

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет

информационных технологий
и электроники

наименование факультета

Кафедра

Нано – и микроэлектроника
наименование кафедры

Направление подготовки

11.03.04. «Электроника и наноэлектроника».
код и наименование

Профиль

Микроэлектроника и твердотельная электроника
наименование

БАКАЛАВРСКАЯ РАБОТА

ПензГУ 02. 11.03.04. 006 ПЗ

обозначение, присваиваемое кафедрой

на тему

Разработка программно-аппаратной части лабораторного стенда по
вакуумной электронике
тема бакалаврской работы

Студент группы 19ПД1

номер группы

подпись, дата

13.06.23 Д.В. Зубанов

инициалы, фамилия

Руководитель доцент каф. НиМЭ

должность

подпись, дата

15.06.23 А.А. Вольсков

инициалы, фамилия

ПГУ

наименование организации

Нормоконтролер ст. преподаватель

должность

подпись, дата

16.06.23 Т.Н. Рыжова

инициалы, фамилия

Работа допущена к защите в ГЭК (протокол заседания кафедры от 15.06.23
№ 11)

Заведующий кафедрой

подпись, дата

13.06.23 И.А. Пронин

инициалы, фамилия

Работа защищена с оценкой хор. (протокол ГЭК от 28.06.23 № 11)

Секретарь ГЭК

подпись, дата

28.06.23 А.А. Вольсков

инициалы, фамилия

РЕФЕРАТ

Пояснительная записка состоит из 81 страницы, 20 рисунков, списка использованных источников из 10 наименований и 6 приложений.

Ключевые слова: микроконтроллер, вакуумная лампа, аналого-цифровой преобразователь, широтно-импульсная модуляция, среда программирования.

Цель: Разработать программно-аппаратную часть лабораторного стенда по вакуумной электронике

Задачи:

1. Выбрать программно-аппаратные средства для реализации проекта.
2. Разработать алгоритм приёма данных со стенда.
3. Разработать алгоритм управления напряжением на сетке.
4. Разработать алгоритм управления стендом.
5. Сделать полное описание готовой программы.

Актуальность данной темы заключается в том, что Электровacuумные приборы находят незаменимое применение в спутниковых передатчиках, в самолетах, на кораблях и в центрах связи на Земле.

Объект исследования: вакуумные лампы.

Предмет исследования: параметры вакуумных ламп.

Полученные результаты: Разработана программно-аппаратная часть стенда по вакуумной электронике для исследования параметров и характеристик вакуумных ламп.

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	докум.	Подпись	аДа		

СОДЕРЖАНИЕ

Введение	5
1. Выбор программно-аппаратных средств для реализации проекта	6
1.1 Выбор микроконтроллера	6
1.2 Выбор программного обеспечения для создания приложения	14
1.3 Выбор программы для прошивки микроконтроллера	18
2. Разработка алгоритмов	22
2.1 Разработка алгоритма приёма данных со стенда	22
2.2 Разработка алгоритма управления напряжением на сетке при помощи ШИМ сигнала	25
2.3 Разработка алгоритма управления стендом	26
2.4 Алгоритм обработки результатов эксперимента	27
3 Изготовление макета учебного стенда	28
3.1 Изготовление аппаратной части учебного стенда	28
3.2 Описание программной части стенда	28
3.3 Проверка работоспособности стенда	29
4. Обеспечение безопасных условий труда при разработке учебного стенда	33
Заключение	36
Список использованных источников	37
Приложение А. Техническое задание на бакалаврскую работу	38
Приложение Б. Алгоритм работы стенда	41
Приложение В. Код программы для программирования микроконтроллера	49
Приложение Г. Код приложения для учебного стенда	54
Приложение Д. Заявление и протокол проверки ВКР на оригинальность в системе «Антиплагиат. ВУЗ»	65
Приложение Е. Презентация бакалаврской работы	68

ВВЕДЕНИЕ

Электронная лампа, радиолампа представляет собой электровакуумный прибор (точнее, вакуумный электронный прибор), работающий за счёт управления интенсивностью потока электронов, движущихся в вакууме или разрежённом газе между электродами.

Лампы используются для получения высококачественный звук аппаратуры класса *Hi-End*, где все строится принципиально только на радиолампах. Многие зарубежные производители усилителей используют некоторые типы ламп исключительно производства России.

Электровакуумные приборы находят незаменимое применение в спутниковых передатчиках, в самолетах, на кораблях и в центрах связи на Земле. Только электровакуумные приборы способны обеспечить сверхвысокие частоты с высокой стабильностью и огромными мощностями, транзисторы этого не умеют [1].

Данная дипломная работа заключается в разработке программно-аппаратного устройства для стенда по вакуумной электронике. Стенд предназначен для получения и анализа характеристик вакуумных ламп. Схемотехническую часть стенда разрабатывает студент группы 21ПДм1 Янкин В.А. Алгоритм находится в приложении Б.

Программно-аппаратная часть должна управлять напряжением на сетке, снимать значения анодных напряжений и токов. Так же программно-аппаратная часть должна строить зависимости анодного напряжения от тока, анодного напряжения от напряжения на сетке и рассчитывать крутизну и другие дополнительные параметры.

Для реализации управления необходимо использовать микроконтроллер с возможностью генерации ШИМ-сигнала. Так же микроконтроллер необходим для измерения значений напряжения. Для построения графиков необходимо приложение, для разработки которого необходима программная среда разработки.

					ПГУ.2.11.03.04.006ПЗ		
Изм.	Лист	№ докум.	Подпись	Дат			
Разраб.		Зубанов Д.В.			Введение	Лит.	Лист
Провер.		Вольсков А.А.					Листов
							4
							80
Н. Контр.		Рыжова Т.Н.		19.06	группа 19ПД		
Утверд.							

1 Выбор аппаратных средств для реализации проекта

1.1 Выбор микроконтроллера

1.1.1 Требования к микроконтроллеру.

Проанализировав алгоритм удалось вывести следующие минимальные требования к микроконтроллеру:

- 1 канал с возможностью генерации ШИМ сигнала с максимальным напряжением 5 В для управления напряжением на сетке
- 3 аналоговых канала для снятия значений напряжения
- 8 цифровых каналов для переключения реле сигнала с максимальным напряжением 5 В

- аппаратный *UART* для приёма-передачи данных

Данным требованиям удовлетворяет множество микроконтроллеров.

Далее проведено сравнение нескольких из них.

1.1.2 *MH-ETLIVE*.

Плата *MH-ET LIVE* представляет собой аналог *Arduino Nano* но, в отличие от *Arduino Nano*, разработана на основе микроконтроллера *ATtiny88*. *ATtiny88* – это 8-ми разрядный микроконтроллер изготовленный по технологии комплексной структуры метал-оксид-полупроводник (КМОП или *CMOS*) с низким энергопотреблением. Выполняя мощные инструкции за один такт, *ATtiny88* достигает пропускной способности достигающей к 1 *MIPS/МГц*.

ATtiny88 использует следующие режимы для экономии электроэнергии:

1. Режим ожидания: останавливает процессор, позволяя таймеру/счетчику, АЦП, аналоговому компаратору, *SPI*, *TWI* и системе прерываний продолжать работать.

					ПГУ.2.11.03.04.006ПЗ		
Изм.	Лист	№ докум.	Подпись	Дат			
Разраб.		Зубанов Д.В			Лит.	Лист	Листов
Провер.		Вольсков А.А.				5	80
Н. Контр.		Рыжова Т.Н.		18.06	Выбор программно-аппаратных средств для реализации проекта		
Утверд.							
					группа 19ПД		

2. Режим шумоподавления АЦП: минимизирует шум переключения во время преобразования АЦП за счет остановки процессора и всех модулей ввода-вывода, кроме АЦП.

3. Режим отключения питания: регистры сохраняют свое содержимое, все функции микросхемы отключены до следующего прерывания или аппаратного сброса.

Устройство изготовлено с использованием технологии энергонезависимой памяти высокой плотности Atmel. Встроенная флэш-память ISP позволяет перепрограммировать память программ в системе через последовательный интерфейс *SPI*, с помощью обычного программатора

энергонезависимой памяти или с помощью встроенного загрузчика программ, работающего на ядре *AVR*. Загрузочная программа может использовать любой интерфейс для загрузки прикладной программы во флэш-память. Объединяя 8-разрядный процессор *RISC* с встроенной самопрограммируемой флэш-памятью на монолитном чипе, *AtmelATtiny88* представляет собой мощный микроконтроллер, который обеспечивает очень гибкое и экономичное решение для многих встроенных приложений управления.

AVR ATtiny88 поддерживается полным набором инструментов для разработки программ и систем, включая компиляторы C, макроассемблеры, отладчики/симуляторы программ и оценочные наборы.

ATtiny88 имеет на борту:

- *flash* память: 8 кБ (6780 Байт с учетом загрузчика);
- *SRAM* память: 512 Б;
- *EEPROM* память: 64;Б
- таймеры: 1х 8 Бит без ШИМ + 1х полноценный 16 БИТ;
- *GPIO*: 26 пинов;
- прерывания: 2х *INT*, все пины – *PCINT*;
- ШИМ: 2 пина;

- АЦП: 8 пинов, внешнего опорного нет;
- интерфейсы: *I2C*, *SPI*.

Не имеет на борту:

- аппаратный *UART*, про *Serial* можно забыть;
- аппаратное умножение. Вычисления будут медленнее.

Распиновка *MH-ETLIVE* изображена на рисунке 2.

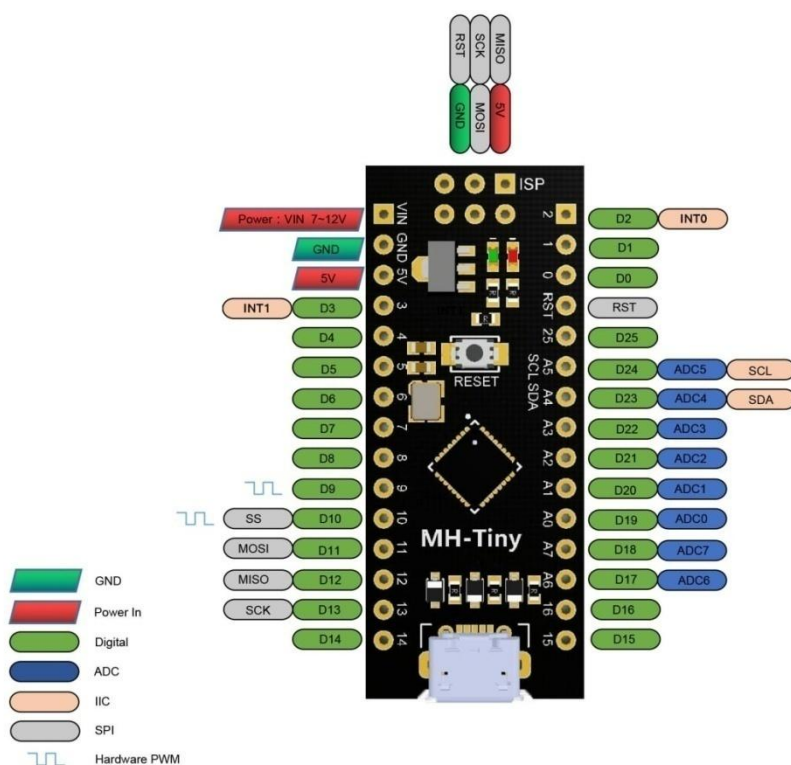


Рисунок 1 – Распиновка *MH-ETLIVE*

Для достижения максимальной производительности и параллелизма, AVR использует Гарвардскую архитектуру процессора – с отдельной памятью и шинами для программ и данных. Инструкции в памяти программы выполняются с помощью одноуровневой конвейерной обработки. Пока выполняется одна инструкция, следующая инструкция предварительно извлекается из памяти программы. Эта концепция позволяет выполнять инструкции в каждом такте. Память программ представляет собой внутрисистемную перепрограммируемую флэш-память.

Файл регистров быстрого доступа содержит 32 8-разрядных рабочих регистра общего назначения со временем доступа в один такт. Это позволяет

арифметическому логическому устройству (*ALU*) выполнять операции за один цикл. В типичной операции *ALU* два операнда получают из файла регистров, выполняется операция и результат сохраняется обратно в файле регистров – за один такт.

Шесть из 32-х регистров могут использоваться в качестве трех 16-ти разрядных регистров указателей косвенных адресов для данных. Пространственная адресация – позволяет эффективно вычислять адреса. Один из этих указателей адреса также может использоваться в качестве указателя адреса для поиска таблиц во флэш-памяти программы. Регистрами с дополнительным функционалом являются 16-разрядные регистры *X*, *Y* и *Z*.

ALU поддерживает арифметические и логические операции между регистрами или между константой и регистром. Операции с одним регистром также могут выполняться в *ALU*. После выполнения арифметической операции обновляется регистр состояния (*SREG*), чтобы отобразить информацию о результате выполнения операции.

Выполнение программы обеспечивается инструкциями условного и безусловного перехода и вызова подпрограмм способными непосредственно обращаться ко всему адресному пространству памяти программ. Большинство инструкций *AVR* имеют единый 16-разрядный формат слов, но существуют также инструкции 32-разрядного формата. Фактический набор инструкций варьируется, так как некоторые микроконтроллеры реализуют только часть набора инструкций.

Во время прерываний и вызовов подпрограмм, адрес возврата берётся из счетчика команд (*PC*) помещается в стек. Стек находится в *SRAM* памяти общих данных и, следовательно, размер стека ограничен только общим размером *SRAM* и её использованием в программе. Все пользовательские программы должны инициализировать указатель стека (*SP*) в процедуре сброса до выполнения подпрограмм или прерываний. Указатель стека (*SP*) доступен для чтения/записи в пространстве адресов регистров ввода-вывода. К данным

SRAM можно легко получить доступ с помощью пяти различных режимов адресации, поддерживаемых архитектурой *AVR*. Все области памяти в архитектуре *AVR* являются линейными и регулярными картами памяти.

Гибкий модуль прерываний имеет свои управляющие регистры в адресном пространстве регистров ввода-вывода и дополнительный глобальный Бит разрешения/запрета прерываний в регистре состояния (*SREG*). Все прерывания имеют отдельный вектор прерывания в таблице векторов прерываний. Прерывания имеют приоритет в соответствии с номером их вектора прерывания. Чем ниже адрес вектора прерывания, тем выше приоритет.

64 регистра ввода-вывода обслуживают периферию *CPU*, это управляющие регистры *SPI* модуля и другие. Доступ к памяти регистров ввода/вывода можно получить напрямую или по адресу в пространстве данных *SRAM*. Регистры ввода/вывода располагаются в *SRAM* сразу за файлом регистров быстрого доступа по адресам 0x20 - 0x5F. Кроме того, в *ATtiny88* появились дополнительные регистры ввода-вывода которые занимают адреса с 0x60 до 0xFF в *SRAM*. В адресном пространстве дополнительных регистров ввода-вывода (0x60 до 0xFF) можно использовать только инструкции *ST/STS/STD* и *LD/LDS/LDD*[2].

Исходя из вышеперечисленных характеристик и особенностей данного микроконтроллера, *MH-ETLIVE* не подходит для реализации проекта, так как для корректного взаимодействия необходим последовательный порт *USART*, который отсутствуету микроконтроллера, на базе которого разработана данная плата. Главным плюсом данной платы является её цена – 100 - 200 рублей.

1.1.3 *ArduinoNano*.

Arduino Nano – это интеллектуальная плата разработки, предназначенная для создания более быстрых прототипов с минимальными размерами. *Arduino Nano*, будучи старшим членом семейства *Nano*, обеспечивает достаточное количество интерфейсов для ваших приложений на макетной плате. Сердцем платы является микроконтроллер *ATmega328*, работающий на частоте 16 МГц и

обладающий примерно теми же функциональными возможностями, что и *Arduino Duemilanove*. Плата имеет 22 цифровых входа/выхода, 8 аналоговых контактов и порт *mini-USB*. Целевыми областями *Arduino Nano* являются производство, безопасность, экология, робототехника и системы управления.

Arduino Nano – это первый встраиваемый микроконтроллер серии Nano с минимальными функциональными возможностями, предназначенный для мини-проектов сообщества мейкеров. Большое количество входных/выходных контактов дает преимущество использования нескольких последовательных коммуникаций, таких как *UART*, *SPI* и *I2C*. Оборудование совместимо с *Arduino IDE*, *Arduino CLI* и веб-редактором.

Безопасность: Высокая производительность и низкое энергопотребление дают возможность разрабатывать приложения, основанные на безопасности, например, системы контроля доступа с использованием датчиков отпечатков пальцев. Гибкость интерфейса датчиков и внешних устройств с использованием последовательной связи расширила сферу применения.

Экология: Функция низкого энергопотребления микроконтроллера и возможности питания платы расширили возможности реализации удаленных *IoT*-проектов, связанных с экологическими проблемами.

Основным процессором в плате *Arduino Nano v3.3* является высокопроизводительный и маломощный 8-битный микроконтроллер *ATmega 328*, работающий на тактовой частоте 16 МГц. Возможность взаимодействия с внешними устройствами посредством последовательной связи поддерживается микросхемой с помощью *UART TTL (5V)*, *I2C (TWI)* и *SPI*. *Arduino Nano* можно программировать с помощью программного обеспечения *Arduino*, что снижает входные барьеры для новых пользователей. Встроенное оборудование наименьшего размера делает его идеальным выбором для проектов с макетной платой, предлагаемых сообществом мейкеров.

Микроконтроллер *ATmega 328* имеет высокопроизводительный маломощный 8-битный процессор достигает до 16 *MIPS* при тактовой частоте

16 МГц 32 КБ, из которых 2 КБ используются загрузчиком, 2 КБ внутренней памяти *SRAM*, 1 КБ *EEPROM*, 32 x 8 рабочих регистров общего назначения, счетчик реального времени с отдельным генератором, шесть каналов ШИМ, программируемый последовательный интерфейс *USART*, ведущий/ведомый последовательный интерфейс *SPI*. Питание платы может осуществляться через *USB*-подключение *Mini-B*, 6-20 В нерегулируемый внешний источник питания (контакт 30) или 5 В регулируемый внешний источник питания (контакт 27)

Плата обладает следующими режимами работы: холостой ход, снижение шума, АЦП, энергосбережение, отключение питания, режим ожидания, расширенный режим ожидания. Так-же плата имеет 22 цифровых, 8 Аналоговых пинов и 6 ШИМ-выходов. На рисунке 1 изображена распиновка платы *Arduino Nano*.

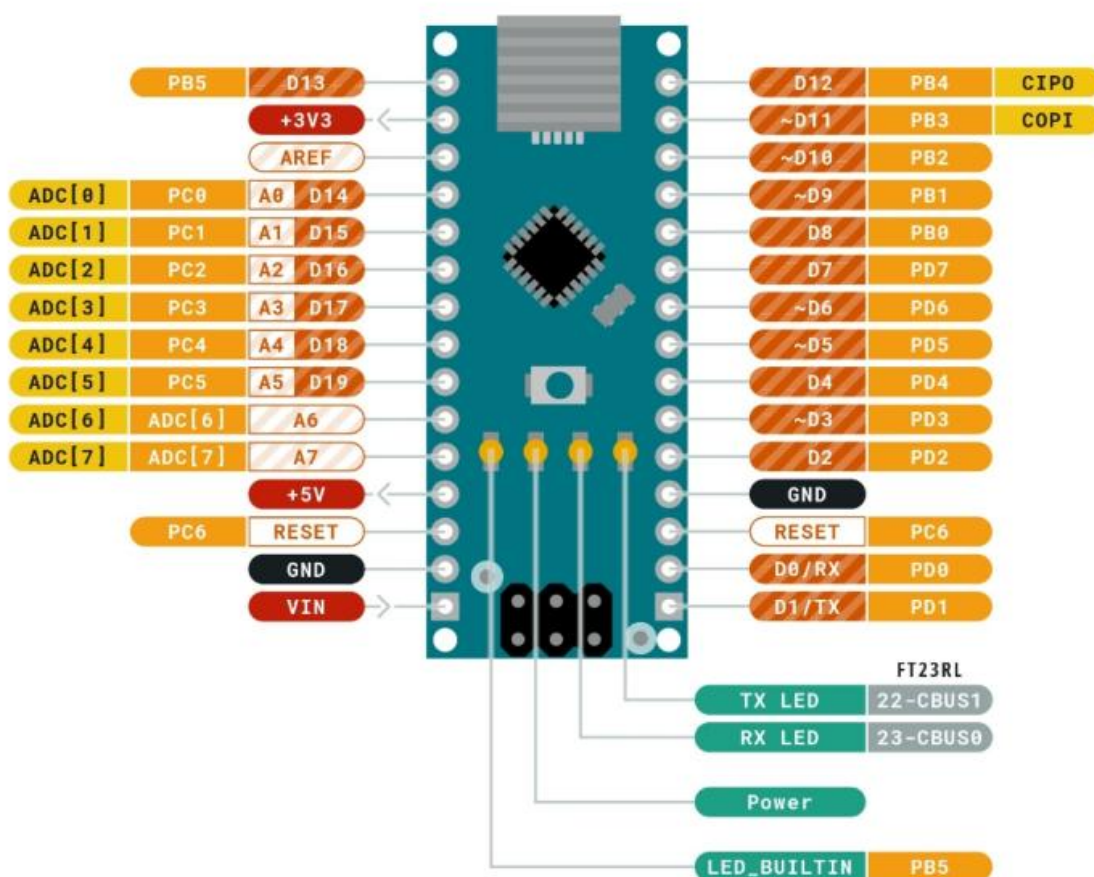


Рисунок 2 – Распиновка платы *Arduino Nano*

Платы *Arduino* полностью соответствуют требованиям Регламента Европейского Союза (ЕС) 1907 /2006 о регистрации, оценке, разрешении и ограничении химических веществ (*REACH*). Ни одно из *SVHC*, список веществ-кандидатов на получение разрешения, опубликованный *ECHA*, не присутствует во всех продуктах (а также в упаковке) в количестве, равном или превышающем 0,1%. Продукция не содержит ни одного из веществ, перечисленных в "Списке разрешенных веществ" и веществ, вызывающих очень сильное беспокойство (*SVHC*), в сколько-нибудь значительных количествах [3].

Данная плата, на основе микроконтроллера *Atmega 328* удовлетворяет условия задания. *ArduinoNano* имеет последовательный интерфейс *USART*, что позволит корректно получать данные и управлять ей с ПК. Так же данная плата позволяет измерять значения постоянного напряжения а так же использовать ШИМ сигнал в качестве управляющего.

Для реализации проекта была выбрана *Arduino Nano*. Она Удовлетворяет минимальные условия для реализации проекта, более проста в освоении и более доступна.

1.2 Выбор программного обеспечения для создания приложения

1.2.1 Требования к программному обеспечению для создания приложения.

Приложение должно иметь возможность разрабатывать программы с простым и интуитивно понятным интерфейсом. Так же приложение должно иметь широкий набор инструментов и возможность работы с последовательным портом.

1.2.2 *Visualstudio*.

Microsoft Visual Studio- линейка продуктов компании *Microsoft*, включающих интегрированную среду разработки программного обеспечения и ряд других инструментов. Данные продукты позволяют разрабатывать как консольные приложения, так и игры и приложения с графическим интерфейсом, в том числе с поддержкой технологии *WindowsForms*, *UWP* а

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		12

также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых *Windows*, *WindowsMobile*, *WindowsCE*, *.NETFramework*, *.NETCore*, *.NET*, *MAUI*, *Xbox*, *WindowsPhone* *.NETCompactFramework* и *Silverlight*. После покупки компании *Xamarin* корпорацией *Microsoft* появилась возможность разработки *IOS* и *Android* программ.

VisualStudio включает в себя редактор исходного кода с поддержкой технологии *Intelli Sense* и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных. *VisualStudio* позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как, например, *Subversion* и *VisualSourceSafe*), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования) или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент *TeamExplorer* для работы с *TeamFoundationServer*) [4].

1.2.3 *Processing*.

Среда разработки *Processing (PDE)* позволяет легко писать программы для *Processing*. Программы пишутся в текстовом редакторе и запускаются нажатием кнопки *Run*. В *Processing* компьютерная программа называется скетчем. Скетчи хранятся в папке *Sketchbook*, которая находится на вашем компьютере.

Скетчи могут рисовать двух- и трехмерную графику. Рендерер по умолчанию предназначен для рисования двухмерной графики. Рендерер *P3D* позволяет рисовать трехмерную графику, которая включает управление камерой, освещением и материалами. Рендерер *P2D* - это быстрый, но менее точный рендерер для рисования двухмерной графики. Оба рендера *P2D* и *P3D* ускоряются, если на вашем компьютере установлена видеокарта, совместимая с *OpenGL*.

Возможности *Processing* расширяются с помощью библиотек и инструментов. Библиотеки позволяют эскизам выполнять действия, выходящие за рамки основного кода *Processing*. Существуют сотни библиотек, созданных сообществом разработчиков *Processing*, которые можно добавить к вашим скетчам, чтобы получить новые возможности, такие как воспроизведение звуков, компьютерное зрение и работа с расширенной 3D-геометрией. Инструменты расширяют *PDE*, чтобы облегчить создание эскизов, предоставляя интерфейсы для таких задач, как выбор цветов.

Processing имеет различные режимы программирования, чтобы можно было развернуть эскизы на разных платформах и программировать по-разному. По умолчанию используется режим *Java*. Другие режимы программирования можно загрузить, выбрав "Добавить режим..." из меню в правом верхнем углу *PDE*.

Среда разработки *Processing (PDE)* состоит из простого текстового редактора для написания кода, области сообщений, текстовой консоли, вкладок для управления файлами, панели инструментов с кнопками для общих действий и ряда меню.

Среда разработки *Processing (PDE)* обладает широкими возможностями настройки. Наиболее распространенные настройки можно изменить в окне *Preferences*, расположенном в меню *File* в *Windows* и *Linux* и в меню *Processing* в *macOS*. Полный список предпочтений хранится в файле "*preferences.txt*". Этот файл можно открывать и редактировать напрямую, только когда *Processing* не

запущен. Расположение этого файла на компьютере можно узнать, прочитав в левом нижнем углу окна *Preferences*.

Отладчик *ProcessingDebugger* - это инструмент для диагностики проблем в скетче. Включите его, чтобы приостановить выполнение скетча и продвигаться по коду по одной строке за раз. Отладчик включается через меню Файл (Отладка> Включить отладчик) или щелчком по значку отладчика - бабочке в правом верхнем углу *PDE*.

Когда отладчик включен, программа выполняется как обычно, но останавливается в "точках останова". Чтобы создать точку останова, установите курсор на строку, на которой вы хотите приостановить выполнение эскиза, и выберите *Debug>ToggleBreakpoint*. Сокращение клавиш - *Command-B*. Чтобы удалить точку останова, снова выберите *ToggleBreakpoint*. При добавлении точки останова номер строки заменяется символом: <>.

Запуск эскиза в режиме отладки приводит к тому, что эскиз приостанавливается в любых точках останова. В режиме паузы текущие значения переменных отображаются в отдельной панели. Вы можете перейти к следующей точке останова, выбрав "Продолжить", или продвигаться построчно по коду с помощью "Шага". Шаг работает только в пределах текущей выполняемой функции.

Processing позволяет создавать эскизы двух уровней сложности, которые мы называем "статическими" и "активными" эскизами. Новички в программировании часто начинают со статических эскизов, чтобы узнать о координатах, переменных и других основах, прежде чем переходить к активным эскизам.

Статические эскизы не включают анимацию или взаимодействие, чтобы сосредоточиться на первых основах программирования. Короткие строки кода имеют прямое представление на экране.

Активные скетчи включают в себя структуру *setup()*, которая запускается один раз при запуске скетча, и структуру *draw()*, которая по

умолчанию постоянно выполняет циклический просмотр кода внутри. Эта дополнительная структура позволяет писать пользовательские функции и классы, а также использовать события клавиатуры и мыши [5].

Для реализации проекта была выбрана среда программирования *VisualStudio*. Эта среда программирования более проста в освоении, так как существует множество видео-уроков а так же статей посвящённых работе с данной средой программирования. Также она более доступна и работе с данной средой нас обучали по учебной программе

1.3 Выбор программы для прошивки микроконтроллера

1.3.1 Требования к среде программирования для программирования микроконтроллера.

Среда программирования для программирования Микроконтроллера должна быть доступной по цене и простой в освоении. Так же среда программирования для прошивки микроконтроллера должна обладать возможностью прошить выбранный микроконтроллер.

1.3.2 *ArduinoIDE*.

ArduinoIDE — интегрированная среда разработки для *Windows*, *i* и *Linux*, разработанная на *C* и *C ++*, предназначенная для создания и загрузки программ на *Arduino*-совместимые платы, а также на платы других производителей.

Исходный код для среды выпущен под общедоступной лицензией *GNU* версии 2. Поддерживает языки *C* и *C ++* с использованием специальных правил структурирования кода. *ArduinoIDE* предоставляет библиотеку программного обеспечения из проекта *Wiring[en]*, которая предоставляет множество общих процедур ввода и вывода. Для написанного пользователем кода требуются только две базовые функции для запуска эскиза и основного цикла программы, которые скомпилированы и связаны с заглушкой программы *main ()* в исполняемую циклическую программу с цепочкой инструментов *GNU*, также

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		16

включённой в дистрибутив *IDE*. Использует программу *avrdude* для преобразования исполняемого кода в текстовый файл в шестнадцатеричной кодировке, который загружается в плату *Arduino* программой-загрузчиком во встроенном программном обеспечении платы.

С ростом популярности *Arduino* другие поставщики в качестве программной платформы начали внедрять пользовательские компиляторы и инструменты с открытым исходным кодом (ядра), которые могут создавать и загружать эскизы в другие микроконтроллеры, не поддерживаемые официальной линейкой микроконтроллеров *Arduino*.

В октябре 2019 года организация *Arduino* начала предоставлять ранний доступ к новой *ArduinoProIDE* с отладкой и другими расширенными функциями[6]. В настоящее время при помощи *ArduinoIDE* можно запрограммировать множество микроконтроллеров различных производителей.

1.3.3 *Microchip Studio*.

Microchip Studio - это интегрированная среда разработки (*IDE*) для разработки и отладки приложений на микроконтроллерах *AVR®* и *SAM*. Она объединяет все функции *Atmel Studio* с хорошо поддерживаемым портфолио инструментов разработки *Microchip*, предоставляя бесшовную и простую в использовании среду для написания, сборки и отладки приложений, написанных на *C / C++* или ассемблерном коде. *Microchip Studio* также может импортировать эскизы *Arduino* в виде проектов на *C++*, чтобы обеспечить простой переход из *makerspace* в *marketplace*.

Имеется возможность использовать *Microchip Studio* с отладчиками, программистами и наборами разработки, которые поддерживают устройства *AVR* и *SAM*. Несмотря на то, что у нее новое название и внешний вид, по-прежнему возможно использовать любую существующую документацию и видеоролики об *Atmel Studio*, чтобы научиться пользоваться *Microchip Studio*.

Microchip Studio поставляется с установленным и готовым к использованию компилятором *MPLABXC8*. Последняя версия лицензии *MPLABXC8 PRO* включает оптимизации, которые конкурируют с более дорогими компиляторами, представленными на рынке, для уменьшения размера кода и повышения эффективности. Также доступна лицензия на функциональную безопасность, сертифицированная *TÜVSÜD*

Основные характеристики:

- поддержка более 500 устройств *AVR* и *SAM*;
- поддержка компилятора *MPLAB® XC8*;
- обширная библиотека исходного кода, включающая драйверы, стеки связи, более 1600 примеров проектов с исходным кодом, графические сервисы и сенсорную функциональность с помощью *Advanced Software Framework (ASF)*;
- расширения *IDE* через *Microchip Gallery*, онлайн-магазин приложений для инструментов разработки и встроенного программного обеспечения от *Microchip* и третьих сторон;
- *QTouch Composer* для настройки емкостных сенсорных экранов, проверки производительности системы, мониторинга энергопотребления и построения графиков и трассировки данных в режиме реального времени;
- *Wireless Composer* для настройки и тестирования производительности беспроводных устройств;
- встроенный компилятор для написания и отладки *C / C++* и ассемблерного кода;
- расширенные функции отладки, включая сложные точки останова данных, поддержку ненавязчивой трассировки (устройства *SAM3* и *SAM4*), статистическое профилирование кода, трассировку прерываний / мониторинг, трассировку опрошенных данных (устройства на базе *Arm® Cortex-M0+*), отслеживание переменных в реальном времени с дополнительной временной меткой;

- встроенный редактор с визуальной поддержкой;
- мастер создания проектов, позволяющий создавать проекты с нуля или на основе большой библиотеки примеров дизайна;
- внутрисистемное программирование и отладка, обеспечивающее интерфейс для всех встроенных программистов и отладчиков *Microchip*;
- прозрачные отладочные представления в процессоре и периферийных устройствах для упрощения разработки кода и отладки;
- полнофункциональное моделирование для точной модели процессора, прерываний, периферийных устройств и внешних раздражителей.

Подключаемый модуль *Data Visualizer* собирает и отображает данные об энергопотреблении приложения во время выполнения при использовании с *Power Debugger* или поддерживаемой платой *Xplained PRO*. Имеется возможность профилировать энергопотребление приложения в рамках стандартного сеанса отладки. Так же имеется возможность сопоставить скачки напряжения с кодом, который их вызвал, путем выборки программного счетчика во время измерений мощности.

Microchip Studio позволяет импортировать проекты, созданные в среде разработки *Arduino*, одним щелчком мыши. Эскиз, включая все библиотеки, на которые он ссылается, импортирован в *Studio* как проект на C++. После импорта эскиза можно использовать все возможности *Studio* для точной настройки и отладки вашего дизайна. *Microchip Studio* полностью поддерживает мощный встроенный отладчик на плате *Arduino Zero*. Адаптеры *Shield*, открывающие отладочные разъемы, доступны для других плат *Arduino*, или можно воспользоваться возможностью переключиться на одну из множества доступных плат *Xplained Mini* или *Xplained PRO*, чтобы в полной мере использовать экосистему аппаратных средств [7].

Для прошивки микроконтроллера была выбрана среда программирования *ArduinoIDE*. Она бесплатная, а так же обладает возможностью прошить выбранный микроконтроллер и имеет множество статей и видео-уроков посвящённых работе в ней, что позволяет легко освоиться в ней. Так же *Arduino IDE* основана на C++, языке программирования высокого уровня, которого вполне достаточно для реализации проекта.

2 Разработка алгоритмов

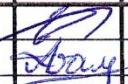
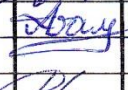
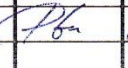
2.1 Разработка алгоритма приёма данных сенда

Для записи значения тока используется мощный резистор с малым сопротивлением и значение напряжения с этого резистора. Поскольку значение напряжения на резисторе больше 5 В, для корректного получения значений напряжения используется резисторный делитель напряжения.

Для записи значения измеренного напряжения, поданного на аналоговый канал, используется функция *analogRead(pin)*. Для оцифровки значения используется встроенный АЦП. АЦП находится в одном корпусе с микроконтроллером. В используемой модели *ArduinoNano* есть восемь аналоговых входов. На распиновке *ArduinoNano* аналоговые входы имеют буквенно-цифровые обозначения A0, A1, ..., A7. Для того, чтобы использовать АЦП в *ArduinoNano*, используются следующие команды:

```
analogRead(pin);  
analogReference();  
analogReadResolution(bits);
```

Каналы АЦП *ArduinoNano* имеют по умолчанию значение опорного 5 В. Это означает, что максимальное входное значение напряжения для каждого канала АЦП *Arduino* составляет 5 В. Но некоторые датчики имеют выходное напряжение в диапазоне 0-2,5 В, поэтому если использовать опорное напряжение по умолчанию (5 В), то точность измерений будет меньше. В связи с этим у *ArduinoNano* имеется возможность изменить значение опорного напряжения. Это делается с помощью команды "analogReference();".

					ПГУ.2.11.03.04.006ПЗ		
Изм.	Лист	№ докум. №	Подпись	Дат			
Разраб. Раз		Зубанов			Разработка алгоритмов	Лит. Ли	Лист Ли
Провер. Про		Вольсков					
Н. Контр. Н.		Рыжова Т.Н.		19.06		20	86
Утверд. Ут						группа 19ПДгруппа	

Значение, полученное с аналогового канала, будет находиться в диапазоне от 0 да 1023. У каждого устройства АЦП есть параметр разрядности. От этого параметра напрямую зависит точность измерений. Для АЦП с разрядностью 1, подавая на вход любое напряжения от 0 до 2,5 вольт, на выходе получится 0. Любое же напряжение от 2,5 до 5 вольт даст единицу. То есть 1-битный АЦП сможет распознать только два уровня напряжения. Графически это изображено на рисунке 3.

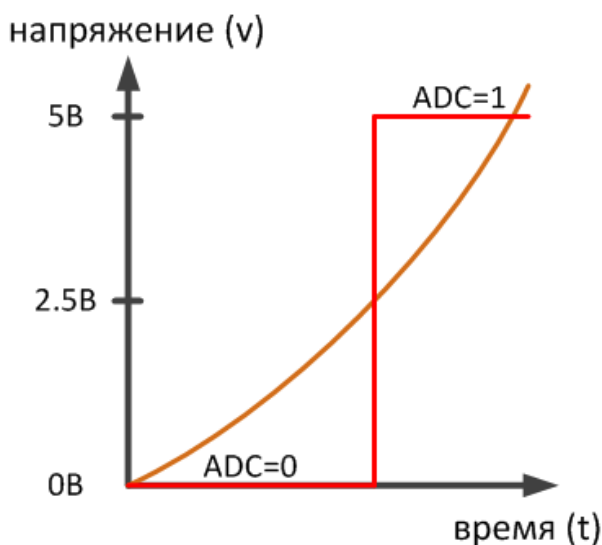


Рисунок 3 – Графическое изображение 1 битного АЦП

АЦП с разрядностью 2 распознает уже четыре уровня напряжения:

- от 0 до 1,25 — это 0;
- от 1,25 до 2,5 — это 1;
- от 2,5 до 3,75 — это 2;
- от 3,75 до 5 — это 3.

На рисунке 4 изображена работа АЦП с разрядностью 2 и 3 бит:

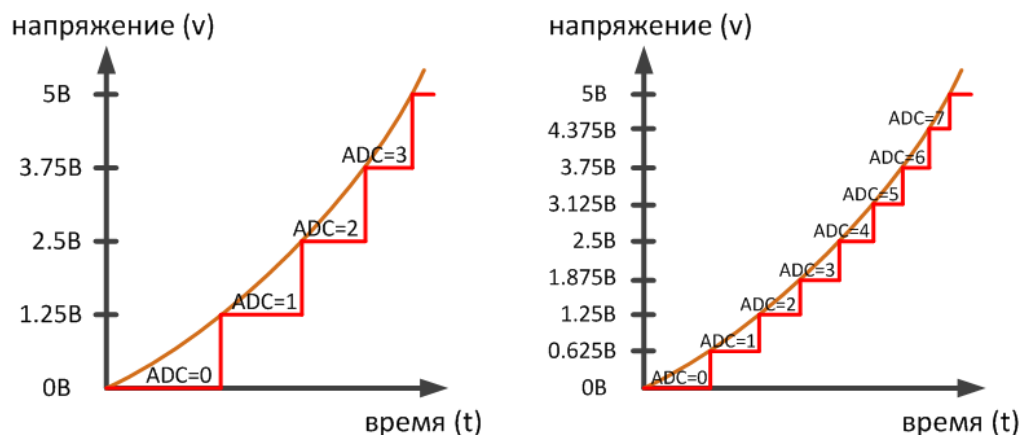


Рисунок 4 – Графическое изображение 2 битного и 3 битного АЦП

В *Arduino Nano* установлен 10-битный АЦП. Это значит, что любое напряжение на аналоговом входе в диапазоне от 0 до 5 вольт будет преобразовано в число с точностью 0.005 мВ.[8]

Для приема данных в *VisualStudio* используется функция `ReadLine()`. Данная функция считывает значения из выходного буфера и до значения `SerialPort.NewLine` и возвращает содержимое входного буфера до значения `SerialPort.NewLine`. Для получения более точного значения проводится 1000 итераций записи и находится среднее значение напряжения. Это позволяет получить более точный результат, а так же минимизировать воздействие шумов.

Поскольку функция `analogRead()` возвращает значение с АЦП в диапазоне от 0 до 1023, необходимо преобразование по формуле:

$$V = A / 1024 * V_{op} * K_{dn} \quad (1)$$

где

- *A*-переменная, которой присвоено значение возвращённое функцией `analogRead()`,
- *V_{op}* – значение опорного напряжения,
- *K_{dn}* – коэффициент делителя напряжения.

Для диодной вакуумной лампы необходимы значение напряжения на самой лампе и значение напряжения на резисторе включенным последовательно. Для триодной и пентодной вакуумной лампы кроме

ранее перечисленных значений, необходимо значение напряжение на резисторе подключённого к сетке, следовательно алгоритм записи должен быть привязан к выбранной вакуумной лампе. Блок-схема алгоритма записи изображена на рисунке 5.

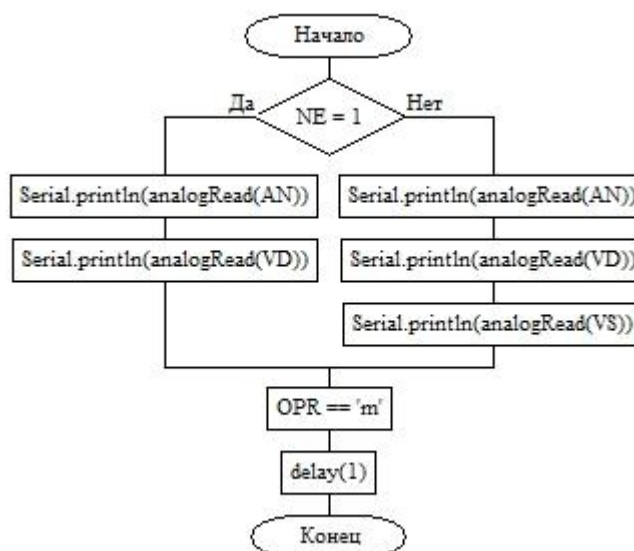


Рисунок 5 – Блок-схема алгоритма записи

2.2 Разработка алгоритма управления напряжением на сетке при помощи ШИМ сигнала

Для управления напряжением на сетке используется ШИМ сигнал. Он задаётся при помощи команды *analogWrite(pin, meaning)*, где *pin* подразумевает канал, на который нужно подать ШИМ сигнал, а *meaning* означает значение коэффициента заполнения ШИМ сигнала от 0 до 255.

Широтно-импульсная модуляция – процесс управления мощностью методом пульсирующего включения и выключения потребителя энергии. Различают аналоговую ШИМ и цифровую ШИМ, двоичную (двухуровневую) ШИМ и троичную (трёхуровневую) ШИМ [9]. График “суммарного” (интегрированного) напряжения за некоторый период относительно коэффициента заполнения изображён на рисунке 6.

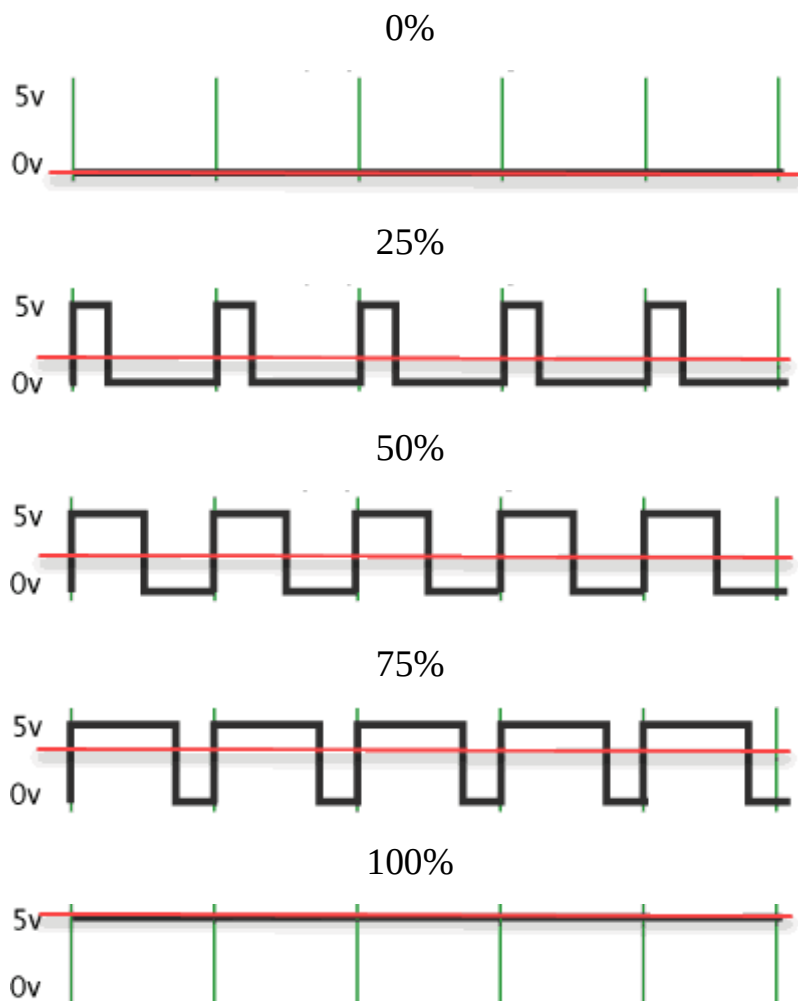


Рисунок 6 – График “суммарного” (интегрированного) напряжения за некоторый период относительно коэффициента заполнения

Основной причиной применения ШИМ является стремление к повышению КПД при построении вторичных источников питания электронной аппаратуры и в других узлах. Реализуемый в контроллерах широтно-импульсный модулятор состоит из двух блоков: линейного интегратора (И-звена) и трехпозиционного релейного элемента. Установленными при изготовлении изделия параметрами схемы являются: постоянная времени И-звена (t_i) и уровень сигнала на выходе релейного элемента.

Широтно-импульсный модулятор генерирует последовательность импульсов со скважностью, пропорциональной уровню сигнала на его входе. Параметр его настройки, то есть минимальная длительность импульса,

устанавливается с помощью зоны нечувствительности релейного элемента широтно-импульсного модулятора[10].

Для повышения быстродействия значения коэффициента заполнения ШИМ сигнала заданы в массиве и загружаются на микроконтроллер. Каждому значению присвоен свой номер. На микроконтроллер поступает сигнал со значением номера ШИМ сигнала. После, при помощи функции *analogWrite()*. Блок-схема управления напряжением изображена на рисунке 7.

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		25

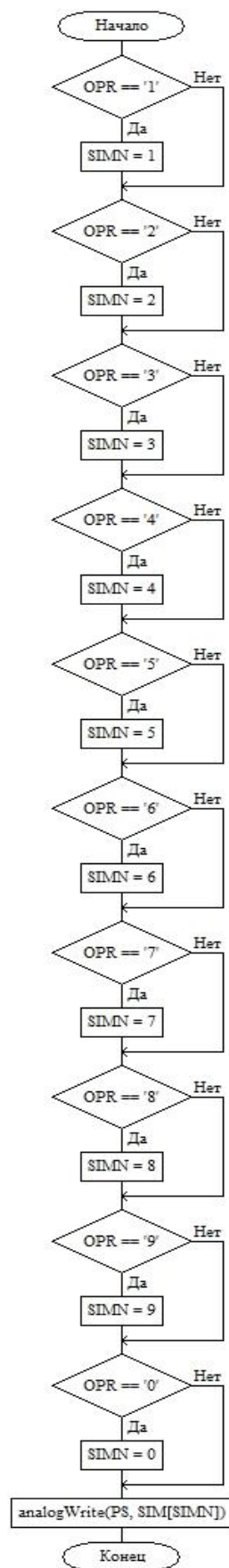


Рисунок 7 – Блок-схема управления напряжением

2.3 Разработка алгоритма управления стендом

Для выбора вакуумной лампы, с которой будут проводиться замеры, используются реле с логическим уровнем единицы равным напряжению 5 В. Для подачи логической единицы на требуемый цифровой канал используется функция *digitalWrite(pin, LOW/HIGH)*. *LOW* и *HIGH* означают логический ноль и логическую единицу. Блок-схема алгоритма управления стендом изображена на рисунке 8.

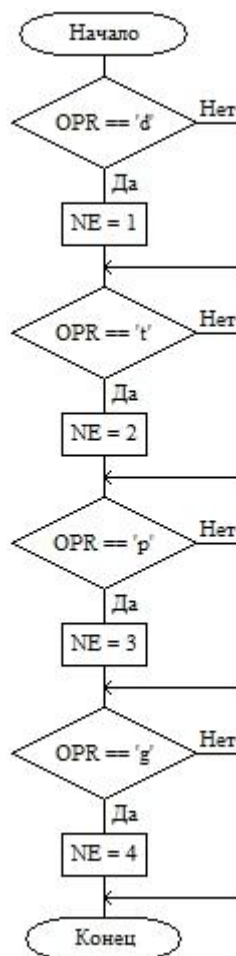


Рисунок 8 – Блок-схема алгоритма управления стендом

2.4 Алгоритм обработки результатов эксперимента

После записи значений напряжения необходимо рассчитать значения тока анода для диода и значения тока анода и сетки для триода и пентода. Так же необходимо провести расчёты крутизны и внутреннего сопротивления для диода, триода и пентода.

Значения токов рассчитываются по закону Ома:

$$I = \frac{U}{R} \quad (2)$$

Для расчёта крутизны ВАХ используется следующая формула:

$$S = \frac{dI_a}{dU_a} \quad (3)$$

Значение параметра крутизны рассчитываются для каждой разности значений напряжений и токов. Значение параметра внутреннего сопротивления для каждой пары значений напряжения и тока для диодной вакуумной лампы рассчитываются по формуле:

$$R_i = \frac{1}{S} \quad (4)$$

После расчёта параметров диодной вакуумной лампы нужно открыть новую форму и вывести на неё графики ВАХ, график зависимости крутизны от тока и график зависимости внутреннего сопротивления от тока. Для триодной и пентодной вакуумной лампы добавляется сеточная ВАХ.

3 Изготовление макета учебного стенда

3.1 Изготовление аппаратной части учебного стенда

Для реализации аппаратной части учебного стенда была выбрана *ArduinoNano* на основе микроконтроллера *AtMega328*. Данный микроконтроллер обладает 12 цифровыми каналами и 8 аналоговыми каналами. Для управления стендом были выбраны каналы, обозначенные символами *D2, D4, D5, D6, D7, D8, D9, D10*. На выходе каждого управляющего канала установлен резистор, номиналом 1.1 кОм, для ограничения тока. Данное ограничение необходимо для того, чтобы при подаче логической единицы на определённый канал, замыкались только выбранные реле. Фотография аппаратной части стенда изображена на рисунке 9.

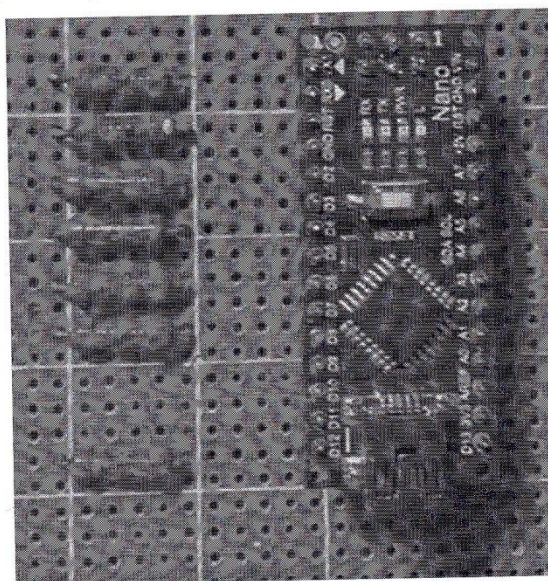


Рисунок 9 – Фотография аппаратной части стенда

Для управления напряжением на сетке был выбран канал, обозначенный символами *D3*. Выбранный канал обладает возможностью генерации ШИМ сигнала. Для измерения значений напряжения были выбраны каналы, обозначенные символами *A0* и *A1*. Данные каналы подключены внутреннему АЦП и могут измерять значение напряжения с точностью до 0.005 вольт.

					ПГУ.2.11.03.04.006ПЗ			
Изм.	Лист	№ докум.	Подпись	Дат	Изготовление макета учебного стенда	Лит.	Лист	Листов
Разраб.		Зубанов Д.В.						
Провер.		Вольсков А.А.					29	80
Н. Контр.		Рыжова Т.Н.		18.08		группа 19ПД		
Утверд.								

3.2 Описание программной части стенда

На главном окне программы находятся основные элементы управления. Кнопка “Подключиться”, при помощи команды `serialPort1.Open()`, открывает порт и подключает микроконтроллер к ПК. Блок-схема алгоритма подключения изображена на рисунке 10.

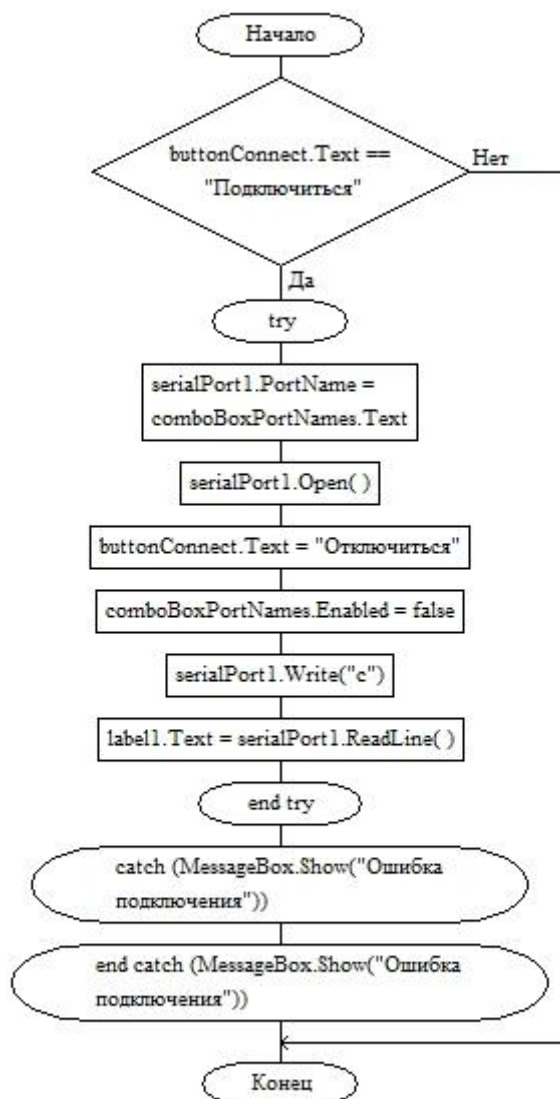


Рисунок 10 – Блок-схема алгоритма подключения

Кнопка “Подключиться” работает в тандеме с кнопкой “Обновить список доступных портов”, расположенной ниже, и списком доступных портов, расположенным левее. Кнопка “Обновить список доступных портов”, при помощи команды `string[] ports = SerialPort.GetPortNames()`, сканирует

доступные порты и выводит их список в *comboBox* слева от кнопки “Подключиться”. Блок-схема алгоритма обновления портов изображена на рисунке 11.

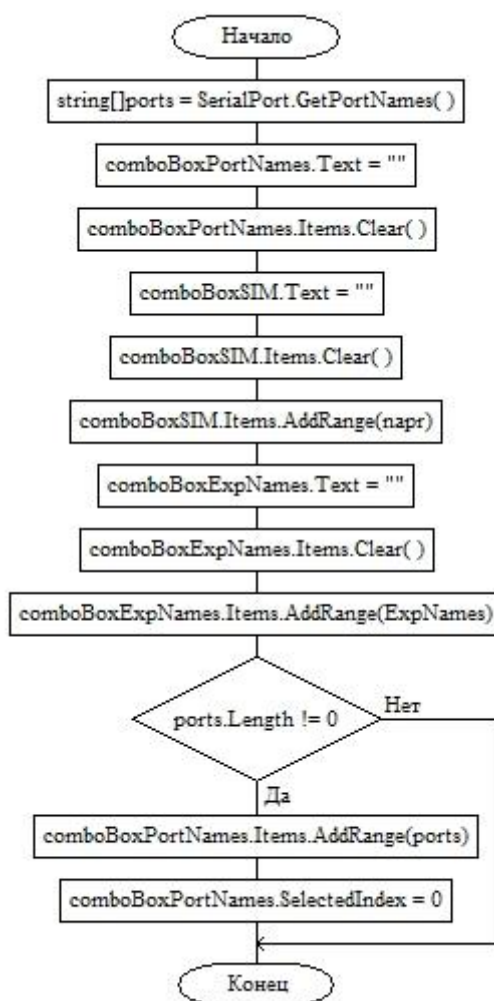


Рисунок 11 – Блок-схема алгоритма обновления портов

Кнопка “Установить эксперимент”, при помощи команды `serialPort1.Write()` передаёт на микроконтроллер сигнал с информацией о типе выбранной вакуумной лампы. Тип вакуумной лампы определяется выбором из списка, расположенного левее данной кнопки. Блок схема управления изображена на рисунке 12.

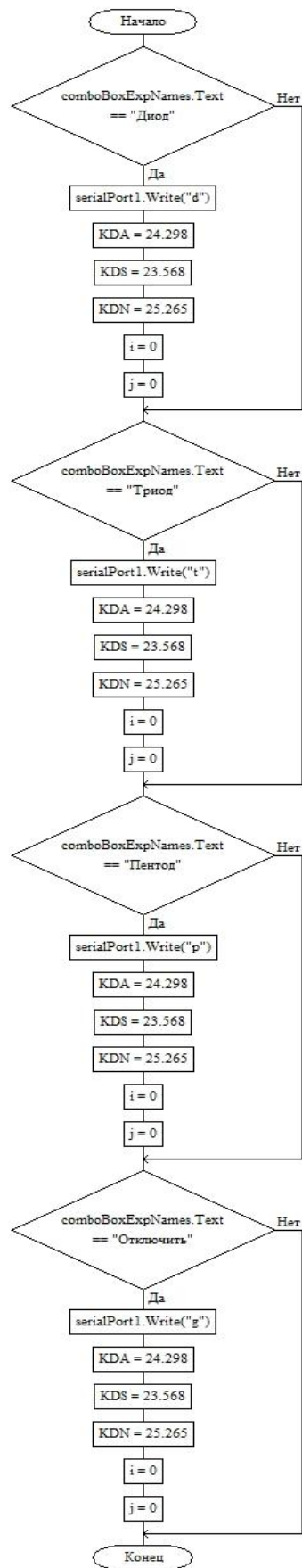


Рисунок 12 – Блок схема управления

Подобным образом работает кнопка “Установить напряжение сетки”. Блок-схема управления напряжением на сетке изображена на рисунке 13.

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		33

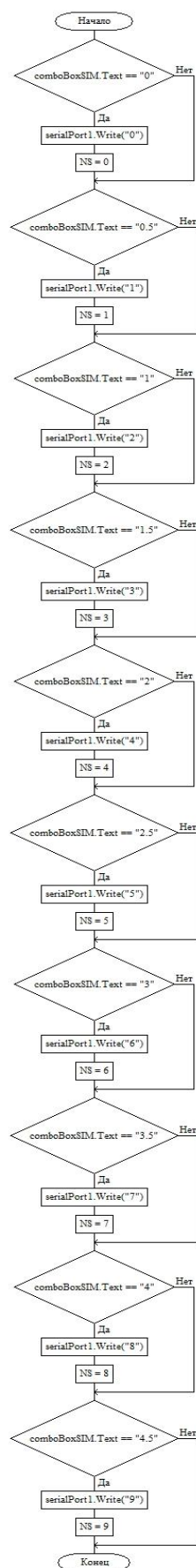


Рисунок 13 – Блок-схема управления напряжением на сетке

Изм.	Лист	№ докум.	Подпись	Дат

ПГУ.2.11.03.04.006ПЗ

Лист

34

При нажатии на кнопку “Записать значения” программа передаёт на микроконтроллер сигнал о начале записи напряжения. Микроконтроллер передаёт на ПК оцифрованные значения напряжений. Данный алгоритм повторяется 1000 раз. После окончания цикла, полученные значения интегрируются, записываются в массив и выводятся в поля, расположенные левее кнопки “Записать значения”. Блок-схема алгоритма записи изображена на рисунке 14.

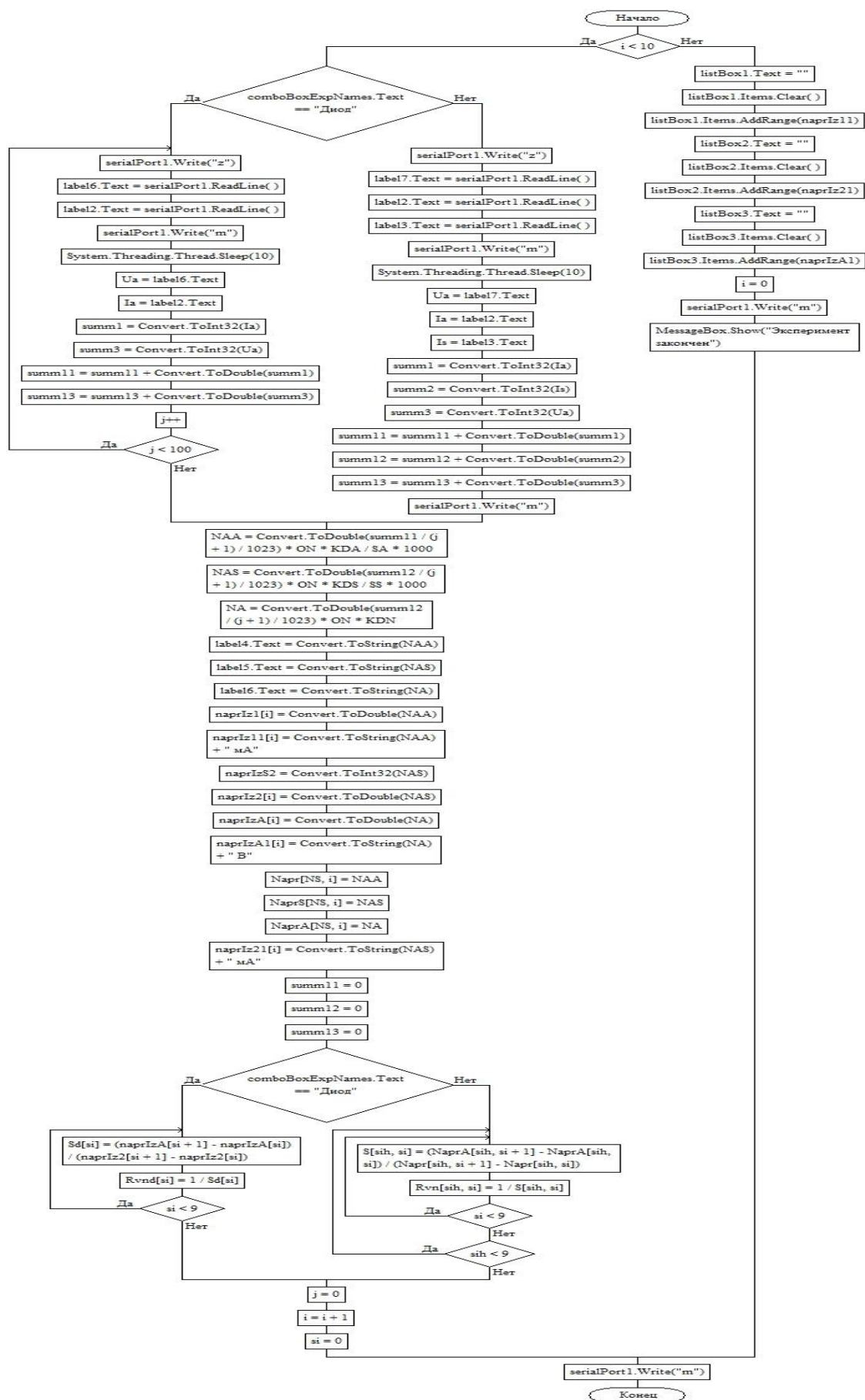


Рисунок 14 – Блок-схема алгоритма записи

Кнопка “Показать графики” выводит графики и значения параметров вакуумных ламп в зависимости от типа выбранной вакуумной лампы. Блок-схема алгоритма работы кнопки “Показать графики” изображена на рисунке 15.

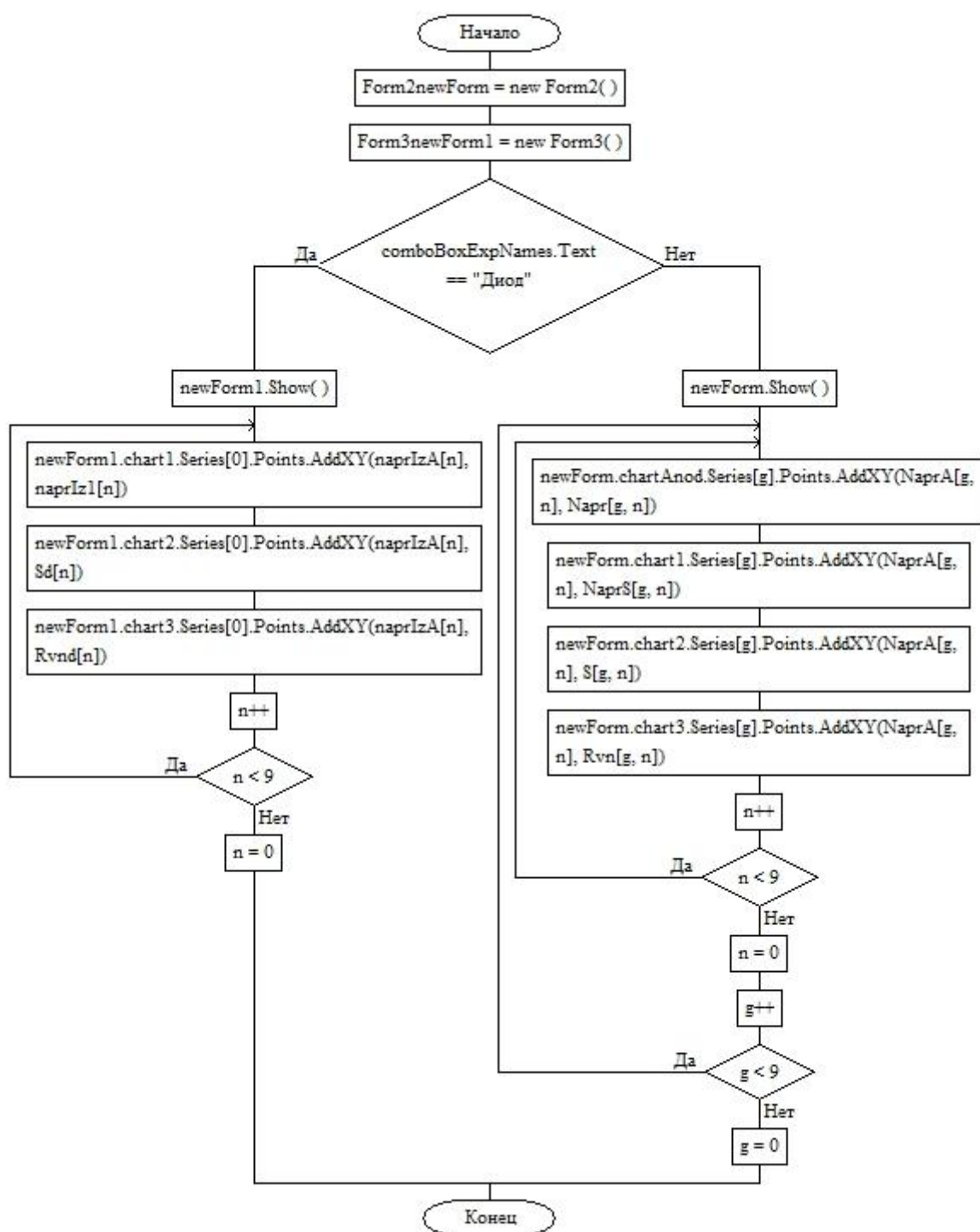


Рисунок 15 – Блок-схема алгоритма работы кнопки “Показать графики”
Скриншот главного экрана программы изображён на рисунке 16.

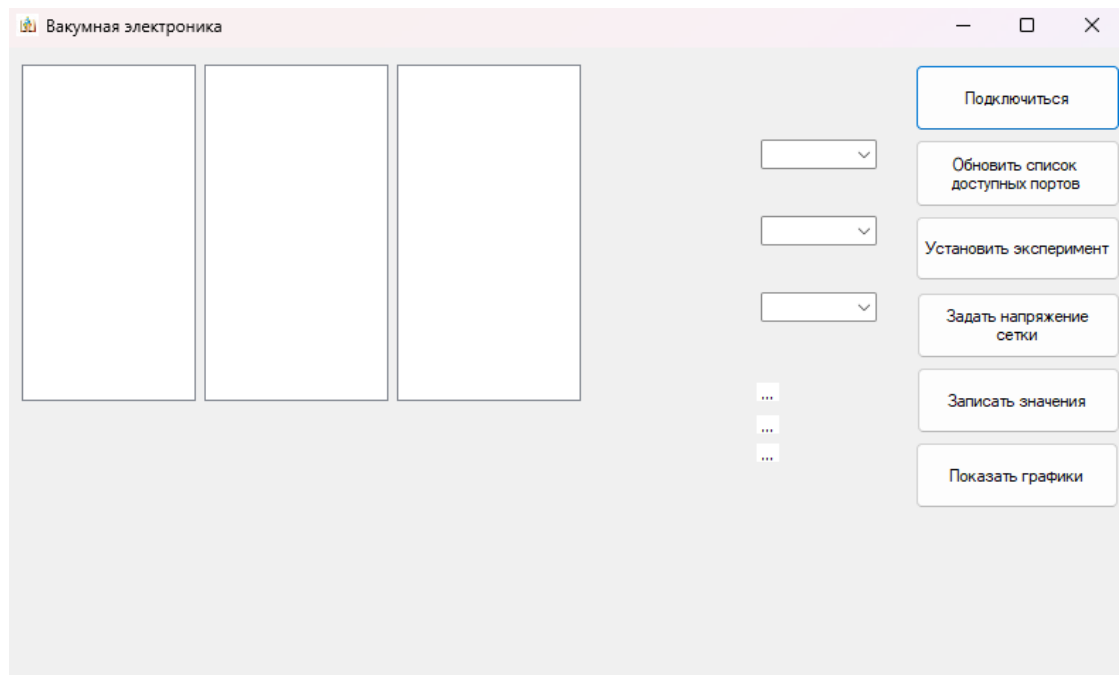


Рисунок 16 – Скриншот главного экрана программы

После проведения 10 итераций записи полученные значения выводятся в виде списка в верхнем левом углу.

3.3 Проверка работоспособности стенда

Для проверки работоспособности стенда необходимо проверить работу всех алгоритмов. С начала проверяется алгоритм подключения стенда к персональному компьютеру. Для этого стенд подключается к ПК через USBпорт. Затем открывается программа. Далее необходимо обновить список портов. В списке слева выбирается необходимый порт. Для подключения стенда к ПК нужно нажать на кнопку “Подключиться”. Если подключение прошло успешно, то в нижнем левом углу программы появится надпись “*Arduinoconnected*”. Скриншот программы после выполнения алгоритма подключения изображён на рисунке 17.

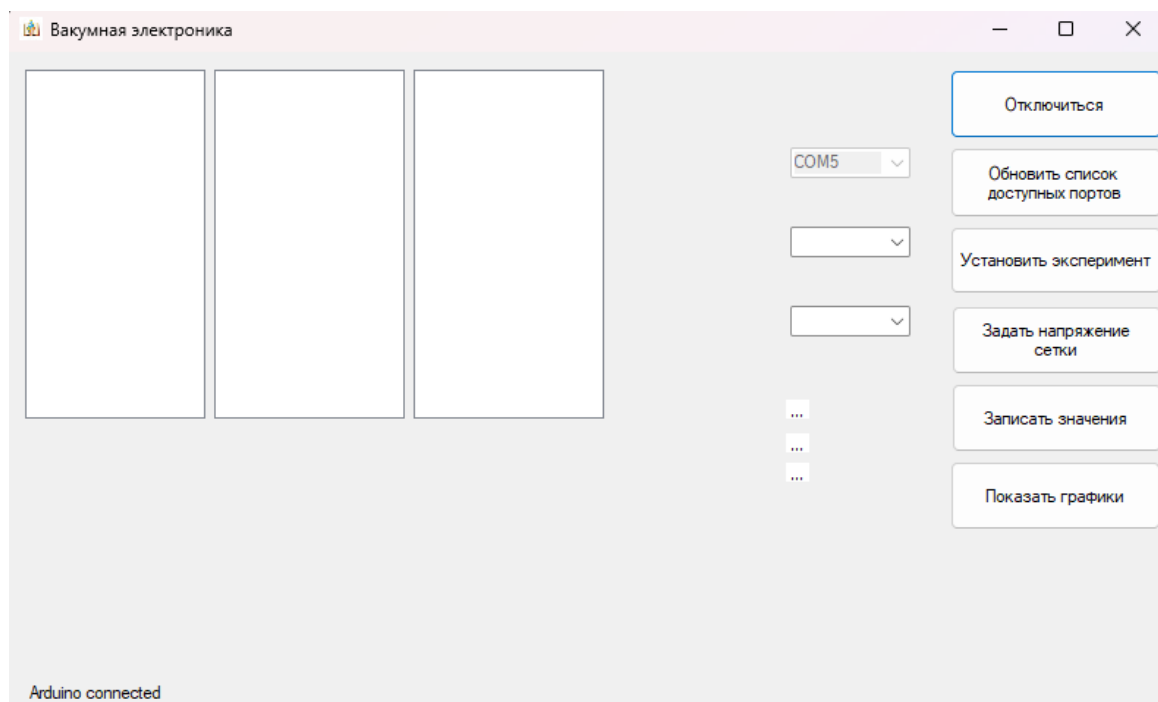


Рисунок 17 – Скриншот программы после выполнения алгоритма подключения

Для проверки алгоритма управления стендом необходимо в списке слева выбрать необходимый тип вакуумной лампы и нажать на кнопку “Установить эксперимент”. При нажатии на эту кнопку на микроконтроллер подаётся необходимый управляющий сигнал и на каналы, которые подключены к реле отвечающие за переключение вакуумных ламп, будет подан сигнал, соответствующий уровню логической единицы. Управление напряжением на сетке осуществляется подобным образом. В списке слева выбирается необходимое значение напряжения. Для передачи данного значения на микроконтроллер необходимо нажать на кнопку “Задать напряжение на сетке”. При нажатии на эту кнопку на микроконтроллер будет подан управляющий сигнал с порядковым номером коэффициента заполнения ШИМ сигнала, который соответствует значению напряжения поданного на сетку через ЦАП. Затем необходимый ШИМ сигнал подаётся на канал, подключенный к ЦАП. Скриншот программы после выбора вакуумной лампы и напряжения на сетке изображён на рисунке 18.

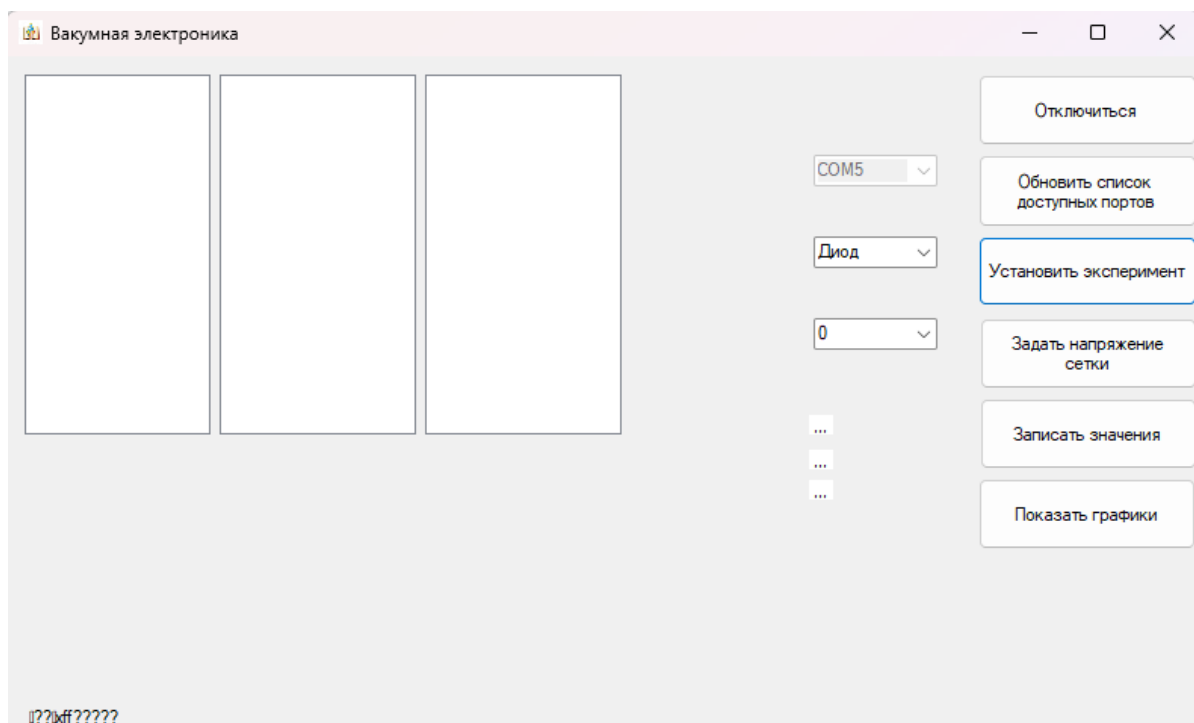


Рисунок 18 – Скриншот программы после выбора вакуумной лампы и напряжения на сетке

Для проверки алгоритма записи необходимо нажать на кнопку “Записать значения”. Если в роли измеряемого прибора был выбран “диод”, то при нажатии на эту кнопку слева от неё должны измениться значения токов. После проведения 10 итераций записи появится сообщение с текстом “Эксперимент закончен”, а в верхнем левом углу появится список значений токов. Если в роли измеряемого прибора была выбрана другая лампа, то необходимо выбрать значение напряжения на сетке равное 0, и через каждые 10 итераций записи менять это значение. Для получения изображения графиков и значений параметров необходимо нажать на кнопку “Показать графики”. Если в роли измеряемого прибора был выбран “диод”, то при нажатии на эту кнопку откроется новое окно, в котором будет изображена анодная ВАХ, а ниже будут графики параметров крутизны и внутреннего сопротивления. Скриншот программы после нажатия кнопки “Показать графики” для диодной вакуумной лампы изображён на рисунке 19.

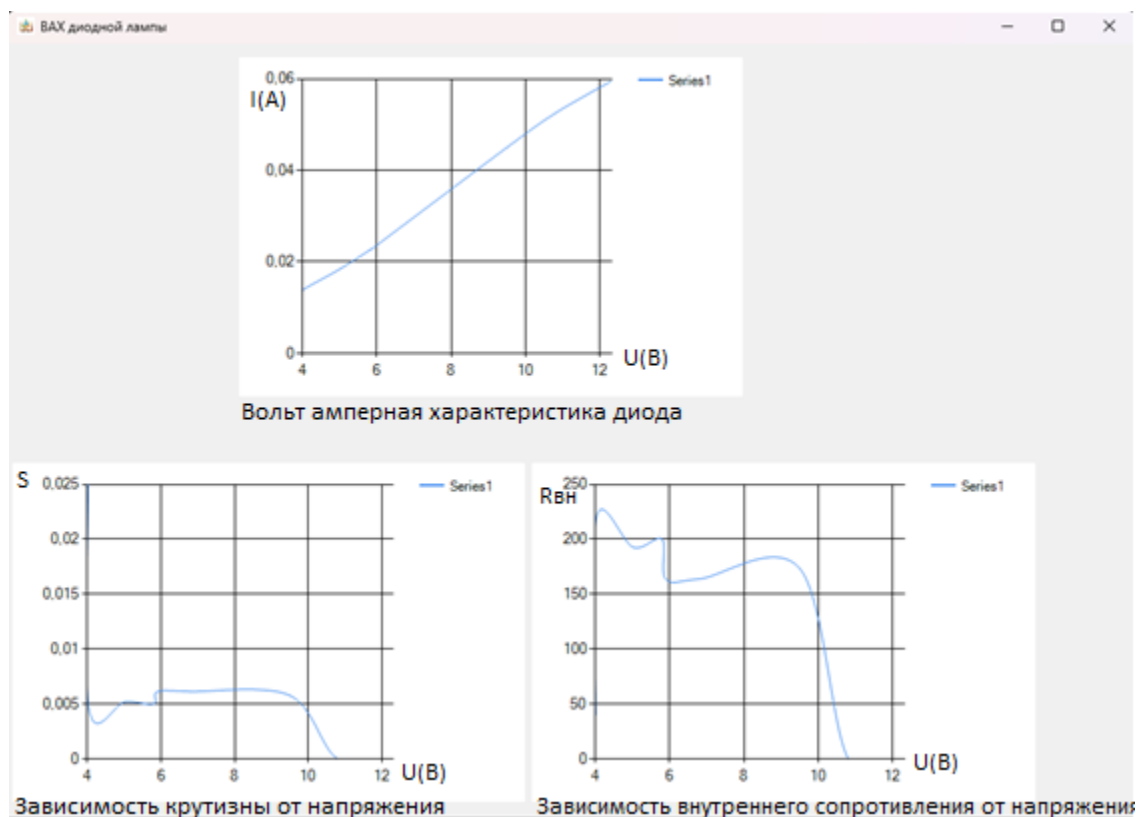


Рисунок 19 – Скриншот программы после нажатия кнопки “Показать графики” для диодной вакуумной лампы

Если в роли измеряемого прибора была выбрана другая лампа, то в новом окне появятся графики анодной и сеточной ВАХ. Под графиками будут отображены графики параметров крутизны и внутреннего сопротивления. Скриншот программы после нажатия кнопки “Показать графики” для триодной или пентодной вакуумной лампы изображен на рисунке 20.

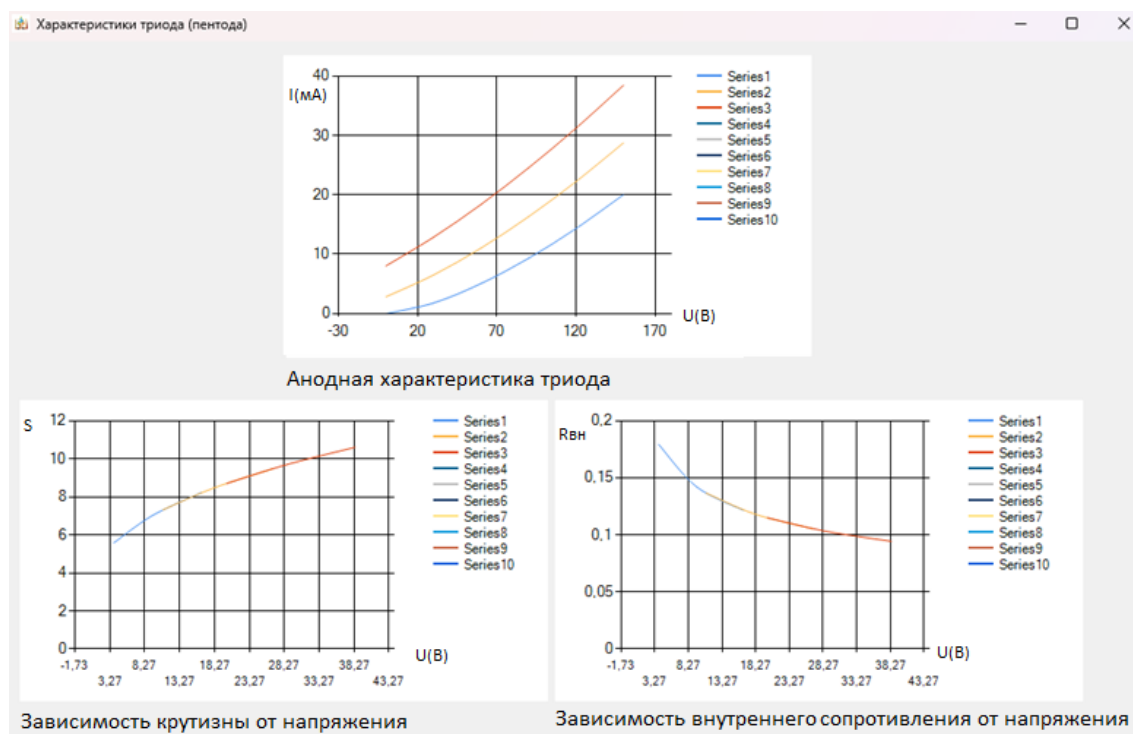


Рисунок 20 – Скриншот программы после нажатия кнопки “Показать графики” для триодной или пентодной вакуумной лампы.

4. Обеспечение безопасных условий труда при разработке учебного станда,

К работе на персональном компьютере допускаются лица, прошедшие обучение безопасным методам труда, вводный инструктаж, первичный инструктаж на рабочем месте. При эксплуатации персонального компьютера на человека могут оказывать действие следующие опасные и вредные производственные факторы:

- повышенный уровень электромагнитных излучений;
- повышенный уровень статического электричества;
- пониженная ионизация воздуха;
- статические физические перегрузки;
- перенапряжение зрительных анализаторов.

Работник обязан выполнять только ту работу, которая определена его должностной инструкцией, содержать в чистоте рабочее место, соблюдать режим труда и отдыха в зависимости от продолжительности, вида и категории трудовой деятельности и соблюдать меры пожарной безопасности.

Рабочие места с компьютерами должны размещаться таким образом, чтобы расстояние от экрана одного видеомонитора до тыла другого было не менее 2,0 м, а расстояние между боковыми поверхностями видеомониторов - не менее 1,2 м. Рабочие места с персональными компьютерами по отношению к световым проемам должны располагаться так, чтобы естественный свет падал сбоку, преимущественно слева. Оконные проемы в помещениях, где используются персональные компьютеры, должны быть оборудованы регулируемыми устройствами типа: жалюзи, занавесей, внешних козырьков и др. Рабочая мебель для пользователей компьютерной техникой должна отвечать следующим требованиям:

- высота рабочей поверхности стола должна регулироваться в пределах 680 - 800 мм; при отсутствии такой возможности высота рабочей поверхности стола должна составлять 725 мм;

					ПГУ.2.11.03.04.006ПЗ			
Изм.	Лист	№ докум.	Подпись	Дат	Обеспечение безопасных условий труда при разработке учебного станда	Лит.	Лист	Листов
Разраб.		Зубанов Д.В.						
Провер.		Вольсков А.А.					43	80
Н. Контр.		Рыжова Т.Н.		19.06		группа 19ПД		
Утверд.								

- рабочий стол должен иметь пространство для ног высотой не менее 600 мм, глубиной на уровне колен не менее 450 мм и на уровне вытянутых ног не менее 650 мм;

- рабочий стул (кресло) должен быть подъемно - поворотным и регулируемым по высоте и углам наклона сиденья и спинки, а также - расстоянию спинки от переднего края сиденья;

- рабочее место должно быть оборудовано подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах до 150 мм и по углу наклона опорной поверхности подставки до 20 градусов; поверхность подставки должна быть рифленой и иметь по переднему краю бортик высотой 10 мм;

- рабочее место с персональным компьютером должно быть оснащено легко перемещаемым пюпитром для документов.

Для нормализации аэроионного фактора помещений с компьютерами необходимо использовать устройства автоматического регулирования ионного режима воздушной среды. Женщины со времени установления беременности и в период кормления грудью к выполнению всех видов работ, связанных с использованием компьютеров, не допускаются.

Перед началом разработки программно-аппаратной части учебного стенда человек обязан подготовить рабочее место, отрегулировать освещение на рабочем месте, убедиться в отсутствии бликов на экране и проверить правильность подключения оборудования к электросети. Так же работник обязан проверить исправность проводов питания и отсутствие оголенных участков проводов, убедиться в наличии заземления системного блока, монитора и защитного экрана, протереть антистатической салфеткой поверхность экрана монитора и защитного экрана и проверить правильность установки стола, стула, подставки для ног, пюпитра, угла наклона экрана, положение клавиатуры, положение "мыши" на специальном коврик, при необходимости произвести регулировку рабочего стола и кресла, а также расположение элементов компьютера в соответствии с требованиями

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		44

эргономики и в целях исключения неудобных поз и длительных напряжений тела.

Во время выполнения лабораторной работы по разработке программно-аппаратной части стенда работнику запрещается:

- прикасаться к задней панели системного блока (процессора) при включенном питании;
- переключать разъемы интерфейсных кабелей периферийных устройств при включенном питании;
- допускать попадание влаги на поверхность системного блока (процессора), монитора, рабочую поверхность клавиатуры, дисководов, принтеров и других устройств;
- производить самостоятельное вскрытие и ремонт оборудования;
- работать на компьютере при снятых кожных;
- отключать оборудование от электросети и выдергивать электровилку, держа за шнур.

Продолжительность непрерывной работы с компьютером без регламентированного перерыва не должна превышать 2-х часов. Во время регламентированных перерывов с целью снижения нервно - эмоционального напряжения, утомления зрительного анализатора, устранения влияния гиподинамии и гипокинезии, предотвращения развития познотонического утомления выполнять комплексы упражнений.

Во всех случаях обрыва проводов питания, неисправности заземления и других повреждений, появления гари, немедленно отключить питание и сообщить об аварийной ситуации руководителю и не приступать к работе до устранения неисправностей. При окончании работ по разработке программно-аппаратной части стенда необходимо:

- отключить питание компьютера;
- привести в порядок рабочее место.
- выполнить упражнения для глаз и пальцев рук на расслабление.

ЗАКЛЮЧЕНИЕ

Электронная лампа, радиолампа представляет собой электровакуумный прибор (точнее, вакуумный электронный прибор), работающий за счёт управления интенсивностью потока электронов, движущихся в вакууме или разрежённом газе между электродами.

Лампы используются для получения высококачественный звук аппаратуры класса *Hi-End*, где все строится принципиально только на радиолампах. Многие зарубежные производители усилителей используют некоторые типы ламп исключительно производства России.

Электровакуумные приборы находят незаменимое применение в спутниковых передатчиках, в самолетах, на кораблях и в центрах связи на Земле. Только электровакуумные приборы способны обеспечить сверхвысокие частоты с высокой стабильностью и огромными мощностями, транзисторы этого не умеют.

В ходе выполнения ВКР была разработана программно-аппаратная часть стенда по вакуумной электронике. Было разработано приложение с интуитивно понятным интерфейсом для получения ВАХ, а так же проведения расчётов параметров вакуумных ламп.

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		46

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1) <https://infopedia.su/13x40bc.html>
- 2) https://robotclass.ru/tutorials/arduino_adc/
- 3) <https://github.com/SpenceKonde/ATTinyCore/issues/673>
- 4) <https://visualstudio.microsoft.com/>
- 5) <https://processing.org/>
- 6) <https://docs.arduino.cc/learn/starting-guide/the-arduino-software-ide>
- 7) <https://www.microchip.com/en-us/tools-resources/develop/microchip-studio>
- 8) Преобразователи частоты — Элементная база и схемотехника устройств силовой электроники.
- 9) Сабанин В. Р. Автоматические системы регулирования на основе нейросетевых технологий / В. Р. Сабанин, Н. И. Смирнов, А. И. Репин // Труды Международной научной конференции Control-2003. М.: Издательство МЭИ, 2003. С. 45—51.
- 10) <http://www.falstad.com/circuit/>

Приложение А

(обязательное)

Техническое задание на бакалаврскую работу

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		48

«Утверждаю»

Зав. кафедрой НиМЭ

Пронин И.А.

(фамилия, инициалы)

«09» июня 2023 г.

З А Д А Н И Е
на бакалаврскую работу

1. Студент(ка) гр. 19ПД1 факультета информационных технологий и электроники

Направление подготовки 11.03.04. «Электроника и нанoeлектроника»

Профиль Микроэлектроника и твердотельная электроника

Зубанов Данила Владимирович

(фамилия, имя, отчество)

2. Руководитель Вольсков Андрей Александрович

(фамилия, имя, отчество)

3. Тема бакалаврской работы Разработка программно-аппаратной части лабораторного стенда по вакуумной электронике

Тема утверждена приказом ПГУ № 359а/о от « 4 » апреля 2023 г.

4. Техническое задание на бакалаврскую работу (вид работы: разработка, проектирование, исследование, и т.д.; назначение; основные параметры; условия функционирования и т.д.)

Разработать аппаратную часть лабораторного стенда по вакуумной электронике. Разработать программное обеспечение лабораторного стенда по вакуумной электронике. Автоматизированное управление лабораторным стендом. Программное обеспечение должно работать в среде Windows и обеспечивает взаимодействие стенда с пользователем через ПК. Обеспечивает построение характеристик и расчёт параметров. Питание через сеть 220 В 50 Гц.

5. Объем и содержание основной части бакалаврской работы

5.1. Пояснительная записка (перечень вопросов, подлежащих разработке, обоснований, описаний, расчетов и т.д.)

1. Выбор программно-аппаратных средств для реализации проекта

1.1 Выбор микроконтроллера

1.2 Выбор программного обеспечения для создания приложения

1.3 Выбор программы для прошивки микроконтроллера

2. Разработка алгоритмов

2.1 Разработка алгоритма приёма данных стенда

2.2 Разработка алгоритма управления напряжением на сетке при помощи ШИМ сигнала

2.3 Разработка алгоритма управления стендом

2.4 Алгоритм обработки результатов эксперимента

3 Изготовление макета учебного стенда

3.1 Изготовление аппаратной части учебного стенда

3.2 Описание программной части стенда

3.3 Проверка работоспособности стенда

4. Обеспечение безопасных условий труда при разработке учебного стенда

5.2. Иллюстративная часть (перечень и содержание чертежей и слайдов презентации)

1. Титульный лист (1 слайд)

2. Задание на ВКР (1 слайд)

3. Алгоритм работы стенда (6 слайдов)

4 Выбор программно-аппаратных средств для реализации проекта (3 слайда)

5. Главное окно программы (1 слайд)

6. Алгоритм подключения (1 слайд)

7. Алгоритм управления стендом (1 слайд)

8. Алгоритм управления напряжением на сетке (2 слайда)

9. Алгоритм приёма данных (1 слайд)

10 Результат работы программы (2 слайда)

11 Заключение (1 слайд)

6. Консультанты и содержание дополнительных разделов

6.1. _____

Консультант _____

7. Календарный график по выполнению бакалаврской работы

Наименование этапов работы	Объем работы	Срок выполнения	Подпись руководителя, консультанта
Выбор программно-аппаратных средств для реализации проекта	10%	10.06.2023	<u>Авану</u>
Разработка алгоритмов	10%	11.06.2023	<u>Авану</u>
Изготовление макета учебного стенда	50%	14.06.2023	<u>Авану</u>
Обеспечение безопасных условий труда при разработке учебного стенда	5%	14.06.2023	<u>Авану</u>
Иллюстративная часть	10%	14.06.2023	<u>Авану</u>
Оформление пояснительной записки	15%	14.06.2023	<u>Авану</u>

Руководитель бакалаврской работы

Авану
подпись

Вольсков А.А.
фамилия, инициалы

Дата выдачи задания

«09» июня 2023 г.

Задание к исполнению принял

Зубанов Д.В.
подпись

Зубанов Д.В.
фамилия, инициалы

«09» июня 2023 г.

Бакалаврскую работу к защите допустить

Декан факультета

Кревчик В.Д.
подпись

Кревчик В.Д.
фамилия, инициалы

«25» июня 2023 г.

Приложение Б
(обязательное)
Алгоритм работы стенда

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.№	Подпись	Дат		525

Алгоритмы проведения экспериментов:

1. Исследование характеристик диода:

1. Замкнуть первое реле и подать анодное напряжения на диод;
2. Установить напряжение на блоке питания анода равное 3В;
3. Нажать на кнопку «Старт измерений»;
4. Замкнуть реле *AREF3*;
5. Измерить анодное напряжение;
6. Разомкнуть реле *AREF3*;
7. Замкнуть реле *AREF1*;
8. Измерить напряжение на *R1*;
9. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
10. Разомкнуть реле *AREF1*;
11. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
12. Повторить пункты 2-11 для напряжений на блоке питания анода в диапазоне от 4 до 12В с шагом 1В;
13. Сохранить результаты в виде *excel*;
14. Установить минимальное напряжение на блоке питания анода;
15. Разомкнуть все реле;

2. Исследование характеристик триода:

1. Замкнуть второе реле и подать анодное напряжение на триод;
2. Замкнуть четвёртое реле и подать сеточное напряжение на триод;
3. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 0\text{В}$;
4. Выставить длительность ШИМ сигнала блока питания сетки = 1%, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
5. Установить напряжение на блоке питания анода равное 3В;
6. Нажать на кнопку «Старт измерений»;
7. Замкнуть реле *AREF3*;
8. Измерить анодное напряжение;

9. Разомкнуть реле *AREF3*;
10. Замкнуть реле *AREF1*;
11. Измерить напряжение на *R1*;
12. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
13. Разомкнуть реле *AREF1*;
14. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
15. Повторить пункты 5-14 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;
16. Сохранить результаты в виде *excel*;
17. Установить минимальное напряжение на блоке питания анода;
18. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 0,5\text{В}$;
19. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0,5\text{В}$;
20. Установить напряжение на блоке питания анода равное 3В;
21. Нажать на кнопку «Старт измерений»;
22. Замкнуть реле *AREF3*;
23. Измерить анодное напряжение;
24. Разомкнуть реле *AREF3*;
25. Замкнуть реле *AREF1*;
26. Измерить напряжение на *R1*;
27. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
28. Разомкнуть реле *AREF1*;
29. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
30. Повторить пункты 20-29 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;
31. Сохранить результаты в виде *excel*;
32. Установить минимальное напряжение на блоке питания анода;

33. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 1\text{В}$;
34. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 1\text{В}$;
35. Установить напряжение на блоке питания анода равное 3В ;
36. Нажать на кнопку «Старт измерений»;
37. Замкнуть реле $AREF3$;
38. Измерить анодное напряжение;
39. Разомкнуть реле $AREF3$;
40. Замкнуть реле $AREF1$;
41. Измерить напряжение на $R1$;
42. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
43. Разомкнуть реле $AREF1$;
44. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
45. Повторить пункты 35-44 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В ;
46. Сохранить результаты в виде *excel*;
47. Установить минимальное напряжение на блоке питания анода;
48. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
49. Выбрать режим замера анодно сеточной характеристики;
50. Установить напряжение на блоке питания анода равное 60В ;
51. Замкнуть реле $AREF3$;
52. Измерить анодное напряжение;
53. Разомкнуть реле $AREF3$;
54. Сохранить результат измерения анодного напряжения;
55. Выставить длительность ШИМ сигнала блока питания сетки $= 1\%$;
56. Замкнуть реле $AREF1$;
57. Измерить напряжение на $R1$;
58. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
59. Разомкнуть реле $AREF1$;

60. Сохранить результат расчёта тока анода;
 61. Повторить пункты 55-60 для длительности ШИМ сигнала блока питания сетки от 2% до 100% с шагом 1%;
 62. Сохранить результаты в виде *excel*;
 63. Установить минимальное напряжение на блоке питания анода;
 64. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
 65. Выбрать режим замера сеточной характеристики;
 66. Установить напряжение на блоке питания анода равное 60В;
 67. Замкнуть реле *AREF3*;
 68. Измерить анодное напряжение;
 69. Разомкнуть реле *AREF3*;
 70. Сохранить результат измерения анодного напряжения;
 71. Выставить длительность ШИМ сигнала блока питания сетки = 1%;
 72. Замкнуть реле *AREF2*;
 73. Измерить напряжение на *R2*;
 74. Рассчитать по закону Ома ток сетки, протекающий через $R2=5\text{кОм}$;
 75. Разомкнуть реле *AREF2*;
 76. Сохранить результат расчёта тока сетки;
 77. Повторить пункты 71-76 для длительности ШИМ сигнала блока питания сетки от 2% до 100% с шагом 1%;
 78. Сохранить результаты в виде *excel*;
 79. Установить минимальное напряжение на блоке питания анода;
 80. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
 81. Разомкнуть все реле;
3. Исследование характеристик пентода:
1. Замкнуть третье реле и подать анодное напряжение на пентод;
 2. Замкнуть пятое реле и подать сеточное напряжение на пентод;
 3. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 0\text{В}$;

4. Выставить длительность ШИМ сигнала блока питания сетки = 1%, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
5. Установить напряжение на блоке питания анода равное 3В;
6. Нажать на кнопку «Старт измерений»;
7. Замкнуть реле $AREF3$;
8. Измерить анодное напряжение;
9. Разомкнуть реле $AREF3$;
10. Замкнуть реле $AREF1$;
11. Измерить напряжение на $R1$;
12. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
13. Разомкнуть реле $AREF1$;
14. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
15. Повторить пункты 5-14 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;
16. Сохранить результаты в виде *excel*;
17. Установить минимальное напряжение на блоке питания анода;
18. Выбрать режим замера VAX при $U_{\text{сетки}} = 0,5\text{В}$;
19. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0,5\text{В}$;
20. Установить напряжение на блоке питания анода равное 3В;
21. Нажать на кнопку «Старт измерений»;
22. Замкнуть реле $AREF3$;
23. Измерить анодное напряжение;
24. Разомкнуть реле $AREF3$;
25. Замкнуть реле $AREF1$;
26. Измерить напряжение на $R1$;
27. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
28. Разомкнуть реле $AREF1$;

29. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
30. Повторить пункты 20-29 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;
31. Сохранить результаты в виде *excel*;
32. Установить минимальное напряжение на блоке питания анода;
33. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 1\text{В}$;
34. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 1\text{В}$;
35. Установить напряжение на блоке питания анода равное 3В;
36. Нажать на кнопку «Старт измерений»;
37. Замкнуть реле *AREF3*;
38. Измерить анодное напряжение;
39. Разомкнуть реле *AREF3*;
40. Замкнуть реле *AREF1*;
41. Измерить напряжение на *R1*;
42. Рассчитать по закону Ома ток анода, протекающий через $R1 = 20\text{ Ом}$;
43. Разомкнуть реле *AREF1*;
44. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
45. Повторить пункты 35-44 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;
46. Сохранить результаты в виде *excel*;
47. Установить минимальное напряжение на блоке питания анода;
48. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
49. Выбрать режим замера анодно сеточной характеристики;
50. Установить напряжение на блоке питания анода равное 60В;
51. Замкнуть реле *AREF3*;
52. Измерить анодное напряжение;
53. Разомкнуть реле *AREF3*;

54. Сохранить результат измерения анодного напряжения;
55. Выставить длительность ШИМ сигнала блока питания сетки = 1%;
56. Замкнуть реле *AREF1*;
57. Измерить напряжение на *R1*;
58. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
59. Разомкнуть реле *AREF1*;
60. Сохранить результат расчёта тока анода;
61. Повторить пункты 55-60 для длительности ШИМ сигнала блока питания сетки от 2% до 100% с шагом 1%;
62. Сохранить результаты в виде *excel*;
63. Установить минимальное напряжение на блоке питания анода;
64. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
65. Выбрать режим замера сеточной характеристики;
66. Установить напряжение на блоке питания анода равное 60В;
67. Замкнуть реле *AREF3*;
68. Измерить анодное напряжение;
69. Разомкнуть реле *AREF3*;
70. Сохранить результат измерения анодного напряжения;
71. Выставить длительность ШИМ сигнала блока питания сетки = 1%;
72. Замкнуть реле *AREF2*;
73. Измерить напряжение на *R2*;
74. Рассчитать по закону Ома ток сетки, протекающий через $R2=5\text{кОм}$;
75. Разомкнуть реле *AREF2*;
76. Сохранить результат расчёта тока сетки;
77. Повторить пункты 71-76 для длительности ШИМ сигнала блока питания сетки от 2% до 100% с шагом 1%;
78. Сохранить результаты в виде *excel*;
79. Установить минимальное напряжение на блоке питания анода;
80. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
81. Разомкнуть все реле.

Приложение В
(обязательное)
Код программы для программирования микроконтроллера

					ПГУ.2.11.03.04.006ПЗ	Лист
						60
Изм.	Лист	№ докум.	Подпись	Дат		

```

//настройка
#define VDA0 //пин подключения точки с которой снимается анодный ток
#define VSA1 //пин подключения точки измерения сетки. Коэффициент
умножения = 21,9998(24,297992 )
#define VP 4.6 //напряжение питания. Необходимо точное напряжение
питания
#define MTS 1000 //Множитель тока сетки
#define MTD 1000 //множитель тока диода(A - 1,мА - 1000)
#define RARD 19.8 //Соппротивление анодного резистора диода в омах
#define RARS 5050 //Соппротивление сеточного резистора
#define PD 2 //Номер пина к которому подключены реле для диодной
вакуумной лампы
#define PT 4 //Номер пина к которому подключены реле для триодной
вакуумной лампы
#define PP 5 //Номер пина к которому подключены реле для пентодной
вакуумной лампы
#define PS 3 //Номер пина с ШИМ сигналом
#define AREF1 6 //СЕТКА
#define AREF2 7 // АНОД
#define AREF3 8 //
#define PV2 9 //ПИН ВОЛЬТМЕТРА 2
#define PV3 10 //пинвольтметра 3
#define AN A2 // пинанодноговольтметра
#define KDAND 24.297992 //коэффициентделения анодного напряжения диода
#define KDANT 24.297992 //коэффициент деления анодного напряжения триода
#define KDATD 24.297992 //коэффициентделения анодного тока диода
#define KDATT 24.297992 //коэффициент деления анодного тока триода

```

```

uint32_t ttmr1; // переменная таймера
uint32_t ttmr2; // переменная таймера
uint32_t ttmr3; // переменная таймера
uint32_t ttmr4; // переменная таймера
uint32_t ttmr5; // переменная таймера
double V0 = 0;
double V1 = 0;
int V0S;
int V1S;
double VDS;
double VDSR;
double KDT;
float SIM[] = { 0, 177, 184, 188, 191, 194, 198, 201, 206, 255 };
float SIM2[9];
char OPR;
int NE;

```

```

int NAREF;
inti;
int j;
int KI = 0;
int SIMN = 0;
double NAPR;
void setup() {
    uint32_t tmr1; // переменная таймера
    pinMode(PT, OUTPUT);
    pinMode(PD, OUTPUT);
    pinMode(PV2, OUTPUT);
    pinMode(PV3, OUTPUT);
    pinMode(PS, OUTPUT);
    pinMode(A0, INPUT);
    pinMode(A1, INPUT);
    pinMode(A2, INPUT);
    pinMode(AREF1, OUTPUT);
    pinMode(AREF2, OUTPUT);
    pinMode(AREF3, OUTPUT);
    Serial.begin(9600);
    //analogReference(EXTERNAL);
}
void loop() {
    if (Serial.available()) {
        OPR = (char)Serial.read();
    }
    if (OPR == 'd') {
        NE = 1;
        KDT = KDATD;
    }
    if (OPR == 't') {
        NE = 2;
        KDT = KDATT;
    }
    if (OPR == 'p') {
        NE = 3;
    }
    if (OPR == 'g') {
        NE = 4;
    }
    if (OPR == 'c') {
        Serial.println("Arduino connected");
        OPR = 'J';
    }
}

```

```

if (OPR == 'z') {
  Serial.println(analogRead(VD));
  Serial.println(analogRead(VS));
  OPR == 'm';
  delay(1);
}
if (OPR == '1') {
  SIMN = 1;
}
if (OPR == '2') {
  SIMN = 2;
}
if (OPR == '3') {
  SIMN = 3;
}
if (OPR == '4') {
  SIMN = 4;
}
if (OPR == '5') {
  SIMN = 5;
}
if (OPR == '6') {
  SIMN = 6;
}
if (OPR == '7') {
  SIMN = 7;
}
if (OPR == '8') {
  SIMN = 8;
}
if (OPR == '9') {
  SIMN = 9;
}
if (OPR == '0') {
  SIMN = 0;
}
switch (NE) {

  //if (NE == 1)
  case 1:{
    digitalWrite(PD, HIGH);
    digitalWrite(PT, LOW);
    digitalWrite(PP, LOW);
  }

```

```
break;
    //if (NE == 2)
case 2:{
digitalWrite(PD, LOW);
digitalWrite(PT, HIGH);
digitalWrite(PP, LOW);
    }
break;
    //if (NE == 3)
case 3:{
digitalWrite(PD, LOW);
digitalWrite(PT, LOW);
digitalWrite(PP, HIGH);
    }
break;
    //if (NE == 4)
case 4:{
digitalWrite(PD, LOW);
digitalWrite(PT, LOW);
digitalWrite(PP, LOW);
    }
break;
    }
analogWrite(PS, SIM[SIMN]);
    // Serial.println(analogRead(VD));
}
```

ПриложениеГ
(обязательное)
Код приложения для учебного стенда

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		65

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.IO.Ports;

namespace Diplom2
{
    public partial class Form1 : Form
    {
        int i = 0;
        int j = 0;
        double[] naprIz1 = new double[10];
        double[] naprIzA = new double[10];
        double[] naprIz2 = new double[10];
        double[,] Napr = new double[10, 10];
        double[,] NaprS = new double[10, 10];
        double[,] NaprA = new double[10, 10];
        string[] naprIz11 = new string[10];
        string[] naprIzA1 = new string[10];
        string[] S1 = new string[10];
        string[] Rvh1 = new string[10];
        string[] naprIz21 = new string[10];
        string[] napr = new string[] { "0", "0.5", "1", "1.5", "2", "2.5", "3", "3.5", "4", "4.5" };
    };

    double[] napr1 = new double[] { 0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 };
    double[] napr2 = new double[] { 0, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5 };
    string[] ExpNames = new string[] { "Диод", "Триод", "Пентод", "Отключить" };
    double Rvh;
    double ON = 4.6; //опорное напряжение АЦП
    int naprIzS1 = 0;
    int n = 0;
    int NS;
    int g;
    int si;
    int sih;
    double NA;
    double NAA;
    double NAS;

```

```

double KDA;
double KDS;
double KDN;
double[] Sd = new double[9];
double[,] S = new double[9,9];
double[] Rvnd = new double[9];
double[,] Rvn = new double[9,9];
double SA = 19.8;
double SS = 5050;
double summ11;
double summ12;
double summ13;
string naprS1;
string naprS2;
int naprIzS2 = 0;
public Form1()
{
    InitializeComponent();
}

```

```

private void buttonUpdatePorts_Click(object sender, EventArgs e)
{
    string[] ports = SerialPort.GetPortNames();
    comboBoxPortNames.Text = "";
    comboBoxPortNames.Items.Clear();
    comboBoxSIM.Text = "";
    comboBoxSIM.Items.Clear();
    comboBoxSIM.Items.AddRange(napr);
    comboBoxExpNames.Text = "";
    comboBoxExpNames.Items.Clear();
    comboBoxExpNames.Items.AddRange(ExpNames);
    if (ports.Length != 0)
    {
        comboBoxPortNames.Items.AddRange(ports);
        comboBoxPortNames.SelectedIndex = 0;
    }
}

```

```

private void buttonConnect_Click(object sender, EventArgs e)
{
    if (buttonConnect.Text == "Подключиться")
    {

```

```

    try
    {
        serialPort1.PortName = comboBoxPortNames.Text;
        serialPort1.Open();
        buttonConnect.Text = "Отключиться";
        comboBoxPortNames.Enabled = false;
        serialPort1.Write("c");
        label1.Text = serialPort1.ReadLine();
    }
    catch
    {
        MessageBox.Show("Ошибка подключения");
    }
}
else if (buttonConnect.Text == "Отключиться")
{
    try
    {
        serialPort1.Close();
        buttonConnect.Text = "Подключиться";
        comboBoxPortNames.Enabled = true;
    }
    catch
    {
        MessageBox.Show("Ошибка отключения");
    }
}

}
private void buttonExp_Click(object sender, EventArgs e)
{
    if (comboBoxExpNames.Text == "Диод")
    {
        serialPort1.Write("d");
        KDA = 24.298;
        KDS = 23.568;
        KDN = 25.265;
        i = 0;
        j = 0;
    }
    if (comboBoxExpNames.Text == "Триод")
    {
        serialPort1.Write("t");
    }
}

```

```

        KDA = 24.298;
        KDS = 23.568;
        KDN = 25.265;
        i = 0;
        j = 0;
    }
    if (comboBoxExpNames.Text == "Пентод")
    {
        serialPort1.Write("p");
        KDA = 24.298;
        KDS = 23.568;
        KDN = 25.265;
        i = 0;
        j = 0;
    }
    if (comboBoxExpNames.Text == "Отключить")
    {
        serialPort1.Write("g");
        KDA = 24.298;
        KDS = 23.568;
        KDN = 25.265;
        i = 0;
        j = 0;
    }
}

private void comboBoxSIM_SelectedIndexChanged(object sender, EventArgs e)
{

}

private void buttonSIM_Click(object sender, EventArgs e)
{
    if (comboBoxSIM.Text == "0")
    {
        serialPort1.Write("0");
        serialPort1.Write("0");
        serialPort1.Write("0");
        serialPort1.Write("0");
        NS = 0;
    }
    if (comboBoxSIM.Text == "0.5")
    {
        serialPort1.Write("1");
    }
}

```

```
    serialPort1.Write("1");
    serialPort1.Write("1");
    serialPort1.Write("1");
    serialPort1.Write("1");
    NS = 1;
}
if (comboBoxSIM.Text == "1")
{
    serialPort1.Write("2");
    NS = 2;
}
if (comboBoxSIM.Text == "1.5")
{
    serialPort1.Write("3");
    NS = 3;
}
if (comboBoxSIM.Text == "2")
{
    serialPort1.Write("4");
    NS = 4;
}
if (comboBoxSIM.Text == "2.5")
{
    serialPort1.Write("5");
    serialPort1.Write("5");
    serialPort1.Write("5");
    serialPort1.Write("5");
    serialPort1.Write("5");
    NS = 5;
}
if (comboBoxSIM.Text == "3")
{
    serialPort1.Write("6");
    NS = 6;
}
if (comboBoxSIM.Text == "3.5")
{
    serialPort1.Write("7");
    NS = 7;
}
if (comboBoxSIM.Text == "4")
{
    serialPort1.Write("8");
    NS = 8;
}
```

```

    }
    if (comboBoxSIM.Text == "4.5")
    {
        serialPort1.Write("9");
        NS = 9;
    }
}

```

```

private void buttonGraph_Click(object sender, EventArgs e)
{

```

```

    Form2 newForm = new Form2();
    Form3 newForm1 = new Form3();
    if (comboBoxExpNames.Text == "Диод")
    {
        newForm1.Show();
        do
        {
            newForm1.chart1.Series[0].Points.AddXY(naprIzA[n], naprIz1[n]);
            newForm1.chart2.Series[0].Points.AddXY(naprIzA[n], Sd[n]);
            newForm1.chart3.Series[0].Points.AddXY(naprIzA[n], Rvnd[n]);
            n++;
        } while (n < 9);

```

```

        n = 0;
    }
    else
    {
        newForm.Show();
        do
        {
            do
            {
                newForm.chartAnod.Series[g].Points.AddXY(NaprA[g, n], Napr[g,n]) ;
                newForm.chart1.Series[g].Points.AddXY(NaprA[g, n], NaprS[g,n]);
                newForm.chart2.Series[g].Points.AddXY(NaprA[g,n], S[g,n]);
                newForm.chart3.Series[g].Points.AddXY(NaprA[g, n], Rvn[g,n]);
                n++;
            } while (n < 9);
            n = 0;
            g++;
        }
        while (g < 9);
    }
}

```

```
    g = 0;  
}
```

```
}
```

```
private void button1_Click_1(object sender, EventArgs e)  
{  
    serialPort1.Write("z");  
    if (i < 10)  
    {  
        if (comboBoxExpNames.Text == "Диод")  
        {  
            do  
            {  
                label6.Text = serialPort1.ReadLine();  
                label2.Text = serialPort1.ReadLine();  
                //label3.Text = serialPort1.ReadLine();  
                string Ua = label6.Text;  
                string Ia = label2.Text;  
                //string Is = label3.Text;  
                int summ1 = Convert.ToInt32(Ia);  
                //int summ2 = Convert.ToInt32(Is);  
                int summ3 = Convert.ToInt32(Ua);  
                summ11 = summ11 + Convert.ToDouble(summ1);  
                //summ12 = summ12 + Convert.ToDouble(summ2);  
                summ13 = summ13 + Convert.ToDouble(summ3);  
  
                System.Threading.Thread.Sleep(10);  
                j++;  
            } while (j < 100);  
        }  
        else  
        {  
            label7.Text = serialPort1.ReadLine();  
            label2.Text = serialPort1.ReadLine();  
            label3.Text = serialPort1.ReadLine();  
            string Ua = label7.Text;  
            string Ia = label2.Text;  
            string Is = label3.Text;
```

```

        int summ1 = Convert.ToInt32(Ia);
        int summ2 = Convert.ToInt32(Is);
        int summ3 = Convert.ToInt32(Ua);
        summ11 = summ11 + Convert.ToDouble(summ1);
        summ12 = summ12 + Convert.ToDouble(summ2);
        summ13 = summ13 + Convert.ToDouble(summ3);
    }
    NAA = Convert.ToDouble(summ11 / (j + 1) / 1023) * ON * KDA / SA *
1000;
    NAS = Convert.ToDouble(summ12 / (j + 1) / 1023) * ON * KDS / SS * 1000;
    NA = Convert.ToDouble(summ12 / (j + 1) / 1023) * ON * KDN;
    label4.Text = Convert.ToString(NAA);
    label5.Text = Convert.ToString(NAS);
    label6.Text = Convert.ToString(NA);
    naprIz1[i] = Convert.ToDouble(NAA);
    naprIz11[i] = Convert.ToString(NAA) + " mA";
    naprIzS2 = Convert.ToInt32(NAS);
    naprIz2[i] = Convert.ToDouble(NAS);
    naprIzA[i] = Convert.ToDouble(NA);
    naprIzA1[i] = Convert.ToString(NA) + " B";

    Napr[NS, i] = NAA;
    NaprS[NS,i] = NAS;
    NaprA[NS, i] = NA;
    naprIz21[i] = Convert.ToString(NAS) + " mA";
    summ11 = 0;
    summ12 = 0;
    summ13 = 0;
    if (comboBoxExpNames.Text == "Диод")
    {
        do
        {
            Sd[si] = (naprIzA[si+1] - naprIzA[si]) / (naprIz2[si+1] - naprIz2[si]);
            Rvnd[si] = 1 / Sd[si];
        }
        while (si < 9);
    }
    else
    {
        do
        {
            do
            {

```

```

        S[sih, si] = (Napra[sih,si+1] - Napra[sih, si]) / (Napra[sih, si+1] -
Napra[sih, si]);
        Rvn[sih, si] = 1 / S[sih, si];
    } while (si < 9);
} while (sih < 9);
}
j = 0;
i = i + 1;
si = 0;
serialPort1.Write("m");
}
else
{
    listBox1.Text = "";
    listBox1.Items.Clear();
    listBox1.Items.AddRange(napriz11);
    listBox2.Text = "";
    listBox2.Items.Clear();
    listBox2.Items.AddRange(napriz21);
    listBox3.Text = "";
    listBox3.Items.Clear();
    listBox3.Items.AddRange(naprizA1);
    i = 0;
    serialPort1.Write("m");
    MessageBox.Show("Эксперимент закончен");
}
serialPort1.Write("m");

}

private void buttonSIM_Click_1(object sender, EventArgs e)
{
    if (comboBoxSIM.Text == "0")
    {
        serialPort1.Write("0");
        serialPort1.Write("0");
        serialPort1.Write("0");
        serialPort1.Write("0");
        NS = 0;
    }
    if (comboBoxSIM.Text == "0.5")
    {
        serialPort1.Write("1");
        serialPort1.Write("1");
    }
}

```

```

        serialPort1.Write("1");
        serialPort1.Write("1");
        serialPort1.Write("1");
        NS = 1;
    }
    if (comboBoxSIM.Text == "1")
    {
        serialPort1.Write("2");
        serialPort1.Write("2");
        serialPort1.Write("2");
        serialPort1.Write("2");
        serialPort1.Write("2");
        NS = 2;
    }
    if (comboBoxSIM.Text == "1.5")
    {
        serialPort1.Write("3");
        serialPort1.Write("3");
        serialPort1.Write("3");
        serialPort1.Write("3");
        serialPort1.Write("3");
        NS = 3;
    }
    if (comboBoxSIM.Text == "2")
    {
        serialPort1.Write("4");
        serialPort1.Write("4");
        serialPort1.Write("4");
        serialPort1.Write("4");
        serialPort1.Write("4");
        NS = 4;
    }
    if (comboBoxSIM.Text == "2.5")
    {
        serialPort1.Write("5");
        serialPort1.Write("5");
        serialPort1.Write("5");
        serialPort1.Write("5");
        serialPort1.Write("5");
        NS = 5;
    }
    if (comboBoxSIM.Text == "3")
    {
        serialPort1.Write("6");
    }

```

```

        serialPort1.Write("6");
        serialPort1.Write("6");
        serialPort1.Write("6");
        serialPort1.Write("6");
        NS = 6;
    }
    if (comboBoxSIM.Text == "3.5")
    {
        serialPort1.Write("7");
        serialPort1.Write("7");
        serialPort1.Write("7");
        serialPort1.Write("7");
        serialPort1.Write("7");
        NS = 7;
    }
    if (comboBoxSIM.Text == "4")
    {
        serialPort1.Write("8");
        serialPort1.Write("8");
        serialPort1.Write("8");
        serialPort1.Write("8");
        serialPort1.Write("8");
        NS = 8;
    }
    if (comboBoxSIM.Text == "4.5")
    {
        serialPort1.Write("9");
        serialPort1.Write("9");
        serialPort1.Write("9");
        serialPort1.Write("9");
        serialPort1.Write("9");
        NS = 9;
    }
}
}

```

ПриложениеД
(обязательное)

Заявление и протокол проверки ВКР на оригинальность в системе «Антиплагиат.
ВУЗ»

					ПГУ.2.11.03.04.006ПЗ	Лист
						77
Изм.	Лист	№ докум.	Подпись	Дат		

Ректору Пензенского
государственного университета
А. Д. Гулякову
студента 4 курса ФИТЭ
очной формы обучения
Зубанов Данила Владимирович

ЗАЯВЛЕНИЕ

Я, Зубанов Данила Владимирович, студент 4 курса очной формы обучения направления подготовки 11.03.04 Электроника и наноэлектроника. Микроэлектроника и твердотельная электроника, прошу провести проверку с использованием системы «Антиплагиат» бакалаврской выпускной квалификационной работы на тему «Разработка программно-аппаратной части лабораторного стенда по вакуумной электронике», выполненной мной самостоятельно, на содержание элементов плагиата.

Все прямые заимствования из печатных и электронных источников, а также из защищенных ранее выпускных квалификационных работ, кандидатских и докторских диссертаций имеют соответствующие ссылки.

Я ознакомлен с действующим Стандартом университета «Выпускная квалификационная работа обучающихся по образовательным программам высшего образования – программам бакалавриата, программам специалитета и программам магистратуры», согласно которому обнаружение плагиата является основанием для недопуска ВКР к защите и отчисления из университета.

подпись студента

дата

25.06.2023

ВКР представлена на проверку

25.06.2023

Руководитель ВКР А.А. Вольсков



АНТИПЛАГИАТ
ОБНАРУЖЕНИЕ ЗАИМСТВОВАНИЙ

СПРАВКА

о результатах проверки текстового документа
на наличие заимствований

Пензенский государственный университет

ПРОВЕРКА ВЫПОЛНЕНА В СИСТЕМЕ АНТИПЛАГИАТ.ВУЗ

Автор работы: Зубанов Данила Владимирович
Самоцитирование
рассчитано для: Зубанов Данила Владимирович
Название работы: Применение импульсно-токовой обработки при изготовлении тонкопленочных
тензорезистивных датчиков
Тип работы: Выпускная квалификационная работа
Подразделение: Кафедра "Нано- и микроэлектроника"

РЕЗУЛЬТАТЫ

■ ОТЧЕТ О ПРОВЕРКЕ КОРРЕКТИРОВАЛСЯ: НИЖЕ ПРЕДСТАВЛЕНЫ РЕЗУЛЬТАТЫ ПРОВЕРКИ ДО КОРРЕКТИРОВКИ

СОВПАДЕНИЯ	36.65%	СОВПАДЕНИЯ	36.65%
ОРИГИНАЛЬНОСТЬ	53.05%	ОРИГИНАЛЬНОСТЬ	53.05%
ЦИТИРОВАНИЯ	10.3%	ЦИТИРОВАНИЯ	10.3%
САМОЦИТИРОВАНИЯ	0%	САМОЦИТИРОВАНИЯ	0%

ДАТА ПОСЛЕДНЕЙ ПРОВЕРКИ: 25.06.2023

ДАТА И ВРЕМЯ КОРРЕКТИРОВКИ: 25.06.2023 19:04

Структура документа: Проверенные разделы: основная часть с.1, 3-34, содержание с.2, библиография с.35-36
Модули поиска: ИПС Адилет; Библиография; Сводная коллекция ЭБС; Интернет Плюс*; Сводная коллекция РГБ; Цитирование; Переводные заимствования (RuEn); Переводные заимствования по eLIBRARY.RU (EnRu); Переводные заимствования по коллекции Гарант: аналитика; Переводные заимствования по коллекции Интернет в английском сегменте; Переводные заимствования по Интернету (EnRu); Переводные заимствования по коллекции Интернет в русском сегменте; Переводные заимствования издательства Wiley ; eLIBRARY.RU; СПС ГАРАНТ: аналитика; СПС ГАРАНТ: нормативно-правовая документация; IEEE; Медицина; Диссертации НББ; Коллекция НБУ; Перефразирования по eLIBRARY.RU; Перефразирования по СПС ГАРАНТ: аналитика; Перефразирования по Интернету; Перефразирования по Интернету (EN); Перефразированные заимствования по коллекции Интернет в английском сегменте; Перефразированные заимствования по коллекции Интернет в русском сегменте; Перефразирования по коллекции

Работу проверил: Головяшкин Алексей Николаевич

ФИО проверяющего

Дата подписи: 25.06.2023

Подпись проверяющего



Чтобы убедиться
в подлинности справки, используйте QR-код,
который содержит ссылку на отчет.

Ответ на вопрос, является ли обнаруженное заимствование
корректным, система оставляет на усмотрение проверяющего.
Предоставленная информация не подлежит использованию
в коммерческих целях.

Протокол

проверки на оригинальность в системе «Антиплагиат.ВУЗ» выпускной квалификационной работы на тему «Разработка программного обеспечения для проектирования тонкопленочных конденсаторов»

студента 4 курса Зубанов Д.В.

направления подготовки 11.03.04 Электроника и нанoeлектроника,

профиль «Микроэлектроника и твердотельная электроника»

Руководитель ВКР Вольсков А.А.

 **АНТИПЛАГИАТ**
Проверка оригинальности

СПРАВКА
Предоставлена информация о результатах проверки документа на наличие заимствования

Проверка выполнена в системе АНТИПЛАГИАТ.ВУЗ

Автор работы: Зубанов Даниил Владимирович
Самодипломирование: Зубанов Даниил Владимирович
Расчетчик для: Проверка оригинальности выпускной квалификационной работы
Название работы: Разработка программного обеспечения для проектирования тонкопленочных конденсаторов
Тип работы: Выпускная квалификационная работа
Подразделение: Кафедра «Техника микросистемной техники»

РЕЗУЛЬТАТЫ

СОВПАДЕНИЕ	ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ
75.63%	23.83%	75.63%	23.83%
ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ
ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ	ОРИГИНАЛЬНОСТЬ

Дата отправки документа: 25.06.2023

Структура документа: Проверка оригинальности документа

Модуль поиска: ИТС Антиплагиат.ВУЗ

Работу проверил: Голубовский Алексей Николаевич

Дата подписи: 25.06.2023



Руководитель ВКР

« 25 » июня

2023 г.

Подпись

Вольсков А.А.

фамилия, инициалы

ПриложениеЕ
(обязательное)
Презентация бакалаврской работы

					ПГУ.2.11.03.04.006ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дат		81

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Кафедра «Нано – и микроэлектроника»

Разработка программно- аппаратной части учебного стенда по вакуумной электронике

Выполнил: студент группы 19ПД1 Зубанов Д.В
Руководитель: к.т.н. доцент Вольсков А.А

работы: Разработка программно-аппаратной части лабораторного стенда по вакуумной электронике

ш.

Выбор программно-аппаратных средств для реализации проекта
Разработка алгоритма управления напряжением на сетке при помехе ИМ сигнала

Разработка алгоритма управления стендом

Разработка алгоритма приёма данных стенда
Зотовление макета учебного стенда

Алгоритм работы стенда

ы проведения экспериментов:

ание характеристик диода:

снуть первое реле и подать анодное
 яжения на диод;
 новить напряжение на блоке питания
 да равное 3В;
 ать на кнопку «Старт измерений»;
 снуть реле A_{REF3} ;
 ерить анодное напряжение;
 мкнуть реле A_{REF3} ;
 снуть реле A_{REF1} ;
 ерить напряжение на $R1$;
 читать по закону Ома ток анода,
 текущий через $R1=20\text{ Ом}$;
 мкнуть реле A_{REF1} ;
 ранить результаты измерения анодного
 яжения и расчёта тока анода;
 торить пункты 2-11 для напряжений на
 се питания анода в диапазоне от 4 до
 с шагом 1В;
 ранить результаты в виде *excel*;
 новить минимальное напряжение на
 се питания анода;
 мкнуть все реле;

Исследование характеристик триода:

1. Замкнуть второе реле и подать анодное напряжение на триод;
2. Замкнуть четвёртое реле и подать сеточное напряжение на триод;
3. Выбрать режим замера ВАХ при $U_{сетки} = 0\text{В}$;
4. Выставить длительность ШИМ сигнала блока питания сетки = 1%, чтобы получить $U_{сетки} = 0\text{В}$;
5. Установить напряжение на блоке питания анода равное 3В;
6. Нажать на кнопку «Старт измерений»;
7. Замкнуть реле A_{REF3} ;
8. Измерить анодное напряжение;
9. Разомкнуть реле A_{REF3} ;
10. Замкнуть реле A_{REF1} ;
11. Измерить напряжение на $R1$;
12. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
13. Разомкнуть реле A_{REF1} ;
14. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
15. Повторить пункты 5-14 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;

Алгоритм работы стенда

Сохранить результаты в виде *excel*;
Установить минимальное напряжение на блоке питания анода;
Выбрать режим замера ВАХ при $U_{\text{сетки}} = 0,5\text{В}$;
Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0,5\text{В}$;
Установить напряжение на блоке питания анода равное 3В ;
Нажать на кнопку «Старт измерений»;
Замкнуть реле $AREF3$;
Измерить анодное напряжение;
Разомкнуть реле $AREF3$;
Замкнуть реле $AREF1$;
Измерить напряжение на $R1$;
Рассчитать по закону Ома ток анода, протекающий через $R1 = 20\text{Ом}$;
Разомкнуть реле $AREF1$;
Сохранить результаты измерения анодного напряжения и расчёта тока анода;
Повторить пункты 20-29 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;

31. Сохранить результаты в виде *excel*;
32. Установить минимальное напряжение на блоке питания анода;
33. Выбрать режим замера ВАХ при $U_{\text{сетки}}$
34. Выставить длительность ШИМ сигнала так чтобы получить $U_{\text{сетки}} = 1\text{В}$;
35. Установить напряжение на блоке питания анода равное 3В ;
36. Нажать на кнопку «Старт измерений»;
37. Замкнуть реле $AREF3$;
38. Измерить анодное напряжение;
39. Разомкнуть реле $AREF3$;
40. Замкнуть реле $AREF1$;
41. Измерить напряжение на $R1$;
42. Рассчитать по закону Ома ток анода, протекающий через $R1 = 20\text{Ом}$;
43. Разомкнуть реле $AREF1$;
44. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
45. Повторить пункты 35-44 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 1В;

Алгоритм работы стенда

хранить результаты в виде *excel*;
 ановить минимальное напряжение на блоке
 ания анода;
 ставить длительность ШИМ сигнала так,
 бы получить $U_{\text{сетки}} = 0\text{В}$;
 брать режим замера анодно сеточной
 аактеристики;
 ановить напряжение на блоке питания
 да равное 60В ;
 кнуть реле *AREF3*;
 мерить анодное напряжение;
 омкнуть реле *AREF3*;
 кранить результат измерения анодного
 ряжения;
 ставить длительность ШИМ сигнала блока
 ания сетки $= 1\%$;
 кнуть реле *AREF1*;
 мерить напряжение на *R1*;
 считать по закону Ома ток анода,
 протекающий через $R1=20\text{Ом}$;
 омкнуть реле *AREF1*;
 кранить результат расчёта тока анода;

61. Повторить пункты 55-60 для длительности сигнала блока питания сетки от 2% до 10% шагом 1%;
62. Сохранить результаты в виде *excel*;
63. Установить минимальное напряжение на блоке питания анода;
64. Выставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0\text{В}$;
65. Выбрать режим замера сеточной характеристики;
66. Установить напряжение на блоке питания равное 60В ;
67. Замкнуть реле *AREF3*;
68. Измерить анодное напряжение;
69. Разомкнуть реле *AREF3*;
70. Сохранить результат измерения анодного напряжения;
71. Выставить длительность ШИМ сигнала блока питания сетки $= 1\%$;
72. Замкнуть реле *AREF2*;
73. Измерить напряжение на *R2*;
74. Рассчитать по закону Ома ток сетки, протекающий через $R2=5\text{кОм}$;
75. Разомкнуть реле *AREF2*;

Алгоритм работы стенда

хранить результат расчёта тока сетки;

вторить пункты 71-76 для

ительности ШИМ сигнала блока питания сетки от 2% до 100% с шагом 1%;

хранить результаты в виде ms ;

тановить минимальное

пряжение на блоке питания анода;

ставить длительность ШИМ сигнала так, чтобы получить сетки = 0В;

замкнуть все реле;

Исследование характеристик пентода

1. Замкнуть третье реле и подать анодное напряжение на пентод;
2. Замкнуть пятое реле и подать сеточное напряжение на пентод;
3. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 0\text{В}$;
4. Выставить длительность ШИМ сигнала блока питания сетки = 1%, чтобы получить $U_{\text{сетки}}$
5. Установить напряжение на блоке питания анода равное 3В;
6. Нажать на кнопку «Старт измерений»;
7. Замкнуть реле $AREF3$;
8. Измерить анодное напряжение;
9. Разомкнуть реле $AREF3$;
10. Замкнуть реле $AREF1$;
11. Измерить напряжение на $R1$;
12. Рассчитать по закону Ома ток анода, протекающий через $R1=20\text{ Ом}$;
13. Разомкнуть реле $AREF1$;
14. Сохранить результаты измерения анодного напряжения и расчёта тока анода;
15. Повторить пункты 5-14 для напряжений на блоке питания анода в диапазоне от 4 до 90В с шагом 10В;
16. Сохранить результаты в виде excel ;
17. Установить минимальное напряжение на блоке питания анода;
18. Выбрать режим замера ВАХ при $U_{\text{сетки}} = 0,5\text{В}$;
19. Выставить длительность ШИМ сигнала так, чтобы

Алгоритм работы стенда

ать на кнопку «Старт измерений»;
 нуть реле *AREF3*;
 зрить анодное напряжение;
 мкнуть реле *AREF3*;
 нуть реле *AREF1*;
 зрить напряжение на *R1*;
 читать по закону Ома ток анода,
 протекающий через *R1=20 Ом*;
 мкнуть реле *AREF1*;
 зранить результаты измерения анодного
 яжения и расчёта тока анода;
 торить пункты 20-29 для напряжений на
 се питания анода в диапазоне от 4 до 90В с
 ом 1В;
 зранить результаты в виде *excel*;
 новить минимальное напряжение на блоке
 ния анода;
 зать режим замера ВАХ при *Uсетки = 1В*;
 тавить длительность ШИМ сигнала так,
 ы получить *Uсетки = 1В*;
 новить напряжение на блоке питания
 а равное 3В;
 ать на кнопку «Старт измерений»;
 нуть реле *AREF3*;
 зрить анодное напряжение;
 мкнуть реле *AREF3*;
 нуть реле *AREF1*;

41. Измерить напряжение на *R1*;
42. Рассчитать по закону Ома ток анода,
протекающий через *R1=20 Ом*;
43. Разомкнуть реле *AREF1*;
44. Сохранить результаты измерения анодного
напряжения и расчёта тока анода;
45. Повторить пункты 35-44 для напряжений на
блоке питания анода в диапазоне от 4 до 90В
шагом 1В;
46. Сохранить результаты в виде *excel*;
47. Установить минимальное напряжение на блоке
питания анода;
48. Выставить длительность ШИМ сигнала так,
получить *Uсетки = 0В*;
49. Выбрать режим замера анодно сеточной
характеристики;
50. Установить напряжение на блоке питания
равное 60В;
51. Замкнуть реле *AREF3*;
52. Измерить анодное напряжение;
53. Разомкнуть реле *AREF3*;
54. Сохранить результат измерения анодного
напряжения;
55. Выставить длительность ШИМ сигнала бло
питания сетки = 1%;
56. Замкнуть реле *AREF1*;
57. Измерить напряжение на *R1*;
58. Рассчитать по закону Ома ток анода,
протекающий через *R1=20 Ом*;

Алгоритм работы стенда

вторить пункты 55-60 для длительности ШИМ сигнала блока питания от 2% до 100% с шагом 1%;

хранить результаты в виде *excel*;

тановить минимальное напряжение на блоке питания анода;

оставить длительность ШИМ сигнала так, чтобы получить $U_{\text{сетки}} = 0$

брать режим замера сеточной характеристики;

тановить напряжение на блоке питания анода равное 60В;

мкнуть реле *AREF3*;

мерить анодное напряжение;

зomкнуть реле *AREF3*;

хранить результат измерения анодного напряжения;

оставить длительность ШИМ сигнала блока питания сетки = 1%;

мкнуть реле *AREF2*;

мерить напряжение на *R2*;

считать по закону Ома ток сетки, протекающий через $R2=5\text{кОм}$;

зomкнуть реле *AREF2*;

хранить результат расчёта тока сетки;

вторить пункты 71-76 для длительности ШИМ сигнала блока питания от 2% до 100% с шагом 1%;

хранить результаты в виде *excel*;

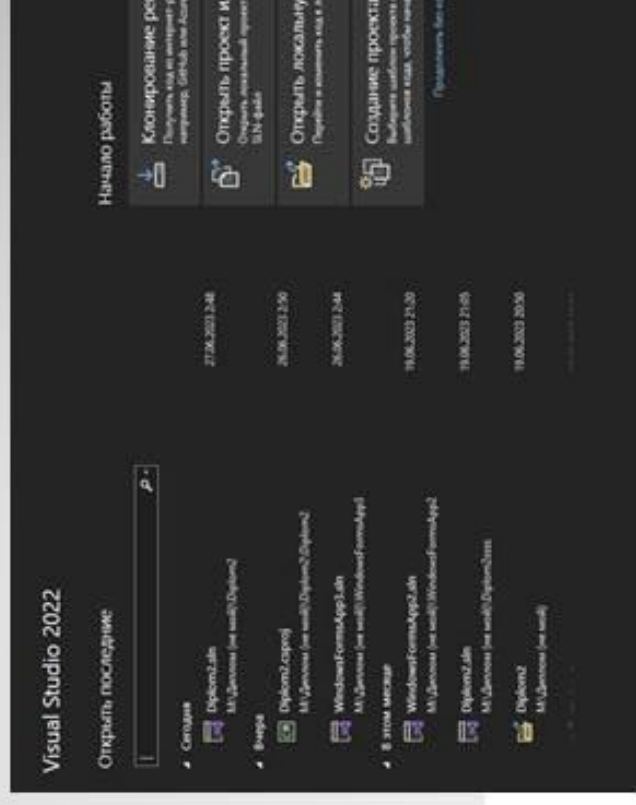
тановить минимальное напряжение на блоке питания анода;

Выбор программно-аппаратных средств для реализации проекта

Microsoft Visual Studio - линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментов.



Visual Studio обладает обширным набором инструментов для разработки приложения на основе .NET Framework. для реализации проекта



ыбор программно-аппаратных средств для реализации
проекта



ArduinoIDE — интегрированная среда разработки для *Windows*, *Linux*, разработанная на *C* и *C++* предназначенная для создания и загрузки программ на *Arduino*-совместимые платы, а также на платы других производителей.

ino IDE позволяет
ро и качественно
ограничить
контроллер а также,
одаря встроенному
тору порта,
лнять отладку
раммы без
льзования сторонних
ожений



Главное окно программы

аккумуляторная электроника



Подключиться

Обновить список
доступных портов

Установить экспери

Задать напряжени
сетки

Записать значени

Показать график

>

>

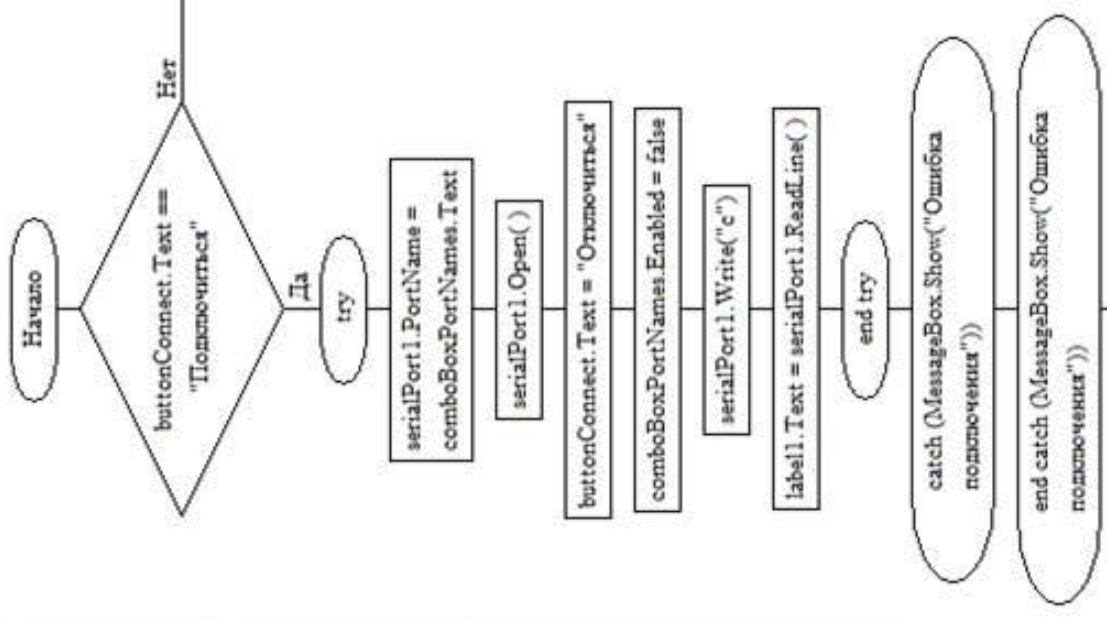
>

...

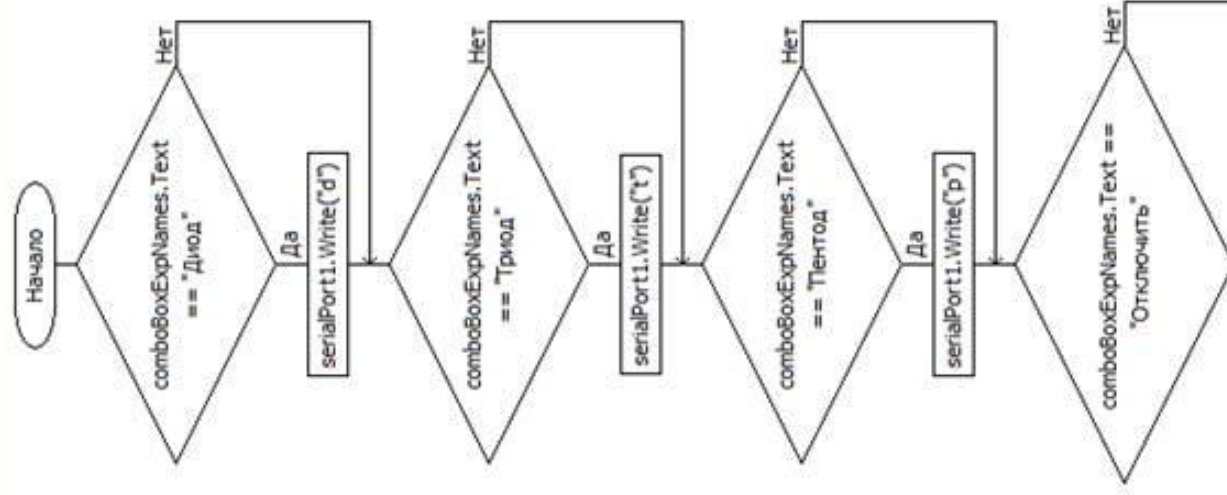
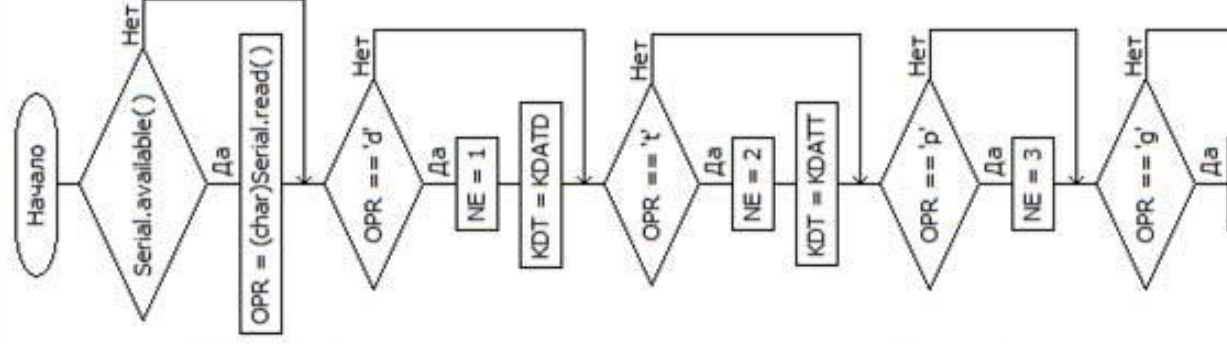
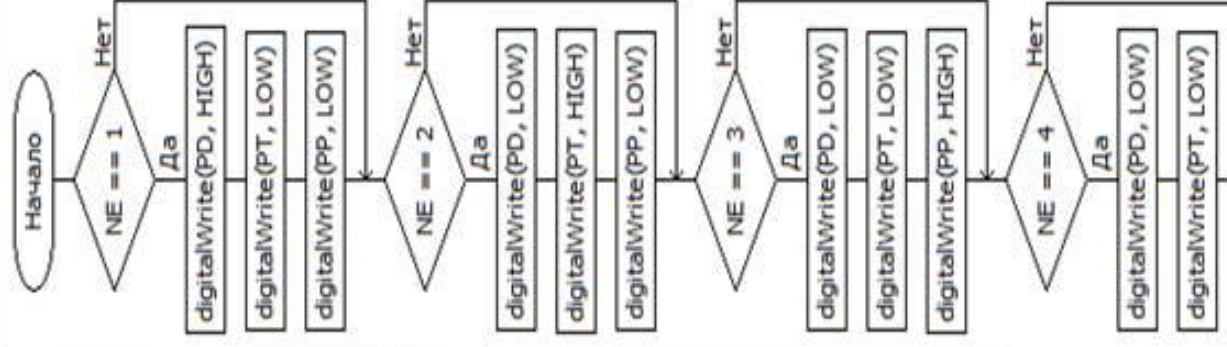
...

...

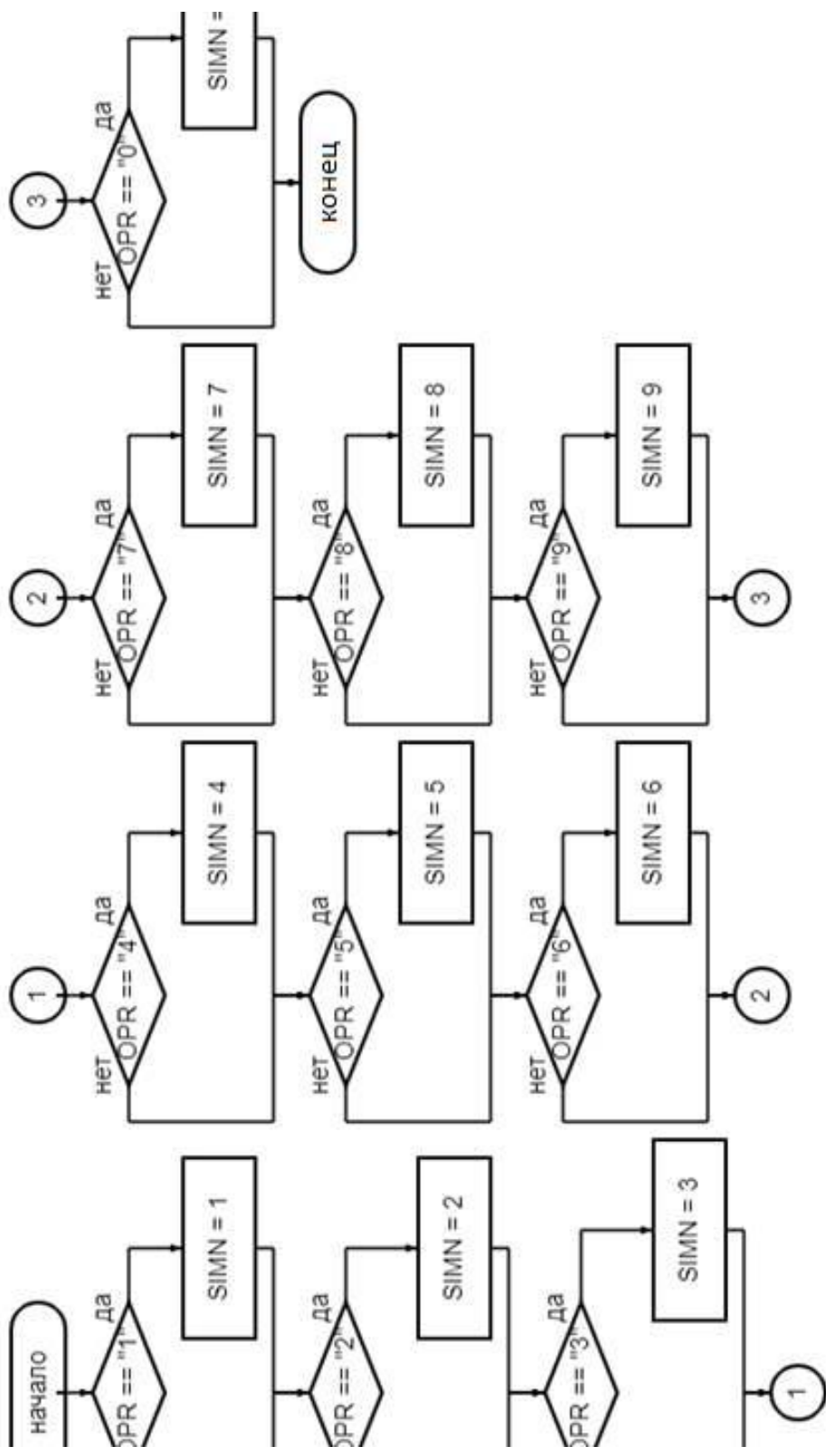
Алгоритм подключения



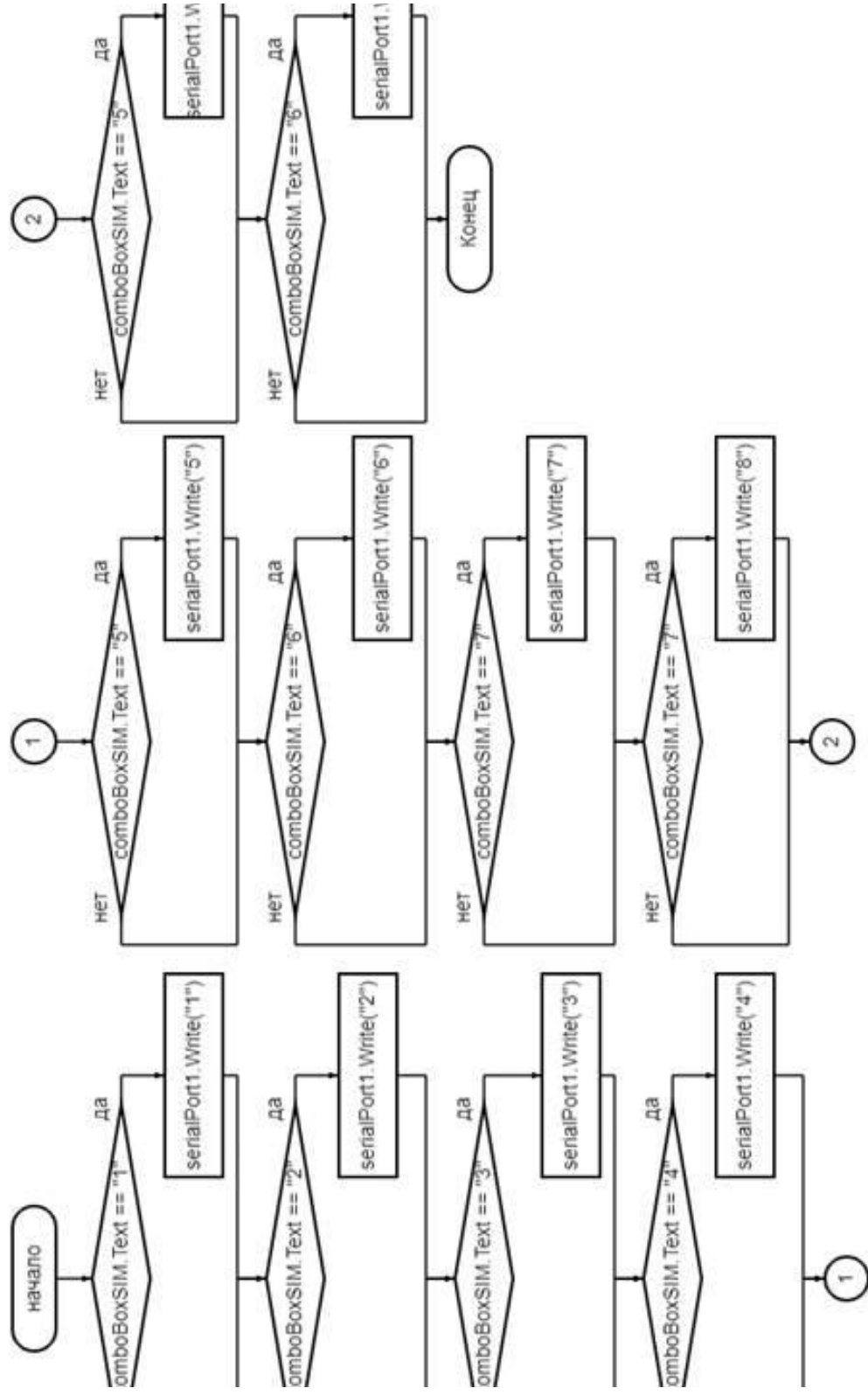
Алгоритм управления стендом



алгоритм управления напряжением на сетке



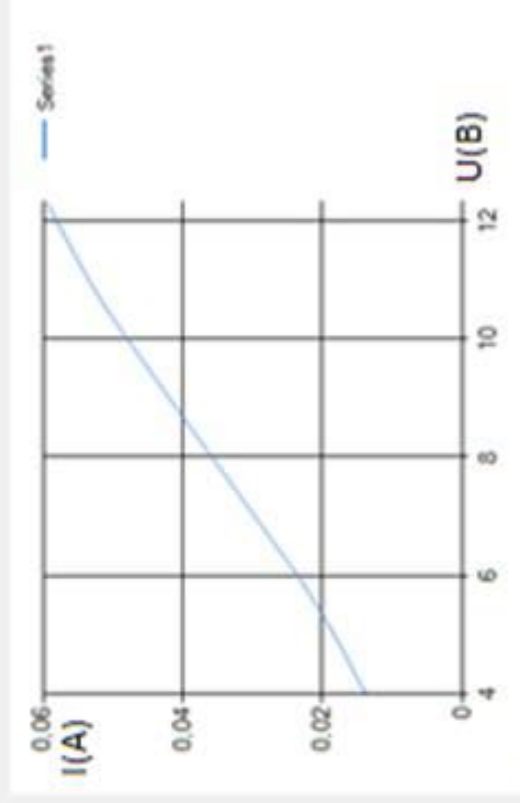
Алгоритм управления напряжением на сетке



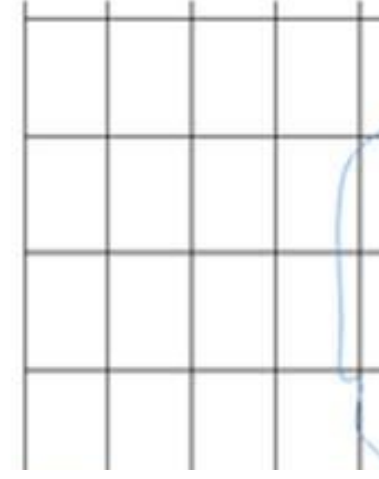
Результат работы программы

Результат работы программы

Результат работы программы



Вольт амперная характеристика диода

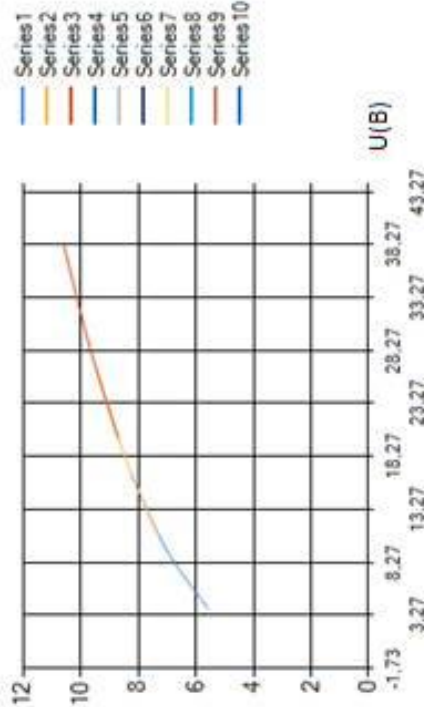
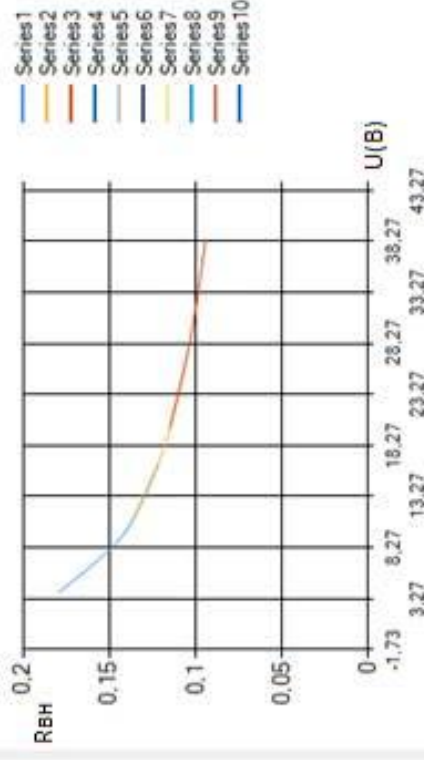


Результат работы программы

Характеристики триода (пентода)



Анодная характеристика триода



Заключение

В ходе выполнения данной работы были получены следующие результаты:

Выбраны программно-аппаратные средства для реализации проекта

Разработка алгоритма управления напряжением на сетке при помощи ПИД-регулятора

Разработка алгоритма управления стендом

Разработан алгоритм приёма данных стенда

Изготовление макета учебного стенда