

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ «ГРОДНЕНСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИМЕНИ ЯНКИ КУПАЛЫ»

УДК 53.07:531.14

Физико-технический факультет

Кафедра общей физики

ИЛЮКОВ ИЛЬЯ АЛЕКСАНДРОВИЧ
МОДЕРНИЗАЦИЯ ЛАБОРАТОРНОЙ УСТАНОВКИ «МАШИНА
АТВУДА»

Дипломная работа
студента 4-го курса 1-ой группы дневного отделения

Научный руководитель,
Доктор физико-математических наук,
Доцент Маскевич А. А.

Допущен к защите
зав. каф. общей физики,
доктор физ.-мат. наук, доцент

_____/Маскевич А.А./

ГРОДНО 2020

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ.

LCD	жидко-кристаллический экран.
DD	графическое обозначение микроконтроллером.
VCC	положительное напряжение источника питания и устройств.
GND	общий провод источника питания и устройств.
SDA	шина последовательных данных.
SLC	шина тактирования.
ДУ	дистанционное управление
ТТЛ	транзисторно-транзисторная логика
МК	микроконтроллер
АЦП	аналого-цифровой преобразователь
ППЗУ	постоянное запоминающее устройство

РЕФЕРАТ

МОДЕРНИЗАЦИЯ ЛАБОРАТОРНОЙ УСТАНОВКИ «МАШИНА АТВУДА»

Дипломная работа 49 стр., 3 табл., 6 рис., 17 источников, 8 приложений.

Микроконтроллеры, лабораторные установки, «Машина Атвуда».

Объектами исследований являлись микроконтроллеры и лабораторная установка «Машина Атвуда».

В данной дипломной работе произведена модернизация лабораторной установка “Машина Атвуда” отвечающая следующим техническим характеристикам:

- 1) Возможность дистанционного управления моментом начала и конца счёта времени.
- 2) Блок измерения времени имеет следующие режимы работы:
 - а) Измерение времени с дистанционным управлением.
 - б) Измерение периода колебательных процессов, по сигналам от внешних устройств.
 - в) Функции секундомера с ручным и дистанционным управлением.
 - г) Функции таймера с ручной установки отсчитываемого времени.
- 3) Модернизированная лабораторная установка имеет цифровую и звуковую индикации окончания отсчёта времени.

Применение микроконтроллера позволило создать малогабаритное устройство, отличающееся низким электропотреблением. Наличие большого количества свободно распространяемой информации по применению микроконтроллеров и программному обеспечению позволили произвести лёгкое адаптированное для выполняемых задач, указанных в дипломной работе.

Данное устройство является многофункциональным. Предполагается его использование в составе универсального измерительного блока для модернизации ряда лабораторных работ лаборатории «Механика» кафедры общей физики, физико-технического факультета.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	6
1. ОБЗОР СУЩЕСТВУЮЩЕЙ УСТАНОВКИ	8
2. МОДЕРНИЗИРОВАННАЯ ЛАБОРАТОРНАЯ УСТАНОВКА “МАШИНА АТВУДА”	11
3. РАБОТА УСТРОЙСТВА ПО СТРУКТУРНОЙ, ФУНКЦИОНАЛЬНОЙ И ПРИНЦИПИАЛЬНОЙ СХЕМЕ	12
3.1. Работа устройства по структурной схеме	12
3.2. Выбор элементной базы	15
3.3. Работа устройства по функциональной схеме	18
3.4. Работа устройства по принципиальной схеме	24
4. КОНСТРУКЦИЯ МОДЕРНИЗИРОВАННОЙ ЛАБОРАТОРНОЙ УСТАНОВКИ “МАШИНА АТВУДА”	28
5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ	34
5.1 Первоначальные настройки	34
5.2 Выбор режима работы	35
5.3 Принцип работы режима “СЕКУНДОМЕР”	39
5.4. Принцип работы режима “ТАЙМЕР”	39
6. ОБЩИЕ ТРЕБОВАНИЯ БЕЗОПАСНОСТИ.	41
ЗАКЛЮЧЕНИЕ	46
СПИСОК ЛИТЕРАТУРЫ	48
ПРИЛОЖЕНИЕ А	50
ПРИЛОЖЕНИЕ Б	51
ПРИЛОЖЕНИЕ В	52
ПРИЛОЖЕНИЕ Г	53
ПРИЛОЖЕНИЕ Д	54
ПРИЛОЖЕНИЕ Е	55
ПРИЛОЖЕНИЕ Ж	56
ПРИЛОЖЕНИЕ З	57

ВВЕДЕНИЕ

В настоящее время жизнь любого человека невозможно представить без электронных устройств. Внедрение электронных систем позволяет уменьшить человеческие риски, за счёт автоматизации человеческого труда, уменьшить стоимость конечных устройств и увеличить качество выполняемой работы, за счёт улучшения характеристик электронных устройств.

Также не стоит забывать и про спектр использования данных устройств. Электронно-вычислительные устройства можно использовать во всех сферах жизни человека: топливной, химической, машиностроительной, деревообрабатывающей и т.д. Спектр применения ограничен лишь фантазией и сообразительностью разработчиков электронных установок.

Кроме того, в течение последних десятилетий в научно-технической сфере, все чаще стали применяться различные микроконтроллеры, которые позволяют ускорить и упростить достаточно большой спектр задач, таких как выполнение лабораторных работ. Выполнение лабораторных работ является важной частью учебной деятельности при подготовке студентов всех технических специальностей. А навыки, получаемые в лаборатории: понимание наблюдаемых процессов, использование измерительных приборов, обработка полученных результатов, позволяют студентам закреплять свои теоретические навыки на практике.

Классические лабораторные работы, выполняемые в рамках практикума, базируются на устаревших лабораторных установках, многие элементы которых изношены и морально устарели. Это не способствует привлекательности предмета «Физика». Исходя из выше сказанного, была поставлена цель по модернизации лабораторной установки «Машина Атвуда» с возможностью управления через внешние устройства ввода, а также и через собственные органы управления и вывод измеренных результатов.

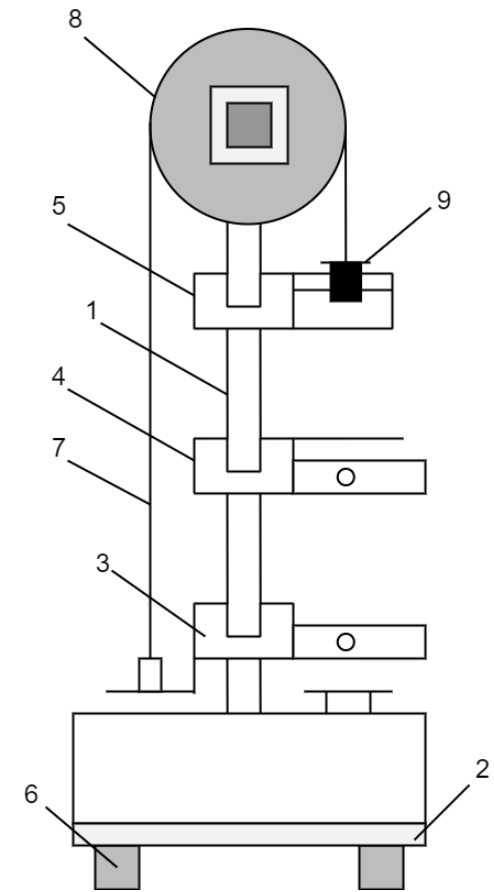
Основные задачи, решаемые при разработке устройства измерения временных интервалов на микроконтроллере:

- Анализ существующих схемных решений и подбор литературы
- Выбор из существующей компонентой базы микроконтроллера отвечающего на следующие требования: быстродействие, тактовая частота, ПЗУ, ОЗУ, электропотребление, возможность применения универсальных программ, низкое электропотребление, стоимость.

- Составление схемы электрической структурной.
- Составление схемы электрической функциональной.
- Разработка схемы электрической принципиальной.
- Адаптация программного продукта из открытых источников, для управления микроконтроллером в режиме измерения времени.

1. ОБЗОР СУЩЕСТВУЮЩЕЙ УСТАНОВКИ

Машина Атвуда — лабораторное устройство для изучения поступательного движения с постоянным ускорением. Была изобретена в 1784 году английским физиком и математиком Джорджем Атвудом, она имеет вертикальную шкалу, на верхнем конце которой имеется легкий блок, вращающийся с очень малым трением. Через блок перекинута легкая нить с грузами одинаковой массы m . Ниже представлена схема существующей установки с описание основных компонентов:



1 - вертикальная колонна, 2 – основание установки, 3 – нижний кронштейн, 4 – средний кронштейн, 5 – верхний кронштейн, 6 – регулируемые ножки, 7 – нерастяжимая нить, 8 – неподвижный блок, 9 – кронштейн для снятия перегрузки с груза

Рисунок 1 – Внешний вид лабораторной установки “Машина Атвуда”

Описание установки:

Общий вид машины Атвуда показан на рис 1. На вертикальной колонне 1, закрепленной на основании 2, установлены три кронштейна: нижний 3, средний 4 и верхний 5. Основание оснащено регулируемыми ножками 6, которые позволяют произвести выравнивание положения прибора. Верхний и средний кронштейны могут передвигаться вдоль колонны и фиксироваться в любом положении, что позволяет изменить расстояние, которое проходит груз при равномерном и равноускоренном движении.

Для определения расстояния на колонне имеется миллиметровая шкала. Грузы одинаковой массы соединены нерастяжимой нитью 7. Нить переброшена через неподвижный блок 8. На среднем кронштейне закреплен фотоэлектрический датчик. Кронштейн снимает с падающего вниз груза перегрузок 9. В это время срабатывает фотоэлектрический датчик и запускает секундомер, который отсчитывает продолжительность равномерного движения груза. Нижний кронштейн оснащен резиновым амортизатором и фотоэлектрическим датчиком, который срабатывает во время завершения движения груза.

Таблица 1 - Технические характеристики существующей установки:

Погрешность измерения времени	5 мс
Максимальная граница измерения времени	9999.9 с (2 часа 46 минут 39.9 секунд)
Максимальное расстояния измерения	100 см
Погрешность метрической линейки	1 мм

Отметим наиболее существенные недостатки лабораторных установок.

1. Блок измерения. В качестве блока измерения временных интервалов применяется электронный секундомер СТЦ-1 с индикацией, выполненной на газоразрядных индикаторах типа ИН-12А, которые имеют повышенное напряжение питания 250 В и повышенное энергопотребление.
2. В качестве органа управления используют кнопки типа П2К, которые характеризуются большим дребезгом контактов, что влияет на точность проведения измерений. Вносит ошибку - примерно 10-70 мс.

3. Невозможность дистанционного управления началом и концом отсчёта времени (Студент вручную останавливает отсчёт времени секундомера, после падения тела).
4. Измерение расстояния производится с помощью метрической линейки, что в свою очередь также влияет на точность измерения расстояния.
5. Из-за длительного и интенсивного срока эксплуатации данная установка физически устарела.

В главе 7 приведена таблица результатов измерения значений времени движения груза при равноускоренном движении, выполненные на существующей установке.

2. МОДЕРНИЗИРОВАННАЯ ЛАБОРАТОРНАЯ УСТАНОВКА “МАШИНА АТВУДА”.

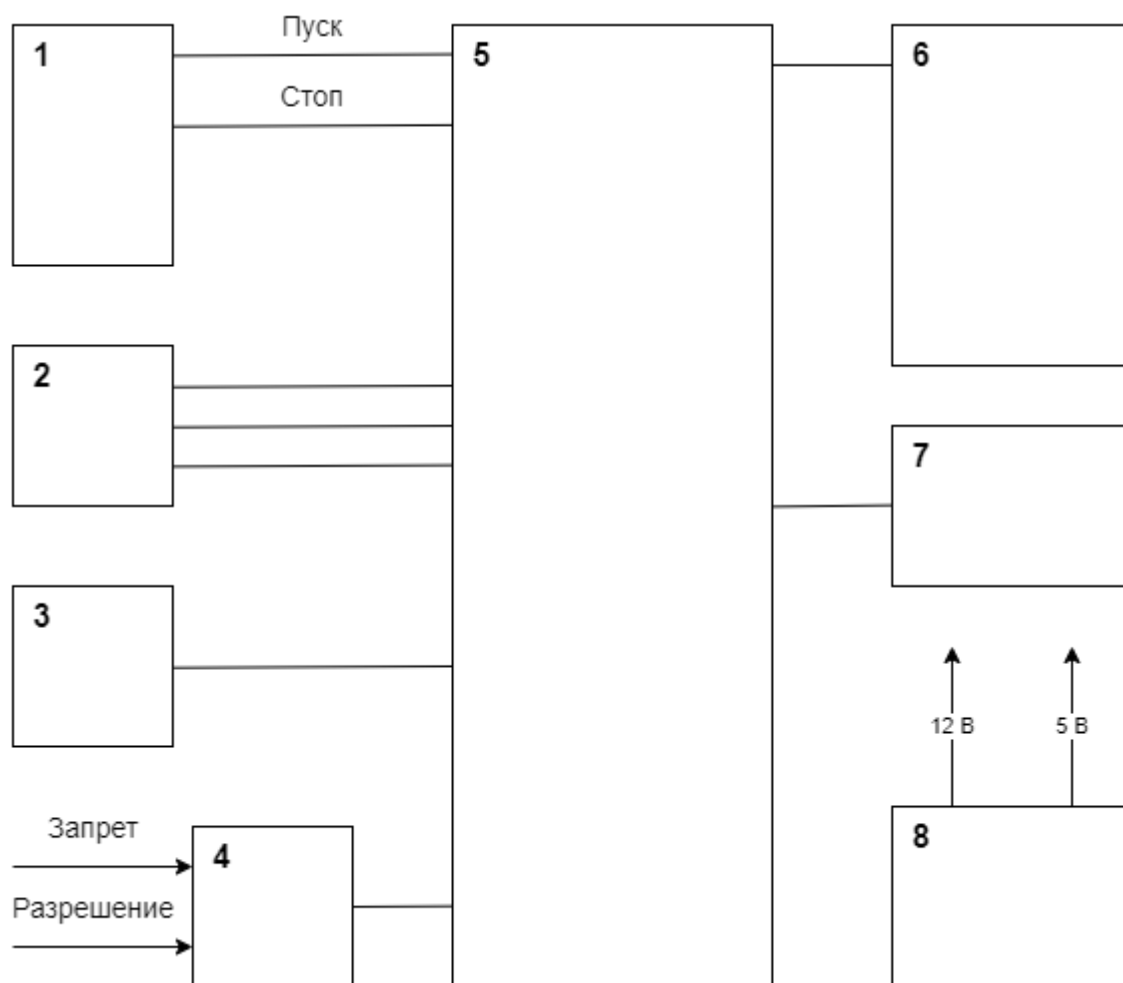
В главе 1 рассмотрена лабораторная установка Машина Атвуда, технические характеристики и её недостатки. В дипломном проекте были сформулированы следующие технические требования для успешной модернизацию данной установки:

1. Изготовить многофункциональное устройства для измерения временных интервалов на микроконтроллере.
2. Изготовление блок измерения на микроконтроллере, с возможностью дистанционного управления началом и концом счёта времени.
3. Блок измерения должно иметь следующие режимы работы:
 - а. Измерение времени с дистанционным управлением.
 - б. Измерение времени с ручным управлением.
 - в. Измерение периода колебательных процессов, по сигналам от внешних устройств.
 - г. Таймер с ручной установкой отсчитываемого времени.
4. Наличие цифровой и звуковой индикации окончания времени отсчёта.

3. РАБОТА УСТРОЙСТВА ПО СТРУКТУРНОЙ, ФУНКЦИОНАЛЬНОЙ И ПРИНЦИПИАЛЬНОЙ СХЕМЕ

3.1. Работа устройства по структурной схеме

Важно сказать, что, начиная разработку электронного секундомера на базе микроконтроллера, была составлена структурная схема устройства. Структурная схема модернизированной установки представлена на (Рис.2).



1 – блок управления в ручном режиме, 2 – блок управления и задач, 3 – внешние цепи, 4 – схема внешних сигналов управления счётом, 5 – блок микроконтроллера, 6 – блок индикации, 7 – звуковой излучатель, 8 – блок питания.

Рисунок 2 - Схема электрическая структурная модернизированной установки

Данная структурная схема определяет основные функциональные части изделия, их назначение и взаимосвязи между ними. Схема отображает принцип действия устройства в самом общем виде. Каждая часть схемы выполняет полностью законченную роль.

Поскольку модернизированная лабораторная установка “Машина Атвуда” разрабатывается как многофункциональное измерительное устройство, в данном варианте реализованы на программном уровне следующие режимы работы:

1. Режим измерения линейного перемещения.
2. Режим измерения углового перемещения.
3. Режим подсчета периодов колебания.
4. Режим таймера.
5. Режим секундомера.

При включении модернизированной лабораторной установки в сеть, устройство совершает аппаратный сброс. Далее, с помощью Блока управления и задач 2, представленного в структурной схеме, происходит установка режима и его выбор. Описание и рассмотрение режимов работы таймера и секундомера описано ниже.

В состав схемы режима таймера входят следующие компоненты: блок управления в ручном режиме 1, блок управления и задач 2, схема внешних сигналов управления счётом 4, блок микроконтроллера 5, блок индикации – 6, звуковой излучатель – 7.

Первым этапом выполнения данного режима является задание времени отсчёта времени при помощи блока управления и задач 2. Задаваемое значение отображается на блоке индикации 6. При поступлении разрешающего сигнала с блока управления в ручном режиме 1 или с схем внешних сигналов управления 4. Данный управляющий сигнал является флагом для перехода блока микроконтроллера 6 во второй этап выполнения управляющей программы.

Вторым этапом выполнения данного режима является отсчёт времени и периодическое информирование состояния выполнения режима на блоке индикации 6. После окончания отсчёта заданного времени на блок индикации 6 выводиться сообщение об окончании работы режима. Так же необходимо отметить, что по окончании режима работы производится звуковая индикация с помощью блока индикации 7. По окончании данного режима, устройство

переходит в исходное положение и может быть использовано снова в данном режиме.

В состав схемы режима секундомера входят следующие компоненты: блок управления в ручном режиме 1, схема внешних сигналов управления счётом 4, блок микроконтроллера 5, блок индикации – 6.

Первым этапом выполнения данного режима является ожидание поступления разрешающего сигнала с блока управления в ручном режиме 1 или со схем внешних сигналов управления 4. Данный управляющий сигнал является флагом для запоминания времени, когда пришел размещающий сигнал, и перехода блока микроконтроллера 6 во второй этап выполнения управляющей программы.

Вторым этапом выполнения данного режима является вычисление времени произошедшего с момента прихода, размещающего (разрешающего?) сигнала и периодическое информирование его на блоке индикации 6. Окончание отсчёта заданного времени производится по запрещающему сигналу с блока управления в ручном режиме 1 или со схемы внешних сигналов управления счётом 4. По окончании данного режима, устройство переходит в исходное положение и может быть использовано снова в данном режиме.

3.2. Выбор элементной базы

3.2.1 Выбор микроконтроллера

Ввиду использования микроконтроллеров в данной дипломной работе, предстояло провести выбор из существующей базы вариантов оптимальный микроконтроллер. Основной особенностью при выборе микроконтроллера являлось их широкая вариация моделей, что требовало с одной стороны сравнить много вариации моделей, но с другой стороны позволяло подобрать конкретную модель под нужную задачу для модернизации лабораторной установки.

Поставленной в дипломной работе задаче соответствуют три типа микроконтроллеров:

1. AVR [1]
2. PIC [2]
3. ARM [3]

Таблице 2 Сравнительные характеристики трёх групп микроконтроллеров

	PIC16F690	Atmega 328P-PU	STM32F100C4T6B
Тип ядра	PIC	AVR	ARM
Тактовая частота	8 МГц	20 МГц	24 МГц
ПЗУ	8 Кб	32 Кб	16 Кб
ОЗУ	1 Кб	2 Кб	4 Кб
Количество портов ввода/вывода	18	23	37
Разрядность шины данных	8 бит	8 бит	32 бит
Цена	4,6 рубля	4,8 рубля	4,6 рубля

Так же был произведен сравнительный обзор микроконтроллеров семейства Intel MCS-51 с выше указанными микроконтроллерами. Семейство данных микроконтроллеров широко применяется в учебном процессе, так как имеется большое количество макетов, позволяющих программно моделировать различные устройства, но данные микроконтроллеры отличаются низкими техническими характеристиками [4] и использование их было признано нецелесообразным.

После детального рассмотрения трех микроконтроллеров было решено остановить свой выбор для решения поставленной задачи в дипломной работе на самом распространенном и по большому наличию открытого программного продукта на микроконтроллере фирмы Atmega, а именно Atmega 328P-PU. Данное решение обосновывается тем фактом, что при разработке конечного устройства на микроконтроллере важную роль играет наличие документации и вспомогательной литературы [5], что позволяет облегчить разработку и последующую поддержку разработанного устройства в будущем.

3.2.2 Выбор блока управления и задач

В качестве блока управления и задач решено было выбрать электромеханический инкрементный энкодер типа KY-040, имеющий 24 импульса на один оборот. Преимуществами данного устройства является его простота, благодаря применению скользящих контактов в данном типе, наличие кнопки. Основным минусом является дребезг контактов, но данный недостаток решается на программном уровне устройства.

3.2.3 Выбор блока индикации

При выборе блока индикации рассматривались два варианта lcd дисплей: 1602 или два семисегментных модуля tm1637. Основным критерием выбора блока индикации работа в динамическом режиме и возможности вывода любой текстовой и цифровой информации.

После сравнения цен на данные блоки индикации и их характеристики было решено выбрать жидкокристаллический дисплей 1602, учитывая возможность выводить в более приятном, читаемом формате информацию.

Основным недостатком данного дисплея является большое количество задействованных портов ввода/вывода микроконтроллера. Данный недостаток решается использованием дисплея с подключением через I2C интерфейс, что уменьшает количество задействованных портов ввода/вывода микроконтроллера. При подключении без интерфейса I2C необходимо задействовать 6 цифровых портов, в тоже время при использовании контроллера шины на основе микросхемы PCF8574AT задействованы лишь 2 цифровых порта.

3.2.4 Выбор блока питания

В качестве блока питания применяется внешнее устройство, готовый источник питания номиналом 9В и силой тока не менее 1 А.

3.2.4 Выбор звукового излучателя

В качестве звукового излучателя был выбран пьезоизлучатель КРМ-G1205В. Преимуществом данного излучателя является его простота при подключении его к блоку микроконтроллера. Для работы пьезоизлучателя необходимо один из его контактов подключить к цифровому порту микроконтроллера.

3.3. Работа устройства по функциональной схеме.

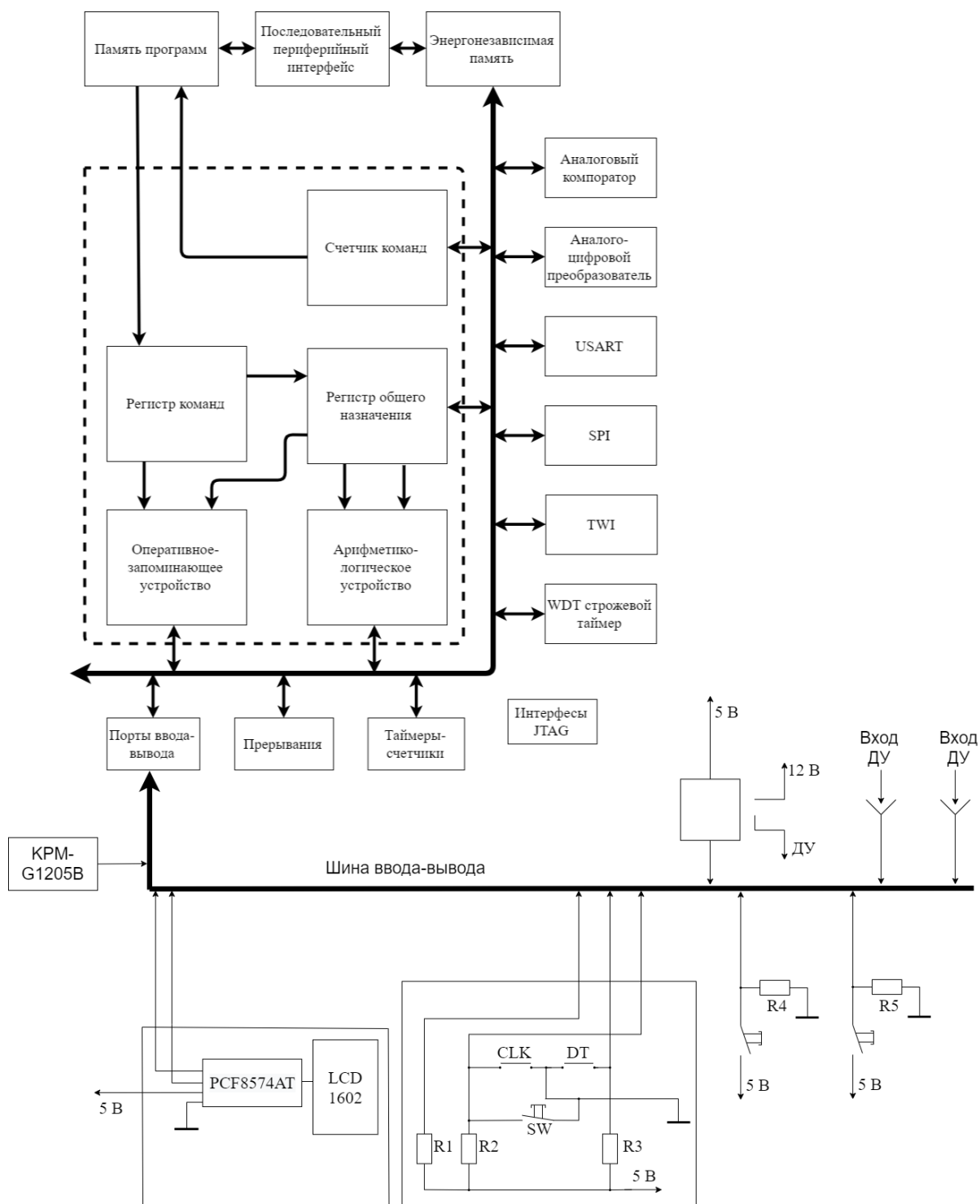


Рисунок 3 – Схема электрическая функциональная модернизированного устройства

3.3.1 Схема электрическая функциональная блока управления счётом и внешних сигналов управления счётом

Функциональный блок управления счётом представляет собой энкодер типа KY-040. Он состоит из 3 управляющих контактов: SW, DT, CLK. Подключённых по следующему принципу к входам микроконтроллера [6]:

- SW к выводу DD4,
- DT к выводу DD3,
- CLK к выводу DD2;

При вращении энкодера с выходов CLK и DT поступают прямоугольные импульсы, на соответствующие входы блока микроконтроллера, сдвинутые друг относительно друга на фазовый угол 90 [7]. Программное определение направления движения вала энкодера происходит по сдвигу фаз импульсов с выходов энкодера CLK и DT. [8]

Управление таймером или секундомером может производиться с помощью двух внешних сигналов, поданных через разъем типа VGA на корпусе устройства.

В обоих случаях управление ведется логической единицей. Реакция на логическую единицу происходит сразу после получения сигнала. Микроконтроллер не ожидает конца разрешающего импульса.

Так же с помощью внешних сигналов управления счётом поступающих на вход D5 и D6 можно управлять внешними цепями. К примеру, в конце работы режима таймер на внешние цепи подается логическая единица.

3.3.2 Схема электрическая функциональная схема звукового излучателя

Сигнализирование о завершение любых событий при работе режимов происходит с помощью пьезоизлучателя. При появлении сигнала о завершение работы, на выводе микроконтроллера 5 к которому подключен пьезоизлучатель формируются два коротких импульса с длительностью 1с которые воспроизводятся пьезоизлучателем.

3.3.3 Схема электрическая функциональная блока микроконтроллера

На рисунке 2 представлена функциональная схема микроконтроллера Atmega328, который является основным компонентом в разрабатываемом устройстве. Блок микроконтроллера состоит из:

1. Постоянное запоминающее устройство (ПЗУ) - предназначено для хранения последовательности команд, управляющих функционированием микроконтроллера. Ее главное достоинство в том, что она построена на принципе электрической перепрограммируемости, т. е. допускает многократное стирание и запись информации. В процессе выполнения программа считывается из этой памяти, а блок управления обеспечивает ее декодирование и выполнение необходимых операций. Содержимое памяти программ не может меняться во время выполнения программы. Поэтому функциональное назначение микроконтроллера не может измениться, пока содержимое его памяти программ не будет перепрограммировано.
2. Последовательный периферийный интерфейс - интерфейс для последовательного обмена данными. В том случае, если в системе присутствует внешний процессор, с помощью этого интерфейса может быть обеспечена межпроцессорная связь.
3. Энергонезависимая память – предназначена для долговременного хранения различной информации, которая может изменяться в процессе функционирования микроконтроллерной системы, используется энергонезависимая память (EEPROM-память). Данный микроконтроллер имеет блок энергонезависимой электрически перезаписываемой памяти данных размером 32 Кбайт. Этот тип памяти, доступный программе микроконтроллера непосредственно в ходе ее выполнения, удобен для хранения промежуточных данных, различных констант, коэффициентов, серийных номеров, ключей и т.п.
4. Аналоговый компаратор - предназначен для сравнения непрерывно изменяющихся сигналов. Аналоговый компаратор имеет два входных аналоговых сигнала: анализируемый и опорный сигнал. После сравнения, на выходе получаем логический сигнал, содержащий 1 бит информации.
5. Аналого-цифровой преобразователь (АЦП) - устройство, выполняющее преобразования входной физической величины в ее числовое представление.
6. USART - удобный и простой последовательный интерфейс последовательного ввода-вывода, который может использоваться для работы с периферийными устройствами.

7. SPI - предназначен для организации обмена данными между двумя устройствами. С его помощью может осуществляться обмен данными между микроконтроллером и различными устройствами. Также с помощью этого интерфейса удобно производить обмен данными между несколькими микроконтроллерами AVR.
8. TWI - интерфейс позволяет объединить вместе до 128 различных устройств с помощью двунаправленной шины, состоящей из линии тактового сигнала (SCL) и линии данных (SDA).
9. WDT сторожевой таймер - предназначен для предотвращения катастрофических последствий от случайных сбоев программы. Он имеет свой собственный RC-генератор, работающий на частоте 1 МГц. Идея использования сторожевого таймера предельно проста и состоит в регулярном его сбрасывании под управлением программы или внешнего воздействия до того, как закончится его выдержка времени и не произойдет сброс процессора.
10. Интерфейсы JTAG – создан для решения проблем тестирования электроники. Стандарт позволяет значительно упростить жизнь за счет встраивания специальной архитектуры в современные чипы, обеспечивающей доступ к выводам (точнее, к специальным блокам ввода-вывода) с помощью 4-х проводного последовательного интерфейса. Эта архитектура позволяет не только контролировать их состояние, но и управлять ими.
11. Таймеры-счётчики – нужны для измерения точных временных промежутков. Точность достигает одного машинного такта.
12. Блок прерывания – один из важнейших блоков в микроконтроллере. Прерывание прекращает нормальный ход программы для выполнения приоритетной задачи, определяемой внутренним или внешним событием. Для каждого такого события разрабатывается отдельная программа, которую называют подпрограммой обработки запроса на прерывание (для краткости - подпрограммой прерывания), и размещается в памяти программ. При возникновении события, вызывающего прерывание, микроконтроллер сохраняет содержимое счетчика команд, прерывает выполнение центральным процессором текущей программы и переходит к выполнению подпрограммы обработки прерывания. После выполнения подпрограммы прерывания осуществляется восстановление предварительно сохраненного счетчика команд и процессор возвращается к выполнению

прерванной программы. Для каждого события может быть установлен приоритет. Понятие приоритет означает, что выполняемая подпрограмма прерывания может быть прервана другим событием только при условии, что оно имеет более высокий приоритет, чем текущее. В противном случае центральный процессор перейдет к обработке нового события только после окончания обработки предыдущего.

13. Порты ввода-вывода служат для подключения внешних устройств.
14. Счетчик команд - это специализированный внутренний регистр микроконтроллера, в котором хранится адрес следующей команды.
15. Регистр команд - регистр, содержащий команду, исполняемую в данный момент процессором.
16. Регистр общего назначения - регистры, используемые для хранения промежуточных итогов вычислений. Имеют два значительных преимущества: обращение к регистрам выполняется намного быстрее, чем к памяти. При обращении к регистрам не нужно указывать адрес (как в случае с памятью).
17. Оперативное - запоминающее устройство - память, предназначенная для временного хранения данных и команд, необходимых процессору для выполнения им операций.
18. Арифметико-логическое устройство - служит для выполнения арифметических и логических преобразований над данными, называемыми в этом случае операндами.

3.3.4 Схема электрическая функциональная блока индикации

Для динамической индикации используется lcd дисплей 1602 подключенный по интерфейсу I2C с контроллером шины PCF8574AT, применяющимся для расширения линий ввода/вывода [9].

Для осуществления процесса обмена информацией по I2C шине, используется всего два сигнала линия данных SDA линия синхронизации SCL. Процедура обмена начинается с того, что ведущий формирует состояние СТАРТ - ведущий генерирует переход сигнала линии SDA из ВЫСОКОГО состояния в НИЗКОЕ при ВЫСОКОМ уровне на линии SCL. Этот переход воспринимается всеми устройствами, подключенными к шине как признак начала процедуры обмена. Процедура обмена завершается тем, что ведущий формирует состояние

СТОП - переход состояния линии SDA из низкого состояния в ВЫСОКОЕ при ВЫСОКОМ состоянии линии SCL.

3.4. Работа устройства по принципиальной схеме

3.4.1. Схема электрическая принципиальная блока управления и задач, кнопок и звукового излучателя

Блок управления и задач представляет из себя электромеханический энкодер. Он состоит из двух скользящих контактов CLK и DT, а также кнопки SW подключенной по схеме с подтягивающим резистором. При вращении энкодера в одном из направлений на контактах CLK и DT возникают прямоугольные импульсы, сдвинутые друг относительно друга на фазовый угол 90° . При нажатии на кнопку вывод замыкается на землю, тем самым генерируя сигнал логического нуля.

Блок управления в ручном режиме состоит из двух кнопок. Во всех режимах работы блока управления защита от дребезга контактов выполнена программно, что дополнительно повышает надежность блока управления.

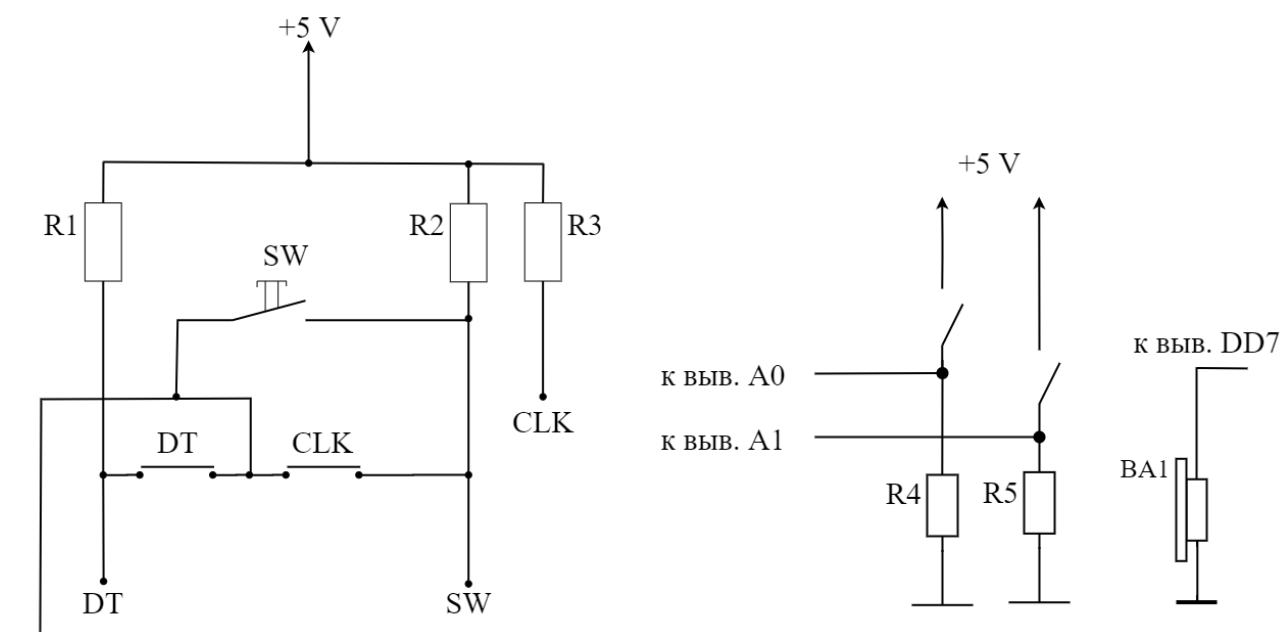


Рисунок 4 – Схема электрическая принципиальная блока управления и задач, кнопок и звукового излучателя

По окончании измерения в режиме таймер с цифрового порта 7 подается логическая единица, вследствие чего звуковой излучатель подаёт сигнал об окончании отсчёта времени.

3.4.2 Схема электрическая принципиальная блока индикации

На рисунке 3а представлена принципиальная схема блока индикации. Данный блок представляет собой жидкокристаллический дисплей 1602 подключенный по интерфейсу I2C через контроллер шины на основе микросхемы PCF8574AT.

ЖК-экран 1602 представляет собой электронный модуль, основанный на драйвере HD44780. Экран имеет 16 контактов и может работать в 4-битном режиме (с использованием только 4 линии данных) или 8-битном режиме.

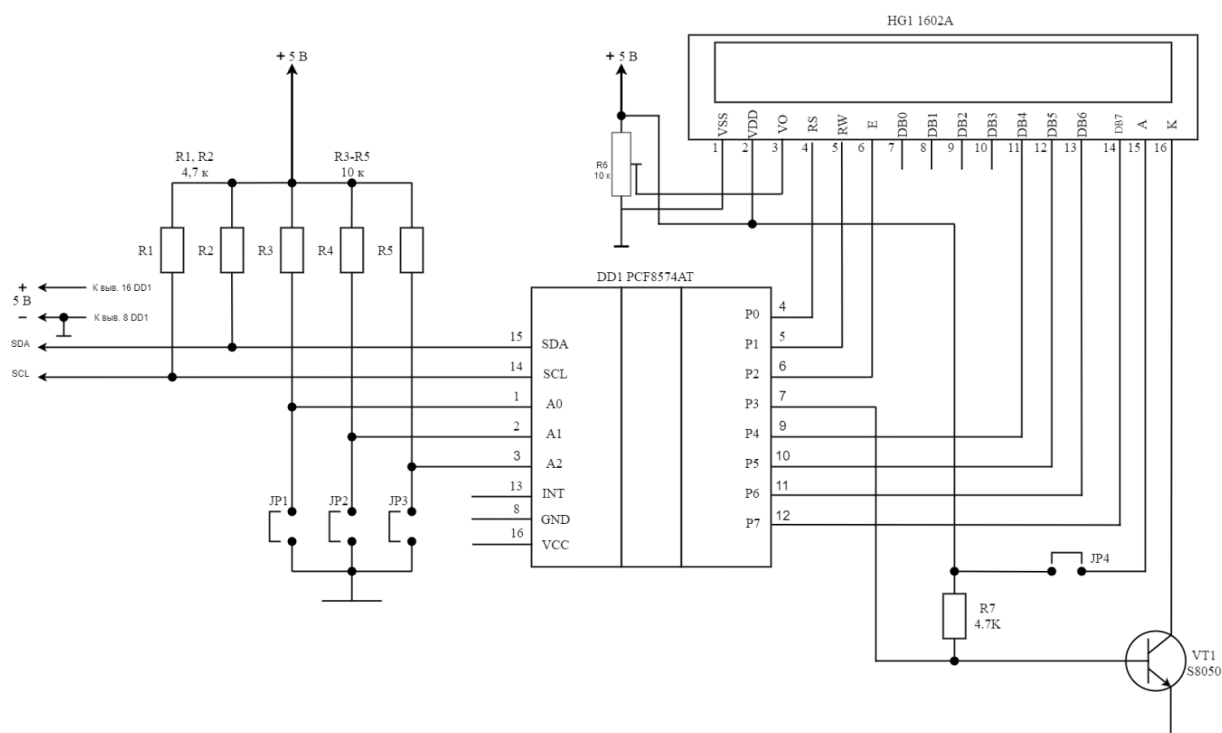


Рисунок 5 – Схема электрическая принципиальная схема блока индикации

Для передачи информации на блок индикации используется две линии для передачи данных: SDA (линия данных) и SCL (линия синхронизации) [10].

Для управления подсветкой дисплея, на плате модуля установлен транзистор, подключенный к линии P3 микросхемы. Таким образом, 3-й бит в байте данных управляет подсветкой, 0 – выключена, 1 – включена. [11]

3.4.3 Схема электрическая принципиальная блока микроконтроллера

7. Принципиальная схема блока микроконтроллера представлена на рисунке

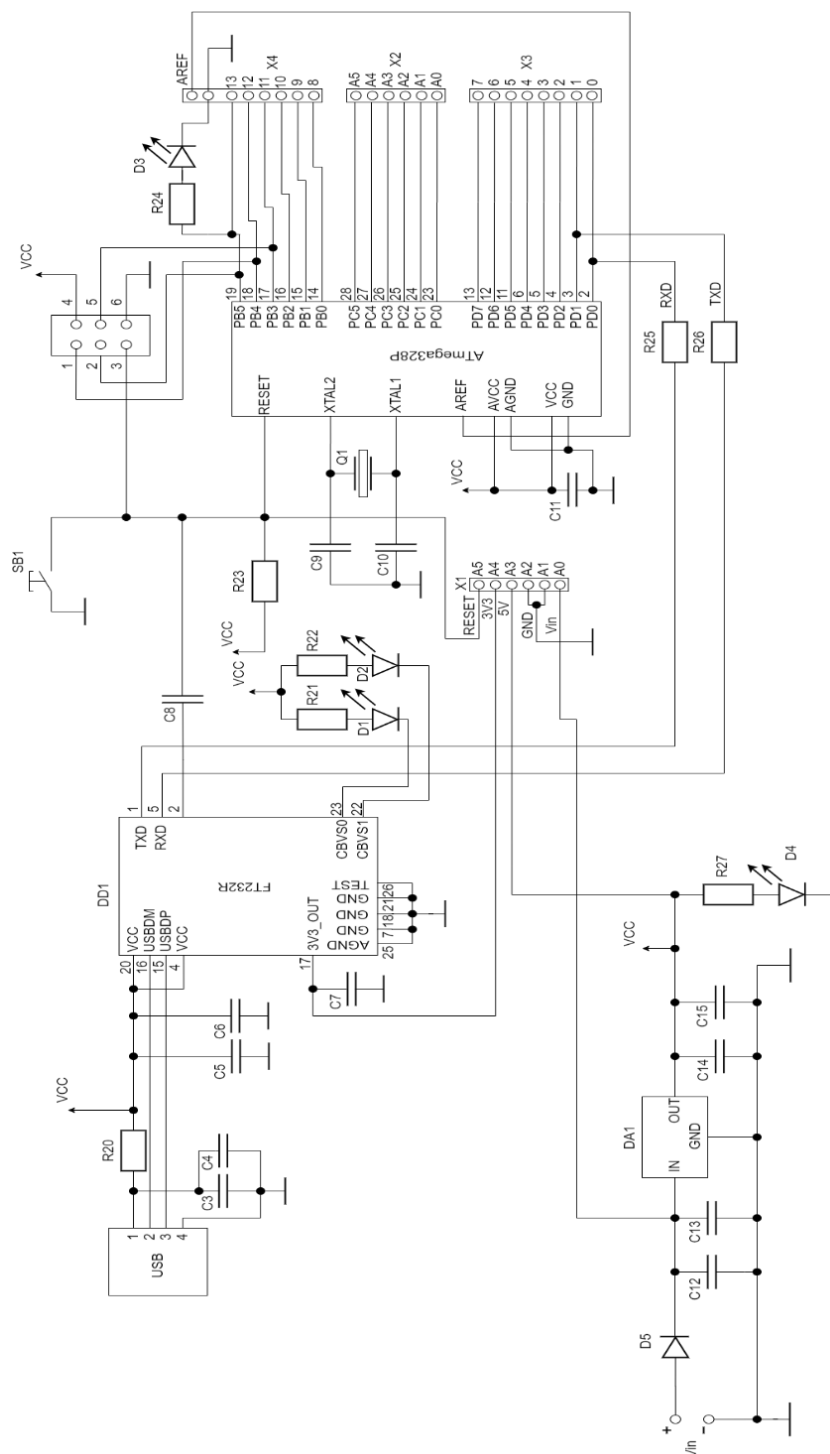


Рисунок 6 — Схема электрическая принципиальная схема блока микроконтроллера

Ввиду ограничения объёма данной дипломной работы и наличию описанию принципа работы блока микроконтроллера по схемам электрическим

структурной и функциональной. Работа по схеме электрической принципиальной блока микроконтроллера не рассматривается. Схема электрическая принципиальная приведена на рисунке 6.

4. КОНСТРУКЦИЯ МОДЕРНИЗИРОВАННОЙ ЛАБОРАТОРНОЙ УСТАНОВКИ “МАШИНА АТВУДА”

Для сравнительного анализа проведена лабораторная работа №4 “Изучение прямолинейного движения с помощью машины Атвуда” на лабораторной установке до модернизации и после модернизации, результаты измерений занесены в таблицу 3 и таблицу 4 соответственно. Сравнение значений и погрешностей будет производиться на примере упражнения № 1. “ Определение ускорения равноускоренного движения”.

В данном упражнении лабораторной работы ускорение определяется экспериментально. Если на один из грузов положить перегрузок m_1 , то система начнет двигаться равноускоренно, и груз с перегрузок пройдет путь S_1 . На кольцо Р перегрузок снимется и дальше система будет двигаться равномерно. Секундомер в данной ситуации измеряет время t равномерного движения груза. Зная путь S_2 , который проходит груз, двигаясь равномерно, можно определить скорость:

$$V = \frac{S_2}{t} \quad (1)$$

Эту скорость груз приобрел, двигаясь равноускорено, следовательно.

$$S_1 = \frac{V^2}{2a} \quad (2)$$

Откуда

$$a = \frac{V^2}{2S_1}$$

Или

$$a = \frac{S_2^2}{2S_1 t^2} \quad (3)$$

Точность полученных значений определяется точностью измерения расстояния S_1, S_2 , а также времени t . В обеих установках расстояние S_1 задается одним способом, отсчёт в модернизированной установке расстояния производится с помощью датчика линейного перемещения с точность 1,0 мм.

Вертикальная колонна классической машины Атвуда имеет линейку с шагом 1 см. Погрешность измерения при измерении расстояния в классической машине Атвуда равняется 5 мм, а в модернизированной лабораторной установке 0.5 мм.

На кольце Р перегрузок m_0 снимется и дальше груз m будет двигаться равномерно. Миллисекундомер СТС-1 на классической машине Атвуда имеет цену деления равную 0.001 секунду, модернизированная лабораторная установка имеет цену деления равную 0.0001 секунду.

Таблица 3 полученных измерений установки до модернизации:

№	m, кг	m_0 , кг	M, кг		$\langle t \rangle$, с			
1	0.1404	0.0188	0.13095	2.084	2.017	0.6	0.276	0.514963
				1.842			0.354	
				2.150			0.260	
				2.023			0.293	
				1.984			0.305	
1	0.1404	0.0188	0.13095	1.784	1.735	0.5	0.314	0.514963
				1.652			0.366	
				1.657			0.364	
				1.790			0.312	
				1.792			0.311	
2	0.1404	0.0153	0.13095	1.740	1.7564	0.6	0.396	0.423149
				1.718			0.407	
				1.734			0.399	
				1.835			0.356	
				1.755			0.390	
2	0.1404	0.0153	0.13095	1.506	1.5340	0.5	0.441	0.423149
				1.593			0.394	
				1.562			0.410	
				1.521			0.432	
				1.488			0.452	
3	0.1404	0.0147	0.13095	2.212	2.1342	0.6	0.245	0.407230
				1.969			0.310	

				2.183			0.252	
				2.387			0.211	
				1.920			0.326	
3	0.1404	0.0147	0.13095	1.687	1.7400	0.5	0.351	
				1.723			0.337	
				1.694			0.348	
				1.845			0.294	
				1.751			0.326	
4	0.1404	0.0200	0.13095	1.697	1.0560	0.6	0.417	
				1.701			0.415	
				1.593			0.473	
				1.632			0.451	
				1.657			0.437	
4	0.1404	0.0200	0.13095	1.453	1.4114	0.5	0.474	
				1.396			0.513	
				1.427			0.491	
				1.389			0.518	
				1.392			0.516	
								0.546038

Таблица 4 полученных измерений установки после модернизации:

№	m, кг	m_i , кг	M, кг		$\langle t \rangle$, с			
1	0.1404	0.0188	0.13095	2.0835	2.039	0.6	0.2764	
				1.8544			0.3490	
				2.1236			0.2661	
				2.1453			0.2607	
				1.9867			0.3040	
1	0.1404	0.0188	0.13095	1.7777	1.728	0.5	0.3164	
				1.6434			0.3703	
				1.7685			0.3642	
				1.6570			0.3113	
								0.514963

				1.7922			0.3113	
2	0.1404	0.0153	0.13095	1.7678	1.7564	0.6	0.3840	0.423149
				1.6987			0.4159	
				1.7342			0.3990	
				1.8367			0.3557	
				1.7552			0.3895	
2	0.1404	0.0153	0.13095	1.5145	1.5340	0.5	0.4360	0.423149
				1.6007			0.3903	
				1.5621			0.4098	
				1.5440			0.4195	
				1.4465			0.4779	
3	0.1404	0.0147	0.13095	2.2122	2.1342	0.6	0.2452	0.407230
				1.9691			0.3095	
				2.1832			0.2518	
				2.4087			0.2068	
				1.9241			0.3241	
3	0.1404	0.0147	0.13095	1.6874	1.7400	0.5	0.3512	0.407230
				1.7224			0.3371	
				1.6941			0.3484	
				1.8455			0.2936	
				1.6751			0.3564	
4	0.1404	0.0200	0.13095	1.7004	1.0560	0.6	0.4150	0.546038
				1.7141			0.4084	
				1.5933			0.4727	
				1.6324			0.4503	
				1.6571			0.4370	
4	0.1404	0.0200	0.13095	1.4523	1.4114	0.5	0.4741	0.546038
				1.3956			0.5134	
				1.4327			0.4872	
				1.3589			0.5415	
				1.3962			0.5130	

Расчёт погрешностей результатов измерений существующей установки:

$$a = \frac{2h}{t^2} \quad (4)$$

$$\Delta a = \sqrt{\left(\frac{\partial a}{\partial h} \Delta h\right)^2 + \left(\frac{\partial a}{\partial t} \Delta t\right)^2} = \sqrt{\left(\frac{2}{t^2} \Delta h\right)^2 + \left(-\frac{4h}{t^3} \Delta t\right)^2} \quad (5)$$

$$\Delta h = \sqrt{0.005^2 + 0.005^2} = 0.00707$$

$$\Delta t = \sqrt{0.005^2 + 0.0005^2} = 0.00502$$

$$t = 2.016 \text{ (с)}$$

$$h = 0.6 \text{ (м)}$$

$$\Delta a = \sqrt{\left(\frac{2}{2.016^2} * 0.00707\right)^2 + \left(-\frac{4 * 0.6}{2.016^3} * 0.00502\right)^2} = 0.00377 \left(\frac{\text{м}}{\text{с}^2}\right)$$

$$\varepsilon = \frac{\Delta a}{< a >} \quad (6)$$

$$< a > = 0.514963 \left(\frac{\text{м}}{\text{с}^2}\right)$$

$$\varepsilon = \frac{0.00377}{0.514963} = 0.00732 = 0.732 \%$$

Расчёт погрешностей результатов измерений модернизированной установки

$$t = 2.039 \text{ (с)}$$

$$h = 0.6 \text{ (м)}$$

$$< a > = 0.514963 \left(\frac{\text{м}}{\text{с}^2}\right)$$

$$\Delta h = \sqrt{0.0005^2} = 0.0005$$

$$\Delta t = \sqrt{0.00005^2} = 0.00005$$

$$\Delta a = \sqrt{\left(\frac{2}{2.039^2} * 0.0005\right)^2 + \left(-\frac{4 * 0.6}{2.039^3} * 0.00005\right)^2} = 0.000241 \left(\frac{\text{м}}{\text{с}^2}\right)$$

$$\varepsilon = \frac{0.000241}{0.514963} = 0.000468 = 0.047 \%$$

Погрешность измерения установки до модернизации составляла 0.732%, после 0.047%, это обусловлено тем, что погрешность измерения времени составляет 0.5 мс в модернизированной установке против 5 мс до модернизации.

Стоит также отметить, что остановка отсчёта времени в модернизированной установке производится в автоматическом режиме против ручного до модернизации, что также снижает погрешность измерения.

Кроме погрешности измерения времени в точность проводимых измерений вносится погрешность измерения линейных расстояний. Модифицируемая установка имеет линейную шкалу измерения расстояния с погрешность 0.5 см. Также при измерении расстояния стоит учитывать погрешность при измерении величины. В модернизируемой установки измерение расстояния производится в автоматическом режиме, величина погрешности при таком методе равняется 0.5 мм.

Во многих упражнения при подсчёте производных величин от расстояния, таких как скорость и ускорение, приходится вычислять их значения возводя в степень расстояния или время, что в свою очередь увеличивает погрешность измерения.

Модернизация лабораторной установки позволила в 15 раз уменьшить погрешность измерений временных интервалов, расстояния. Так же разработанный блок может быть использован для любых лабораторных работах, в которых производятся измерения временных интервал, линейного, углового перемещения при помощи ручного или дистанционного управления.

5. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Так как для выполнения задачи разработки секундомера с дистанционным управлением работе был выбран микроконтроллер Atmega328P, то следует сказать, что написание программного кода для данного микроконтроллера осуществляется с помощью языка C++. В приложении Ж приведена блок-схема, отражающая структурную работу программного обеспечения, так же приложение И приведен полный программный код, под которым работает микроконтроллер. В данной главе будут описаны принципы, по которым разрабатывался программный код, его основные структурные элементы.

5.1 Первоначальные настройки

Запуск исполняющей программы в микроконтроллере начинается с функции *setup()*. В функции *setup()* задается первоначальная настройка работы микроконтроллера. В список первоначальных настроек можно отнести:

1. Инициализация библиотеки для работы с ЖК-экраном.
2. Инициализация переменных экрана, с помощью которых производится взаимодействие с экраном.
3. Инициализация портов вывода для работы с ЖК-экраном.
4. Включение и настройка яркости ЖК-экрана.
5. Инициализация адреса цифрового вывода звуковой индикации.
6. Инициализация портов ввода энкодера.
7. Инициализация функция энкодера.
8. Включение и настройка яркости ЖК-экрана.
9. Инициализация аналоговых портов кнопок.
10. Инициализация портов ввода энкодера.

Из списка приведенного выше отдельного пояснения требует настройка работы энкодера. Функция *GetEncoderState()* отвечающая за опрос состояния энкодера. Первым делом проверяют, чтобы последний опрос энкодера был не раньше, чем через 5 мс после предыдущего для уменьшения ложных срабатываний [12].

Ниже представлен программный код, отвечающий за работу с энкодером.

```
eEncoderState GetEncoderState(){  
    eEncoderState Result = eNone;
```

```

CurrentTime = millis();
if (CurrentTime - LastTime >= 5){
    LastTime = CurrentTime;
    if (digitalRead(pin_Btn) == LOW){
        if (ButtonPrev){
            Result = eButton;
            ButtonPrev = 0;
        }
    }else{
        ButtonPrev = 1;
        EncoderA = digitalRead(pin_DT);
        EncoderB = digitalRead(pin_CLK);
        if ((!EncoderA) && (EncoderAPrev)){
            if (EncoderB)
                Result = eRight;
            else
                Result = eLeft;
        }
        EncoderAPrev = EncoderA;
    }
}
return Result;
}

```

Управляющим сигналом на кнопке энкодера является логический 0. В случае получения логической 1 на условии `digitalRead(pin_Btn) == LOW` это соответствует нажатию кнопки. Проверка `ButtonPrev` служит для нейтрализации ложных срабатываний. Так как функция работы с энкодером опрашивает его с периодичностью не чаще 5 мс может произойти ситуация, что кнопку была зажата гораздо дольше чем на 5 мс. При отпускании кнопки и новом вызове функции получится логический 0 на условии `digitalRead(pin_Btn) == LOW`, что в следствии изменит флаг `ButtonPrev` на противоположный и позволит программному коду распознавать новые нажатия [13].

Проверка значения флага `EncoderB` позволяет определить направление вращения вала энкодера.

5.2 Выбор режима работы

Для выбора режима работы (“СЕКУНДОМЕР” или “ТАЙМЕР”) функция loop() вызывает следующий программный код:

```
switch (GetEncoderState()){  
    case eNone:  
        return;  
    case eButton:  
        isEncoderButtonPressed = true;  
        break;  
    case eLeft:  
        counter--;  
        prtintTitle();  
        printOption();  
        isEncoderButtonPressed = false;  
        break;  
    case eRight:  
        counter++;  
        prtintTitle();  
        printOption();  
        isEncoderButtonPressed = false;  
        break;  
}
```

Программная функция GetEncoderState() возвращает одно из 4 значений:

1. eNone – отсутствие значения.
2. eButton – нажатие кнопки энкодера.
3. eLeft – вращение оси вала энкодера против часовой стрелки.
4. eRight - вращение оси вала энкодера по часовой стрелки.

После получения нового состояния энкодера программный код выполняет несколько код:

1. eNone – отсутствие нового состояния, программный код игнорирует данную итерации.
2. eButton – нажатие кнопки энкодера приведет к изменению глобального флага isEncoderButtonPressed в значение true.

3. eLeft - вращение оси вала энкодера против часовой стрелки приведет к уменьшению глобального фразы counter на единицу и вызова функций prtintTitle(), printOption().
4. eRight - вращение оси вала энкодера по часовой стрелки приведет к увеличению глобального фразы counter на единицу и вызова функций prtintTitle(), printOption().

Функции printTitle() и printOption() отвечают за отрисовки текстово-цифровой информации о выбранном режиме. Программный код функции printTitle():

```
if (counter > 1 || counter < 0){  
    counter = 0;  
}  
lcd.setCursor(0, 0);  
lcd.clear();  
lcd.print(L"РЕЖИМ");
```

В данной функции проверяется значения глобальной переменной counter, так как в данной лабораторной установке имеется выбор двух режимов работы, значение больше 1 или меньше 0 являются не корректными, и они будут изменены на 0. Значения переменной counter равное 0 соответствует режиму “СЕКУНДОМЕР”, значение равное 1 соответствует режиму “ТАЙМЕР”.

После проверки на невалидные значения программный код устанавливает курсор вывода текстово-цифровой информации верхней строки, а также стирает старые значение и на их места записывает новое значение.

Программный код функции printOption() также как и printTitle() содержит проверка на не валидные значения. Программный код функции printOption():

```
lcd.setCursor(0, 1);  
lcd.clear();  
  
if(counter < 0 || counter > 1){  
    counter = 0;  
}
```

```

switch (counter){

    case 0:
        lcd.print(L"0. СЕКУНДОМЕР");
        break;

    case 1:
        lcd.print(L"1. ТАЙМЕР");
        break;

```

Выбор определенной строки для вывода опирается на значения counter. В случае значения переменной 0 – “0. СЕКУНДОМЕР”, в случае значения 1 – “1. ТАЙМЕР”.

Переход к выбранному режиму осуществляется в теле функции loop() при значении глобального флага isEncoderButtonPressed в true:

```

if (isEncoderButtonPressed){
    goToOption();
    isNeedPreView = true;
}

```

Программный код функции goToOption() отвечает за выбор режима по значению глобального флага counter и после завершения работы режима возвращению глобальным флагам исходных значений с помощью функции exitFromOption(). Программный код функции goToOption():

```

switch (counter){
    case 0:
        stopwatch();
        exitFromOption();
        break;

    case 1:
        timer();

```

```
exitFromOption();  
break;  
}
```

5.3 Принцип работы режима “СЕКУНДОМЕР”

Принцип работы режима “СЕКУНДОМЕР” будет рассматриваться с помощью блок-схемы программного кода. Весь программный код и блок-схему программы можно посмотреть в приложении Ж и приложении И соответственно.

После выбора режима “СЕКУНДОМЕР” первым делом программный код проверяет состояния кнопки “ВЫХОД”. В случае получения результата равной логической 1 программный код выходит из режима “СЕКУНДОМЕР” и возвращается в меню выбора режима. В случае логического 0 программный код выполняет проверку состояния кнопки “ПУСК”, если кнопка не нажата, то программный код возвращается к проверки первого условия, до тех пор, пока либо не будет нажата кнопка “ПУСК” или “ВЫХОД” [14].

После нажатия кнопки “ПУСК” программный код запоминает время начало отсчёта и проверяет, нажата ли кнопка “ВЫХОД”. В случае не нажатия кнопки программа подсчитывает время работы секундомера и выводит новое значения на блок индикации. В случае нажатия кнопки подсчёт времени останавливается, выводятся последние данные, и программный код возвращается в начало режима “СЕКУНДОМЕР” [15].

5.4. Принцип работы режима “ТАЙМЕР”

Принцип работы режима “ТАЙМЕР” будет рассматриваться с помощью блок-схемы программного кода. Весь программный код и блок-схему программы можно посмотреть в приложении Ж и И соответственно [16].

После выбора режима “ТАЙМЕР” первым делом программный код проверяет состояния кнопки “ВЫХОД”. В случае получения результата равной логической 1 программный код выходит из режима “ТАЙМЕР” и возвращается в меню выбора режима. В случае логического 0 программный код выполняет проверку состояния кнопки “ПУСК”, в результате чего программный код имеет развилку. В случае логического 0 опрашивается энкодер для получения его состояния:

- Энкодер повернули по часовой стрелке, значит увеличение значения времени на шаг.
- Энкодер повернули против часовой стрелки, значит уменьшение значения времени на шаг.
- Нажата кнопка энкодера – выход из режима и возвращение в меню выбора режима.
- Отсутствие состояния – программа возвращается в начало режима “ТАЙМЕР” и программный код выполняется заново.

В случае получения логической 1 при проверке состояния нажатия кнопки “ПУСК” программный код определяет значения времени, при котором работу таймер стоит прекратить. С периодичностью проверяется оставшееся время работы таймера, в случае положительного ответа разница во времени между текущим моментом времени и моментом отключения таймера вычисляется и выводится на блок индикации. После окончания работы режима “ТАЙМЕР” производится звуковой сигнал, и буквенно-цифровая индикация об окончании отсчёта времени и система возвращается в первоначальное состояние [17].

6. ОБЩИЕ ТРЕБОВАНИЯ БЕЗОПАСНОСТИ.

1. Общие сведения о технике безопасности

1.1 К лабораторным и практическим работам допускаются студенты, прошедшие в университете обучение и проверку знаний по требованиям безопасности при проведении лабораторных и практических работ, которые оформлены соответствующим протоколом, освоившие безопасные методы и приемы выполнения работ.

1.2. Категорически запрещается приходить в лабораторию в верхней одежде и приносить с собой вещи и предметы, загромождающие рабочие столы. На рабочих столах должны находиться только описания лабораторных работ, приборы и соединительные провода.

1.3. В лаборатории запрещается громко разговаривать, самовольно покидать рабочие места.

1.4. Перед выполнением лабораторных и практических работ студент обязан пройти обучение мерам безопасности при ведении экспериментов с использованием электротехнических средств обучения.

1.5. Будьте внимательны и дисциплинированы, точно выполняйте указания преподавателя.

1.6. Не допускается:

- выполнять действия, не предусмотренные методикой проведения лабораторной или практической работы;
- самостоятельно производить какие-либо переключения на электрических стендах лаборатории;
- умышленная порча оборудования;

1.7. Студент обязан:

- выполнять настоящие требования безопасности;
- соблюдать правила пожарной и электробезопасности;
- соблюдать порядок и чистоту в лаборатории;
- уметь оказывать первую доврачебную помощь.

2. Требования безопасности перед началом лабораторной или практической работы

2.1. Перед выполнением лабораторной или практической работы необходимо внимательно изучить ее содержание и последовательность выполнения. Приступая к работе с электротехническими устройствами, помните

об опасности поражения электрическим током, будьте осторожны, внимательны и дисциплинированы.

2.2. Перед началом выполнения работы следует ознакомиться с приборами и стендами, способами их включения, эксплуатации, выключения и убедиться, что все выключатели находятся в положении «Выключено».

2.3. Размещать приборы, материалы, оборудование на месте проведения работы следует таким образом, чтобы исключить их падение или опрокидывание.

2.4. Провести осмотр аппаратуры и приборов, убедиться в отсутствии у них повреждений.

2.5. Не допускается использование приборов и аппаратов с неисправными клеммами, проводов с поврежденной изоляцией, неисправных реостатов, патронов ламп, тумблеров и др.

2.6. При обнаружении неисправностей в приборах и оборудовании необходимо немедленно сообщить о них преподавателю или лаборанту и до их устранения к работе не приступать.

Не разрешается самовольно производить какой либо ремонт оборудования и аппаратуры.

2.7. Все электрическое оборудование, расположенное в лаборатории, должно быть подключено к устройству защитного отключения (УЗО).

2.8. Не приступайте к выполнению лабораторной или практической работы без разрешения преподавателя.

3. Требования безопасности при выполнении лабораторной или практической работы

При работе в лаборатории с электрическим током необходимо соблюдать следующие правила:

3.1. Запрещается нагружать измерительные приборы выше предельных значений, обозначенных на их шкалах.

3.2. Используйте провода с наконечниками и предохранительными изоляционными чехлами. Убедитесь, что их изоляция не имеет повреждений.

3.3. При сборке электрических цепей избегайте пересечения проводов.

3.4. Запрещается пользоваться выключателями с открытыми контактами при напряжениях выше 42 В.

3.5. При подключении установок к сети переменного тока напряжением 220 В необходимо использовать только штепсельные соединения.

3.6. Сборку и разборку, внесение изменений в цепь можно производить только при отключенном источнике питания. Источник электропитания подключается к собранной электрической цепи в последнюю очередь.

Собранную цепь можно включать только после проверки и с разрешения преподавателя или лаборанта.

3.7. Не прикасайтесь к находящимся под напряжением элементам цепей, лишенным изоляции, к корпусам стационарного электрооборудования, к зажимам отключенных конденсаторов. Разряд конденсатора производить с помощью изолированного проводника.

3.8. Наличие напряжения в цепи проверяйте только с помощью приборов или указателей напряжения.

3.9. Обнаружив неисправность в электрических устройствах, необходимо немедленно отключить источник электропитания.

3.10. До включения электро- и радиоприборов в сеть необходимо убедиться в соответствии положения переключателя сетевого напряжения его номинальной величине, а также в исправности предохранителей.

3.11. Присоединять однополюсную вилку (щуп) электроизмерительного прибора к цепи следует только одной рукой, причем вторая рука не должна касаться шасси, корпуса прибора и других электропроводящих предметов.

3.12. При настройке и регулировке включенного радиоустройства (подстройка контуров, регулировка переменных конденсаторов или резисторов) необходимо пользоваться инструментом с надежной изоляцией.

3.13. При эксплуатации осциллографов необходимо с особой осторожностью обращаться с электронно-лучевой трубкой. Недопустимы удары по трубке или попадание на нее расплавленного припоя, так как это может вызвать взрыв.

3.14. При появлении запаха гари, искрении, перегреве деталей следует немедленно отключить устройство от источника электропитания.

3.15. При выполнении работы запрещается:

- проводить какие либо переключения на главном распределительном щите лаборатории;
- прикасаться к панелям с оголенными токоведущими частями, находящимися под напряжением;
- прикасаться к конденсатору даже после его отключения т электрической цепи до его полной разрядки;

- прислоняться к лабораторным стендам и опираться на них при снятии показаний приборов.

3.16. Все необходимые переключения нужно проводить только при выключенном напряжении.

3.18. Любое изменение в цепи должно быть проверено преподавателем.

3.19. При проведении опытов на испытательной панели стенда, находящейся под напряжением, все переключения и регулировки с помощью переключателей, включение и выключение тумблеров и тому подобные операции должны проводиться одним человеком и только одной рукой. Вторая рука должна быть свободной и не должна касаться аппаратуры

3.20. Не разрешается оставлять без присмотра и переносить аппаратуру, включенную в электрическую сеть.

4. Требования безопасности в аварийных ситуациях

4.1. Обнаружив неисправность в электрических устройствах, находящихся под напряжением, при появлении дыма, искрения, или специфического запаха перегретой изоляции, немедленно отключите источник электропитания и сообщите об этом преподавателю. Необходимо четко выполнять его указания, при необходимости следует организованно, без паники эвакуироваться из помещения согласно планам эвакуации. Для тушения пожара в лаборатории имеется огнетушитель.

4.2. В случае поражения человека электрическим током следует немедленно отключить групповой щиток. В случае, когда отключение напряжения не может быть произведено быстро, нужно принять меры к отдалению пострадавшего от частей оборудования, находящихся под напряжением. Для этого необходимо воспользоваться диэлектрическими перчатками, диэлектрическими ковриком, а при их отсутствии можно применять такие подручные средства, как сухая одежда, сухие доски, палки и другие хорошие изоляторы. Далее следует сразу же приступить к оказанию первой доврачебной помощи.

Во всех случаях поражения электрическим током необходимо вызвать врача независимо от состояния пострадавшего по тел. 103.

4.3. В случае ухудшения состояния здоровья (отравление, потери сознания и т.п.) студент (если в состоянии) или другие студенты должны немедленно поставить в известность преподавателя, который обязан оказать первую доврачебную помощь и, при необходимости, вызвать скорую медицинскую помощь и организовать отправку пострадавшего в медицинское учреждение.

5. Требования безопасности по окончании работы

5.1. Отключите источник электропитания, разрядите конденсаторы, после чего разберите электрическую цепь. Приступайте к разбору схемы с разрешения преподавателя.

5.2. Разобрав схему, необходимо сложить соединительные провода, а приборы установить на указанные места, место проведения лабораторной работы необходимо привести в порядок и предъявить преподавателю.

5.3. Обо всех недостатках, обнаруженных во время проведения лабораторной или практической работы, следует поставить в известность преподавателя или лаборанта.

ЗАКЛЮЧЕНИЕ

В процессе выполнения дипломной работы изготовлено многофункциональное устройство для измерения временных интервалов на микроконтроллере с возможностью дистанционного управления начала и конца счёта времени.

Многофункциональный блок имеет следующие режимы работы:

1. Измерение времени с дистанционным управлением.
2. Измерение периода колебательных процессов, по сигналам от внешних устройств.
3. Секундомер с ручным и дистанционным управлением.
4. Таймер с ручной установкой отсчитываемого времени.

В данное устройство было дополнительно введено устройства линейного и углового перемещения (Дипломная работа Вороновича), что позволило задавать расстояния перемещения тел, измерять их линейное и угловое перемещение.

Разработанное устройство позволило модернизировать лабораторную установку “Машина Атвуда”, что привело к уменьшению погрешности измерения прибора при отсчёте времени в 10 раз, с 5 мс до 0.5 мс, при подсчете расстояния в 10 раз, с 5 мм до 0.5 мм. В итоге погрешность определения физических величин, таких как время и расстояние, при выполнении лабораторной работы уменьшилась в 15.6 раз с 0.733% до 0.047%.

Следует отметить, что была спроектирована структурная, функциональная, принципиальная схема, которая полностью отвечает задачам, поставленным в данной дипломной работе.

Также, разработано программное обеспечение, достоинством которого является использование встроенных программных решений. Это позволяет гарантировать бесперебойность и стабильную работу устройства.

Поскольку разработанное устройство является универсальным, благодаря нему можно в значительной степени упростить, ускорить, а кроме того увеличить точность большого количества учебно-исследовательских работ, связанных с измерением интервалов времени.

Также стоит отметить, что разработанный блок может быть использован и как дополнительный инструмент в работах, где требуется измерение линейных и угловых передвижений тел [6-8]. Это позволит еще больше расширить диапазон

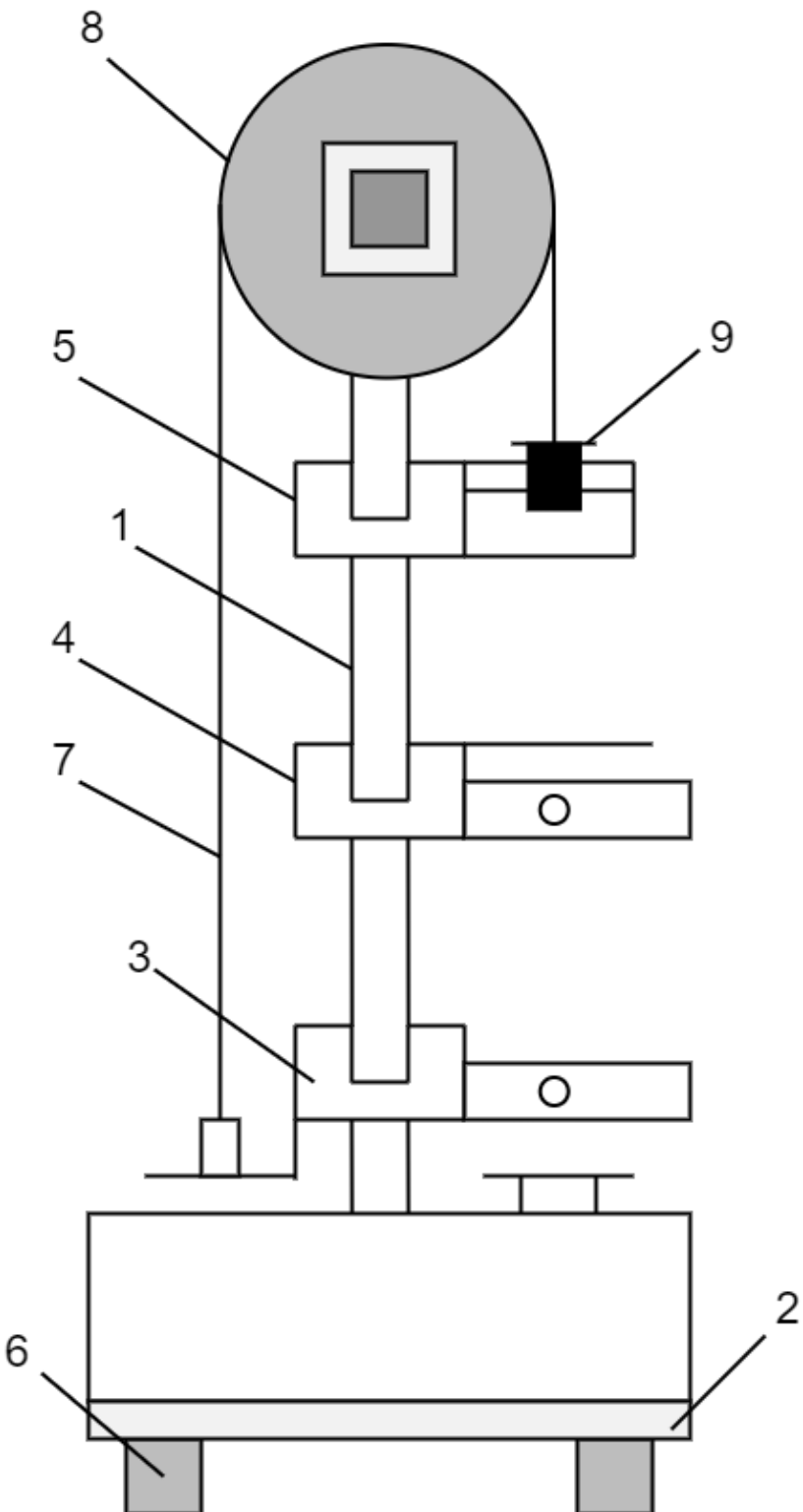
выполняемых лабораторных работ на кафедре общей физики ГрГУ им. Янки Купалы.

СПИСОК ЛИТЕРАТУРЫ

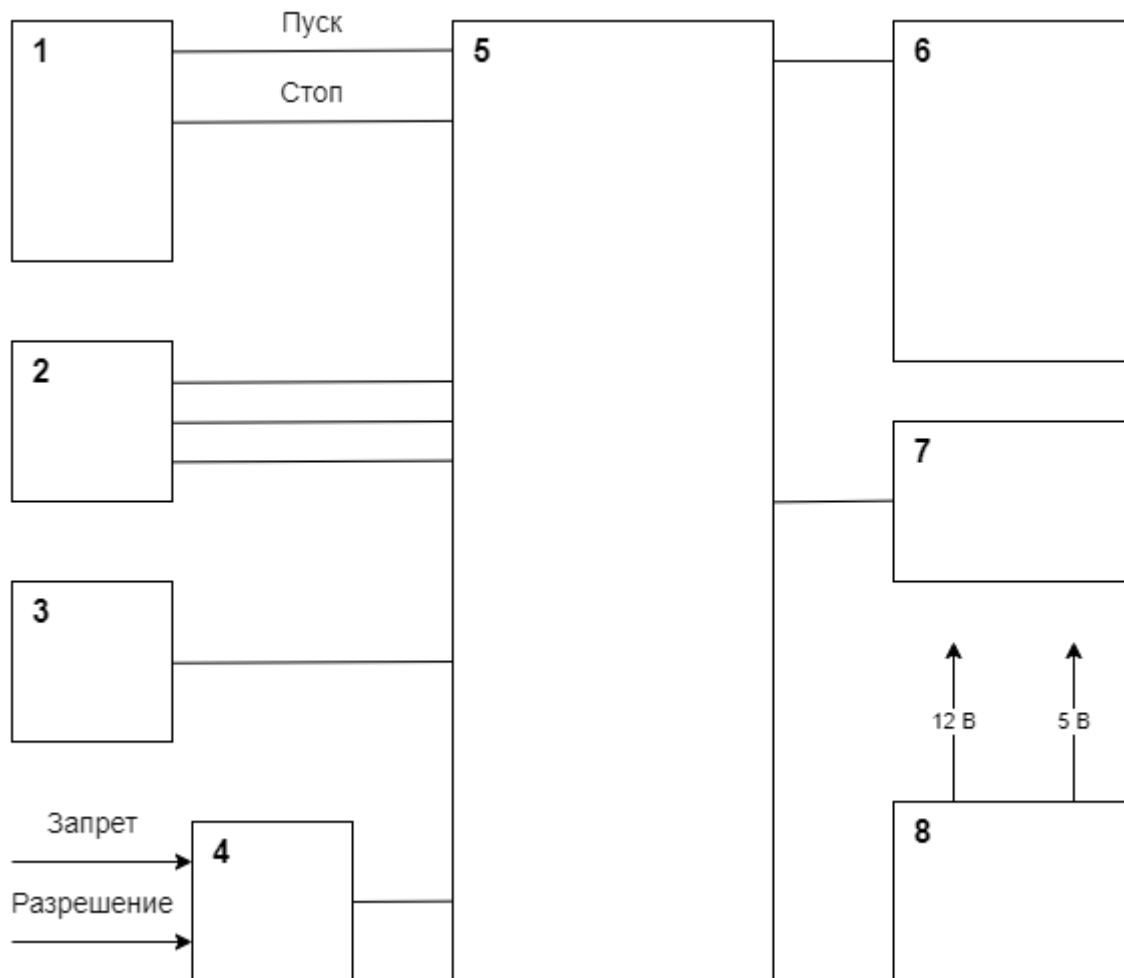
1. Atmel 8-bit Microcontroller with 4/8/16/32KBytes InSystem Programmable Flash [Электронный ресурс], 2013. – 31с. – Режим доступа: <https://www.farnell.com/datasheets/1693866.pdf>. – Дата доступа: 24.11.2019.
2. PIC16F631/677/685/687/689/690 Data Sheet [Электронный ресурс], 2006. – 294с. – Режим доступа: <http://ww1.microchip.com/downloads/en/devicedoc/41262c.pdf>. – Дата доступа:
3. Low & medium-density value line, advanced ARM®-based 32-bit MCU with 16 to 128 KB Flash, 12 timers, ADC, DAC & 8 comm interfaces [Электронный ресурс], 2016. – 96с. – Режим доступа: <https://www.st.com/resource/en/datasheet/stm32f100cb.pdf>. Дата доступа: 14.11.2019.
4. Герман, А.Е. Микроконтроллеры семейства Intel MCS-51: учебное пособие / А.Е. Герман- 2003. – С. 20-34.
5. Баранов, В.Н. Применение микроконтроллеров AVR : схемы, алгоритмы, программы: учебник / В.Н. Баранов – 2006. – С. 12-16.
6. Arduino Playground – RotaryEncoders [Электронный ресурс]. – Режим доступа: <https://playground.arduino.cc/Main/RotaryEncoders/>. - Дата доступа: 03.04.2020.
7. Rotary Encoders & Artmega: An Artmega Rotary Encoder Tutorial | Arrow.com | Arrow.com [Электронный ресурс]. – Режим доступа: <https://www.arrow.com/en/research-and-events/articles/rotary-encoders-how-to-pair-with-an-arduino-board>. Дата доступа: 17.02.2020.
8. Using Rotary Encoders with Arduino | DroneBot Workshop [Электронный ресурс] – Режим доступа: <https://dronebotworkshop.com/rotary-encoders-arduino/>. Дата доступа: 13.02.2020.
9. Character I2C LCD with Artmega Tutorial [Электронный ресурс] – Режим доступа: <https://www.makerguides.com/character-i2c-lcd-arduino-tutorial/>. Дата доступа: 03.04.2020.
10. LCD 1602 With Artmega Uno R3 [Электронный ресурс]- Режим доступа: <https://www.instructables.com/id/LCD-1602-With-Arduino-Uno-R3/>. Дата доступа: 09.02.2020.
11. In-Depth Tutorial to Interface 16x2 Character LCD Module [Электронный ресурс] – Режим доступа: <https://lastminuteengineers.com/arduino-1602-character-lcd-tutorial/>. Дата доступа: 08.03.2020.

12. Подключаем энкодер [Электронный ресурс] – Режим доступа: <https://portal-pk.ru/news/243-podklyuchaem-enkoder-k-arduino.html>. Дата доступа: 12.03.2020.
13. How Rotary Encoder Works and Interface It [Электронный ресурс] – Режим доступа: <https://lastminuteengineers.com/rotary-encoder-arduino-tutorial/>. Дата доступа: 18.02.2020.
14. Физический практикум/ Под ред. В.И. Ивероной¹ В 2 ч. М.: Физматгиз, 1967 - 1968. Ч.1-2. – С. 51-54.
15. ЭВМ в курсе общей физики/Под ред. П.С. Булкин – 1982. 230 с.
16. Методы физических измерений: Лабораторный практикум по физике/Под ред. Р.И. Солоухяна – 1975. – С. 145.
17. Arduino Reference [Электронный ресурс] – Режим доступа: <https://www.arduino.cc/reference/en/language/functions/time/micros/>. Дата доступа: 07.05.2020.

Внешний вид лабораторной установки “Машина Атвуда”.

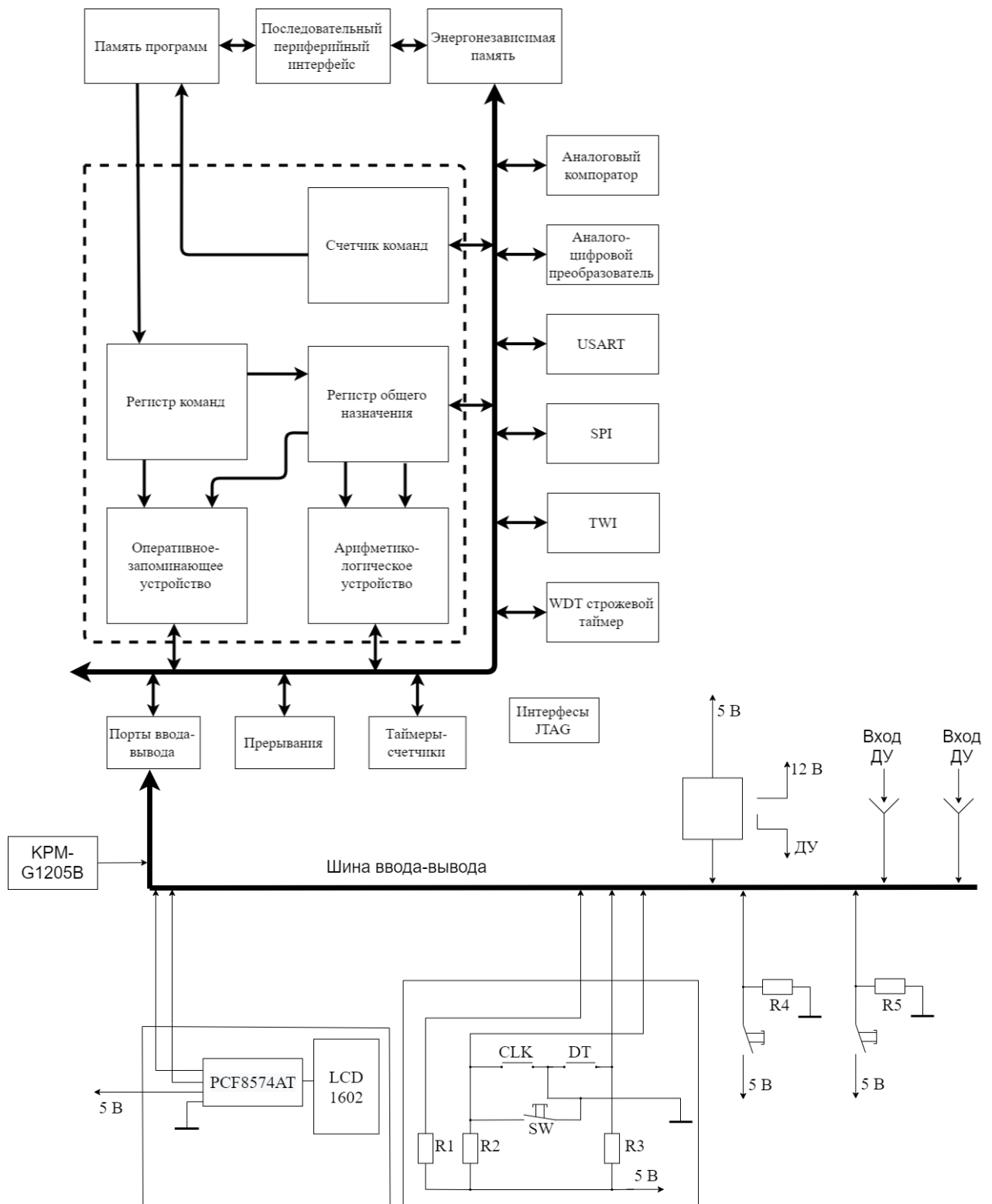


ПРИЛОЖЕНИЕ Б
Схема электрическая структурная модернизированной установки
“Машина Атвуда”.



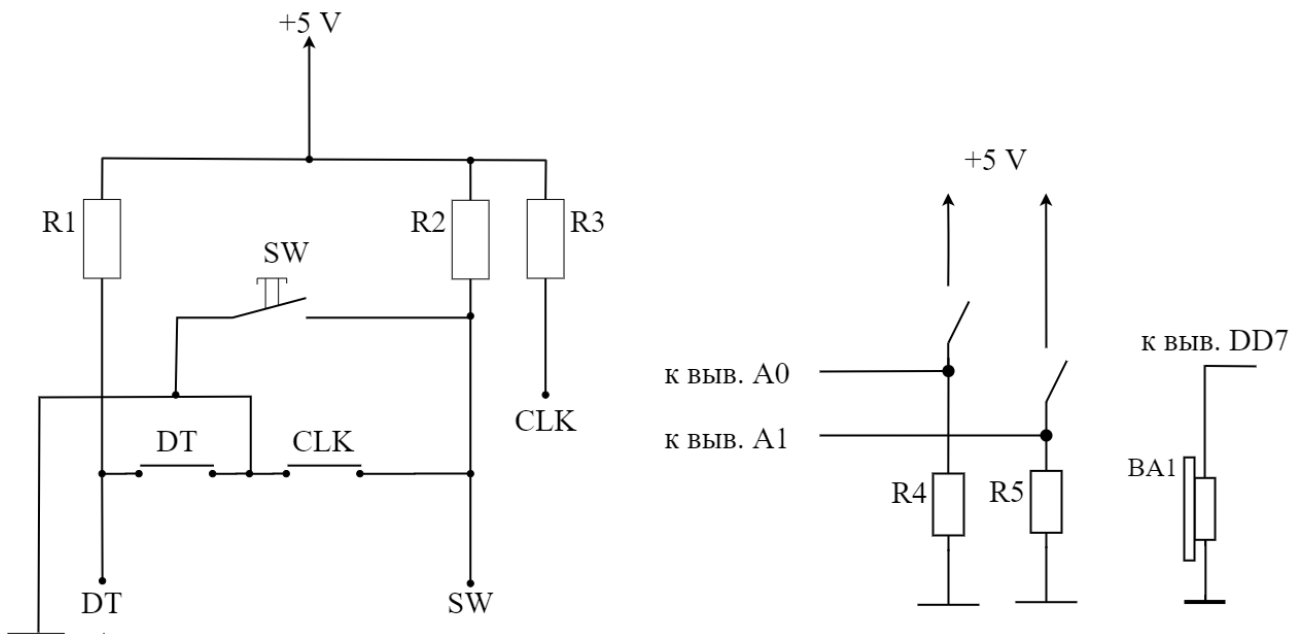
ПРИЛОЖЕНИЕ В

Схема электрическая функциональная модернизированного устройства.



ПРИЛОЖЕНИЕ Г

Схема электрическая принципиальная блока управления и задач, кнопок и звукового излучателя.



ПРИЛОЖЕНИЕ Д **Схема электрическая принципиальная блока индикации.**

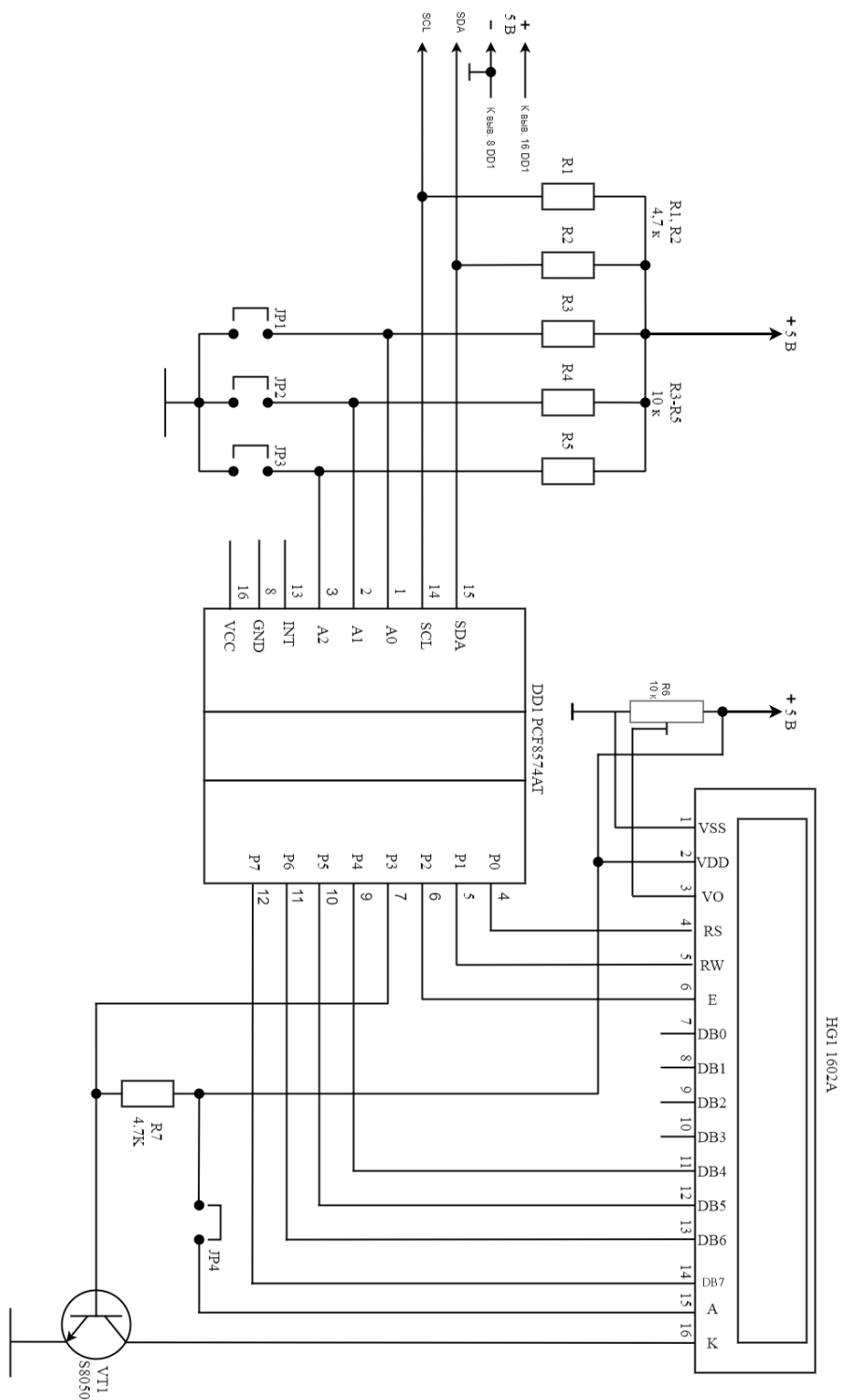
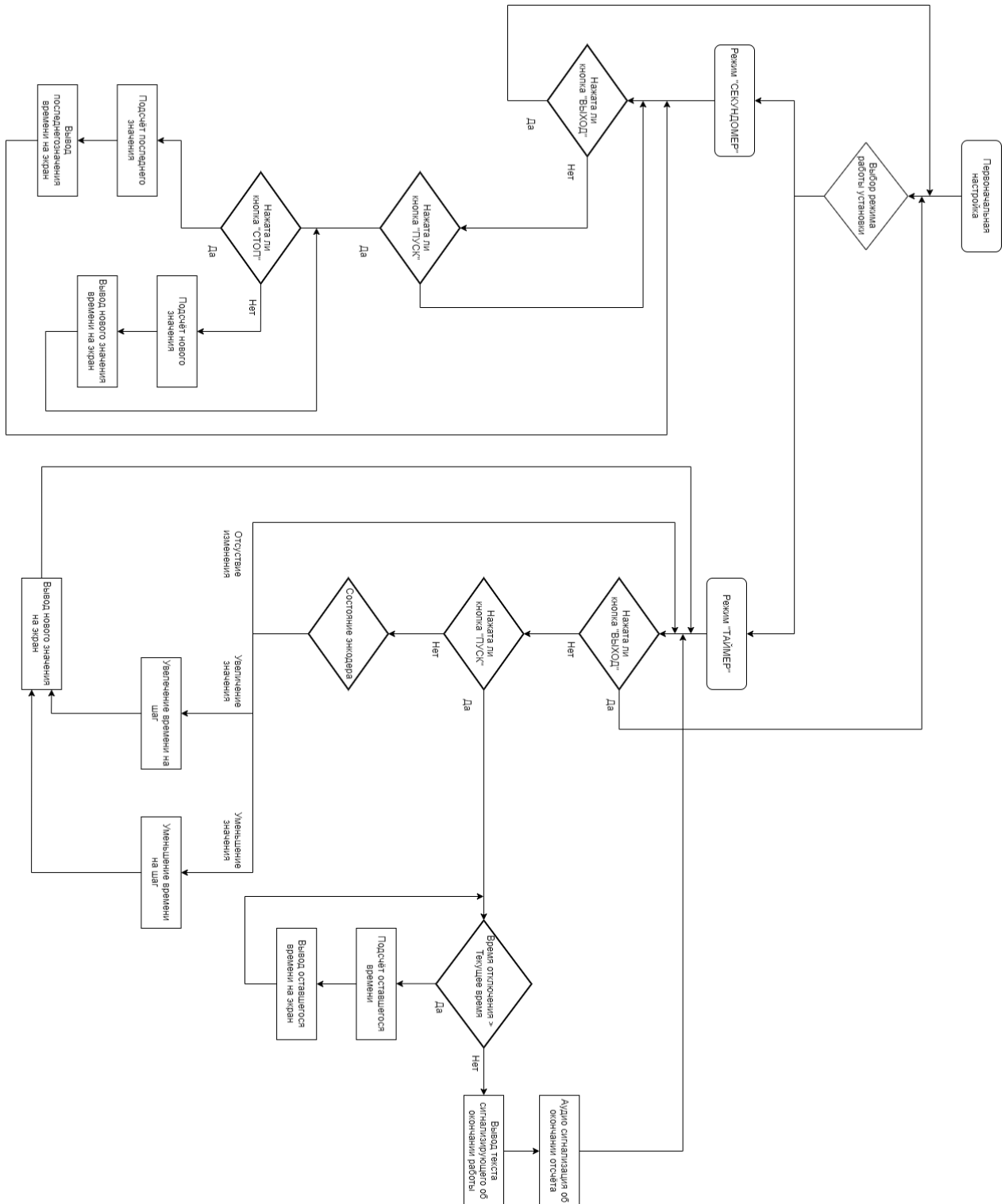


Схема электрическая принципиальная блока микроконтроллера.



ПРИЛОЖЕНИЕ Ж **Блок схема работы программного обеспечения.**



ПРИЛОЖЕНИЕ 3

Программное обеспечение.

```
#include <LCD_1602_RUS.h>

LCD_1602_RUS lcd(0x27, 16, 2);

#define pin_CLK 4

#define pin_DT 2

#define pin_Btn 9

const int pin_Sound_Signal = 3;

unsigned long CurrentTime, LastTime;

enum eEncoderState{

    eNone,

    eLeft,

    eRight,

    eButton

};

uint8_t EncoderA, EncoderB, EncoderAPrev;

int8_t counter = 0;

bool ButtonPrev;

eEncoderState GetEncoderState(){

    eEncoderState Result = eNone;

    CurrentTime = millis();

    if (CurrentTime - LastTime >= 5){

        LastTime = CurrentTime;
```

```

if (digitalRead(pin_Btn) == LOW){
    if (ButtonPrev){
        Result = eButton;
        ButtonPrev = 0;
    }
} else {
    ButtonPrev = 1;
    EncoderA = digitalRead(pin_DT);
    EncoderB = digitalRead(pin_CLK);
    if ((!EncoderA) && (EncoderAPrev)){
        if (EncoderB){
            Result = eRight;
        } else {
            Result = eLeft;
        }
    }
    EncoderAPrev = EncoderA
}
return Result;
}

void setup(){
    Wire.begin();
    lcd.init();

```

```

lcd.backlight();

pinMode(pin_DT, INPUT);

pinMode(pin_CLK, INPUT);

pinMode(pin_Btn, INPUT_PULLUP);

pinMode(pin_Sound_Signal, OUTPUT);

Serial.begin(9600);

counter = 0;

}

bool isNeedPreView = true;

bool isEncoderButtonPressed = false;

void loop(){

    if (isNeedPreView){

        prtintTitle();

        printOption();

        isNeedPreView = false;

    }

    if (isEncoderButtonPressed){

        goToOption();

        isNeedPreView = true;

    }

    switch (GetEncoderState()){

        case eNone:

            return;

        case eButton:

```

```

        isEncoderButtonPressed = true;

        break;
    case eLeft:

        counter--;

        prtintTitle();

        printOption();

        isEncoderButtonPressed = false;

        break;
    case eRight:

        counter++;

        prtintTitle();

        printOption();

        isEncoderButtonPressed = false;

        break;
    }

    delay(100);
}

void encoder(){

    switch (GetEncoderState()){

        case eButton:

            isEncoderButtonPressed = true;

            break;

        case eLeft:

            counter++;

```

```

        isEncoderButtonPressed = false;

        break;

    case eRight:

        counter--;

        isEncoderButtonPressed = false;

        break;

    case eNone:

        isEncoderButtonPressed = false;

        return;

    }

}

void prtintTitle(){

    if (counter > 1 || counter < 0){

        counter = 0;

    }

    lcd.setCursor(0, 0);

    lcd.clear();

    lcd.print(L"РЕЖИМ");

}

void printOption(){

    lcd.setCursor(0, 1);

    if(counter < 0 || counter > 1){

        counter = 0;

    }

}

```

```

switch (counter){
    case 0:
        lcd.print(L"0. СЕКУНДОМЕР");
        break;

    case 1:
        lcd.print(L"1. ТАЙМЕР");
        break;
}
}

void goToOption(){
    switch (counter){
        case 0:
            stopwatch();
            exitFromOption();
            break;
        case 1:
            timer();
            exitFromOption();
            break;
    }
}

void exitFromOption(){
    counter = 1;

```

```

encoder();

delay(500);

}

void stopwatch(){

    const byte analogPinForStartButtonOfStopWatch = 0;

    const byte analogPinForStopButtonOfStopWatch = 1;

    long timeOfStopWatch = 0;

    long currentTime = 0;

    bool isNeedPreView = true;

    do {

        if (isNeedPreView) {

            printPreViewForStopWatch();

            isNeedPreView = false;

        }

        if (isAnalogButtonPressed(analogPinForStartButtonOfStopWatch)) {

            timeOfStopWatch = micros();

            while (!isAnalogButtonPressed(analogPinForStopButtonOfStopWatch)) {

                currentTime = micros() - timeOfStopWatch;

                printTimeForStopWatch(currentTime);

                delay(100);

            }

            printResultTimeForStopWatch(currentTime);

            delay(2500);

            isNeedPreView = true;

```

```

    }

    encoder();

} while (!isEncoderButtonPressed);

}

void printTimeForStopWatch(long timeToPrint){

    lcd.setCursor(0, 0);

    lcd.clear();

    lcd.print(L"ВРЕМЯ ");

    long second = timeToPrint / 1000000;

    long millisecond = timeToPrint - second * 1000000;

    millisecond = millisecond / 100;

    lcd.print(String(second));

    lcd.print(".");

    lcd.print(String(millisecond));

    lcd.setCursor(0, 1);

    lcd.print(L"    секунд    ");

}

void printPreViewForStopWatch() {

    lcd.setCursor(0, 0);

    lcd.clear();

    lcd.print(L"    ДЛЯ НАЧАЛА    ");

    lcd.setCursor(0, 1);

    lcd.print(L" НАЖМИТЕ СТАРТ ");

}

```



```

void printResultTimeForStopWatch(long timeToPrint){
    lcd.setCursor(0, 0);
    lcd.clear();
    lcd.print(L"  PE3YJbTAT: ");
    long second = timeToPrint / 1000000;
    long millisecond = timeToPrint - second * 1000000;
    millisecond = millisecond / 100;
    lcd.setCursor(0, 1);
    lcd.print(String(second));
    lcd.print(".");
    lcd.print(String(millisecond));
    lcd.print(L" сек.");
}

void timer() {
    const int countOfSound = 2;
    const int delayBetweenSound = 500;
    byte analogPinForStartButtonOfTimer = 0;
    double stepForTimer = 0.5;
    double setTimeForTimer = 0;
    double currentTime = 0;
    double needTimeForTimer = 0;
    bool isNeedPreView = true;
    counter = 0;
    int last = counter;

```

```

bool isNeedPrint = true;

do {
    if (last != counter || isNeedPreView){
        counter = counter <= 0 ? 0 : counter;

        last = counter;

        setTimeForTimer = last * stepForTimer;

        printPreViewForTimer(setTimeForTimer);

        isNeedPreView = false;
    }

    if (isAnalogButtonPressed(analogPinForStartButtonOfTimer)){
        needTimeForTimer = micros() + ( setTimeForTimer * 1000000 );

        while (needTimeForTimer > micros()) {
            delay(100);

            currentTime = needTimeForTimer - micros();

            printRemaimingTimeForTimer(currentTime);
        }

        printWhenTimeEndsForTimer();

        soundSignal(numberSound, delayBetweenSound);

        delay(1000);

        counter = 0;

        isNeedPreView = true;
    }

    encoder();
} while (!isEncoderButtonPressed);

```

```

}

void printPreViewForTimer(double setTimeToPrint){

    lcd.setCursor(0, 0);

    lcd.clear();

    lcd.print(L"УСТАНОВИТЕ ВРЕМЯ");

    lcd.setCursor(0, 1);

    lcd.print(String(setTimeToPrint));

    lcd.print(L" сек.");

}


void printRemaimingTimeForTimer(double timerTimeToPrint){

    lcd.setCursor(0, 0);

    lcd.clear();

    lcd.print(L"ВРЕМЯ ");

    long second = timeToPrint / 1000000;

    long millisecond = timeToPrint - second * 1000000;

    millisecond = millisecond / 100;

    lcd.print(String(second));

    lcd.print(".");

    lcd.print(String(millisecond));

    lcd.setCursor(0, 1);

    lcd.print(L" сек.");

}

void printWhenTimeEndsForTimer(){

```

```

    lcd.setCursor(0, 0);

    lcd.clear();

    lcd.print(L"    CTOП! ");
}

bool isAnalogButtonPressed(byte analogPinNumber){
    return analogRead(analogPinNumber) < 300;
}

bool isPinHigh(byte pinNumber){
    return digitalRead(pinNumber) == HIGH;
}

void soundSignal(int numbersOfSound ,int timeBetweenSonds){
    const int highLevelOfSignal = 100;
    const int lowLevelOfSignal = 0;
    for (int j = 0; j < numbersOfSound; j++){
        analogWrite(pin_Sound_Signal, highLevelOfSignal);
        delay(timeBetweenSonds);
        analogWrite(pin_Sound_Signal, lowLevelOfSignal);
        delay(timeBetweenSonds);
    }
}

```

ПРИЛОЖЕНИЕ И

Внешний вид модернизированного устройства.

