

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Забайкальский государственный университет»
(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет

Кафедра информатики, вычислительной техники и прикладной математики

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

Тема: «Универсальный блок ввода/вывода для цифровых программируемых устройств,
разрабатываемых на базе ООО «Энергоинновационный центр»»

Выполнил:

студент гр. ВМК-16 Вол Волошинский Сергей Александрович

Руководитель ВКР:

к.ф.-м.н., доцент кафедры ИВТ и ПМ Б Батухтина Ирина Юрьевна

Чита
2020

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Забайкальский государственный университет»
(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет
Кафедра информатики, вычислительной техники и прикладной математики

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к выпускной квалификационной работе

по направлению подготовки 09.03.01 Информатика и вычислительная техника
профиль Вычислительные машины, комплексы, системы и сети

на тему «Универсальный блок ввода/вывода для цифровых программируемых устройств,
разрабатываемых на базе ООО «Энергоинновационный центр»»

Выполнил студент группы ВМК-16 Волошинский Сергей Александрович

Консультанты:

а) Экономическая часть: Каш старший преподаватель кафедры ЭиБУ,
Т.И. Кашурникова

б) Безопасность и экологичность проекта: БД доцент кафедры ВХЭиПБ, к.т.н.,
И.А. Бондарь

в) Специальная часть: БД доцент кафедры ИВТ и ПМ, к.ф.-м.н.,
И.Ю. Батухтина

Нормоконтроль: Шев старший преподаватель кафедры ИВТ и ПМ, Е.Б. Шевелёва

Руководитель работы: БД доцент кафедры ИВТ и ПМ, к.ф.-м.н.,
Ирина Юрьевна Батухтина

Допускаю к защите:

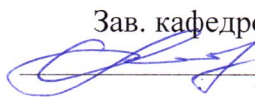
Зав. кафедрой ИВТ и ПМ Валова О.В. Валова

15 июня 2020 г.

Чита
2020

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Забайкальский государственный университет»
(ФГБОУ ВО «ЗабГУ»)

Энергетический факультет
Кафедра информатики, вычислительной техники и прикладной математики

УТВЕРЖДАЮ
Зав. кафедрой ИВТ и ПМ

О.В. Валова
11 мая 2020 г.

ЗАДАНИЕ
на выпускную квалификационную работу

Студенту Волошинскому Сергею Александровичу
по направлению 09.03.01 Информатика и вычислительная техника
профиль Вычислительные машины, комплексы, системы и сети

1 Тема выпускной квалификационной работы «Универсальный блок ввода/вывода для цифровых программируемых устройств, разрабатываемых на базе ООО «Энергоинновационный центр»»

Утверждена приказом (распоряжением) ректора по университету от 20.12.2019 № 1003-с

2 Срок подачи студентом законченной работы 15 июня 2020 г.

3 Исходные данные к работе:

- документация по микроконтроллерам Atmega328;
- документация по использованным дополнительным модулям;
- спецификация языка C.

4 Перечень подлежащих разработке в выпускной квалификационной работе вопросов:

- а) специальная часть;
- б) экономическая часть;
- в) безопасность и экологичность проекта.

5 Перечень графического материала (если имеется): –

6 Консультанты по выпускной квалификационной работе (с указанием, относящихся к ним разделов):

- а) экономическая часть: Кашурникова Тина Иннокентьевна Тес
б) безопасность и экологичность проекта: Бондарь Ирина Алексеевна ИР
в) нормоконтроль: Шевелёва Екатерина Борисовна ШЕВ
г) специальная часть: Батухтина Ирина Юрьевна Б

Дата выдачи задания 11 мая 2020 г.

Руководитель ВКР И. Ю. Батухтина И. Ю. Батухтина
(подпись, расшифровка подписи)

Задание принял к исполнению


11 мая 2020 г.

Подпись студента Вол / Сергей Александрович Волошинский /
(имя, отчество, фамилия)

Календарный план

№	Наименование раздела выпускной квалификационной работы	Неделя							
		1	2	3	4	5	6	7	8
1	Теоретическая часть								
2	Специальная часть								
3	Экономическая часть								
4	Безопасность и экологичность проекта								
5	Защита выпускной квалификационной работы								

(подпись, расшифровка подписи)



/ И. Ю. Батухтина /

РЕФЕРАТ

Пояснительная записка – 62 с., 7 рис., 12 табл., 6 источников, 5 прил.

БЛОК ВВОДА/ВЫВОДА, AVR, ATMEGA328, ПРОТОКОЛ ПЕРЕДАЧИ ДАННЫХ, I2C, ИНТЕРФЕЙС СВЯЗИ, СТРУКТУРА МЕНЮ, ОПЕРАТОР, ПРЕДСТАВЛЕНИЕ ДАННЫХ

Целью проекта является разработка универсального блока ввода/вывода для цифровых программируемых устройств, разрабатываемых на базе ООО «Энергоинновационный центр».

Поставленная цель достигается путем создания аппаратной и программной части блока ввода/вывода. Помимо этого, разработана структура представления данных меню, а также протокол передачи данных.

Данный блок предназначен для оператора и подключается к цифровым программируемым устройствам при помощи линии связи I2C, позволяя автоматически генерировать меню для взаимодействия с основным устройством. Данные для меню принимаются по специальному протоколу, и также по данному протоколу отправляются управляющие команды основному устройству.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	9
1 Постановка и анализ задачи	10
1.1 Описание предметной области	10
1.2 Применение устройства в ООО «Энергоинновационный центр»	12
1.3 Анализ существующих аналогов.....	14
1.4 Обоснование выбора средств реализации	15
2 Анализ данных.....	16
3 Программная реализация.....	19
4 Аппаратная реализация	26
5 Тестирование	27
6 Документирование	28
6.1 Техническое задание.....	28
6.2 Руководство пользователя.....	29
7 Экономическая часть	31
7.1 Обоснование целесообразности разработки проекта.....	31
7.2 Основные положения экономической части.....	32
7.3 Определение трудоемкости разработки программно- аппаратного комплекса.....	33
7.4 Определение стоимости комплекса	38
7.5 Оценка экономической эффективности.....	42
8 Безопасность и экологичность проекта	44
8.1 Требования к разрабатываемому комплексу.....	44
8.2 Требования к ПЭВМ	44
8.3 Требования к помещениям для эксплуатации ПЭВМ.....	46
8.4 Требования к шуму и вибрации в помещениях с эксплуатируемым ЭВМ	47
8.5 Требования к освещению помещений и рабочих мест с ПЭВМ.....	47
8.6 Требования к организации и оборудованию рабочих мест с ПЭВМ	48

8.7 Требования к организации медицинского обслуживания пользователей ПЭВМ	50
8.8 Требования электробезопасности.....	50
8.9 Меры оказания первой медицинской помощи при поражении электрическим током.....	51
ЗАКЛЮЧЕНИЕ	53
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	54
ПРИЛОЖЕНИЯ.....	55

ВВЕДЕНИЕ

В настоящее время цифровые устройства с большой скоростью проникают в жизнь каждого человека. Такие устройства имеют в своей основе компонент, который называется микроконтроллер. Он совмещает в себе процессор для произведения вычислений, а также всю необходимую периферию: различные интерфейсы связи, аналого-цифровой преобразователь, а также разные типы памяти для хранения данных программы и значений, которые вычисляются в процессе работы. И часто необходимо следить за параметрами работы программы, которую выполняет микроконтроллер. Для этого можно подключить дополнительные компоненты для вывода данных, такие как жидкокристаллический дисплей и компоненты для ввода, например кнопочную клавиатуру. Данные компоненты можно объединить в одно законченное устройство, включая дополнительные компоненты, которое будет иметь возможность подключения к устройству с микроконтроллером по специальному интерфейсу связи, для управления его параметрами.

Целью данного проекта является разработка универсального блока ввода/вывода, который будет подключаться по интерфейсу связи I2C к цифровым программируемым устройствам, разрабатываемым на базе организации ООО «Энергоинновационный центр». Данный блок будет генерировать меню, посредством которого оператор будет управлять различными элементами, вводить параметры функционирования и видеть на экране данные о работе устройства. Данные для меню представлены в виде специальной структуры, описывающей все необходимые параметры для генерации и взаимодействия с устройством. В момент подключения меню будет передаваться по специальному протоколу, по которому также будут передаваться управляющие команды.

1 Постановка и анализ задачи

1.1 Описание предметной области

В настоящее время применяется огромное количество цифровых программируемых устройств, для решения различных задач. Основой таких устройств обычно является микроконтроллер, в котором представление данных организовано в виде байтов информации, хранящихся в памяти. Также у микроконтроллеров имеются цифровые порты, которые могут работать как на вход, так и на выход. Из чего следует, что микроконтроллер способен получать внешние сигналы и выдавать информацию после обработки обратно.

Помимо этого, в настоящее время активно разрабатываются интерактивные системы, основной особенностью которых является возможность программным способом перестраивать их для работы с различными входными параметрами и менять режимы работы. При этом изменений в аппаратной части не требуется. Для таких устройств необходимо иметь возможность вносить изменения во внутренние параметры для изменения режимов работы, а также получать обратно информацию о состоянии системы.

Реализация такого взаимодействия для различных устройств может быть организована разными способами. Некоторые устройства включают дополнительные компоненты для внешнего взаимодействия с оператором в собственную комплектацию. Если габариты устройства не позволяют разместить дополнительные компоненты, в таком случае обычно оставляют возможность для подключения внешних компонентов для взаимодействия с оператором по различным интерфейсам связи.

Устройства, разрабатываемые на базе ООО «Энергоинновационный центр» были спроектированы таким образом, что предусматривают подключение внешних компонентов для взаимодействия с оператором при помощи выделенного интерфейса связи.

Решением является создание устройства, которое позволит удобно вводить информацию и получать обратно результаты работы устройства в понятном для человека виде.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Определить функциональные возможности конечного устройства.
2. Построить принципиальную и монтажную схему.
3. Разработать формат представления данных.
4. Разработать протокол связи.

Основные возможности взаимодействия оператора с блоком ввода/вывода, а также взаимодействие блока с устройствами, к которым он будет подключен, продемонстрированы на рисунке 1 в виде диаграммы вариантов использования.

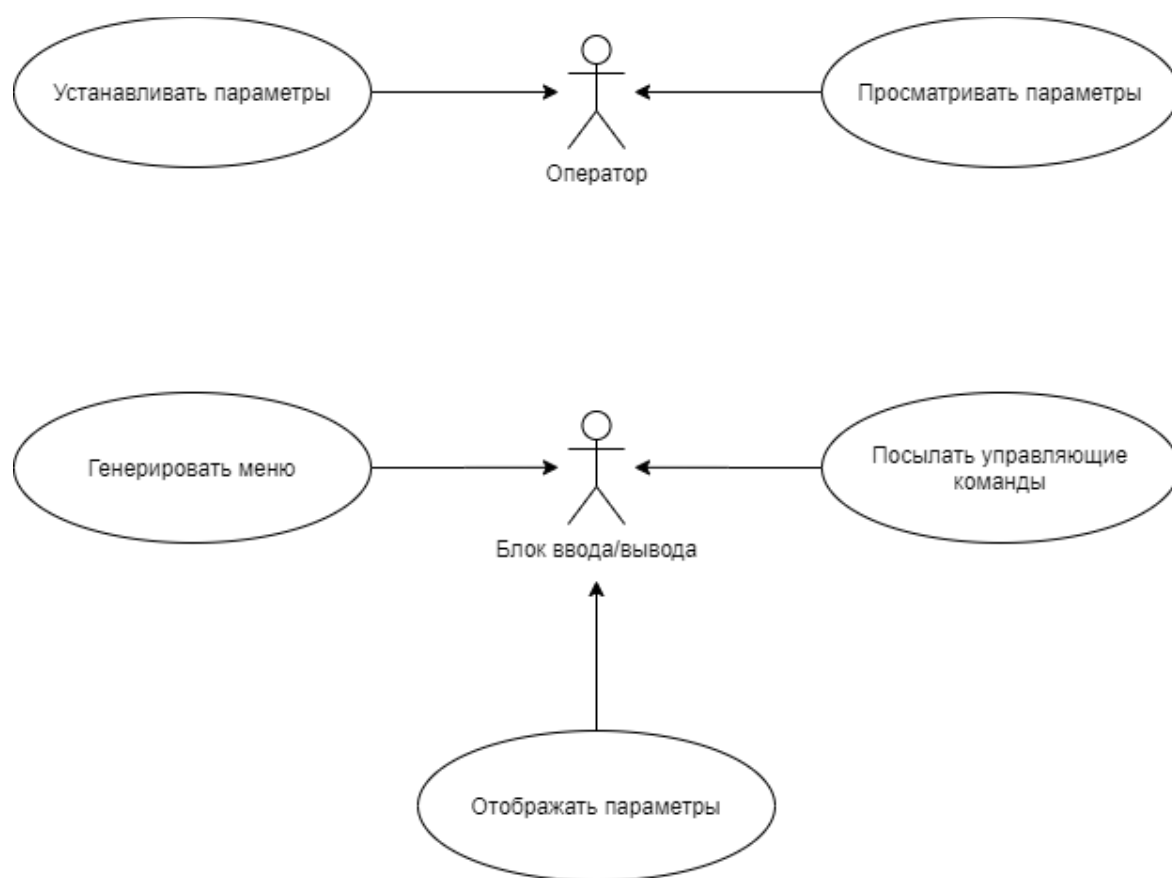


Рисунок 1 – Диаграмма вариантов использования

1.2 Применение устройства в ООО «Энергоинновационный центр»

На базе ООО «Энергоинновационный центр» разрабатываются различные программно-аппаратные комплексы, которые должны взаимодействовать с блоком ввода/вывода.

Первым проектом, для которого изначально и разрабатывался данный блок ввода/вывода, это система управления поворотной установки для солнечных панелей.

В составе этого проекта разрабатывается поворотная установка для солнечных панелей, позволяющая ориентировать их в точку с максимальной вырабатываемой мощностью. На данный момент в проекте разработана и собрана миниатюрная поворотная установка, предназначенная для проведения полевых испытаний алгоритма позиционирования. Блок ввода/вывода должен применяться для просмотра параметров установки во время испытаний, а также для изменения внутренних параметров и настроек.

Помимо этого, в дальнейшем планируется доработка разработанной ранее в данном проекте полноразмерной поворотной установки для солнечных панелей большей мощности, которые планируется объединить в солнечную электростанцию схематичный вид которой представлен на рисунке 2.

В данной цепи устройств блок ввода/вывода будет использоваться для настройки параметров всей электростанции, для отслеживания параметров её работы, а также для ручного управления каждой из установок при возникновении внештатных ситуаций, в которых автоматический режим позиционирования ведет себя некорректно.

Также блок ввода/вывода планируется применять в комплексе автоматического управления скважинными насосами, который разрабатывается для повышения экономической эффективности и надежности эксплуатации участков первого подъема.

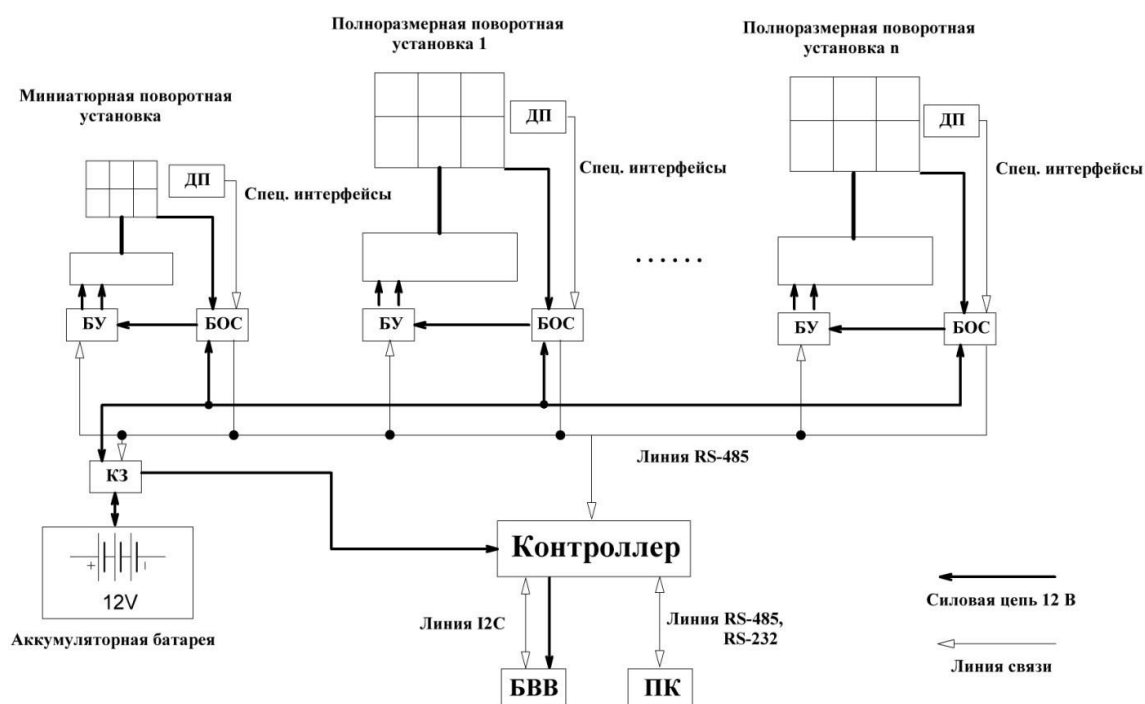


Рисунок 2 – Схема солнечной электростанции

В настоящее время на территории Забайкальского края и многих других регионах Российской Федерации широко распространены системы водоснабжения с накапливающим резервуаром, позволяющие стабилизировать процесс водоснабжение при динамически изменяющемся разборе. Схема такой системы приведена на рисунке 3.



Рисунок 3 – Система водоснабжения с накапливающим резервуаром

Универсальный блок ввода/вывода, будет связующим звеном между устройством управления и оператором, а устройство управления будет получать информацию с датчиков, блока ввода/вывода и вычислять управляющее воздействие, посредством которого будет осуществляться управление производительностью насосов.

1.3 Анализ существующих аналогов

Аналогом разрабатываемого устройства является продукт компании «ОВЕН»: ИПП120 информационная программируемая панель оператора. Его внешний вид представлен на рисунке 4.

Данная панель оснащена двухстрочным жидкокристаллическим дисплеем, позволяющим выводить 16 символов в строке. На лицевой панели находятся 6 кнопок управления внутренним меню. Устройство имеет возможность для крепления в отверстиях в щите. Используется интерфейс связи RS-485 и протокол Modbus RTU/ASCII (режимы Master/Slave).



Рисунок 4 – ОВЕН ИПП120

Одним из недостатков является стационарное крепление в щите, что снижает портативность устройства. Данный способ крепления предполагает подключение к системе, в которой компоненты расположены стационарно и на относительно небольшом расстоянии друг от друга.

При взаимодействии с панелью без закрепления в щите, дизайн устройства не позволяет использовать управление компонентами одной рукой, что также сказывается на удобстве и еще больше доказывает, что устройство не предусмотрено для портативного использования.

Также важными недостатками являются: сильная зависимость от других продуктов компании «ОВЕН», программирование панели только в специальной среде и высокая стоимость устройства.

1.4 Обоснование выбора средств реализации

В качестве микроконтроллера для блока ввода/вывода был выбран микроконтроллер Atmega328. Данный микроконтроллер является достаточно мощным решением и обладает всеми нужными интерфейсами, а также является недорогим решением для разработки устройств, по качеству, приближенных к промышленному типу.

При разработки программной части блока были выбраны языки: С, С++. Данные языки являются мощным инструментом при разработке программ для микроконтроллеров, а также имеют большое количество сред разработки и компиляторов, предназначенных для микроконтроллеров AVR.

Для подключения блока ввода/вывода к различным устройствам используется интерфейс связи I²C (Inter-Integrated Circuit) или TWI (Two Wire Interface) (I2C). Данный интерфейс создавался для связи между интегральными схемами внутри электронных приборов и достаточно гибок в применении, поэтому для передачи небольших сообщений этот интерфейс идеально подходит. Очевидным преимуществом данного интерфейса – аппаратная реализация на микроконтроллерах AVR, что улучшит скорость обработки полученного сообщения, а также существенно сократит затраты, связанные со временем и сложностью разработки программного обеспечения.

2 Анализ данных

Основной задачей блока ввода/вывода в момент подключения его к какому-либо устройству – это получение данных для генерации меню, при помощи которого будет производиться взаимодействие с устройством.

Прежде чем передавать данные, связанные с меню, нужно понять в каком виде они будут представлены на устройстве.

На рисунке 5 показана структура меню. Она разработана таким образом, чтобы представить данные в формате, с которыми будет удобно взаимодействовать оператору, а также для простого представления пунктов меню при передаче их между устройствами.

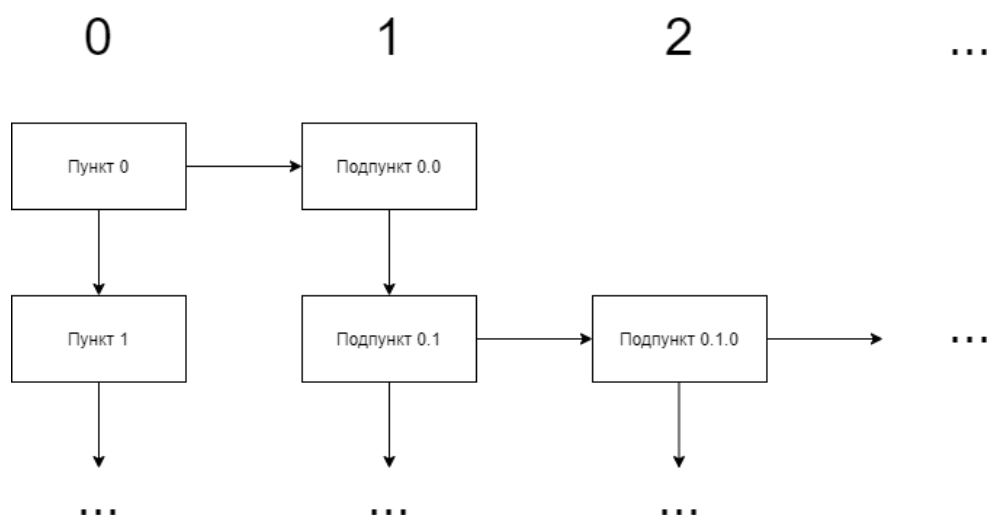


Рисунок 5 – Структура меню

Положение каждого пункта меню в данной структуре можно однозначно определить благодаря данным параметрам:

- уровень пункта (на рисунке 5 это число над пунктами);
- комплексный параметр, содержащий номера пунктов предыдущих уровней, на которых происходит переход на следующий уровень.

Например, для пункта, названного на рисунке 5 «Подпункт 0.1.0» эти параметры будут выглядеть таким образом: {2, [0, 1]}. Эта конструкция говорит нам, что «Подпункт 0.1.0» находится на втором уровне при

индексации с нуля. На предыдущих уровнях переход на новый уровень осуществляется из пунктов, находящихся на нулевой позиции для нулевого уровня и на первой позиции для первого уровня. А «Пункт 1» будет иметь такие параметры: {0, []}. Второй параметр пуст, так как количество значений в нем равно уровню первого параметра и в данном случае равно нулю.

Также при хранении пунктов меню в памяти микроконтроллера, важным условием является их расположение относительно друг друга в массиве. Они должны располагаться таким образом, что все пункты, по которым необходимо пройти, чтобы попасть в конкретный пункт меню, должны быть описаны выше него в списке.

Помимо параметров расположения, рассмотрим, что еще требуется для представления пункта меню в системе. Следует указать наименование данного пункта, чтобы оператор мог однозначно понять, за что он отвечает.

Следующий параметр отвечает за категорию пункта. Пункт может иметь одну из следующих категорий:

- отображаемое значение;
- редактируемое значение;
- переход в подменю;
- управляющее действие.

Когда указывается категория отображаемого или редактируемого значения, нужно указать адрес переменной и её тип. А в случае с категорией управляющего действия указывается адрес функции, которая будет выполнена при вызове данного пункта.

Помимо хранения меню его требуется передавать от устройства, к которому будет производиться подключение, блоку ввода/вывода для последующего генерирования и отображения. Для этого необходимо определить протокол передачи данных.

Для корректного обмена информацией между устройством, к которому произведено подключение и блоком ввода/вывода была разработана структура передачи сообщений. Общение между устройствами будет

производиться по принципу запрос-ответ. Сообщения будут иметь структуру, представленную на рисунке 6.



Рисунок 6 – Структура сообщения

В структуру сообщения входит:

- адрес устройства, к которому в данный момент происходит обращение;
- код функции, которую необходимо выполнить. Данный код позволяет однозначно определить, что именно необходимо сделать в конкретный момент времени;
- данные необходимые для обработки запроса или записи в память.

Когда устройство получает сообщение, оно сравнивает адрес из него со своим адресом, и если эти адреса совпадают, то оно продолжает принимать остальные данные. Если адреса не совпадают, то устройство ничего не делает и продолжает выполнение текущих задач.

Все полученные данные сначала сохраняются во внутреннем регистре модуля TWI, и для того, чтобы обработать эти данные их необходимо выгрузить во входной буфер. После они обрабатываются и исходя из полученного кода и данных, формируется ответное сообщение, в котором код остается тем же, что был получен, а также добавляются соответствующие данные, которые ожидает запрашивающее устройство.

В момент подключения блока ввода/вывода, он запрашивает все необходимые данные для генерирования меню последовательностью из нескольких сообщений, отправляя новое после получения ответа от устройства на предыдущее.

После получения всех данных блоку ввода/вывода необходимо их обработать, поэтому после завершения работы модуля TWI, он производит

интерпретацию полученных данных и изменение значений переменных либо выполняет другие функции.

Эта структура – основной вид сообщений, при работе системы в штатном режиме. Она необходима в большинстве случаев для обновления сведений о системе, которые в последствии будут выведены на ЖК дисплей.

3 Программная реализация

При разработке программно-аппаратного комплекса была реализована программа для прошивки микроконтроллера.

Отличия в программировании микроконтроллеров от написания программ для персональных компьютеров заключаются в ограниченности ресурсов микроконтроллера. Например, объем памяти для хранения кода программы на микроконтроллере Atmega328, который использовался при создании блока ввода/вывода, равен 32 килобайта, а размер оперативной памяти составляет всего 2 килобайта. Из-за того, что данные значения в сотни тысяч раз отличаются от характеристик персональных компьютеров, необходимо разрабатывать программы только для конкретных задач, а заставлять микроконтроллер делать все и сразу не целесообразно.

Помимо этого, свои ограничения вносит этап работы с низкоуровневой периферией. Это может быть работа с портами ввода вывода, аналого-цифровым преобразователем, различными модулями интерфейсов связи, такими как UART или TWI.

Первым делом рассмотрим структуру, в которой будут храниться данные для создания меню. Она располагается на устройстве, к которому будет производиться подключение. Код структуры приведен в приложении А.

Данная структура имеет название menuItemData и имеет следующий набор значений:

1. Значение Level – однобайтное значение, указывающее на то, на каком уровне располагается конкретный пункт меню.
2. Значение Transition – массив однобайтных значений, имеющий размерность равную параметру Level и содержащий номера пунктов предыдущих уровней, на которых происходит переход на следующий уровень.
3. Значение Text – массив символов, который содержит наименование

пункта.

4. Значение `Select` – однобайтовое значение, хранящее одну из возможных категорий пунктов меню:

- отображаемое значение;
- изменяемое значение;
- переход в подменю;
- управляющее действие.

5. Значение `Value` – двухбайтное значение хранящее значение адреса переменной, либо функции для всех пунктов с категорией отличной от перехода в подменю.

6. Значение `Type` – однобайтное значение, которое хранит тип переменной, расположенной по адресу, сохраненному в параметре `Value`, для пунктов с категориями: отображаемое значение и изменяемое значение.

Пример создания меню на основе описанной выше структуры также показан в приложении А.

Далее рассмотрим структуру, в которую будут приниматься данные на блоке ввода/вывода. Она имеет некоторые отличия, необходимые для удобного хранения данных на блоке. Её код представлен в приложении Б.

Рассмотрим отдельно параметры, которые имеют отличия в представлении на блоке ввода/вывода от устройства, к которому происходит подключение.

В блоке ввода/вывода имеется регистр – массив символов `strReg` для хранения данных, формирующих наименования пунктов меню. Поэтому в структуре для хранения этих данных, вместо параметра `Text` появляются два параметра: `TextIndex` и `TextLength`. Первый содержит индекс, начиная с которого в регистре с символами хранится наименование данного пункта, а второй содержит длину строки и позволяет узнать сколько нужно считать символов из регистра, чтобы получить целиком наименование пункта.

Помимо этого, есть массив `valueReg`, который содержит значения переменных любого типа в их байтовом представлении. Он необходим для

удобного хранения и обращения с данными, связанными с переменными. В структуру данных добавляется параметр `ValueIndex`, он содержит индекс, начиная с которого в регистре с байтами переменных хранится необходимое значение. Количество байт, которое необходимо считать из регистра с байтами определяет параметр `Type`, благодаря которому можно узнать тип переменной и соответственно её размер.

Главная структура, которая называется `menuItem`, хранит в себе все основные данные необходимые для обращения с пунктами меню во время работы программы, после того как оно уже сгенерировано. Её код также представлен в приложении Б.

На основе данной структуры создается список из пунктов меню. Каждый элемент помимо основных данных хранит 4 указателя:

1. Указатель `Previous` – указывает на предыдущий элемент на том же уровне.
2. `Next` – указывает на следующий элемент на том же уровне.
3. `Parent` – указывает на элемент родитель на предыдущем уровне.
4. `Child` – указывает на элемент потомок на следующем уровне.

В том случае, когда у конкретного пункта нет предыдущего или следующего элемента, а также если отсутствует родитель или потомок, то он указывает на элемент заглушку `NULL_ENTRY` значения всех параметров которого равны 0.

Далее рассмотрим функцию, которая добавляет новый элемент меню. Её код представлен в приложении В.

Функция `addMenuItem` принимает несколько параметров:

1. Параметр `currentItem` – указатель на текущий элемент меню, изначально в который передается указатель на корень и в процессе выполнения функции он изменяется, перемещаясь до нужного места в меню.
2. Параметр `addedItem` – указатель на участок памяти, в котором сохраняются данные добавляемого элемента.

3. Параметр `itemData` – данные необходимые для добавления нового пункта в список.

При первом вызове функции указатель на корень меню будет пуст, и в этом случае функция его создаст. В последующие вызовы относительно корня будут добавляться остальные пункты. Если пункт добавляется на тот же уровень, где находится корень меню, то при помощи функции `endLevel` будет достигнут последний элемент на данном уровне, после которого добавится новый пункт. В том случае, когда добавляемый пункт находится не на начальном уровне функция будет спускаться на столько элементов, сколько указано в значении параметра `Transition` для текущего уровня и вызовет функцию `changeLevel`, которая перейдет на следующий уровень. При достижении нужного места в меню, проверяется есть ли уже на новом уровне, куда требуется добавить пункт, потомок. Если его нет, то он создается, в случае, когда на новый уровень уже были добавлены некоторые элементы, вызывается функция `endLevel`, для того чтобы спуститься в самый низ и добавить пункт в конец текущего уровня.

Теперь рассмотрим функцию `processMessage`, которая обрабатывает приходящие в ответ на запрос сообщения от устройства, к которому произведено подключение. Код функции показан в приложении Г.

Когда сообщение приходит на микроконтроллер по интерфейсу I2C, оно сохраняется из буфера модуля TWI в специальный кольцевой буфер, который хранит принятые и еще не обработанные сообщения. Когда данный буфер не пуст, вызывается функция обработки сообщения. Первым делом она считывает значение кода, на основе которого будет определен дальнейший разбор пришедшего сообщения.

Коды, которые могут присутствовать в сообщении:

- код `0x00` – код для сообщений, настраивающих соединение между устройствами;
- код `0x01` – код сообщений, содержащих данные с информацией о пунктах меню;

- код 0x02 – код сообщений, предназначенных для обновления данных, связанных с переменными;

- код 0x03 – код сообщений, посылающих запрос на выполнение управляющего действия.

При подключении блок ввода/вывода посылает сообщение с кодом 0x00 на широковещательный адрес I2C (0x00) и в поле данных кладет собственный адрес. При получении сообщения с таким кодом, устройством, к которому подключен блок, оно сохраняет адрес блока и отправляет на только что полученный адрес ответное сообщение с таким же кодом, только добавляя в поле данных собственный адрес. После установленного соединения блок запрашивает данные для создания меню.

Код сообщения ставится 0x01, как и для всех последующих сообщений, а в поле данных кладется номер, который имеет следующие значения:

- код 0x00 – указывает на данные, которые содержат количество элементов меню, длину всех наименований пунктов и суммарный размер всех переменных. Данные нужны для динамического выделения памяти под массивы для хранения данных;

- код 0x01 – показывает, что в данном сообщении в поле данных содержится значение параметра Level и значения параметра Transition;

- 0x02 – в сообщении с данным номером содержится наименование пункта;

- 0x03 – сообщение с таким номером содержит всю основную информацию: категорию пункта, адрес переменной, её тип и значение.

Также в сообщениях, приходящих с подключенного устройства, помимо данных указывается индекс пересылаемого пункта.

Для передачи измененных значений с блока на подключенное устройство отправляются сообщения с кодом 0x02, также с таким кодом отправляются данные с подключенного устройства для обновления регистра со значениями переменных.

Когда с блока вызывается управляющее действие, отправляется сообщение с кодом 0x03 и адресом функции, которую необходимо выполнить.

4 Аппаратная реализация

Аппаратная часть блока ввода/вывода состоит из следующих компонентов:

1. Микроконтроллер Atmega328.
2. Жидкокристаллический дисплей, рассчитанный на вывод одновременно 32 символов по 16 на каждой из двух строк.
3. Четыре кнопки для навигации и две кнопки для цепей сброса.
4. Два световых индикатора состояния системы.
5. Разъем для внутрисхемного программирования, к которому подключается программатор.
6. Разъем подключения блока к другим устройствам, в формате гнезда USB 3.0.

Принципиальная схема блока ввода/вывода продемонстрирована на рисунке Д.1 в приложении Д. Также на схеме показаны основные компоненты конечного устройства и то, как они будут подключены в конечном устройстве.

Монтажная схема блока ввода/вывода продемонстрирована на рисунке Д.2 в приложении Д. Схема отображает конечное положение электронных компонентов на печатной плате устройства.

5 Тестирование

При тестировании устройства применялись в основном методы детерминированного тестирования. Этот вид тестирования является достаточно трудоёмким, но зато даёт хорошие результаты по обнаружению ошибок и дальнейшего их устранения.

Специально для тестирования был установлен светодиод. На ранних стадиях разработки, в состав прошивки входили алгоритмы для моргания светодиодом с разной частотой, в случае корректного приёма или передачи данных, а также при выполнении иных действий. Данная возможность значительно упростила поиск ошибок, при тестировании устройства в реальных условиях.

Программная часть комплекса тестировалась при помощи инструментов отладки программной среды AVR Studio 4. Эти инструменты позволяют просмотреть значения регистров, флагов, переменных, и записывать по этим адресам необходимые разработчику значения.

Также для проверки работоспособности всей программы использовался эмулятор, позволяющий провести тестирование комплекса на предмет динамически возникающих ошибок, которые невозможно отловить до возникновения каких-либо определенных событий.

При помощи AVR Studio 4 у разработчика имеется возможность подключиться к функционирующему устройству, и пронаблюдать реальные процессы и события, происходящие во время выполнения программы, записанной в память микроконтроллера и выполняющаяся посредством вычислительных мощностей этого микроконтроллера.

6 Документирование

6.1 Техническое задание

6.1.1 Введение

Программно-аппаратный комплекс «Универсальный блок ввода/вывода для цифровых программируемых устройств» предназначен для взаимодействия с устройствами, разрабатываемыми на базе ООО «Энергоинновационный центр», для ввода данных и вывода параметров.

6.1.2 Назначение разработки

Изначально данное устройство проектировалось для функционирования в проекте поворотной установки для солнечных панелей. Оно позволит производить управление поворотной установкой при проведении практических исследований.

Помимо этого, данный блок ввода/вывода планируется использовать с другими устройствами, разрабатываемыми на базе ООО «Энергоинновационный центр», например для взаимодействия с устройством управления скважинными насосами участка первого подъема системы водоснабжения с накапливающим резервуаром.

6.1.3 Требования к функциональным характеристикам

Блок ввода вывода имеет возможность:

- динамически генерировать меню на основе данных, получаемых от устройства, к которому произведено подключение;
- производить вывод информации о состоянии устройства на ЖК дисплей, а также информировать оператора о различных внештатных ситуациях посредством световых сигналов;

- обновление значений параметров, для последующей передачи этих значений на устройство;

Оператор имеет возможность:

- вводить параметры, необходимые для корректной работы устройства;
- просматривать параметры.

6.1.4 Требования к надёжности

Устройство блока ввода/вывода выполнено при помощи монтажа электронных компонентов на печатной плате, что обеспечивает устойчивую работу, а корпус, в который установлена плата, защищает её от внешних механических воздействий.

Программная часть комплекса должна обеспечивать стабильное выполнение поставленных задач. При возникновении исключений и при прекращении стабильной работы устройства предусмотрена возможность оповещения оператора при помощи светодиодной индикации.

6.1.5 Минимальные требования к устройствам

Устройство оснащено корпусом, защищающим устройство от внешних воздействий окружающей его среды. Размеры устройства должны быть достаточными, для удобного взаимодействия оператора с ним.

6.1.6 Требования к документации

Документация к программной и аппаратной части должна содержать руководство пользователя.

6.2 Руководство пользователя

6.2.1 Общее назначение

Устройство предназначено для вывода и настройки параметров программно-аппаратных комплексов, к которым производится подключение при помощи линии связи, за счет автоматической генерации меню.

6.2.2 Установка и использование

Блок ввода/вывода подключается к устройству при помощи проводов питания, провода линии сброса, а также проводов интерфейса связи I2C.

После подключения проводов, убеждаемся в правильности подключения всех узлов при помощи светодиодной индикации наличия питания и началу загрузки меню на экране.

7 Экономическая часть

7.1 Обоснование целесообразности разработки проекта

Программно-аппаратный комплекс, который разрабатывается в данном проекте, предназначен для подключения к цифровым программируемым устройствам для управления, ввода параметров и вывода их на экран. Разрабатываемый продукт предполагается использовать в случаях, когда имеется устройство или комплекс устройств, для которых необходимо взаимодействие с оператором.

Вследствие этого, приобретая данный продукт, покупатель получает аппаратный комплекс, позволяющий взаимодействовать оператору с устройством или комплексом устройств по линии связи, к которому будет подключен.

С точки зрения маркетинга главным в продукте является его потребительские свойства, т.е. способность удовлетворить потребности того, кто им владеет.

При помощи системы трёхуровневого анализа проанализируем потребительские свойства данного программно-аппаратного продукта.

Рассмотрим основные составляющие трёхуровневого анализа:

1. Сущность продукта заключается в том, что один и тот же товар способен решать разные проблемы потребителей и удовлетворять их различные потребности.

2. Фактический продукт состоит из различных характеристик той формы, которую принимает наш продукт.

3. Добавления к продукту — это то, что мы можем предложить нашим покупателям помимо основного товара. Предложение дополнительных услуг повышает ценность продукта для покупателя и позволяет выгодно отличить товар от товаров конкурентов.

Трёхуровневый анализ разработанного программно-аппаратного продукта представлен в таблице 1.

Таблица 1 – Трёхуровневый анализ продукта

Уровень	Описание
Сущность продукта	Подключение к цифровым программируемым устройствам для взаимодействия с оператором
Фактический продукт	Программно-аппаратный комплекс, позволяющий оператору подключать его к цифровым программируемым устройствам для управления, ввода значений, и вывода параметров работы
Добавление к продукту	Программный продукт, позволяющий генерировать меню через удобный графический интерфейс

Схема трёхуровневого анализа программно-аппаратного продукта представлена в соответствии с рисунком 7.



Рисунок 7 – Схема трёхуровневого анализа продукта

7.2 Основные положения экономической части

Целью написания экономической части является определение стоимости разработки программно-аппаратного продукта, определение трудоёмкости и эффективности использования.

Стоимость программного обеспечения может рассматриваться с двух точек зрения:

- с точки зрения формирования объектов нематериальных активов, как

исключительное право для использования непосредственно на предприятии;

– с точки зрения формирования объектов нематериальных активов, как исключительное право для непосредственного коммерческого использования, продажи и продвижения программного обеспечения на рынке.

В данном разделе будет определена, при помощи расчётов стоимости, экономическая эффективность программно-аппаратного продукта.

7.3 Определение трудоемкости разработки программно-аппаратного комплекса

Трудоёмкость определяется отношением затрат рабочего времени к производству единицы продукции. Трудоёмкость программного средства рассчитывается на основе определённых типовых правил в сфере деятельности, соответствующей выбранной предметной области.

Для разработки в области компьютерных и информационных технологий существуют типовые нормы времени. Они необходимы для определения нормы труда специалистов, занятых созданием программного обеспечения для электронно-вычислительных машин, установление численности исполнителей, а также обоснование трудности разработки программного обеспечения. Данные нормы времени включают комплексы задач всех систем и подсистем управления, статические задачи, а также различные задачи, связанные с расчетами.

При расчёте трудоёмкости выделяют следующие стадии выполнения работ: техническое задание, эскизный проект, технический проект, рабочий проект, внедрение.

На стадии «Техническое задание» заказчик формирует требования к программно-аппаратному комплексу или системе и производится консультация разработчиков по сформированному вопросу. Следующим этапом производится обоснование принципиальной возможности решения

данной задачи, разрабатывается ведущая концепция, определяются и согласовываются сроки разработки комплекса. Расход времени на выполнение задачи обозначается за T_1 .

В свою очередь T_1 имеет значение равное 42 чел./дн., которое определяется в соответствии с таблицей 2 [3].

Таблица 2 – Затраты времени при выполнении работ на стадии «Техническое задание»

Подсистемы	Степень новизны			
	А	Б	В	Г
Перспективное планирование развития и размещения отрасли, управление проектированием и капитальным строительством.	79	57	37	34
Управление материально – техническим снабжением, управление сбытом продукции.	105	76	42	30
Бухгалтерский учет, управление финансовой деятельностью	103	72	48	35
Управление организацией труда и заработной платой, управление кадрами.	63	46	30	19

Стадия «Эскизный проект» включает проработку технического задания, итогом которой производится выбор и разработка математической модели, эскиз аппаратной части и алгоритм разработки программной части комплекса.

Таблица 3 – Затраты времени при выполнении работ на стадии «Эскизный проект»

Подсистемы	Степень новизны			
	А	Б	В	Г
Перспективное планирование развития и размещения отрасли, управление проектированием и капитальным строительством.	175	117	77	53
Управление материально – техническим снабжением, управление сбытом продукции.	115	79	53	35
Бухгалтерский учет, управление финансовой деятельностью	166	112	73	57
Управление организацией труда и заработной платой, управление кадрами.	151	101	67	46

Расход времени при выполнении данной задачи обозначается за T_2 , и имеет значение равное 53 (чел./дн.), которое определяется в соответствии с таблицей 3 [4].

Стадия «Технический проект» включает разработку программной документации и выбор конфигурации технических средств, которые максимально эффективно способствуют реализации комплекса.

Таблица 4 – Коэффициенты для определения нормы времени при выполнении работ на стадии «Технический проект»

Подсистемы	Разработчик	Коэффициенты		
		А	Б	В
Перспективное планирование развития и размещения отрасли, управление проектированием и капитальным строительством.	ПЗ	30,04	0,45	0,34
	ПО	8,34	4,56	0,17
Управление материально – техническим снабжением, управление сбытом продукции.	ПЗ	20,99	0,46	0,35
	ПО	9,33	0,48	0,16
Бухгалтерский учет, управление финансовой деятельностью	ПЗ	17,01	0,56	0,4
	ПО	8,19	0,59	0,19
Управление организацией труда и заработной платой, управление кадрами.	ПЗ	16,9	0,45	0,34
	ПО	8,37	0,53	0,7

Для данных трёх стадий предварительные нормы затрат времени определяются по следующей формуле:

$$T_{3,4,5} = r_1 \cdot \Phi_1^{r_2} \cdot \Phi_2^{r_3} \text{ (чел./дн.)},$$

где r_1 , r_2 , r_3 – коэффициенты для каждой стадии проекта, принятые в соответствии с таблицами 4 – 6 [3];

Φ_1 – количество макетов входной информации;

Φ_2 – количество разновидностей форм выходной информации.

Так как разработчик участвовал в подготовке информационного обеспечения, на стадиях «Технический проект» и «Рабочий проект» коэффициенты r_1 , r_2 , r_3 были увеличены в 1,1 раз (повышающий коэффициент).

Для разрабатываемого проекта $\Phi_1 = 3$, а $\Phi_2 = 2$.

Стадия «Рабочий проект» является наиболее трудоемкой. На ней выполняется разработка аппаратной и программной частей, их испытание и отладка. Результатом стадии является готовый программно-аппаратный комплекс, включающий руководство пользователя и рабочую документацию.

Таблица 5 – Коэффициенты для определения нормы времени при выполнении работ на стадии «Рабочий проект»

Подсистемы	Разра- ботчик	Коэффициенты		
		А	Б	В
Перспективное планирование развития и размещения отрасли, управление проектированием и капитальным строительством.	ПЗ	8,11	0,47	0,49
	ПО	50,05	0,44	0,42
Управление материально – техническим снабжением, управление сбытом продукции.	ПЗ	10,32	0,46	0,48
	ПО	33,81	0,45	0,43
Бухгалтерский учет, управление финансовой деятельностью	ПЗ	8,1	0,54	0,52
	ПО	31,99	0,55	0,49
Управление организацией труда и заработной платой, управление кадрами.	ПЗ	5,1	0,47	0,51
	ПО	51,78	0,42	0,41

Ниже представлен расчет норм временных затрат T_3 , T_4 , T_5 .

$$T_3 = 10,26 \cdot 3^{0,52} \cdot 2^{0,17} = 20,44 \text{ (чел./дн.)};$$

$$T_4 = 37,19 \cdot 3^{0,49} \cdot 2^{0,47} = 88,3 \text{ (чел./дн.)};$$

$$T_5 = 8 \cdot 3^{0,46} \cdot 2^{0,51} = 18,83 \text{ (чел./дн.)}.$$

Тогда общие трудозатраты будут равны:

$$T_{\Sigma} = T_1 + T_2 + T_3 + T_4 + T_5 = 42 + 53 + 20,44 + 88,3 + 18,83 = 222,57 \text{ (чел./дн.)}.$$

Временные затраты на создание аппаратной части комплекса примерно составили 65 (чел./дн.).

Значит общие временные затраты на создание комплекса составляют:

$$T_{\text{вр. комп.}} = 222,57 + 65 = 287,57 \text{ (чел./дн.)}$$

Численность исполнителей, необходимая для выполнения работ по созданию комплекса вычисляется по формуле:

$$\text{Ч} = \frac{T_{\text{вр.комп}}}{\Phi_{\text{пл}}}, \quad (1)$$

где $\Phi_{\text{пл}}$ – фонд рабочего времени, планируемый на одного разработчика;

$T_{\text{вр. комп}}$ – временные трудозатраты на создание комплекса, (чел./дн.)

Стадия «Внедрение» включает проведение проверки правильности выполнения работы комплекса на практике в процессе подготовки требуемой документации, а после программно-аппаратный комплекс выпускается пользователям в эксплуатацию.

Таблица 6 – Коэффициенты для определения нормы времени при выполнении работ на стадии «Внедрение»

Подсистемы	Разра- ботчик	Коэффициенты		
		А	Б	В
Перспективное планирование развития и размещения отрасли, управление проектированием и капитальным строительством.	ПЗ	9,10	0,44	0,44
	ПО	10,89	0,38	0,48
Управление материально – техническим снабжением, управление сбытом продукции.	ПЗ	8,74	0,49	0,45
	ПО	8	0,46	0,51
Бухгалтерский учет, управление финансовой деятельностью	ПЗ	9,16	0,43	0,43
	ПО	7,12	0,43	0,43
Управление организацией труда и заработной платой, управление кадрами.	ПЗ	9,1	0,44	0,44
	ПО	10,91	0,38	0,48

Комплекс разрабатывается в течении девяти месяцев с сентября по май, что составляет 182 рабочих дня, соответственно рекомендуемая численность разработчиков проекта составляет:

$$\text{Ч} = 287,57 / 182 = 1,58 \sim 2 \text{ (чел.)}.$$

Фактически разработку данного программно-аппаратного комплекса ведет один программист в установленные сроки с учетом того, что некоторые этапы разработки допускают параллельное выполнение.

7.4 Определение стоимости комплекса

Себестоимость – это стоимостная оценка используемых в процессе производства материальных, трудовых и других затрат, а также затрат на реализацию продукции. Стоимость программно-аппаратного комплекса определена на основе укрупненного метода учета затрат по материальным затратам, расходам на оплату труда, отчислений на социальное страхование, амортизации основных фондов и прочих расходов.

Стоимость рассчитывается по формуле:

$$C_{по} = МЗ + ФЗП + A_0 + П,$$

где $МЗ$ – материальные затраты;

$ФЗП$ – фонд заработной платы разработчика программного продукта;

A_0 – амортизационные отчисления;

$П$ – прочие расходы.

Количество рабочих часов, затраченных на разработку данного программно-аппаратного комплекса, учитывая, что проект разрабатывался в течение 182 рабочих дня, а длительность рабочего дня составляет 8 часов, получим общее количество рабочих часов:

$$КЧ_{об} = T_{\Sigma ок} \cdot 8 = 182 \cdot 8 = 1456 \text{ (час.)}.$$

Материальные затраты включают в себя:

1. Расходные материалы:

– канифоль – 80 (р.);

– припой – 250 (р.);

– стеклотекстолит – 500 (р.).

2. Расходы на электроэнергию.

Определение затрат на электроэнергию ведется из расчета того, что потребление энергии компьютером составляет 0,5 кВт/час. Для тестирования устройства в среднем потреблялось 0.01 кВт/час. А стоимость 1 кВт/час. равна 4,2 (р.), тогда затраты на электроэнергию составят:

$$1456 \cdot 0,5 \cdot 4,2 + 1456 \cdot 0,01 \cdot 4,2 = 3118,76 \text{ (р.)}.$$

Таблица 7 – Затраты на аппаратную часть

Наименование	Цена
АТmega 328P (1 шт.)	170 (р.)
BCB1602-03, ЖК-дисплей 16x2 (1 шт.)	280 (р.)
Разъем USB 3,0 (1 шт.)	55 (р.)
Вилка на кабель USB 3,0 (1 шт.)	45 (р.)
Корпус G1024B (1 шт.)	250 (р.)
Прочие компоненты	1500 (р.)

Аппаратная часть комплекса также являются материальной ценностью. Список затрат на данную часть приведены в таблице 7.

Руководствуясь таблицей 7, стоимость использованного оборудования составляет 2300 (р.).

Материальные затраты на разработку составляют:

$$МЗ = 3118,76 + 2300 + 80 + 250 + 500 = 6248,76 \text{ (р.)}.$$

Заработная плата определяется на основе трудового договора между работником и работодателем. Средний должностной оклад инженера-программиста согласно штатному расписанию ООО «Энергоинновационный центр» составляет 13000 (р.).

По закону Забайкальского края от 26.09.2008 № 39-ЗЗК «О районном коэффициенте и процентной надбавке к заработной плате работников бюджетных организаций» установлен четкий районный коэффициент в размере 20 %, а также предусмотрена надбавка 30 % за стаж работы в районах Крайнего Севера и приравненных к ним местностях, а также в остальных районах Севера.

Таким образом, ежемесячная заработная плата составляет:

$$13000 \cdot 1.5 = 19500 \text{ (р.)}.$$

Учитывая, что разработка программно-аппаратного комплекса ведется в течение 9 месяцев, фонд заработной платы будет равен:

$$ФЗП = 19500 \cdot 9 = 175500 \text{ (р.)}.$$

Порядок уплаты страховых взносов во внебюджетные фонды

определяется статьёй 425 НК РФ от 03.07.2016 №243-ФЗ «О страховых взносах в Пенсионный фонд Российской Федерации, Фонд социального страхования Российской Федерации, Федеральный фонд обязательного медицинского страхования».

Начиная с 1 января 2019 г., взносы на социальное страхование составляют 30 % от общей заработной платы. Законом установлены тарифы страховых взносов, описанных ниже:

- Пенсионный фонд Российской Федерации – 22,00 %;
- Фонд социального страхования Российской Федерации – 2,9 %;
- Федеральный фонд обязательного медицинского страхования – 5,1 %.

Таким образом, отчисления составили:

$$175500 \cdot 0,3 = 52650 \text{ (р.)}.$$

Основное средство, подлежащее амортизации при разработке программной системы – это компьютер. Первоначальная стоимость одного компьютера 25000 (р.).

Вычислительная техника входит в амортизационную группу 2. По установленным нормам, постановление Правительства Российской Федерации от 18.11.2006 г. № 697, время службы компьютера берется от 2 лет до 3 лет включительно. Установим норму службы компьютера равную 3 годам (3 года = 36 мес.).

Амортизационные отчисления определяются по формуле

$$A = \frac{\text{Первонач. ст} - \text{сть} \cdot N_a}{100\%} \quad (2)$$

где N_a – норма амортизационных отчислений (%).

Норма амортизационных отчислений составляет:

$$N_a = 1 / 36 \cdot 100 \% = 2,78 \%$$

Для аппаратной части комплекса амортизация рассчитывается по тем же критериям. Таким образом, амортизация за девять месяцев составит:

$$A (9 \text{ мес.}) = 25000 \cdot 2,78 / 100 \% \cdot 9 = 6255 \text{ (р.)}.$$

Величина прочих расходов составляет 15 % от основной заработной

платы. В прочие расходы входит амортизация нематериальных активов, а также накладные расходы:

$$\Pi = 19500 \cdot 0,15 \cdot 9 = 26325 \text{ (p.)}.$$

Таблица 8 – Сценарии ценообразования

Сценарий	Описание	Цена за одну копию
Пессимистический	Спрос – низкий. Ожидаемое количество реализованных экземпляров равно 1. Рентабельность равна 0 %	266978,76 (p.)
Оптимистический	Спрос – высокий. Ожидаемое количество реализованных экземпляров равно 100. Рентабельность равна 25 %	$266978,76 \cdot 1,25/100 = 3337,23$ (p.)
Рационалистический	Спрос – средний. Ожидаемое количество реализованных экземпляров равно 35. Рентабельность равна 5 %	$266978,76 \cdot 1,05 / 35 = 8009,36$ (p.)

Таким образом, общая стоимость затрат на создание программно-аппаратного комплекса составляет:

$$C_{\text{ПО}} = 6248,76 + 175500 + 52650 + 6255 + 26325 = 266978,76 \text{ (p.)}.$$

Программно-аппаратный комплекс является объектом нематериальных активов, и ее первоначальная стоимость оценивается по совокупным затратам на создание, т.е. равна 266978,76 (p.).

Рассмотрим три возможных сценария ценообразования в таблице 8.

Как видно из таблицы 3, цена за один комплекс, в пессимистическом сценарии, слишком высока. Оптимистический же и рационалистический сценарии, предполагают вполне подходящую стоимость для целевой аудитории. Примем оптимистический сценарий.

В таком случае, окончательная цена будет равна 3337,23 (p.)

7.5 Оценка экономической эффективности

Эффективность является общим экономическим понятием. За эффективность принимается характеристика системы с точки зрения результатов ее функционирования и соотношения затрат.

Как простейший случай, экономическая эффективность программного изделия определяется сопоставлением получаемого эффекта от применения программного средства с затратами на программное средство.

Применение программного изделия предполагает повышение производительности труда пользователя программного средства. Это означает, что пользователь выполняет тот же объем работ с меньшими затратами. Сопоставив затраты при первоначальном варианте и в случае использования рассматриваемого программного продукта, получаем оценку экономической эффективности.

Зависимость для определения экономического эффекта может быть представлена в следующем виде:

$$E = C_{\text{б}} - C_{\text{н}},$$

где E – экономический эффект от внедрения новой разработки;

$C_{\text{б}}$ – стоимость обработки экономической информации при использовании базового варианта;

$C_{\text{н}}$ – стоимость обработки экономической информации при предлагаемом варианте.

Расчет экономической эффективности измеряемой точки представлен в соответствии с таблицей 9.

Таблица 9 – Трудоемкость получения данных о работе устройства

Операция	Затраты при добавлении компонентов для вывода информации (чел. часов)	Затраты при использовании программно-аппаратного комплекса (чел. часов)
Получение данных о работе устройства	3	0,5

Трудоемкость объема работ по получению данных о работу устройства в том случае, когда человек сам добавляет компоненты для вывода информации и пишет программный код составляет 3 (чел./час.), а при использовании программно-аппаратного комплекса время сокращается примерно до 0,5 (чел./час.)

Экономический эффект от внедрения новой разработки:

$$E = 3 - 0,5 = 2,5 \text{ (чел./час.)}$$

Использование комплекса позволяет:

1. Снизить затраты на дополнительные компоненты для вывода информации о работе устройства, а также снизить время, затрачиваемое на их установку и программирование.
2. Уменьшить расходы на дорогостоящее оборудование. В состав комплекса входят наиболее дешевые компоненты, широко распространенные на рынке, что приводит к снижению себестоимости комплекса.

8 Безопасность и экологичность проекта

8.1 Требования к разрабатываемому комплексу

Программно-аппаратный комплекс предназначен для подключения к цифровым программируемым устройствам с целью управления, изменения параметров и вывода их на экран. Следовательно, можно обозначить следующие требования к комплексу: степень защиты корпуса должна обеспечивать нормальное функционирование устройства, не давая возможности нарушить его внешним воздействием, а также в случае попадания влаги, и должна соответствовать уровню IP54 [ГОСТ 14254-2015].

Так как комплекс будет подключаться к другим цифровым устройствам необходимо обеспечить токоведущие части комплекса изоляцией, а также предотвратить возникновение помех при передаче данных.

8.2 Требования к ПЭВМ

Эксплуатация персонально электронно-вычислительных машин (ПЭВМ) обязана происходить в соответствии с текущими санитарными требованиями. Каждый тип ПЭВМ должен проходить санитарно-эпидемиологическую экспертизу. Экспертиза проводится в специализированных испытательных лабораториях, которые являются аккредитованными в установленном порядке [4].

Таблица 10 – Уровни допустимых значений звукового давления

Уровни звукового давления в октавных полосах со среднегеометрическими частотами								
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ

Уровни допустимых значений звукового давления, создаваемого

ПЭВМ представлены в таблице 10. Измерение данных уровней звукового давления производится на расстоянии 50 см. от поверхности оборудования.

Временные допустимые уровни (ВДУ) электромагнитных полей (ЭМП), создаваемых ПЭВМ, не должны превышать значений, представленных в таблице 11.

Таблица 11 - Временные допустимые уровни ЭМП, создаваемые ПЭВМ

Наименование параметров		ВДУ ЭМП
Напряженность электрического поля	в диапазоне частот от 5 Гц до 2 кГц	25 В/м
	в диапазоне частот от 2 кГц до 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот от 5 Гц до 2 кГц	250 нТл
	в диапазоне частот от 2 кГц до 400 кГц	25 нТл
Электростатический потенциал экрана видеомонитора		500 В

Допустимые визуальные параметры устройств отображения информации представлены в таблице 12.

Таблица 12 – Допустимые параметры устройств отображения информации

N	Параметры	Допустимые значения
1	Яркость белого поля	Не менее 35 кд/кв. м
2	Неравномерность яркости рабочего поля	Не более 20 %
3	Контрастность (для монохромного режима)	Не менее 3:1
4	Временная нестабильность изображения (непреднамеренное изменение во времени яркости изображения на экране дисплея)	Не должна фиксироваться
5	Пространственная нестабильность изображения (непреднамеренные изменения положения фрагментов изображения на экране)	Не более $2 \cdot 10^{-4}L$, где L – проектное расстояние наблюдения, мм

Концентрация вредных веществ в воздушном пространстве, производимых во время эксплуатации ПЭВМ, не должна превышать предельно допустимой концентрации (ПДК) рассчитанных для атмосферного

воздуха.

Корпус ПЭВМ должен быть окрашен в мягких и спокойных тонах с диффузным рассеиванием света. ПЭВМ и периферийные устройства должны иметь матовую поверхность с отсутствием деталей способных создавать блики. При этом коэффициент отражения поверхности должен варьироваться от 0,4 до 0,6.

8.3 Требования к помещениям для эксплуатации ПЭВМ

Эксплуатация ПЭВМ разрешена в помещениях, в которых присутствует естественные и искусственные источники освещения. В случае отсутствия естественного источника освещения, использование ПЭВМ возможно только при соответствующем обосновании и наличии положительного санитарно-эпидемиологического заключения, которое выдаётся в установленном порядке.

Для внутренней отделки интерьера помещения применяются диффузно отражающие материалы с коэффициентом отражения для потолка от 0,7 до 0,8; для стен от 0,5 до 0,6; для пола от 0,3 до 0,5. Полимерные материалы используются при наличии санитарно-эпидемиологического заключения.

Уровни естественного и искусственного освещения должны соответствовать действующим требованиям нормативной документации. Эксплуатация ПЭВМ наиболее рекомендована в помещениях с расположением окон на северной или северо-восточной стороне. Оконные проемы при этом должны быть оборудованы регулируемыми жалюзи или занавесами, которые позволяют полностью закрыть оконные проемы.

Помещения с размещенными в них рабочими местами ПЭВМ, должны быть оборудованы защитным заземлением, которое удовлетворяет технические требования по эксплуатации. Нежелательна схема размещения рабочих мест ПЭВМ, если вблизи присутствуют силовые кабели,

высоковольтные трансформаторы или технологическое оборудование, которые способны создать помехи в работе ПЭВМ.

8.4 Требования к шуму и вибрации в помещениях с эксплуатируемым ЭВМ

В помещениях с рабочими местами, оборудованными ПЭВМ уровень шума должен быть ниже предельно допустимых значений, установленных для данных видов работ в соответствии санитарно-эпидемиологическими нормативами [5].

Оборудование, имеющее высокий уровень шума (например, печатающее устройство), которое превышает установленные нормативы, должно размещаться вне помещения с рабочим местом ПЭВМ.

При выполнении работ с использованием ПЭВМ в производственных помещениях уровень вибрации не должен превышать допустимых значений вибрации для рабочих мест в соответствии с действующими санитарно-эпидемиологическими требованиями.

8.5 Требования к освещению помещений и рабочих мест с ПЭВМ

Искусственное освещение в помещениях эксплуатации ПЭВМ должно осуществляться системой общего равномерного освещения. В производственных и административно - общественных помещениях, в случаях преимущественной работы с документами, допускается применение системы комбинированного освещения (к общему освещению дополнительно устанавливаются светильники местного освещения, предназначенные для освещения зоны расположения документов).

Освещенность на поверхности стола в зоне размещения рабочего документа должна быть от 300 до 500 лк.

В качестве источников света при искусственном освещении должны применяться преимущественно люминесцентные лампы типа ЛБ. Общее освещение следует выполнять в виде сплошных или прерывистых линий светильников, расположенных сбоку от рабочих мест, параллельно линии зрения пользователя при рядном расположении ПЭВМ. Для освещения помещений с ПЭВМ следует применять светильники серии ЛПО36 с зеркализированными решетками, укомплектованные высокочастотными пускорегулирующими аппаратами (ВЧ ПРА). Допускается применять светильники серии ЛПО36 без ВЧ ПРА только в модификации «Кососвет», а также светильники прямого света - П, преимущественно прямого света - Н, преимущественно отраженного света - В. Применение светильников без рассеивателей и экранирующих решеток не допускается.

Яркость светильников общего освещения в зоне углов излучения от 50 до 90 градусов с вертикалью в продольной и поперечной плоскостях должна составлять не более 200 кд/ м², защитный угол светильников должен быть не менее 40 градусов.

8.6 Требования к организации и оборудованию рабочих мест с ПЭВМ

При размещении рабочих мест с ПЭВМ расстояние между рабочими столами, на которых расположены мониторы, должно быть не менее 2,0 м, расстояние между боковыми поверхностями мониторов не менее 1,2 м. Рабочие места с ПЭВМ по отношению к световым проемам должны располагаться так, чтобы естественный свет падал сбоку на поверхность монитора, преимущественно слева.

Экран монитора должен находиться от глаз пользователя на расстоянии от 600 до 700 мм.

Конструкция рабочего стола должна обеспечивать оптимальное использование свободного пространства рабочего места и свободное

размещение используемого оборудования, с учетом его количества и конструктивных особенностей (размер ПЭВМ, клавиатуры, и др.), характера выполняемой работы. При всем при этом допускается использование столов различной конструкции, соответствующих современным требованиям эргономики.

Рабочий стул (кресло) должен быть выбран с учетом роста пользователя, характера и длительности выполняемой при помощи ПЭВМ работы. Оптимальным считается стул с возможностью регулирования по высоте и углам наклона сиденья и спинки, а также с возможностью поворота на 360 градусов. При этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию.

Поверхность элементов рабочего стула (сиденье, спинка, подлокотники) должна быть полумягкой, с нескользящим, слабо электризующимся и воздухопроницаемым покрытием, обеспечивающим легкую чистку от различных загрязнений. Так же конструкция стула должна обеспечивать: ширину и глубину сиденья не менее 400 мм, поверхность сиденья с закругленным передним краем, стационарные или съемные подлокотники. И обладать возможностью регулировки и надежной фиксации: высоты поверхности сиденья в пределах от 400 до 550 мм и углов наклона вперед до 15 градусов и назад до 5 градусов, высоту опорной поверхности спинки от 280 до 320 мм, ширину - не менее 380 мм и радиус кривизны горизонтальной плоскости – 400 мм, расстояния спинки от переднего края сиденья в пределах от 260 до 400 мм, подлокотников по высоте над сиденьем в пределах от 200 до 260 мм и внутреннего расстояния между подлокотниками в пределах от 350 до 500 мм.

Высота рабочего стола для взрослых пользователей должна регулироваться в пределах от 680 до 800 мм. При отсутствии возможности регулировки высоты рабочей поверхности стола, высота должна составлять 725 мм. Также рабочее место (рабочий стол), должен быть оснащен пространством для ног, высотой не менее 600 мм, шириной – не менее 500

мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног не менее 650 мм.

Обязательно должна проводиться влажная ежедневная уборка помещений и систематическое проветривание после каждого часа работы с ПЭВМ.

8.7 Требования к организации медицинского обслуживания пользователей ПЭВМ

Для лиц, профессионально связанных с эксплуатацией ПЭВМ и, проводящих более 50 % своего рабочего времени за работой с ПЭВМ, должны пройти обязательное предварительное при поступлении на работу и периодические медицинские осмотры в установленном порядке. К непосредственной работе с ПЭВМ допускаются лица, не имеющие медицинских противопоказаний.

Женщины со времени установления беременности и в период кормления грудью не допускаются к работам связанным с использованием ПЭВМ и переводятся на работы, не требующие использования ПЭВМ [6].

Медицинское освидетельствование студентов высших учебных заведений проводится в порядке и в сроке, установленные соответственно Минздравмедпромом России, Госкомсанэпиднадзором России, Госкомвузом России и Минобразования России.

8.8 Требования электробезопасности

Для защиты от поражения электрическим током все токоведущие части и узлы ПЭВМ должны быть надежно защищены от случайных прикосновений кожухами, по правилам устройства электроустановок (ПУЭ), корпус устройства должен быть заземлен. Заземление выполняется при помощи изолированного медного проводника, с сечением 1,5 мм², который, в

свою очередь, при помощи сварки присоединяется к общей шине заземления, с сечением 48 мм². Питание ПЭВМ должно осуществляться от силового щита, через автоматически срабатывающий предохранитель, срабатывающий при коротком замыкании.

Эксплуатация устройства должна производиться лицами, получившими допуск, имеющими квалификационную группу и допуск по электробезопасности не ниже третьего. Работа по устранению неисправностей и наладка должна производиться персоналом с квалификационной группой по технике безопасности не ниже третьей и только после снятия напряжения питания с устройства.

8.9 Меры оказания первой медицинской помощи при поражении электрическим током

Если пострадавший соприкасается с токоведущими частями, необходимо быстро освободить его от действия электрического тока. Прикасаться к человеку, находящемуся под напряжением, опасно для жизни. Поэтому нужно быстро отключить ту часть установки, которой касается пострадавший. Для освобождения пострадавшего от провода следует воспользоваться сухой одеждой, доской или каким-либо другим предметом, не проводящему электрический ток или взяться за его одежду, если она сухая, избегая при этом прикосновения к металлическим предметам и открытым частям тела.

Далее необходимо:

- уложить пострадавшего на спину на твердую поверхность;
- проверить наличие у пострадавшего дыхания (определить по подъему грудной клетки, запотеванию зеркала и пр.);
- проверить наличие пульса на лучевой стороне у запястья или на сонной артерии на переднебоковой поверхности шеи;

- выяснить состояние зрачка, широкий зрачок указывает на резкое ухудшение кровоснабжения мозга;
- вызов врача по телефону 03 во всех случаях обязателен.

ЗАКЛЮЧЕНИЕ

Подводя итоги по проделанной работе, можно отметить, что поставленные на данный момент задачи были выполнены.

Создан программно-аппаратный комплекс ввода/вывода для цифровых программируемых устройств.

В ходе выполнения проекта был реализован программно-аппаратный комплекс. Была реализована программная составляющая, которая представляет из себя прошивку микроконтроллера.

Данный продукт является недорогим и функциональным средством для подключения к цифровым программируемым устройствам, с целью управления, ввода настроек, а также вывода параметров работы устройства.

Из перспектив можно отметить:

- создание приложения, при помощи которого пользователь сможет создавать меню посредством графического интерфейса;
- создание библиотеки для возможности использования блока ввода/вывода с Arduino проектами.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Си (Язык программирования) Википедия [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Си_\(язык_программирования\)](https://ru.wikipedia.org/wiki/Си_(язык_программирования)) (дата обращения: 25.04.2020 г.).
2. I²C Википедия [Электронный ресурс] – Режим доступа: [http://ru.wikipedia.org/wiki/I²C](http://ru.wikipedia.org/wiki/I%C2) (дата обращения: 25.04.2020 г.).
3. Андреев Г.И. Практикум по оценке интеллектуальной собственности: учебное пособие / Г.И. Андреев, В.В. Витчинка. – Москва: Финансы и статистика, 2003. – 176 с.
4. Гигиенические требования к персональным электронно-вычислительным машинам и организации работы. Санитарно-эпидемиологические правила и нормативы СанПиН 2.2.2/2.4.1340-03.
5. Шум на рабочих местах, в помещениях жилых, общественных зданий и на территории жилой застройки. Санитарно-эпидемиологические правила и нормативы СанПиН 2.2.4/2.1.8.562-96.
6. Гигиенические требования к условиям труда женщин. Санитарно-эпидемиологические правила и нормативы СанПиН 2.2.0.555-96.

ПРИЛОЖЕНИЕ А

(Обязательное)

Структура, хранящая данные для создания меню и пример описания меню

```
struct menuItemData {
    uint8_t Level;
    uint8_t Transition[5];
    char *Text;
    uint8_t Select;// 0 - отображаемое значение
    // 1 - изменяемое значение
    // 2 - подменю
    // 3 - действие
    uint16_t Value;
    uint8_t Type;
};

#define ITEMS_COUNT 7
menuItemData menuItems[ITEMS_COUNT] = {
    {0, {}, "Servo", 2},
    {1, {0}, "Angle", 1, (uint16_t)&pos, T_UINT8},
    {1, {0}, "Turn", 3, (uint16_t)&turnServo},
    {0, {}, "Led", 2},
    {1, {1}, "Freq", 1, (uint16_t)&blinkPeriod, T_UINT16},
    {1, {1}, "Blink", 3, (uint16_t)&blinkLed},
    {1, {1}, "On/Off", 3, (uint16_t)&ledOn}
};
```

ПРИЛОЖЕНИЕ Б

(Обязательное)

Структура, принимающая данные для создания меню и структура для хранения пунктов меню

```
struct menuItemData {  
    uint8_t Level;  
    uint8_t Transition[5];  
    uint8_t TextIndex;  
    uint8_t TextLength;  
    uint8_t Select; // 0 - отображаемое значение  
    // 1 - изменяемое значение  
    // 2 - подменю  
    // 3 - действие  
    int16_t ValueAddress;  
    int8_t Type;  
    int8_t ValueIndex;  
};
```

```
struct menuItem {  
    void *Next;  
    void *Previous;  
    void *Parent;  
    void *Child;  
    uint8_t TextIndex;  
    uint8_t TextLength;  
    uint8_t Select;  
    int16_t ValueAddress;  
    int8_t Type;  
    int8_t ValueIndex;  
};
```


ПРИЛОЖЕНИЕ В

(Обязательное)

Код функций добавления элемента в меню

```
menuItem* endLevel(menuItem *currentItem) {
    while (currentItem->Next != &NULL_ENTRY) {
        currentItem = currentItem->Next; // переход к следующему узлу
    }
    return currentItem;
}

menuItem* changeLevel(menuItem *currentItem) {
    currentItem = currentItem->Child;
    return currentItem;
}

void addMenuItem(menuItem *currentItem, menuItem *addedItem, menuItemData
*itemData) {
    if (currentItem != &NULL_ENTRY) {
        if (itemData->Level == 0) {
            currentItem = endLevel(currentItem);
            currentItem->Next = addedItem;
            addedItem->Previous = currentItem;
            addedItem->Next = &NULL_ENTRY;
            addedItem->Parent = &NULL_ENTRY;
            addedItem->Child = &NULL_ENTRY;
            addedItem->TextIndex = itemData->TextIndex;
            addedItem->TextLength = itemData->TextLength;
            addedItem->Select = itemData->Select;
            addedItem->ValueAddress = itemData->ValueAddress;
            addedItem->Type = itemData->Type;
            addedItem->ValueIndex = itemData->ValueIndex;
        } else if (itemData->Level > 0) {
            for (uint8_t i = 0; i < itemData->Level; i++) {
                for (uint8_t j = 0; j < itemData->Transition[i]; j++) {
                    currentItem = currentItem->Next;
                }
                if (itemData->Level - i != 1) {
                    currentItem = changeLevel(currentItem);
                }
            }
        }
        if (currentItem->Child == &NULL_ENTRY) {
            addedItem->Next = &NULL_ENTRY;
            addedItem->Previous = &NULL_ENTRY;
            addedItem->Parent = currentItem;
            addedItem->Child = &NULL_ENTRY;
            currentItem->Child = addedItem;

            addedItem->TextIndex = itemData->TextIndex;
            addedItem->TextLength = itemData->TextLength;
            addedItem->Select = itemData->Select;
            addedItem->ValueAddress = itemData->ValueAddress;
```

```

addedItem->Type = itemData->Type;
addedItem->ValueIndex = itemData->ValueIndex;
} else {
addedItem->Parent = currentItem;
currentItem = changeLevel(currentItem);
currentItem = endLevel(currentItem);
currentItem->Next = addedItem;

addedItem->Previous = currentItem;
addedItem->Next = &NULL_ENTRY;
addedItem->Child = &NULL_ENTRY;

addedItem->TextIndex = itemData->TextIndex;
addedItem->TextLength = itemData->TextLength;
addedItem->Select = itemData->Select;
addedItem->ValueAddress = itemData->ValueAddress;
addedItem->Type = itemData->Type;
addedItem->ValueIndex = itemData->ValueIndex;
}
} else {
addedItem->Next = &NULL_ENTRY;
addedItem->Previous = &NULL_ENTRY;
addedItem->Parent = &NULL_ENTRY;
addedItem->Child = &NULL_ENTRY;

addedItem->TextIndex = itemData->TextIndex;
addedItem->TextLength = itemData->TextLength;
addedItem->Select = itemData->Select;
addedItem->ValueAddress = itemData->ValueAddress;
addedItem->Type = itemData->Type;
addedItem->ValueIndex = itemData->ValueIndex;

headMenu = addedItem;
}
}

```

ПРИЛОЖЕНИЕ Г

(Обязательное)

Код функции обработки сообщений

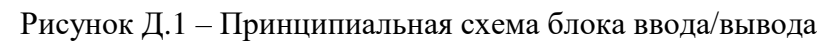
```
void processMessage() {
    uint8_t code = bufferI2C.read();
    if (code == 0) {
        CONNECTED_ADDR = bufferI2C.read();
        connectedFlag = true;
    } else if (code == 1) {
        uint8_t number = bufferI2C.read();
        uint8_t index;
        if (number == 0) {
            itemsCount = bufferI2C.read();
            valueCount = bufferI2C.read();
            strCount = bufferI2C.read();
            menuItems = calloc(itemsCount, sizeof(menuItemData));
            valueReg = calloc(valueCount, sizeof(uint8_t));
            strReg = calloc(strCount, sizeof(char));
            processFlag = true;
        } else if (number == 1) {
            index = bufferI2C.read();
            uint8_t level = bufferI2C.read();
            uint8_t transition[level];
            for (uint8_t i = 0; i < level; ++i) {
                transition[i] = bufferI2C.read();
                menuItems[index].Transition[i] = transition[i];
            }
            menuItems[index].Level = level;
            processFlag = true;
        } else if (number == 2) {
            static uint8_t strRegIndex = 0;
            index = bufferI2C.read();
            uint8_t length = 0;
            while (bufferI2C.movePeek(length) != '\0') {
                ++length;
            }
            for (uint8_t i = 0; i < length; ++i) {
                strReg[strRegIndex + i] = bufferI2C.read();
            }
            menuItems[index].TextIndex = strRegIndex;
            menuItems[index].TextLength = length;
            strRegIndex += length;
            bufferI2C.read();
            processFlag = true;
        } else if (number == 3) {
            static uint8_t regIndex = 0;
            index = bufferI2C.read();
            uint8_t select = bufferI2C.read();
        }
    }
}
```

```

if (select != 2) {
    uint16_t valueAddr = bufferI2C.read() << 8 | bufferI2C.read();
    if (select != 3) {
        uint8_t type = bufferI2C.read();
        for (uint8_t i = 0; i < sizeofValue(type); ++i) {
            valueReg[regIndex + i] = bufferI2C.read();
        }
        menuItems[index].Type = type;
        menuItems[index].ValueIndex = regIndex;
        regIndex += sizeofValue(type);
    } else {
        menuItems[index].Type = -1;
        menuItems[index].ValueIndex = -1;
    }
    menuItems[index].ValueAddress = valueAddr;
} else {
    menuItems[index].ValueAddress = -1;
    menuItems[index].Type = -1;
    menuItems[index].ValueIndex = -1;
}
menuItems[index].Select = select;
processFlag = true;
}
}
}

```

Схемы устройства



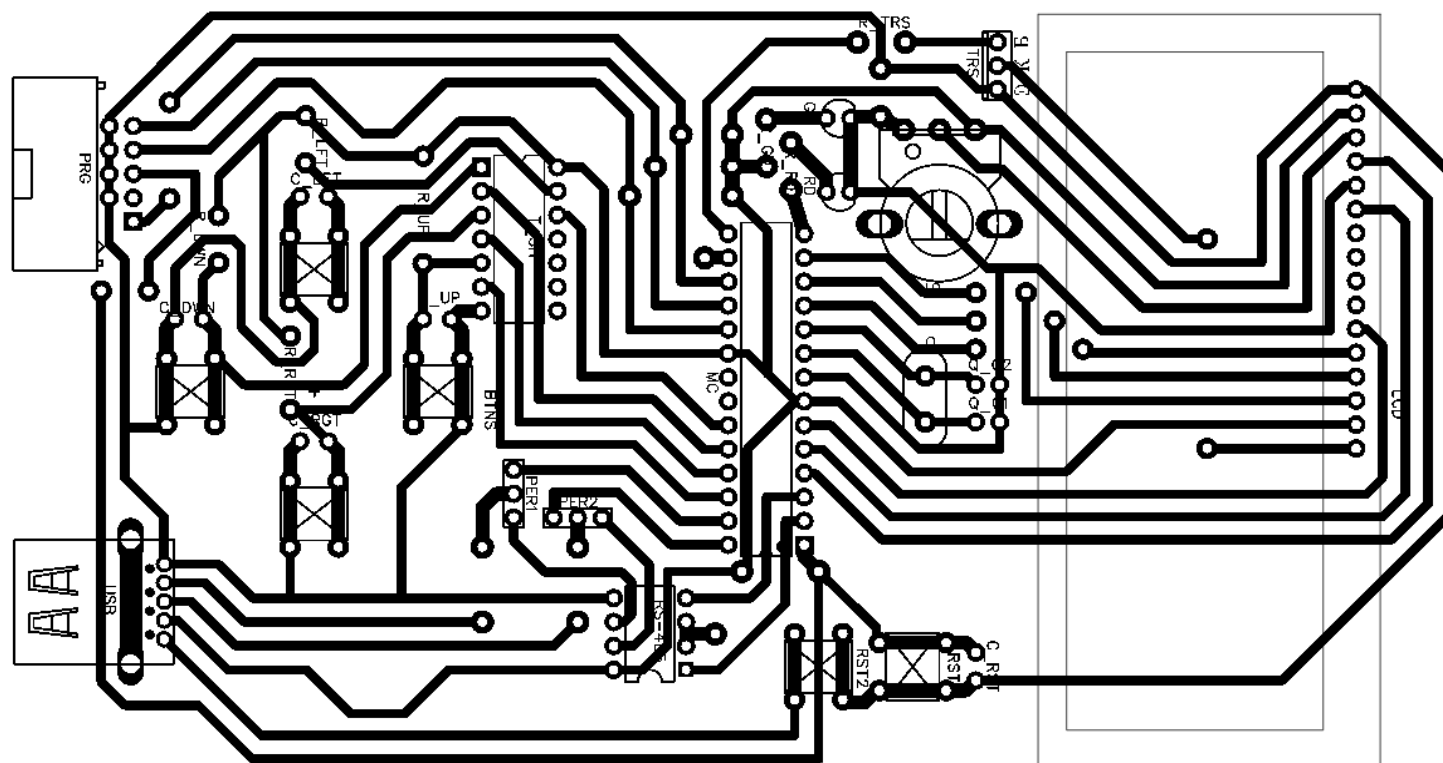
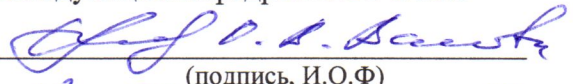


Рисунок Д.2 – Монтажная схема блока ввода/вывода

УТВЕРЖДАЮ

Заведующий кафедрой ИВТ и ПМ



(подпись, И.О.Ф)

«18» 06 2020 г.

Заключение

о результатах проверки выпускной квалификационной работы в системе «Антиплагиат»

В выпускной квалификационной работе обучающегося

Волошинский Сергей Александрович

(фамилия, имя, отчество)

Энергетического факультета группы ВМК-16

Универсальный блок ввода/вывода для цифровых программируемых устройств,


(название выпускной квалификационной работы)

разрабатываемых на базе ООО «Энергоинновационный центр»

доля оригинального текста составляет 73,43 процентов.

Расширенный отчет об источниках, с которыми были обнаружены совпадения
фрагментов текста работы, находится на кафедре ИВТ и ПМ.

Менеджер кафедры


(подпись)

Р.С. Долгих
(И.О.Ф.)

Дата «18» 06 2020 г.

Отзыв

о работе выпускника Забайкальского государственного университета

Виноградова Сергей Александрович

(фамилия, имя, отчество)

по выполнению выпускной квалификационной работы по направлению подготовки _____

09.03.01 Информатика и вычислительная техника

Тема Универсальный блок ввода/вывода для цифровых программируемых устройств, разработанный на базе ООП "Энергосмодуляционный элемент"

Объем работы:

количество листов пояснительной записки 62

количество листов чертежей —

количество схем —

Заключение о степени соответствия выполненной работы заданию работа полностью соответствует поставленной задаче на выпускную квалификационную работу

Проявленная выпускником самостоятельность при выполнении работы работа выполнена выпускником самостоятельно

Плановость и дисциплинированность в работе работа выполнялась строго в соответствии с установленными сроками до сдачи

Умение пользоваться литературным, справочным, нормативным материалом выпускник проявил умение пользоваться технической и справочной литературой

Умение представить результаты выпускной квалификационной работы (в виде плакатов, мультимедийной презентации и др.) результаты ВКР представлены в виде мультимедийной презентации на высоком уровне

Индивидуальные способности выпускника при выполнении работы выпускник проявил самостоятельность, инициативу и творческую деятельность

Положительные стороны работы работа имеет большую практическую составляющую и значимость

Недостатки работы существенных недостатков не выявлено

Характеристика общенаучной специальной подготовки выпускника выпускник имеет отличную общенаучную специальную подготовку

Заключение и предполагаемая оценка работа выполнена самостоятельно, в соответствии с заданием, на высоком профессиональном уровне и заслуживает оценки "отлично" при соответствии всем требованиям

Руководитель

С.М. Исаев

Дата « 15 » июня 2020 г.

