания**

Для анализа выбранной силовой цепи необходимо подобрать компоненты и произвести моделирование в специализированном ПО [6].

В качестве нагрузки выбраны четыре модуля светодиодов *GL*-05-18-350, каждый из которых имеет напряжение питания *U*вых = 18 В и потребляемый ток *I*мод = 112,5 мА, следовательно необходимый ток на выходе источника питания при параллельном соединении данных модулей будет равен *I*вых = 450 мА.

Исходные данные:

* входное напряжение: *U*вх= 220 В ± 20% (176-264 В);
* выходное напряжение/ток: *U*вых= 18 В, *I*вых= 450 мА;
* частота преобразования: *f* = 100 кГц;
* допустимые пульсации напряжения на выходе: *U*п= 0,1 В.

1) Расчёт и выбор входного конденсатора *C*1

Данный конденсатор можно выбрать из соотношения 1-2 мкФ на 1 Вт мощности, либо воспользоваться формулой (1):

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

где ;

*F* – частота питающей сети, Гц;

– минимальное постоянное напряжение сети, В;

максимальная мощность на выходе источника питания, Вт.

Для того, чтобы рассчитать параметры конденсатора *C*1 необходимо найти минимальное постоянное напряжение сети и максимальную мощность на выходе источника питания по формулам (2), (3):

|  |  |  |
| --- | --- | --- |
|  |  | (2) |

|  |  |  |
| --- | --- | --- |
|  |  | (3) |

Подставляя (2) и (3) в (1), получаем следующее выражение:

Ближайший по номиналу конденсатор 10 мкФ 400 В.

1. Расчет импульсного трансформатора *Т*1

Расчет трансформатора необходимо начать с определения индуктивностей обмоток. Будем считать, что трансформатор должен находиться в режиме разрывных токов при перегрузке в 20%. При этом мы будем уверены, что в режиме регулирования напряжения у нас не возникнет проблем с петлей обратной связи, и не возникнет дополнительных неучтенных динамических потерь в силовом ключе при его открывании.

Для расчёта трансформатора необходимо найти время открытого и закрытого состояния транзистора. Для этого необходимо подобрать коэффициент заполнения *D*. Примем его равным 43%. Для начала необходимо найти период по формуле (4).

|  |  |  |
| --- | --- | --- |
|  |  | (4) |

где *f* – номинальная частота преобразования, Гц.

Далее находим время открытого состояния транзистора по формуле (5).

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

*.*

Зная время открытого состояния транзистора, можно найти время закрытого состояния транзистора по формуле (6):

|  |  |  |
| --- | --- | --- |
|  |  | (6) |

Найдем удельную энергию, запасаемую в трансформаторе за время открытого ключа по формуле (7):

|  |  |  |
| --- | --- | --- |
|  |  | (7) |

Для нахождения импульсного тока в первичной цепи воспользуемся формулой:

|  |  |  |
| --- | --- | --- |
|  |  | (8) |

где – индуктивность первичной обмотки, которую можно найти по (9).

|  |  |  |
| --- | --- | --- |
|  |  | (9) |

где

Подставим полученное значение индуктивности в (8) и найдем импульсный ток в первичной цепи:

Далее находим среднеквадратичное значение тока в первичной цепи:

|  |  |  |
| --- | --- | --- |
|  |  | (10) |

где

Определим индуктивность вторичной обмотки:

|  |  |  |
| --- | --- | --- |
|  |  | (11) |

где

Импульсный ток вторичной обмотки рассчитаем по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (12) |

Найдем коэффициент трансформации:

|  |  |  |
| --- | --- | --- |
|  |  | (13) |

Рассчитаем напряжение на силовом ключе:

|  |  |  |
| --- | --- | --- |
|  |  | (14) |

где

Выберем сердечник *ETD39/20/13* из феррита *N87* с воздушным зазором 0,5 мм.

− индуктивность на виток.

− эффективная площадь сечения магнитопровода.

Рассчитаем количество витков первичной обмотки по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (15) |

Примем количество витков первичной обмотки равным 84.

Найдем размах индукции в сердечнике:

|  |  |  |
| --- | --- | --- |
|  |  | (16) |

Найдем количество витков вторичной обмотки:

|  |  |  |
| --- | --- | --- |
|  |  | (17) |

Примем количество витков вторичной обмотки равным 14.

Рассчитаем уточнённый коэффициент трансформации по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (18) |

Найдем уточнённое напряжение на силовом ключе:

|  |  |  |
| --- | --- | --- |
|  |  | (19) |

Рассчитаем номинальный ток в первичной цепи:

|  |  |  |
| --- | --- | --- |
|  |  | (20) |

Найдем номинальное время открытого состояния ключа по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (21) |

Найдем среднеквадратичное значение тока первичной цепи:

|  |  |  |
| --- | --- | --- |
|  |  | (22) |

Рассчитаем ток вторичной цепи по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (23) |

Далее найдём время обратного хода:

|  |  |  |
| --- | --- | --- |
|  |  | (24) |

Определим среднеквадратичное значение тока вторичной цепи по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (25) |

Постоянная составляющая тока вторичной цепи находится по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (26) |

Найдём переменную составляющую тока вторичной цепи:

|  |  |  |
| --- | --- | --- |
|  |  | (27) |

1. Выбор силового ключа

Выбор силового ключа выполним с учётом рассчитанных значений:

- максимальный ток

- напряжение на ключе

В качестве силового ключа выберем транзистор *IRFBC*40.

Время выключения транзистора

Рассчитаем потери в ключе по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (28) |

Найдем динамические потери в силовом ключе:

|  |  |  |
| --- | --- | --- |
|  |  | (29) |

1. Выбор элементов цепи *RCD*-снаббера

Рассчитаем отраженное напряжение трансформатора:

|  |  |  |
| --- | --- | --- |
|  |  | (30) |

Примем фиксируемое напряжение .

Рассчитаем величину сопротивления резистора, на который будет разряжаться конденсатор снабберной цепи:

|  |  |  |
| --- | --- | --- |
|  |  | (31) |

Для сглаживания пульсаций в снабберной цепи определим ёмкость фильтрующего конденсатора:

|  |  |  |
| --- | --- | --- |
|  |  | (32) |

В качестве диода *VD*1 выберем *S*1*J* c максимальным временем восстановления 1,8 мкс.

1. Выбор диода Шоттки

Рассчитаем обратное напряжение на диоде *VD*2 по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (33) |

Выберем диод Шоттки *SKL*18.

Ток, протекающий через диод, равен:

1. Выбор конденсатора фильтра *C*3

Найдём минимальную ёмкость конденсатора *C*3:

|  |  |  |
| --- | --- | --- |
|  |  | (34) |

Вычислим среднеквадратичное значение тока через выходной конденсатор:

|  |  |  |
| --- | --- | --- |
|  |  | (35) |

Выбираем электролитический конденсатор К50-35 47 мкФ × 25 В.

ESR = 0,9 Ом − эквивалентное последовательное сопротивление.

Рассчитаем пульсации на конденсаторе *С*3 по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (36) |

1. Выбор элементов *LC*-фильтра

В качестве дросселя *LC*-фильтра возьмём *B*82464*G*4682*M* с индуктивностью 6,8 мкГн.

Рассчитаем ёмкость конденсатора *C*4 по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (37) |

После выбора компонентов была создана модель для исследования работы силовой части обратноходового преобразователя с нагрузкой, представляющей собой 12 параллельных линеек светодиодов по 9 в каждой (рисунок 12).

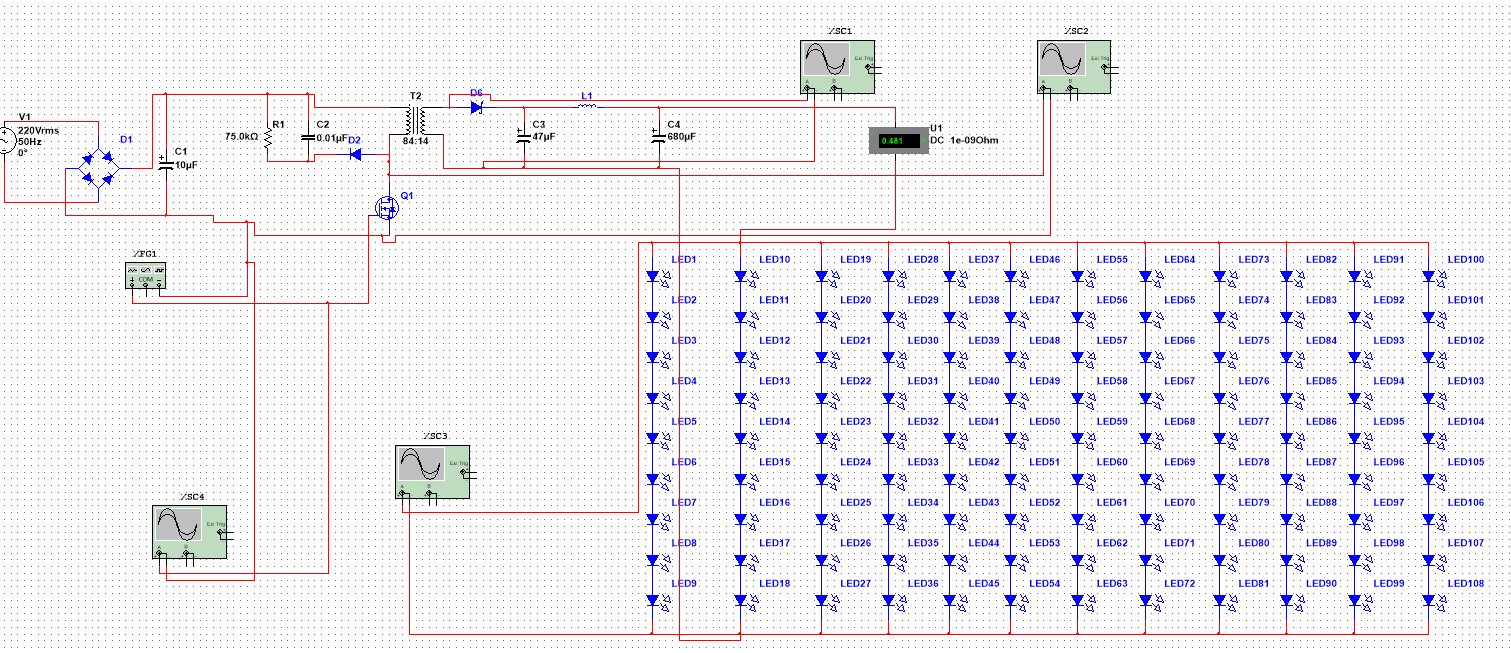


Рисунок 12 − Модель обратноходового преобразователя с подключенной нагрузкой

Для исследования работы схемы к ней подключены осциллографы, а вместо ШИМ-регулятора подключен функциональный генератор. На рисунках 13-16 представлены осциллограммы напряжений на выходе функционального генератора, на транзисторе, на вторичной обмотке импульсного трансформатора и на светодиодной нагрузке.

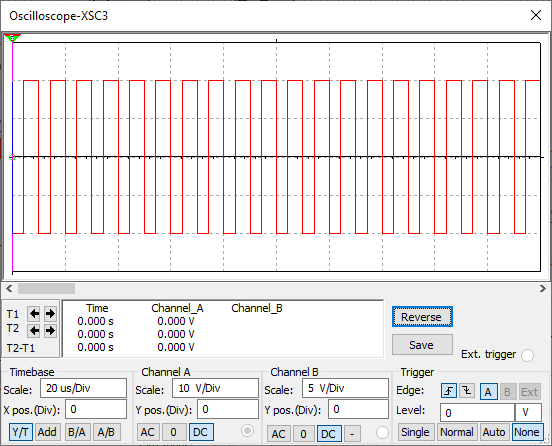


Рисунок 13 – Осциллограмма напряжения на выходе функционального генератора

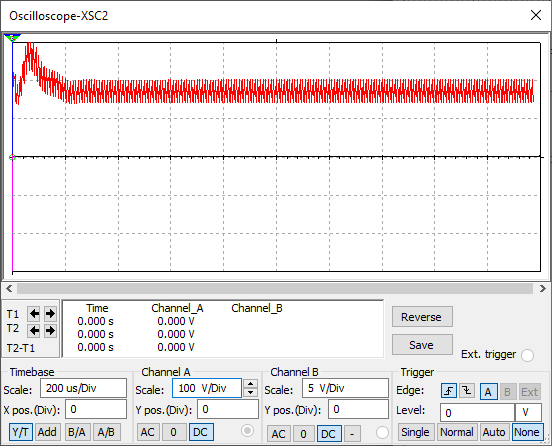


Рисунок 14 – Осциллограмма напряжения на полевом транзисторе

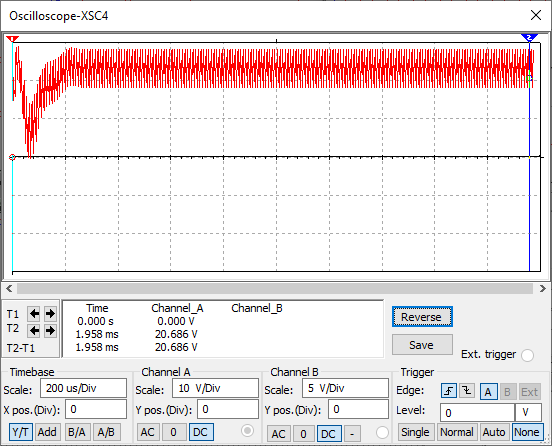


Рисунок 15 – Осциллограмма напряжения на вторичной обмотке трансформатора

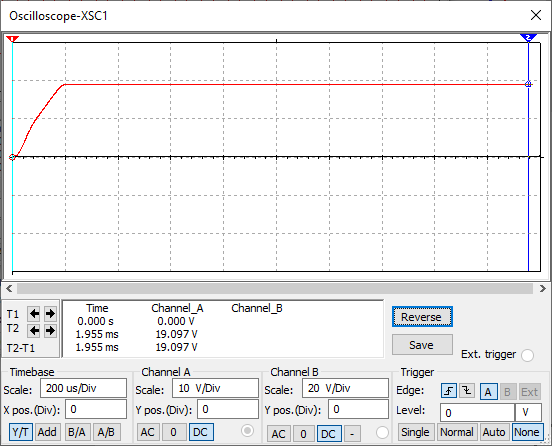


Рисунок 16 – Осциллограмма напряжения на нагрузке

Как видно из осциллограмм, представленных на рисунках 14-15, форма напряжения на выходе импульсного трансформатора искажена пульсациями на частоте ШИМ, но на нагрузку поступает уже сглаженное напряжение без пульсаций. Достигается это установкой на выходе преобразователя *CLC*-фильтра, который подавляет эти пульсации. Для измерения тока в цепь нагрузки включен амперметр, который показывает ток нагрузки в процессе моделирования. В установившемся режиме показания амперметра не превышают 500 мА. Рассматривая результаты моделирования работы силовой части ИИП и исходные данные расчётов, можно сделать вывод, что компоненты силовой части импульсного источника питания подобраны верно. В приложении Б представлен перечень элементов.

## 2.2 Разработка системы управления

## 2.2.1 Описание схемы

Схема импульсного источника питания для светодиодов с фиксированной шириной импульса не обеспечивает защиту нагрузки от изменений значений тока. Для компенсации изменений тока от номинального значения применяются схемы источников импульсных источников питания с обратной связью по току. Функциональная схема такого источника питания представлена на рисунке 18.

Выпрямитель и сглаживающий фильтр

Силовой ключ

Импульсный трансформатор

Выпрямитель и сглаживающий фильтр

Датчик тока

Нагрузка

Система управления

Uвх

Рисунок 18 – Функциональная схема импульсного источника питания

На вход импульсного источника питания подаётся переменное напряжение, которое выпрямляется с помощью диодного моста. После выпрямителя напряжение протекает через фильтрующий конденсатор и поступает на трансформатор. В качестве системы управления выступает управляемый ШИМ-генератор, реализованный с помощью микросхемы. В начальный момент времени он генерирует импульсы заданной частоты. Со вторичной обмотки трансформатора снимается пониженное напряжение и выпрямляется с помощью диода. Затем выпрямленное напряжение поступает на фильтр вторичной цепи, и далее – на нагрузку. В цепь нагрузки для контроля изменений тока подключен датчик тока, падение напряжения с которого поступает на измерительный вход управляемого генератора ШИМ-сигнала. На уставке генератора устанавливается напряжение, равное требуемому падению напряжения. В дальнейшем с помощью внутреннего компенсатора происходит расчёт сигнала ошибки и подаётся сигнал на встроенный в контроллер ШИМ-генератор, чтобы на силовой ключ поступал сигнал с измененной шириной импульса, для поддержания требуемой величины тока. В качестве компенсатора обратной связи обычно используется ПИ-регулятор. Он позволяет быстро и точно регулировать напряжение во вторичной цепи.

## 2.2.2 Разработка модели управляемого ШИМ-генератора и моделирование работы ИИП с обратной связью по току

Для изучения ИИП с управляемым ШИМ-генератором была разработана имитационная модель в среде *Matlab Simulink.* Силовая часть источника питания является обратноходовым преобразователем [7]. Имитационная модель ИИП представлена на рисунке 19.

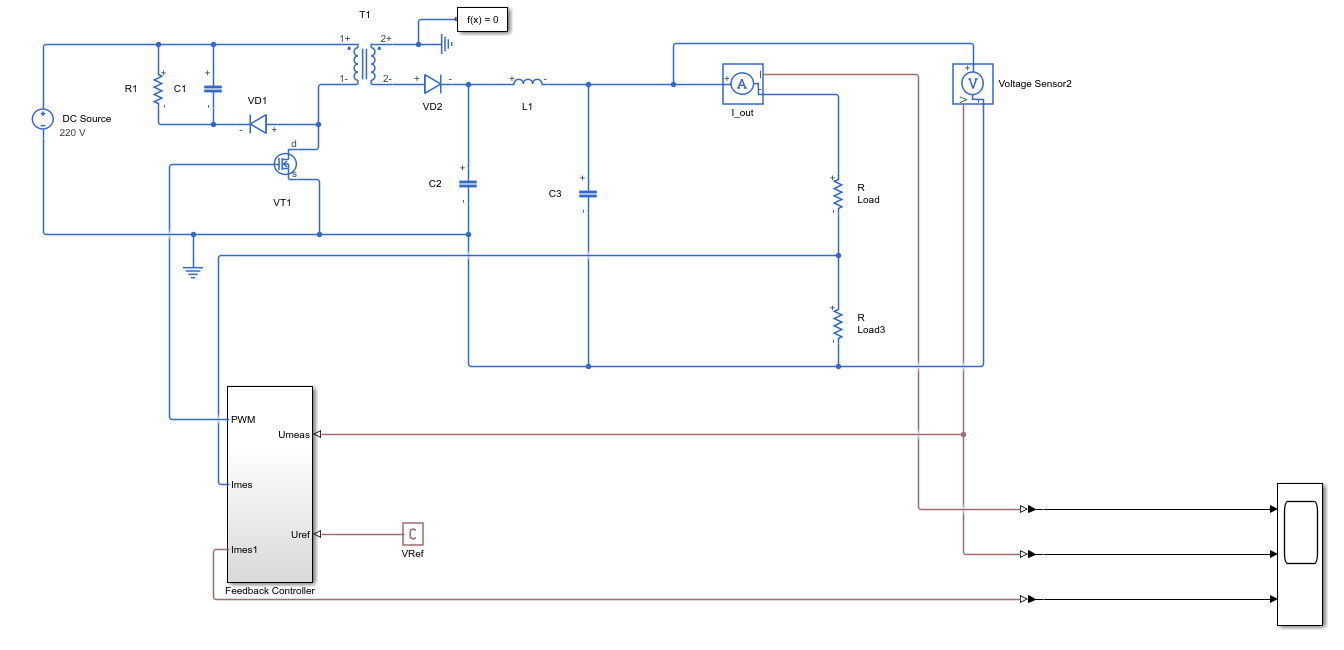


Рисунок 19 – Имитационная модель ИИП

В данной модели в качестве датчика тока используется резистор *Rs*, сопротивление которого равно 1 Ом. *Voltage sensor* измеряет напряжение на нагрузке, которое поступает на измерительный вход управляемого ШИМ-генератора. Данный регулятор имеет две уставки: по току и напряжению. В качестве опорного значения на уставке по току выступает требуемое падение напряжения, которое рассчитывается по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (38) |

где − опорное значение тока;

− сопротивление токового шунта.

В качестве элемента управления полевым транзистором *VT*1 в схеме собрана подсистема *Feedback Controller*. Она является управляемым генератором ШИМ-сигнала с ПИ-регулятором и компаратором. Структура подсистемы *Feedback Controller* представлена на рисунке 20.

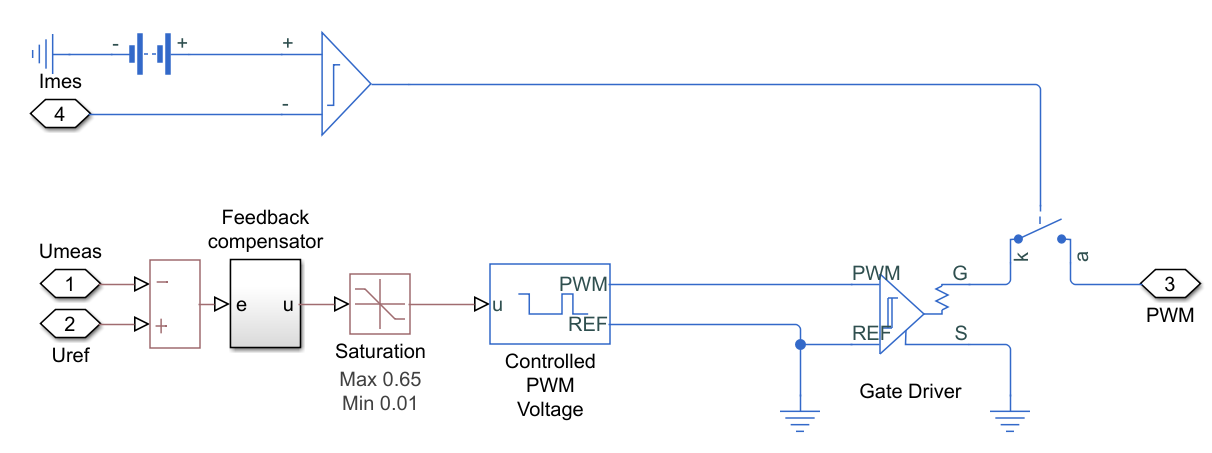


Рисунок 20 – Структурная схема подсистемы *Feedback Controller*

Подсистема *Feedback Controller* состоит из следующих блоков:

* управляемый ШИМ-генератор;
* драйвер полевого транзистора;
* компаратор;
* ПИ-регулятор.

ПИ-регулятор является элементом компенсации ошибки регулирования в цепи регулирования напряжения. Структурная схема блока ПИ-регулятора представлена на рисунке 21.

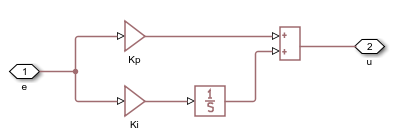


Рисунок 21 – Структурная схема ПИ-регулятора

Параметры ПИ-регулятора представлены на рисунках 22, 23.

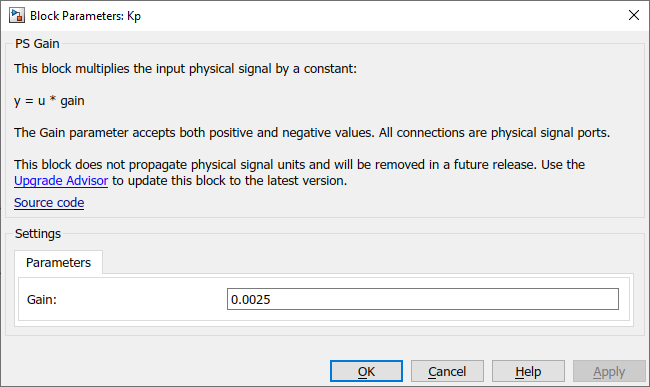


Рисунок 22 – Параметры пропорциональной составляющей

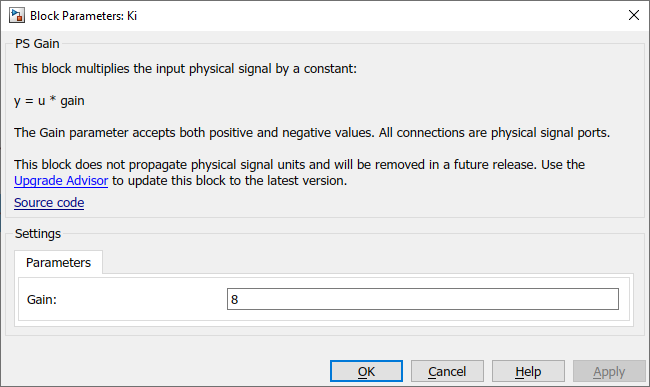


Рисунок 23 – Параметры интегральной составляющей

Для ограничения значения тока в цепи установлен компаратор, который при превышении заданного значения тока компаратор подаёт логический «0» на затвор полевого транзистора, тем самым регулируя длительность его открытого состояния. После настройки ПИ-регулятора была промоделирована работа импульсного источника питания. В качестве эквивалента нагрузки в цепи включен резистор сопротивлением 40 Ом. Результаты работы имитационной модели ИИП представлены на рисунке 24.

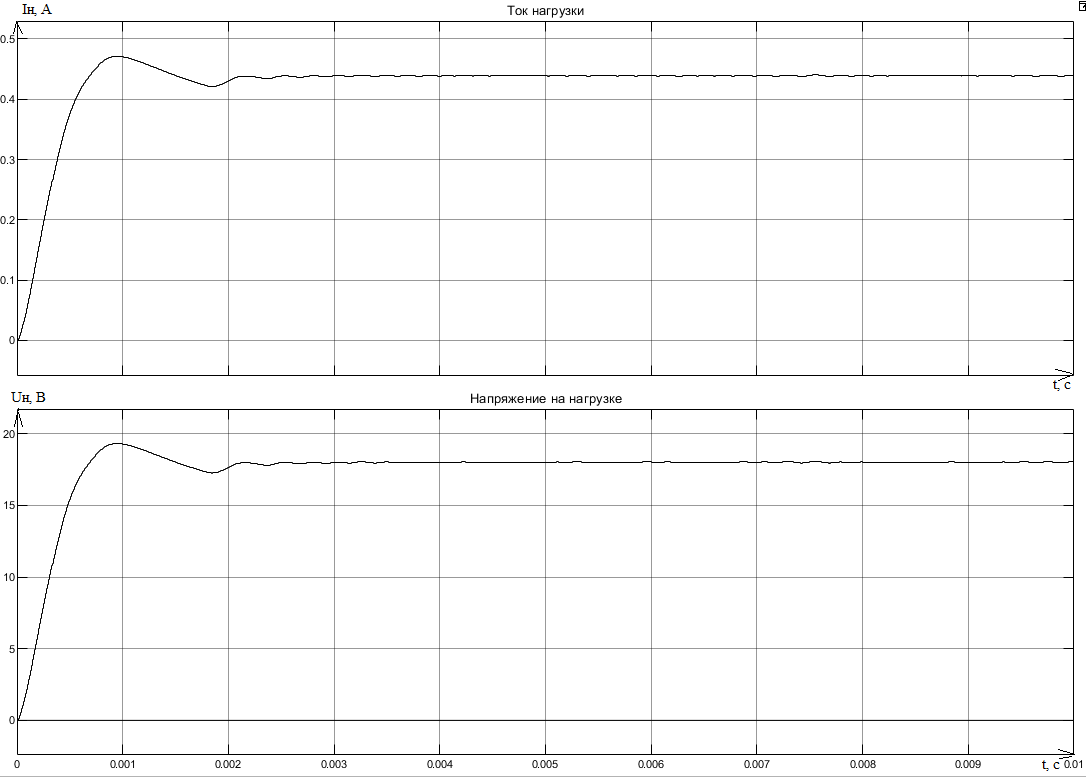


Рисунок 24 – Результат моделирования работы ИИП

По временным диаграммам тока и напряжения на нагрузке видно, что ток и напряжение принимают значения 0,45 А и 18 В за время переходного процесса 0,002 с. Перерегулирование не превышает 10%, следовательно система стабилизации позволяет достичь необходимых показателей качества регулирования.

## 2.3 Исследование работы импульсного источника питания с цифровой системой управления при изменении нагрузки

Для исследования работы источника питания при изменении нагрузки была разработана модель (рисунок 25), которая имитирует выход из строя одного из светодиодных модулей, включенных параллельно, либо отдельно взятого светодиода в модуле.

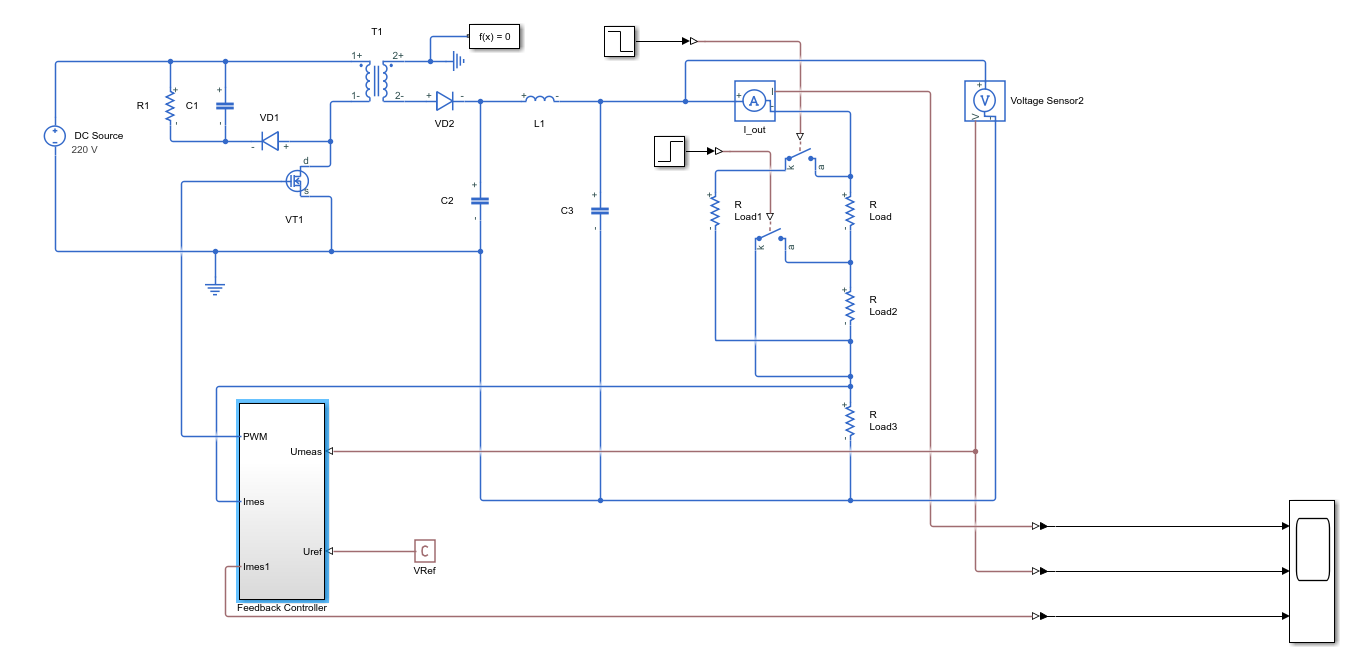


Рисунок 25 − Имитационная модель ИИП с реализацией изменения нагрузки

В данной модели для имитации выхода из строя одного из двух светодиодных модулей включены параллельно сопротивления. Неполадку имитирует элемент *Ideal Switch*, который управляется блоком *Step* и отключает параллельно включенную ветвь, имитируя тем самым выход из строя одного из двух светодиодных модулей. На рисунке 26 представлены временные диаграммы тока и напряжения на нагрузке и напряжения на выходе компаратора, который входит в состав системы управления.

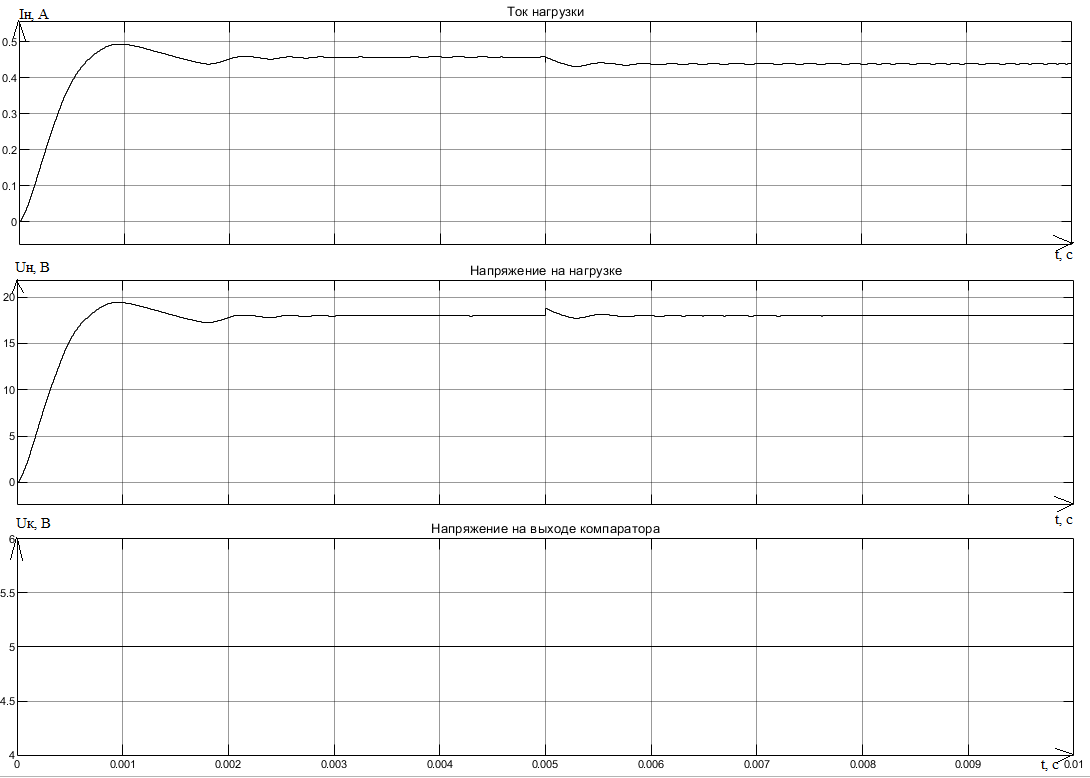


Рисунок 26 − Временные диаграммы тока и напряжения на нагрузке при выходе из строя светодиодного модуля

По временным диаграммам видно, что при выходе из строя светодиодного модуля, включенного параллельно, ток на нагрузке снижается и принимает значение 0,43 А. Напряжение на нагрузке слегка повышается на 0,5 В, но за счёт работы ПИ-регулятора принимает значение, заданное уставкой (18 В).

Для имитации выхода из строя одного светодиода в модуле в модели установлен *Ideal Switch*, управляемый блоком *Step*, который имитирует пробой светодиода. При переключении ключа ток протекает в обход последовательно подключенного резистора, имитируя тем самым выход из строя светодиода. На рисунке 27 представлены временные диаграммы тока и напряжения на нагрузке и напряжения на выходе компаратора, который входит в состав системы управления.

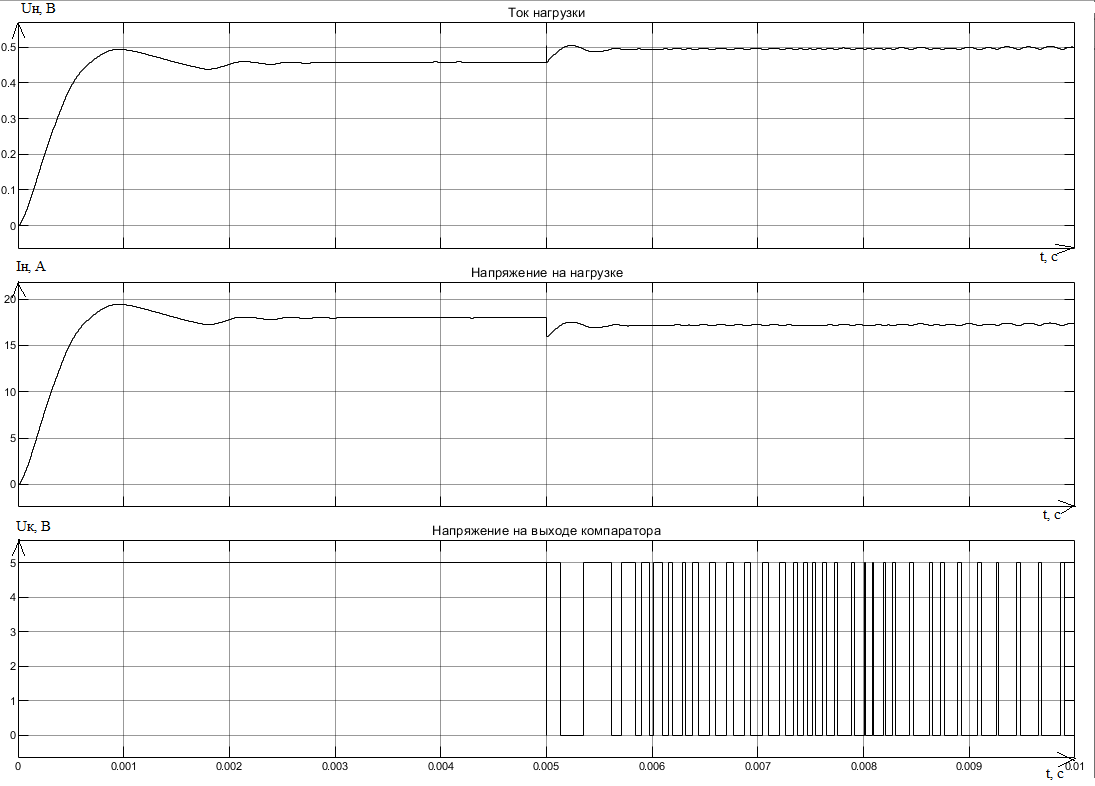


Рисунок 27 − Временные диаграммы тока и напряжения на нагрузке при выходе из строя светодиода в модуле

По временным диаграммам видно, что при выходе из строя светодиода повышается ток на нагрузке, но так как он превышает заданное уставкой компаратора значение (0,5 А), компаратор пытается граничить его значение, изменяя ширину импульса. Также наблюдается небольшое падение напряжения на нагрузке на 0,5 В, вызванное работой компаратора.

## 2.4 Разработка программного обеспечения контроллера

В качестве элемента регулирования тока в обратноходовом импульсном источнике питания выступает микроконтроллер. Он работает в качестве управляемого генератора ШИМ-сигнала с обратной связью по току. Для его корректной работы необходимо написать программное обеспечение, которое будет описывать поведение контроллера в ИИП. В зависимости от выбранного МК необходимо подбирать определённую среду программирования. Достаточно простой и универсальной средой является *Flowcode*. Данная среда программирования позволяет создавать прошивку контроллера не углубляясь в изучение какого-либо языка программирования. Достаточно знать основы алгоритмизации и с помощью блок-схемы создать необходимые инструкции для микроконтроллера. В качестве микроконтроллера, для которого создается программный код в данной ВКР выбран *Attiny13*, который имеет достаточно компактный корпус. Его технические характеристики представлены ниже:

1. Передовая *RISC* архитектура

* 120 команд, большинство которых выполняется за один тактовый цикл
* 32 8 битных рабочих регистра общего применения
* Полностью статическая архитектура

2 Энергонезависимая память программ и данных

* 1 КБ внутрисистемно программируемой *Flash* памяти программы, способной выдержать 10 000 циклов записи/стирания
* 64 байта внутрисистемно программируемой *EEPROM* памяти данных, способной выдержать 100 000 циклов записи/стирания
* 64 байта встроенной *SRAM* памяти (статическое ОЗУ)
* Программируемая защита от считывания самопрограммируемой *Flash* памяти программы и *EEPROM* памяти данных

3 Характеристики периферии

* Один 8- разрядный таймер/счетчик с отдельным предделителем и два ШИМ канала
* 4 канальный 10 битный АЦП со встроенным ИОН
* Программируемый сторожевой таймер со встроенным генератором
* Встроенный аналоговый компаратор

4 Специальные характеристики микроконтроллера

* Встроенный отладчик *debugWIRE*
* Внутрисистемное программирование через *SPI* порт
* Внешние и внутренние источники прерывания
* Режимы пониженного потребления *Idle, ADC Noise Reduction* и *Power-down*
* Усовершенствованная схема формирования сброса при включении
* Программируемая схема обнаружения кратковременных пропаданий питания
* Встроенный откалиброванный генератор

5 Порты ввода - вывода и корпусное исполнение

* 8 выводные *PDIP* и *SOIC* корпуса: 6 программируемых линий ввода-вывода

6 Диапазон напряжения питания

* от 1.8 до 5.5 В

7 Потребление

* Активный режим:
* 290 мкА при частоте 1 МГц и напряжении питания 1.8 В
* Режим пониженного потребления
* 0.5 мкА при напряжении питания 1.8 В

Блок-схема *Attiny13* представлена на рисунке 28.

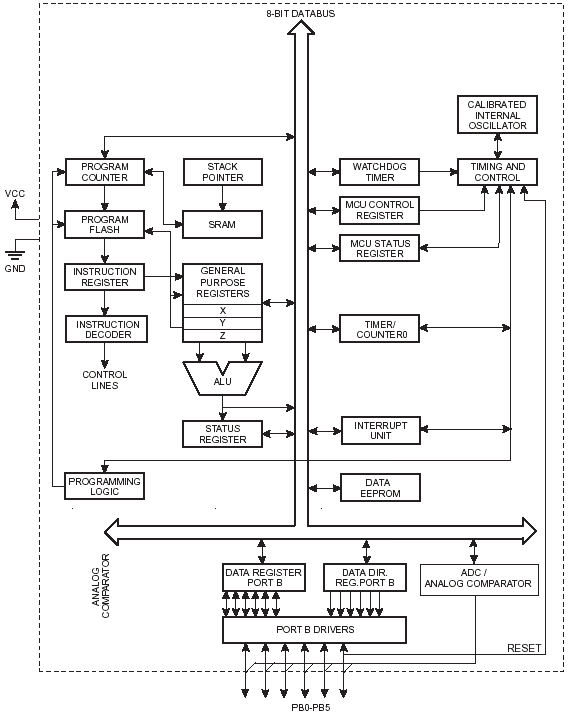


Рисунок 28 – Блок-схема микроконтроллера

Для старта работы в среде программирования *Flowcode* необходимо создать проект и выбрать контроллер. Меню выбора контроллера представлено на рисунке 29.



Рисунок 29 – Меню выбора контроллера

После выбора микроконтроллера необходимо указать его тактовую частоту в настройках проекта (рисунок 30).

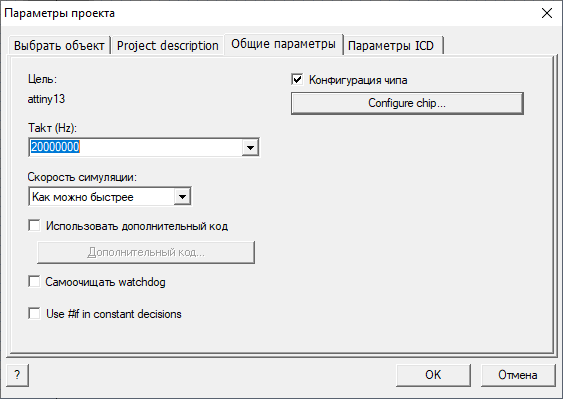


Рисунок 30 – Окно настройки проекта

После настройки проекта открывается окно рабочей области (рисунок 31), в котором можно с помощью блок-схемы создать инструкции для контроллера.

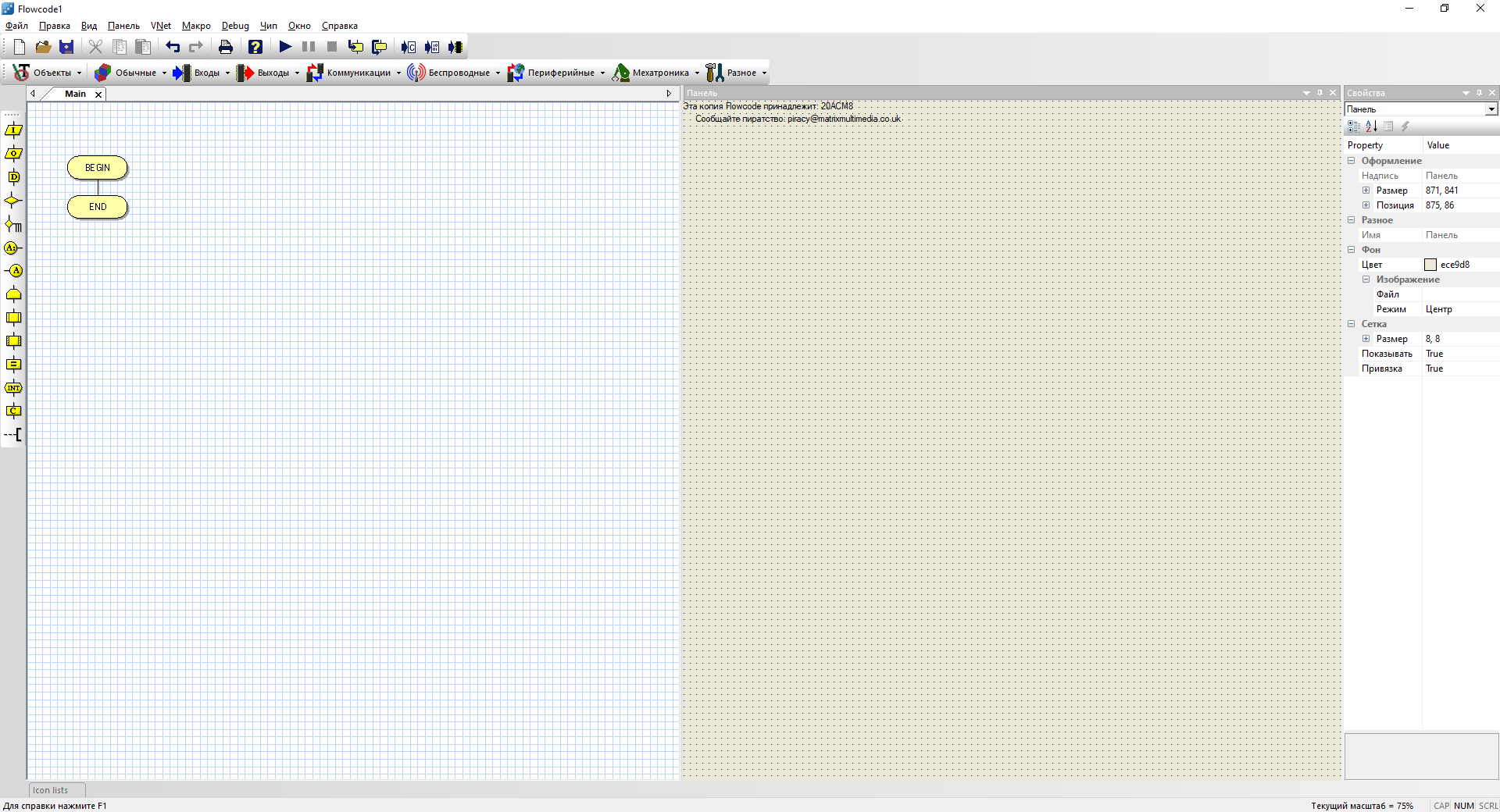


Рисунок 31 – Рабочая область *Flowcode*

В качестве системы управления в импульсном источнике питания выбран ПИ-регулятор, который регулирует ширину импульсов ШИМ-сигнала. Данный вид регулятора можно получить, реализовав ПИД-регулятор, в котором дифференциальная составляющая будет равна нулю. На псевдокоде реализация ПИД-регулятора будет выглядеть следующим образом:

|  |
| --- |
| previous\_error = 0  integral = 0  loop:  error = setpoint - measured\_value  integral = integral + error \* dt  derivative = (error - previous\_error) / dt  output = Kp \* error + Ki \* integral + Kd \* derivative  previous\_error = error  wait(dt)  goto loop |

Данный код можно перенести и на контроллер. С помощью АЦП он будет считывать падение напряжения на токовом шунте, и сравнивать со значением, которое задано в качестве опорного напряжения. Опорное напряжение рассчитывается по формуле (35). Далее происходит вычитание измеренного напряжения из опорного, и результат сохраняется в переменной *error*. По полученному значению шибки рассчитываются значения всех составляющих ПИД-регулятора (пропорциональной, интегральной и дифкркциальной).

Блок-схема программной реализации ПИД-регулятора, разработанного в среде *Flowcode*, представлена на рисунке 32. Принцип работы программы поясняется следующим алгоритмом:

1. разрешается работа генератора ШИМ-сигнала;
2. устанавливается требуемое значение напряжения на уставки;
3. устанавливается требуемое значение тока уставки;
4. вычисляются коэффициенты ПИД-регулятора;
5. запускается цикл, в ходе которого происходят следующие действия:

* считывается с помощью АЦП падение напряжения на токовом шунте и преобразуется в значение тока;
* сравнивается значение тока на нагрузке с током уставки, и, если оно больше заданного значения, производится блокирование ШИМ-сигнала;
* считывается с помощью АЦП значение напряжения на нагрузке;
* рассчитываются коэффициенты ПИД-регулятора;
* рассчитывается управляющее воздействие на выходе ПИД-регулятора;
* формируется ШИМ-сигнал, ширина импульсов которого пропорциональна управляющему воздействию.

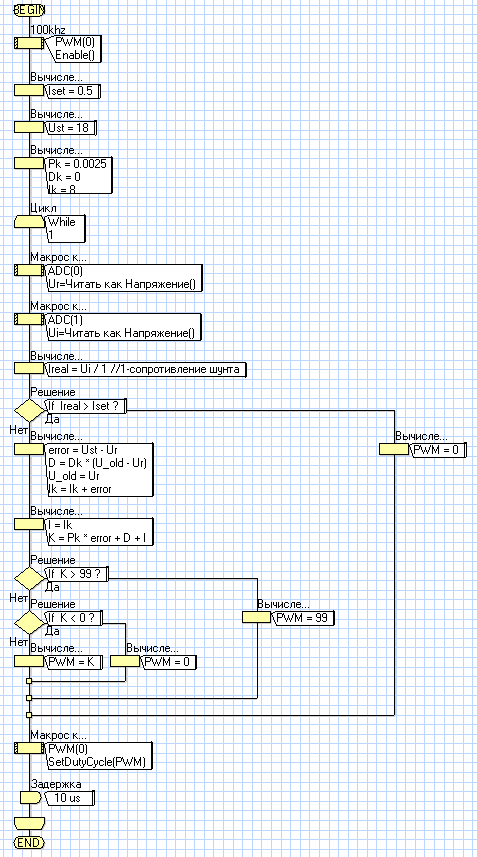


Рисунок 32 – Блок-схема программы ПИ-регулятора

Для формирования ШИМ-сигнала, управляющего сиорвым транзистором, в микроконтроллере используется внутренний генератор. *Flowcode* предусматривает настройку тактовой частоты этого генератора с помощью блока *PWM*, который находится в панели используемых элементов. Изображения панели элементов и параметров генератора ШИМ-сигнала представлены на рисунках 33 и 34 соответственно.

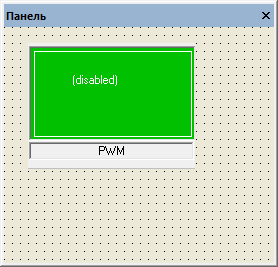


Рисунок 33 – Панель элементов

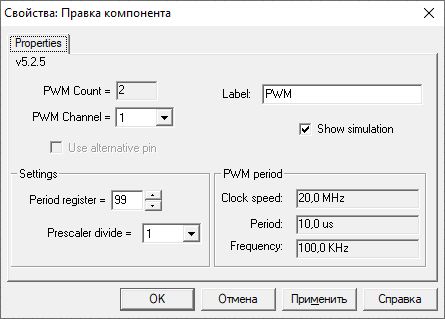


Рисунок 34 – Параметры ШИМ-генератора

После выполнения всех настроек и составления блок-схемы программы, она была записана во *Flash*-память микроконтроллера. Скомпилировать программу можно в следующих вариантах:

* компиляция в *C*;
* компиляция в *HEX*;
* компиляция и запись напрямую в микроконтроллер.

В приложении В представлен листинг кода, скомпилированный на языке *C*.

## 2.5 Разработка печатной платы импульсного источника питания

После того, как выбраны топология ИИП и тип микроконтроллера, который будет осуществлять регулирование тока с целью защиты светодиодов, необходимо разработать печатную плату. Существует множество специализированного ПО для разработки печатных плат, как в ручном режиме, так и в автоматизированном. В качестве инструмента для проектирования печатной платы была выбрана САПР *Altium Designer* [8]. Она позволяет производить трассировку печатных проводников, как в автоматическом режиме, так и в ручном. Для разработки печатной платы были определены типоразмеры всех дискретных электронных компонентов и интегральной микросхемы контроллера *ATtiny*13. На рисунке 35 представлено изображение корпуса микроконтроллера с названиями выводов портов.

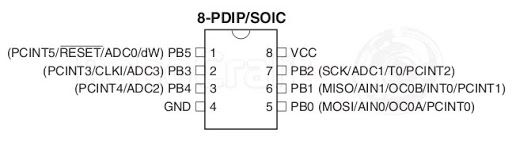


Рисунок 35 – Расположение выводов микроконтроллера *ATtiny*13

В соответствии с ранее разработанным программным кодом определяются порты, необходимые для реализации регулятора. В данном случае задействуются порты питания (*Vcc* и *GND*), АЦП (*PB*5) и ШИМ (*PB*0).

Разработка печатной платы в *Altium Designer* состоит из нескольких этапов.

**Создание проекта**

Для создания проекта необходимо перейти *File→New→Project* (рисунок 36). После чего открывается окно создания проекта, которое представлено на рисунке 37.

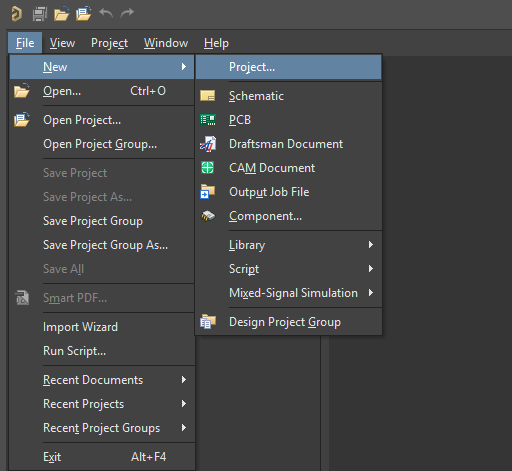


Рисунок 36 – Создание нового проекта

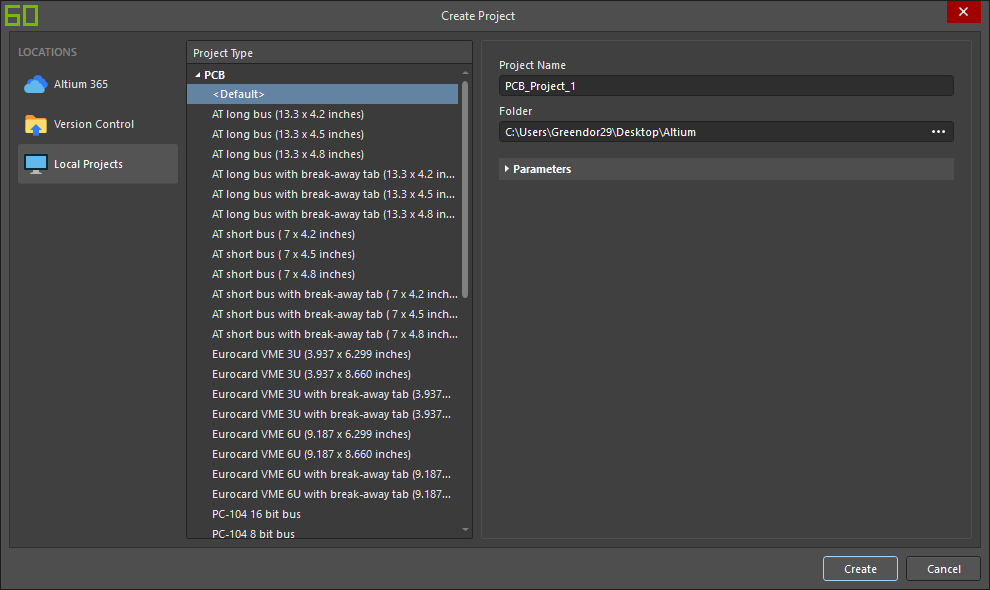


Рисунок 37 – Окно создания проекта

После всех вышеперечисленных манипуляций новый проект появится в списке файлов (рисунок 38).

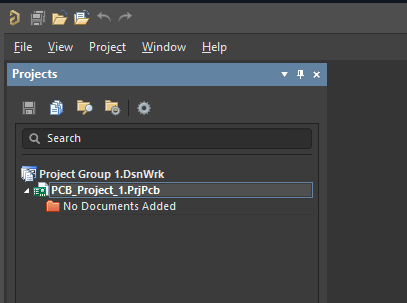


Рисунок 38 – Новый проект в дереве документов

**Разработка библиотеки графических обозначений**

Для создания библиотеки графических обозначений необходимо нажать правой кнопкой мыши на проекте и выбрать *Add New to Project→Schematic Library* (Рисунок 39).

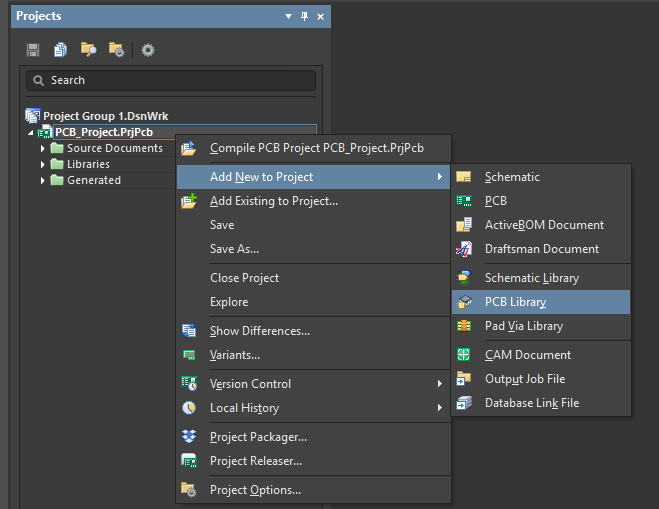


Рисунок 39 – Создание библиотеки условно графических обозначений

После создание файла библиотеки откроется окно рабочей области. Оно представлено на рисунке 40.

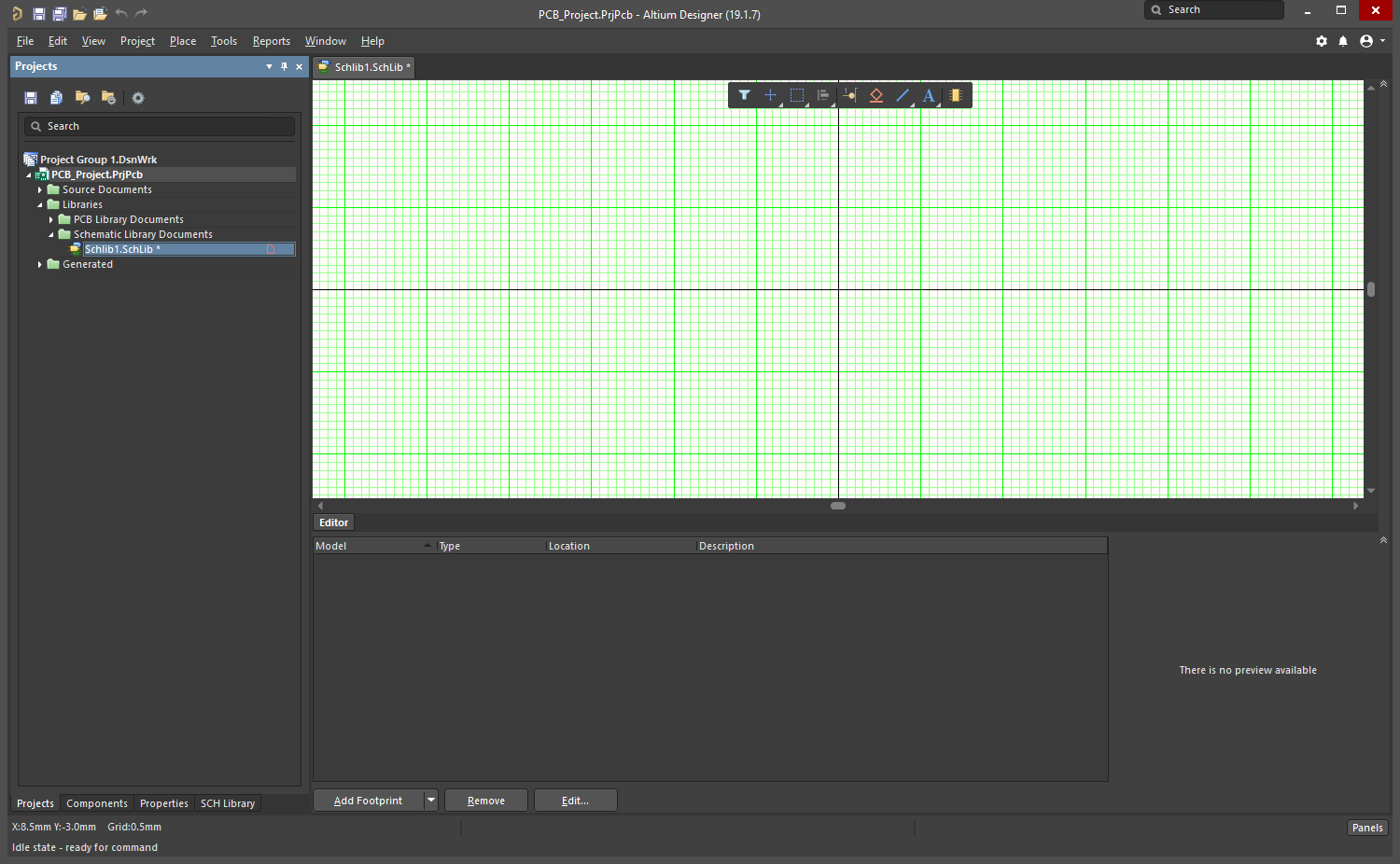


Рисунок 40 – Рабочая область

В появившейся рабочей области необходимо нарисовать условное обозначение элемента (рисунок 41).

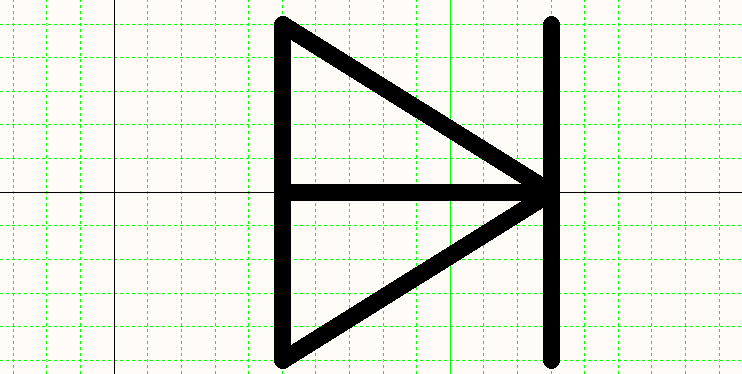


Рисунок 41 – Условное обозначение диода

Далее необходимо к условному обозначению дорисовать электрические выводы. Нарисованные электрические выводы представлены на рисунке 42.

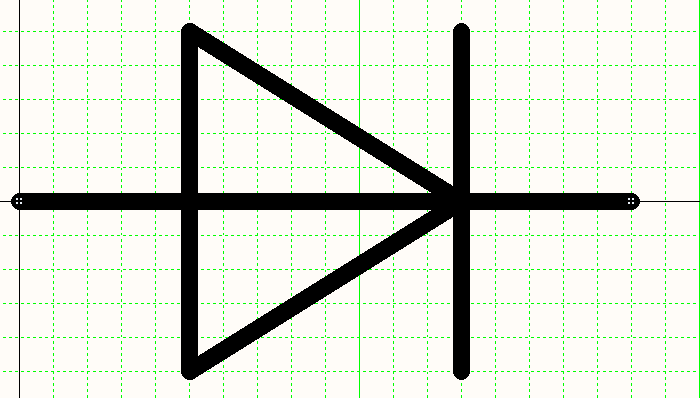


Рисунок 42 – Условное обозначение диода с электрическими выводами

После отрисовки электрических выводов им необходимо в настройках указать *Designator*, для привязки к нему посадочных мест. Настройка выводов указана на рисунке 43.

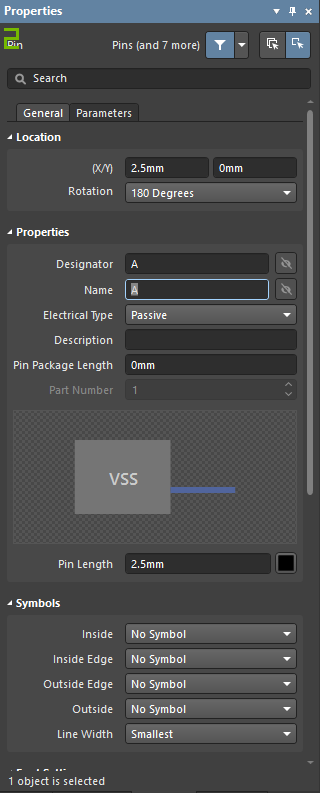


Рисунок 43 – Настройка электрических выводов

После настройки и отрисовки элемента необходимо ему присвоить условное обозначение на схеме. Это производится в графе *Designator* в параметрах элемента. После условного обозначения не обходимо ставить знак «?», для дальнейшей автоматической нумерации компонентов. Настройка условного обозначения диода представлена на рисунке 44.

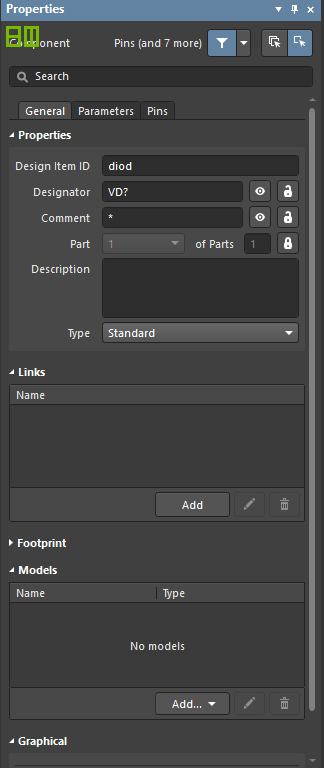


Рисунок 44 – Настройки условного обозначения диода

**Разработка библиотеки посадочных мест**

Для создания библиотеки посадочных мест необходимо нажать правой кнопкой мыши на проекте и выбрать *Add New to Project→PCB* Library (рисунок 45).

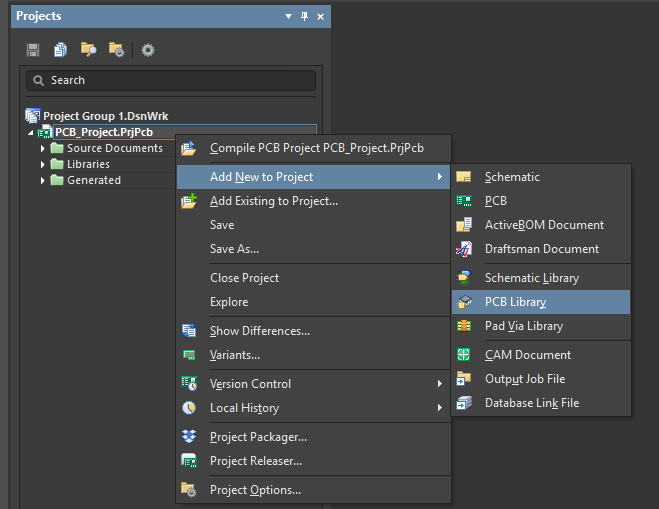


Рисунок 45 – Создание библиотеки посадочных мест

Создать посадочные места для компонентов можно через *Footprint Wizard* (для простых элементов, типа резистора, диода и конденсатора) или в ручном режиме.

При активации *Footprint Wizard* появляется окно, представленное на рисунке 46.

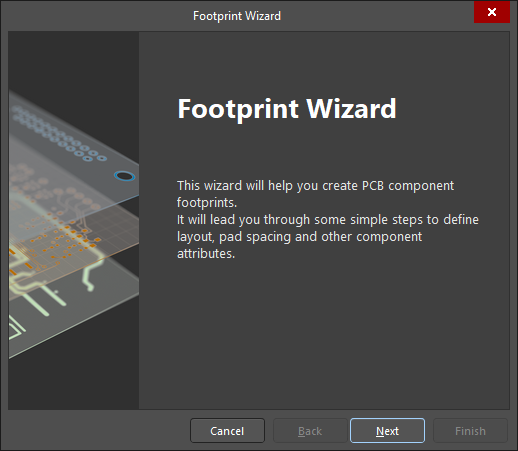


Рисунок 46 – Главное окно Footprint Wizard

Далее необходимо выбрать тип элемента, настроить способ монтажа, параметры отверстий и расстояние между ними, толщину линий шелкографии и название элемента. Данные настройки представлены на рисунках 47-52. Полученное посадочное место представлено на рисунке 53.

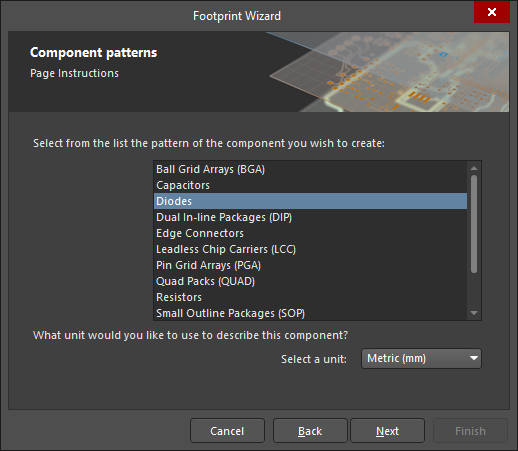


Рисунок 47 – Выбор типа элемента

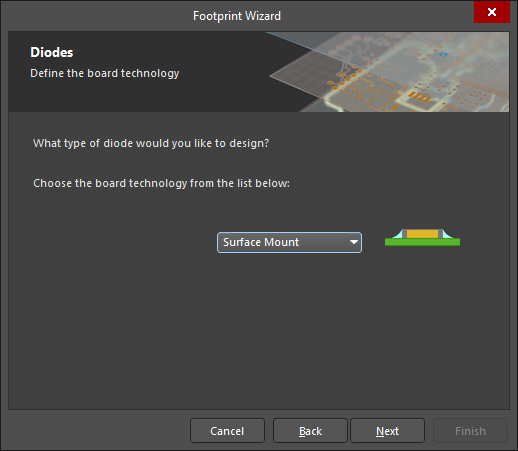


Рисунок 48 – Выбор способа монтажа

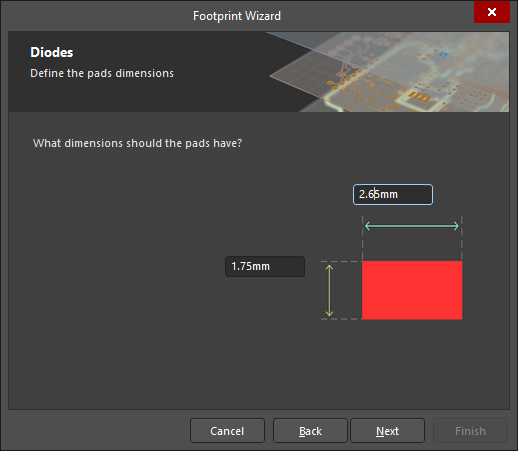


Рисунок 49 – Настройка площадки для установки компонента

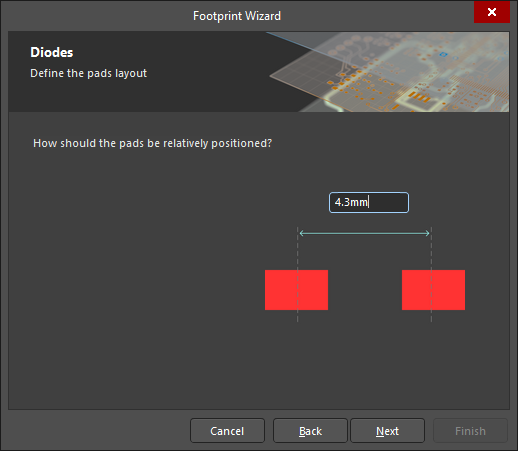


Рисунок 50 – Настройка расстояния между выводами

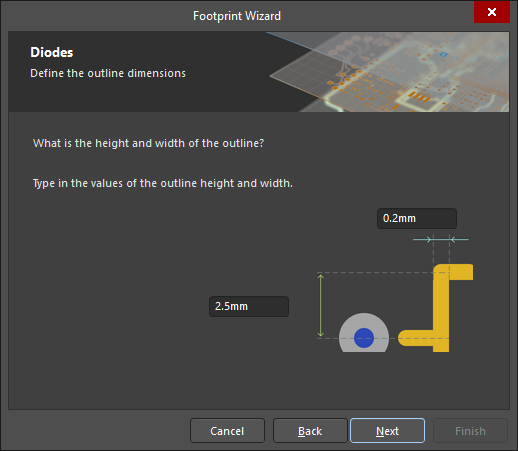


Рисунок 51 – Настройка толщины линий шелкографии

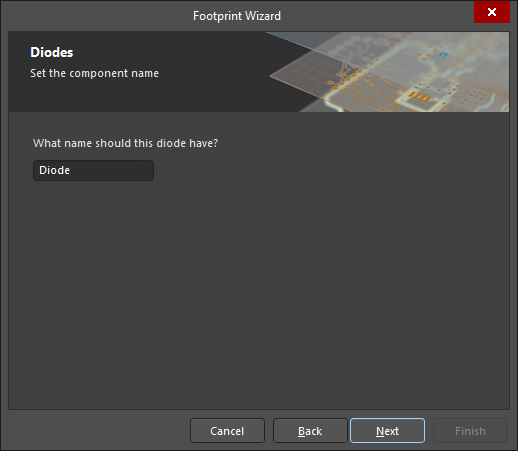


Рисунок 52 – Присвоения названия посадочному месту

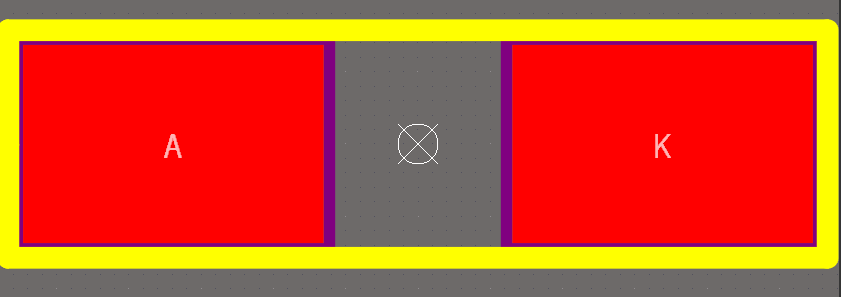


Рисунок 53 – Полученное посадочное место диода

**Разработка схемы электрической принципиальной схемы**

Для создания схемы электрической принципиальной необходимо нажать правой кнопкой мыши на проекте и выбрать *Add New to Project→Schematic* (рисунок 54).

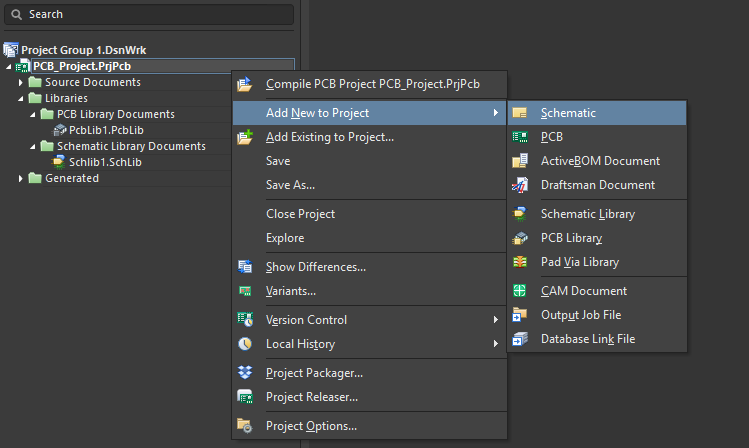


Рисунок 54 – Добавление файла схем

После создания схемного документа откроется рабочая область, которая представлена на рисунке 55.

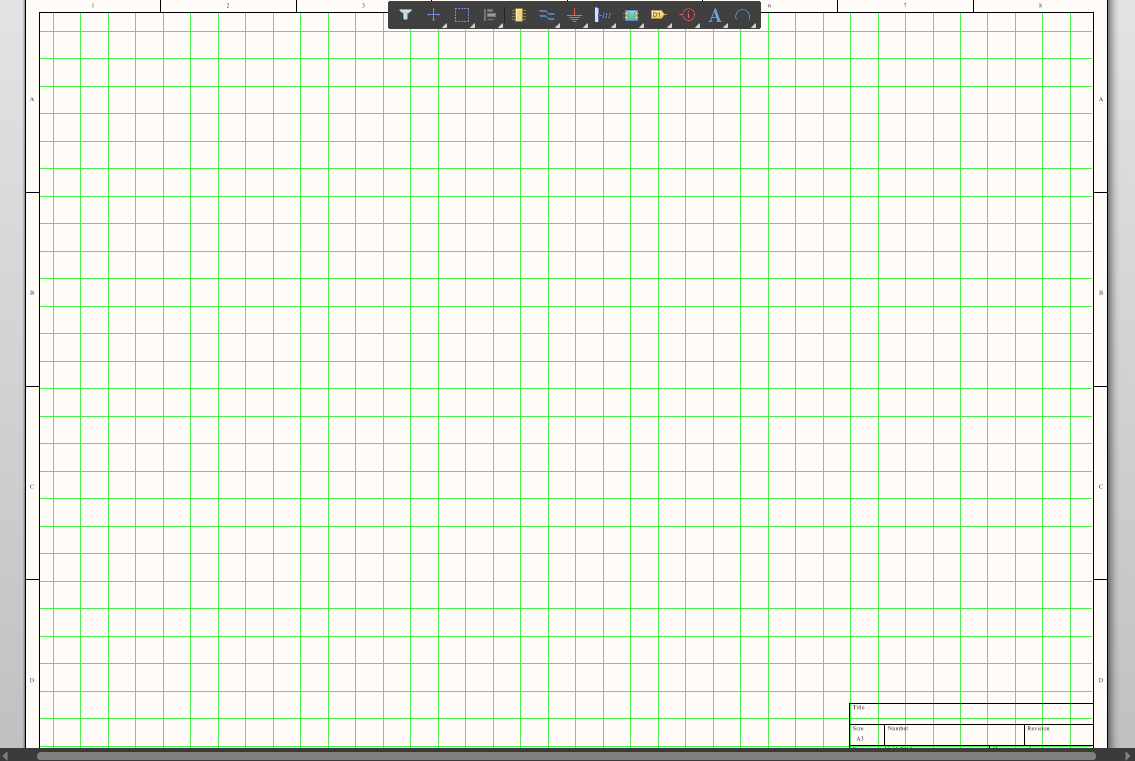


Рисунок 55 – Рабочая область для создания принципиальной схемы

В рабочей области размещаются элементы схемы, электрические выводы и проводники, соединяющие элементы. Полученная электрическая схема представлена на рисунке 56 и в приложении А.

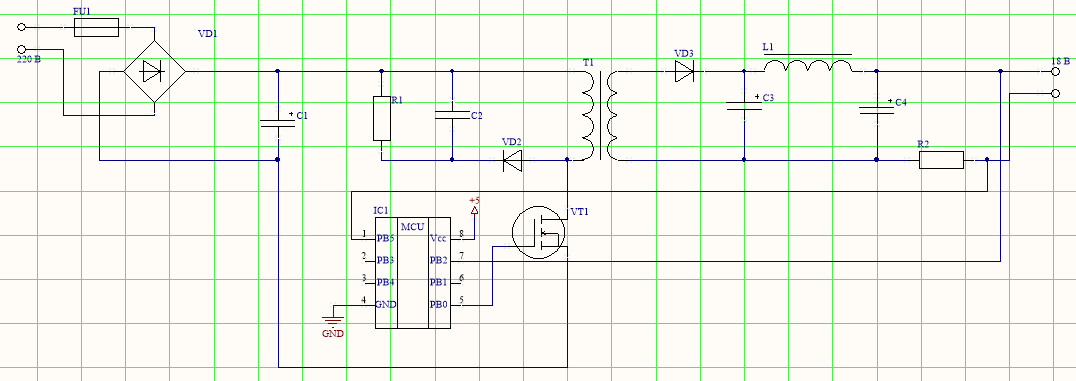


Рисунок 56 – Изображение принципиальной схемы

**Разработка печатной платы**

Для создания печатной платы необходимо нажать правой кнопкой мыши на проекте и выбрать *Add New to Project→PCB* (рисунок 57).

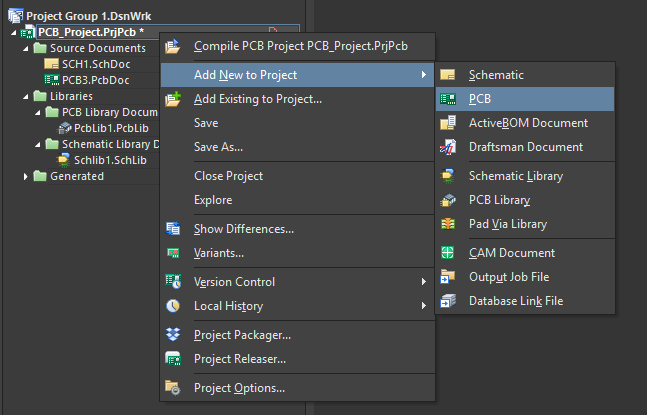
****

Рисунок 57 – Добавление файла печатной платы в проект

После добавления файла печатной платы необходимо со схемного документа импортировать все компоненты (рисунок 58). Результат выполнения этой операции представлен на рисунке 59.

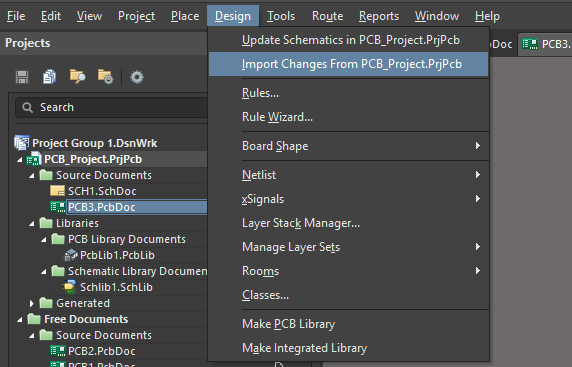
****

Рисунок 58 – Импорт элементов со схемного документа

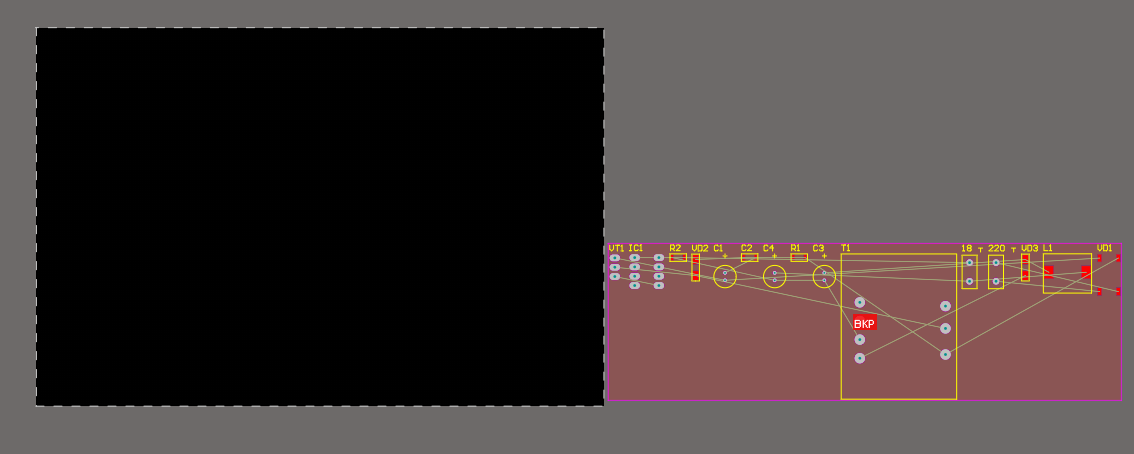


Рисунок 59 – Импортированные элементы

После импорта посадочные места размещаются в области печатной платы, после чего производится автоматическая трассировка контактных дорожек. Настройки автоматической трассировки и ее результат представлены на рисунках 60-65.

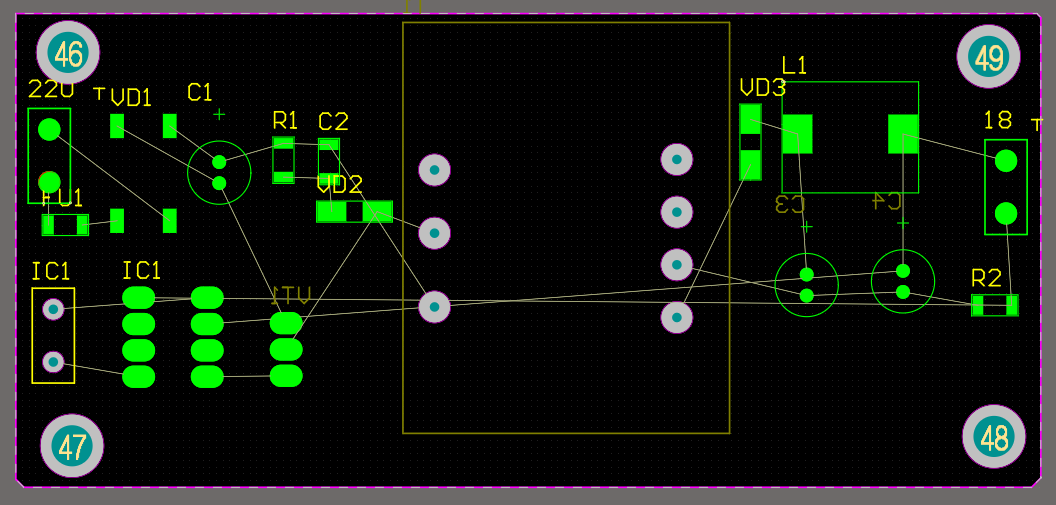


Рисунок 60 – Размещение элементов на печатной плате

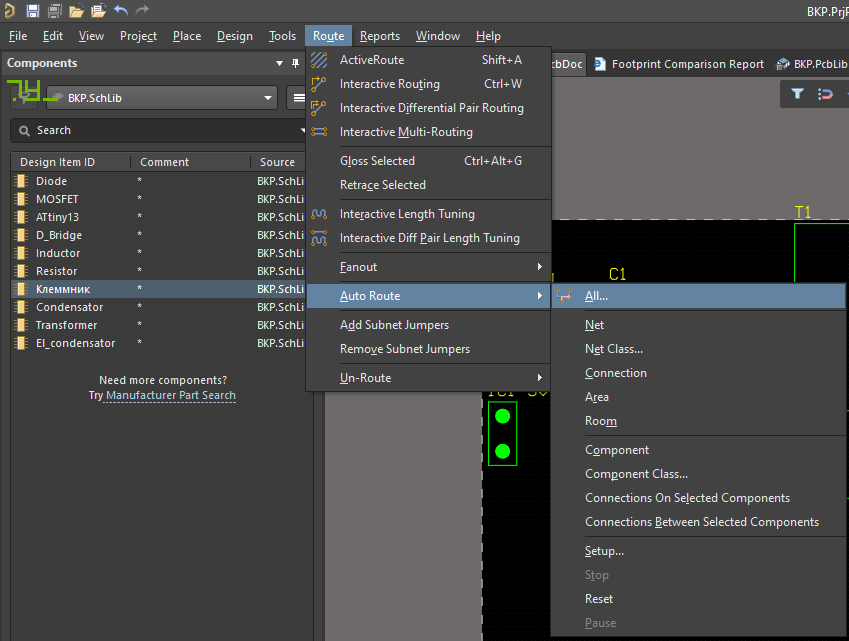


Рисунок 61 – Активация автоматической разводки контактных дорожек

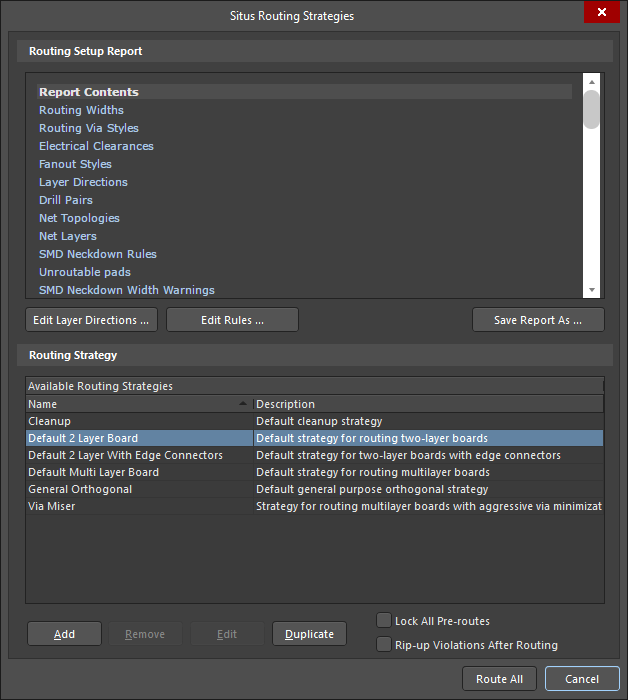


Рисунок 62 – Выбор правил разводки контактных дорожек

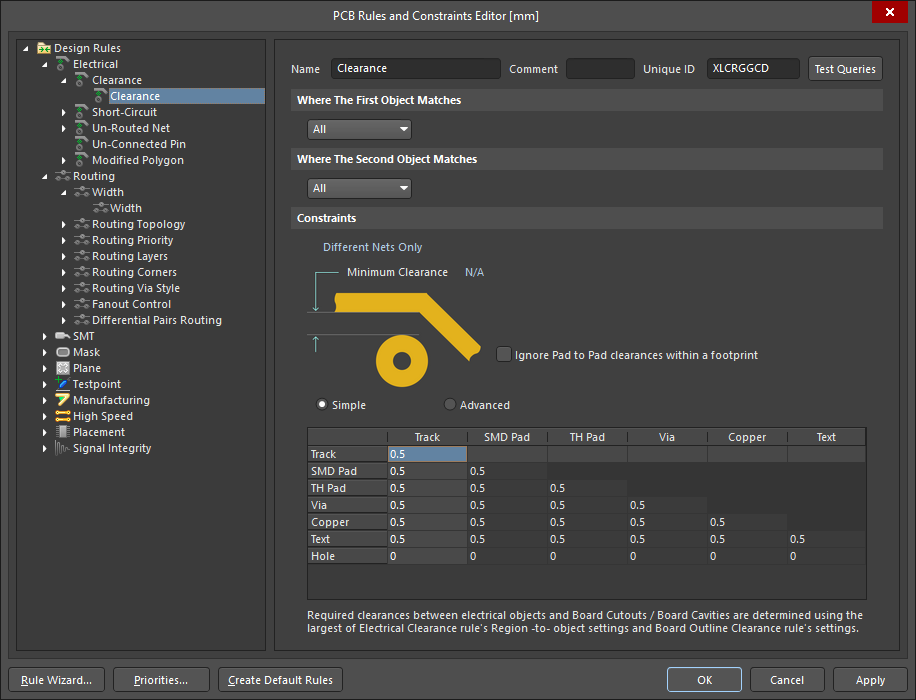


Рисунок 63 - Настройка расстояния между дорожками

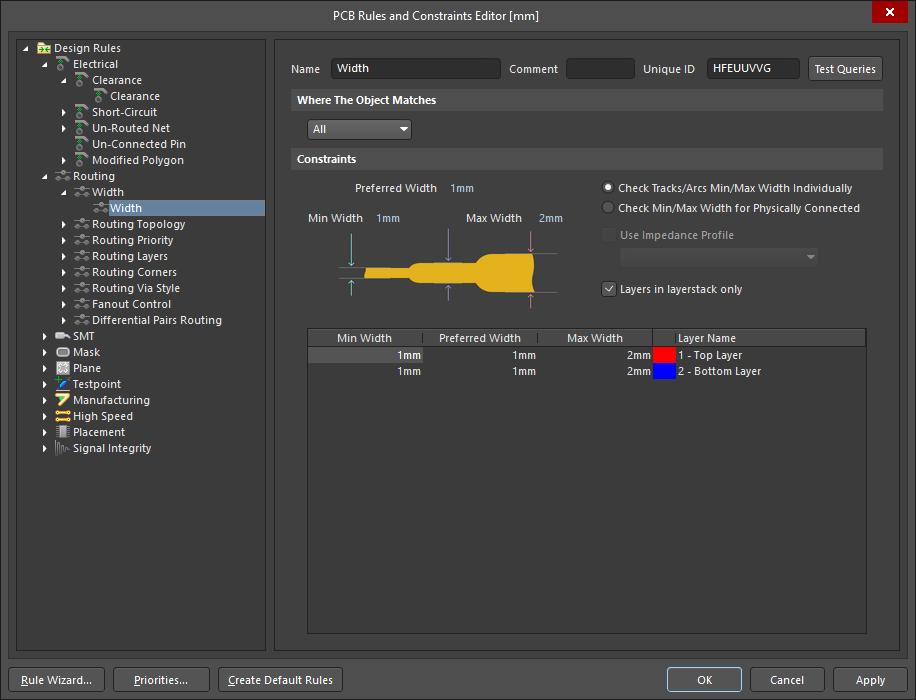


Рисунок 64– Настройка ширины печатных проводников

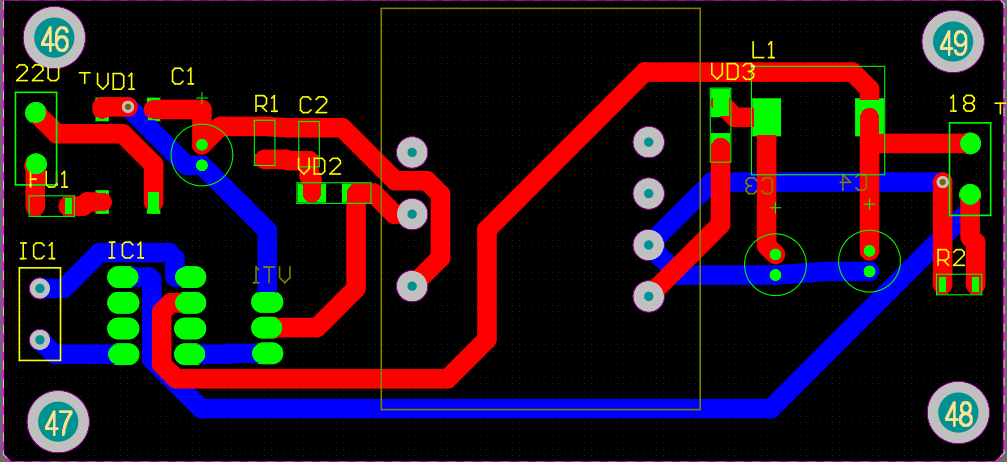


Рисунок 65 – Топология печатной платы

## 3 РАСЧЁТ ЗАТРАТ НА РАЗРАБОТКУ ЦИФРОВОЙ СИСТЕМЫ УПРАВЛЕНИЯ ИМПУЛЬСНЫМ ИСТОЧНИКОМ ПИТАНИЯ

При расчёте затрат на разработку цифровой системы управления импульсным источником питания необходимо учесть все работы, выполненные исполнителем ВКР и руководителем проекта, а также все необходимые материалы и ресурсы. Каждый вид работ характеризуется своей трудоёмкостью, то есть количеством затраченного времени на производство продукта.

Расчёт себестоимости осуществляется по следующим статьям:

- материалы;

- накладные расходы;

- основная и дополнительная заработная плата;

- отчисления на страховые взносы.

Основной составляющей расчёта себестоимости проекта является оплата труда. В основе расчёта заработной платы исполнителей проекта является определение трудоёмкости работ (таблица 1), связанных с разработкой выпускной квалификационной работы.

Таблица 1

|  |  |  |
| --- | --- | --- |
| Наименование работ | Трудоёмкость, чел./дни | |
|  | Руководитель | Инженер |
| Изучение литературы по теме ВКР | – | 7 |
| Разработка методов решения задачи | 4 | 14 |
| Анализ методов решения задачи | – | 3 |
| Создание моделей силовой части и цифровой системы управления ИИП | – | 10 |
| Тестирование моделей | – | 2 |
| Анализ результатов моделирования | 1 | 1 |
| Разработка программного обеспечения для микроконтроллера | – | 10 |
| Разработка печатной платы | – | 2 |
| Сдача проекта | 1 | 1 |
| Итого | 6 | 50 |

Основная и дополнительная заработная плата исполнителей разработки рассчитывается на основании следующих данных:

- трудоёмкость выполнения работ (таблица 1) Tрук = 6 чел/дней, Тдип= 58 чел/дней;

- дневная ставка научного руководителя инженера-конструктора отдела моделирования систем управления по составляет Друк = 6000 руб.;

- дневная ставка стажёра-инженера Ддип = 1500 руб.;

- процент дополнительной заработной платы – 12%;

- процент отчислений на страховые взносы – 30%;

- процент накладных расходов – 33%.

Основная заработная плата исполнителей рассчитывается по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (39) |

Дополнительная заработная плата составляет 12% от основной заработной платы:

|  |  |  |
| --- | --- | --- |
|  |  | (40) |

Cзд = 111000 13320 руб.

Отчисления на страховые взносы составляют 30% от основной и дополнительной заработной платы:

|  |  |  |
| --- | --- | --- |
|  |  | (41) |

Ссн = (111000+13320)37296 руб.

Накладные расходы рассчитываются по формуле:

|  |  |  |
| --- | --- | --- |
|  |  | (42) |

Снр= (111000+13320)41025,6 руб.

Расчёт количества и стоимости материалов представлены в таблице 2.

Таблица 2

|  |  |  |  |
| --- | --- | --- | --- |
| Материалы | Количество | Цена, руб. | Сумма, руб. |
| Бумага для оргтехники, пачка *Svetocopy* | 1 | 290 | 290 |
| Картридж для принтера *Samsung M2070* | 1 | 690 | 690 |
| Итого | | | 980 |

Расчёт себестоимости разработки проекта представлен в таблице 3.

Таблица 3

|  |  |
| --- | --- |
| Статья затрат | Сумма, руб. |
| Основная заработная плата | 111000 |
| Дополнительная заработная плата | 13320 |
| Отчисление на страховые взносы | 37296 |
| Накладные расходы | 41025,6 |
| Материалы | 980 |
| ИТОГО: | 223621,6 |

## 4 ТЕХНИКА БЕЗОПАСНОСТИ ПРИ МОНТАЖЕ И НАЛАДКЕ ИСТОЧНИКА ПИТАНИЯ

Работа с электричеством опасна тем, что оно не действует на органы чувств до момента соприкосновения с токоведущими проводниками и контактами. Это затрудняет дистанционное обнаружение опасности.

Электрическое напряжение свыше 40 В опасно для жизни. Степень поражения зависит от пути прохождения электрического тока через тело человека и от силы тока, особенно той его части, которая проходит через сердце. Наиболее опасны пути тока «рука-нога» и «рука-рука». Поэтому при настройке радиоаппаратуры и поиске неисправностей необходимо работать одной рукой – во избежание прикосновения к токоведущим частям обеими руками. Особую осторожность необходимо соблюдать, когда устройство питается от осветительной сети 220 В по бестрансформаторной схеме, с помощью импульсного преобразователя или через автотрансформатор. В этом случае выход даже низковольтного источника вторичного питания может оказаться под напряжением сети относительно «земли». Важно изолировать себя от «земли», чтобы исключить поражение электрическим током при случайном прикосновении к элементам устройства или его общей шине (общему проводу).

Монтажные работы следует производить вдали от заземляющих конструкций (водопроводных труб, радиаторов отопления) или при нить необходимые меры, чтобы исключить случайное прикосновение к ним.

Заменять вышедшие из строя или «подозрительные» детали следует только после отключения устройства от сети.

Нельзя проверять исправность предохранителей в устройстве, включенном в сеть, путем их замыкания.

Перед выполнением регулировочных операций под напряжением необходимо принять следующие меры предосторожности:

* установить разделительный трансформатор в цепи питания переменного тока;
* убедиться, что сетевое напряжение в пределах 230 В ± 10%, 50 Гц;
* перед подключением сетевой вилки убедиться, что кнопка включения устройства находится в положении «Выключено».

Категорически запрещается во время проведения регулировочных и ремонтных работ непосредственно на печатных или монтажных платах «разрывать» или «закорачивать» какие-либо элементы в электрической цепи при включенном в сеть источнике питания.

# ЗАКЛЮЧЕНИЕ

Современная микропроцессорная техника позволяет реализовать комбинированное управление импульсными источниками питания, что позволяет минимизировать размеры печатных плат за счёт уменьшения количества элементов электрической цепи.

В выпускной квалификационной работе наглядно показана возможность реализации микропроцессорного управления импульсными источниками питания. Полученные имитационные модели замкнутой системы управления позволяют сделать вывод о том, что с помощью микроконтроллеров можно регулировать в импульсных источниках одновременно и ток, и напряжение, значительно уменьшив количество элементов электрической цепи. При этом полученная система управления является более гибкой в настройке, по сравнению с системами, где используются ШИМ-контроллеры, за счёт возможности программной перенастройки параметров.

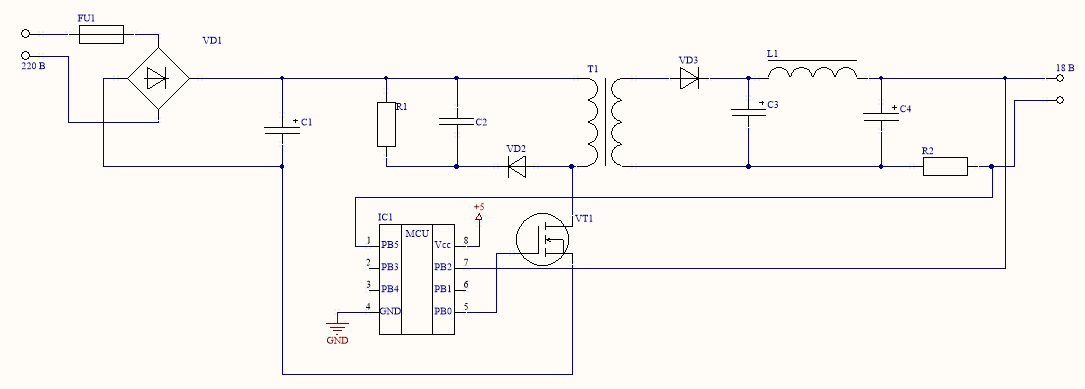
# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Браун М. Источники питания. Расчёт и конструирование / М. Браун. Пер. с англ. – К.: МК – Пресс, 2007. – 288 с
2. Семёнов Б.Ю. Экономичное освещение для всех / Б.Ю. Семёнов. – М.: СОЛОН –Пресс, 2010. – 224 с.
3. Стабилизатор тока светодиода [Электронный ресурс] – режим доступа: http://leddisplays.ru/stabilizator\_toka.html, свободный, (дата обращения 01.04.2020) – Загл. с экрана.
4. Интегральные микросхемы: Микросхемы для импульсных источников питания и их применение. – М.: Додэка, 1997. – c. 15-16. – 224 с.
5. Махлин А. Особенности проектирования блока питания для светодиодных ламп // Полупроводниковая светотехника. – 2011. – № 1. – С. 30–33.
6. Михальченко Г.Я. Математические модели импульсных систем преобразования энергии / Г.Я. Михальченко, С.Г. Михальченко. – Томск: Изд-во Том. гос. ун-та систем упр. и радиоэлектроники, 2013. – 160 с.
7. Диксон Р.К. Вопросы линеаризации математической модели преобразователя в системах электропитания, работающих на основе возобновляемых источников энергии // Изв. Том. политехн. ун-та. Инжиниринг георесурсов / Национ. исследоват. Том. политехн. ун-т. – 2017. –Т. 328, № 1. – С. 89–99.
8. Лопаткин А. Проектирование печатных плат в системе Altium Designer: учеб. Пособие для практических занятий. 2-е изд., перераб. и доп. –М.: ДМК Пресс, 2017. – 554 с.
9. Smart control of road-based LED fixtures for energy saving / A. Lay-Ekuakille, F. D’Aniello, F. Miduri, D. Leonardi, A. Trotta // Proc. IEEE Intell. Data Acquisition Adv. Comput. Syst. – Lecce, 2009. – P. 59–62.
10. Исследование и разработка элементов и узлов системы управления светодиодным источником света с улучшенными техническими и эксплуатационными характеристиками / А.В. Иванов, А.В. Фёдоров, Т.Н. Зайченко, И.В. Целебровский // Доклады Томского университета систем управления и радиоэлектроники. – 2011. – № 2 (24). – Ч. 3. – С. 71–77.
11. Patel U. Focus LED Design on Driver Topology, Circuit Protection. Littelfuse’s LED Lighting Design Guide, 2013, pp. 32–36.
12. Бастион. Преобразователи напряжения [Электронный ресурс] – Режим доступа: https://bast.ru/products/converters/pn-20-75dc12-15-isp5, свободный (дата посещения: 15.03.2020). – Загл. с экрана.
13. Стабилизаторы тока [Электронный ресурс] – Режим доступа: https://digteh.ru/BP/Stabilizat/I/, свободный (дата посещения: 4.03.2020). – Загл. с экрана.
14. Гончаров А. К вопросу энергоэффективности и энергосбережения в освещении / А. Гончаров, И. Денисов, И. Козырева и др. / Полупроводниковая светотехника. – 2011. – № 4. – C. 5–9.
15. Мелешин В.И. Транзисторная преобразовательная техника. – М.: Техносфера, 2006. – 631 с.

# ПРИЛОЖЕНИЕ А

(обязательное)

Схема электрическая принципиальная импульсного источника питания



# ПРИЛОЖЕНИЕ Б

*Примечание*

*Наименование*

Поз.  
обозн.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

1

*52.14.1.6.20.БР.013.00.ПЭ*

Разраб.

*Ямутеев М.В.*

Провер.

*Сковпень С.М.*

Реценз.

Н. Контр.

Сковпень С.М.

Утверд.

Музыка М.М.

*Импульсный источник питания*

Лит.

Листов

1

ИСМАрТ

Кол.

*Диоды*

*DB104S*

*VD1*

1

-

*S1J*

*VD2*

1

*SKL18*

*VD3*

1

*Конденсаторы*

*К50-35-400 В-10 мкФ ±5%*

*C1*

*1*

*SMD 1206-250 В-4700 пФ ±5%*

C2

*1*

*К50-35-25 В-47 мкФ ±5%*

*C3*

*1*

*К50-35-25 В-680 мкФ ±5%*

*C4*

1

*Резисторы*

*SMD 1206-6,6 кОм ±5%*

*R1*

1

*SMD 1206-1 Ом ±5%*

*R2*

*1*

Предохранитель SMD 1206-2A

*FU1*

1

12

*Транзистор IRFBC40*

*VT1*

*1*

*Трансформатор ETD39-0391*

*T1*

*1*

*Интегральная схема ATtiny13*

*IC1*

*1*

# ПРИЛОЖЕНИЕ В

(Обязательное)

Листинг программы цифровой системы управления на языке С

#define MX\_AVR

#define MX\_USES\_UINT8 1

#define MX\_USES\_SINT16 0

#define MX\_USES\_CHAR 0

#define MX\_USES\_FLOAT 1

#define MX\_USES\_SINT32 0

#define MX\_USES\_BOOL 1

#define MX\_USES\_UINT16 0

#define MX\_USES\_UINT32 0

//Defines for microcontroller

#define FC\_CAL\_AVR

#define MX\_TINY

#define MX\_ADC

#define MX\_ADC\_TYPE\_5

#define MX\_ADC\_BITS\_10

#define MX\_EE

#define MX\_EE\_SIZE 64

#define MX\_PWM

#define MX\_PWMTYPE 1

#define MX\_PWM\_CNT 2

#define MX\_PWM\_TMR0\_CNT 2

#define MX\_PWM\_PSCA1

#define MX\_PWM\_PSCA8

#define MX\_PWM\_PSCA64

#define MX\_PWM\_PSCA256

#define MX\_PWM\_PSCA1024

#define MX\_PWM\_1\_PORT PORTB

#define MX\_PWM\_1\_TRIS DDRB

#define MX\_PWM\_1\_PIN 0

#define MX\_PWM\_2\_PORT PORTB

#define MX\_PWM\_2\_TRIS DDRB

#define MX\_PWM\_2\_PIN 1

//Functions

#define F\_CPU 20000000UL

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

#include <avr\io.h>

#include <avr\interrupt.h>

#include <avr\eeprom.h>

#include <avr\wdt.h>

//Configuration data

#pragma DATA 0x0, 0xdf

#pragma DATA 0x1, 0xff

//?????????? ???????

#include "C:\Program Files (x86)\Flowcode(AVR)\v5\FCD\internals.c"

//Macro function declarations

//Variable declarations

#define FCV\_FALSE (0)

#define FCV\_TRUE (1)

volatile MX\_UINT8 FCV\_URR;

volatile MX\_FLOAT FCV\_D;

volatile MX\_FLOAT FCV\_UST;

volatile MX\_FLOAT FCV\_IK;

volatile MX\_UINT8 FCV\_PWM;

volatile MX\_FLOAT FCV\_ISUM;

volatile MX\_FLOAT FCV\_IREAL;

volatile MX\_FLOAT FCV\_ERROR;

volatile MX\_FLOAT FCV\_UI;

volatile MX\_FLOAT FCV\_I;

volatile MX\_UINT8 FCV\_LED;

volatile MX\_FLOAT FCV\_K;

volatile MX\_FLOAT FCV\_DK;

volatile MX\_FLOAT FCV\_ISET;

volatile MX\_FLOAT FCV\_U\_OLD;

volatile MX\_FLOAT FCV\_P;

volatile MX\_FLOAT FCV\_PK;

volatile MX\_UINT8 FCV\_USET;

volatile MX\_FLOAT FCV\_UR;

volatile MX\_FLOAT FCV\_MAX;

volatile MX\_FLOAT FCV\_I\_OLD;

//PWM(0): //Defines:

/\*\*\*\* Macro Substitutions \*\*\*\*

a = Unique Reference

b = PWM Channel - 0=SW / 1-8=HW

c = PWM Alt Pin FCD\_PWM0\_Enable - 0=Standard / 1=Alt

d = PWM Period

e = PWM Prescaler 1

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//Period and Prescaler Defines

#ifndef MX\_PWM\_PERIOD

#define MX\_PWM\_PERIOD 99

#else

#if (MX\_PWM\_PERIOD != 99)

#error "PWM period value must be the same in all PWM modules"

#endif

#endif

#ifndef MX\_PWM\_PRESCALE1

#define MX\_PWM\_PRESCALE1 1

#else

#if (MX\_PWM\_PRESCALE1 != 1)

#error "PWM prescale value must be the same in all PWM modules"

#endif

#endif

//Definitions for PWM slot allocation

#ifndef MX\_PWM\_REF1

#define MX\_PWM\_REF1

#define PWM\_1\_MX\_PWM\_UREF 1

#define MX\_PWM\_CHANNEL\_1 1

#define MX\_PWM\_HWALT\_1 0

#else

#ifndef MX\_PWM\_REF2

#define MX\_PWM\_REF2

#define PWM\_1\_MX\_PWM\_UREF 2

#define MX\_PWM\_CHANNEL\_2 1

#define MX\_PWM\_HWALT\_2 0

#else

#ifndef MX\_PWM\_REF3

#define MX\_PWM\_REF3

#define PWM\_1\_MX\_PWM\_UREF 3

#define MX\_PWM\_CHANNEL\_3 1

#define MX\_PWM\_HWALT\_3 0

#else

#ifndef MX\_PWM\_REF4

#define MX\_PWM\_REF4

#define PWM\_1\_MX\_PWM\_UREF 4

#define MX\_PWM\_CHANNEL\_4 1

#define MX\_PWM\_HWALT\_4 0

#else

#ifndef MX\_PWM\_REF5

#define MX\_PWM\_REF5

#define PWM\_1\_MX\_PWM\_UREF 5

#define MX\_PWM\_CHANNEL\_5 1

#define MX\_PWM\_HWALT\_5 0

#else

#ifndef MX\_PWM\_REF6

#define MX\_PWM\_REF6

#define PWM\_1\_MX\_PWM\_UREF 6

#define MX\_PWM\_CHANNEL\_6 1

#define MX\_PWM\_HWALT\_6 0

#else

#ifndef MX\_PWM\_REF7

#define MX\_PWM\_REF7

#define PWM\_1\_MX\_PWM\_UREF 7

#define MX\_PWM\_CHANNEL\_7 1

#define MX\_PWM\_HWALT\_7 0

#else

#ifndef MX\_PWM\_REF8

#define MX\_PWM\_REF8

#define PWM\_1\_MX\_PWM\_UREF 8

#define MX\_PWM\_CHANNEL\_8 1

#define MX\_PWM\_HWALT\_8 0

#else

#ifndef MX\_PWM\_REF9

#define MX\_PWM\_REF9

#define PWM\_1\_MX\_PWM\_UREF 9

#define MX\_PWM\_CHANNEL\_9 1

#define MX\_PWM\_HWALT\_9 0

#else

#ifndef MX\_PWM\_REF10

#define MX\_PWM\_REF10

#define PWM\_1\_MX\_PWM\_UREF 10

#define MX\_PWM\_CHANNEL\_10 1

#define MX\_PWM\_HWALT\_10 0

#endif

#endif

#endif

#endif

#endif

#endif

#endif

#endif

#endif

#endif

#define PWM\_1\_PWM\_Enable\_Channel CAL\_APPEND(FC\_CAL\_PWM\_Enable\_Channel\_, PWM\_1\_MX\_PWM\_UREF)

#define PWM\_1\_PWM\_Disable\_Channel CAL\_APPEND(FC\_CAL\_PWM\_Disable\_Channel\_, PWM\_1\_MX\_PWM\_UREF)

#define PWM\_1\_PWM\_Set\_Duty\_8Bit CAL\_APPEND(FC\_CAL\_PWM\_Set\_Duty\_8Bit\_, PWM\_1\_MX\_PWM\_UREF)

#define PWM\_1\_PWM\_Change\_Period CAL\_APPEND(FC\_CAL\_PWM\_Change\_Period\_, PWM\_1\_MX\_PWM\_UREF)

#define PWM\_1\_PWM\_Set\_Duty\_10Bit CAL\_APPEND(FC\_CAL\_PWM\_Set\_Duty\_10Bit\_, PWM\_1\_MX\_PWM\_UREF)

extern void PWM\_1\_PWM\_Enable\_Channel ();

extern void PWM\_1\_PWM\_Disable\_Channel ();

extern void PWM\_1\_PWM\_Set\_Duty\_8Bit (MX\_UINT8 duty);

extern void PWM\_1\_PWM\_Change\_Period (MX\_UINT8 period, MX\_UINT16 prescaler);

extern void PWM\_1\_PWM\_Set\_Duty\_10Bit (MX\_UINT16 duty);

//PWM(0): //Macro function declarations

static void FCD\_PWM0\_Enable();

static void FCD\_PWM0\_Disable();

static void FCD\_PWM0\_SetDutyCycle(MX\_UINT8 nDuty);

static void FCD\_PWM0\_ChangePeriod(MX\_UINT8 nPeriodVal, MX\_UINT8 nPrescalerVal);

static void FCD\_PWM0\_SetDutyCycle10bit(MX\_SINT16 nDuty);

//ADC(0): //Defines:

/\*\*\*\* Macro Substitutions \*\*\*\*

a = Unique Reference

b = Which ADC Channel

c = Acquisition time

d = Conversion Speed

e = VRef+ Option

f = VRef Voltage x 0.01V

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//Common Defines

#define ADC\_2\_MX\_ADC\_CHANNEL 0

#define ADC\_2\_MX\_ADC\_ACTIME 40

#define ADC\_2\_MX\_ADC\_CONVSP 3

#define ADC\_2\_MX\_ADC\_VREFOP 0

#define ADC\_2\_MX\_ADC\_VREFVOL 500

#ifndef MX\_ADC\_CHANNEL\_0

#define MX\_ADC\_CHANNEL\_0 //Inform CAL ADC channel 0 is now in use.

#endif

#ifndef MX\_ADC\_REF //Inform CAL ADC peripheral is now in use

#define MX\_ADC\_REF

#endif

extern void FC\_CAL\_Enable\_ADC (MX\_UINT8 Channel, MX\_UINT8 Conv\_Speed, MX\_UINT8 Vref, MX\_UINT8 T\_Charge);

extern MX\_UINT16 FC\_CAL\_Sample\_ADC (MX\_UINT8 Sample\_Mode);

extern void FC\_CAL\_Disable\_ADC (void);

//ADC(0): //Macro function declarations

static void FCD\_ADC0\_SampleADC();

static MX\_UINT8 FCD\_ADC0\_ReadAsByte();

static MX\_UINT16 FCD\_ADC0\_ReadAsInt();

static MX\_FLOAT FCD\_ADC0\_ReadAsVoltage();

static void FCD\_ADC0\_ReadAsString(MX\_CHAR\* FCR\_RETVAL, MX\_UINT8 FCR\_RETVAL\_SIZE);

static void FCD\_ADC0\_ADC\_RAW\_Configure\_Channel();

static MX\_UINT8 FCD\_ADC0\_ADC\_RAW\_Sample\_Channel\_Byte();

static MX\_UINT16 FCD\_ADC0\_ADC\_RAW\_Sample\_Channel\_Int();

static MX\_UINT8 FCD\_ADC0\_ADC\_RAW\_Average\_Channel\_Byte(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs);

static MX\_UINT16 FCD\_ADC0\_ADC\_RAW\_Average\_Channel\_Int(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs);

static void FCD\_ADC0\_ADC\_RAW\_Disable\_Channel();

//ADC(1): //Defines:

/\*\*\*\* Macro Substitutions \*\*\*\*

a = Unique Reference

b = Which ADC Channel

c = Acquisition time

d = Conversion Speed

e = VRef+ Option

f = VRef Voltage x 0.01V

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//Common Defines

#define ADC\_3\_MX\_ADC\_CHANNEL 0

#error "Error: ADC pin not connected"

#define ADC\_3\_MX\_ADC\_ACTIME 40

#define ADC\_3\_MX\_ADC\_CONVSP 3

#define ADC\_3\_MX\_ADC\_VREFOP 0

#define ADC\_3\_MX\_ADC\_VREFVOL 500

#ifndef MX\_ADC\_CHANNEL\_0

#error "Error: ADC pin not connected"

#define MX\_ADC\_CHANNEL\_0

#error "Error: ADC pin not connected"

//Inform CAL ADC channel 0

#error "Error: ADC pin not connected"

is now in use.

#endif

#ifndef MX\_ADC\_REF //Inform CAL ADC peripheral is now in use

#define MX\_ADC\_REF

#endif

extern void FC\_CAL\_Enable\_ADC (MX\_UINT8 Channel, MX\_UINT8 Conv\_Speed, MX\_UINT8 Vref, MX\_UINT8 T\_Charge);

extern MX\_UINT16 FC\_CAL\_Sample\_ADC (MX\_UINT8 Sample\_Mode);

extern void FC\_CAL\_Disable\_ADC (void);

//ADC(1): //Macro function declarations

static void FCD\_ADC1\_SampleADC();

static MX\_UINT8 FCD\_ADC1\_ReadAsByte();

static MX\_UINT16 FCD\_ADC1\_ReadAsInt();

static MX\_FLOAT FCD\_ADC1\_ReadAsVoltage();

static void FCD\_ADC1\_ReadAsString(MX\_CHAR\* FCR\_RETVAL, MX\_UINT8 FCR\_RETVAL\_SIZE);

static void FCD\_ADC1\_ADC\_RAW\_Configure\_Channel();

static MX\_UINT8 FCD\_ADC1\_ADC\_RAW\_Sample\_Channel\_Byte();

static MX\_UINT16 FCD\_ADC1\_ADC\_RAW\_Sample\_Channel\_Int();

static MX\_UINT8 FCD\_ADC1\_ADC\_RAW\_Average\_Channel\_Byte(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs);

static MX\_UINT16 FCD\_ADC1\_ADC\_RAW\_Average\_Channel\_Int(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs);

static void FCD\_ADC1\_ADC\_RAW\_Disable\_Channel();

//PWM(0): //Macro implementations

static void FCD\_PWM0\_Enable()

{

PWM\_1\_PWM\_Change\_Period (MX\_PWM\_PERIOD, MX\_PWM\_PRESCALE1);

PWM\_1\_PWM\_Enable\_Channel ();

}

static void FCD\_PWM0\_Disable()

{

PWM\_1\_PWM\_Disable\_Channel();

}

static void FCD\_PWM0\_SetDutyCycle(MX\_UINT8 nDuty)

{

PWM\_1\_PWM\_Set\_Duty\_8Bit(nDuty);

}

static void FCD\_PWM0\_ChangePeriod(MX\_UINT8 nPeriodVal, MX\_UINT8 nPrescalerVal)

{

PWM\_1\_PWM\_Change\_Period (nPeriodVal, nPrescalerVal);

}

static void FCD\_PWM0\_SetDutyCycle10bit(MX\_SINT16 nDuty)

{

PWM\_1\_PWM\_Set\_Duty\_10Bit(nDuty);

}

//ADC(0): //Macro implementations

static void FCD\_ADC0\_SampleADC()

{

//unused

}

static MX\_UINT8 FCD\_ADC0\_ReadAsByte()

{

MX\_UINT8 retVal;

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_2\_MX\_ADC\_CHANNEL , ADC\_2\_MX\_ADC\_CONVSP , ADC\_2\_MX\_ADC\_VREFOP , ADC\_2\_MX\_ADC\_ACTIME );

retVal = FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte

FC\_CAL\_Disable\_ADC ();

return (retVal);

}

static MX\_UINT16 FCD\_ADC0\_ReadAsInt()

{

MX\_UINT16 retVal;

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_2\_MX\_ADC\_CHANNEL , ADC\_2\_MX\_ADC\_CONVSP , ADC\_2\_MX\_ADC\_VREFOP , ADC\_2\_MX\_ADC\_ACTIME );

retVal = FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16

FC\_CAL\_Disable\_ADC ();

return (retVal);

}

static MX\_FLOAT FCD\_ADC0\_ReadAsVoltage()

{

MX\_UINT16 iSample;

MX\_FLOAT fSample, fVoltage, fVperDiv;

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_2\_MX\_ADC\_CHANNEL , ADC\_2\_MX\_ADC\_CONVSP , ADC\_2\_MX\_ADC\_VREFOP , ADC\_2\_MX\_ADC\_ACTIME );

#ifdef MX\_ADC\_BITS\_8

iSample = FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte

#else

iSample = FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16

#endif

FC\_CAL\_Disable\_ADC (); //Switch off ADC peripheral

fVoltage = flt\_fromi( ADC\_2\_MX\_ADC\_VREFVOL ); //Convert reference voltage count to floating point (0 - 500 x 10mV)

fVoltage = flt\_mul(fVoltage, 0.01); //Convert reference voltage count to actual voltage (0 - 5)

#ifdef MX\_ADC\_BITS\_8

fVperDiv = flt\_mul(fVoltage, 0.00390625); //Convert actual voltage to voltage per division (VRef / 256)

#endif

#ifdef MX\_ADC\_BITS\_10

fVperDiv = flt\_mul(fVoltage, 0.000976); //Convert actual voltage to voltage per division (VRef / 1024)

#endif

#ifdef MX\_ADC\_BITS\_12

fVperDiv = flt\_mul(fVoltage, 0.00024414); //Convert actual voltage to voltage per division (VRef / 4096)

#endif

fSample = flt\_fromi(iSample); //Convert to floating point variable

fVoltage = flt\_mul(fSample, fVperDiv); //Calculate floating point voltage

return (fVoltage);

}

static void FCD\_ADC0\_ReadAsString(MX\_CHAR\* FCR\_RETVAL, MX\_UINT8 FCR\_RETVAL\_SIZE)

{

MX\_FLOAT fVoltage;

fVoltage = FCD\_ADC0\_ReadAsVoltage();

FCI\_FLOAT\_TO\_STRING(fVoltage, 2, FCR\_RETVAL, FCR\_RETVAL\_SIZE); //Convert to String

}

static void FCD\_ADC0\_ADC\_RAW\_Configure\_Channel()

{

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_2\_MX\_ADC\_CHANNEL , ADC\_2\_MX\_ADC\_CONVSP , ADC\_2\_MX\_ADC\_VREFOP , ADC\_2\_MX\_ADC\_ACTIME );

}

static MX\_UINT8 FCD\_ADC0\_ADC\_RAW\_Sample\_Channel\_Byte()

{

return FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte

}

static MX\_UINT16 FCD\_ADC0\_ADC\_RAW\_Sample\_Channel\_Int()

{

return FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16

}

static MX\_UINT8 FCD\_ADC0\_ADC\_RAW\_Average\_Channel\_Byte(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs)

{

MX\_UINT32 average = 0;

MX\_UINT8 count;

for (count=0; count<NumSamples; count++)

{

average = average + FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte - add to average

if (DelayUs)

delay\_us(DelayUs); //If delay is not 0 then pause between samples

}

average = average / count;

return (average & 0xFF); //Return average as byte

}

static MX\_UINT16 FCD\_ADC0\_ADC\_RAW\_Average\_Channel\_Int(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs)

{

MX\_UINT32 average = 0;

MX\_UINT8 count;

for (count=0; count<NumSamples; count++)

{

average = average + FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16 - add to average

if (DelayUs)

delay\_us(DelayUs); //If delay is not 0 then pause between samples

}

average = average / count;

return (average & 0x1FFF); //Return average as MX\_SINT16

}

static void FCD\_ADC0\_ADC\_RAW\_Disable\_Channel()

{

FC\_CAL\_Disable\_ADC (); //Disable ADC Channel

}

//ADC(1): //Macro implementations

static void FCD\_ADC1\_SampleADC()

{

//unused

}

static MX\_UINT8 FCD\_ADC1\_ReadAsByte()

{

MX\_UINT8 retVal;

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_3\_MX\_ADC\_CHANNEL , ADC\_3\_MX\_ADC\_CONVSP , ADC\_3\_MX\_ADC\_VREFOP , ADC\_3\_MX\_ADC\_ACTIME );

retVal = FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte

FC\_CAL\_Disable\_ADC ();

return (retVal);

}

static MX\_UINT16 FCD\_ADC1\_ReadAsInt()

{

MX\_UINT16 retVal;

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_3\_MX\_ADC\_CHANNEL , ADC\_3\_MX\_ADC\_CONVSP , ADC\_3\_MX\_ADC\_VREFOP , ADC\_3\_MX\_ADC\_ACTIME );

retVal = FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16

FC\_CAL\_Disable\_ADC ();

return (retVal);

}

static MX\_FLOAT FCD\_ADC1\_ReadAsVoltage()

{

MX\_UINT16 iSample;

MX\_FLOAT fSample, fVoltage, fVperDiv;

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_3\_MX\_ADC\_CHANNEL , ADC\_3\_MX\_ADC\_CONVSP , ADC\_3\_MX\_ADC\_VREFOP , ADC\_3\_MX\_ADC\_ACTIME );

#ifdef MX\_ADC\_BITS\_8

iSample = FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte

#else

iSample = FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16

#endif

FC\_CAL\_Disable\_ADC (); //Switch off ADC peripheral

fVoltage = flt\_fromi( ADC\_3\_MX\_ADC\_VREFVOL ); //Convert reference voltage count to floating point (0 - 500 x 10mV)

fVoltage = flt\_mul(fVoltage, 0.01); //Convert reference voltage count to actual voltage (0 - 5)

#ifdef MX\_ADC\_BITS\_8

fVperDiv = flt\_mul(fVoltage, 0.00390625); //Convert actual voltage to voltage per division (VRef / 256)

#endif

#ifdef MX\_ADC\_BITS\_10

fVperDiv = flt\_mul(fVoltage, 0.000976); //Convert actual voltage to voltage per division (VRef / 1024)

#endif

#ifdef MX\_ADC\_BITS\_12

fVperDiv = flt\_mul(fVoltage, 0.00024414); //Convert actual voltage to voltage per division (VRef / 4096)

#endif

fSample = flt\_fromi(iSample); //Convert to floating point variable

fVoltage = flt\_mul(fSample, fVperDiv); //Calculate floating point voltage

return (fVoltage);

}

static void FCD\_ADC1\_ReadAsString(MX\_CHAR\* FCR\_RETVAL, MX\_UINT8 FCR\_RETVAL\_SIZE)

{

MX\_FLOAT fVoltage;

fVoltage = FCD\_ADC1\_ReadAsVoltage();

FCI\_FLOAT\_TO\_STRING(fVoltage, 2, FCR\_RETVAL, FCR\_RETVAL\_SIZE); //Convert to String

}

static void FCD\_ADC1\_ADC\_RAW\_Configure\_Channel()

{

//Configure & Enable ADC Channel

FC\_CAL\_Enable\_ADC ( ADC\_3\_MX\_ADC\_CHANNEL , ADC\_3\_MX\_ADC\_CONVSP , ADC\_3\_MX\_ADC\_VREFOP , ADC\_3\_MX\_ADC\_ACTIME );

}

static MX\_UINT8 FCD\_ADC1\_ADC\_RAW\_Sample\_Channel\_Byte()

{

return FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte

}

static MX\_UINT16 FCD\_ADC1\_ADC\_RAW\_Sample\_Channel\_Int()

{

return FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16

}

static MX\_UINT8 FCD\_ADC1\_ADC\_RAW\_Average\_Channel\_Byte(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs)

{

MX\_UINT32 average = 0;

MX\_UINT8 count;

for (count=0; count<NumSamples; count++)

{

average = average + FC\_CAL\_Sample\_ADC( 0 ); //Perform Sample - Return as byte - add to average

if (DelayUs)

delay\_us(DelayUs); //If delay is not 0 then pause between samples

}

average = average / count;

return (average & 0xFF); //Return average as byte

}

static MX\_UINT16 FCD\_ADC1\_ADC\_RAW\_Average\_Channel\_Int(MX\_UINT8 NumSamples, MX\_UINT8 DelayUs)

{

MX\_UINT32 average = 0;

MX\_UINT8 count;

for (count=0; count<NumSamples; count++)

{

average = average + FC\_CAL\_Sample\_ADC( 1 ); //Perform Sample - Return as MX\_UINT16 - add to average

if (DelayUs)

delay\_us(DelayUs); //If delay is not 0 then pause between samples

}

average = average / count;

return (average & 0x1FFF); //Return average as MX\_SINT16

}

static void FCD\_ADC1\_ADC\_RAW\_Disable\_Channel()

{

FC\_CAL\_Disable\_ADC (); //Disable ADC Channel

}

#include "C:\Program Files (x86)\Flowcode(AVR)\v5\CAL\includes.c"

//Macro implementations

int main()

{

//Initialization

MCUSR=0x00;

wdt\_disable();

//Interrupt initialization code

//100khz

//?????? ??????????: Enable()

FCD\_PWM0\_Enable();

//??????????:

// Iset = 0.5

FCV\_ISET = 0.5;

//??????????:

// Ust = 18

FCV\_UST = 18;

//??????????:

// Pk = 0.0025

// Dk = 0

// Ik = 8

FCV\_PK = 0.0025;

FCV\_DK = 0;

FCV\_IK = 8;

//????: While 1

while (1)

{

//?????? ??????????: Ur=?????? ?a? ??????????()

FCV\_UR = FCD\_ADC0\_ReadAsVoltage();

//?????? ??????????: Ui=?????? ?a? ??????????()

FCV\_UI = FCD\_ADC1\_ReadAsVoltage();

//??????????:

// Ireal = Ui / 1 //1-????????????? ?????

FCV\_IREAL = flt\_div(FCV\_UI, 1);

//???????: Ireal > Iset?

if (flt\_gt(FCV\_IREAL, FCV\_ISET))

{

//??????????:

// PWM = 0

FCV\_PWM = 0;

} else {

//??????????:

// error = Ust - Ur

// D = Dk \* (U\_old - Ur)

// U\_old = Ur

// Ik = Ik + error

FCV\_ERROR = flt\_sub(FCV\_UST, FCV\_UR);

FCV\_D = flt\_mul(FCV\_DK, (flt\_sub(FCV\_U\_OLD, FCV\_UR)));

FCV\_U\_OLD = FCV\_UR;

FCV\_IK = flt\_add(FCV\_IK, FCV\_ERROR);

//??????????:

// I = Ik

// K = Pk \* error + D + I

FCV\_I = FCV\_IK;

FCV\_K = flt\_add(flt\_add(flt\_mul(FCV\_PK, FCV\_ERROR), FCV\_D), FCV\_I);

//???????

//???????: K > 99?

if (flt\_gt(FCV\_K, 99))

{

//??????????:

// PWM = 99

FCV\_PWM = 99;

} else {

//???????: K < 0?

if (flt\_lt(FCV\_K, 0))

{

//??????????:

// PWM = 0

FCV\_PWM = 0;

} else {

//??????????:

// PWM = K

FCV\_PWM = flt\_toi(FCV\_K);

}

}

}

//?????? ??????????: SetDutyCycle(PWM)

FCD\_PWM0\_SetDutyCycle(FCV\_PWM);

//????????: 10 us

delay\_us(10);

}

mainendloop: goto mainendloop;

}