

Министерство науки и высшего образования и Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
учреждения высшего образования
«Пермский национальный исследовательский политехнический университет»

Факультет профессионального образования

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

на тему «Разработка проекта «Умная копилка» на базе микроконтроллера
семейства AVR»
студента группы КСК9-18-1спо по специальности 09.02.01 Компьютерные системы и
комплексы

Худяковой Екатерины Владимировны _____

Руководитель работы: _____ Л.Г.Вилькова

Консультант по
экономической части: _____ К.В.Кондратьева

Консультант по промышленной экологии
и охране труда: _____ А.К.Тороцин

Рецензент: _____ (_____)

Допуск к защите: _____ М.Н.Апталаев

Лысьва, 2022 г

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
учреждения высшего образования
«Пермский национальный исследовательский политехнический университет»
ПЦК «Естественнонаучных дисциплин»

Утверждаю:
Председатель ПЦК
_____ М.Н.Апталаев
«__» _____ 2022 г.

ЗАДАНИЕ
на выполнение выпускной квалификационной работы

студенту ХУДЯКОВОЙ Екатерине Владимировне курса 4

группы КСК9-18-1спо
специальность 09.02.01 Компьютерные системы и комплексы

Тема задания «Разработка проекта «Умная копилка» на базе микроконтроллера семейства AVR»

Структура работы такова:

а) Введение. Аргументировать актуальность выбранной темы, ее теоретическое значение и практическую значимость для организации, сформулировать цель и конкретные задачи исследования. Конкретизировать объект и предмет исследования. Увязать решение темы ВКР с общими научно-техническими задачами цифровизации экономики страны.

б) Исследовательский раздел. Понятие «Устройства с микроконтроллерным управлением». Обзор существующих на рынке копилок. Формирование требований к проектируемой системе.

в) Конструкторский раздел. Обоснование и выбор инструментального обеспечения проекта. Разработка структурной и функциональной схем проекта. Выбор элементной базы. Разработка печатной платы. Разработка алгоритма и управляющей программы поилки. Разработка аппаратной части.

г) Охрана труда и промышленная экология. Анализ вредных и опасных производственных факторов на рабочем месте инженера-электроника. Разработка средств защиты от воздействия выбранного ВиОПФ. Экологические требования к утилизации вычислительной и оргтехники, а также их расходных материалов.

д) Организационно-экономический раздел. Выполнение технико-экономической оценки проекта.

е) Заключение. Краткое изложение решенных задач, актуальность работы, соответствие полученных результатов теме и заданию ВКР.

ж) Список использованных источников.

з) Приложения.

ВВЕДЕНИЕ

1 Исследовательский раздел

1.1 Сущность устройства с микроконтроллерным управлением

1.2 Обзор электрических схем копилок

1.3 Формирование требований к проектируемой системе

2 Конструкторский раздел

2.1 Выбор инструментального обеспечения проектирования

- 2.2 Разработка структурной и функциональной схем устройства
- 2.3 Выбор элементной базы
- 2.4 Разработка печатной платы
- 2.5 Разработка алгоритмов работы управляющей программы устройств. Разработка программного обеспечения устройства
- 3 Организационно-экономический раздел
 - 3.1 Расчет себестоимости проекта
 - 3.2 Расчет экономической эффективности проекта
- 4 Охрана труда и промышленная экология
 - 4.1 Анализ вредных и опасных производственных факторов при пайке деталей, узлов и наладке электронных устройств
 - 4.2 Расчет технических средств обеспечения безопасности труда на рабочем месте инженера-электроника
 - 4.3 Утилизация компьютерной и оргтехники техники

ЗАКЛЮЧЕНИЕ
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ
ПРИЛОЖЕНИЯ

Дата выдачи _____
Срок окончания _____

Руководитель ВКР
_____/Л. Г. Вилькова/
«__» _____ 2022 г.

Задание утверждено на заседании ПЦК «Естественнонаучных дисциплин» протокол №____
от _____ 2021 г.

Председатель ПЦК _____ / М.Н. Апталаев /
«__» _____ 2022 г.

Министерство науки и высшего образования Российской Федерации
Лысьвенский филиал федерального государственного автономного образовательного
 учреждения высшего образования
 «Пермский национальный исследовательский политехнический университет»

ПЦК «Естественнонаучных дисциплин»

Утверждаю
 Председатель ПЦК
М.Н. Апталаев
 «15» февраля 2022 г.

График выполнения выпускной квалификационной работы

студента группы КСК9-18-1сн

специальности 09.02.01 Компьютерные системы и комплексы

Худяковой Екатерины Владимировны
 (фамилия, имя, отчество)

Содержание	Сроки	Отметка выполнения	Дата							Подпись	
			08.03-20.04	21.04-30.04	01.04-17.05	18.05-25.05	26.05-30.05	31.05-06.06	07.06-15.06		16.06-23.06
Введение	08.03 –										
1. Теоретическая часть	20.04										
2. Охрана труда и БЖД	20.04 –										
	30.04										
3. Промышленная экология	01.05 –										
	17.05										
4. Инженерно-технической часть	18.05-25.05										
5. Организационно-экономическая часть	26.05-30.05										
Оформление дипломного проекта: Заключение Список используемой литературы Приложения	31.05 – 06.06										
Выполнение чертежей, оформление демонстрационных материалов к ВКР	01.06-03.06										
Представление проекта на проверку и отзыв руководителя <u>Нормоконтроль</u>	03.06 – 05.06										
Подготовка доклада и демонстрационных материалов к предзащите	04.06 – 06.06										
Предоставление работы на проверку председателю ПЦК	04.06 – 05.06										
Предзащита ВКР Иметь к преддипломной защите: - рецензия <u>нормоконтроля</u> ; - отзыв руководителя; - подпись руководителя по экономической части - подпись руководителя по охране труда - подпись руководителя по промышленной экологии - презентация	06.06 – 07.06										

- доклад																			
Устранение замечаний по всей ВКР	08.06- 15.06																		
Рецензирование Сдача работ на кафедру	20.06																		
Диск с материалами ВКР	21.06																		
Защита ВКР	22.06 – 23.06																		

Руководитель ВКР _____ / _____ /

« ____ » _____ 2022 г.

Студент _____ / _____ /

« ____ » _____ 2022 г.

Согласовано:

Председатель ПЦК «Естественнонаучных дисциплин» _____ / М.Н. Аптаев /
« ____ » _____ 2022 г.

РЕФЕРАТ

Худякова Е.В. РАЗРАБОТКА ПРОЕКТА «УМНАЯ КОПИЛКА» НА БАЗЕ МИКРОКОНТРОЛЛЕРА СЕМЕЙСТВА AVR, выпускная квалификационная работа: стр. 78, рис. 16; табл. 12, прилож. 3, части 4, исп. ист. 20.

МИКРОКОНТРОЛЛЕР, ЭЛЕМЕНТНАЯ БАЗА, АЛГОРИТМ РАБОТЫ, ПРОГРАММНЫЙ КОД, УМНАЯ КОПИЛКА.

Объект исследования – копилка для денег.

Цель выпускной квалификационной работы – разработка проекта «Умная копилка» на базе микроконтроллера семейства AVR.

В ходе работы был разработан проект «Умная копилка» на базе микроконтроллера семейства AVR».

Копилки нужны для каждой семьи, потому что в каждом доме имеется место для хранения монет. Просто и удобно в специальную емкость складывать мелкие вещи, но для их извлечения нужно вытряхнуть или разбить ее. Поэтому, разработка «Умной копилки» позволит людям узнавать сумму монет находящуюся в ней, не доставая монеты, что в разы упростит подсчет денег. Данное устройство будет простым в эксплуатации и легко заменит обычную копилку.

Основные конструктивные и экономические показатели: простота использования.

Разработанное устройство может применяться в быту. Оно подойдет как взрослым, так и детям.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	9
1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ	11
1.1 Сущность устройства с микроконтроллерным управлением	11
1.2 Обзор электрических схем копилок	23
1.3 Формирование требований к проектируемой системе	32
2 КОНСТРУКТОРСКИЙ РАЗДЕЛ	35
2.1 Выбор инструментального обеспечения проектирования	35
2.2 Разработка структурной и функциональной схем устройства	38
2.3 Выбор элементной базы	40
2.4 Разработка печатной платы	43
2.5 Разработка алгоритмов работы управляющей программы устройств. Разработка программного обеспечения устройства	44
3 ОХРАНА ТРУДА И ПРОМЫШЛЕННАЯ ЭКОЛОГИЯ	45
3.1 Мероприятия по снижению воздействия выявленных вредных и опасных производственных факторов	45
3.2 Мероприятия по оптимизации освещенности на рабочем месте	46
3.3 Утилизация использованного оборудования при разработке «Умной копилки»	49
3.4 Вывод по разделу	50
4 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКАЯ ЧАСТЬ	52
4.1 Расчет затрат на разработку программы для микроконтроллера	52
4.2 Расчет затрат на внедрение программы для микроконтроллера	55
4.3 Расчет эксплуатационных текущих затрат по программе для микроконтроллера	58

4.4 Расчет экономической целесообразности разработки и внедрения информационных технологий.....	60
ЗАКЛЮЧЕНИЕ	66
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	67
ПРИЛОЖЕНИЕ А – Техническое задание	68
ПРИЛОЖЕНИЕ Б – Код программы.....	68
ПРИЛОЖЕНИЕ В – Блок-схема.....	68

ВВЕДЕНИЕ

Основная особенность автоматизированных устройств – относительная автономность. Пользователям нужно только настроить устройства по имеющимся пожеланиям и предпочтениям, а потом встроенную функцию использовать без каких-либо ограничений. Нужно только обеспечить изделиям периодическую подзарядку и установить дополнительную программу (только в случае соответствующей надобности).

У умных устройств существуют следующие функциональные возможности:

- удаленное управление, т.е. большинство автоматизированных устройств могут быть легко управляемыми на расстоянии, например, через телефон. Этот вариант позволяет изменять настройки, а также отслеживать любые отклонения.

- автоматизация процесса, в многих случаях для того, чтобы выполнить функцию, достаточно одной кнопки. В системе уже заложены все необходимые программы. Остается только их активация.

- универсальность, современное устройство автоматизации обладает множеством возможностей различных направлений. Например, умные часы позволяют не только определить время, но и выполнить экстренный звонок.

Копилки нужны для каждой семьи, потому что в каждом доме имеется место для хранения монет. Просто и удобно в специальную емкость складывать мелкие вещи, но для их извлечения нужно вытряхнуть или разбить ее. Поэтому, разработка «Умной» копилки позволит людям узнавать сумму монет находящуюся в ней, не доставая монеты, что в разы упростит подсчет денег. Данное устройство будет простым в эксплуатации и легко заменит обычную копилку.

Объект исследования – средства учета и хранения денежных средств.

Предмет исследования – проектирование автоматизированных счетных устройств.

Цель выпускной квалификационной работы – разработка проекта «Умная копилка» на базе микроконтроллера семейства AVR.

В соответствии с поставленной целью необходимо решить следующие задачи:

- проанализировать предметную область исследования и выполнить анализ существующих копилок на рынке;
- разработать технические требования к проектируемой системе;
- разработать аппаратную и программную части автоматизированного устройства;
- выполнить технико-экономический расчет проекта;
- проанализировать и выбрать элементную базу и инструментальное обеспечение проектирования.

1 ИССЛЕДОВАТЕЛЬСКИЙ РАЗДЕЛ

1.1 Сущность устройства с микроконтроллерным управлением

Микроконтроллер это микросхема, состоящая из:

- центрального процессора, он включает в себя регистры, блок управления, ПЗУ ;
- периферии, с портами ввода-вывода, генераторы импульсов, таймеры контроллеры прерываний, аналоговые преобразователи и похожие элементы.

Некоторые называют микроконтроллер – микропроцессором, что является не совсем верным утверждением, Микропроцессор выполняет только логические и математические операции. Часть микроконтроллера также включает микропроцессор с другими элементами, являющимися лишь частью микроконтроллера. На рисунке 1 представлен микроконтроллер ATtiny2313 американской фирмы Atmel.

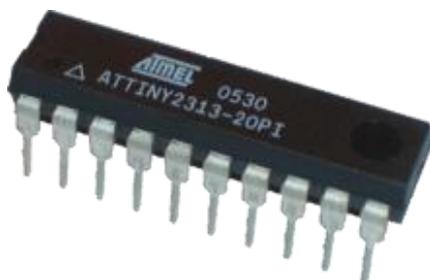


Рисунок 1 – Микроконтроллер ATtiny2313 [4]

На рисунке 2 показано устройство микроконтроллера AVR.

У микроконтроллера очень прост принцип работы, несмотря на то, что у него сложная конструкция. Он основывается на похожем принципе действий. Система знает только две команды («есть сигнал» и «нет сигнала»). В его память записывается код любой команды из этих сигналов. Он выполняет ее, когда происходит считывание команды микроконтроллером.

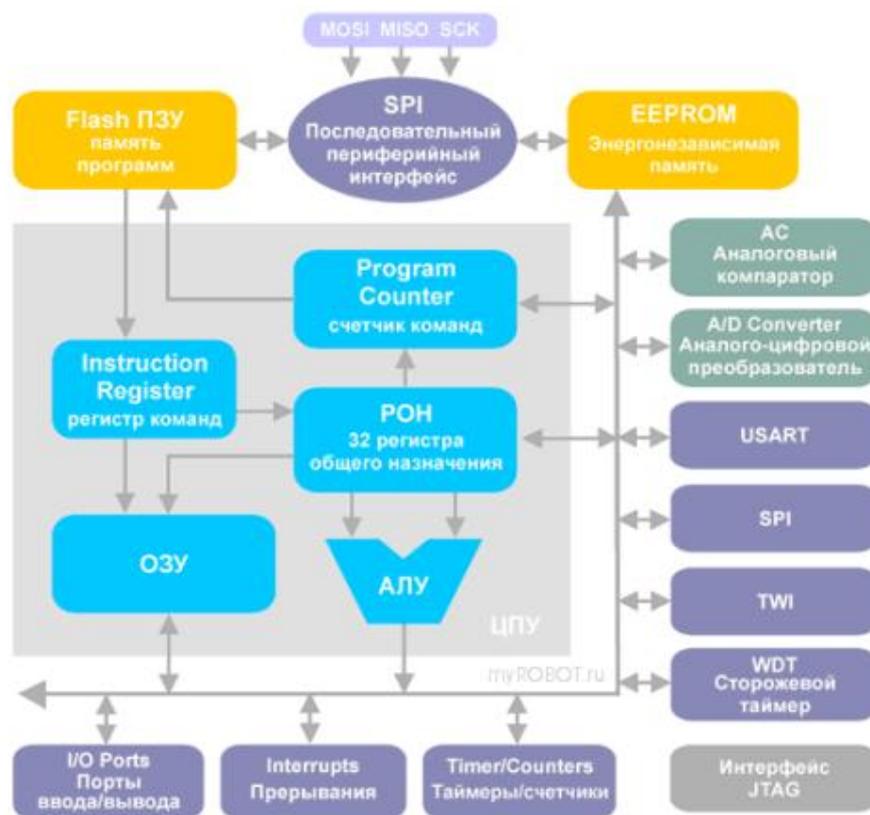


Рисунок 2 – Устройство микроконтроллера AVR [4]

У микроконтроллера очень прост принцип работы, несмотря на то, что у него сложная конструкция. Он основывается на похожем принципе действий. Система знает только две команды («есть сигнал» и «нет сигнала»). В его память записывается код любой команды из этих сигналов. Он выполняет ее, когда происходит считывание команды микроконтроллером.

Базовый набор команд имеет каждый микроконтроллер. И только он может их принимать и выполнять. При объединении между собой отдельных команд, можно составить уникальный программный комплекс, в котором любое электронное устройство будет работать точно так, как нужно.

В зависимости от находящихся в микроконтроллере программного набора, они подразделяются на:

- CISC – базовые команды комплекса большого числа;
- RISC – только нужные команды.

В многих МК присутствует набор RISC. Это объясняется тем, что изготовление такого микроконтроллера легче, оно дешевле, и у разработчиков электронной техники пользуется спросом.

Благодаря очень простому использованию микроконтроллеров AVR, они обладают высокой интегрированной способностью и низкой потребляемой мощностью, их применение разнообразно:

- железнодорожный транспорт;
- светофоры, семафоры;
- управление шлагбаумами;
- медоборудование;
- бытовая техника;
- электронные музыкальные инструменты;
- компьютеры, телефоны;
- электронные детские игрушки;
- промышленное оборудование;
- самолето- и судостроение;
- робототехника;
- автомобилестроение.

В этом перечне представлены не все области применения МК.

Назначением МК является контроль всех процессов, происходящих на его платформе.

МК контролирует состояние некоторых переменных, а также при динамическом режиме изменение системы.

Работа МК требует энергии. Напряжение микроконтроллера ATMEЛ AVR работает в диапазоне 1.8-5.5 Вольт, в зависимости от серии и модели. В основном устройства работают от 5 Вольт. Также могут встретиться низкочастотные модели (Attiny 2313) с нижней границей от 1,8 В.

Частота поступающего тока также влияет на работу МК. Также не высоких частотных пределов требуют низкие напряжения. Чем больше частоты, тем более быстро работают большинство моделей.

Чтобы работали контроллеры AVR серии, на входы с плюсом необходимо подать 5 В и заземлить нулевой.

Подача питания на аналогово–цифровой преобразователь осуществляется через дополнительный фильтр. Это позволит избежать помех, способных изменить показания на напряжение. При этом напряжение подается на плюсовой ввод через фильтрующий дроссель. На цифровые и аналоговые делаются выводы нулевого типа. Их соединение происходит только в одной точке

Также нужно установить конденсаторы, лучше керамические, в расчете 1 на 100 нанофард.

На рисунке 3 представлена схема универсального модуля.

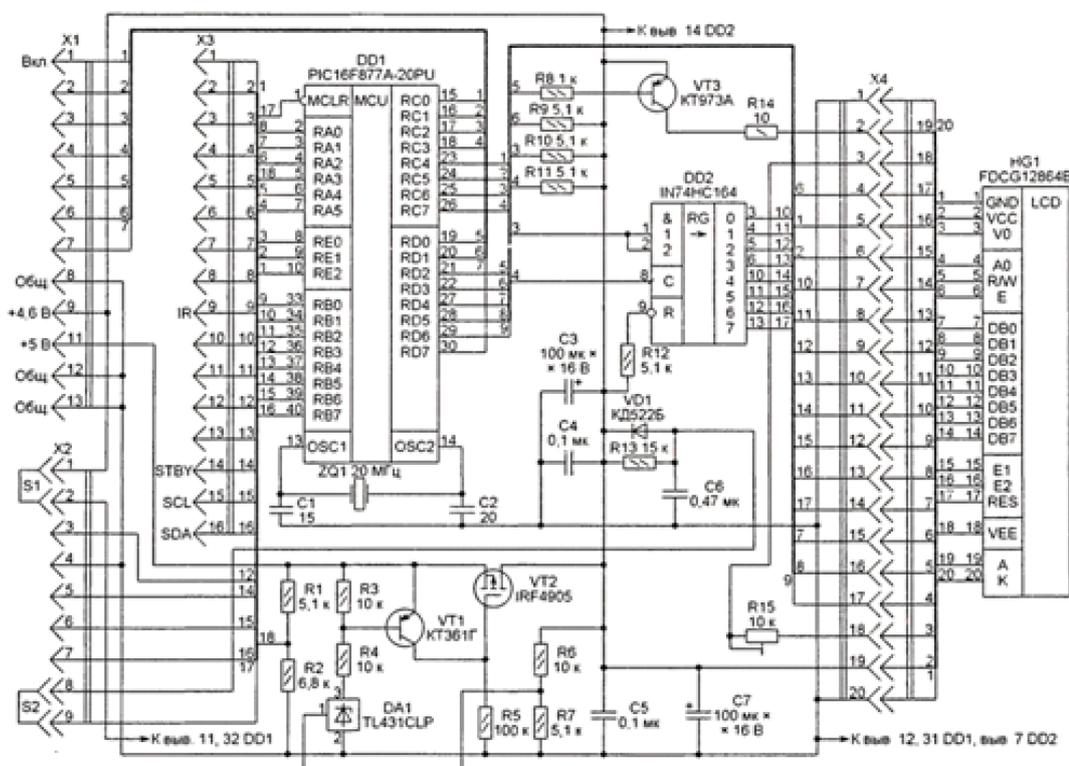


Рисунок 3 – Схема универсального модуля [5]

Микроконтроллер позволяет подключить любой девайс в локальную сеть. В пример такого можно рассматривать Ethernet.

Можно использовать модель OSI для понятия работы локальной сети. В которую входят несколько уровней:

- прикладной, в нем происходит решение задач прикладного характера, для них он и был создан. По эксклюзивному или стандартному протоколу он обменивается данными с внешним миром;

- транспортный, связывает между собой узлы. Предоставление данных транспортным уровнем передается другому узлу и представляется в виде пакета сетей . Он может одновременно отправлять группы пакетов. Если применялся протокол установки соединения, то транспортный уровень установит соединение перед отправкой, проверяет его качество, и лишь после этого передает пакет с данными.

- сетевой. по нему идет передача пакетов. Они передаются по нескольким сетям, различным по физическим и канальным технологиям.

- канальный, через него происходит передача Ethernet-фреймов между локальными узлами.

Каждый следующий уровень обслуживается ранее или ниже. Таким образом появляется вертикальная межуровневая связь. Все уровни обслуживания скрыты от других.

Взаимодействуя с двумя сетями, уровень одной сети контактирует с другой. Благодаря этому появляются горизонтальные связи.

Управления МК выделяет следующие способы:

- проводной путь, управление исполнительными механизмами осуществляется через электрическое соединение цепей исполнительного механизма и управления и. Включение – нажав кнопку на пункт диспетчера или пульт с кнопками;

– беспроводной путь, для данного способа не требуется проводное соединение. С пульта дистанционного управления или передатчика передается сигнал, идущий на приемник.

Соединения беспроводных сигналов подразделяются на:

– оптическими, управление домашней техникой: кондиционеры или телевизоры;

– радио, например: Bluetooth, Wi-Fi и др.

Современные средства связи осуществляют управление контроллерами как через ПЗУ, расположенные вблизи прибора, так и через интернет из любого уголка мира через сеть локальной связи.

Обладает поддержкой сети Wi-Fi МК ESP 8266. Продаваться он может как в виде микросхемы, так и быть распаянным. Его ядро 32-битное, через последовательный порт UART происходит программирование. Существуют более современные платы, которые могут быть прошиты по USB (NodeMCU). Они хранят данные, к примеру, записанные из датчика. Данные платы работают с разными интерфейсами, в том числе. SPI, I2S.

Функции:

- аудиопроигрыватель;
- формирование на выходе ШИМ сигнала;
- канал АЦП;
- таймер;
- планировщик задач и многое другое;

Плату можно использовать как самостоятельное устройство, и как модуль МК для беспроводного соединения.

Тактовая частота микроконтроллера — это выполненное контроллеров количество тактов за секунду. Чем выше, тем больше операций она может выполнять.

Способы тактирования микроконтроллеров зависят от использования:

– внутреннего RC-генератора, работа осуществляется только на частоте 1, 2, 4, 8 МГц. Он не подойдет, если будет необходим частотный режим другого типа. Если необходимо использовать точный временный интервал, этот метод нельзя применять, т.к. его заданная частота изменяется в зависимости от температуры;

– внешнего кварца, более сложный способ соединения. Объём конденсатора должен быть в интервале от 15 до 22 пФ. Один выход соединяется с резонатором, а второй заземляется;

– внешнего генератора, также этот генератор нестабилен в разных температурах, как и внутренний генератор;

– RC-цепочек, для этой схемы подойдёт конденсатор с ёмкостью от 22 пФ и резистором 10-100 кОм.

Внешний и внутренний генератор подойдут для простых МК, а также RC-цепочки. Для более точных микроконтроллеров нужны стабильные тактовые источники.

Под ядром микроконтроллера предполагается набор команд, циклическая работа процессором, организация программной памяти и базы данных, основной набор периферийных устройств, система прерываний.

Различаются между собой представители одного семейства в объеме баз данных и памяти программ, разнообразием периферийных устройств.

Схожесть двоичного программного кода объединяет все МК к семейству.

Подразделение семейств:

– MSC-51, производства Intel – монокристалльный МК на базе архитектуры Гарварда. Главным представителем семейства является 80C51, выполненная по CMOS-технологии. И хоть были разработаны эти контроллеры в 80-е годы прошлого века, они все еще широко применяются.

– PIC (Microchip) – Это МК Гарвардской архитектуры. В его основе лежит архитектура с меньшим числом команд и команд с встроенной

памятью и данных, низким энергопотреблением. Оно включает в себя более 500 разных микроконтроллеров (8-ми, 16-ти, 32-битные) с разнообразной периферией, памятью и другими характеристиками.

– AVR (Atmel) – это высокоскоростные контроллеры разработка которых происходит на собственном архитектурном проекте. Основа контроллера - Гарвардский RISC-процессор, который имеет самостоятельный доступ к программной памяти и базе данных. Каждая из 32 регистров общего назначения может работать в роли регистра-аккумулятора и совместной 16-битной команды.

– ARM (ARM Limited), разработаны на собственном архитектурном проекте. В состав семейства входят 32-битные и 64битные микроконтроллеры. ARM Limited специализируется только на разработке ядер и их инструментов, и на производство других компаний лицензиях.

– STM (STMicroelectronics) – 8-разрядного типа (STM8) контроллеры попадают в категорию высоконадежных изделий с маленьким энергопотреблением. В этом же семействе имеются контроллеры STM32F4 и STM, их основой является 32-битный Cortex.

Внешних различий между микроконтроллерами и другими микросхемами нет. Кристаллы размещаются в корпусах, которые имеют определенное количество входов. Все микроконтроллеры производятся только в трех типах корпуса:

Функционал микропроцессора в компьютере (Micro Processor Unit — MPU) лежит на одном кристалле полупроводникового типа. Соответствует центральному процессору компьютера ЦП (Central Processing Unit — CPU) по своим характеристикам. Применяются: управление системами, выполнение арифметико-логических операций, хранение данных.

Микроконтроллер с входных периферийных устройств получает данные, обрабатывают и передают их выходным периферийным устройствам.

Микроконтроллер включает в себя микропроцессор с необходимыми опорными устройствами, которые находятся в одном чипе. Если необходимо создать устройство, связанное с внешней памятью блоком или блоком ЦАП/ФЦП, то необходимо всего лишь подключить с постоянным напряжением источник питания, источник тактовой частоты и цепь сброса.

Каждый вид контроллеров обладает периферическими устройствами, работающими в автономном режиме, то есть независимо от центрального ядра. После выполнения своей задачи периферическое устройство может сообщить об этом ЦП, а может не сообщить. В зависимости от того как он запрограммирован.

Следующие устройства могут быть на микроконтроллерах:

- аналоговый компаратор, Главной задачей его является сравнение поступающего (измеряемого) напряжения с идеальным;

- аналогово-цифровой преобразователь – происходит замер аналогового напряжения в течение времени и дает ее в цифровом виде. Не у всех есть микроконтроллеры;

- счетчик/таймер - это сочетание двух форм таймера и счетчика. Таймер ставит временные интервалы, а цифровой производит подсчет количества импульсов, приходящих от внутреннего генератора частот или сигналы от внешних источников;

- сторожевой таймер - его задачей является через определенное время производить перезапуск программы;

- модуль прерываний - сообщает микроконтроллерам о случившемся событии и прекращает выполнение программы. После завершения мероприятия возобновляется прерванная программа.

Этапы пути программирования:

- 1) перед началом написания кода программы нужно определить конечную цель,

- 2) составляется программный алгоритм,

3) написание кода программы. Коды написаны на языке Си или Ассемблере,

4) компиляция программы, то есть перевод ее на шестнадцатеричную или двоичную систему 1 или 0. Микроконтроллер только так сможет ее понять,

5) откомпилированные коды вносятся в память контроллера,

6) прошиваем микроконтроллер через программатора. Они имеют два типа соединения: через порт COM и USB. Самый простой и недорогой из программаторов USBASP,

7) проверка и установка микроконтроллера на реальных устройствах.

Язык программирования МК немного отличается от классического компьютерного языка. Основным отличием микроконтроллеров является его ориентированность на работу по периферии. Для архитектуры микроконтроллеров необходимы битово-ориентированные команды. Поэтому были созданы специальные языки для контроллеров:

– Ассемблер. Программы, написанные на нем, получаются большими и непонятными. Но всё же он позволяет максимально полностью раскрывать все возможности контроллеров и получать максимальную скорость и компактный код. Хорошо подходит для малых 8-битных микроконтроллеров.

– C/C++. Уровень языка более высокий. Написанная на нем программа более понятна. В настоящее время существует множество программных средств и библиотек для написания кодов на этом языке. Его компиляторы имеются почти на любой модели микроконтроллера. Является основным языком программирования контроллеров.

От необходимого качества кода и решаемых задач зависит выбор языка для программирования. Если нужен компактный код, то будет подходить Assembler, для решения более серьезной задачи выбор будет ограничен только C/C++.

На сегодняшний день универсальную среду программирования для микроконтроллеров не найти. Это обусловлено его внешней структурой, наличием технических средств для записи кодов в память контроллера.

Среды программирования:

- FlowCode – универсальная графическая среда. Программируется при построении логических структур блок-схем;

- Algorithm Builder. Коды пишутся в 3-5 раз быстрее Flowcode. В нём находятся графические редакторы, компиляторы, симулятор микроконтроллера, внутрисхемные программаторы. Она объединяет Ассемблер и C/C++. Функции среды позволяют самостоятельно прошивать микроконтроллер;

- Image Craft. Как и ранее, поддерживает языки Ассемблера и C/C++. В её составе имеется библиотека, которая позволяет работать с различными устройствами микроконтроллеров. Популярная среда для любителей. Имеет язык подобный Си, но отличается от других языков. Он более понятен для человека. Поддерживает библиотеки, в которых имеются драйверы для соединения каких-либо платформ.

Бывают платные и бесплатные среды. При выборе определенной среды необходимо отталкиваться от ее функциональности, поддерживаемого интерфейса и портов, языка программирования,.

Прежде чем начинать программирование микроконтроллеров, необходимо определиться с языком. Рекомендуется начинать с ассемблера. Хотя это понятие достаточно сложное, но, приложив силы, и все же понимая его логику, тогда будет ясно, что происходит в контроллере.

Можно начать работу с Си, если Ассемблер окажется сложным. Его преимущество заключается в том, что хорошо перемещает коды от одного типа микроконтроллеров к другому. Но для того, чтобы все прописать верно, нужно разделить рабочий алгоритм и его реализацию в машине в разных частях проекта. Таким образом, алгоритм будет перенесен в другой

контроллер, передавая интерфейсный слой, где прописано обращение к «железу», оставляя рабочие коды без изменения.

Затем действие происходит по следующей схеме:

- 1) установка среды и выбор компилятора;
- 2) выбор нового проекта в среде и ее запуск. Нужно указать местоположение. Необходимо выбирать самый короткий путь;
- 3) проектная настройка. Классическое действие - создать make-файл, где будет прописана вся зависимость. Частота работы микроконтроллера указывается на первой странице;
- 4) путевая настройка. В них необходима добавка директории проекта. Возможна добавка сторонних библиотек;
- 5) постановка задач;
- 6) сбор схемы. На данном этапе необходимо соединить модуль конвертера USB-USART с аналогичным выводом микроконтроллера. Это даст возможность прошить микроконтроллер без программатора. Необходимо накинуть джамперы, которые соединяют диоды LED1 и LED2. С данным действием возможно подключить светодиоды LED1 и 2, подключенные к выводам PD4 и PD5;
- 7) пропись кода;
- 8) добавление заголовков с определениями и библиотек;
- 9) основные возможности. Из одних функций состоит язык Си. Они могут быть вложенными и вызываться в любом порядке и разными способами. Но у всех есть три обязательных показателя: тело функции; передаваемые параметры; возвращаемое значение. В соответствии с данными, все передаваемые или возвращаемые значение должны обладать определенным типом;
- 10) запуск эмуляции и компиляция;
- 11) программная отладка.

После написания программы на языке Си можно наблюдать за тем, как и что происходит в микроконтроллере. Это позволит построить аналогию с программированием в Ассемблере.

1.2 Обзор электрических схем копилок

Несмотря на то, что пластиковые карты и счета в банках широко распространены, копилки не теряют свою популярность, выступая в качестве сувенира, талисмана и декоративного элемента интерьера. В продаже представлены всевозможные емкости для того, чтобы хранить деньги, которые выполнены на любой вкус и кошелек. Для того чтобы выполнить удачную покупку, нужно знать, что нужно учитывать при выборе покупки.

Китай предположительно является родиной копилки. Предметы для накопления монет появились в древности почти сразу после возникновения финансов. В Тунисе археологи обнаружили один из самых старых образцов, который был создан в первой половине тысячелетия до нашей эры. Для производства емкости чаще использовались керамические, гипсовые, фарфоровые, металлические и деревянные материалы. В первых копилках не было отверстий, чтобы можно было извлечь деньги, поэтому их было необходимо разбивать. Со временем стали появляться более удобные в использовании многоразовые изделия. В средние века нередко в качестве копилки брали обычные предметы, такие как вазы или кувшин.

Большинство приспособлений для того, чтобы накопить деньги, были фигуры различных зверей, но самый распространенный вариант - свинья, символизирующий процветание и богатство. Многие англоязычные страны используют словосочетание «piggy bank», означающее «свинья-копилка», относится к любым емкостям, независимо от формы и их использования.

Для хранения монет в дореволюционной России использовались чулки и сосуды из керамики, их часто опускали и закапывали в земле. Затем для этого стали использовать обыкновенные кружки, к примеру, на сбор средств

для постройки храмов. В наше время появилось распространение металлических шкатулок, запирающихся на замок.

Удивительной разнообразностью отличаются современные изделия. Появились модели механических и магнитных изделий, емкости для хранения бумажных банкнот и устройства для подсчета накопленной суммы. Необычным вариантом является – копилка бомба от производителя из Японии TomuCo. Если владелец забывает положить деньги вовремя в нее, то она «взрывается» и рассыпает вокруг все накопленные деньги.

Лучшие бренды предлагают огромный ассортимент копилочек: бюджетных и дорогих моделей, новинок и моделей имеющих более высокий спрос, приспособлений для детей и взрослых. Все копилки могут быть разделены на следующие категории:

- одноразовые. Это копилки из гипса, фарфора или керамики, которые можно применять только один раз. Для того, чтобы извлечь деньги из этой модели, необходимо разбить ее;

- многоразовые. Такой тип копилочек можно изготовить из разных материалов: керамики, пластика, дерева, стекла. Внизу емкость, как правило, закрыта пробкой, так что ее владельцу не нужно беспокоиться, о том как извлечь накопленные деньги.

Есть еще 2 вида копилочек:

- статичные. Это стандартные фигурки в виде животных, баночки, фигурки, сундучок;

- интерактивная. Работают на батарейках и используют более широкий функционал. Популярность подобных моделей объясняется всевозможностью их движения и передачи звука, что делает процесс накопления забавным и интересным.

Современные копилки обладают несколькими способами их использования: хранение денег, игрушек, подарков.

В зависимости от изделия выделяют следующие типы:

- наличие сертификатов о гигиене детских товаров;
- материалы – дерево, пластик, керамика;
- одноразовые (при полном заполнении – разбиваются), многоразовые (предусмотрена крышка или отверстие, чтобы доставать деньги);
- дополнительными функциями – будильник и часы;
- форма – бывают в виде животных (классические) и необычные (рисунок 4).



Рисунок 4 – Необычная копилка для денег [8]

Самая большая копилка – Рейчел, свинка из бронзы, весом 270 килограммов. рядом с рынком Pike Place Market в Сиэтле, США, когда была открыта выставка сельских хозяйств в 1977 году. За год можно собрать 8.500-9.000 долларов США. Собранные деньги отдаются в социальную поддержку работникам рынка.

Для изготовления различных моделей копий используются:

- фарфор – дорогой и хрупкий материал, коллекционными моделями;
- керамика – является самым популярным и экологическим материалом;
- металл – долговечный;

- дерево – изготавливаются этнические изделия и сувениры;
- пластик – это прочный и легкий материал, разной формы и цветов.

Форма копилки имеет различное значение (согласно верованиям отдельных народов):

- свинья, является символом накопления и бережливости (рисунок 5);
- сова – трата, практическое применение;
- кошка - привлекает дополнительные источники доходов, богатых покровителей;
- собака – рациональная трата, защита от краж;
- лягушка – привлекает деньги, восточный символ богатства.



Рисунок 5 – Копилка для денег в форме свиньи [8]

Для того чтобы не теряться в огромном ассортименте копилочек и выяснить, какую лучше приобрести, нужно учитывать критерии выбора:

- цена: большинство людей при выборе ориентируются только на цену копилки, стараются купить изделие подешевле. Финансовые возможности, конечно, у всех различны, но не стоит тратить время на чрезмерную дешевизну и приобретать некачественные изделия;

– производитель: очень важный вопрос - какую фирму лучше купить. Рекомендации, как выбрать производителя, очень просты: необходимо выбирать продукцию известных и надежных брендов;

– материалы: для создания копилки используются различные материалы. Лучшие из них – металл, керамика и древесина. А вот пластиковые и стеклянные модели не самые лучшие, так как считается, что деньги утекают из них, будто вода;

– форма: необходимо выбирать копилки круглой формы без острых углов, в них быстрее копятся денежные средства;

– размер: по древнему учению фен-шуй необходимо сделать свой выбор, это своего рода послание вселенной о том, чтобы привлечь богатство и благополучие.

Можно купить копилку для денег в магазине розничной сети или онлайн в интернет магазине. Второй вариант позволяет ознакомиться с описанием и фото разных моделей, не выходя из дома и купить вещь по выгодным ценам.

Так же можно не покупать копилку для денег в магазине, а самим сделать ее. В Интернете можно найти подробную информацию, как самостоятельно сделать копилку, например из коробки из под обуви или пластиковой бутылочки. Собственная копилка создается с душой и фантазией, она способна оригинально украсить интерьер, но не все имеют желание и время для ее изготовления.

На рисунках 6 и 7 показаны схемы «Умных» копилки с автономным питанием и от USB.

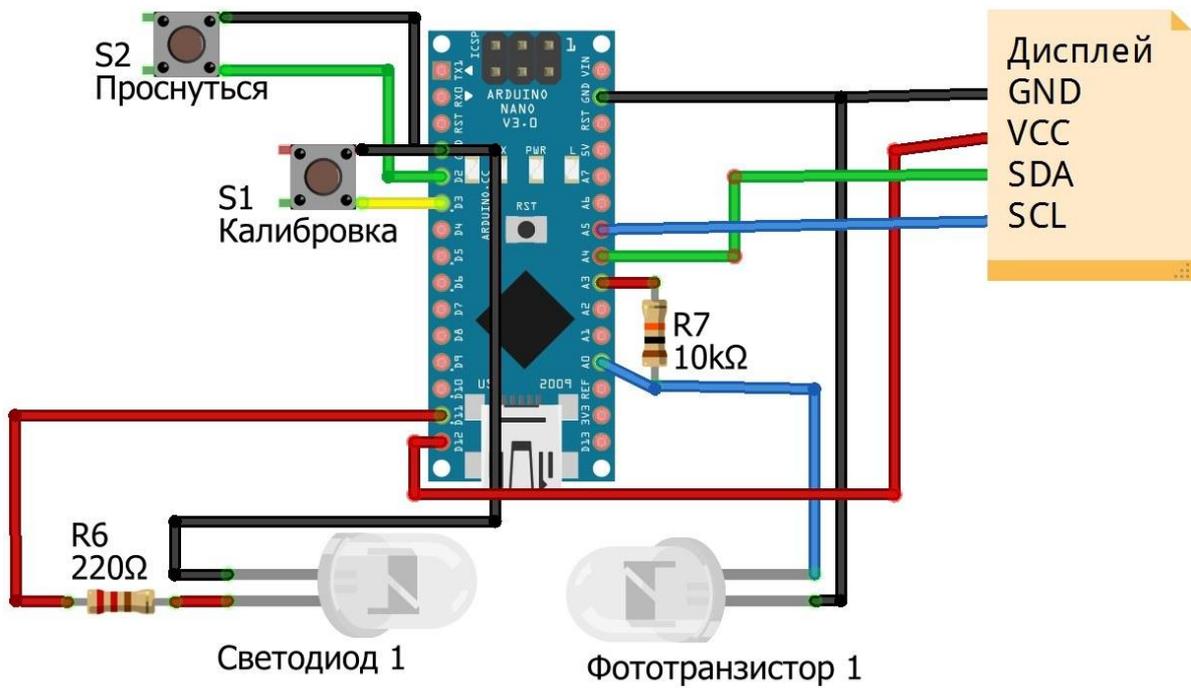


Рисунок 6 – Схема «Умной копилки» с питанием от USB

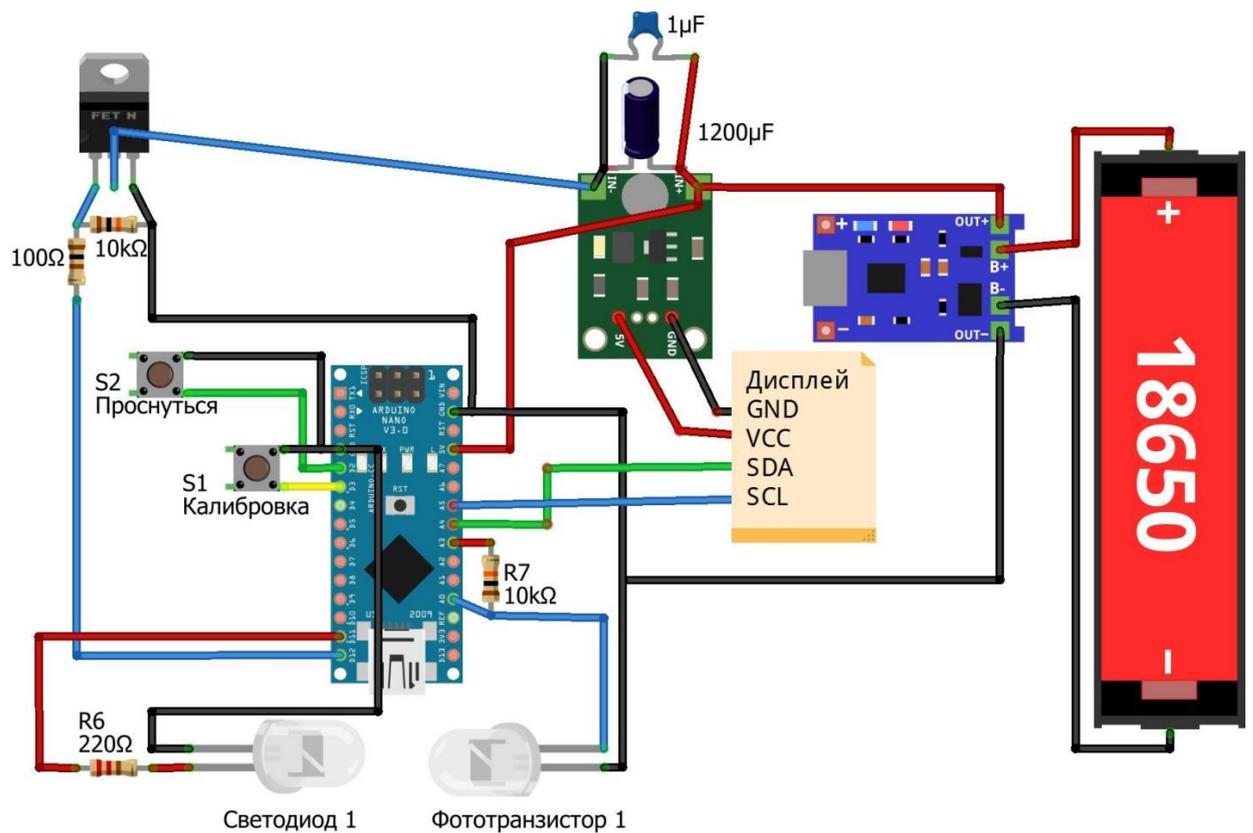


Рисунок 7 – Схема «Умной копилки» с автономным питанием

Элементы:

– кнопки: S2 (Отвечает за калибровку заново, если есть такая необходимость); S1 (Отвечает за то, чтобы копилка вышла из режима сна, и после этого можно бросать монеты);

– VT1 (Отвечает за прием ИК-сигнала с VD1 и VD2);

– дисплей (Отвечает за вывод суммы и количества монет);

– повышающий преобразователь (отвечает повышение питания дисплея, так как надо 5 В, а аккумулятор выдает 3.7);

– модуль USB (Отвечает за зарядку аккумулятора);

Виды автоматизированных копилок для денег:

Электронный распознаватель монет (по размеру) для копилки со счетчиком суммы и статистикой по каждому типу монет (рис. 8).

Возможности:

– распознавание размера с высокой точностью и его привязка к стоимости каждой монеты;

– вычисление общей суммы монет в копилке;

– статистика по числу монет каждого типа;

– все настройки сохраняются в энергозависимую память и не сбрасываются при питании;

– накопленная сумма тоже хранится в энергозависимой памяти и не боится сбоев питания;

– режим глубокого энергосбережения: в спящем режиме потребляется 0,07 мА, в схеме без преобразователя 0,02 мА;

– поддержка любого числа монет разного размера;

– автоматическая калибровка типов монет;

– сброс накопленного количества.



Рисунок 8 – Автоматизированная копилка со счетчиком монет [10]

Автоматизированная копилка панда-воришка (рисунок 9):

- для работы копилки, необходимо на белую тарелочку положить монетку, а панда лапкой заберет ее;
- механизм переносит монетку в отдельный отсек для монет, где хранится остальная сумма;
- деньги можно легко достать из копилки, нужно всего лишь открыть крышку на дне устройства;
- подходит для всех типов стандартных монет;
- благодаря своему дизайну и взрослым и детям будет весело и интересно использовать эту копилку.



Рисунок 9 – Автоматизированная копилка панда-воришка [14]

Автоматизированная копилка с лицом (рисунок 10):

- необходимо положить монету в рот лицу-копилки и она «съест» ее;
- при приближении руки к нему будет активироваться под ртом встроенный датчик движения;
- «съеденная» монета попадает внутрь копилки в специальный отсек для монет где хранится вся сумма денег;
- подходит для всех типов стандартных монет;
- благодаря своему дизайну людям разного поколения будет интересно, увлекательно и занимательно использовать эту копилку.

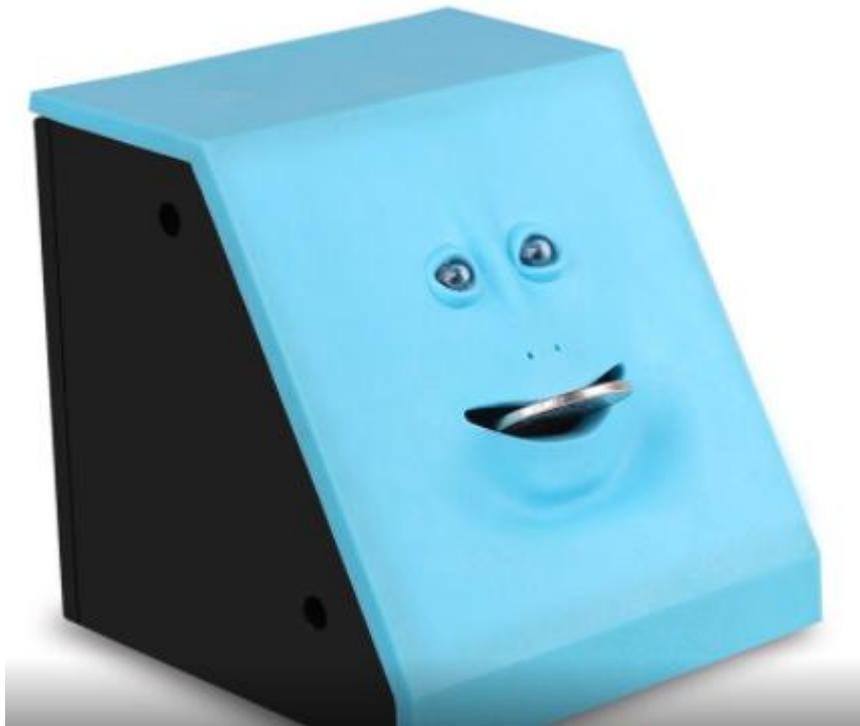


Рисунок 10 – Автоматизированная копилка с лицом [14]

Анализ рынка умных копилок позволил сформулировать требования, предъявляемые к проектируемому устройству.

1.3 Формирование требований к проектируемой системе

Следующие функциональные подсистемы предполагается выделить в системе:

- блок распознавания размера монет, вычисления общей суммы монет в копилке, статистики по числу монет каждого типа;
- блок, где настройки сохраняются в энергозависимую память и не сбрасываются при питании;
- подсистема формирования и визуализации отчетности.

Под аварийным состоянием подразумевается аварийное окончание процесса и «зависание» процесса.

В работе системы могут быть установлены следующие аварийные ситуации, действующие на системную надежность:

- сбой электроснабжения;

– «умная копилка» не выявляет ошибок, не установленных при тестировании и проверке системы.

Надёжность устройства соответствует требованиям:

– необходимо использовать средства с более высоким уровнем надёжности;

– применять технические средства, соответствующие задачам класса;

– в случае сбоя данные обязательно должны быть восстановлены.

Требования к внешнему оформлению:

– должен использоваться язык: русский;

– должен быть использован шрифт: Times New Roman;

– должен использоваться 12 размер шрифта;

– должна быть мерная палитра цветов;

– в цветовой палитре корпуса должен быть использован цвет: голубой;

Критерии, которые должны сохранить системную информацию: потеря питания, отказ технических средств, аварии, и т. п.

Требования к защите от действия внешних воздействий:

– крепкий корпус изделия (Пластмасса вида: HDPE или PE HD);

– водонепроницаемое покрытие;

– аккумуляторные батарейки (Щелочные батарейки DURACELL LR6.

Рабочее напряжение 1,5 В).

Требования безопасности:

– перед работой должна быть проверена работа аккумуляторных батарей, и всех составных частей УК;

– соблюдать меры предосторожности;

– работать с изделием чистыми руками.

Требования к информационному обеспечению:

– к способам организации данных в системе, к структуре и составу;

– к обмену между компонентами системы информацией;

– к совместимости информацией со смежными системами;

- к структуре представления данных, передачи данных в системе, обработки и процесса сбора;

- защита данных от повреждения при аварийных и неисправностях электропитания системы;

- к контролю, восстановлению, хранению и обновлению данных.

Структурное хранение данных в «Умной копилке» включает следующие области:

- область данных временного хранения;

- область данных постоянного хранения;

- область витрин данных.

Требования к защите данных от разрушений при авариях и сбоях в электропитании системы:

- информация в базе системной информации должна сохраняться при возникновении аварийных ситуаций, связанных с отключением электроснабжения.

Требования к контролю данных:

- система должна проводить протоколы всех мероприятий, связанных с изменением ее информационной структуры.

- для хранения данных предъявляются следующие условия:

- данные не должны стираться при выключении питания.

- к обновлению данных предъявляются следующие требования:

- данные должны быть обновлены в процессе работы с «Умной копилкой».

Язык для чтения информации по количеству монет: русский.

Требования к метрологическому обеспечению не предъявляются.

2 КОНСТРУКТОРСКИЙ РАЗДЕЛ

2.1 Выбор инструментального обеспечения проектирования

Рассмотрим несколько программ, которые помогут в разработке проекта «Умная копилка», выберем подходящую.

Fritzing – это бесплатная программа для проектирования и макетирования схем.

Разработано программное обеспечение «Fritzing» для создания электронных устройств, начиная с прототипа в виде макета, заканчивая конечным продуктом в виде печатной платы. Устройства создаются из готовых элементов, список которых можно посмотреть в правом верхнем углу программы, включая плату Arduino, различные аналоговые и цифровые микросхемы, аккумуляторы, транзисторы, макеты, светодиоды, кнопки, резисторы, конденсаторы, и даже моторы. Для размещения их в диаграмме нужно просто выбрать из списка, а затем перетащить их в область рабочего стола, используя левую кнопку мышки. Нарисовать схему можно как в макетном, так и в схематическом режиме. Среда проектирования Fritzing представлена на рисунке 11.

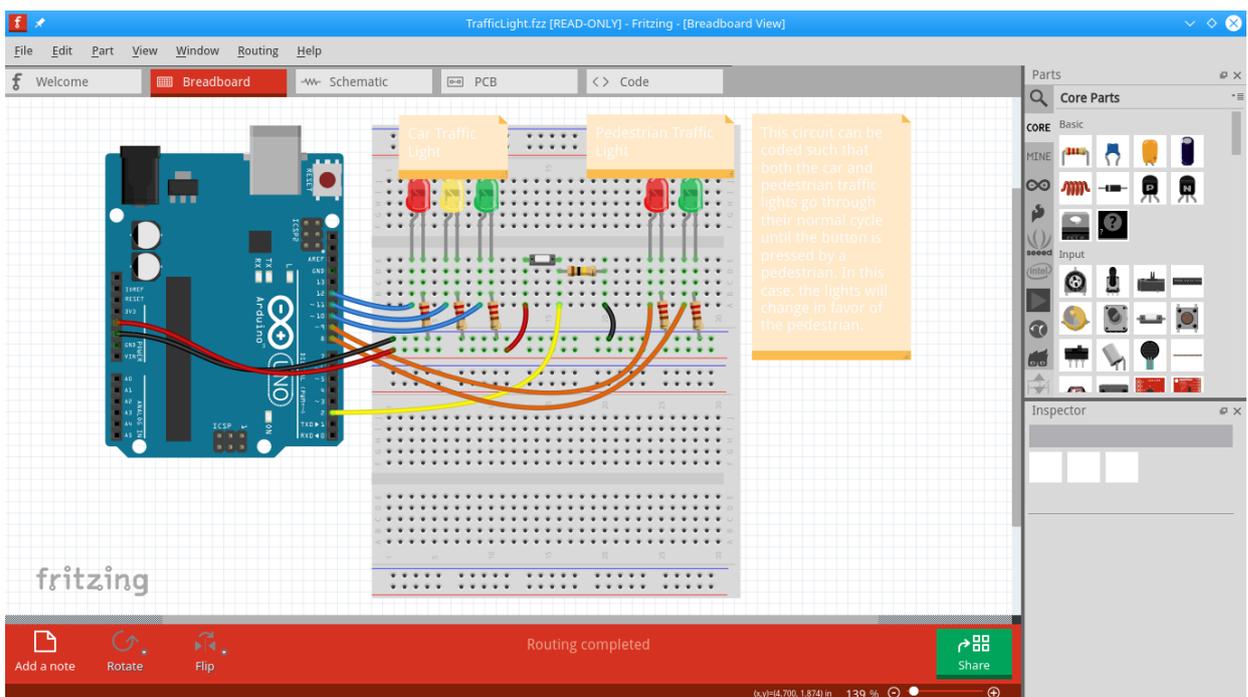


Рисунок 11 – Среда проектирования Fritzing

EasyEDA - это кроссплатформенная веб-среда автоматизации для проектирования электроники, которая включает редактор схем, редактор компоновки печатных плат, симулятор SPICE, облачное хранилище данных, систему управления проектами и инструменты для заказа изготовления печатных плат.

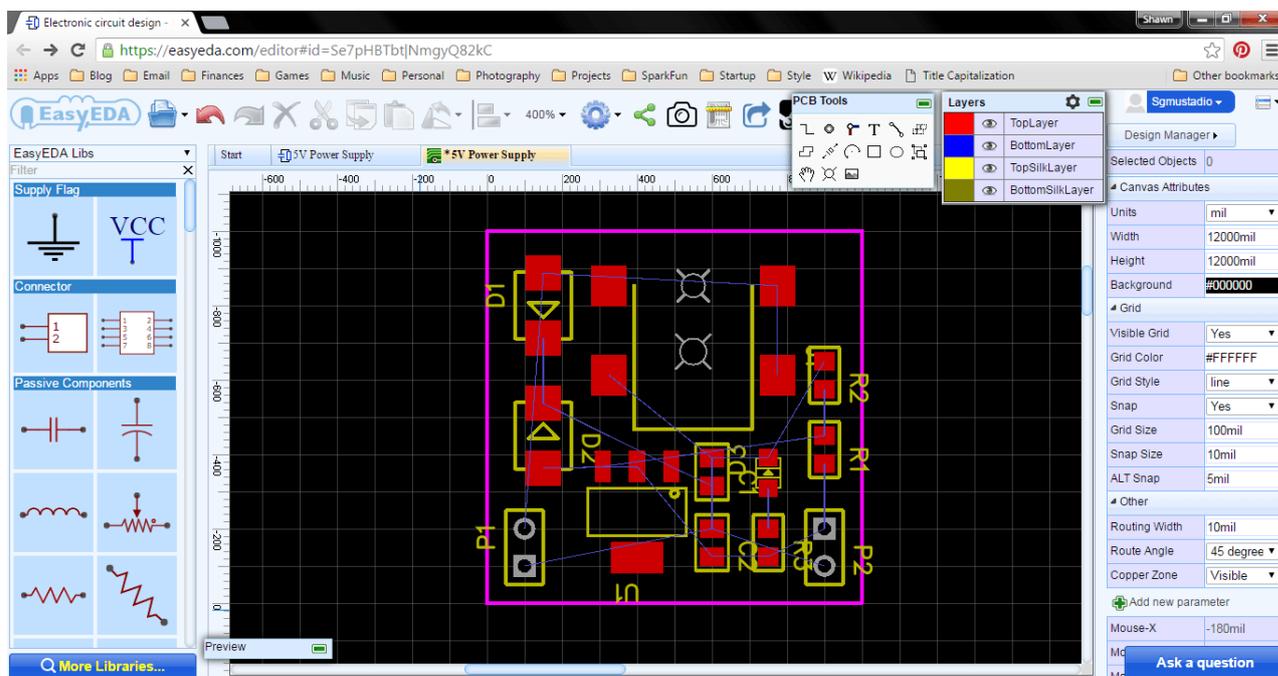


Рисунок 12 – Среда проектирования EasyEDA

В качестве программы для проектирования будет использоваться EasyEDA, так как у нее более удобный интерфейс и больший функционал.

Для программирования МК «Умной копилки», рассмотрим несколько программ и выберем подходящую.

Arduino IDE — является программной средой разработки, которая использует C++ и предназначена для программирования всех плат ряда Ардуино (Arduino).

Аббревиатура IDE имеет расшифровку - Integrated Development Environment, в переводе – интегрированная среда разработки. Благодаря этой среде программисты пишут программы быстрее..

Arduino IDE позволяет составлять программы в удобном текстовом редакторе, компилировать их в машинный код, и загружать на все версии

платы Arduino. Это приложение можно скачать на официальном сайте Arduino абсолютно бесплатно.

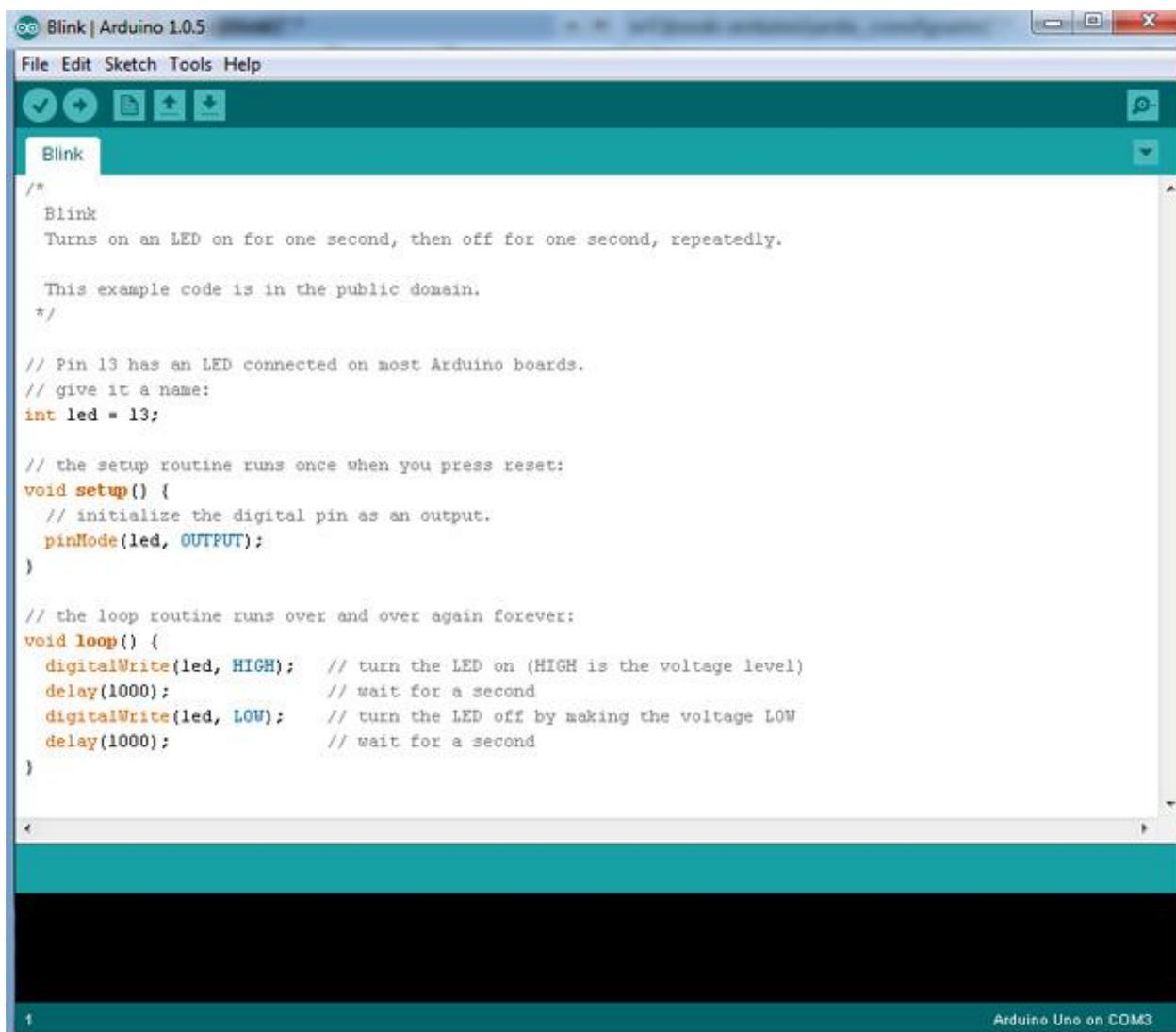


Рисунок 13 – Среда разработки IDE

Programino – это низкобюджетная IDE для Arduino, Genuino или подобных плат с автозаполнением кода, подсветкой пользовательского синтаксиса, аналоговым плоттером, аппаратным просмотром, извлечением HEX-файла и редактором HTML5.

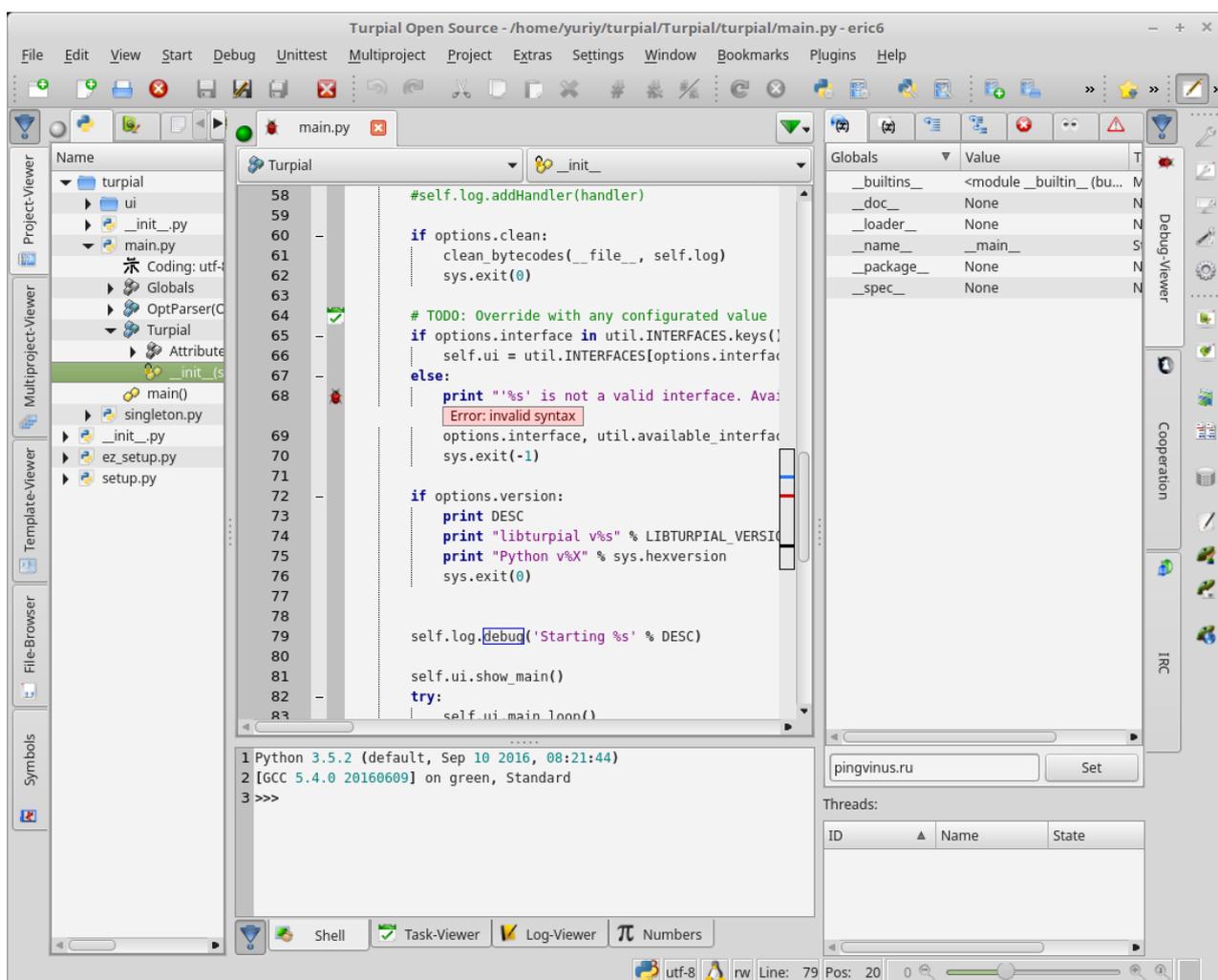


Рисунок 14 – Среда разработки Programino

В качестве среды для программирования будет использоваться Arduino IDE, так как для проекта это программа имеет нужный функционал и доступность.

2.2 Разработка структурной и функциональной схем устройства

Структурная схема — это совокупность элементарных звеньев объекта и связей между ними, один из видов графической модели. Под элементарным звеном подразумевается часть объекта, системы управления и т. д., которая реализует элементарную функцию.

Структурная схема разрабатывается на начальных стадиях проектирования и предшествует разработке схем других типов. Структурная схема определяет основные функциональные части изделия, их назначение и

взаимосвязи между ними. Схема отображает принцип действия изделия в самом общем виде.

Перед разработкой принципиальной схемы устройства необходимо выполнить разработку функциональной схемы.

На рисунке 15 представлена функциональная схема устройства. Сплошной линией изображена подача питания, штриховой линией изображена подача информации.

Элементы, находящиеся в блоках:

- БСМ (блок счетные механизмы): Инфракрасный датчик препятствий, дисплей, преобразователь постоянного тока (повышающий источник питания)
- БУ: Микроконтроллер
- БП: Аккумулятор, зарядная плата

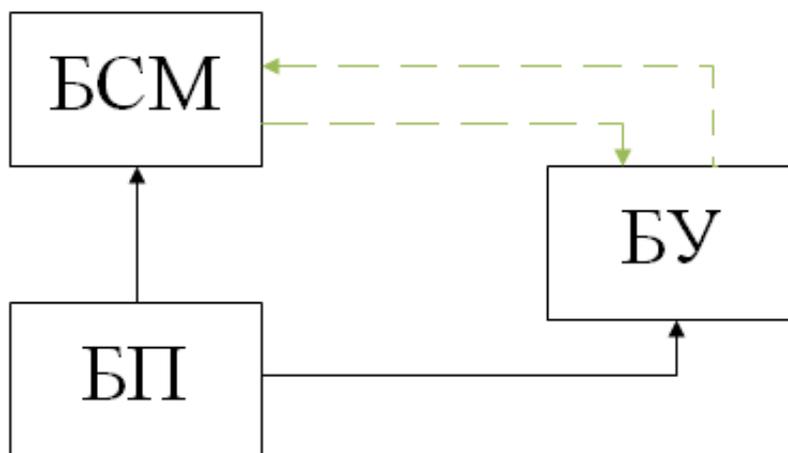


Рисунок 15 – Функциональная схема устройства

Данная функциональная схема, предназначена для того, чтобы дать полное представление о структуре разрабатываемого устройства «Умная копилка».

Принципиальная схема - это схема, на которой каждая деталь обозначена условным графическим обозначением (УГО). Схематические изображения позволяют понять, как работает устройство и как его части

соединяются друг с другом. Кроме того, принципиальная электрическая схема - это первая задача дизайнера. По схеме и перечню элементов разрабатывает дизайн печатной платы и всего изделия.

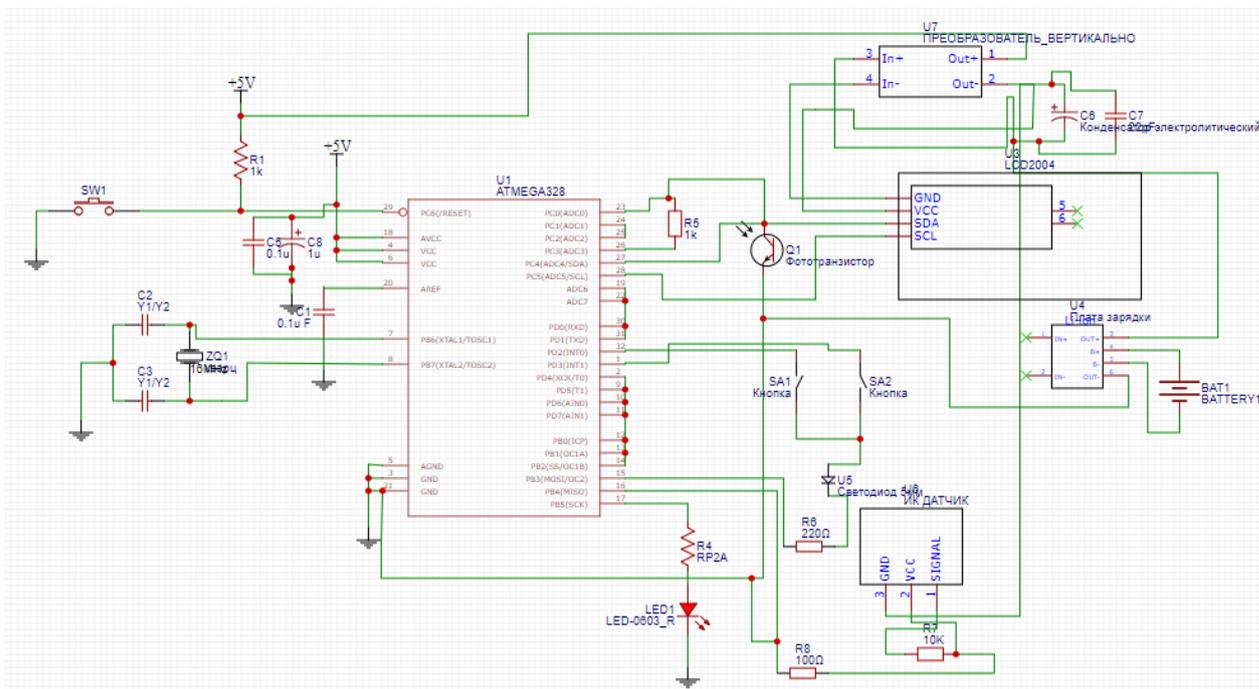


Рисунок 16 - Принципиальная схема устройства

Элементная база проектируемого устройства представлена в следующем параграфе.

2.3 Выбор элементной базы

Перед разработкой устройства необходимо выполнить выбор элементной базы устройства.

В таблице 1 представлены сравнительные характеристики инфракрасного датчика двух моделей.

Таблица 1 – Выбор инфракрасного датчика препятствия

Характеристики	Инфракрасный датчик препятствия (YL-63)	Инфракрасный датчик препятствия (KY-032)
Дальность срабатывания	От 2 до 40	От 2 до 30
Ток	40 мА	20 мА
Напряжение	5 В	3,3-5 В
Угол обзора	35°	35°
Рабочая температура	-40° до +85°С	-10° до +50°С
Цена, руб.	26,71	208

Выбор был сделан в пользу инфракрасного датчика препятствий модели (YL-63), так как у него более высокая дальность срабатывания, ток потребления и рабочая температура. Цена датчика данной модели намного ниже, чем этой модели KY-032.

В таблице 2 представлены сравнительные характеристики двух моделей дисплея.

Таблица 2 – Выбор дисплея

Характеристики	Дисплей LCD 10602 12С	Дисплей LCD 10602А
Количество символов в строке	16	16
Напряжение питания	5 В	4,7-5,3 В
Диапазон рабочих температур	От -20° до +70°С	От 0° до +60°С
Размеры модуля	80 x 35 x 11	80x36x11
Цена, руб.	230	299

Выбор был сделан в пользу дисплея LCD 10602 12С, так как у него лучше диапазон рабочих температур, меньше размер модуля. Дешевле чем дисплей LCD 10602А.

В таблице 3 представлены сравнительные характеристики двух моделей преобразователя постоянного тока.

Таблица 3 – Выбор преобразователя постоянного тока

Характеристики	Повышающий DC-DC преобразователь на базе E50D	Повышающий преобразователь DC-DC MT3608
Вес	20 г	12 г
Габариты	18 × 33 × 9 мм	43 × 19 × 12 мм
Входное напряжение	от 5 В	от 32 В
Ток	600 мА	4 А
Частота преобразования	80 кГц	400 кГц
Цена, руб.	99	219

Выбор был сделан в пользу повышающего DC-DC преобразователя на базе E50D, так как у него меньше габариты и цена.

В таблице 4 представлены сравнительные характеристики двух моделей микроконтроллеров ATmega.

Таблица 4 – Выбор микроконтроллера ATmega

Характеристики	Микроконтроллер ATmega328	Микроконтроллер ATmega128-16AU
Тактовая частота	0-20 мгц	16 мгц
Напряжение питания	1,8-5,5 В	4,5-5,5 В
Объем EEPROM	1к x 8	4к x 8
Ширина шины данных	8-бит	8-бит
Объем RAM	2к x 8	4к x 8
Цена, руб.	220	530

Выбор был сделан в пользу микроконтроллера ATmega 328, так как у него выше тактовая частота, намного ниже цена.

В таблице 5 представлены сравнительные характеристики моделей двух аккумуляторов.

Таблица 5 – Выбор аккумулятора

Характеристики	FinePower 18650	ROBITON SAM2600
Форм-фактор батареи	18650	18650
Номинальная емкость	2600 мА*ч	2600 мА*ч
Емкость (энергия) измеренная	8939 мВт*ч	8926 мВт*ч
Напряжение	4,15. В	4,18 В
Внутреннее сопротивление	49 мОм	48 мОм
Цена, руб.	350	550

Выбор был сделан в пользу аккумулятора FinePower 18650 , так как у него выше емкость (энергия) измерения, выше внутренне сопротивление. Цена у этого аккумулятора ниже, чем у аккумулятора ROBITON SAM2600.

В таблице 6 представлены сравнительные характеристики двух моделей зарядной платы.

Таблица 6 – Выбор зарядной платы

Характеристики	Модуль заряда литиевой Батареи 1А TP40561A	Модуль RP038. Контроллер заряда и разряда для Литий-Ионных (Li-Po, Li-Ion) аккумуляторов. EM4056A
Напряжение питания Uвх.	4,5-5,5 В	4,5-8,0 В
Ток заряда	1 А	1 А
Размер	23*17	17*28
Разъем	MicroUSB	MicroUSB
Цена, руб.	85	200

Выбор был сделан в пользу модуля заряда литиевой батареи 1А TP40561A, так как у него меньше размер и цена.

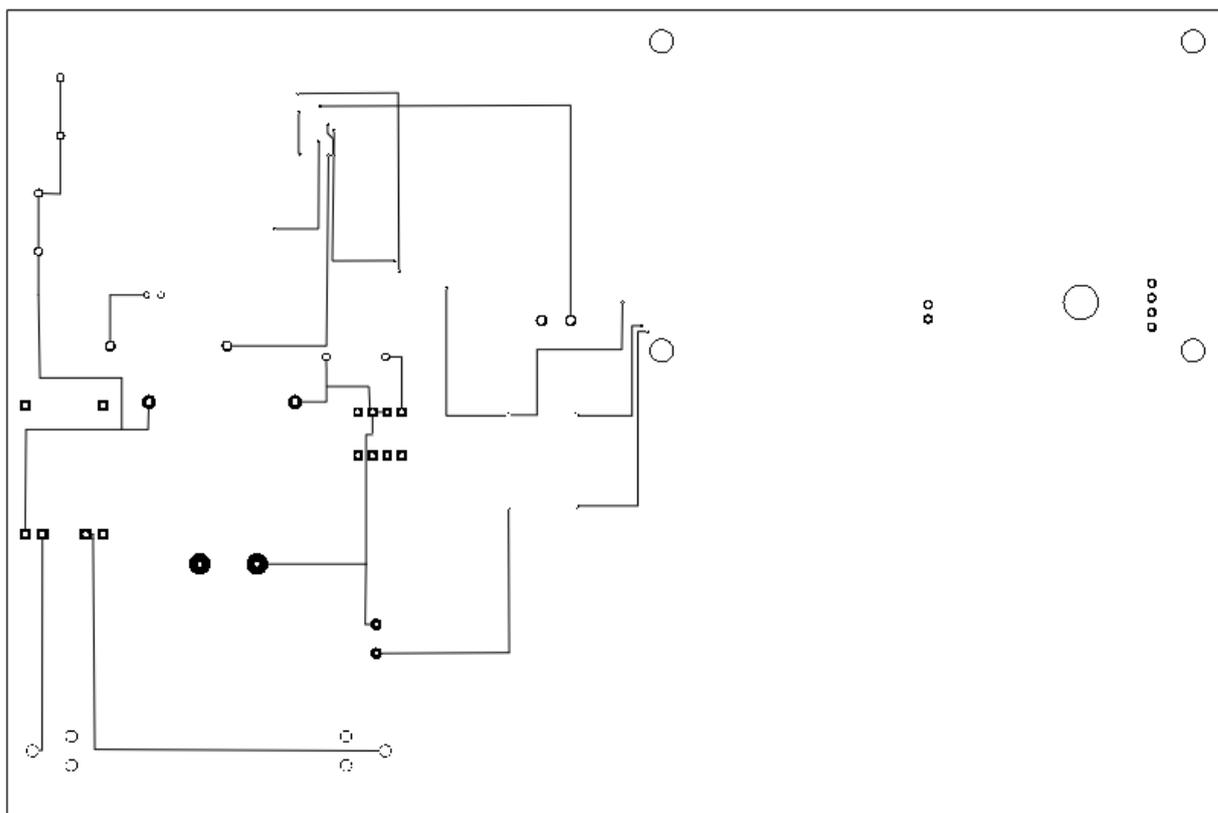


Рисунок 18 – Печатная плата устройства задняя сторона

2.5 Разработка алгоритмов работы управляющей программы устройств. Разработка программного обеспечения устройства

Для обеспечения работы проектируемого устройства с заданными техническими требованиями необходимо запрограммировать наш микроконтроллер на определенную обработку данных и выдачу определенных сигналов

Управляющая программа разработана в ArduinoIDE.

Код программы находится в приложении Б.

Блок-схема – наглядный способ представления алгоритма. И чтобы наглядно посмотреть на алгоритм, разработана блок-схема алгоритма (приложение В).

3 ОХРАНА ТРУДА И ПРОМЫШЛЕННАЯ ЭКОЛОГИЯ

3.1 Мероприятия по снижению воздействия выявленных вредных и опасных производственных факторов

Существует большое разнообразие мероприятий по снижению воздействия выявленных вредных и опасных производственных факторов. Впрочем, методы охраны труда могут классифицироваться по определённым принципам и одна и та же методика может защищать трудящихся одновременно от каких-либо вредных, так и опасных факторов в производственной среде и трудовом процессе.

В качестве критерия можно принять принцип защиты. По данному критерию методы защиты работников представлены следующими образцами.

- нормализация трудовых условий, суть данного метода заключается в осуществлении организационных, технологических и других мероприятий по уменьшению уровня рисков для здоровья и приведению значений вредного и опасного производственного фактора к нормированным значениям.

- улучшение технологического процесса, направленного на снижение вредных выбросов, шумов, вибраций и др.;

- модернизация или замена оборудования, которая не соответствует современным стандартам безопасности и санитарно-гигиенических норм;

- обеспечение помещений, оборудования и рабочих мест необходимыми средствами коллективной защиты (вентиляцией, приборами освещения, ограждениями и др.);

- проведение ремонтно-профилактических работ по средствам коллективной безопасности, которые находятся в организации, но частично или полностью не выполняют своих функций защиты.

Защита расстоянием. Этот метод защиты предполагает устранение зон пересечения пространства гомосферы, где действует человек, и пространства ноксосферы, где может быть установлено опасное и вредное производственное воздействие. Это достигается посредством:

– ограждение опасных зон, чтобы создать физическую преграду, препятствующую приближению человека к причине опасности, исключая возможность захвата одежды или части тела движением элементов техники, ожога от нагретой поверхности и так далее;

– удаление операторов из опасной зоны с помощью автоматизации оборудования, применяя дистанционное управление, роботов и манипуляторов;

– нормирование минимального допустимого расстояния оператора от источника повышенного риска опасности и т.д.

Защита временем. Этот метод применяется в случаях невозможности применения первых двух методов из-за технических причин или их реализация не даёт удовлетворительного эффекта. В этом случае установлено нормативное время пребывания человека в зоне высокой опасности и вредности (например, при воздействии ионизирующих лучей, близких к мощным источникам электромагнитных лучей и т.д.). Работник может установить: сокращенную рабочую неделю или уменьшенную рабочую смену, наибольшую длительность непрерывного труда в условиях воздействия вредного производственного фактора, а также время дополнительного перерыва в рабочую смену.

Методы защиты различаются и в местах их использования.

В наиболее радикальных случаях признаны способы противодействия вредным и опасным производственным факторам в источниках его образования и методы устранения прямой связи с ними сотрудника. Эти методы реализуются по самым разным способам, которые имеют свою особенность, зависящую от характера риска или вреда.

3.2 Мероприятия по оптимизации освещенности на рабочем месте

Освещение оказывает большое влияние на организм и работоспособность человека. Чтобы увеличить производительность труда в

15% и сократить число несчастных случаев необходимо правильное освещение.

Из-за неправильного освещения могут возникнуть такие болезни, как зрительное утомление, аккомодация, спазм, близорукость и др., снижает физическую и умственную производительность, увеличивает число несчастных случаев и аварий, ошибок в производственных процессах.

Рациональным называется освещение, которое соответствует санитарно-гигиеническим и техническим нормам.

В помещениях применяется освещение: искусственное и естественное. Естественное освещение подразумевает солнечный свет, который попадает внутрь здания через окна.

Естественное освещение изменяется в зависимости от суток и времени года, а также от явлений атмосферы. Оказывают влияние на освещение месторасположение и конструкция зданий, расстояние между ними, расположение и форма окон, величина застекленной поверхности и др.

Для зданий и помещений нормы естественного освещения определяются в зависимости от их назначения.

Естественный свет является самым благоприятным для человека, но оно не может полностью обеспечить необходимую освещенность в помещениях. Поэтому, искусственное освещение широко применяется в практических целях.

Наиболее распространенным видом искусственного освещения является электрическое освещение. У него также, как и у естественного, есть свои нормы для различных помещений.

Освещенность помещения определяется с помощью люксметра. Он представляет собой миллиамперметр и селеновый элемент. Когда свет попадает на селеновый фотоэлемент появляется фототок, который в миллиамперметре оказывает влияние на стрелку прибора, которая показывает освещенность рабочей поверхности по шкале прибора, проградуированной в люксах. Если отсутствует люксметр для определения

освещенности, на практике следует руководствоваться нормами электрического освещения, выраженными в ваттах на 1 м² площади.

Люминесцентные и лампы накаливания используются для искусственных электрических светильников. Люминесцентная лампа является имитацией естественного света и обеспечивает высокий уровень качества. Они экономичны по сроку службы, световой отдаче и расходу электроэнергии.

Электрические лампы для осветления помещений помещаются в специальную оболочку раз личного типа. Арматура осуществляет передачу светопотока, получаемого от электрической лампы, с минимальными потерями и защищает от ослепительной яркости глаза работников, а во многих случаях меняет спектральный состав светоисточника. Светильником принято называть арматуру и лампу.

Исходя из характера распределения светового потока, светильники делятся на 3 группы: рассеянный, отраженный и прямой свет.

Коэффициент полезного действия светильников с лампами накаливания может достигать 80-85%.

В последние годы для освещения помещений были широко распространены осветительные приборы встроенного вида. Они помогают созданию равномерной освещенности помещений, оказывают хорошее воздействие на работу человека.

В процессе обслуживания осветительных приборов необходимо следить за постоянным напряжением и устранять причины потери напряжения или колебания. Контроль измерения освещенности следует проводить не реже 1 раза в 3 месяца.

Освещение и управление осветительными системами осуществляется на предприятиях ведомственными органами надзора.

Средний уровень освещенности на рабочем месте, где постоянно находятся люди, должен быть не менее 200 лк.

3.3 Утилизация использованного оборудования при разработке «Умной копилки»

Современное, электронное оборудование, даже бытовые приборы, нередко содержит драгметаллы, токсичные и прочие опасные для экологии и здоровья человека вещества. Фактически, порядок утилизации (избавления от) компьютеров и оргтехники производится с учетом двух федеральных законов:

- 89-ФЗ от 24.06.1998 – «Об отходах производства и потребления»;
- 41-ФЗ от 26.03.1998 – «О драгоценных металлах и драгоценных камнях».

Справочник наилучших российских доступных технологий ИТС 15-2016 предусматривает 2 технологии утилизации и обезвреживания ОЭЭО:

- разборка с извлечением компонентов с ресурсной ценностью и передача организациям на вторичную переработку;
- измельчение не разобранных устройств с дальнейшей сортировкой лома и получением полиметаллического концентрата.

Сбор, перевозку и разборку электронного мусора в России проводят:

- производители электронных устройств;
- заводы по переработке ЭО;
- специальные пункты сбора и хранения ЭО;
- пункты сбора вторсырья.

Утилизация использованного оборудования при разработке устройства, в простой мусорный бак категорически запрещена, потому что наносится вред экологии, а также окружающей природе.

В некоторых городах России в целях утилизации бытовой электроники и ее запчастей организованы специальные приемные пункты. Туда можно принести старую технику и получать награду за правильную утилизацию. Когда в рабочем пункте будет достаточно оборудования, сотрудники разберут его на части и отправят его на вторичную переработку.

К сожалению, не везде есть пункты приемки техники. Если в городе нет такого, можно сдать не нужные приборы в пункт приема металлолома.

По законодательству Российской Федерации происходит нарушение Федерального закона об отходах производства и потребления, выбрасывая на свалку электронную технику.

Компоненты, которые подлежат утилизации после сборки устройства: монтажный провод, ненужные электронные части.

3.4 Вывод по разделу

При разработке устройств определяется много опасных производственных факторов, которые наносят вред разработчику. Существует большое количество мероприятий по снижению воздействия выявленных вредных и опасных производственных факторов. Они позволяют максимально обезопасить работу, например, осуществляя организационные, технологические и другие мероприятия по уменьшению уровня рисков для здоровья и проведения значений вредного и опасного производственного фактора к нормированным значениям.

Также опасным для человека фактором можно считать плохую освещенность. Правильное освещение рабочего места является одним из основных составляющих успешной работы и хорошего самочувствия. Если не придерживаться данного положения, могут возникнуть проблемы со зрением и значительно ухудшится трудовая деятельность.

Чтобы избежать нарушений связанных со зрением, необходимо придерживаться рассмотренных выше пунктов, они позволят в полной мере увеличить работоспособность и не нанести вред здоровью.

Утилизация использованного оборудования при разработке устройства, в простой мусорный бак категорически запрещена, потому что наносится вред экологии, а также окружающей природе. Существуют специальные приемные пункты, где работники разбирают устройства на части и отправляют его на вторичную переработку. На сегодняшний день утилизация

электроники осуществляется по закону, который был установлен государственными органами для предотвращения загрязнения окружающей среды и регулировки утилизации электротехники.

4 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКАЯ ЧАСТЬ

4.1 Расчет затрат на разработку программы для микроконтроллера

Затраты на разработку программы для микроконтроллера рассчитываются по следующей формуле:

$$Z_{РПР} = Z_{ФОТР} + Z_{ОВФ} + Z_{ЭВМ} + Z_{СПП} + Z_{ХОН} + P_H, \quad (1)$$

где $Z_{ФОТР}$ – общий фонд оплаты труда разработчиков программы,

$Z_{ОВФ}$ – начисления на заработную плату разработчиков программы во внебюджетные фонды,

$Z_{ЭВМ}$ – затраты, связанные с эксплуатацией техники,

$Z_{СПП}$ – затраты на специальные программные продукты, необходимые для разработки программы для микроконтроллера,

$Z_{ХОН}$ – затраты на хозяйственно-операционные нужды (бумага, литература, носители информации и т.п.),

P_H – накладные расходы ($P_H = 30\%$ от $Z_{ФОТР}$).

При разработке программы для микроконтроллера общее время разработки составило 1 месяца.

Фонд оплаты труда за время работы над программой для микроконтроллера:

$$Z_{ФОТР} = \sum_{j=1}^m O_{Pj} \cdot T_{РПРj} \cdot (1 + k_D)(1 + k_Y), \quad (2)$$

где O_{Pj} – оклад j -го разработчика. В разработке участвовал 1 человек, его оклад составляет 20000 руб.,

$T_{РПРj}$ – общее время работы над программой в месяцах, $T_{РПР} = 1$,

k_D – коэффициент дополнительной зарплаты, $k_D = 20\% = 0,2$,

k_Y – районный коэффициент, $k_Y = 1,15$.

Таким образом,

$$Z_{ФОТР} = 20000 \cdot 1 \cdot (1 + 0,2) \cdot (1 + 1,15) = 51\,600 \text{ руб.}$$

Страховой взнос во внебюджетные фонды складываются из взносов на обязательное социальное страхование от несчастных случаев на производстве и профессиональных заболеваний.

Страховые взносы складываются из обязательного пенсионного страхования (ОПС), отчислений в фонд социального страхования и отчислений в фонд обязательного медицинского страхования.

Значения всех используемых ставок приведены в таблице 7.

Таблица 7 – Значения ставок страховых взносов

№	Наименование внебюджетного фонда	Размер ставок, %
1	Пенсионный фонд	22
2	Фонд социального страхования	2,9
3	Федеральный фонд обязательного медицинского страхования	5,1
	Итого:	30

Сумма начислений на заработную плату во внебюджетные фонды составляет:

$$Z_{\text{ОВФ}} = 0,3 \cdot Z_{\text{ФОТР}}, \quad (3)$$

$$Z_{\text{ОВФ}} = 0,3 \cdot 51\,600 = 15480 \text{ руб.}$$

Затраты, связанные с использованием вычислительной и оргтехники:

$$Z_{\text{ЭВМ}} = T_{\text{МРПР}} \cdot k_{\Gamma} \cdot n \cdot C_{\text{М-ч}} \quad (4)$$

где k_{Γ} – коэффициент готовности ЭВМ, $k_{\Gamma} = 0,95$,

n – количество единиц техники, равно 1,

$C_{\text{М-ч}}$ – себестоимость машино–часа, $C_{\text{М-ч}} = 10$ руб.,

$T_{\text{МРПР}}$ – машинное время работы над программой, равно 1 мес.

Перевод рабочего времени в часы осуществляется по формуле:

$$T_{\text{час}} = T_{\text{мес}} \cdot \text{Ч}_{\text{РД}} \cdot T_{\text{см}} \cdot K_{\text{см}}, \quad (5)$$

где $T_{\text{час}}$ – рабочее время, ч,

$T_{\text{мес}}$ – рабочее время, мес, ($T_{\text{мес}} = 2$),

$Ч_{\text{рд}}$ – число рабочих дней, ($Ч_{\text{рд}} = 22$),

$T_{\text{см}}$ – продолжительность рабочей смены, ($T_{\text{см}} = 8$ ч),

$K_{\text{см}}$ – количество рабочих смен, ($K_{\text{см}} = 1$).

Таким образом, время на разработку программы для микроконтроллера с использованием ЭВМ составляет:

$$T_{\text{час}} = 2 \cdot 22 \cdot 8 \cdot 1 = 352 \text{ часа,}$$

$$З_{\text{ЭВМ}} = 352 \cdot 0,95 \cdot 1 \cdot 10 = 3344 \text{ руб.}$$

Затраты на специальные программные продукты, необходимые для разработки программы для микроконтроллера рассчитываются по формуле:

$$З_{\text{СПП}} = \sum_{p=1}^n Ц_p \quad (6)$$

где $Ц_p$ – цена p -го специального программного продукта.

Перечень программных продуктов специального назначения приведен в таблице 8.

Таблица 8 – Программные продукты специального назначения

№	Название ПП	Цена, руб.
1	Среда разработки Arduino IDE	0
2	EasyEDA	0
	Итого:	0

Использованные программные продукты бесплатны, поэтому:

$$З_{\text{СПП}} = 0 \text{ руб.}$$

Затраты на хозяйственно–организационные нужды приведены в таблице 9 и вычисляются по формуле:

$$Z_{\text{ХОИ}} = \sum_{\tau=1}^n C_{\tau} \cdot K_{\tau}, \quad (7)$$

где C_{τ} – цена τ -го товара, руб.,

K_{τ} – количество τ -го товара.

Таблица 9 – Затраты на хозяйственно–организационные нужды

Наименование	Цена за единицу (руб.)	Кол-во (шт.)	Всего (руб.)
USB–флеш–накопитель	600	1	600
Бумага	0,5	140	70
Итого:			670

$$Z_{\text{ХОИ}} = 600 \cdot 1 + 0,5 \cdot 140 = 670 \text{ руб.}$$

Накладные расходы:

$$P_H = Z_{\text{ФОТР}} \cdot k_{\text{НР}}, \quad (8)$$

$$P_H = 51600 \cdot 0,3 = 15480 \text{ руб.}$$

Таким образом, затраты на разработку программы для микроконтроллера рассчитанные по формуле 1 составят:

$$Z_{\text{ППР}} = 51600 + 15480 + 3344 + 0 + 670 + 15480 = 86574 \text{ руб.}$$

4.2 Расчет затрат на внедрение программы для микроконтроллера

Затраты на внедрение программы для микроконтроллера ($Z_{\text{ВПР}}$) рассчитываются по формуле:

$$Z_{\text{ВПР}} = Z_M + Z_{\text{КТС}} \cdot (1 + k_{\text{ТУН}}) + Z_{\text{ПО}} + Z_{\text{ФОТВ}} + Z_{\text{ОФВ}} + Z_{\text{ЭВМ}} + P_{\text{КОМ}} + P_H, \quad (9)$$

где Z_M – затраты на приобретение материалов, руб.,

$Z_{\text{КТС}}$ – затраты на приобретение комплекса технических средств, руб.,

$Z_{\text{ПО}}$ – затраты на приобретение программного обеспечения (включают стоимость разработанной программы, а также других существующих ПП, необходимых для функционирования системы), руб.,

$Z_{\text{ФОТВ}}$ – затраты на оплату труда работников, занятых внедрением проекта, руб.,

$Z_{\text{ОФВ}}$ – страховой взнос во внебюджетные фонды с заработной платы работников, занятых внедрением проекта, руб.,

$Z_{\text{ЭВМ}}$ – затраты, связанные с эксплуатацией ЭВМ при внедрении проектного решения, руб.,

$R_{\text{ком}}$ – командировочные расходы, руб.,

$R_{\text{н}}$ – накладные расходы, руб.,

$kt_{\text{ун}}$ – коэффициент транспортирования, установки и наладки комплекса технических средств, определяется действующими нормативами организации, а также спецификой конкретного проекта.

Затраты на приобретение материалов (ЗМ) приведены в таблице 10.

Таблица 10 – Затраты на приобретение материалов

Наименование	Цена за единицу (руб.)	Кол-во (шт.)	Всего (руб.)
Инфракрасный датчик препятствия (YL-63)	26	1	26
Микроконтроллер	200	1	200
Дисплей LCD 10602 12С	230	1	230
Повышающий DC-DC преобразователь на базе E50D	99	1	99
Аккумулятор FinePower 1865	350	1	350
Модуль заряда литиевой Батарей 1А TP40561A	85	1	85

Продолжение таблицы 10

Резистор 10 кОм	5	1	5
Резистор 220 Ом	5	1	5
Светодиоды	1	1	1
Кнопки	10	2	20
Монтажный провод	100 (5м)	1	100
Заглушки для труб	50	2	100
Муфта	100	1	100
Кабель USB	250	1	250
		Итого:	1571

Дополнительного приобретения компьютеров или других КТС не требуется, следовательно, $Z_{\text{КТС}} = 0$.

Затраты на приобретение программного обеспечения в данном случае равны затратам на разработку и составляют $Z_{\text{ПО}} = 86574$ руб.

Внедрением занят один системный инженер с окладом 15000 руб. Время внедрения – 0,2 месяца. По формуле рассчитываем затраты на оплату труда и страховой взнос во внебюджетные фонды.

$$Z_{\text{ФОТВ}} = 15000 \cdot 0,2 = 3000 \text{ руб.}$$

$$Z_{\text{ОВФ}} = 3000 \cdot 0,3 = 900 \text{ руб.}$$

Затраты, связанные с эксплуатацией ЭВМ при внедрении проектного решения составят:

$$Z_{\text{ЭВМ}} = 0,2 \cdot 22 \cdot 8 \cdot 10 = 352 \text{ руб.}$$

Командировочные расходы при внедрении программы для микроконтроллера не планируются, следовательно, $R_{\text{ком}} = 0$.

Так как коэффициент накладных расходов по данным организации составляет $k_{\text{НР}} = 0,3$, то величина накладных расходов равна 352 руб.

Суммарные затраты на внедрение составят:

$$Z_{ВПР} = 1571 + 0 + 86574 + 3000 + 900 + 352 + 0 + 352 = 92749 \text{ руб.}$$

4.3 Расчет эксплуатационных текущих затрат по программе для микроконтроллера

Годовые затраты на обработку результатов до внедрения разработанной программы для микроконтроллера рассчитываются по формуле:

$$C_1 = ЗП_1 + ОТ_{ВН1} + З_{ЭВМ1} + M_{31} + НР_1, \quad (10)$$

где $ЗП_1$ – затраты на оплату труда сотрудника на выполнение функций до внедрения проектного решения,

$ОТ_{ВН1}$ – страховой взнос во внебюджетные фонды,

$З_{ЭВМ1}$ – затраты, связанные с эксплуатацией ЭВМ,

M_{31} – годовые материальные затраты на сопровождение программы для микроконтроллера составляют 1500 руб.,

$НР_1$ – накладные расходы.

Временные затраты работы сотрудника в месяцах рассчитываются по формуле:

$$T_{1мес} = \frac{T_{1час}}{Ч_{рд} \cdot Ч_{рч}}, \quad (11)$$

где $T_{1мес}$, $T_{1час}$ – время, затрачиваемое сотрудником на обработку результатов, в месяцах и часах соответственно ($T_{1час} = 564$ часов),

$Ч_{рд}$ – число рабочих дней в месяц,

$Ч_{рч}$ – число рабочих часов в день.

$$T_{1мес} = \frac{564}{22 \cdot 8} = 3,2 \text{ мес.}$$

Тогда затраты на оплату труда сотрудника составят:

$$ЗП_1 = O_c \cdot T_{1мес} \cdot (1 + k_d) \cdot (1 + k_y), \quad (12)$$

где O_c – оклад сотрудника (оклад составляет 15000 руб.),

$$3\Pi_1 = 15000 \cdot 3,2 \cdot (1 + 0,2) \cdot (1 + 1,15) = 123840 \text{ руб.}$$

Страховой взнос до внедрения вычисляют по формуле:

$$OT_{BH1} = 3\Pi_1 \cdot 0,3, \quad (13)$$

$$OT_{BH1} = 123840 \cdot 0,3 = 37152 \text{ руб.}$$

Рассчитываем затраты, связанные с эксплуатацией ЭВМ до внедрения по формуле:

$$3_{ЭВМ1} = T_{1\text{час}} \cdot C_{M-Ч}, \quad (14)$$

$$3_{ЭВМ1} = 564 \cdot 10 = 5640 \text{ руб.}$$

Подставив соответствующие значения в формулу 10, получим:

$$C_1 = 123840 + 37152 + 5640 + 1500 = 168132 \text{ руб.}$$

Годовые затраты на эксплуатацию системы после внедрения программы для микроконтроллера рассчитываются аналогично по формуле:

$$C_2 = 3\Pi_2 + OT_{BH2} + 3_{ЭВМ2} + M_{з2} + НР_2, \quad (15)$$

где $3\Pi_2$ – затраты на оплату труда сотрудника после внедрения,

OT_{BH2} – страховой взнос во внебюджетные фонды,

$3_{ЭВМ2}$ – затраты, связанные с эксплуатацией ЭВМ после внедрения,

$M_{з2}$ – материальные затраты, годовые материальные затраты на сопровождение программы для микроконтроллера составляют 2000 руб.,

$НР_2$ – накладные расходы.

Временные затраты работы сотрудника в месяцах:

$$T_{2\text{мес}} = \frac{T_{2\text{час}}}{\text{Ч}_{рд} \cdot \text{Ч}_{рч}}, \quad (16)$$

где $T_{2\text{мес}}$, $T_{2\text{час}}$ – время, затрачиваемое сотрудником на обработку результатов, в месяцах и часах соответственно ($T_{2\text{час}} = 300$ часов),

$\text{Ч}_{рд}$ – число рабочих дней в месяц,

Чрч – число рабочих часов в день.

$$T_{2мес} = \frac{300}{22 \cdot 8} = 1,70 \text{ мес.}$$

Тогда затраты на оплату труда сотрудника:

$$ЗП_2 = O_c \cdot T_{2мес} \cdot (1 + k_d) \cdot (1 + k_y), \quad (17)$$

где O_c – оклад сотрудника (оклад составляет 15000 руб.)

$$ЗП_2 = 15000 \cdot 1,70 \cdot (1 + 0,2) \cdot (1 + 1,15) = 65790 \text{ руб.}$$

Страховой взнос после внедрения вычисляются по формуле:

$$OT_{ВН2} = ЗП_2 \cdot 0,3, \quad (18)$$

$$OT_{ВН2} = 65790 \cdot 0,3 = 19737 \text{ руб.}$$

Рассчитываем затраты, связанные с эксплуатацией ЭВМ после внедрения по формуле:

$$З_{ЭВМ2} = T_{2час} \cdot C_{М-Ч}, \quad (19)$$

$$З_{ЭВМ2} = 300 \cdot 10 = 3000 \text{ руб.}$$

Подставив соответствующие значения в формулу, получим:

$$C_2 = 65790 + 19737 + 3000 + 2000 = 90527 \text{ руб.}$$

Таким образом, текущие затраты на содержание системы до внедрения разработанной программы для микроконтроллера составляют 168132 руб., после внедрения 90527 руб.

4.4 Расчет экономической целесообразности разработки и внедрения информационных технологий

Для разрабатываемого проекта расчет экономической эффективности производится исходя из следующих условий:

– годовые текущие затраты до внедрения автоматизированной системы, $C_1 = 168132$ руб.,

– годовые текущие затраты после внедрения системы, $C_2 = 90527$ руб.,

- горизонт расчета принимается исходя из срока использования разработки, $T = T_n = 2$ годам,
- шаг расчета равен одному году, $t = 1$ году,
- капитальные вложения равны затратам на создание системы, $K = 92749$ руб.,
- норма дисконта равна норме дохода на капитал, $E = 12\%$.

Ожидаемая условно-годовая экономия от внедрения системы рассчитывается по формуле:

$$\mathcal{E}_{yg} = C_1 - C_2 + \sum \mathcal{E}_i, \quad (20)$$

где \mathcal{E}_{yg} – ожидаемая условно-годовая экономия, руб.,

C_1 – годовые текущие затраты до внедрения автоматизированной системы, руб.,

C_2 – годовые текущие затраты после внедрения системы, руб.,

$\Sigma \mathcal{E}_i$ – ожидаемый дополнительный эффект от различных факторов, руб.

Так как основным фактором, по которому производится расчет экономической эффективности от внедрения программы для микроконтроллера, является уменьшение времени обработки результатов тестирования и дополнительный эффект не учитывается, то $\Sigma \mathcal{E}_i = 0$.

Подставив вычисленные выше значения в формулу, получим:

$$\mathcal{E}_{yg} = 168132 - 90527 = 77605 \text{ руб.}$$

где \mathcal{E}_{yg} – ожидаемая условно-годовая экономия, руб.

Величина ожидаемого годового экономического эффекта от внедрения ИС рассчитывается по формуле:

$$\mathcal{E}_g = \mathcal{E}_{yg} - K \cdot E_n, \quad (21)$$

где \mathcal{E}_g – ожидаемый годовой экономический эффект, руб.,

\mathcal{E}_{yg} – ожидаемая условно-годовая экономия, руб.,

K – капитальные вложения (равны затратам на создание ИС), руб.,

E_n – нормативный коэффициент экономической эффективности капитальных вложений.

Нормативный коэффициент экономической эффективности капитальных вложений определяется по формуле:

$$E_n = \frac{1}{T_n}, \quad (22)$$

где T_n – нормативный срок окупаемости капитальных вложений, лет.

Подставив вычисленные выше значения в формулу, получим:

$$\mathcal{E}_z = 77605 - 92749 \cdot 0,5 = 31230,5 \text{ руб.}$$

Расчетный коэффициент экономической эффективности капитальных вложений составляет:

$$E_p = \frac{\mathcal{E}_{yz}}{K}, \quad (23)$$

где E_p – расчетный коэффициент экономической эффективности капитальных вложений,

\mathcal{E}_{yz} – ожидаемая условно–годовая экономия, руб.,

K – капитальные вложения на создание системы, руб.

Подставив вычисленные выше значения в формулу, получим:

$$E_p = \frac{77605}{92749} = 0,83$$

Расчетный срок окупаемости капитальных вложений составляет:

$$T_p = \frac{1}{E_p}, \quad (24)$$

где E_p – коэффициент экономической эффективности капитальных вложений.

Подставив вычисленные выше значения в формулу, получим:

$$T_p = \frac{1}{0,83} = 1,3 \text{ год.}$$

Срок окупаемости без дисконтирования 1,3 год.

Чистый дисконтированный доход (ЧДД) определяется как сумма текущих эффектов за весь расчетный период, приведенная к начальному шагу, или как превышение интегральных результатов над интегральными затратами.

Если в течение расчетного периода не происходит инфляционного изменения цен или расчет производится в базовых ценах, то величина ЧДД для постоянной нормы дисконта вычисляется по формуле:

$$\text{ЧДД} = \sum_{t=1}^T (P_t - Z_t) \cdot \frac{1}{(1 + E)^t} - K, \quad (25)$$

где P_t – ожидаемые результаты от внедрения предложенной ИС, руб.,

Z_t – ожидаемые затраты (капитальные и текущие) на создание и эксплуатацию ИС, руб.,

$\Delta t = (P_t - Z_t)$ – эффект, достигаемый на t -м шаге расчета,

K – капитальные вложения,

t – номер шага расчета ($t = 1, 2$),

T – горизонт расчета,

E – постоянная норма дисконта, 12%.

$\Delta t = (P_t - Z_t) = \Delta_{\text{уг}} = 77605$ руб. В том случае, если текущие затраты (Z_t) на весь срок использования разработки равны 0.

$t = 1, 2$ год, т.к. предполагается, что результат от внедрения предложенной ИС будет с текущего года внедрения ИС.

Если ЧДД инвестиционного проекта положителен, то проект является эффективным (при данной норме дисконта).

Тогда суммарный чистый дисконтированный доход за весь горизонт расчета равен:

$$\text{ЧДД} = \Delta_1 \cdot \frac{1}{(1 + E)} + \Delta_2 \cdot \frac{1}{(1 + E)^2}, \quad (26)$$

$$ЧДД = \frac{77605}{(1 + 0,12)} + \frac{77605}{(1 + 0,12)^2} - 92749 = 45831,35 \text{ руб.}$$

Положительное значение чистого дисконтированного дохода, $ЧДД > 0$, свидетельствует о том, что инвестирование целесообразно и данная ИС может приносить прибыль в установленном объеме.

Индекс доходности (ИД) представляет собой отношение суммы приведенных эффектов к величине капитальных вложений и определяется по формуле:

$$ИД = \frac{1}{K} \sum_{t=1}^T (P_t - Z_t) \cdot \frac{1}{(1 + E)^t}, \quad (27)$$

где K – величина капиталовложений или стоимость инвестиций.

$$ИД = \frac{138580,35}{92749} = 1,4$$

Инвестиции считаются эффективными, если индекс доходности выше единицы, $ИД > 1$, следовательно, инвестиции в данную ИС, эффективны.

Внутренняя норма доходности (ВНД):

при $E_1 \rightarrow ЧДД_1 > 0$

$E_2 \rightarrow ЧДД_2 < 0$

$$ВНД = E_1 + \frac{ЧДД_1}{ЧДД_1 - ЧДД_2} \cdot (E_2 - E_1), \quad (28)$$

при $E_1 \rightarrow ЧДД_1 > 0$

$E_2 \rightarrow ЧДД_2 > 0$

$$ВНД = E_1 + \frac{ЧДД_1}{ЧДД_1 + ЧДД_2} \cdot (E_2 - E_1) \quad (29)$$

$E_1 = 0,11$

$$ЧДД_1 = \frac{77605}{(1 + 0,11)} + \frac{77605}{(1 + 0,11)^2} - 92749 = 40151,3 \text{ руб.}$$

$$E_1 = 0,13$$

$$ЧДД_2 = \frac{77605}{(1 + 0,13)} + \frac{77605}{(1 + 0,13)^2} - 92749 = 36704 \text{ руб.}$$

$$E_1 \rightarrow ЧДД_1 > 0$$

$$E_2 \rightarrow ЧДД_2 > 0$$

$$ВНД = 0,10 + \frac{40151,3}{40151,3 + 36704} \cdot (0,13 - 0,10) = 0,01.$$

Таким образом, норма дисконта должна быть в пределах 10% – 13%.

Показатели экономической целесообразности разработки и внедрения программы для микроконтроллера сведены в таблицу 11.

Таблица 11 – Показатели экономической целесообразности разработки и внедрения программы для микроконтроллера

Наименование показателя	Значения
Затраты на разработку и внедрение ПП, руб.	92749
Ожидаемая экономия от внедрения ПП, руб.	77605
Чистый дисконтированный доход, руб.	45831,35
Индекс доходности	1,4
Внутренняя норма доходности	0,01
Дисконтированный срок окупаемости, лет	1,3
Срок морального старения, года	2

Произведенные расчеты свидетельствуют, что внедрение, разработанной в ВКР программы для микроконтроллера, позволит сократить временные затраты на обработку результатов тестирования, что приведет к сокращению годовых текущих затрат на 77605 рублей

Опираясь на оценку экономической эффективности можно сделать вывод о том, что разработка и внедрение предлагаемой программы для микроконтроллера является экономически обоснованной и целесообразной.

ЗАКЛЮЧЕНИЕ

В ходе работы был разработан проект «Умная копилка» на базе микроконтроллера семейства AVR» и получены следующие результаты.

Был проведен обзор существующих на рынке копилок и наиболее популярные их виды, а также исполнения их электрических схем.

Были сформулированы минимальные требования к простоте систем на основе проведенного анализа рынка.

В конструкторском разделе проведено сравнение инструментов для решения поставленных задач, выбраны средства Easy EDA, Arduino IDE.

После этого были разработаны структурная, функциональная, принципиальная схемы устройства.

Был проведен анализ элементной базы, необходимой для реализации проекта, и подобраны комплектующие для создания устройства.

Разработка печатной платы выполнена на основе разработанных функциональной и принципиальной схемах и отобранной элементной базе.

Для функционирования копилки разработан алгоритм управляющей программы и написан непосредственно ее код.

Были рассмотрены необходимые мероприятия по снижению воздействия выявленных вредных и опасных производственных факторов, по оптимизации освещенности на рабочем месте, утилизация оборудования при разработке «Умной копилки».

Экономические показатели рассчитаны, установлено, что проект является экономически эффективным, и нет объективных причин для отказа от его реализации.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ Р 50962-96. Посуда и изделия хозяйственного назначения из пластмасс. Общие технические условия. [Электронный ресурс]-URL: <https://internet-law.ru/> (Дата обращения: 18.02.2022)
2. ГОСТ Р 55102-2012. Национальный стандарт Российской Федерации Ресурсосбережение обращение с отходами. [Электронный ресурс]-URL: <https://docs.cntd.ru> (Дата обращения: 20.02.2022)
3. ГОСТ Р 55710-2013. Национальный стандарт Российской Федерации. Освещение рабочих мест внутри зданий Нормы и методы измерений. [Электронный ресурс]-URL: <https://docs.cntd.ru> (Дата обращения: 28.02.2022)
4. Что такое микроконтроллер, зачем он нужен и как его используют. [Электронный ресурс]-URL: <https://future2day.ru> (Дата обращения: 15.02.2022)
5. Что такое микроконтроллер? Определение характеристик и архитектуры. [Электронный ресурс]-URL: <https://bulze.ru> (Дата обращения: 16.02.2022)
6. Микроконтроллеры по способу программирования <https://molotokrus.ru> (Дата обращения: 17.02.2022)
7. AVR. Учебный Курс. Программирование на Си. Часть 1 | Электроника для всех. [Электронный ресурс]-URL: <https://easyelectronics.ru> (Дата обращения: 18.02.2022)
8. Лучшие копилки для денег. [Электронный ресурс]-URL: <https://yanashla.com> (Дата обращения: 19.02.2022)
9. Умные устройства: разновидности, практическая польза, тенденции развития. [Электронный ресурс]-URL: <https://digitalsquare.ru> (Дата обращения: 19.02.2022)
10. Копилка со счётчиком монет своими руками. [Электронный ресурс]-URL: | AlexGyver (Дата обращения: 20.02.2022)

11. Требования к системе. [Электронный ресурс]-URL: <https://mydocx.ru> (Дата обращения: 20.02.2022)
12. Характеристика объектов автоматизации - Техническое задание на создание автоматизированной системы Частное охранное предприятие. [Электронный ресурс]-URL: <https://topuch.ru> (Дата обращения: 20.02.2022)
13. Что такое микроконтроллер, зачем он нужен и как его используют. [Электронный ресурс]-URL: <https://svetl-ok.ru> (Дата обращения: 21.02.2022)
14. Копилка со счетчиком монет. [Электронный ресурс]-URL: <https://AlexGyver/MoneyBox.ru> (Дата обращения: 25.02.2022)
15. Умная копилка на базе Ардуино. [Электронный ресурс]-URL: <https://xn--j1ahfl.xn--p1ai.ru> (Дата обращения: 26.02.2022)
16. Техническое задание по ГОСТ 34 - разделы 4-8. [Электронный ресурс]-URL: <https://RuGost.ru> (Дата обращения: 26.02.2022)
17. Как написать Техническое задание по ГОСТу. [Электронный ресурс]-URL: | <https://DocPlace.ru> (Дата обращения: 27.02.2022)
18. Умные устройства: разновидности, практическая польза, тенденции развития | Статьи. [Электронный ресурс]-URL: <https://digitalsquare.ru> (Дата обращения: 28.02.2022)
19. Мероприятия по снижению опасных и вредных факторов [Электронный ресурс]-URL: <https://studwood.net/> (Дата обращения: 13.04.2022)
20. Оптимизация освещения помещений и рабочих мест [Электронный ресурс]-URL: <https://studfile.net/preview/9480244/page:50/> (Дата обращения: 19.04.2022)

ПРИЛОЖЕНИЕ А – Техническое задание

наименование организации - разработчика ТЗ на АС

УТВЕРЖДАЮ
Руководитель _____

УТВЕРЖДАЮ
Руководитель _____

Личная подпись

Расшифровка
подписи

Личная подпись

Расшифровка
подписи

Печать

Дата

Печать

Дата

наименование вида АС

наименование объекта автоматизации

сокращенное наименование АС

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На 7 листах

Действует с _____

СОГЛАСОВАНО

Руководитель _____

Личная
подпись _____

Расшифровка
подписи _____

Печать
Дата

Лысьва 2022г.

1. Общие сведения

1.1. Наименование системы

1.1.1. Полное наименование системы

Полное наименование системы: «Умная» копилка

1.1.2. Краткое наименование системы

Краткое наименование: УК.

1.2. Основания для проведения работ

Работа выполняется на основании индивидуального задания на выпускную квалификационную работу.

1.3. Наименование организаций – Заказчика и Разработчика

1.3.1. Заказчик

Заказчик: ЛФ ПНИПУ

Адрес фактический: г. Лысьва 2

Телефон: 8(34249)6-12-55

1.3.2. Разработчик

Разработчик: Худякова Екатерина Владимировна гр. КСК9-18-1 спо

Адрес фактический: г. Лысьва ...

Телефон: 8(95233).....

1.4. Плановые сроки начала и окончания работы

Начало работы: 18.04.22

Конец работы: 05.06.22

1.5. Источники и порядок финансирования

Финансирование не предусмотрено.

1.6. Порядок оформления и предъявления заказчику результатов работ

1. Выбор темы работы;

2. Разработка технического задания;

3. Выбор элементной базы проекта;

4. Выбор инструментального обеспечения проекта;

5. Выбор инструментального обеспечения проекта;

6. Разработка принципиальной схемы устройства;

7. Разработка печатной платы устройства;
8. Разработка программной части устройства.

2. Назначение и цели создания системы.

2.1. Назначение системы.

Электронный распознаватель монет (по размеру) для копилки со счетчиком суммы и статистикой по каждому типу монет.

Возможности:

- распознавание размера с высокой точностью и его привязка к стоимости каждой монеты;
- вычисление общей суммы монет в копилке;
- статистика по числу монет каждого типа;
- все настройки сохраняются в энергозависимую память и не сбрасываются при питании;
- накопленная сумма тоже хранится в энергозависимой памяти и не боится сбоев питания;
- режим глубокого энергосбережения: в спящем режиме потребляется 0,07 мА, в схеме без преобразователя 0,02 мА;
- поддержка любого числа монет разного размера;
- автоматическая калибровка типов монет;
- сброс накопленного количества.

2.2. Цели создания системы

УК создается с целью совершенствования стандартной копилки для монет.

3. Требования к системе

3.1. Требования к системе в целом

3.1.1. Требования к функционированию системы

В Системе предлагается выделить следующие функциональные подсистемы:

- блок распознавания размера монет, вычисления общей суммы монет в копилке, статистики по числу монет каждого типа;
- блок, где настройки сохраняются в энергозависимую память и не сбрасываются при питании;
- подсистема формирования и визуализации отчетности.

3.1.2. Требования к надежности

3.1.2.1. Перечень аварийных ситуаций, по которым регламентируются требования к надежности

Под аварийной ситуацией понимается аварийное завершение процесса, а также «зависание» этого процесса.

При работе системы возможны следующие аварийные ситуации, которые влияют на надежность работы системы:

- сбой в электроснабжении;
- ошибки УК, не выявленные при отладке и испытании системы.

3.1.2.2. Требования к надежности технических средств

К надежности оборудования предъявляются следующие требования:

- должны использоваться средства с повышенной надежностью;
- применение технических средств соответствующих классу решаемых задач;
- в случае сбоев данные должны иметь возможность восстановления.

3.1.3. Требования к эргономике к технической эстетике

В части внешнего оформления:

- язык: русский;
- должен использоваться шрифт: Times New Roman;
- размер шрифта должен быть: 12;
- цветовая палитра должна быть: серая;
- цветовая палитра корпуса должна быть: серая;
- на корпусе изделия должен использоваться логотип Заказчика.

3.1.4. Требования по сохранности информации при авариях

Критерии, при которых должна быть обеспечена сохранность информации в системе: аварии, отказ технических средств, потеря питания.

3.1.5. Требования к защите от влияния внешних воздействий

Требования к защите от влияния внешних воздействий:

1. Крепкий корпус изделия (Пластмасса вида: HDPE или PE HD);
2. Водонепроницаемое покрытие;
3. Аккумуляторные батарейки .

3.1.6. Дополнительные требования

3.1.7. Требования безопасности

- перед работой должна быть проверена работа аккумуляторных батарей, и всех составных частей УК;
- соблюдать меры предосторожности;
- работать с изделием чистыми руками.

3.2. Требования к видам обеспечения

3.2.1. Требования к информационному обеспечению

Требования:

- к составу, структуре и способам организации данных в системе;
- к информационному обмену между компонентами системы;
- к информационной совместимости со смежными системами;
- к структуре процесса сбора, обработки, передачи данных в системе и представлению данных;
- к защите данных от разрушений при авариях и сбоях в электропитании системы;
- к контролю, хранению, обновлению и восстановлению данных.

3.2.1.1. Требования к структуре и способам организации данных в системе

Структура хранения данных в УК должна состоять из следующих основных областей:

- область временного хранения данных;
- область постоянного хранения данных;
- область витрин данных.

3.2.1.2. Требования к защите данных от разрушений при авариях и сбоях в электропитании системы

Информация в базе данных системы должна сохраняться при возникновении аварийных ситуаций, связанных со сбоями электропитания.

3.2.1.3. Требования к контролю, хранению, обновлению и восстановлению данных

К контролю данных предъявляются следующие требования:

- система должна протоколировать все события, связанные с изменением своего информационного наполнения.

К хранению данных предъявляются следующие требования:

- данные не должны стираться при выключении питания.

К обновлению данных предъявляются следующие требования:

- данные должны обновляться при работе с УК;

3.2.2. Требования к лингвистическому обеспечению

Язык для чтения информации по количеству монет: русский.

3.2.3. Требования к метрологическому обеспечению

Не предъявляются.

4. Состав и содержание работ по созданию системы

Таблица 12 – Состав и содержание работ по созданию системы

№ пп	Этапы работы	Объём этапа, %	Сроки выполнения		Примечан ие
			Начало	Конец	
1.	Выбор темы работы	5	13.09.22	30.10.22	
2.	Разработка технического задания	15	11.03.22	15.03.22	
3.	Выбор элементной базы проекта	20	16.03.22	17.03.22	

Продолжение таблицы 12

4.	Выбор инструментального обеспечения проекта	25	18.03.22	19.03.22	
5.	Разработка функциональной схемы устройства	40	20.03.22	21.03.22	
6.	Разработка принципиальной схемы устройства	50	22.03.22	24.03.22	
7.	Разработка печатной платы устройства	70	25.03.22	28.03.22	
8.	Разработка программной части устройства	85	29.03.22	07.04.22	
9.	Сборка устройства	90	08.04.22	1.05.22	
10	Защита работы	100	22.06.22	22.06.22	

5. Порядок контроля и приемки системы

5.1. Виды и объем испытаний

Виды испытаний:

1. Предварительные испытания.

2. Опытная эксплуатация.

3. Приемочные испытания.

6. Требования к документированию

– ГОСТ 34.601-90 – Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания;

– ГОСТ 34.201-89 – Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплексность и обозначение документов при создании автоматизированных систем;

– ГОСТ 7.32-2017 «Отчет о научно-исследовательской работе. Структура и правила оформления».

7. Источники разработки

Настоящее Техническое Задание разработано на основе следующих документов и информационных материалов:

- ГОСТ 7.32-2017 «Отчет о научно-исследовательской работе. Структура и правила оформления»;
- ГОСТ 15.016-2016 «Система разработки и постановки продукции на производство»;
- ГОСТ 34.602-89 «Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы»;
- ГОСТ 12.1.004-91 «ССБТ. Пожарная безопасность. Общие требования».

ПРИЛОЖЕНИЕ Б – Код программы

```
//-----НАСТРОЙКИ-----
#define coin_amount 5 // число монет, которые нужно распознать
float coin_value[coin_amount] = {0.5, 1.0, 2.0, 5.0, 10.0}; // стоимость монет
String currency = "RUB"; // валюта (английские буквы!!!)
int stb_time = 10000; // время бездействия, через которое система уйдёт в сон (миллисекунды)
//-----НАСТРОЙКИ-----

int coin_signal[coin_amount]; // тут хранится значение сигнала для каждого размера монет
int coin_quantity[coin_amount]; // количество монет
byte empty_signal; // храним уровень пустого сигнала
unsigned long standby_timer, reset_timer; // таймеры
float summ_money = 0; // сумма монет в копилке

//-----БИБЛИОТЕКИ-----
#include "LowPower.h"
#include "EEPROM.h"
#include "LCD_1602_RUS.h"
//-----БИБЛИОТЕКИ-----

LCD_1602_RUS lcd(0x27, 16, 2); // создать дисплей
// если дисплей не работает, замени 0x27 на 0x3f

boolean recogn_flag, sleep_flag = true; // флажки
//-----КНОПКИ-----
#define button 2 // кнопка "проснуться"
#define calibr_button 3 // скрытая кнопка калибровки сброса
#define disp_power 12 // питание дисплея
#define LEDpin 11 // питание светодиода
#define IRpin 17 // питание фототранзистора
#define IRSens 14 // сигнал фототранзистора
//-----КНОПКИ-----
int sens_signal, last_sens_signal;
boolean coin_flag = false;

void setup() {
  Serial.begin(9600); // открыть порт для связи с ПК для отладки
  delay(500);

  // подтягиваем кнопки
  pinMode(button, INPUT_PULLUP);
  pinMode(calibr_button, INPUT_PULLUP);

  // пины питания как выходы
  pinMode(disp_power, OUTPUT);
  pinMode(LEDpin, OUTPUT);
  pinMode(IRpin, OUTPUT);

  // подать питание на дисплей и датчик
  digitalWrite(disp_power, 1);
  digitalWrite(LEDpin, 1);
  digitalWrite(IRpin, 1);

  // подключить прерывание
  attachInterrupt(0, wake_up, CHANGE);

  empty_signal = analogRead(IRSens); // считать пустой (опорный) сигнал
```

```

// инициализация дисплея
lcd.init();
lcd.backlight();

if (!digitalRead(calibr_button)) { // если при запуске нажата кнопка калибровки
  lcd.clear();
  lcd.setCursor(3, 0);
  lcd.print(L"Сервис");
  delay(500);
  reset_timer = millis();
  while (1) {
    if (millis() - reset_timer > 3000) { // бесконечный цикл
      // если кнопка всё ещё удерживается и прошло 3 секунды
      // очистить количество монет
      for (byte i = 0; i < coin_amount; i++) {
        coin_quantity[i] = 0;
        EEPROM.writeInt(20 + i * 2, 0);
      }
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print(L"Память очищена");
      delay(100);
    }
    if (digitalRead(calibr_button)) { // если отпустили кнопку, перейти к калибровке
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print(L"Калибровка");
      break;
    }
  }
}

while (1) {
  for (byte i = 0; i < coin_amount; i++) {
    lcd.setCursor(0, 1); lcd.print(coin_value[i]); // отобразить цену монеты, размер которой калибруется
    lcd.setCursor(13, 1); lcd.print(currency); // отобразить валюту
    last_sens_signal = empty_signal;
    while (1) {
      sens_signal = analogRead(IRsens); // считать датчик
      if (sens_signal > last_sens_signal) last_sens_signal = sens_signal; // если текущее значение больше предыдущего
      if (sens_signal - empty_signal > 3) coin_flag = true; // если значение упало почти до "пустого", считать что монета улетела
      if (coin_flag && (abs(sens_signal - empty_signal) < 2)) { // если монета точно улетела
        coin_signal[i] = last_sens_signal; // записать максимальное значение в память
        EEPROM.writeInt(i * 2, coin_signal[i]);
        coin_flag = false;
        break;
      }
    }
  }
}
}

```

```

        break;
    }
}

// при старте системы считать из памяти сигналы монет для дальнейшей работы, а также их количество в банке
for (byte i = 0; i < coin_amount; i++) {
    coin_signal[i] = EEPROM.readInt(i * 2);
    coin_quantity[i] = EEPROM.readInt(20 + i * 2);
    summ_money += coin_quantity[i] * coin_value[i]; // ну и сумму сразу посчитать, как произведение цены монеты на количество
}

/*
// для отладки, вывести сигналы монет в порт
for (byte i = 0; i < coin_amount; i++) {
    Serial.println(coin_signal[i]);
}
*/
standby_timer = millis(); // обнулить таймер ухода в сон
}

void loop() {
    if (sleep_flag) { // если проснулись после сна, инициализировать дисплей и вывести текст, сумму и валюту
        delay(500);
        lcd.init();
        lcd.clear();
        lcd.setCursor(0, 0); lcd.print(L"На яхту");
        lcd.setCursor(0, 1); lcd.print(summ_money);
        lcd.setCursor(13, 1); lcd.print(currency);
        empty_signal = analogRead(IRsens);
        sleep_flag = false;
    }

    // далее работаем в бесконечном цикле
    last_sens_signal = empty_signal;
    while (1) {
        sens_signal = analogRead(IRsens); // далее такой же алгоритм, как при калибровке
        if (sens_signal > last_sens_signal) last_sens_signal = sens_signal;
        if (sens_signal - empty_signal > 3) coin_flag = true;
        if (coin_flag && (abs(sens_signal - empty_signal) < 2) {
            recogn_flag = false; // флажок ошибки, пока что не используется
            // в общем нашли максимум для пролетевшей монетки, записали в last_sens_signal
            // далее начинаем сравнивать со значениями для монет, хранящихся в памяти
            for (byte i = 0; i < coin_amount; i++) {
                int delta = abs(last_sens_signal - coin_signal[i]); // вот самое главное! ищем АБСОЛЮТНОЕ (то бишь по модулю)
                // значение разности полученного сигнала с нашими значениями из памяти
                if (delta < 30) { // и вот тут если эта разность попадает в диапазон, то считаем монетку распознанной
                    summ_money += coin_value[i]; // к сумме тупо прибавляем цену монетки (дада, сумма считается двумя разными способами. При старте системы суммой всех монет, а тут прибавление

```

```

        lcd.setCursor(0, 1); lcd.print(summ_money);
        coin_quantity[i]++; // для распознанного номера монетки прибавляем количество
        recogn_flag = true;
        break;
    }
}
coin_flag = false;
standby_timer = millis(); // сбросить таймер
break;
}

// если ничего не делали, время таймера вышло, спим
if (millis() - standby_timer > stb_time) {
    good_night();
    break;
}

// если монетка вставлена (закрывает контакты) и удерживается 2 секунды
while (!digitalRead(button)) {
    if (millis() - standby_timer > 2000) {
        lcd.clear();

        // отобразить на дисплее: сверху цены монет (округлено до целых!!!!), снизу их количество
        for (byte i = 0; i < coin_amount; i++) {
            lcd.setCursor(i * 3, 0); lcd.print((int)coin_value[i]);
            lcd.setCursor(i * 3, 1); lcd.print(coin_quantity[i]);
        }
    }
}
}
}

// функция сна
void good_night() {
    // перед тем как пойти спать, записываем в EEPROM новые полученные количества монет по адресам начиная с 20го (пук кек)
    for (byte i = 0; i < coin_amount; i++) {
        EEPROM.updateInt(20 + i * 2, coin_quantity[i]);
    }
    sleep_flag = true;
    // вырубить питание со всех дисплеев и датчиков
    digitalWrite(disp_power, 0);
    digitalWrite(LEDpin, 0);
    digitalWrite(IRpin, 0);
    delay(100);
    // и вот теперь спать
    LowPower.powerDown(SLEEP_FOREVER, ADC_OFF, BOD_OFF);
}

// просыпаемся по ПЕРЕРЫВАНИЮ (эта функция - обработчик прерывания)
void wake_up() {
    // возвращаем питание на дисплей и датчик
    digitalWrite(disp_power, 1);
    digitalWrite(LEDpin, 1);
    digitalWrite(IRpin, 1);
    standby_timer = millis(); // и обнуляем таймер
}
}

```

ПРИЛОЖЕНИЕ В – Блок-схема

