**Министерство образования Иркутской области**

Государственное бюджетное профессиональное образовательное учреждение

Иркутской области

«Иркутский Авиационный техникум»

(ГБПОУИО «ИАТ»)

|  |
| --- |
| **ДП.09.02.01.20.161.6.ПЗ** УТВЕРЖДАЮ |
| Зам. директора по УР, к.т.н. |
| \_\_\_\_\_\_\_\_\_ Коробкова Е.А. |

ИЗМЕРИТЕЛЬ ЁМКОСТИ АККАМУЛЯТОРА НА МИКРОКОНТРОЛЛЕРЕ

|  |  |  |
| --- | --- | --- |
| Зав. отделения | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (С.Н. Быкова) |
| Нормоконтролер: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (А.Э. Кондратенко) |
| Консультант по экономической части | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (М.А. Рачкова) |
|  |  |  |
| Руководитель: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (Д.В. Шатурский) |
| Студент: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) | (А.Е. Кондратьев) |

Содержание

[Список условных сокращений 3](#_Toc40712459)

[Введение 4](#_Toc40712460)

[1 Предпроектное исследование 5](#_Toc40712461)

[1.1 Обзор существующих устройств, способных измерять ёмкость аккумуляторов 5](#_Toc40712462)

[1.2 Вывод сравнительного анализа с выделением критерий сравнения представленых устройств, способных выполнять поставленную цель дипломного проекта 5](#_Toc40712463)

[1.3 Обоснование создаваемого устройства 9](#_Toc40712464)

[1.4 Область использования устройства 10](#_Toc40712465)

[2 Описание алгоритма работы прибора 12](#_Toc40712466)

[2.1 Функциональный блок – схема с учётом всех контактов 12](#_Toc40712467)

[2.2 Описание функциональной блок – схемы 12](#_Toc40712468)

[2.3 Описание алгоритма работы устройства 13](#_Toc40712469)

[3 Выбор микроконтроллера 16](#_Toc40712470)

[4 Выбора элементной базы 24](#_Toc40712471)

[5 Описание выбора системы автоматизированного проектирования (САПР) 26](#_Toc40712472)

[5.1 Обзор системы автоматизированного проектирования 26](#_Toc40712473)

[5.2 Элементная база выбранной системы автоматизированного проектирования 28](#_Toc40712474)

[6 Описание технической реализации устройства 38](#_Toc40712475)

[6.1 Разработка принципиальной схемы в выбранном САПРе 38](#_Toc40712476)

[6.2 Разработка электрической схемы в выбранном САПРе 39](#_Toc40712477)

[7 Описание сборки и настройки устройства 40](#_Toc40712478)

[7.1 Описание сборки устройства 40](#_Toc40712479)

[7.2 Описание настройки устройства в его функциональной особенности 41](#_Toc40712480)

[7.3 Эскиз печатной платы и проекта 42](#_Toc40712481)

[8 Описание тестирования и отладки платы 43](#_Toc40712482)

[8.1 Составление алгоритма тестирования 43](#_Toc40712483)

[8.2 Инструменты для аппаратного тестирования 43](#_Toc40712484)

[8.3 Проведение тестирования и его результаты 44](#_Toc40712485)

[9 Экономическая часть 48](#_Toc40712486)

[9.1 Организационно–экономическое обоснование проекта 48](#_Toc40712487)

[9.2 Расчёт себестоимости 48](#_Toc40712488)

[9.3 Расчёт затрат на электроэнергию и амортизацию оборудования 50](#_Toc40712489)

[9.4 Расчет начислений на заработную плату 52](#_Toc40712490)

[9.5 Расчет общей себестоимости проекта 53](#_Toc40712491)

[Заключение 55](#_Toc40712492)

[Список использованных источников 56](#_Toc40712493)

[Приложение А – Листинг AKB 57](#_Toc40712494)

[Приложение Б – Библиотека AKB 65](#_Toc40712495)

Список условных сокращений

АКБ – аккумуляторная батарея;

АЦП – аналогово-цифровой преобразователь;

Введение

Самым важным параметром каждой АКБ является ёмкость аккумулятора. Она определяет количество энергии, отдаваемой им за каждый отрезок времени. Это касается всех батарей от автомобильных до телефонных. Знать о них и понимать устройство важно, поскольку при использовании батареи не той емкости могут возникнуть серьезные проблемы при запуске этих устройств.

Единицы измерения этой величины – Амперы или Миллиамперы / час. По данному параметру осуществляют выбор АКБ для техники, руководствуясь рекомендованными значениями. При нарушении рекомендаций, к примеру, автомобиль может не запуститься в холодное время суток.

Задачи:

– провести предпроектное исследование существующих устройств, способных измерять ёмкость аккумуляторов;

– составить алгоритм работы устройства;

– проанализировать существующие микроконтроллеры, способные выполнить поставленную цель дипломного проекта;

– выбрать элементную базу для разработки устройства и обосновать выбор;

– выполнить проектирование устройства в системе автоматизированного проектирования;

– описать техническую часть реализации устройства;

– провести тестирование устройства и составить отчёт;

– составить техническое руководство по использованию устройства;

– произвести расчёт стоимости устройства и затрат на весь проект;

– составить заключение.

Цель: Спроектировать и реализовать устройство, способное измерять ёмкость аккумуляторов с минимальной погрешностью.

# 1 Предпроектное исследование

1.1 Обзор существующих устройств, способных измерять ёмкость аккумуляторов

На каждой АКБ указывают ее вместимость, но эти сведения не всегда правдивы. Для измерения емкости аккумулятора необходимо выполнить ряд действий. Для этого потребуются разные приборы и материалы.

Некоторое количество электричества, передаваемое от АКБ при его разряжении, называют энергоемкостью аккумулятора. При проведении расчетов характеристику выражают в ампер-часах (Ач) или ватт-часах (Втч). Большая вместимость батареи позволяет устройству долго работать без подзарядки, что важно при выборе АКБ для машины.

В этом случае вместимость будет зависеть от мощности генератора. В некоторых случаях емкость обозначается по-английски A/hour, то есть Ач по-русски. Ее не нужно путать с таким параметром, как напряжение, выражаемое в вольтах (В).

Ёмкость аккумуляторов можно производить с помощью следующих устройств:

– тестер (мультиметр);

– линейный стабилизатор;

– самодельные устройства.

1.2 Вывод сравнительного анализа с выделением критерий сравнения представленых устройств, способных выполнять поставленную цель дипломного проекта

Тестер – комбинированный электроизмерительный прибор, объединяющий в себе несколько функций.

Для данной задачи можно использовать тестеры двух типов:

– USB тестер;

– Multimetr.

1.2.1 USB тестер

Для измерения вместимости батареи часто используют USB-тестер. На рисунке 1 представлен USB тестер.



Рисунок 1 – USB тестер

У приспособления есть много функций, позволяющих узнать емкость АКБ смартфона, планшета и внешнего портативного зарядного устройства.

USB тестер подключается к любому портативному устройству посредством провода или переходника. Полученные показатели помогают определить изношенность аккумулятора.

На тестере присутствует кнопка управления, предназначенная для измерения разных показателей. С ее помощью переключаются режимы работы и ячейки памяти устройства. Когда подключаемый аппарат обладает необходимым напряжением, тестер начинает работать.

Снизу в углу появляется значение вместимости батареи. Стоит учесть, что тестер имеет низкую погрешность, то есть результат может быть отклонён от действительной ёмкости до одного процента измеренной ёмкости АКБ.

Так же стоит учитывать тот факт, что некоторые производители указывают на устройстве номинальную емкость, превышающую фактический показатель. Эта информация неверная и является рекламой.

1.2.2 Multimeter

Для анализа цифрового мультиметра был выбран мультиметр «Mastech MY60». На рисунке 2 представлен мультиметр Mastech MY60.



Рисунок 2 – Мультиметр Mastech MY60

Для измерения можно применить простой метод, предварительно воспользовавшись мультиметром. В процессе нужно определить значение тока любой батареи. При этом измеряется точное время, при котором элемент питания отдавал электрохимическую энергию. Замеры не будут стопроцентно точными, но покажут максимально близкое число.

В литий-ионных АКБ присутствует своя шкала разрядов, показывающая зависимость напряжения от заряда. Чтобы она не отразилась на измерениях, собирают линейное приспособление в 2,7-3 В.

1.2.3 Линейный стабилизатор

При использовании линейного устройства необходимо установить значение тока, которое рассчитывают из напряжения (U) батарейки 2,7 В.

С помощью стабилизатора проводят подключение резистора, который можно изготовить своими руками или приобрести готовый прибор в магазине.

После этого измеряют ток в цепи и устанавливают секундомер. В дальнейшем нужно следить за показателями U на клеммах. Секундомер отключают при достижении уровня в 2,7 В и записывают показания.

Значение высчитывают, умножая начальное время на ток, передвигающийся по цепи с помощью сопротивления.

Так можно более точно узнать объем емкости АКБ.

1.2.4 Самодельные устройства

Когда необходимая техника отсутствует, прибор можно сделать самому, исходя из информации, которую можно найти в интернете.

Из готовых приборов необходимо взять вольтметр, а оставшиеся детали сооружают из подручных средств. Трудности возникнут при расчетах и создании внутреннего сопротивления, для которого потребуется ток.

Подходящим материалом является нихромовая проволока, используемая для создания нагревательных спиралей в электрических плитках. Нихромовые элементы можно заменить металлической полосой из прочих нагревательных устройств.

Для напряжения 12 В показатель тока должен находиться в рамках 80-120 Ампер, а сопротивление – 0,1-0,15 Ом. Прибор для измерения такого сопротивления сложно найти. По этой причине подбирают длину одного элемента и измеряют ток, который он пропускает. После этого совмещают несколько подобных деталей.

Самодельный аппарат делают последовательно:

– подбирают нихромовую проволоку или нагревательную полосу и измеряют мультиметром до 15 А ток. Элемент должен пропускать 10-12 А;

– соединяют 10 таких деталей, получая нагрузку в 100-120 А. Проволоку необходимо надежно скручивать;

– полученный элемент помещают в подходящий корпус и фиксируют в нем. Если коробка небольшая, то проволоку несколько раз сгибают так, чтобы витки не касались друг друга. Параллельное соединение должно быть надежным, что обеспечивается изолирующими цилиндрами, которые устанавливают на изгибы;

– концы скрутки припаивают к контактам на выходе, а снаружи – к соединительным проводам;

– подключают вольтметр;

– крепят на концы соединительного кабеля зажимы, которые потом подсоединяют к аккумулятору.

После сборки устройства, можно преступать к измерению ёмкости АКБ.[]

1.3 Обоснование создаваемого устройства

Состояние аккумуляторной батареи необходимо периодически контролировать. Это позволит существенно продлить срок её службы, сохранить ёмкость и другие параметры на первоначальном уровне.

Существующие измерительные приборы позволяют достаточно точно установить силу тока в АКБ, напряжение и определить плотность электролита.

При выявлении отклонений любого из указанных параметров рекомендуется выяснять причину и проводить необходимые обслуживающие мероприятия.

Устройства для точного определения емкости АКБ в настоящее время не существует. Единственный способ, который дает относительно точные результаты, – это его полная разрядка с одновременной фиксацией большого количества различных параметров.

Однако эта процедура является очень продолжительной, и редко кто готов тратить время на контроль ёмкости таким способом. Остальные методики, как и специальный тестер, дают лишь приблизительные результаты.

По этой причине было принято решение по разработке устройства, способного измерять ёмкость АКБ с минимальной погрешностью.

1.4 Область использования устройства

Для того, чтобы установить область использования устройства по замере ёмкости АКБ, необходимо установить от чего зависит текущая емкость АКБ.

В процессе эксплуатации удельная емкость АКБ меняется. Сначала пластины разрабатываются, поэтому показатели вместимости высокие. После этого прибор начинает работать стабильно, значение не отходит от одного уровня. Затем ёмкость начинает уменьшаться из-за изнашивания пластин.

На вместимость влияют активные материалы, конструкция электролитов, электродов, их температура и концентрация, амортизация батареи, величина разрядного и пускового тока, содержание налета в электролитах и прочие факторы.

Снижение начинается при увеличении разрядного тока. Если АКБ разряжают специально, то устройство теряет меньше вместимости, чем при плавном режиме с низким уровнем тока. Это позволяет зафиксировать на корпусе показатели для разного времени разряжения.

Емкость одинаковых аккумуляторов редко меняется. Низкие показатели характерны для небольших промежутков разрядки, а высокие – для больших временных отрезков.

Значение вместимости начинает меняться, когда повышается температура электролитов. Если из-за этого превышаются допустимые нормы, то срок службы снижается. Высокая температура электролитов понижает их вязкость, и они попадают в действующую массу. При этом сопротивление начинает расти.

По этой причине коэффициент использования активной массы при разряжении больше, чем при заряжении с низкой температурой. В связи с этим нужно проверять емкость АКБ на каждом этапе его эксплуатации.

Разрабатываемое устройство можно использовать без каких-либо ограничений в сфере, где используются АКБ:

– ремонт и техническое обслуживание бытовой техники;

– ремонт и техническое обслуживание автотранспорта.

# 2 Описание алгоритма работы прибора

2.1 Функциональный блок – схема с учётом всех контактов

Для упрощения разработки устройства и понимания работы, необходимо составить блок – схему работы устройства и описать его. На рисунке 3 представлена блох – схема измерителя ёмкости АКБ.

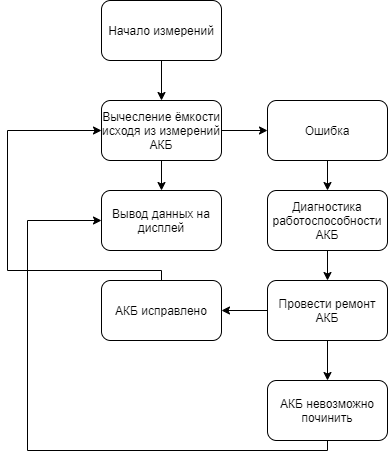


Рисунок 3 – Блок – схема измерителя ёмкости АКБ

2.2 Описание функциональной блок – схемы

Функциональная блок – схема включает в себя краткое содержание циклов и модулей, входящих в состав разрабатываемого устройства.

Для удобства измерений ёмкости АКБ, к устройству необходимо подключить контакты, способные регулировать руками и подключать на АКБ различных размеров – клешни.

Чтобы начать производить замеры, необходимо подключить клешни устройства к полюсам АКБ. При подключении клешней, устройство автоматически начнёт вычислять ёмкость АКБ исходя из заранее загруженной программы на микроконтроллер составлено на основе физических формул по определению ёмкости. Так же контроллер будет выполнять функцию «Вычисление ёмкости».

Для понимания и наглядного определения ёмкости на АКБ, необходимо установить дисплей. Дисплей будет выводить информацию о результатах измерений, либо ошибки, в ходе измерений АКБ.

2.3 Описание алгоритма работы устройства

При начале измерений ёмкости на АКБ, контроллер должен учитывать формулы, согласно которым можно рассчитать ёмкость.

Для этого будут заранее загружены следующие формулы:

1.Зависимость емкости АКБ от тока разряда.

При подключении к аккумулятору защищенной нагрузки без преобразователя, величина тока не меняется. При этом время работы АКБ будет выглядеть как соотношение емкости к току. В виде формулы эта зависимость выглядит следующим образом:

Q = I\*T,

где Q – емкость аккумулятора (А\*ч или мА\*ч);

I – постоянный ток разряда аккумулятора (А);

T – время разряда батареи (ч).

2. Зависимость емкости АКБ от энергии.

Способность к накоплению энергии батареей связана с напряжением: чем оно больше, тем больше энергии накопит АКБ. Таким образом, электрическая энергия определяется как произведение тока, напряжения и времени, что представлено на рисунке 4.

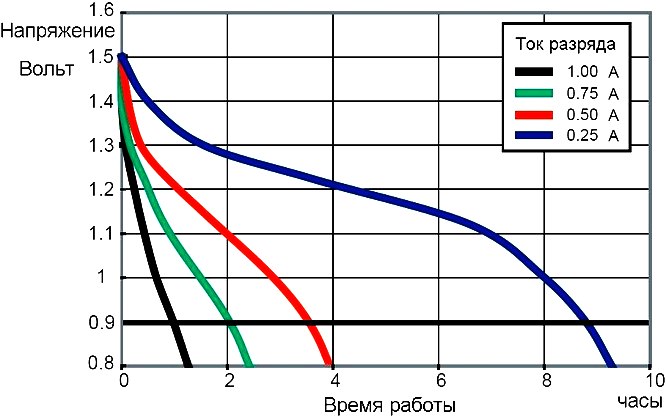


Рисунок 4 – График зависимость электроэнергии, напряжения и времени

W= I\*U\*T

где W – энергия накопленная батареей (Дж);

U – напряжение аккумулятора (В);

I – постоянный ток (А);

T – время разряда (ч).

3. Энергетическая ёмкость.

Под этим понятием подразумевается энергия, которую отдает полностью заряженный аккумулятор при разряде до минимального напряжения. Для расчета энергетической емкости используют формулу:

Q = W/4

где Q – емкость аккумулятора (А\*ч);

W – энергетическая емкость аккумулятора (Вт/элемент).

4. Резервная емкость.

Этот термин используется при расчете емкости автомобильных аккумуляторов.

Он описывает способность АКБ питать электрооборудование автомобиля, когда встроенный генератор не функционирует. Резервная емкость рассчитывается следующим образом:

Q = T/2

где Q – емкость аккумулятора (А\*ч).

T – резервная емкость аккумулятора (мин).

Рассчитав ёмкость автономно, контроллер, управляя дисплеем, выведет информацию на дисплей.

В случае, если рассчитать ёмкость не получается, то есть на дисплее выводится надпись «Ошибка», то необходимо проверить исправность АКБ.

Если, АКБ невозможно починить, то его необходимо утилизировать в специальных местах, так как АКБ чрезвычайно долго разлагаются.

Если же, АКБ возможно починить, то необходимо произвести ремонт и восстановление. После чего снова проверить ёмкость с помощью разработанного устройства.

3 Выбор микроконтроллера

Для управления всем устройством, автономным расчётом ёмкости и вывода информации на дисплей, необходимо использовать микроконтроллер или процессор.

Использование процессора для такого устройства является не рациональным расходом средств, так как для проекта подойдёт более низкопроизводительнный компонент, который можно приобрести в любом магазине радиодеталей.

Так же микроконтроллеры являются более дешёвыми компонентами, нежели процессоры.

Но стоит учитывать, что процессоры являются более высокопроизводительными и исполняют те же задачи, что и микроконтроллер.

Для сравнения микроконтроллеров были выбраны микроконтроллеры исходя из отзывов радиолюбителей и личного опыта работы с микроконтроллерами.

Список микроконтроллеров для измерителя ёмкости АКБ:

– PIC16F873;

– ATtiny26;

– ATmega328.

Микроконтроллер PIC16F873 является однокристальным 8-разрядным FLASH CMOS микроконтроллером компании «Microchip Technology Incorporated».

Характеристика микроконтроллера:

– высокоскоростная РIС архитектура;

– все команды выполняются за один цикл;

– ОС-20МГц, тактовый сигнал;

– до 8кх 14 слов FLASH памяти программ;

– до 368 х 8 байт памяти данных (ОЗУ);

– до 256 х 8 байт EEPROM памяти данных;

– система прерываний (до 14 источников);

– программируемая защита памяти программ;

– полностью статическая архитектура;

– программирование в готовом устройстве (используется два вывода микроконтроллера);

– низковольтный режим программирования;

– режим внутрисхемной отладки (используется два вывода микроконтроллера);

– широкий диапазон напряжений питания от 2.08 до 5.58;

– повышенная нагрузочная способность портов ввода/вывода (25мА).

Малое энергопотребление:

– <06мА 3.08, 4.0МГц;

– 20мкА 3.08, З2\*Гц;

– <1мКА в режиме энергосбережения;

На рисунке 5 представлен микроконтроллер PIC16F873.

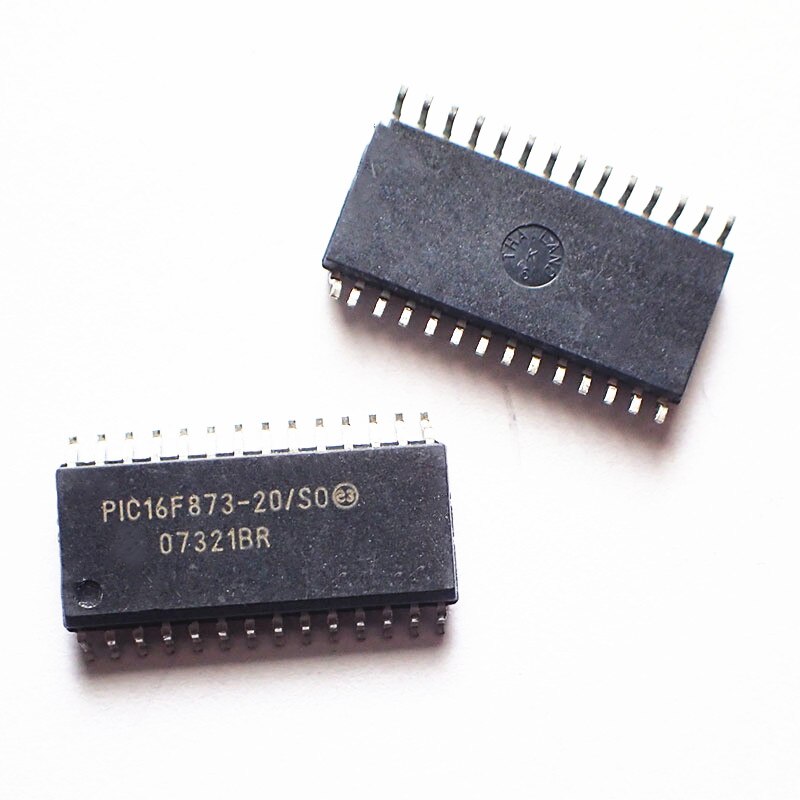


Рисунок 5 – Микроконтроллер PIC16F873

На рисунке 6 представлена схема расположения контактов микроконтроллера PIC16F873.

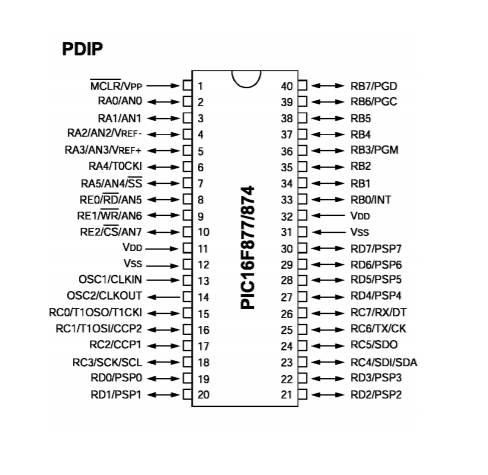


Рисунок 6 – схема расположения контактов микроконтроллера PIC16F873

ATtiny26 является микроконтроллером семейства «megaAVR».

Характеристика микроконтроллера:

– высокая производительность при малом потреблении;

– RISC архитектура;

– 118- команд, большинство исполняемых за один машинный такт;

– 328 рабочих регистра общего назначения;

– полностью статический режим;

– 16 MIPS производительность при 16 МГц.

Память:

– 2К байтов FLASH памяти программ с внутрисистемным программированием;

– 1000 циклов записи- стирания;

– 128 байтов EEPROM c внутрисистемным программированием;

– 100 000 циклов записи- стирания;

– 128 байтов внутренней SRAM;

– программируемый ключ доступа к программам и памяти данных.

Периферия:

– 8- битный таймер/счётчик с программируемым предделителем;

– 8- битный скоростной делитель с программируемым предделителем;

– 2 скоростных ШИМ выхода с отдельным выходным регистром сравнения не совмещённый выход инверсной ШИМ;

– универсальный последовательный интерфейс с детектором старта;

– 10- бит АЦП;

– 11 простых униполярных входа;

– 8 дифференциальных входа;

– 7 дифференциальных входа с программируемым усилением (1, 10);

– встроенный аналоговый компаратор.

Внешние прерывания:

– 11 прерываний по изменению потенциала вывода;

– программируемый Watchdog с переключаемыми генераторами;

– внутрисистеммное программирование через SPI порт;

– внутренний калиброванный RC генератор;

– 20 - выводной корпус PDIP или SOIC;

– 16 программируемых входа-выхода.

Рабочее напряжение питания:

– 2.7 В до 5.5 В;

– ATtiny26L 4.5 В до 5.5 В.

Рабочая тактовая частота ровна от 0 до 8 МГц.

На рисунке 7 представлен микроконтроллер ATtiny26.



Рисунок 7 – Микроконтроллер ATtiny26

На рисунке 8 представлена распиновка контактов микроконтроллера ATtiny26.

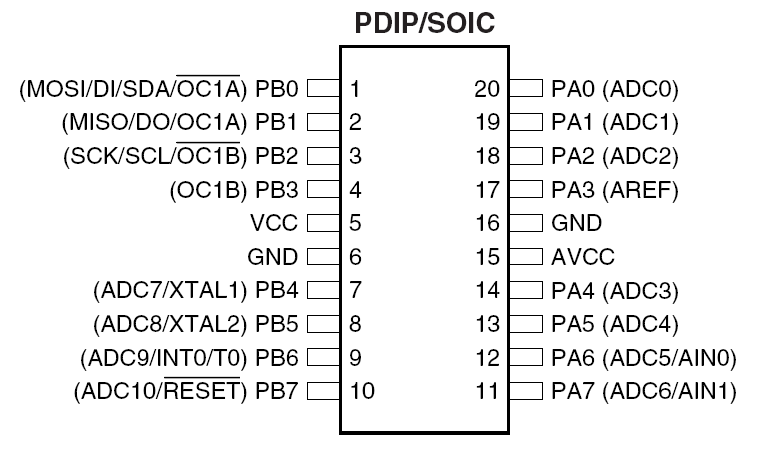


Рисунок 8 – Распиновка контактов микроконтроллера ATtiny26

ATmega328 является так же микроконтроллером семейства «megaAVR».

Микроконтроллер ATMega328 является 8-ми разрядным CMOS микроконтроллером с низким энергопотреблением, основанным на усовершенствованной AVR RISC архитектуре.

ATmega328/P — микроконтроллер семейства AVR, как и все остальные имеет 8-битный процессор и позволяет выполнять большинство команд за один такт.

Характеристика микроконтроллера:

Память:

32 kB Flash (память программ, имеющая возможность самопрограммирования);

2 kB ОЗУ;

1 kB EEPROM (постоянная память данных).

Периферийные устройства:

два 8-битных таймера/счетчика с модулей сравнения и делителями частоты;

16-битный таймер/счетчик с модулем сравнения и делителем частоты, а также с режимом записи;

счетчик реального времени с отдельным генератором;

шесть каналов PWM (аналог ЦАП);

6-канальный ЦАП со встроенным датчиком температуры;

программируемый последовательный порт USART;

последовательный интерфейс SPI;

интерфейс I2C;

программируемый сторожевой таймер с отдельным внутренним генератором;

внутренняя схема сравнения напряжений;

блок обработки прерываний и пробуждения при изменении напряжений на выводах микроконтроллера;

специальные функции микроконтроллера ATmega328:

сброс при включении питания и программное распознавание снижения напряжения питания;

внутренний калибруемый генератор тактовых импульсов;

обработка внутренних и внешних прерываний;

6 режимов сна (пониженное энергопотребление и снижение шумов для более точного преобразования АЦП);

Напряжения питания и скорость процессора:

1.8 — 5.5 В при частоте до 4 МГц;

2.7 — 5.5 В при частоте до 10 МГц;

4.5 — 5.5 В при частоте до 20 МГц;

На рисунке 9 представлен микроконтроллер ATmega328.



Рисунок 9 – Микроконтроллер ATmega328

На рисунке 10 представлена распиновка контактов микроконтроллера ATmega328.

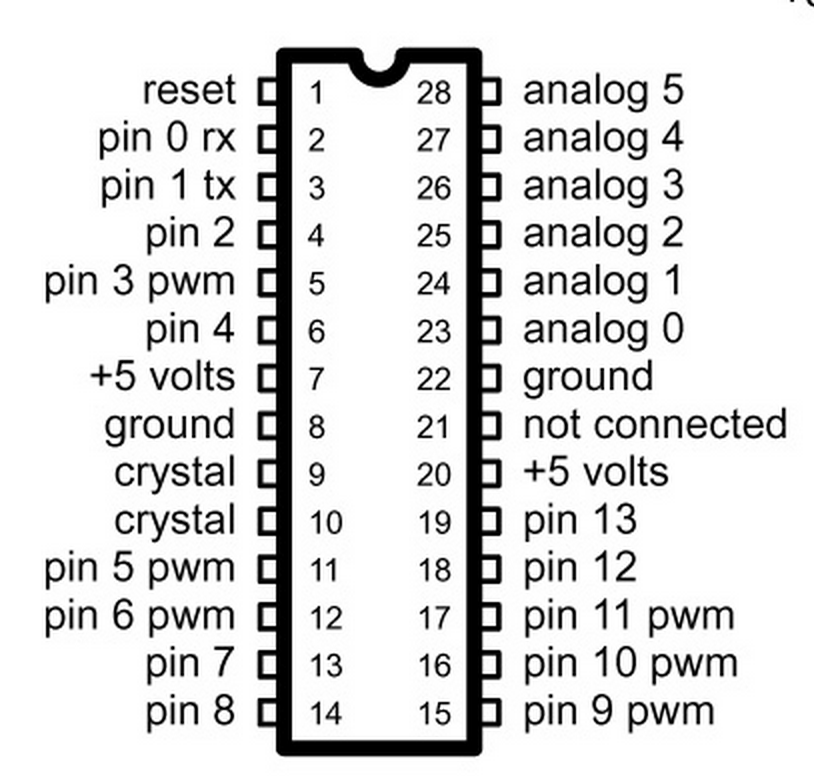


Рисунок 10 – Распиновка контактов микроконтроллера ATmega328

Исходя из описания характеристик, отзывов в интернете, доступности микроконтроллеров и наличие источников по урокам в работе с микроконтроллерами, было принято решение использовать для проекта микроконтроллер ATmega328.

4 Выбора элементной базы

Для вывода информации, необходимо использовать LCD дисплей или семисегментный дисплей. Так как дисплей имеет больше возможностей и является более дорогостоящим, было принято решение для прототипа устройства использовать семисегментный дисплей.

На рисунке 11 представлен семисегментный дисплей с модулем.

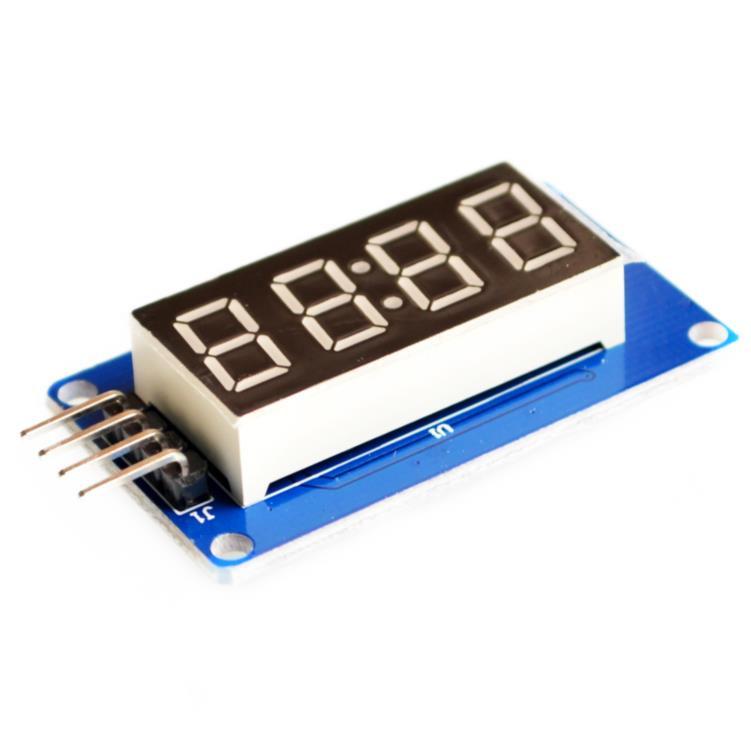


Рисунок 11 – Семисегментный дисплей с модулем

Дисплей выбран с модулем так, как все необходимые компоненты для работы дисплея уже собраны на самом модуле, что упрощает работу по сборке и программированию устройства.

Во избежание нагрева оборудования при увеличении мощности АКБ, необходимо использовать связку резисторов. Были выбраны резисторы 20 штук с сопротивлением равным 5 Ом и мощностью 2 Ватта.

Для включения тестера будет добавлена кнопка запуска.

Чтобы подать сигнал об окончании измерений, необходимо добавить спикер (пищалка).

Для управления зажимами в режиме измерений, необходимо добавить реле и, конечно же, сами зажимы.

Так же будет добавлено реле.

Сам же контроллер во избежание проблем с программированием было принято изменить прошивку на язык программирования «Arduino C».

Все данные устройства и компоненты способные работать при напряжении 5В. Так же каскад из 20 резисторов необходимы для защиты от нагревания всей схемы. Можно выбрать иные способы защиты от нагревания, например, система охлаждения, но по подсчётам, данный способ является самым выгодным из тех, что были найдены в интернете.

# 5 Описание выбора системы автоматизированного проектирования (САПР)

5.1 Обзор системы автоматизированного проектирования

Система автоматизированного проектирования – автоматизированная система, реализующая информационную технологию выполнения функций проектирования, представляет собой организационно-техническую систему, предназначенную для автоматизации процесса проектирования, состоящую из персонала и комплекса технических, программных и других средств автоматизации его деятельности. Также для обозначения подобных систем широко используется аббревиатура САПР.

Данная система позволяет спроектировать устройство, то есть наглядно изобразить схему устройства. Некоторые программы позволяют виртуально проверить схемы на работоспособность и правильность составления.

В качестве системы автоматизированного проектирования была выбрана среда «EasyEDA».

EasyEDA – кросс-платформенная среда автоматизации проектирования электроники включающая в себя редактор принципиальных схем, редактор топологии печатных плат, SPICE-симулятор, облачное хранилище данных, систему управления проектами, а также средства заказа изготовления печатных плат.

Данная среда позволяет работать в ней как через браузер, так и в качестве приложения, которое можно скачать на устройство.

Среда является бесплатной, а также все её возможности, кроме заказа печатных плат – они платные.

Состав среды EasyEDA:

– редактор принципиальных схем;

– редактор печатных плат;

– SPICE-симулятор;

– редактор электронных компонентов;

– генератор файлов формата GERBER;

– программа просмотра файлов формата GERBER;

– система управления проектами;

– система заказа изготовления печатных плат;

– облачное хранилище файлов.

Среда работает на основе клиент-серверной модели. Клиентская часть приложения полностью выполняется в браузере, поддерживающим HTML5. Инсталляция каких-либо программ или плагинов не требуется. Графическая среда приложения использует движок векторного графического формата SVG, доступный в современных браузерах. Хранение файлов осуществляется на сервере. Система никак не привязана к файлам пользователя на локальном компьютере и позволяет в любой момент продолжить работу на любом компьютере, подключенном к сети Интернет.

Редактор электрических схем, компонентов и готовых модулей с обширной автоматически обновляемой библиотекой, содержащей сотни тысяч комплектующих, символы компонентов как в американском, так и в привычном нам, европейском форматах.

В данной среде можно как создавать собственные компоненты и модули, так и редактировать существующие трассировщик, редактор топологии (проводящего рисунка) печатных плат симулятор схем (модули ESP8266, конечно же, не поддаются симуляции), движок симулятора от ngspice просмотрщик файлов формата Gerber изготовление печатных плат по ценам (9,8$ за 10шт. +доставка в РФ – 6,6$ (получается 1,64$ за двухстороннюю плату размером 50×50мм с учетом доставки.

Заказ плат на EasyEDA не является обязательным условием использования этого сервиса и вполне можно либо не заказывать платы вообще, либо заказать их у любого изготовителя по вашему выбору (есть бесплатный экспорт плат в формате Gerber).

Сервис бесплатный с возможностью получить более быструю техподдержку.

Для подавляющего большинства пользователей вполне будет достаточно возможностей, предоставляемых на бесплатном тарифе.

В настоящий момент EasyEDA русифицирован примерно на 90%, что позволяет вполне комфортно работать людям, предпочитающим локализованные версии.

5.2 Элементная база выбранной системы автоматизированного проектирования

5.2.1 Интерфейс среды САПР EesyEDA.

Фильтр.

Перед использованием фильтра нужно выбрать нужный модуль на левой навигационной панели, а затем можно быстро и легко найти проекты, файлы, детали и посадочные места, просто набрав несколько букв заголовка. Например, если необходимо найти все файлы, содержащие «NE555» в заголовке, просто нужно ввести «555», без учета регистра. На рисунке 12 представлен пример.

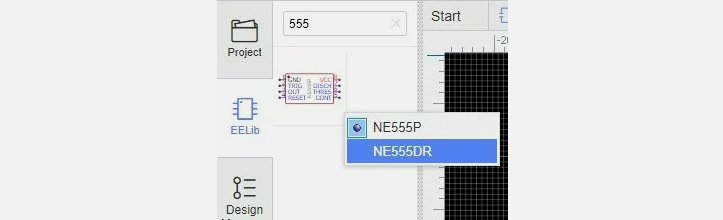


Рисунок 12 – Пример поиска компонента

Фильтр ищет только заголовки и названия проектов, файлов и деталей. Он не выполняет поиск в полях «Описание» и «Содержание».

Панель навигации.

Панель навигации очень важна для EasyEDA, именно здесь можно найти все свои проекты, файлы, детали и элементы footprint, пример приведён на рисунке 13.

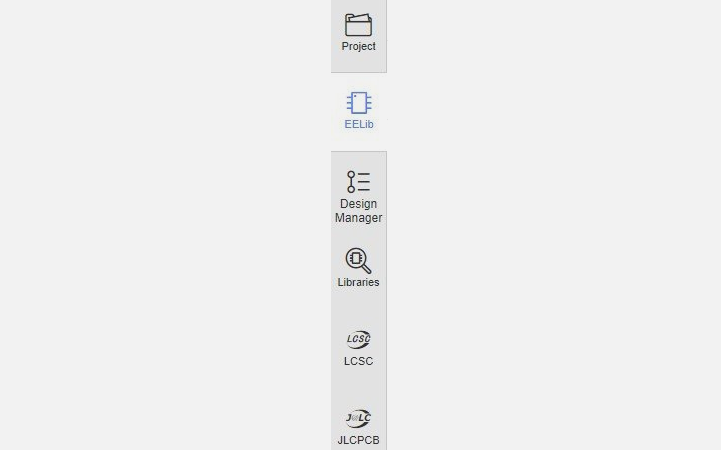


Рисунок 13 – Панель навигации

**Проект -** здесь, вы можете найти все личные проекты, которые являются частными или общедоступными, или разрабатываются чужими.

За исключением System IC, эти параметры имеют меню содержимого. Например, если перейти в «Мои проекты» и щелкнуть правой кнопкой мыши на элементе, вы получите древовидное меню, пример приведён на рисунке 14.

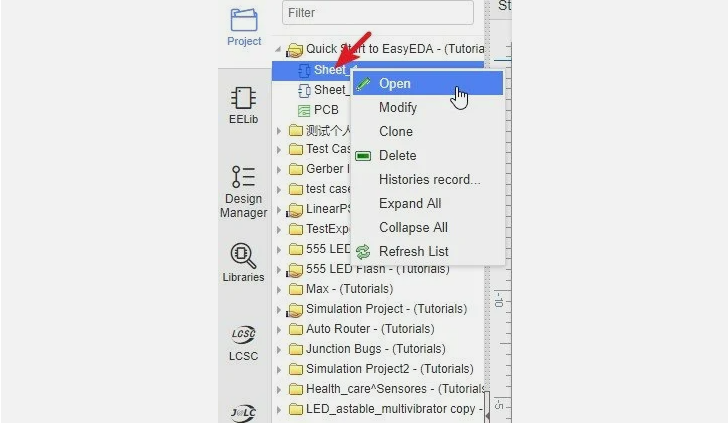


Рисунок 14 – Пример поиска своего проекта

**EELib.**

EElib означает «Библиотеки EasyEDA». Он содержит множество компонентов с имитационными моделями, многие из которых были разработаны для EasyEDA, чтобы упростить процесс моделирования.

**Design Manager.**

Менеджер дизайна, где можно легко проверить каждый компонент и сеть, и он обеспечит проверку правил проектирования, чтобы помочь вашему дизайну, как представлено на рисунке 15.

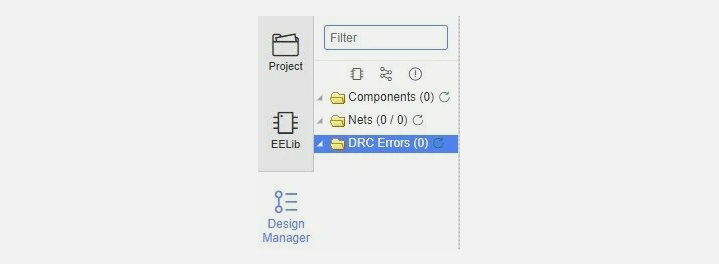


Рисунок 15 – Менеджер дизайна

**Libraries (Библиотеки).**

Содержит условные обозначения и контуры печатной платы для многих легкодоступных компонентов и проектов. Все библиотеки и модули появятся здесь, пример приведён на рисунке 16.

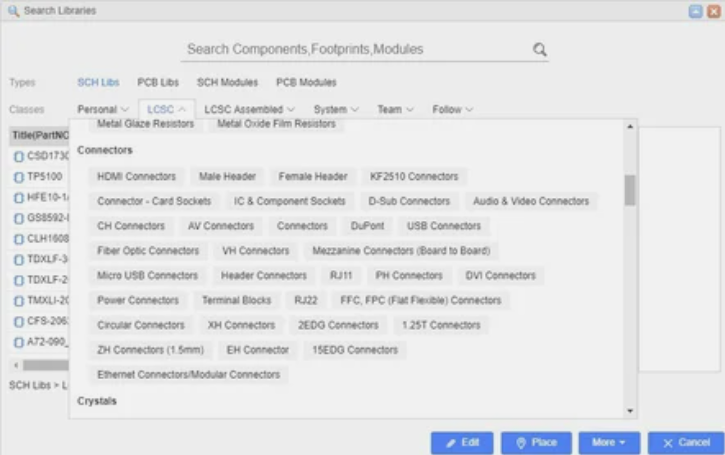


Рисунок 16 – Библиотеки

**LCSC (Love Components Save Cost).**

Если необходимо купить дополнительные компоненты и закончить свою печатную плату, компания EasyEda предлагает приобрести модуль LCSC. «LCSC.com» и «EasyEDA» одна и та же компания.

**JLCPCB.**

Компания по производству печатных плат, специализируется на быстром прототипе печатной платы и мелкосерийном производстве. Доступные, качественные платы серийного производства, полностью изготовленные в Китае.  
JLCPCB.com, LCSC.com и EasyEDA – это **одна и та же** группа компаний.

Атрибуты холста.

На правой части рабочей панели представлено меню «Атрибуты холста», пример приведён на рисунке 17. Можно настроить цвет фона и сетки, а также стиль, размер, видимость и атрибуты сетки. Область холста может быть установлена ​​непосредственно шириной и высотой или из доступных предустановленных размеров.

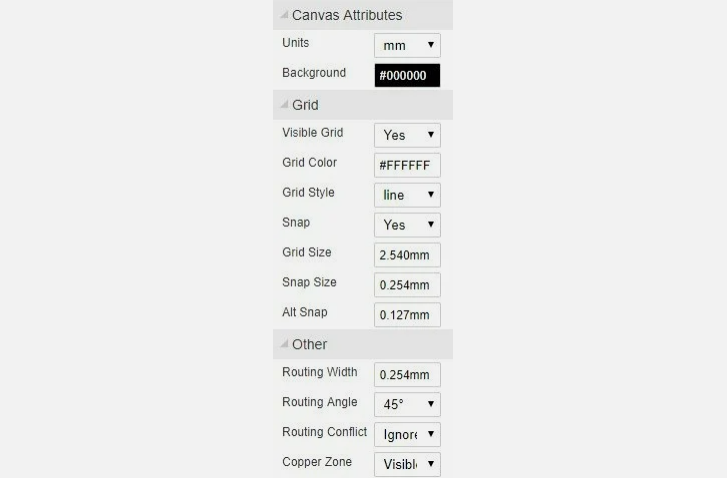
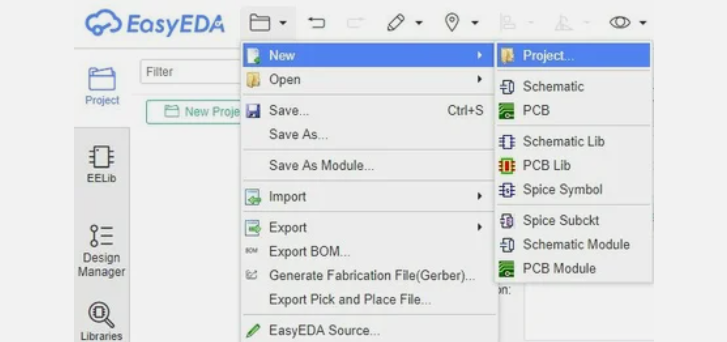


Рисунок 17 – Атрибут холста

Создание нового проекта.

После регистрации или входа в систему можно приступить к созданию нового проекта.

**«Документ > Создать > Создать новый проект / Схема / Печатная плата», как представлено на рисунке 18.**



**Рисунок 18 – Создание нового проекта**

Концепция проекта важна в EasyEDA, потому что она является основой того, как организовать ваши проекты, поэтому при создании проекта, необходимо дополнить его информацией, как представлено на рисунке 19.

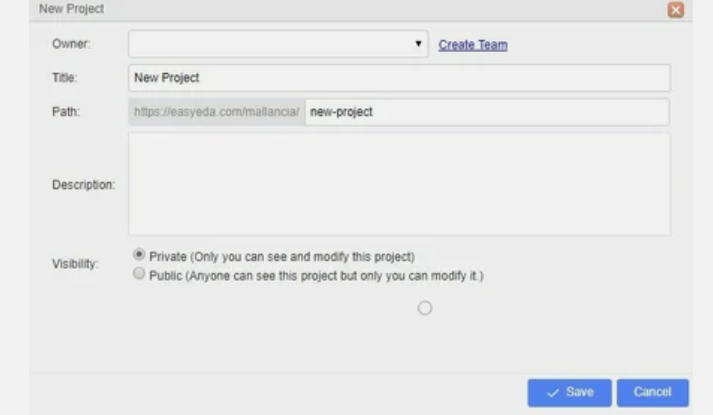


Рисунок 19 – Информация о проекте

**Owner (Владелец)** – можно изменить владельца этого проекта.

**Title (Заголовок)** – обозначить заголовок, он будет отображаться в дереве проекта на левой панели.

**Path (Путь)** – EasyEDA позволяет указать путь к проекту, если необходимо поделиться с кем-то, это будет полезно. Путь не может быть изменен после создания проекта.

**Visibility (Видимость)** – можно сделать свой проект общедоступным или закрытым, установив его «Видимость».

Если решите сделать ваш проект общедоступным, «**Категории»** позволяют выбрать категорию, под которой необходимо разместить свой проект на веб-сайте. Если сохраняете свой проект закрытым, категория по-прежнему применяется, но не имеет прямого использования при сортировке проектов, поскольку это поле не ищется в поле «Фильтр» на левой панели.

**Description (Описание)** – добавление краткого описания поможет.

Чтобы изменить проект после его создания, необходимо щелкнуть правой кнопкой мыши по нему в дереве проекта на левой панели и выбрать поле***«***Edit (Редактирование)», как представлено на рисунке 20.

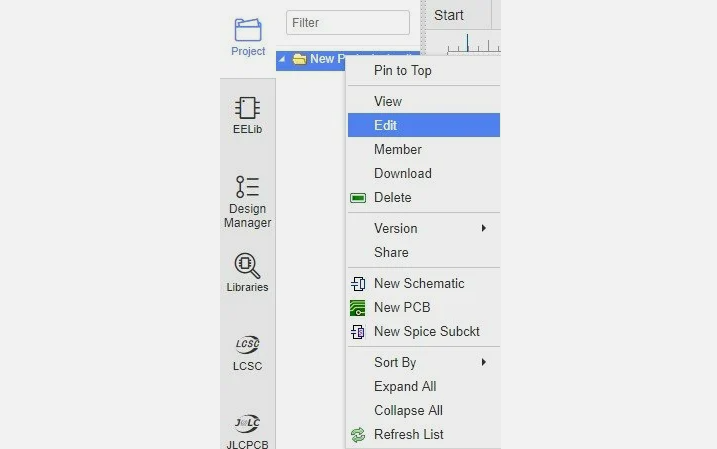


Рисунок 20 – Редактирование

Затем откроется веб-страница, на которой можно редактировать свой проект, как представлено на рисунке 21.



Рисунок 21 – Страница редактирования проекта

Отсюда можно изменить видимость, позволить другим людям комментировать проект и ввести более подробное описание содержания проекта. Чтобы помочь выделить свой проект или просто сделать подробное описание проекта более понятным, можно использовать синтаксис Markdown.

Создание схемы.

С помощью EasyEDA можно создавать высокопрофессиональные схемы.

Поскольку EasyEDA имеет некоторые простые, но мощные возможности рисования, можно создавать собственные символы, либо копируя существующие символы в собственную библиотеку, а затем редактируя и сохраняя их, либо рисуя их с нуля.

Существует также **Schematic Library Wizard** (**Мастер Схематической библиотеки)** для быстрого рисования новых символов для DIP, QFP и SIP.

Особенностью EasyEDA является то, что помимо обширных библиотек обычных «2D» графических схематических символов, она имеет библиотеку «3D» компонентов, то есть символов, которые выглядят как физические компоненты, которые они представляют.

Другая характерная особенность в том, что также можно импортировать символы из Kicad, Eagle и Altium библиотек.

Создание печатной платы.

Когда будете удовлетворены результатами своей схемы и моделирования, вы сможете быстро приступить к изготовлению готовой и заполненной печатной платы, не покидая EasyEDA.

Холст EasyEDA для печатных плат помогает быстро и легко разметить даже сложные многослойные конструкции на основе схем, которые вы уже создали на холсте схемы, или непосредственно в виде макета без схемы.

Передача EasyEDA Schematic в редактор PCB Design проста: просто нажать на значок «**Convert to PCB»** на верхней панели инструментов, как представлено на рисунке 22.

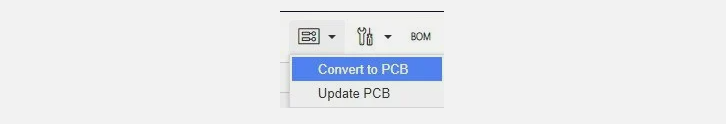


Рисунок 22 – Передача EasyEDA Schematic в редактор PCB Design

EasyEDA имеет обширные библиотеки футпринтов. Также можно создать собственную библиотеку необычных и специализированных деталей, копируя и изменяя существующие детали, или с нуля, используя инструменты создания и редактирования EasyEDA.

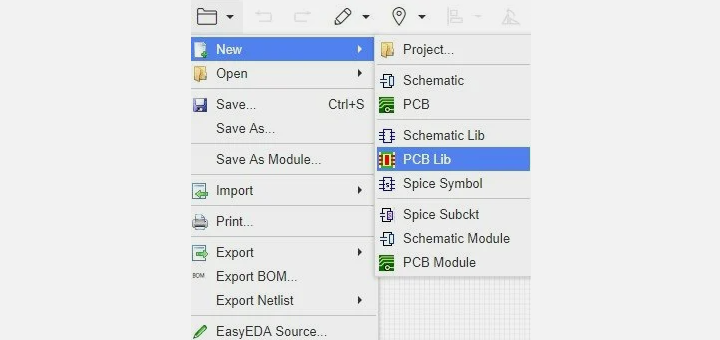


Рисунок 23 – Библиотеки футпринтов

Аналогично тому, как на холсте «Схема», чтобы помочь вам находить элементы и ориентироваться при работе на холсте «Печатная плата», существует менеджер дизайна. **Левая панель навигации > Design Manager** (**Менеджер дизайна).**

Менеджер дизайна печатных плат представляет собой инструмент для поиска компонентов, дорожек (сетей) и площадок (сетевых площадок). 

Нажмите на любой элемент, выделите компонент и переместите его в центр окна.

Ширина дорожек, зазоры и размеры отверстий можно настроить в диалоговом окне «Проверка правил проектирования», которое открывается с помощью: **Настройка> Настройка правил проектирования.**Начиная с первой настройки проверки правил проектирования (**DRC**) в начале макета платы, выполняется DRC — это почти последний шаг в проверке дизайна вашей печатной платы, прежде чем вы создадите файлы **Gerber** и **Drill** для изготовления платы.

Последний шаг — проверка файлов **Gerber** и **Drill** с помощью простого и открытого программного обеспечения Gerber Viewer.

# 6 Описание технической реализации устройства

6.1 Разработка принципиальной схемы в выбранном САПРе

Удобство среды САПР EasyEDA по мимо выше перечисленных, можно считать и то, что в среде находится большой количество устройств и компонентов, а если чего-то может и не быть, любой пользователь может добавить сам.

Так, например, все устройства необходимые для реализации данного дипломного проекта имеются в среде EasyEDA.

На рисунке 24 представлена принципиальная схема разрабатываемого устройства.

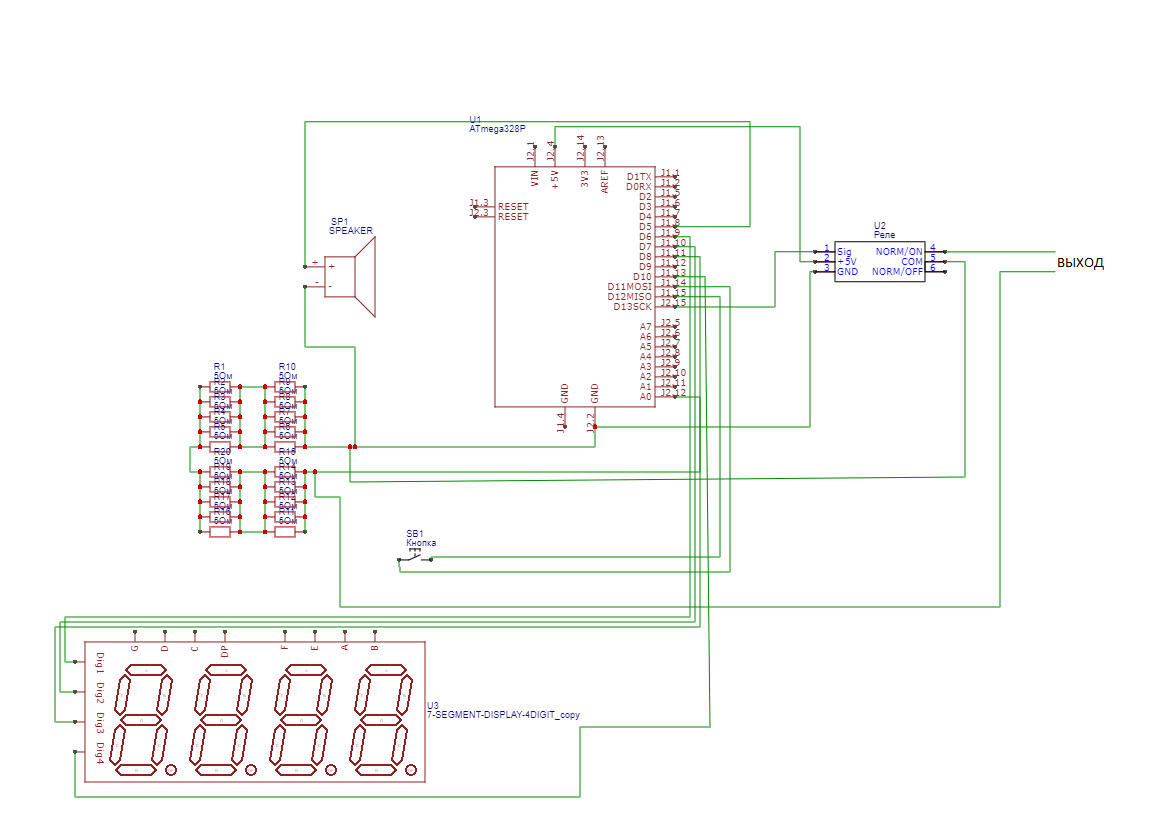


Рисунок 24 – Схема измерителя ёмкости

Выход – обозначено место подключения, измеряемого АКБ.

6.2 Разработка электрической схемы в выбранном САПРе

На рисунке 25 представлена электрическая схема микроконтроллера ATmega328.

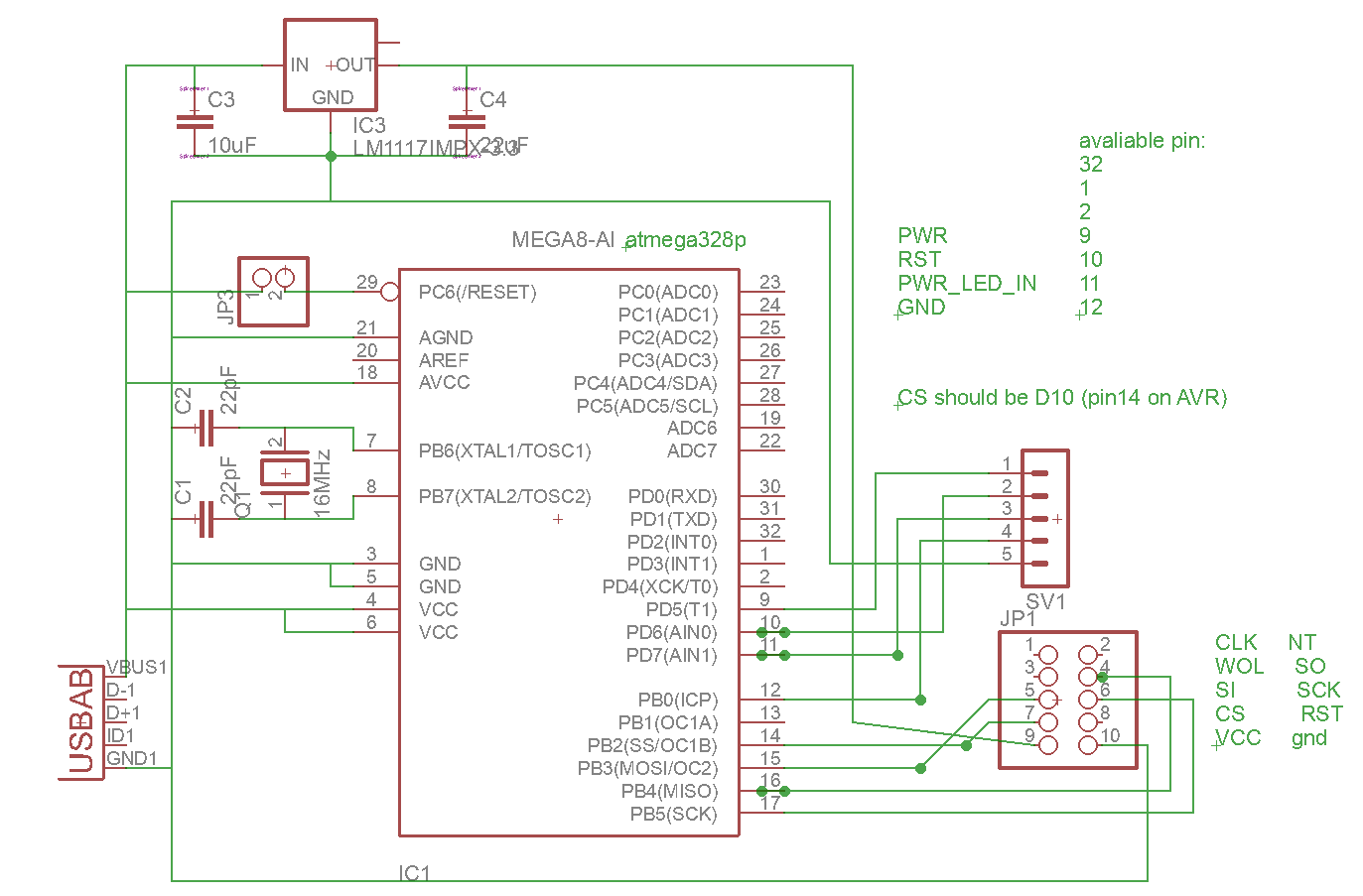


Рисунок 25 – Электрическая схема ATmega328

# 7 Описание сборки и настройки устройства

7.1 Описание сборки устройства

Перед сборкой любого устройства, необходимо протестировать устройство на макетной плате. Для этого будут необходимы все компоненты устройства, а также макетная плата и провода.

Сборка на макетной плате производится согласно принципиальной схеме.

После сборки устройства и отладки программы, можно приступать к монтажной работе. Для начала для простоты и удобства было принято решение сборки навесным монтажом. Результат сборки навесным монтажом представлен на рисунке 26.



Рисунок 26 – Результат прототипа сборки навесным монтажом

Для удобства в сборки вместо самого микроконтроллера, была выбрана плата Arduino micro, так как эта платформа строится на микроконтроллере Atmega328, что значительно облегчит настройку и отладку схемы на макетной и печатной платах, сам же контроллер сложнее запрограммировать, чем платформу Arduino.

Результат измерений при навесном монтаже был весьма точен, но имел некоторую погрешность, по сравнению с мультиметром.

Стоит отметить, что заряд кроны мультиметра так же не был максимален, что могло повлиять на качества измерений АКБ и с мультиметра.

После тестирования, было принято решение демонтажа схемы и монтаж на печатной плате.

Разводка для печатной платы представлена на рисунке 27.

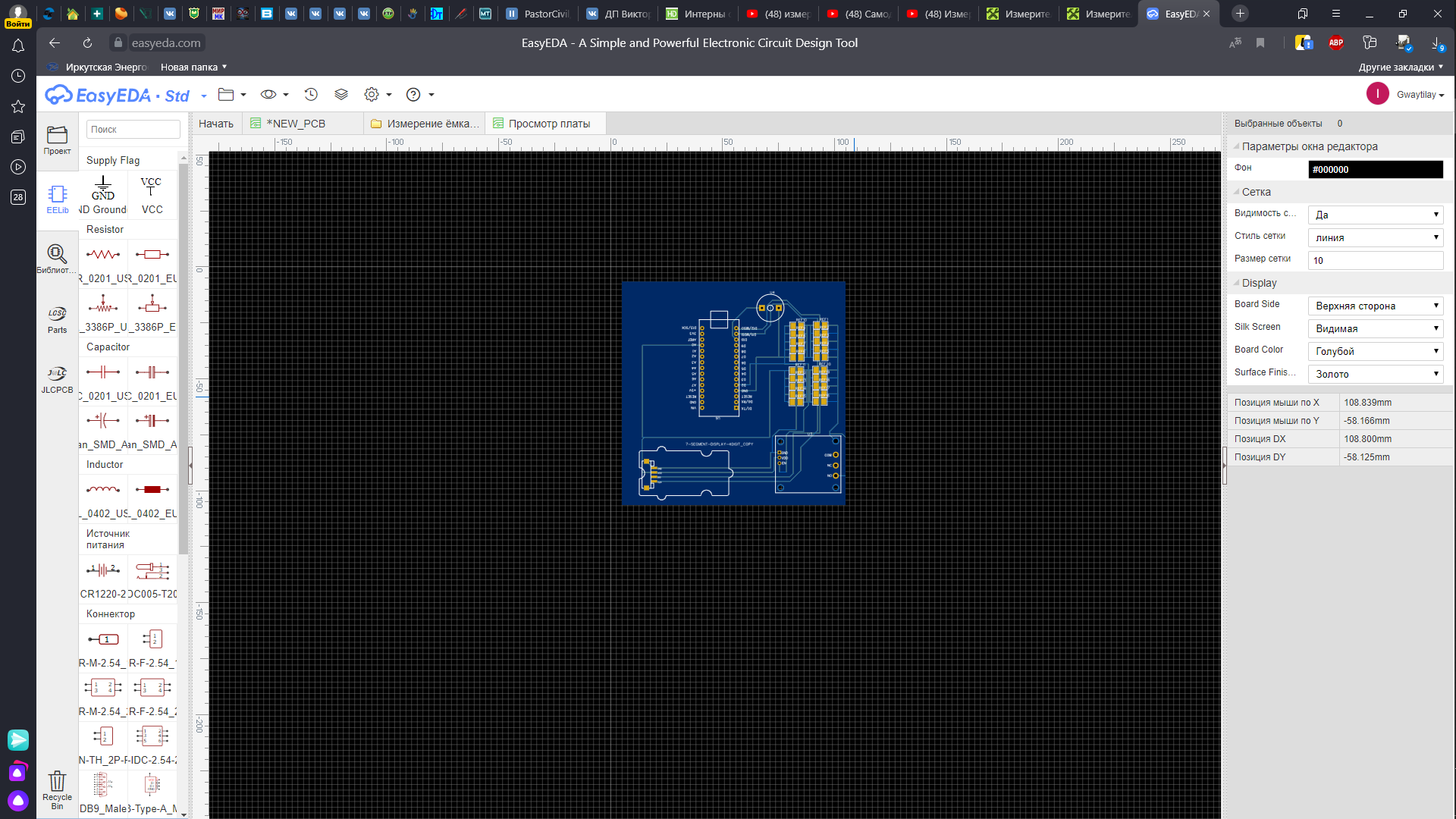


Рисунок 27 – Разводка на печатной плате для проекта

7.2 Описание настройки устройства в его функциональной особенности

Вся настройка данного проекта производилась на стадии тестирования на макетной плате, так как именно на ней удобнее всего калибровать все модули, тестировать программный код и искать альтернативные решения некоторых сложностей.

Сам программный код представлен в «Приложение А – Листинг AKB».

Так же для работы программы, была подключена дополнительная библиотека, которая представлена в «Приложение Б – Библиотека AKB».

7.3 Эскиз печатной платы и проекта

Так же в среде EasyEDA есть возможность просматривать печатную плату в 3D формате, что является большим преимуществом среди аналогичных программ.

На рисунке 28 представлен эскиз печатной платы для измерителя АКБ.

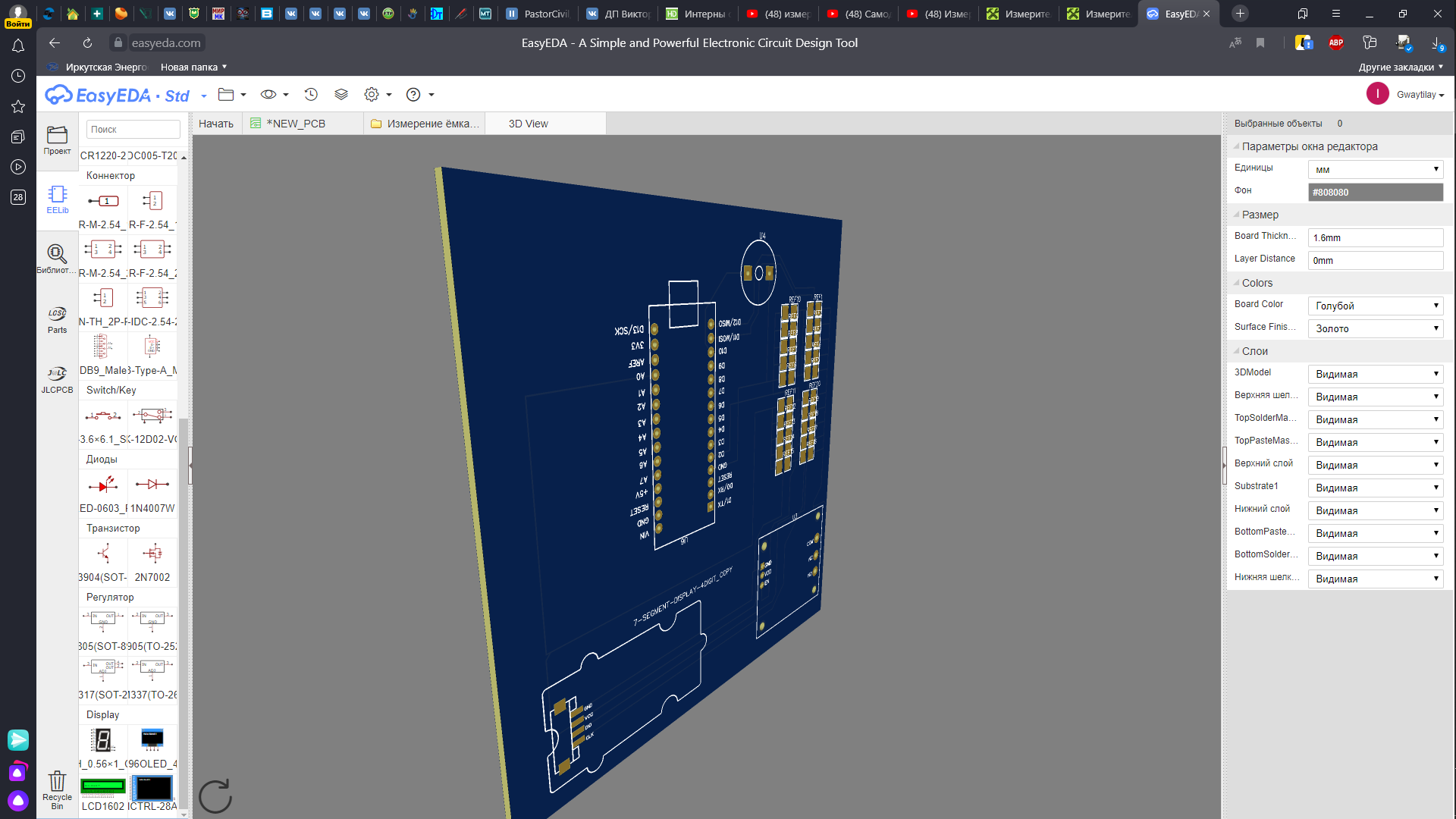


Рисунок 28 – Эскиз печатной платы

# 8 Описание тестирования и отладки платы

8.1 Составление алгоритма тестирования

После сборки и загрузки программы на микроконтроллер, можно приступать к тестированию устройства.

Перед началом измерений, стоит отметить, что данное устройство можно подключать только к АКБ с напряжением ниже, чем 5В, так как напряжение выше выведет устройство из работоспособности.

Для начала тестирования, необходимо включить устройство, подав на микроконтроллер питание в 5В.

После чего пройдёт загрузка устройства в течении 3 секунд и на дисплеи будут светиться четыре нуля.

Для удобства измерений, были припаяны два крокодильчика красным и чёрным цветами соответственно своей полярности.

Для выведения показаний на устройстве, необходимо прижать контакты к полярности АКБ в соответствии с цветовой маркой, где красный цвет является плюсом, а чёрный минусом.

8.2 Инструменты для аппаратного тестирования

В качестве устройств для проверки дипломного проекта будет использоваться мультиметр.

После проверки устройства измерителя ёмкости, необходимо снять показания с дисплея и провести те же измерения мультиметром и сравнить их.

Так же для программного тестирования можно использовать среду программирования Arduino IDE, так как там имеется монитор порта, с помощью которого можно отслеживать данные с портов схемы.

8.3 Проведение тестирования и его результаты

Для разработанного устройства в тестирование были выбраны четыре аккумуляторные батареи.

На рисунке 29 представлен образец первой АКБ, которая была выбрана для тестирования.



Рисунок 29 – Образец АКБ

На рисунке 30 представлен образец второй АКБ, которая была выбрана для тестирования.



Рисунок 30 – Образец АКБ

На рисунке 31 представлен образец третий АКБ, которая была выбрана для тестирования.

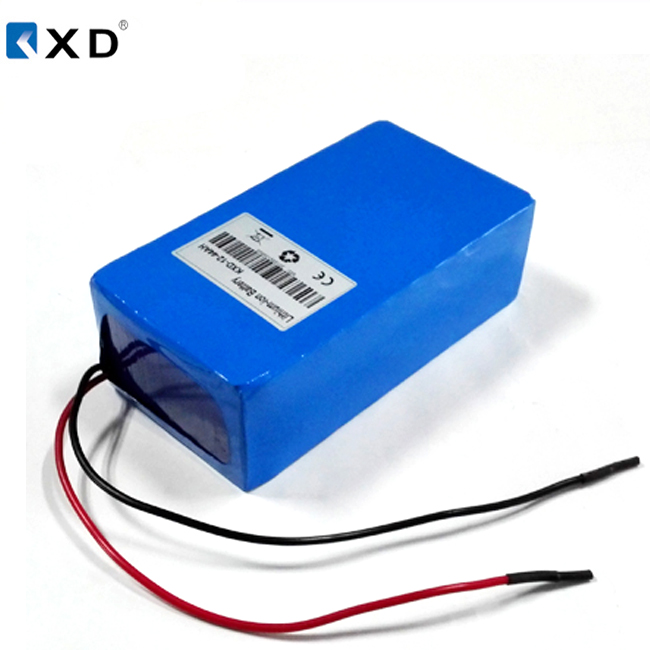


Рисунок 31 – Образец АКБ

На рисунке 32 представлен образец четвёртый АКБ, которая была выбрана для тестирования.

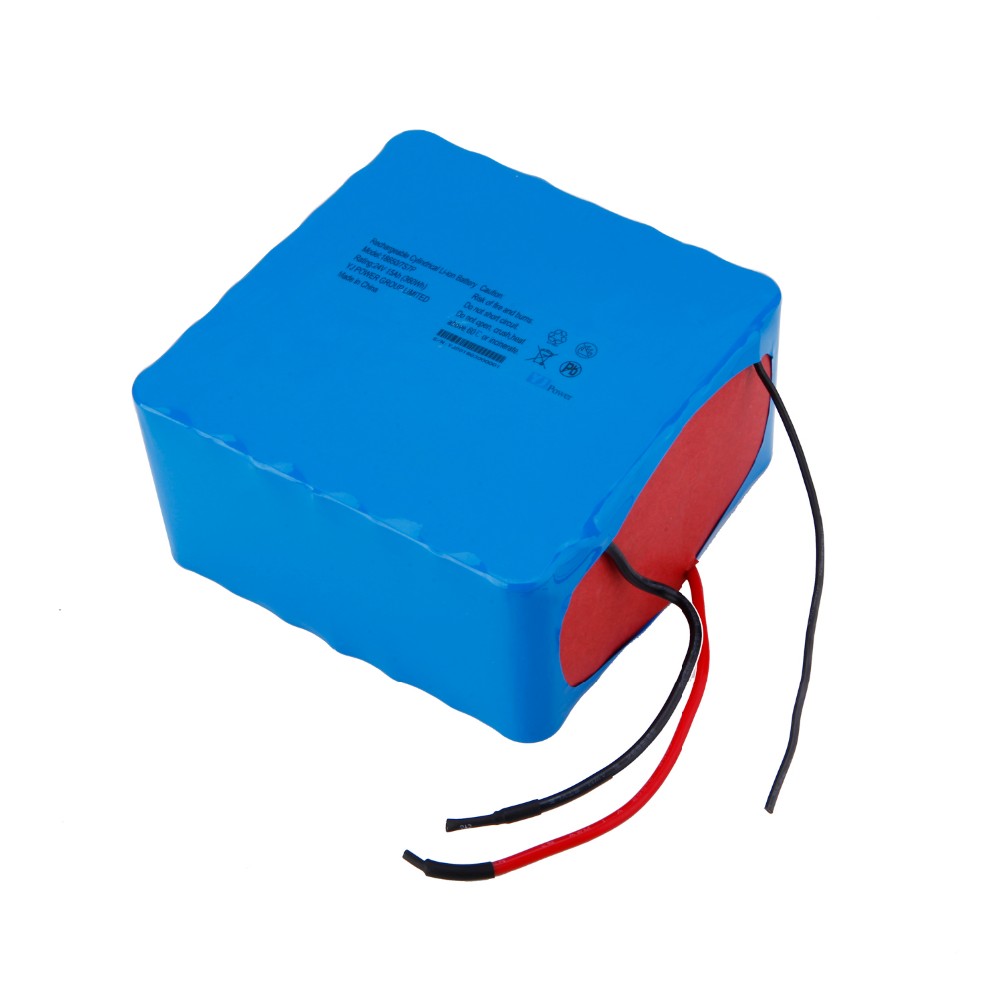


Рисунок 32 – Образец АКБ

В таблице 1 представлена сравнительная характеристика с результатов измерителя ёмкости и документационным номиналом АКБ.

|  |  |  |
| --- | --- | --- |
| Наименование | Номинал | Результат |
| Литиевый АКБ 3.7V 110 mah | 110 mah | 108 mah |
| Литиевый АКБ 3.7V 800 mah | 800 mah | 691 mah |
| Литиевый АКБ 3.7V 8000 mah | 8000 mah | 7911 mah |
| Литиевый АКБ 3.7V 10000 mah | 10000 mah | 8917 mah |

После снятий показаний с специализированного устройства для определения ёмкости АКБ, была выявлена погрешность с разрабатываемого устройства почти в 6% от общей суммы показаний, что является относительно высокой погрешностью, но дальнейшая калибровка не допустима с данным устройством.

Есть смысл переработки устройства и замена деталей на более дорогостоящие и профессиональные, что поспособствует снижению процента погрешности и увеличению правильности измерений.

# 9 Экономическая часть

9.1 Организационно–экономическое обоснование проекта

Экономически данный проект является самым бюджетным проектом, созданный в домашних условиях относительно всех других устройств, что можно найти в среде интернета, а также относительно высокоточным среди них.

К сожалению, данное устройство уступают по точности профессиональным, но стоит заметить, что профессиональные устройства собираются на заводах и в лабораториях, а не в домашних условиях.

9.2 Расчёт себестоимости

Данный раздел содержит расчёт оборудования, необходимый для создания проекта в количестве одной штуке.

Стоимость оборудование представлено в таблицы 2.

Таблица 2 – Стоимость используемого оборудования

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование оборудования | Закупочная цена (руб.) | Количество (шт) | Стоимость (руб.) |
| Ноутбук Lenovo IdeaPad V340-17IWL | 34800 | 1 | 34800 |
| Arduino mini | 120 | 1 | 120 |
| Радиоэлектронные компоненты | 350 | 1 | 350 |
| Транспортные расходы (3 %) | | | 1060 |
| Итого | | | 36420 |

Программное обеспеченье для реализации проекта имеется в общем доступе в интернете бесплатно.

Стоить отметить, что Ноутбук Lenovo IdeaPad V340-17IWL не потребуются для создания последующих копий данного проекта, так как он не является расходным материалом в данном проекте.

В таблице 3 представлен расчёт калькуляции расходным материалов.

Таблица 3 – Калькуляция расходных материалов

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование | Кол-во, шт. | Цена ед., руб. | Сумма, руб. |
| Тактовая кнопка | 1 | 100 | 100 |
| Спикер | 1 | 20 | 20 |
| Модуль реле | 1 | 60 | 60 |
| ATmega328 | 1 | 100 | 100 |
| Крокодильчик | 2 | 10 | 20 |
| Дисплей | 1 | 180 | 180 |
| Резистор 5Ом 2Вт | 20 | 10 | 200 |
| Текстолит | 1 | 220 | 100 |
| Припой | 1 | 350 | 15 |
| Флюс | 1 | 50 | 5 |
| Раствор для травки печатной платы | 1 | 80 | 100 |
| Итого | 31 | 1180 | 900 |

Для монтажа печатной платы будут необходимы все выше перечисленные материалы, но припой, текстолит и флюс понадобятся в маленьком количестве, поэтому они растрачены не полностью

На рисунке 33 представлена круговая диаграмма примерных затрат на создание проекта.

Рисунок 33 – Круговая диаграмма затрат оборудования и расходных материалов на проект

9.3 Расчёт затрат на электроэнергию и амортизацию оборудования

Затраты на электроэнергию рассчитываются по следующей формуле:

Е = 𝑊 × 𝑡 × 𝑇, (1)

где 𝑊 – мощность, потребляемая ПК, кВт/час.;

𝑡 – время работы ПК, дн.; 𝑇 – тариф электроэнергии, руб.

Затраты на электроэнергию представлены в таблице 4.

Таблица 4 – Затраты на электроэнергию

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Оборудование | Время эксплуатации (час) | Количество | Мощность (кВт/час) | Тариф на электроэнергию (руб./кВт/ч) | Затраты на электроэнергию (руб.) |
| Ноутбук | 60 | 1 | 0,1 | 1,11 | 6,6 |
| Паяльное оборудование | 10 | 1 | 0,4 | 1,11 | 4,4 |
| Итог | | | | | 11 |

Амортизация рассчитывается по следующей формуле:

𝐴 = 𝑆×𝑞𝑎𝑚 12 , (2)

где 𝑆 – первоначальная стоимость ПК, руб.;

𝑞𝑎𝑚 – процент амортизации в год.

Процент амортизации для группы основных средств «Вычислительная техника и периферийные устройства» в среднем принимается 30% в год. Амортизация равна 9060 рублей.

Трудоемкость работы характеризуется перечнем основных этапов и видов работ, которые были выполнены в проекте.

Оплата труда рассчитывается исходя из часовой тарифной ставки и затрат времени на работу по формуле:

ЗП = Ст1 ∗ Фвр, (3)

где Ст1 – часовая тарифная ставка (для специалиста техника по компьютерным системам);

Фвр – фонд фактического времени, затраченного на ремонт/ обслуживание, час (в таблице это трудоемкость). Расчет представлен в таблице 7.

Средняя заработная плата в Иркутске составляет 109 рублей в час. Учитывая престижность данной работы, оплата труда была поднята до 150 рублей за час.

Таблица 5 – Расчёт оплаты труда

|  |  |  |
| --- | --- | --- |
| Виды работы | Часовая тарифная ставка руб/ч | Трудоёмкость, ч. |
| Поиск информации в интернете | 150 | 4 |
| Настройка ПК | 150 | 4 |
| Составление схем | 150 | 2 |

Продолжение таблицы 5

|  |  |  |
| --- | --- | --- |
| Монтажная работа | 150 | 8 |
| Составление программного кода и отладка | 150 | 6 |
| Тестирование | 150 | 2 |
| Описание проделанной работы | 150 | 2 |
| Итого (эффективный фонд времени) | | 20 |
| Заработная плата (рублей) | | 4200 |

9.4 Расчет начислений на заработную плату

Начисления на заработную плату за полную готовую систему, в зависимости от категории плательщика, указанных в ФЗ № 212–ФЗ, рассчитываются по ставкам, представленных в таблице 6.

Таблица 6 – Начисления на заработную плату

|  |  |  |
| --- | --- | --- |
| Начисления на заработную плату | Процент, % | Сумма, руб. |
| Пенсионный фонд (ПФ):  – страховая часть  – накопительная часть | 16  6 | 672  252 |
| Фонд социального страхования (ФСС) | 2,9 | 121,8 |
| Федеральный фонд обязательного медицинского страхования (ФФОМС) | 5,1 | 214,2 |
| Итого | 30 | 1260 |

9.5 Расчет общей себестоимости проекта

Начисления на заработную плату, в зависимости от категории плательщика, указанных в ФЗ № 212-ФЗ.

На основании полученных расчетов затрат, определить себестоимость проекта. Статья затрат представлена в таблице 7.

Таблица 7 – Расчёт себестоимости проекта

|  |  |
| --- | --- |
| Наименование статей затрат | Сумма, руб. |
| Стоимость оборудования | 36420 |
| Расходные материалы | 900 |
| Стоимость электроэнергии | 11 |
| Заработная плата | 4200 |
| Налог | 1260 |
| Итого | 42791 |

Стоимость проекта будет установлена в сумме 1500 рублей, учитывая альтернативные разработки на рынке.

Данные для расчета заработной платы:

— среднее количество дней в месяце – 22;

— рабочий день – 8 часов;

— оклад рабочего – указывается в среднем заработная плата данных работников.

Расчет экономического эффекта

Экономическим эффектом (выгодой) является предполагаемая прибыль от реализации созданной разработки (программного продукта):

Предполагаемая прибыль = Доход – Затраты.

Учитывая, что на одно устройство работник буде тратить 20 часов, в месяц он будет производить 8.8 устройств. Месячный доход сотрудника равен 26400 рублей.

Доход с устройств будет равен 12000 рублей, так как 8 устройств по 1500 рублей будут производиться за определённый отрезок времени.

Чистый доход составит примерно 6900 рублей, если учесть все затраты.

Заключение

В заключении необходимо отметить, что разработка такого устройства является актуальным и оригинальным способом выйти на рынок с совершенно новым устройством, который будет состоять вне конкуренции из-за стоимости аналогов на рынке. Так же стоит отметить, что устройству необходима доработка и создание корпуса, что так же повлечёт за собой новые затраты, что в свою очередь повысит стоимость устройства на рынке, но тем наименее, устройство по-прежнему будет являться дешевле аналогов.

Общие затраты на разработку проекта составили 42791 рублей. Из них 900 рублей – стоимость материалов для прототипа устройства. Эта сумма вполне оправдывает себя, так как стоимость подобных устройств начинается от 5000 рублей.

Практической реализацией устройства является бытовое применение радиолюбителей, сервисных центров, учебные учреждения среднего профессионального образования с уклоном на электронику.

Все поставленные задачи дипломного проекта, связанные с анализом, проектированием и практической реализацией системы выполнены, цель достигнута.

Список использованных источников

1 Аккумуляторы. Справочник. Андрей Кашкаров. 2017 год.

2 Исследование методов диагностики аккумуляторных батарей. Д. П. Чупин. 2012 год.

3 Литий-ионные и литий-полимерные аккумуляторы. Владимир Василье. 2011 год.

4 Аккумуляторы в мире портативных устройств. Руководство по аккумуляторам для неинженеров. Перевод Владимира Васильева.

5 Радио кот [Электронный ресурс]. Режим доступа - https://www.radiokot.ru/forum/viewtopic.php?t=2793 (Дата обращения: 11.05.20).

6 АМПЕРОФ [Электронный ресурс]. Режим доступа - https://amperof.ru/sovety-elektrika/proverit-emkost-akkumulyatora-multimetrom.html3 (Дата обращения: 16.05.20).

7 Яндекс. Дзен. [Электронный ресурс]. Режим доступа - https://zen.yandex.ru/media/hyperu/kak-proverit-emkost-akkumuliatora-telefona-5bea944935713d00aa21d528 (Дата обращения: 16.05.20).

8 Как измерить емкость аккумулятора мультиметром: пошаговая инструкция - [Электронный ресурс]. Режим доступа - https://fb.ru/article/302717/kak-izmerit-emkost-akkumulyatora-multimetrom-poshagovaya-instruktsiya (Дата обращения: 18.05.20).

9 ТехноСова [Электронный ресурс]. Режим доступа - https://technosova.ru/cifrovaja-tehnika/akkumulyatory/emkost-kak-izmerit/ (Дата обращения: 11.05.20).

10 Drive2.ru [Электронный ресурс]. Режим доступа - https://www.drive2.ru/l/488990480218980830/ (Дата обращения: 18.05.20).

# Приложение А – Листинг AKB

#define NUM\_READS 100

#include "TM1637.h" //библиотека дисплея

byte load\_pin=0; //пин нагрузки (аналоговый!!!!)

byte buzz\_gnd=2; //динамик земля

byte buzz\_pin=5; //динамик сигнал

byte disp\_gnd=6; //земля дисплея

byte CLK = 7; //пин дисплея

byte DIO = 8; //пин дисплея

byte disp\_vcc=10; //питание дисплея

TM1637 disp(CLK,DIO); //обозвать дисплей disp

byte butt\_pin=11; //состояние кнопки

byte butt\_gnd=12; //земля кнопки

byte relay\_pin=13; //пин реле

const float typVbg = 1.095; // 1.0 -- 1.2

float Voff = 1.5; // напряжение отсечки (для Li-ion=2.5 В, для LiPo=3.0 В)

float R = 124; //сопротивление нагрузки

float I;

float cap = 0; //начальная ёмкость

float V;

float Vcc;

float Wh = 0;

unsigned long prevMillis;

unsigned long testStart;

String cap\_string;

void setup() {

pinMode(relay\_pin, OUTPUT); //пин реле как выход

digitalWrite(relay\_pin, HIGH); //выключить реле (оно обратное у меня)

pinMode(buzz\_pin, OUTPUT); //пищалка

pinMode(buzz\_gnd, OUTPUT); //пищалка

digitalWrite(buzz\_gnd, 0);

pinMode(butt\_pin, INPUT\_PULLUP); //кнопка подтянута

pinMode(butt\_gnd, OUTPUT); //земля кнопки

digitalWrite(butt\_gnd, 0);

pinMode(disp\_vcc, OUTPUT); //питание дисплея

pinMode(disp\_gnd, OUTPUT); //земля дисплея

digitalWrite(disp\_vcc, 1);

digitalWrite(disp\_gnd, 0);

disp.init(); //инициализация дисплея

disp.set(5); //яркость (0-7)

Serial.begin(9600); //открыть порт для связи с компом

Serial.println("Press any key to start the test..."); //Отправьте любой символ для начала теста

while (Serial.available() == 0) {

disp.display(3,0);

if (digitalRead(butt\_pin)==0) {break;} //или нажмите кнопку, чтобы начать тест!

}

tone(buzz\_pin,200,500);

Serial.println("Test is launched...");

Serial.print("s");

Serial.print(" ");

Serial.print("V");

Serial.print(" ");

Serial.print("mA");

Serial.print(" ");

Serial.print("mAh");

Serial.print(" ");

Serial.print("Wh");

Serial.print(" ");

Serial.println("Vcc");

digitalWrite(relay\_pin, LOW); //Переключить реле (замкнуть акум на нагрузку)

testStart = millis(); //время начала теста в системе отсчёта ардуины

prevMillis = millis(); //время первого шага

}

void loop() {

Vcc = readVcc(); //хитрое считывание опорного напряжения (функция readVcc() находится ниже)

V = (readAnalog(load\_pin) \* Vcc) / 1023.000; //считывание напряжения АКБ

I = V/R; //расчет тока по закону Ома, в Амперах

cap += I\*(millis()-prevMillis)/3600000\*1000; //расчет емкости АКБ в мАч

Wh += I\*V \*(millis() - prevMillis)/3600000; //расчет емкости АКБ в ВтЧ

prevMillis = millis();

sendData(); // отправка данных

if (V < Voff) { //выключение нагрузки при достижении минимального напряжения

digitalWrite(relay\_pin, HIGH); //разорвать цепь (отключить акб от нагрузки)

Serial.println("Test is done"); //тест закончен

for (int i=0; i<5; i++) { //выполнить 5 раз

tone(buzz\_pin,200,500); //пищать на 3 пин частотой 100 герц 500 миллисекунд

disp\_print(cap\_string);

delay(1000);

disp.clearDisplay();

delay(500);

}

while (2 > 1) { //бесконечный цикл, чтобы loop() не перезапустился + моргаем результатом!

disp\_print(cap\_string);

delay(1000);

disp.clearDisplay();

delay(500);

}

}

}

void sendData() { //функция, которая посылает данные в порт

Serial.print((millis() - testStart) / 1000);

Serial.print(" ");

Serial.print(V, 3);

Serial.print(" ");

Serial.print(I, 1);

Serial.print(" ");

Serial.print(cap, 0);

Serial.print(" ");

Serial.print(Wh, 2);

Serial.print(" ");

Serial.println(Vcc, 3);

cap\_string=String(round(cap));

disp\_print(cap\_string);

}

//----------Функция точного определения опорного напряжения для измерения напряжения на акуме-------

float readAnalog(int pin) {

// read multiple values and sort them to take the mode

int sortedValues[NUM\_READS];

for (int i = 0; i < NUM\_READS; i++) {

delay(25);

int value = analogRead(pin);

int j;

if (value < sortedValues[0] || i == 0) {

j = 0; //insert at first position

}

else {

for (j = 1; j < i; j++) {

if (sortedValues[j - 1] <= value && sortedValues[j] >= value) {

// j is insert position

break;

}

}

}

for (int k = i; k > j; k--) {

// move all values higher than current reading up one position

sortedValues[k] = sortedValues[k - 1];

}

sortedValues[j] = value; //insert current reading

}

//return scaled mode of 10 values

float returnval = 0;

for (int i = NUM\_READS / 2 - 5; i < (NUM\_READS / 2 + 5); i++) {

returnval += sortedValues[i];

}

return returnval / 10;

}

//----------Функция точного определения опорного напряжения для измерения напряжения на акуме КОНЕЦ-------

//----------фильтр данных (для уменьшения шумов и разброса данных)-------

float readVcc() {

// read multiple values and sort them to take the mode

float sortedValues[NUM\_READS];

for (int i = 0; i < NUM\_READS; i++) {

float tmp = 0.0;

ADMUX = \_BV(REFS0) | \_BV(MUX3) | \_BV(MUX2) | \_BV(MUX1);

ADCSRA |= \_BV(ADSC); // Start conversion

delay(25);

while (bit\_is\_set(ADCSRA, ADSC)); // measuring

uint8\_t low = ADCL; // must read ADCL first - it then locks ADCH

uint8\_t high = ADCH; // unlocks both

tmp = (high << 8) | low;

float value = (typVbg \* 1023.0) / tmp;

int j;

if (value < sortedValues[0] || i == 0) {

j = 0; //insert at first position

}

else {

for (j = 1; j < i; j++) {

if (sortedValues[j - 1] <= value && sortedValues[j] >= value) {

// j is insert position

break;

}

}

}

for (int k = i; k > j; k--) {

// move all values higher than current reading up one position

sortedValues[k] = sortedValues[k - 1];

}

sortedValues[j] = value; //insert current reading

}

//return scaled mode of 10 values

float returnval = 0;

for (int i = NUM\_READS / 2 - 5; i < (NUM\_READS / 2 + 5); i++) {

returnval += sortedValues[i];

}

return returnval / 10;

}

//----------фильтр данных (для уменьшения шумов и разброса данных) КОНЕЦ-------

void disp\_print(String x) {

disp.point(POINT\_OFF);

switch (x.length()) { //тут измеряется длина строки и соотвествено выводится всё на дисплей

case 1:

disp.display(0,18);

disp.display(1,18);

disp.display(2,18);

disp.display(3,x[0]- '0');

break;

case 2:

disp.display(0,18);

disp.display(1,18);

disp.display(2,x[0]- '0');

disp.display(3,x[1]- '0');

break;

case 3:

disp.display(0,18);

disp.display(1,x[0]- '0');

disp.display(2,x[1]- '0');

disp.display(3,x[2]- '0');

break;

}

}

# Приложение Б – Библиотека AKB

// Author:Fred.Chu

// Date:9 April,2013

// catalex.taobao.com

// This library is free software; you can redistribute it and/or

// modify it under the terms of the GNU Lesser General Public

// License as published by the Free Software Foundation; either

// version 2.1 of the License, or (at your option) any later version.

//

// This library is distributed in the hope that it will be useful,

// but WITHOUT ANY WARRANTY; without even the implied warranty of

// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

// Lesser General Public License for more details.

//

// You should have received a copy of the GNU Lesser General Public

// License along with this library; if not, write to the Free Software

// Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

//

// Modified record:

//

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

#include <TimerOne.h>

#include "TM1637.h"

#define ON 1

#define OFF 0

int8\_t TimeDisp[] = {0x00,0x00,0x00,0x00};

unsigned char ClockPoint = 1;

unsigned char Update;

unsigned char halfsecond = 0;

unsigned char second;

unsigned char minute = 0;

unsigned char hour = 12;

#define CLK 3//pins definitions for TM1637 and can be changed to other ports

#define DIO 2

TM1637 tm1637(CLK,DIO);

void setup()

{

tm1637.set();

tm1637.init();

Timer1.initialize(500000);//timing for 500ms

Timer1.attachInterrupt(TimingISR);//declare the interrupt serve routine:TimingISR

}

void loop()

{

if(Update == ON)

{

TimeUpdate();

tm1637.display(TimeDisp);

}

}

void TimingISR()

{

halfsecond ++;

Update = ON;

if(halfsecond == 2){

second ++;

if(second == 60)

{

minute ++;

if(minute == 60)

{

hour ++;

if(hour == 24)hour = 0;

minute = 0;

}

second = 0;

}

halfsecond = 0;

}

// Serial.println(second);

ClockPoint = (~ClockPoint) & 0x01;

}

void TimeUpdate(void)

{

if(ClockPoint)tm1637.point(POINT\_ON);

else tm1637.point(POINT\_OFF);

TimeDisp[0] = hour / 10;

TimeDisp[1] = hour % 10;

TimeDisp[2] = minute / 10;

TimeDisp[3] = minute % 10;

Update = OFF;

}