

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ АЕРОНАВІГАЦІЇ, ЕЛЕКТРОНІКИ ТА ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ЕЛЕКТРОНІКИ, РОБОТОТЕХНІКИ І ТЕХНОЛОГІЙ
МОНІТОРИНГУ ТА ІНТЕРНЕТУ РЕЧЕЙ

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач випускової кафедри
_____ Володимир ШУТКО

« ___ » _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

ЗДОБУВАЧА ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА
ЗІ СПЕЦІАЛЬНОСТІ 171 «ЕЛЕКТРОНІКА»
ОПП «ЕЛЕКТРОННІ ТЕХНОЛОГІЇ ІНТЕРНЕТУ РЕЧЕЙ»

Тема: «Система моніторингу та контролю напруги літєвих батарей великої
потужності на платформі Arduino»

Виконавець:

студент групи ІР-307Б стн _____ Хобта Дмитро Андрійович

Керівник:

к.т.н. доцент _____ Іванець О.Б.

Нормоконтролер:

_____ Сініцин Р.Б.

КИЇВ 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ АЕРОНАВІГАЦІЇ, ЕЛЕКТРОНІКИ ТА ТЕЛЕКОМУНІКАЦІЙ
КАФЕДРА ЕЛЕКТРОНІКИ, РОБОТОТЕХНІКИ І ТЕХНОЛОГІЙ
МОНІТОРИНГУ ТА ІНТЕРНЕТУ РЕЧЕЙ

171 «ЕЛЕКТРОНІКА»,
ОПП«ЕЛЕКТРОННІ ТЕХНОЛОГІЇ ІНТЕРНЕТУ РЕЧЕЙ»

ДОПУСТИТИ ДО ЗАХИСТУ
Завідувач випускової кафедри
_____ Володимир Шутко

« ___ » _____ 2023 р.

ЗАВДАННЯ

на виконання дипломної роботи

Хобта Дмитро Андрійович

(П.І.Б., випускника)

1. Тема дипломної роботи: «Система моніторингу та контролю напруги літєвих батарей великої потужності на платформі Arduino» затверджена наказом ректора від «23» березня 2023 р. №_387/ст.
2. Термін виконання роботи: з«23» березня 2023р. по«21» травня 2023р
3. Вихідні дані до роботи: розробити систему моніторингу та контролю напруги літєвих батарей великої потужності на платформі Arduino.
4. Зміст пояснювальної записки: 1 Загальний розділ, 2 Спеціальний розділ, 3 Експлуатаційний розділ.
5. Перелік обов'язкового графічного (ілюстрованого) матеріалу: таблиці, рисунки, зображення сенсорів, модулів, пристрою, код програми.

6. Календарний план-графік

№ п/п	Завдання	Термін виконання етапів	Відмітка про виконання
1.	Затвердження теми бакалаврської роботи	23.03.2023р	
2.	Вивчення літератури	24.03.2023р.- 03.04.2023р.	
3.	Теоретичні основи розробки	04.04.2023р.- 09.04.2023р.	
4.	Вибір технічних засобів	10.04.2023р.- 22.04.2023р.	
5.	Апаратно-програмна реалізація	23.04.2023р.- 14.05.2023р.	
6.	Оформлення та усунення недоліків дипломної роботи	15.05.2023р.- 21.05.2023р.	

Дата видачі завдання: «23» березня 2023 р.

Керівник дипломної роботи _____

(підпис керівника)

Іванець О.Б.

(П.І.Б.)

Завдання прийняв до виконання _____

(підпис випускника)

(П.І.Б.)

Хобта Д.А.

РЕФЕРАТ

Пояснювальна записка до дипломної роботи «Система моніторингу та контролю напруги літієвих батарей великої потужності на платформі Arduino» містить: 52 сторінки, 26 рисунків, 3 таблиці, 18 використаних джерела.

Актуальність теми полягає у створенні системи моніторингу та контролю напруги літієвих батарей великої потужності на платформі Arduino. Сучасний розвиток технологій привів до зростання популярності використання портативних пристроїв, таких як мобільні телефони, планшети, ноутбуки та електромобілі. Ці пристрої працюють від акумуляторних батарей, і важливо забезпечити ефективний контроль та управління напругою цих батарей для забезпечення безперебійної роботи пристроїв та забезпечення їх тривалої роботи без підзарядки. Тому тематика систем контролю напруги батарей мають велику актуальність в сучасному технологічному світі. Також є потреба у покращення цих систем для забезпечення більш ефективного масштабування.

Метою даної роботи є розробка ефективної системи контролю напруги батарей. Основним завданням є аналіз сучасних методів контролю напруги, вивчення вимог до стабільності та безпеки електронних пристроїв, а також розробка прототипу системи контролю напруги батарей для його подальшого використання в різних областях, включаючи електроніку, автомобілебудування та інші промислові галузі.

Мета дипломної роботи – розробити систему моніторингу та контролю напруги літієвих батарей великої потужності на платформі Arduino.

Об'єктом дослідження є система контролю напруги батарей, яка включає в себе електронні компоненти, датчики, алгоритми та програмне забезпечення, необхідне для контролю та регулювання напруги батарейного джерела живлення.

Матеріали даної дипломної роботи можуть бути використані для проведення наукових досліджень, у навчальному процесі, а також з можливістю використання в практичній діяльності процесів побудови системи моніторингу.

Ключові слова: МІКРОКОНТРОЛЕР, ЕЛЕКТРОНІКА, ARDUINO.

ЗМІСТ

1. Загальний розділ	6
1.1 Огляд предметної області	6
1.2 Порівняльний аналіз пристрою з аналогічними за призначенням	12
2. Спеціальний розділ	17
2.1 Постановка задачі	17
2.2 Елементна база пристрою	18
2.3 Характеристики компонентів пристрою	19
2.4 Опис середовища розробки	31
3. Експлуатаційний розділ	33
3.1 Технічне завдання на розробку пристрою	33
3.2 Проєктування пристрою	35
3.3 Написання програми для роботи пристрою	41
3.4 Моделювання плати пристрою	47
Висновок	50
Список літератури	51

1. ЗАГАЛЬНИЙ РОЗДІЛ

1.1 Огляд предметної області

Системи моніторингу напруги літєвої батареї використовуються для вимірювання напруги окремих елементів або груп елементів батареї. Зазвичай він складається з електронного модуля, який можна підключати до кількох елементів батареї одночасно, і програмного забезпечення для збору й аналізу даних.

Система моніторингу напруги літєвої батареї дозволяє завчасно виявити проблеми з елементами батареї. Вони можуть стежити за змінами напруги та попереджати про можливі проблеми з батареєю, такі як підвищений нагрів або зменшення ємності.

Також такі системи можуть бути оснащені додатковими функціями, такими як моніторинг температури, захист від короткого замикання, моніторинг струму заряду та розряду, а також можливість підключення до інших систем керування живленням.

Застосування систем моніторингу напруги літєвих батарей може бути дуже різноманітним. Вони використовуються в електромобілях, сонячних станціях, електронних пристроях та інших промислових і побутових застосуваннях.

Таким чином, система моніторингу напруги літєвої батареї виявляє проблеми з батареєю на ранній стадії, запобігаючи їх виникненню та забезпечуючи більш ефективно та безпечно використання літєвих батарей у різних сферах застосування.

Tesla є одним із провідних виробників електромобілів і акумуляторів. Вони розробляють власні системи моніторингу батарей, які використовуються в їхніх автомобілях. Вони збирають велику кількість даних про стан батареї, включаючи температуру, напругу, струм та інші параметри. Ці дані передаються до центрального сервера, де проводяться аналіз і діагностика стану батареї.

Tesla виробляє високомодульну акумуляторну батарею з високою ефективністю, надійністю та безпекою. Як пояснювалося вище, акумуляторна батарея складається з до 16 модулів, з'єднаних разом в одну серію. Напруга

аккумуляторної батареї Tesla становить близько 400 Вольт, і це єдиний найважчий компонент, і всі різні версії одних і тих же автомобілів можуть мати різну аккумуляторну батарею, що змінює вагу і ємність накопичувача енергії.

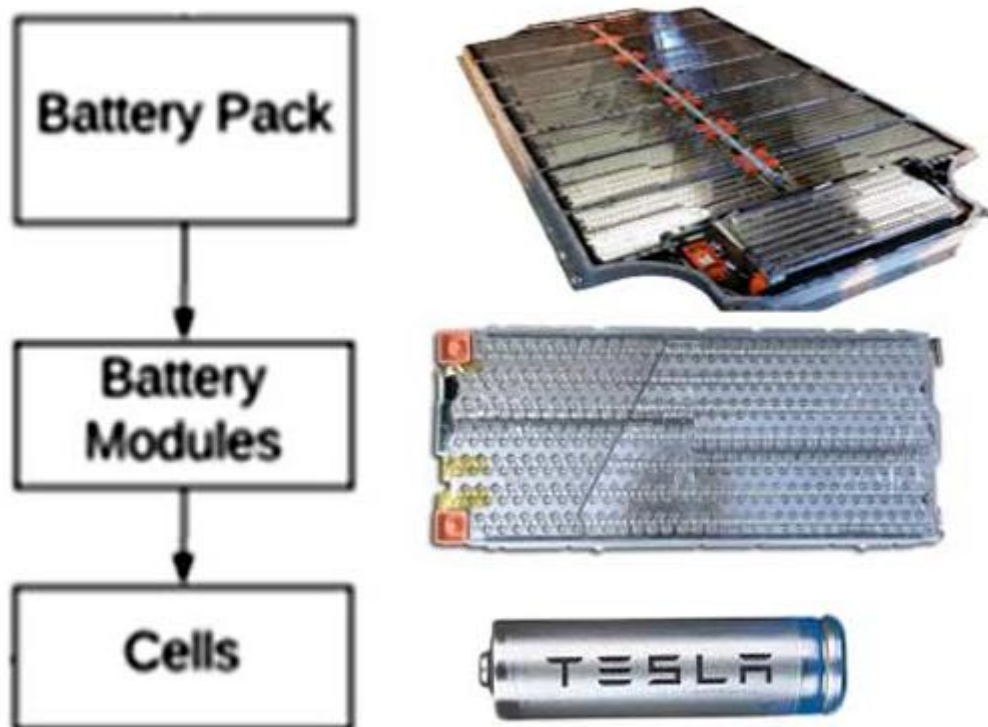


Рисунок 1.1-Структура батареї

Наприклад, аккумуляторна батарея Model S P85 має ємність 90 кВт·год і важить понад 530 кг. Він містить 16 модулів, а це 7104 18650 осередків. Аккумуляторна батарея має центральну шину, яка з'єднує кожен аккумуляторний модуль з контактором, який живить як передні, так і задні електродвигуни. Оскільки кожен модуль становить 5,5 кВт·год, а у нас їх 16 у аккумуляторі Tesla ємністю 90 кВт·год. Таким чином, це робить його модулем 84 кВт·год.

Tesla використовує Battery Management System (BMS), якf постійно моніторить дані з кожної клітинки у батареї за допомогою датчиків. Ця інформація використовується для розрахунку оптимальних швидкостей зарядки та розрядки для кожної клітинки на основі її індивідуальних характеристик. BMS потім налаштовує поточний струм відповідно до безпечних меж кожної клітинки.



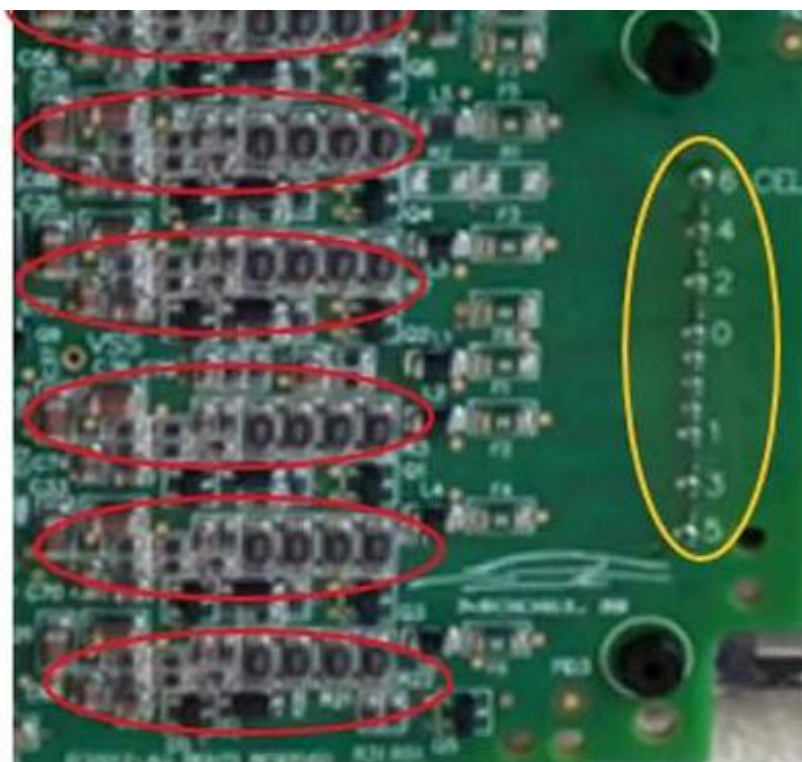
Рисунок 1.2-Плата BMS модуля, зворотна сторона



Рисунок 1.3-Плата BMS модуля

BMS може зв'язуватися один з одним за допомогою зв'язки SPI. Всі BMS модуля діють як ведені BMS і спілкуються з майстер-BMS через ізоляційний бар'єр,

головна BMS спілкується з ЕБУ і зарядним пристроєм і управляє основними контакторами. BMS розміщується збоку від кожного акумуляторного модуля. Вимірювання напруги комірки виконується за допомогою проводів, приварених до сполучних пластин паралельних з'єднань. Про модуль BMS, який використовується в Tesla Model S, доступно не так багато даних, тому про технологію BMS зроблено мало розрахункових припущень.



Cell Terminals

Рисунок 1.4-Місце підключення до модуля

Червона обведена область на наведеному вище зображенні BMS показує, що BMS містить 6 мікросхем моніторингу для кожного послідовного підключення в модулі. Жовта обведена область показує клеми, куди буде підключено з'єднання з 6 комірок. IC by TI, що використовується в BMS, може бути ланцюжковим, щоб потребувати лише одного комунікаційного з'єднання з системою. І, ймовірно, головна використовувана BMS розроблена компанією BOSCH, вся система використовує зв'язок CAN-шини для зв'язку з центральним комп'ютером автомобіля. Якщо у вас є додаткова інформація, будь ласка, не соромтеся опублікувати її, щоб ми могли її оновити.

Наступним досліджуваним кандидатом буде LG Chem.

LG Chem надає комплексні системи моніторингу батарей, які включають в себе сенсори для збору даних про стан батареї, модулі збору даних та програмне забезпечення для аналізу та візуалізації даних.

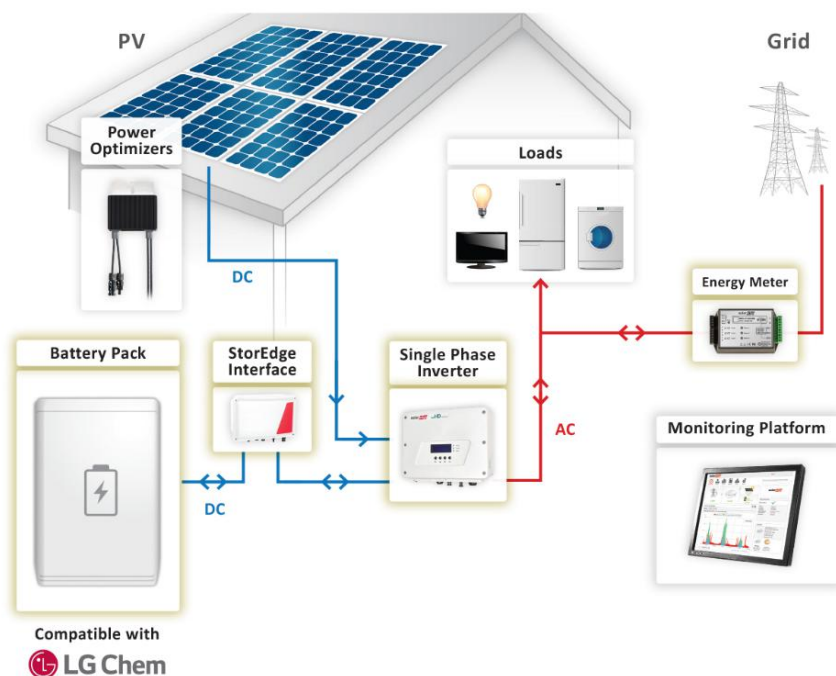


Рисунок 1.5-Приклад використання BMS

LG Chem використовує Battery Management System (BMS), який моніторить стан батареї та забезпечує безпеку її роботи. BMS забезпечує моніторинг параметрів батареї, таких як напруга, струм, температура тощо. Він також контролює зарядку та розрядку батареї

Система моніторингу акумулятора, що включає:

2. датчик струму, електрично з'єднаний послідовно з акумуляторною батареєю;
3. датчик напруги, електрично з'єднаний паралельно елементу акумулятора;
4. мікроконтролер, який отримує значення струму комірки струму, що протікає через елемент батареї від датчика струму при частоті першого оновлення M разів на секунду;

Мікроконтролер отримує значення напруги комірки акумуляторної батареї від датчика напруги, паралельно отриманню значень струму комірки, по відношенню до

електричної землі, при частоті повторного оновлення N разів на секунду, при цьому $M \geq N$;

при третій частоті оновлення R разів на секунду, де $R < M$ і $R < N$, мікроконтролер, що визначає швидкість зміни струму комірки, використовуючи два значення струму комірки значень струму комірки; мікроконтролер, що визначає швидкість зміни напруги комірки, використовуючи два значення напруги комірки значень напруги комірки;

Мікроконтролер, що визначає значення відфільтрованого омичного опору на основі швидкості зміни напруги комірки, швидкості зміни струму комірки і попереднього відфільтрованого омичного значення опору, якщо числовий знак швидкості зміни напруги комірки ідентичний числовому знаку швидкості зміни струму комірки, і швидкість зміни напруги комірки, менша або рівна верхньому порогу допустимих швидкостей зміни напруги комірки, а швидкість зміни напруги комірки більше або дорівнює нижньому порогу допустимих швидкостей зміни напруги комірки.

Ці виробники та багато інших пропонують різноманітні системи моніторингу батарей залежно від конкретних потреб клієнтів. Їхні підходи та методики базуються на зборі даних про стан батареї, аналізі цих даних, передачі інформації до центрального сервера або хмарного сервісу, а також використанні алгоритмів прогнозування для визначення оптимальних стратегій управління та підтримки батарей. LG Chem та Tesla, розробляють комплексні системи моніторингу батарей, які включають сенсори, модулі збору даних та програмне забезпечення для аналізу та візуалізації даних. Вони також використовують Battery Management System (BMS), який моніторить стан батареї, контролює параметри та забезпечує безпеку роботи батареї. Кожна комірка акумулятора має окремий моніторинг, що дозволяє виявляти індивідуальні характеристики та оптимізувати зарядку та розрядку для кожної клітинки. Також можна сказати що в більшості випадків системи моніторингу подібні та виконують однакові ролі. Різниця лише в компонентній базі та алгоритмах обробки даних.

1.2 Порівняльний аналіз пристрою з аналогічними за призначенням

Розробляючи пристрій для моніторингу напруги літієвих батарей, було проведено порівняльний аналіз з аналогічними пристроями на ринку. В результаті було виявлено, що пристрій, розроблений мною, має декілька переваг порівняно з іншими пристроями.

Перевагою мого пристрою є простота конструкції за рахунок модульності. Це спрощує монтаж і обслуговування пристрою. Зокрема, заміна окремого модуля не вимагає заміни всього блоку в цілому.

Модульна структура системи дозволяє легко змінювати та розширювати її функціональні можливості за допомогою додаткових модулів. Крім того, моя система моніторингу має достатньо високий рівень точності вимірювань для більшості домашніх застосувань, що робить її привабливою для використання в різних проектах з вимірювання напруги літієвих батарей.

Крім того, мій пристрій має простий процес складання шляхом пайки SMD компонентів на заводі. Це значно скорочує час виробництва та ризик помилок, пов'язаних із ручним складанням.

Однак, порівняно з іншими пристроями, моя система моніторингу має обмежену кількість каналів вимірювання напруги і потребує додаткових модулів для розширення функціональності. Також, вона може не мати такого високого рівня точності вимірювань, як більш складні та професійні пристрої, що можуть бути використані в професійних застосунках.

В промисловості, наскільки мені відомо, не використовується платформа Arduino, а лише її мікроконтролер, також є рішення побудовані на основі подібних мікроконтролерів.

Для проведення аналізу візьмемо побутові варіанти систем які можна придбати на маркетплейсах. В цьому порівнянні ми будемо брати базову комплектацію подібних по характеристиках систем.

В порівнянні будуть такі системи:

1. Battery Monitor Bluetooth KG140F
2. Daly BMS 16S
3. JBD 10S-17S Lithium Battery Protection Board
4. Daly Home Energy Storage Smart Bms

Для початку розглянемо їх зовнішній вигляд.



Рисунок 1.6-Battery Monitor Bluetooth KG140F



Рисунок 1.7-Daly BMS 16S



Рисунок 1.8-JBD 10S-17S Lithium Battery Protection Board



Рисунок 1.9-Daly Home Energy Storage Smart Bms

Всі системи в базовій комплектації окрім KG140F можуть розривати ланцюг живлення.

Розглянемо характеристики кожної з них. Будемо брати до уваги такі критерії:

1. Розмір (якщо вказано);
2. Діапазон робочих напруг;

3. Кількість вимірюваних елементів;
4. Максимальний струм;
5. Можливість підключення додаткових модулів;
6. Масштабованість;
7. Можливість курування з додатку, або безпосередньо на пристрої;
8. Ціна.

Таблиця 1.1-Порівняльна таблиця

Характеристика\модель	KG140F	Daly BMS 16S	JBD 10S-17S Lithium Battery Protection Board	Daly Home Energy Storage Smart Bms
Розмір мм.	-	130*221*32	202*76*18	-
Діапазон напруг V	120	60 загальна	72	60
Кількість вим. елементів	1	16	20	16
Макс. струм	400	200	200	100
Дод. пристрої	ні	ні	так	так
Маштабування	ні	частково	ні	так
Керування додаток+локал.	так	Лише дод.	Опційно додаток	+
Ціна грн.	2650	4 212,59	4 271,74	5 717,25

Як бачимо з порівняльної таблиці, в кожній з систем є свої особливості і в результаті лідирують по певних характеристиках. Як не дивно, явного лідера нема, але є модель з явними недоліками. Розглянемо більш детально.

Модель 1 має широкий діапазон напруг що підійне для інтеграції в уже готову батарею, має високий поріг вимірювання струму. На цьому переваги закінчились, плюс, її використання дуже специфічне.

Модель 2 має гарні показники та може бути розглянута для використання в якості первинної системи контролю. Її характеристик вистачить для побутових цілей.

Модель 3 має перевагу в кількості підключених батарей, але немає можливості масштабування. Її можна використовувати в електро-транспорті середньої потужності таких як велосипеди та байли.

Модель 4 має недолік в низькому струмі на одну систему, але за рахунок можливості професійного масштабування цей момент можна нівелювати. В додачу до цього є можливість підключення безлічі додаткових пристроїв та є можливість керувати нею безпосередньо є великим плюсом. Але за такі можливості прийдеться добре заплатити.

2. СПЕЦІАЛЬНИЙ РОЗДІЛ

2.1 Постановка задачі

Постановка задачі для системи моніторингу напруги літієвих батарей полягає у створенні пристрою, що дозволяє контролювати напругу на кожній з акумуляторних комірок літієвої батареї в режимі реального часу. Задача включає в себе наступні пункти:

1. Розробка апаратної частини системи моніторингу: необхідно створити схему, що дозволяє зчитувати напругу на кожній комірці батареї. Для цього я використав мультиплексор CD74HC4067 та контролер Arduino Nano, які забезпечують зчитування даних та обробку сигналів;

2. Розробка програмного забезпечення: програмне забезпечення повинно бути розроблене з метою збору та аналізу даних з акумуляторних комірок батареї. Програма повинна показувати рівень напруги на кожній комірці в реальному часі та реагувати на будь-які аномальні показники напруги, що можуть свідчити про проблеми з акумулятором;

3. Встановлення системи моніторингу в батарею: після розробки апаратної та програмної частини, система моніторингу повинна бути встановлена в батарею. Це може включати в себе паяння дротів до контактів комірок батареї та розміщення апаратної частини системи моніторингу в невеликому контейнері;

4. Тестування та налагодження системи моніторингу: після встановлення системи моніторингу, вона повинна бути піддана тестуванню та налагодженню. Необхідно перевірити правильність зчитування даних та відповідність показників реальному рівню напруги на комірках батареї.

2.2 Елементна база пристрою

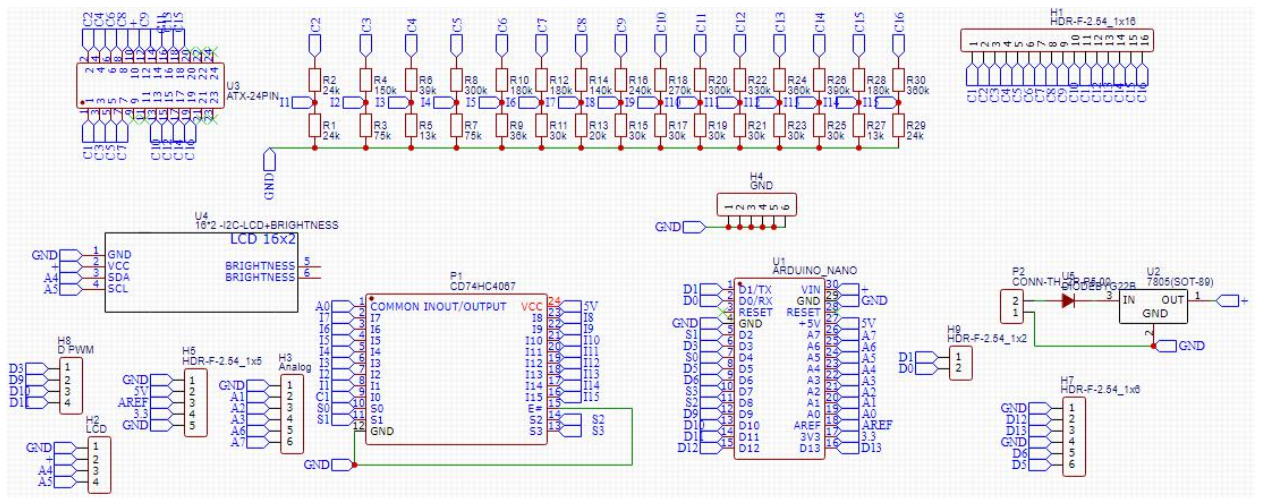


Рисунок 2.1 – Схема пристрою

Елементна база складається з наступних компонентів:

1. Платформа Arduino - це основний елемент системи, який відповідає за обробку та збір даних. В моєму випадку була використана платформа Arduino Nano.
2. Аналоговий 16-ти каналний мультиплексор (комутатор) CD74HC4067.
3. Лінійний стабілізатор - використовується для зниження напруги живлення до рівня живлення компонентів.
4. Дисплей - використовується для відображення інформації батареї та параметрів, які контролюються системою. Дисплей використовується LCD.
5. Дільник напруги - використовується для зниження напруги з батареї до рівня, який може сприйняти Arduino.
6. Інші компоненти, такі як резистори, конденсатори, діоди, роз'єми тощо, які використовуються для захисту та контролю струму в системі.

2.3 Характеристики компонентів пристрою

Огляд компонентів та їх характеристик розпочну з вхідної частини живлення.

L7805 — це лінійний стабілізатор напруги з фіксованою вихідною напругою 5 В. L7805 є широко використовуваним компонентом в електроніці та широко використовується для стабілізації напруги різних електронних пристроїв, таких як джерела живлення, пристрої автоматизації та пристрої зв'язку.

В моєму пристрої він виконує роль вхідного стабілізатора для живлення всіх компонентів схеми.



Рисунок 2.1- L7805

Вхідна напруга V	5-18
Вихідна напруга V	5
Робоча температура °C	До 125
Точність вихідної напруги	±2%

Таблиця 2.1 Характеристики L7805

Також цей стабілізатор має захист від перегріву та КЗ.

SS520

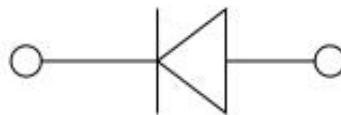
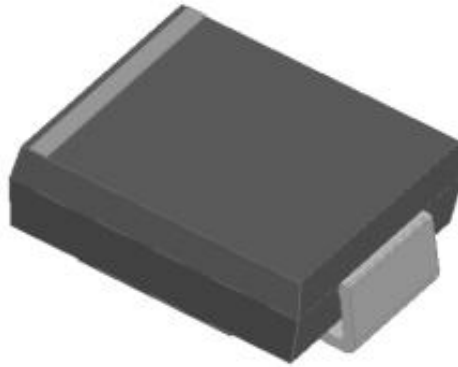


Рисунок 2.2- SS520

Діод SS520 є швидким діодом, який часто використовується в електронних пристроях для випрямлення змінного струму та захисту від зворотного струму. Його швидкодія та низька напруга перемикання роблять його хорошим вибором для систем, що потребують високої швидкості перемикання. Він також компактний та легко монтується на поверхні друкованих плат, що робить його зручним для використання у малогабаритних пристроях.

Таблиця 2.2 Характеристики SS520

Виробник	Yangjie
Імпульсний струм	100 A
Монтаж	SMD
Падіння напруги	0,9 V
Прямий струм	5 A
Зворотня напруга	200 V

Наступним і водночас одним із головних компонентів є платформа Arduino Nano. Arduino Nano — це невелика, але потужна розробна плата на основі мікроконтролера ATmega328. Маленька та легка, вона підходить для вбудованих проектів з обмеженим простором. Вона має всі функції, необхідні для розробки та роботи мого пристрою.

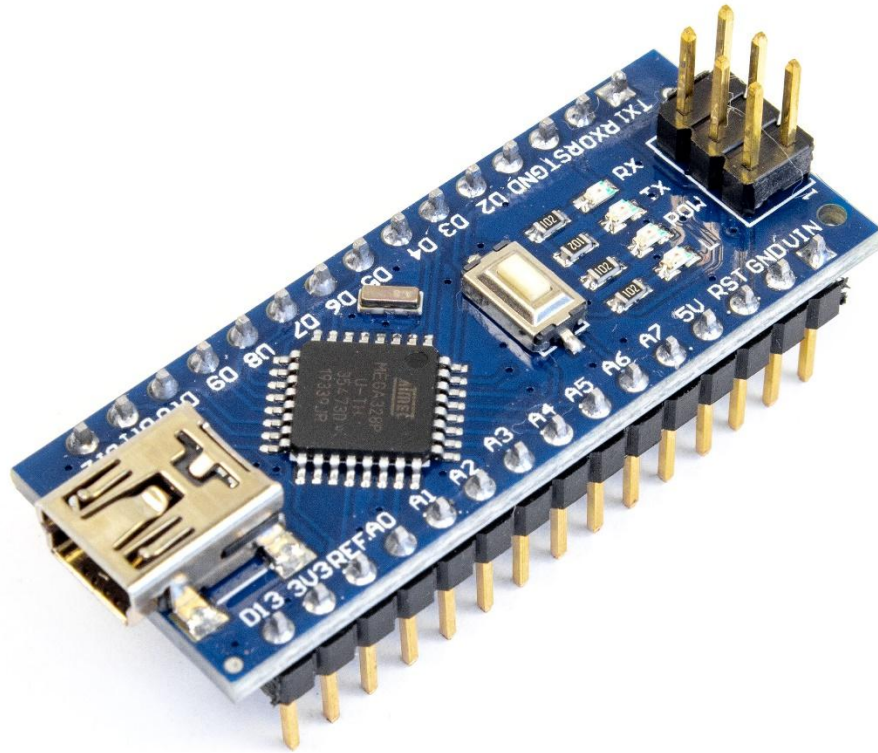


Рисунок 2.3- Arduino Nano

Основні характеристики Arduino Nano включають:

1. Мікроконтролер: ATmega328P
2. Напруга живлення: 5 В
3. Вхідна напруга (рекомендована): 7-12 В
4. Вхідна напруга (максимальна): 20 В
5. Цифрові входи/виходи: 14 (з них 6 можуть бути використані як PWM виходи)
6. Аналогові входи: 8
7. Струми вихідних пінів: 40 мА

8. Flash-пам'ять: 32 КБ (2 КБ використовується для завантажувача)
9. SRAM: 2 КБ
10. EEPROM: 1 КБ
11. Швидкість тактування: 16 МГц
12. Розміри: 18 мм x 45 мм

Важливі для мене характеристики:

1. Мікроконтролер ATmega328 з частотою 16 МГц і 32 кБ флеш-пам'яті.
2. Малий розмір 18 мм x 45 мм, що робить її ідеальною для мого проєкту.
3. Підтримка підключення до комп'ютера через USB-порт, що робить завантаження коду на плату простим і швидким, а також дозволяє провести налаштування та відслідкувати багато параметрів.
4. Достатня кількість цифрових та аналогових входів/виходів, що дозволяє підключати до неї різні датчики, пристрої управління та інше.
5. Підтримка багатьох бібліотек та шілдів, що дозволяє розширити функціональність плати та спростити розробку проєкту.

В моєму проєкті ардуіно виконує роль основного контролера та виконує наступні функції:

1. Зчитування напруги: контролер зчитує напругу на кожному окремому елементі літєвої батареї, а також загальну напругу всієї батареї.

2. Обробка даних: контролер обробляє дані напруг та забезпечує правильне відображення вимірювань в системі моніторингу.

3. Керування індикацією: контролер керує індикацією стану батареї, реалізованою з використанням світлодіодів та дисплею.

4. Керування захистом батареї: контролер може бути підключений до системи захисту батареї та керувати її роботою, якщо виникнуть проблеми з напругою.

Всі вимірювання платформа виконує виключно за допомогою вбудованих можливостей. Оскільки платформа не може виконувати вимірювання на кожній комірці, через обмежену кількість аналогових портів я для розширення функціоналу аналогових нам портів я використав аналоговий мультиплексор.

Для розширення кількості аналогових портів та спрощення вимірювань було використано мультиплексор CD74HC4067.

CD74HC4067

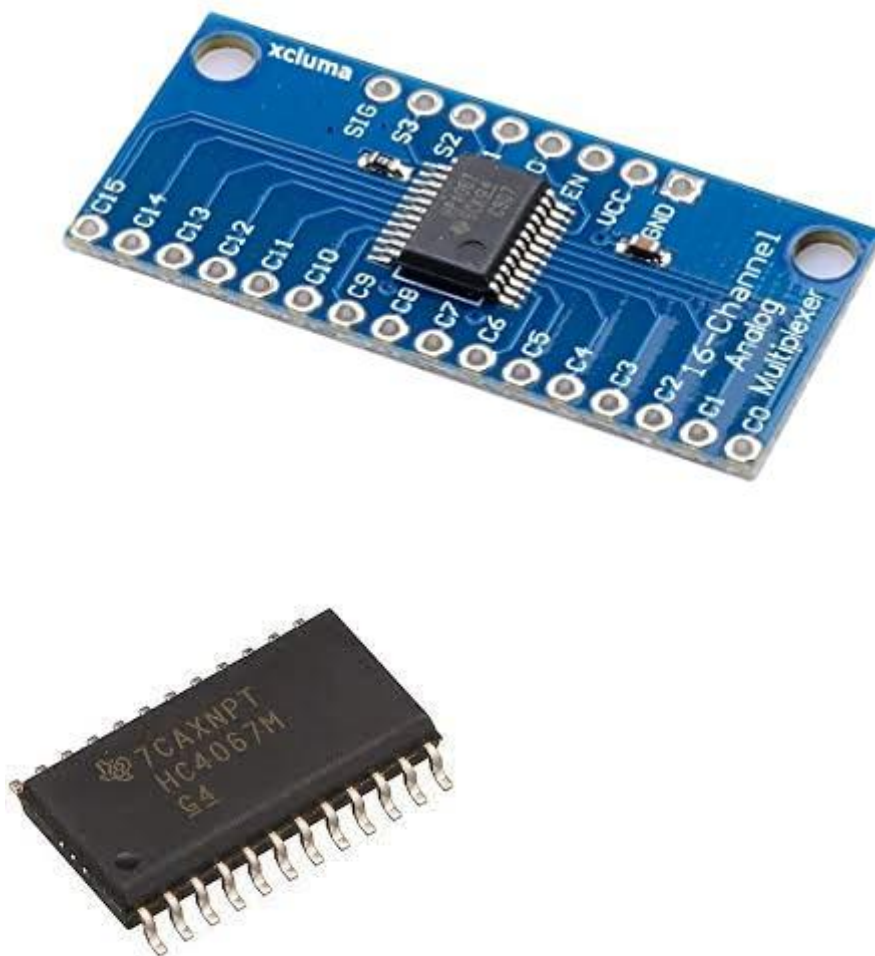


Рисунок 2.4- CD74HC4067

Мультиплексор CD74HC4067 - це інтегральна високошвидкісна схема, що містить 16 аналогових входів, які можна вибрати за допомогою 4-бітного коду, а

також вихідний буфер з високим імпедансом. Вона працює з напругою живлення від 2 до 6 В і забезпечує мінімальний споживаний струм менше 1 мА.

CD74HC4067 використовує технологію КМОП (Complementary Metal-Oxide-Semiconductor) з використанням високопродуктивних MOS-транзисторів і забезпечує широкий діапазон робочих температур від -40 °С до +85 °С. Інтегральна схема підтримує вхідні напруги від -0,5 В до $V_{CC} + 0,5$ В та має високий ступінь стійкості до електростатичного розряду, що дорівнює 2000 В.

У мультиплектора CD74HC4067 є вбудована функція дешифрації адреси, що дозволяє легко вибрати потрібний вхідний сигнал. Він має високу швидкість перемикання, що дорівнює 50 нс, і підтримує роботу з частотами вхідних сигналів до 50 МГц. Також мультиплектор може використовуватися як демультиплектор, при цьому його вихідний буфер з високою імпедансом дозволяє підключати кілька пристроїв до одного входу.

Узагальнюючи, мультиплектор CD74HC4067 є потужним інструментом для вибору одного з 16 вхідних сигналів з високою швидкістю перемикання і може бути використаний у різноманітних електронних пристроях, які потребують вибору різних вхідних сигналів в залежності від умови.

CD74HC4067 є мультиплексором, який може бути використаний з Arduino для підключення до багатьох аналогових вхідних пристроїв. Ось декілька можливостей взаємодії між Arduino та CD74HC4067:

1. Підключення CD74HC4067 до плати Arduino: для підключення CD74HC4067 до плати Arduino необхідно підключити його до порту GPIO або аналогового порту на платі Arduino за допомогою дротів;

2. Керування вхідними сигналами: Arduino може керувати вхідними сигналами CD74HC4067, вибираючи потрібний вхідний канал для зчитування;

3. Зчитування аналогових даних: CD74HC4067 може бути використаний для зчитування аналогових даних з декількох джерел, таких як сенсори, резистори і т.д. Arduino може зчитувати ці дані через CD74HC4067 та обробляти їх, як потрібно;

4. Використання декількох CD74HC4067: в залежності від потреб проекту, може бути використано декілька мультиплексорів CD74HC4067, щоб розширити можливості системи зчитування даних;

5. Управління потоком даних: Arduino може керувати потоком даних, отриманих від CD74HC4067, в залежності від потреб проекту, зчитуючи дані з різних вхідних каналів та обробляючи їх;

6. Використання CD74HC4067 для керування виводами: в деяких проектах CD74HC4067 може бути використаний для керування виводами, наприклад, для підключення до різних пристроїв у системі керування роботами.

Дільники напруги.

Для проведення вимірювань у нас є пряма необхідність підключення мультиплексора до батареї. Але обмеження в напрузі не дозволяє нам цього зробити. Для цього я використав дільник напруги.

Резисторний дільник напруги — це електрична схема, що складається з резистора, призначеного для отримання напруги, зменшеної від початкової напруги. Ці схеми можна використовувати в різноманітних електронних пристроях, таких як джерела живлення, аналогові схеми, датчики та інші пристрої.

Дільник напруги зазвичай складається з двох або більше резисторів, з'єднаних послідовно або паралельно. Значення напруги на виході дільника залежить від співвідношення номіналу резистора до значення вхідної напруги. Як правило, дільники напруги використовуються для зниження вхідної напруги до значення, прийняттого для подальшої обробки в електронних схемах.

Резисторні дільники напруги мають ряд переваг. Вони особливо недорогі та мають малу кількість компонентів, що спрощує установку та знижує вартість виробництва електроніки. Крім того, використання дільника напруги забезпечує високу точність вимірювання порівняно з початковою напругою.

Застосування дільників напруги на основі опору можна знайти в різноманітних електронних пристроях, таких як аналогові схеми, датчики та системи автоматичного керування. Таким чином, дільники напруги на основі

резисторів є важливим елементом у реалізації електронних пристроїв, які мають вимогу знижувати напругу від високого значення до прийняттого значення для подальшої обробки.

На схемі він займає велику частину місця, та досить складно виконаний.

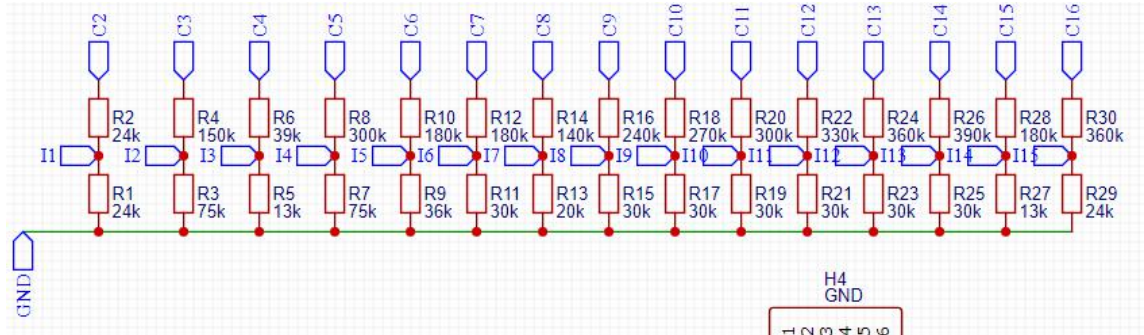


Рисунок 2.5- Дільник напруги

Він виконаний з пар SMD резисторів 0805.



Рисунок 2.6- SMD резистори

Формула для розрахунку дільника напруги на основі резисторів має вигляд:

$$V_{out} = V_{in} \times (R2 / (R1 + R2))$$

де V_{out} - вихідна напруга дільника, V_{in} - вхідна напруга, $R1$ та $R2$ - опір резисторів, підключених послідовно.

Ця формула використовується для розрахунку вихідної напруги дільника на основі значень опору резисторів та вхідної напруги. За допомогою цієї формули можна визначити співвідношення між вхідною та вихідною напругами, а також вибрати значення резисторів, щоб отримати необхідний рівень вихідної напруги.

Виведення значень з Arduino.

Існує кілька способів виведення значень з Arduino, залежно від типу даних та призначення виводу. Деякі з найпоширеніших методів виведення:

1. Використання Serial Monitor - це вбудований інструмент Arduino IDE, який дозволяє виводити значення на монітор комп'ютера. Для цього використовується команда "Serial.print()" або "Serial.println()", в залежності від того, як потрібно вивести значення;

2. Використання LCD дисплею - можна підключити LCD дисплей до Arduino та виводити значення на ньому. Для цього використовуються спеціальні бібліотеки, які дозволяють контролювати дисплей та виводити на ньому текст або значення;

3. Використання LED світлодіодів - можна використовувати світлодіоди для візуального виведення значень. Наприклад, можна підключити світлодіоди до виходів Arduino та вмикати їх в залежності від значень датчиків або внутрішніх змінних.

Я використовую перші 2 способи. Перший спосіб використовує програмні рішення на ньому ми зупинемось пізніше, а LCD обговоримо зараз.

I2C LCD дисплей 16x2

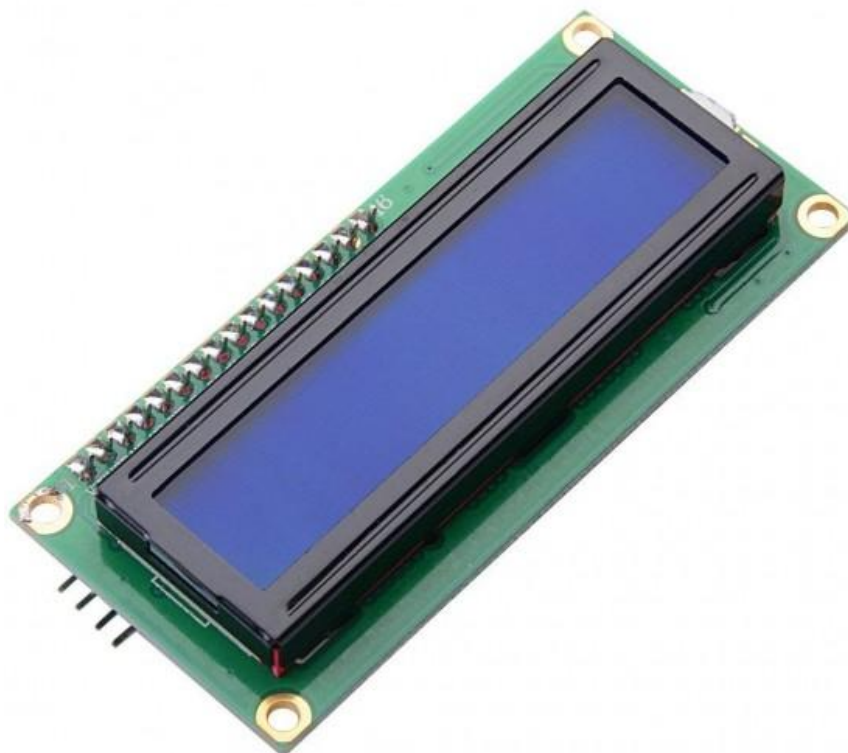


Рисунок 2.7- I2C LCD дисплей 16x2

I2C LCD дисплей 16x2 для Arduino - це зручний інтерфейс для виведення текстової інформації. Даний дисплей підключається до контролера Arduino через I2C протокол і забезпечує простий і зручний спосіб виводу інформації.

Дисплей має 16 символних рядків і 2 стовпці, що дозволяє виводити текст з шириною рядка до 16 символів. Заднє підсвічування дозволяє використовувати дисплей в темних приміщеннях, а контраст можна настроїти за допомогою потенціометра.

Щоб підключити I2C LCD дисплей до Arduino, потрібно підключити SDA та SCL піни дисплею до відповідних пінів на Arduino (зазвичай A4 та A5 відповідно), а також підключити живлення та землю.

Для роботи з дисплеєм можна використовувати готові бібліотеки, такі як LiquidCrystal_I2C.h. Дана бібліотека дозволяє легко виводити текст на дисплей, налаштовувати контраст і підсвічування, а також використовувати спеціальні символи.

В загальному, I2C LCD дисплей 16x2 для Arduino є зручним і простим інтерфейсом для виведення текстової інформації, який дозволяє зосередитись на основній функціональності проекту без необхідності витрачати час на розробку власного інтерфейсу виводу.

2.4 Опис середовища розробки

Сервіс Wokwi - це інтерактивна платформа для моделювання та тестування електронних пристроїв. Він дає можливість користувачам створювати електричні схеми, програмувати мікроконтролери та симулювати роботу пристроїв без потреби в реальних апаратах. Сервіс Wokwi має кілька варіантів використання: як онлайн-сервіс на веб-сторінці, як інструмент для розробки на локальному комп'ютері та як плагін до різноманітних програм для розробки електроніки.

Сервіс Wokwi має дуже широкі можливості для моделювання електронних пристроїв, що дозволяє користувачам додавати до своїх проектів багато елементів, таких як мікроконтролери, датчики, актуатори та інші електронні компоненти. Також він підтримує більшість популярних мікроконтролерів, таких як Arduino, ESP8266, STM32 та інші.

Для створення моделі пристрою на Wokwi потрібно виконати кілька кроків. По-перше, користувач має створити електричну схему свого пристрою за допомогою візуального конструктора, який надається сервісом. Далі, користувач може додати до своєї схеми необхідні елементи, які будуть використовуватися в його проекті. Після цього можна розмістити скетч на мікроконтролері та запустити симуляцію.

Для тестування пристрою, який було створено на Wokwi, необхідно запустити симуляцію. Для цього необхідно вибрати кнопку "Run Simulation", яка знаходиться у верхньому меню.

Після запуску симуляції користувач може бачити всі введені дані та вихідні значення. Крім того, користувач може переглядати всі виконувані команди мікроконтролера. У разі виявлення помилок, користувач може зупинити симуляцію та внести необхідні зміни у код.

Після тестування пристрою та виявлення помилок, користувач може внести необхідні зміни у код і знову запустити симуляцію. Якщо все працює належним

чином, користувач може завантажити код на реальний мікроконтролер та перевірити його роботу на практиці.

Однією з переваг використання Wokwi є те, що користувач може створювати схеми та моделі пристроїв без необхідності у реальному обладнанні та витратити кошти на його придбання. Крім того, Wokwi надає можливість ділитися створеними схемами з іншими користувачами та спільно працювати над однією моделлю.

Однак, використання Wokwi має свої недоліки. Наприклад, можуть бути проблеми з точністю симуляції, оскільки вона не повністю відображає реальне обладнання. Крім того, користувач повинен мати базові знання програмування та роботи з електронними компонентами, щоб успішно створювати та тестувати свої пристрої на Wokwi.

Отже, Wokwi є потужним інструментом для створення та тестування різноманітних пристроїв, включаючи системи моніторингу напруги літієвих батарей. Використання Wokwi дозволяє значно скоротити час розробки та тестування пристрою, а також зменшити витрати на його створення.

3. ЕКСПЛУАТАЦІЙНИЙ РОЗДІЛ

3.1 Технічне завдання на розробку пристрою

Технічне завдання для системи моніторингу напруги літійових батарей повинно містити наступну інформацію:

1. Призначення: система моніторингу призначена для контролю напруги на кожній з акумуляторних комірок літійової батареї в режимі реального часу.

2. Технічні вимоги:

- Система повинна забезпечувати точне зчитування рівня напруги на кожній комірці батареї з точністю не гірше 0,1 В;

- Система повинна бути компактною та мобільною, щоб можна було легко встановлювати та знімати її з батареї;

- Система повинна працювати з мінімальним використанням енергії, щоб не зменшувати енергоефективність батареї;

- Система повинна бути досить простою у використанні, щоб можна було легко налаштувати та зчитувати дані з неї.

3. Апаратна частина:

- Мультиплексор CD74HC4067 та контролер Arduino Nano повинні бути використані для зчитування даних та обробки сигналів;

- Потрібно створити схему, що дозволяє зчитувати напругу на кожній комірці батареї;

- Потрібно використовувати високоякісні компоненти, щоб забезпечити точність зчитування даних.

4. Програмне забезпечення:

- Програма повинна бути розроблена з метою збору та аналізу даних з акумуляторних комірок батареї;

- Програма повинна показувати рівень напруги на кожній комірці в реальному часі та реагувати на будь-які аномальні показники напруги, що можуть свідчити про проблеми з акумулятором;
- Програма повинна мати можливість зберігати дані в файлі або базі даних для подальшого аналізу(не обов'язово);
- Програма повинна мати можливість налаштовувати порогові значення напруги, при досягненні яких буде відбуватись сповіщення про необхідність заміни акумуляторної батареї або її частин;
- Програма повинна мати інтерфейс, який дозволяє користувачам переглядати дані про рівень напруги та стан акумуляторної батареї.

3.2 Проєктування пристрою

Для створення моделі пристрою на Wokwi потрібно виконати кілька кроків. По-перше, я створив електричну схему свого пристрою за допомогою візуального конструктора, який надається сервісом. Далі, додав до своєї схеми необхідні елементи, які будуть використовуватися в моєму проєкті.

Перша схема яку я почав розробляти 3 місяці назад, виглядала дуже просто та примітивно.

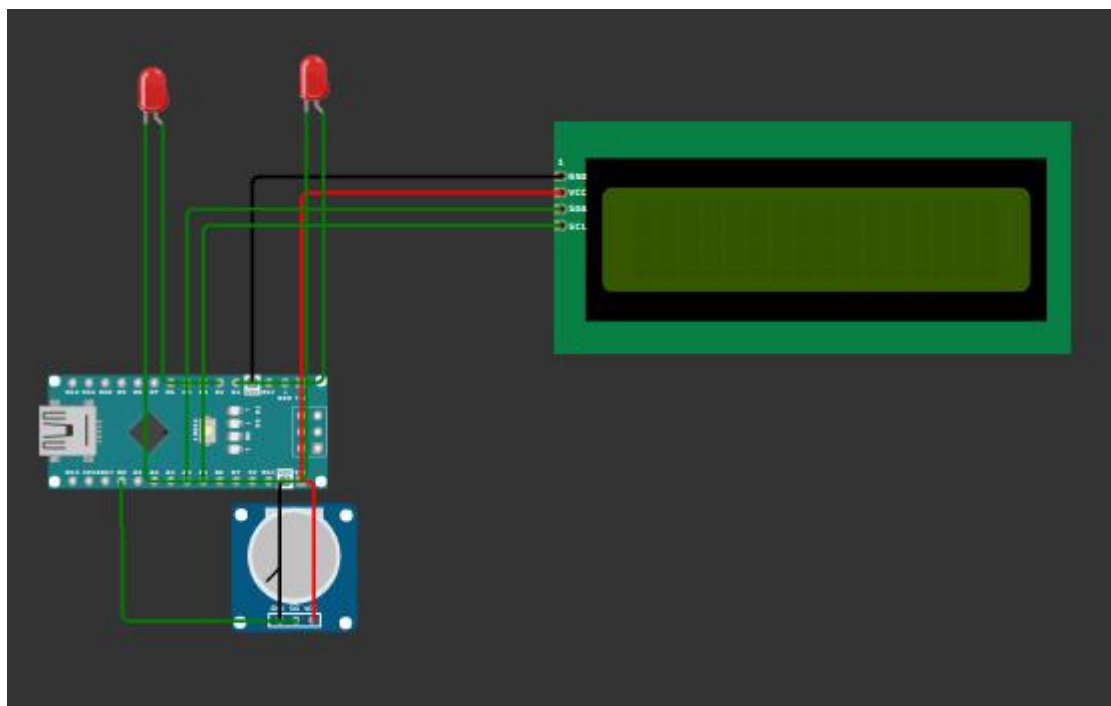


Рисунок 3.1- Зображення тестової схеми

На перших кроках я відточував алгоритми зчитування значень з потрібного входу.

Після багатьох спроб та розуміння процесів, я перейшов до ускладнення моєї схеми та додав головний компонент.

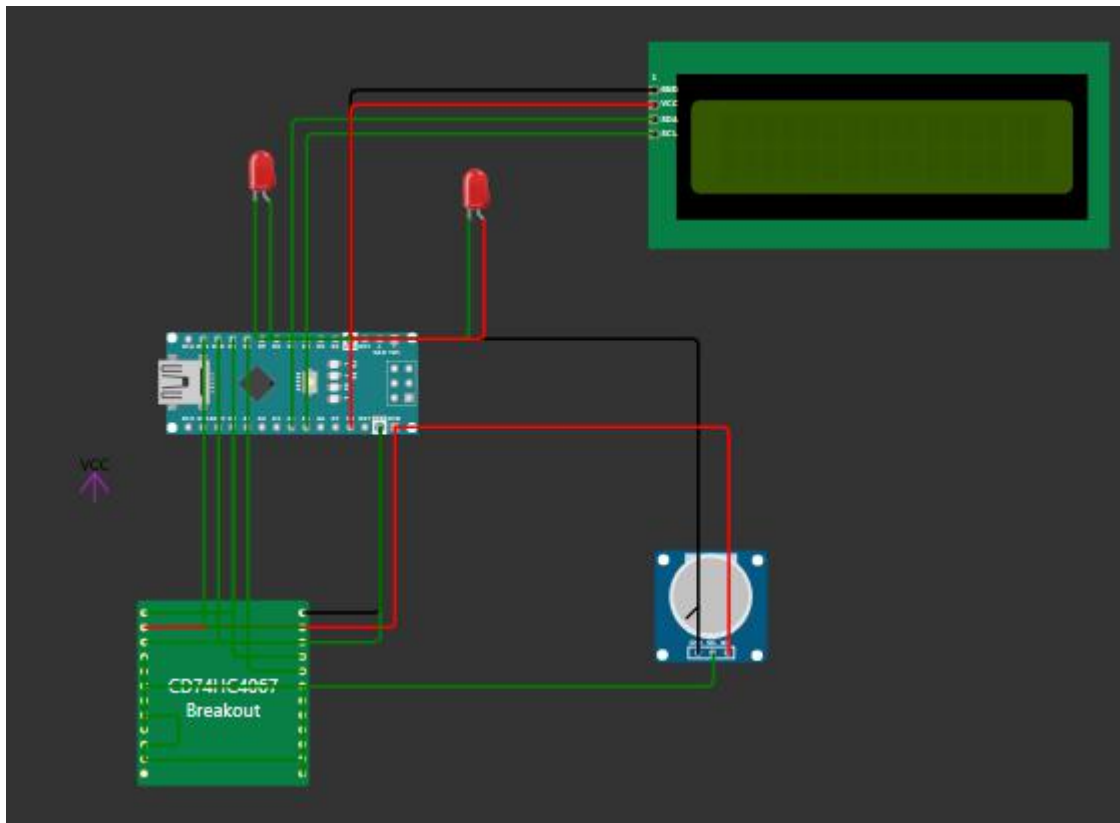


Рисунок 3.2- Зображення тестової схеми

На даному етапі проводилася симуляція схеми, за допомогою якої можна визначити, чи працює схема правильно. Симуляція дозволяла перевірити, як будуть працювати компоненти в різних ситуаціях та визначити ефективність та надійність схеми. Нажаль було дуже багато нюансів які я не передбачив що зайнято тривалий час для їх рішення. В результаті, я навчився працювати з мультиплексором без використання бібліотек та навчився визначати значенні генерації перемикаючих імпульсів за допомогою вираховуючої логіки, а не простого прописування даних.

По успішному завершенню симуляцій, я перейшов на етап отримання даних через мультиплексор.

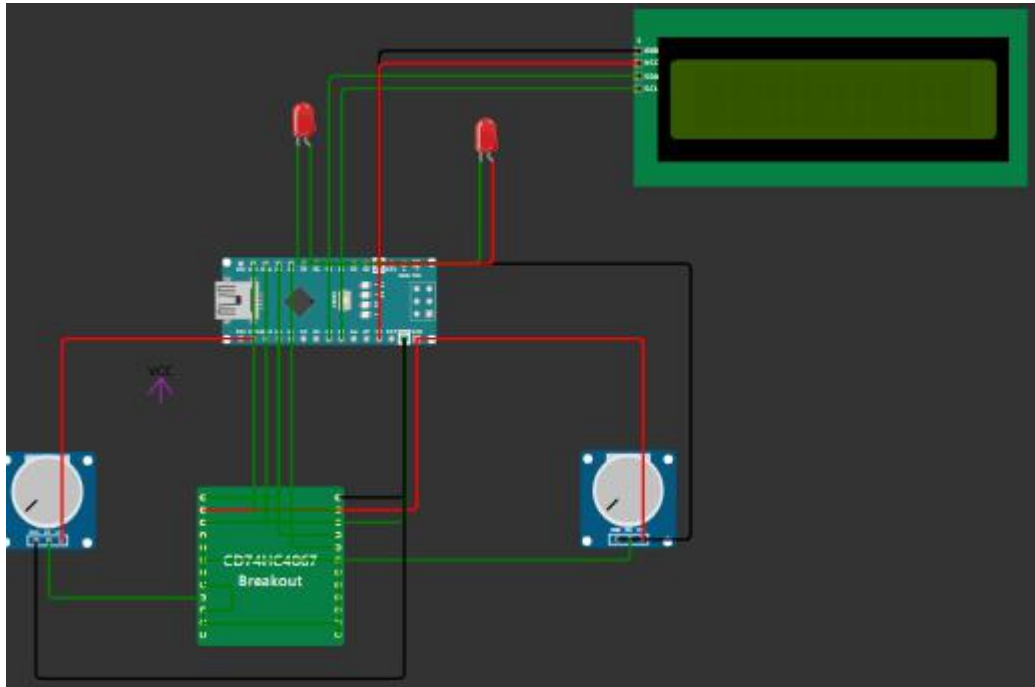


Рисунок 3.3- Зображення тестової схеми

На цьому етапі видно додаткові 2 елементи для проведення симуляцій та подальшого корегування коду. Після успішного закінчення даного етапу перейшов до фінального тестування.

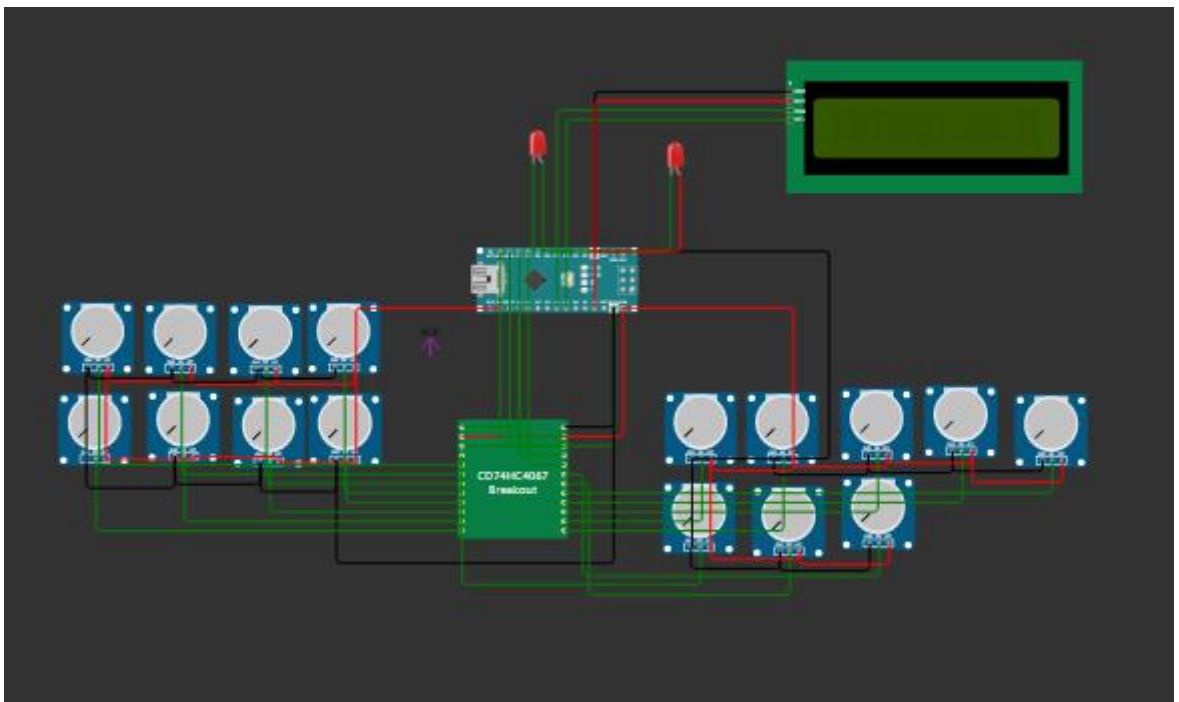


Рисунок 3.4- Зображення тестової схеми

Як видно із схеми, тут додалося багато компонентів для тестування кожного каналу та перевірки коду. Цей етап показав усі нюанси, в результаті чого мені довелося переписати повністю весь код. Було додано 3 масива для збереження даних, частково перероблена структура коду для вимірювання значень, кардинально змінена робота алгоритму перемикавання, з абсолютного нуля насаний алгоритм обробки, запису та обчислення «реальних» значень напруги, додані коефіцієнти корекції значень дільника напруги для кожної комірки та перероблена частина коду з перевірки умов. Сумарно в результаті проектування було більше восьми ревізій коду.

Фінальна схема виглядає наступним чином

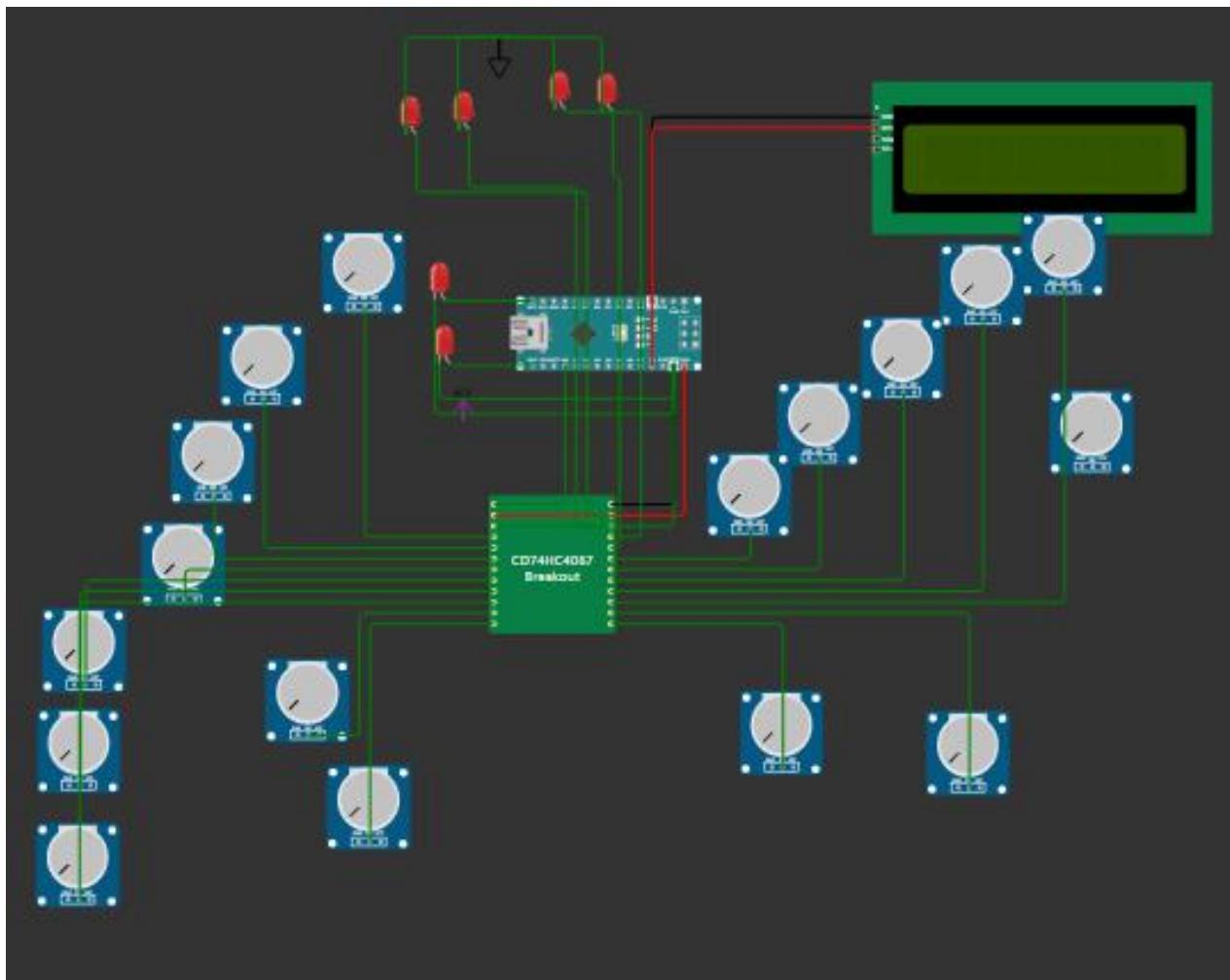
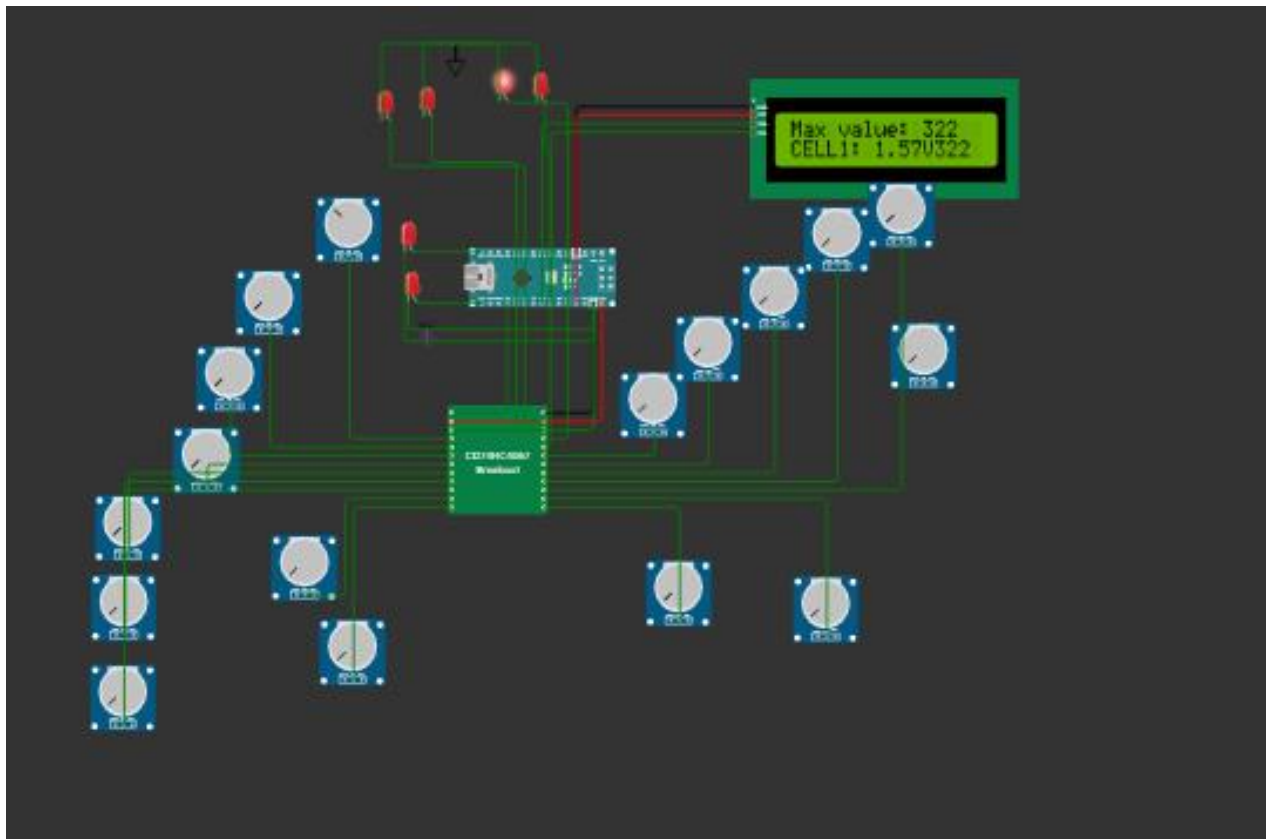


Рисунок 3.5- Зображення тестової схеми

Після багатьох перевірок можна розмістити скетч на мікроконтролері та запусити симуляцію.



Max value: -1.57
Voltage 1: 1.57V
Voltage 2: 0.00V

Рисунок 3.6- Зображення тестової схеми в симуляції

3.3 Написання програми для роботи пристрою

Мова програмування Arduino є спеціально розробленою мовою для програмування мікроконтролерів Arduino. Вона базується на мові C++, але має багато спеціальних функцій та бібліотек, що дозволяють легко зчитувати та записувати дані з різних датчиків, керувати різними актуаторами та взаємодіяти з іншими пристроями.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <CD74HC4067.h>

// Налаштування пінів
#define MUX_S0 4
#define MUX_S1 2
#define MUX_S2 8
#define MUX_S3 7
#define MUX_OUT A0
#define D2_PIN 12
#define D3_PIN 13

// Налаштування параметрів для виміру напруги
#define VOLTAGE_THRESHOLD_HIGH 3.6
#define VOLTAGE_THRESHOLD_LOW 2.9

// Налаштування параметрів для LCD
#define LCD_ADDRESS 0x27
#define LCD_COLS 16
#define LCD_ROWS 2
```

```

// Оголошення змінних
int SetMuxChannel(byte channel)
{
    digitalWrite(MUX_S0, bitRead(channel, 0));
    digitalWrite(MUX_S1, bitRead(channel, 1));
    digitalWrite(MUX_S2, bitRead(channel, 2));
    digitalWrite(MUX_S3, bitRead(channel, 3));
}

LiquidCrystal_I2C lcd(LCD_ADDRESS, LCD_COLS, LCD_ROWS);

const int num_channels = 17; // кількість каналів мультиплектора
int channel = 0; // індекс текущего канала мультиплектора
int readings[num_channels]; // Масив для зберігання даних
int coefficients[] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16};
int results[num_channels];
int diff[num_channels];

void setup() {
    // ініціалізація пінів
    pinMode(MUX_S0, OUTPUT);
    pinMode(MUX_S1, OUTPUT);
    pinMode(MUX_S2, OUTPUT);
    pinMode(MUX_S3, OUTPUT);
    pinMode(D2_PIN, OUTPUT);
    pinMode(D3_PIN, OUTPUT);

    // Ініціалізація LCD
    lcd.init();
    lcd.backlight();
}

```

```

// Инициализация Serial
Serial.begin(9600);
}
void loop() {
// Проходим по всем 16 аккумуляторам
for (int i = 1; i < 17; i++) {

SetMuxChannel (i);
// Выбираем аккумулятор

// Читаем напряжение
readings[i] = analogRead(MUX_OUT);
float voltage = readings[i] * 5.0 / 1023.0;
results[i] = readings[i] * coefficients[i];

// Выводим напряжение на LCD
lcd.setCursor(0, i);
lcd.print("CELL");
lcd.print(i);
lcd.print(": ");

lcd.print(voltage);
lcd.print("V");

// Выводим напряжение в Serial
Serial.print("Voltage ");
Serial.print(i);
Serial.print(": ");
Serial.print(voltage);

```

```
Serial.println("V");  
    delay(500);  
}
```

```
Serial.print("readings ");  
for (int i = 1; i < 17; i++) {
```

```
    Serial.print(readings[i]); // выводим значение на порт  
    Serial.print(" ");  
}
```

```
    Serial.println();  
Serial.print("results ");  
for (int i = 1; i < 17; i++) {  
    results[i]=results[i];
```

```
    Serial.print(results[i]); // выводим значение на порт  
    Serial.print(" ");  
}  
    Serial.println();
```

```
Serial.print("diff ");  
for (int i = 1; i < 17; i++) {
```

```
    diff[i] = results[i] - results[i-1]; // вичисляємо різницю між сусідніми  
значеннями масиву
```

```
    Serial.print(diff[i]); // виводим значення на порт  
    Serial.print(" "); // добавляем пробел між значеннями  
}
```

```

// Знаходимо максимальне і мінімальне значення в масиві
    int max_value = diff[1];
int min_value = diff[1];
for (int i = 1; i < 17; i++) {
    if (min_value > diff[i] ) {
        min_value = diff[i];
    }
    if (max_value < diff[i]) {
        max_value = diff[i];
    }
}

// Виводим максимальне і мінімальне значення в Serial і на LCD
Serial.println();
Serial.print("Max value: ");
Serial.println(max_value * 5.0 / 1023.0);
Serial.print("Min value: ");
Serial.println(min_value * 5.0 / 1023.0);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Max value: ");
lcd.print(max_value);
lcd.setCursor(0, 1);
lcd.print("Min value: ");
lcd.print(min_value);
// Провіряєм порогові значення
if (max_value * (5.0 / 1023.0) > VOLTAGE_THRESHOLD_HIGH) {
    digitalWrite(D3_PIN, HIGH);
}
else

```



```
{  
  digitalWrite(D3_PIN, LOW);  
}  
  //if (voltage < VOLTAGE_THRESHOLD_HIGH) {  
    // digitalWrite(D3_PIN, LOW);  
  // }  
  if (min_value * (5.0 / 1023.0) < VOLTAGE_THRESHOLD_LOW) {  
    digitalWrite(D2_PIN, LOW);  
    delay(500);  
  }  
  else  
  {  
    digitalWrite(D2_PIN, HIGH);  
  }  
  //if (voltage > VOLTAGE_THRESHOLD_LOW) {  
    //digitalWrite(D2_PIN, HIGH);  
  // }  
  delay(2);  
  //break;  
}
```

3.4 Моделювання плати пристрою

Проектування плати - це процес створення електронної схеми та плати, яка забезпечує з'єднання всіх компонентів електронної пристрої з електричними контактами.

Розробка схеми: на цьому етапі проектування визначається функціональність та принцип роботи пристрою.

Схема має наступний вигляд.

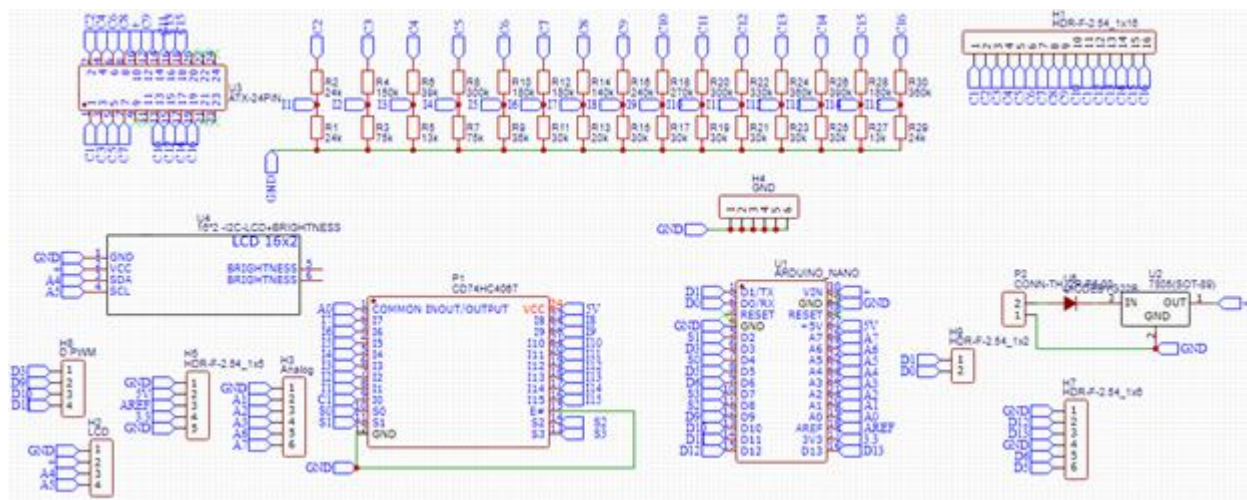


Рисунок 3.7- Зображення тестової схеми

Розміщення компонентів: наступним кроком є розміщення компонентів на платі. На цьому етапі важливо врахувати необхідну кількість елементів, їх розміри та орієнтацію, а також відстань між елементами та з'єднаннями.

Проведення маршрутування(трасування): після розміщення компонентів на платі необхідно під'єднати їх з'єднання за допомогою трасування. Маршрутизація полягає у виборі оптимальних маршрутів з'єднання між компонентами та створення відповідних електричних з'єднань.

Перевірка дизайну: після завершення маршрутизації необхідно перевірити дизайн плати на можливі помилки. Це можна зробити за допомогою спеціального програмного забезпечення, яке дозволяє перевірити електричні з'єднання та виявити можливі проблеми. В моєму випадку ПЗ виконало це автоматично, в результаті ми отримали розведену плату

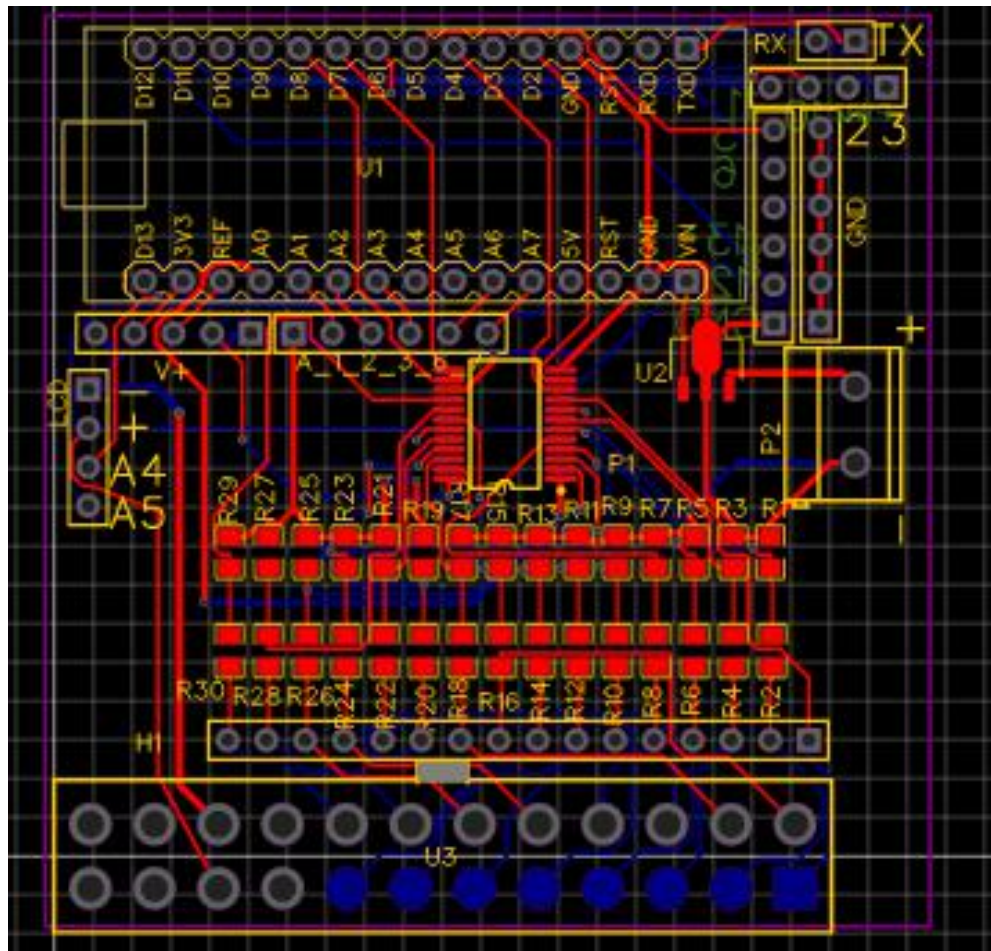


Рисунок 3.8- Зображення відтрасованої схеми

Виготовлення плати: після успішної перевірки дизайну плати можна передавати на виготовлення. Цей процес включає в себе виготовлення фотошаблонів та нанесення шарів міді на плату, вирізання та свердління отворів, монтаж компонентів і т.д. В результаті моєї роботи я отримав плати високої якості.

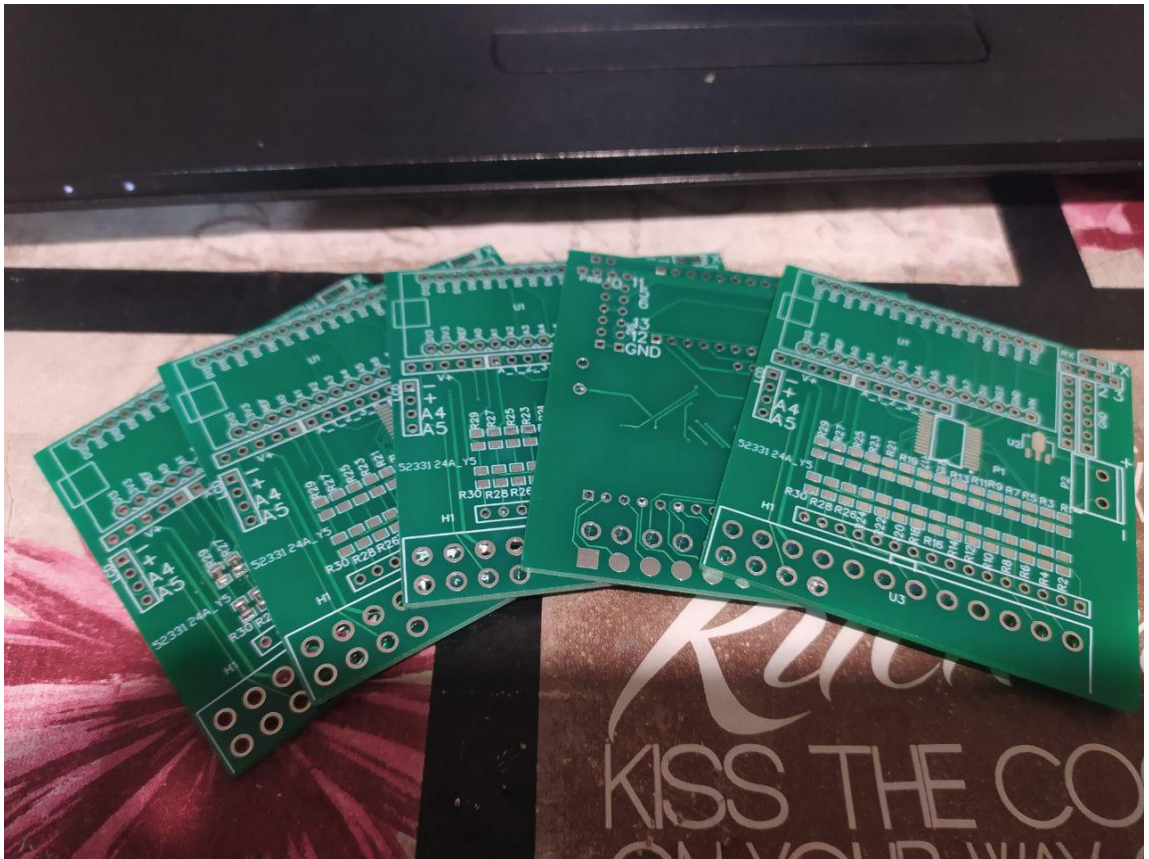


Рисунок 3.9- Готові плати

ВИСНОВОК

Створення системи моніторингу напруги було досить складним технічним завданням, яке вимагало багато часу, знань і досвіду в електроніці та програмуванні.

Система моніторингу напруги є дуже важливим елементом у забезпеченні безпеки літєвих батарей, оскільки відстеження напруги може допомогти попередити небезпечні ситуації, пов'язані з перезарядкою чи перерозрядкою батарей. Створення такої системи може мати додаткові переваги, які можуть бути корисними в різних областях. Наприклад, система моніторингу напруги може використовуватись у сонячних електростанціях, чи в електромобілях для контролю рівня заряду батарей.

У цій роботі я використав основні інструменти та програмні засоби, які використовуються при проектуванні плат, такі як Easy EDA, Wokwi, JLCPCB та Fusion 360.

В цій роботі я дослідив питання проектування та створення електронних плат, що є важливим елементом в багатьох сферах техніки та промисловості. Я розглянув основні етапи проектування плати, включаючи вибір компонентів, схематичний дизайн, вибір матеріалів, маршрутизацію та виробництво.

Підсумуємо, проектування та створення електронних пристроїв - це складний та важливий процес, який потребує високого рівня знань та досвіду у сфері електронної техніки та проектування. Однак, з правильними інструментами та методиками, створення високоякісного та функціонального пристрою може бути реалізовано відносно швидко та ефективно.

СПИСОК ЛІТЕРАТУРИ

1. Системи моніторингу батаней. [Електронний ресурс]- Режим доступу: <https://www.allaboutcircuits.com/industry-white-papers/battery-stack-monitor-maximizes-performance-of-li-ion-batteries-in-hybrid-and-electric-vehicles/>
2. Texas Instruments основи передачі даних. [Електронний ресурс]- Режим доступу: <https://www.ti.com/lit/ml/slua408a/slua408a.pdf?ts=1679841447447>
3. Відеохостинг YouTube [Електронний ресурс]- Режим доступу: <https://www.youtube.com/>
4. Характеристики компонентів [Електронний ресурс]- Режим доступу: <https://html.alldatasheet.com/html-pdf/82833/FAIRCHILD/LM7805/815/2/LM7805.html>
5. The Power Facts [Електронний ресурс]- Режим доступу: <https://thepowerfacts.com/which-battery-management-system-does-tesla-use/>
6. Tesla Model S Battery System: An Engineer's Perspective [Електронний ресурс]- Режим доступу: <https://circuitdigest.com/article/tesla-model-s-battery-system-an-engineers-perspective>
7. Powerwall [Електронний ресурс]- Режим доступу: <https://www.tesla.com/powerwall>
8. BATTERY MONITORING SYSTEM United States Patent Application 20200166579 [Електронний ресурс]- Режим доступу: <https://www.freepatentsonline.com/y2020/0166579.html>
9. LG Energy Storage System [Електронний ресурс]- Режим доступу:
10. https://www.lg.com/global/business/download/resources/ess/LG%20ESS_Residential_Datasheet_EN.pdf
11. StorEdge™ On-Grid Solution [Електронний ресурс]- Режим доступу: <https://www.solaredge.com/solutions/self-consumption#/>
12. Battery University [Електронний ресурс]- Режим доступу: <https://batteryuniversity.com/article/bu-903-how-to-measure-state-of-charge>

13. Texas Instruments [Електронний ресурс]- Режим доступу:
https://www.ti.com/lit/eb/slyt701/slyt701.pdf?ts=1684308938888&ref_url=https%253A%252F%252Fwww.bing.com%252F
14. Analog Devices, Inc. [Електронний ресурс]- Режим доступу:
<https://www.analog.com/en/index.html>
15. ResearchGate. Invertibilidade global e equilibrio econômico geral [Електронний ресурс]- Режим доступу:
https://www.researchgate.net/publication/334583191_Invertibilidade_global_e_equilibrio_economico_geral
16. PlimTex. bms qvantex monbat 3 [Електронний ресурс]- Режим доступу:
<https://plimtex.com.ua/monitoring-batarej/>
17. E-energy. Система моніторингу акумуляторних батарей [Електронний ресурс]- Режим доступу: https://e-energy.in.ua/related_products/sistema-monitoringa-akkumulyatornyh-batarej-ee-bms-e1-battery-monitoring-system.html
18. Маркетплейс [Електронний ресурс]- Режим доступу:
<https://www.aliexpress.com/>