

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних технологій
циклова комісія інформатики та комп'ютерних дисциплін

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач відділенням

комп'ютерних технологій

Наталія СТЕФУРАК / _____ /

(підпис)

« ____ » _____ 2024 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи

освітньо-професійного ступеня «фаховий молодший бакалавр»

зі спеціальності 123 «Комп'ютерна інженерія»

на тему: «Система обліку споживання електроенергії у приватному будинку»

Студент групи KI-41

Володимир КОЦУР

(підпис)

Керівник проєкту

Степан ІВАСЬЄВ

(підпис)

Консультанти:

з техніко-економічного

обґрунтування

Любов МЕЛЕНЧУК

(підпис)

Нормоконтролер

Ольга СЛЄПЦОВА

(підпис)

Тернопіль – 2024

Галицький фаховий коледж імені В'ячеслава Чорновола
відділення комп'ютерних технологій
циклова комісія інформатики та комп'ютерних дисциплін

ЗАТВЕРДЖУЮ

Завідувач відділенням

комп'ютерних технологій

Наталія СТЕФУРАК / _____ /

(підпис)

« ____ » _____ 2023 р.

ЗАВДАННЯ

на кваліфікаційну роботу

на здобуття освітньо-професійного ступеня «фаховий молодший бакалавр»

студенту Коцуру Володимиру Сергійовичу

(прізвище, ім'я та по-батькові студента)

1. Тема роботи: «Система обліку споживання електроенергії у приватному будинку», затверджена наказом по коледжу від “27” листопада 2023 р., №234а-н.
2. Термін здачі студентом завершеної роботи “28” червня 2024 р.
3. Вихідні дані до роботи: Система обліку електроспоживання у приватному будинку.
4. Перелік питань, які повинні бути розроблені в роботі:
 - а) основна частина: встановлення та формалізація вимог, проєктування інтерфейсу, програмна реалізація застосунку, тестування.
 - б) техніко-економічне обґрунтування: аналіз ринку збуту, обґрунтування витрат на проєктування, обґрунтування необхідності розробки;
5. Перелік графічного матеріалу: діаграма послідовностей, діаграма станів, діаграма варіантів використання.

6. Консультанти роботи:

Розділ	Консультанти	Підпис, дата	
		Завдання видано	Завдання прийнято
З техніко-економічного обґрунтування	Меленчук Л.І		
	(вчена ступінь, звання П.І.Б.		
	консультанта)		

КАЛЕНДАРНИЙ ПЛАН

виконання кваліфікаційної роботи

№ п/п	Найменування етапу	Терміни	
		початку	завершення
1.	Вибір теми кваліфікаційної роботи	23.11.2023	01.12.2023
2.	Детальне ознайомлення з предметною областю. Аналіз наявних рішень	02.12.2023	05.02.2024
3.	Опрацювання теоретичних матеріалів, написання розділу роботи	29.01.2024	07.02.2024
4.	Формалізація вимог. Аналіз технологій реалізації. Написання розділу роботи	08.02.2024	01.03.2024
5.	Проектування системи обліку. Аналіз функціональних вимог	03.03.2024	05.04.2024
6.	Виготовлення макету системи обліку електроспоживання в приватному будинку	18.03.2024	08.04.2024
7.	Створення відповідного розділу роботи	09.04.2024	09.05.2024
8.	Обґрунтування вартості проєкту	10.05.2024	16.05.2024
9.	Оформлення пояснювальної записки	11.05.2024	17.05.2024
10.	Попередній захист кваліфікаційної роботи	18.06.2024	18.06.2024
11.	Підготовка до захисту та виправлення помилок	19.06.2024	21.06.2024
12.	Захист кваліфікаційної роботи	28.06.2024	28.06.2024

Дата видачі “27” листопада 2023 р. Керівник _____ / Степан ІВАСЬЄВ

Завдання прийняв до виконання _____ / Володимир КОЦУР

Реферат

Кваліфікаційна робота. Система обліку споживання електроенергії у приватному будинку. 54 с., 10 рисунків, 1 додаток, 7 джерел.

Об'єкт дослідження – системи обліку споживання електроенергії у приватному будинку.

Метою роботи є розробка програмного забезпечення системи обліку електроспоживання у приватному будинку, яка базується на сучасних технологіях. Програмний засіб повинен забезпечувати зручний спосіб взаємодії для користувача, а також оптимізований і точний процес моніторингу та обліку споживання електроенергії.

Крім того, необхідно спроектувати та реалізувати можливість налаштування параметрів моніторингу залежно від потреб користувача та умов використання, що повинно забезпечити зручність для користувачів.

Для досягнення поставленої мети використано різноманітні технології та інструменти для взаємодії з сенсорами та системами моніторингу. Після завершення процесу розробки було отримано функціональний та готовий до використання програмний засіб.

СИСТЕМА ОБЛІКУ, ESP32, ДАТЧИКИ, ІНТЕРНЕТ РЕЧЕЙ, BLYNK.IO.

Abstract

Graduation Project: Electricity consumption accounting system in a private house. 54 pages, 10 figures, 1 appendice, 7 sources.

The object of research is the system of electricity consumption metering in a private house. The purpose of the work is to develop software for a system of electricity consumption metering in a private house based on modern technologies.

The software should provide a convenient way of interaction for the user, as well as an optimized and accurate process of monitoring and accounting for electricity consumption.

In addition, it is necessary to design and implement the ability to customize monitoring parameters depending on the user's needs and conditions of use, which should provide convenience for users.

To achieve this goal, various technologies and tools were used to interact with sensors and monitoring systems. Upon completion of the development process, a functional and ready-to-use software tool was obtained.

ACCOUNTING SYSTEM, ESP32, SENSORS, INTERNET OF THINGS,
BLYNK.IO

ЗМІСТ

Скорочення та умовні позначки	6
Вступ.....	7
1 Аналіз предметної області та постановка завдання.....	8
1.1 Змістовний аналіз предметної області	8
1.2 Аналіз наявних рішень	9
1.3 Огляд датчиків.....	12
1.4 Постановка завдання.....	17
2 Проектування системи	20
2.1 Проектування функцій системи.....	20
2.2 Визначення компонентів системи	21
2.3 Проектування структури системи	23
2.3 Алгоритм роботи системи	25
3 Реалізація та тестування системи	28
3.1 Вибір компонентів	28
3.2 Реалізація системи.....	33
3.3 Реалізація програмного коду	35
3.4 Підключення та налаштування Blynk.io	40
3.5 Тестування реалізованого пристрою.....	42
4 Техніко-економічне обґрунтування	44
4.1 Аналіз ринку	44
4.2 Розрахунок витрат на розробку проекту	44
4.3 Обґрунтування необхідності розробки	46
Висновки	47
Перелік джерел посилання	48
Додатки.....	49

КР.КІ 24.532.05.000 ПЗ

Зм.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Коцур В.С.			Система обліку споживання електроенергії у приватному будинку	Лім.	Арк.
Перев.		Івасьєв С.В.					5
Рецензент.		Павлюс В.П.				Архивів	54
Н. Контр.		Слетцова О.Я.				ГФК.ВКТ.КІ-41	
Зав. від.		Стефурак Н.А.					

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

VCC – Voltage at the Common Collector

GND – Ground

GPIO – General Purpose Input/Output

Wi-Fi – Wireless Fidelity

Bluetooth – Named after Harald Bluetooth

IoT – Internet of Things\

IDE – Integrated Development Environment

					КР.КІ 24.532.05.000 ПЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Наразі в Україні енергозбереження та контроль за споживанням електроенергії у приватних будинках стають дедалі важливими аспектами. Ці завдання є важливими для сталого енергетичного розвитку країни.

Проблема енергоефективності в Україні є складною та багатогранною. Важливо забезпечити високий рівень енергоефективності у всіх сферах життя, а особливо у приватному секторі. Для цього необхідно впроваджувати нові технології та розробляти ефективні заходи з енергозбереження.

Одним зі способів підвищення енергоефективності є впровадження систем обліку споживання електроенергії. Ці системи дозволяють ефективно контролювати та оптимізувати використання енергетичних ресурсів. Наприклад, можна розглянути підвищення рівня енергетичної ефективності в будинках та розробити заходи з енергозбереження.

Однак, для успішної реалізації таких проектів необхідна належна кваліфікація спеціалістів у галузі енергозбереження. Тому однією з важливих задач є підготовка кваліфікованих енергоаудиторів та енергоменеджерів, які можуть провести енергоаудит та запровадити систему енергетичного менеджменту у будинках.

Таким чином, впровадження систем обліку та моніторингу електроенергії є важливим кроком у підвищенні енергоефективності та забезпеченні сталого енергетичного розвитку України.

					КР.КІ 24.532.05.000 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ

1.1 Змістовний аналіз предметної області

У сучасних умовах енергозбереження та енергоефективність стають ключовими аспектами для розвитку України. Контроль за споживанням електроенергії в будинках та на підприємствах відіграє важливу роль у забезпеченні сталого енергетичного розвитку країни.

Наразі використання систем обліку споживання електроенергії у приватних будинках є досить розповсюдженим, особливо в районах, де відсутнє централізоване електропостачання або коли власники будинків використовують альтернативні джерела електроенергії. Більшість таких систем базуються на технології вимірювання споживаної електроенергії, яка дозволяє власникам будинків ефективно контролювати та оптимізувати використання енергетичних ресурсів.

Проте, в більшості випадків ці системи не забезпечують можливості віддаленого моніторингу або керування, що може бути незручним для власників будинків у випадках, коли вони перебувають далеко від дому або відсутні на місці. Деякі сучасні рішення включають в себе «розумні» системи обліку, які можуть бути керовані за допомогою мобільних застосунків. Проте, не зважаючи на розвиток технологій, існують певні проблеми з надійністю та точністю деяких таких систем.

Для успішного впровадження систем обліку споживання електроенергії в приватних будинках необхідно врахувати потреби та можливості власників будинків, а також забезпечити надійність та доступність цих систем.

Облік електроенергії є важливим процесом, що забезпечує точне вимірювання спожитої електроенергії для подальшого аналізу та управління. Цей процес здійснюється за допомогою електролічильників, які використовуються як в приватних будинках, так і на підприємствах. Електролічильники можуть бути різних типів, залежно від технології та способу вимірювання.

					КР.КІ 24.532.05.000 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

Механічні лічильники вимірюють кількість обертів диска, що обертається під впливом електромагнітного поля, створеного струмом. Цей метод обліку є надійним та перевіреним часом, однак він поступається в точності сучасним електронним лічильникам. Механічні лічильники широко використовуються завдяки своїй простоті та відносно низькій вартості.

Електронні лічильники використовують електронні компоненти для вимірювання та запису спожитої електроенергії. Вони часто забезпечують більшу точність і мають можливість віддаленого моніторингу, що дозволяє користувачам контролювати споживання електроенергії в режимі реального часу. Ці лічильники стають все більш популярними завдяки своїм перевагам, таким як висока точність, зручність використання та можливість інтеграції з іншими "розумними" системами управління енергією [1].

1.2 Аналіз наявних рішень

Існуюча автоматизована система комерційного обліку електроенергії (АСКОЕ) (рис. 1.1) дозволяє вимірювати, збирати, обробляти та відображати дані про споживання електроенергії в певному місці протягом заданого часу. Ця система орієнтована на комерційне використання, спрямоване на оптимізацію енергетичних процесів і забезпечення ефективного електропостачання для підприємств та інших комерційних об'єктів.

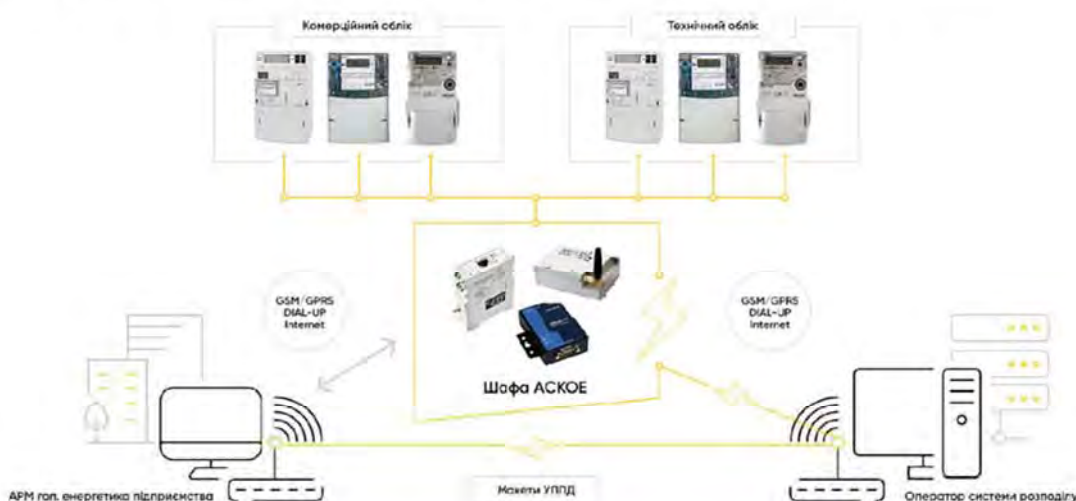


Рисунок 1.1 – Автоматизована система комерційного обліку електроенергії

Змн.	Арк.	№ докум.	Підпис	Дата

КР.КІ 24.532.05.000 ПЗ

Арк.

9

АСКОЕ виступає як один з ключових інструментів для зниження витрат на електроенергію та підвищення енергоефективності підприємства. Зібрані дані використовуються для розрахунків за фактично спожиту електроенергію, фіксуючи кожен кіловат-годину, за яку здійснюється оплата[2].

Завдяки надійному постачальнику послуг комерційного обліку власники підприємств можуть отримати додаткові переваги, включаючи рекомендації щодо ефективного енергоменеджменту. Основні переваги АСКОВЕ включають:

Система гарантує високу точність вимірювань показників електроенергії, що запобігає помилкам у розрахунках споживання.

АСКОЕ може бути інтегрована з іншими системами управління, що дозволяє автоматизувати процеси споживання електроенергії, такі як автоматичне вимкнення освітлення або обладнання під час неактивних періодів.

Зібрані дані можуть бути проаналізовані для виявлення тенденцій, закономірностей та прогнозування майбутнього споживання, що допомагає приймати обґрунтовані рішення щодо енергозабезпечення.

Система дозволяє відстежувати споживання електроенергії в режимі реального часу, що допомагає зрозуміти, як і де використовується електроенергія, і впровадити заходи енергоефективності для зниження витрат.

АСКОЕ дозволяє отримувати та зберігати дані про споживання електроенергії в різний час доби, що дозволяє планувати процеси так, щоб використовувати електроенергію в періоди з нижчими тарифами, сприяючи зниженню витрат на електроенергію.

Загалом, АСКОВЕ є ефективним інструментом для управління енергоспоживанням, що забезпечує точний облік, автоматизацію процесів, аналіз даних, контроль витрат та оптимізацію тарифів, сприяючи підвищенню енергоефективності та зниженню витрат на електроенергію для комерційних об'єктів.

					КР.КІ 24.532.05.000 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

Однак, для приватних будинків та особистого використання існуюча система АСКОЕ може бути надто потужною та складною. У зв'язку з цим, для побутових споживачів, таких як власники приватних будинків, може бути більш доцільним розглядати спеціалізовані системи обліку споживання електроенергії, призначені для особистого користування. Такі системи можуть бути більш доступними, простішими у встановленні та експлуатації, а також більш адаптованими до потреб та можливостей власників приватних будинків.

ЛУЗОД (локальне устаткування збору і обробки даних) – це система, призначена для розрахунків за використану електричну енергію. Вона складається з різних пристроїв обліку, які дозволяють здійснювати вимірювання, збір, накопичення та обробку інформації щодо обсягів та параметрів електричних потоків, а також значень спожитої потужності на певній точці вимірювання протягом заданих періодів часу. Система має інтерфейс для дистанційного зчитування даних і може працювати в складі автоматизованої системи комерційного обліку[3].

Ключова відмінність між ЛУЗОД (рис. 1.2) і АСКОЕ (рис. 1.3) полягає в тому, що користувачі ЛУЗОД не мають можливості віддаленого перегляду та контролю даних про споживання електроенергії та потужності. Цю інформацію зберігає лише обленерго, тоді як система АСКОЕ компанії збирає дані з усіх ЛУЗОД і АСКОЕ своїх користувачів.

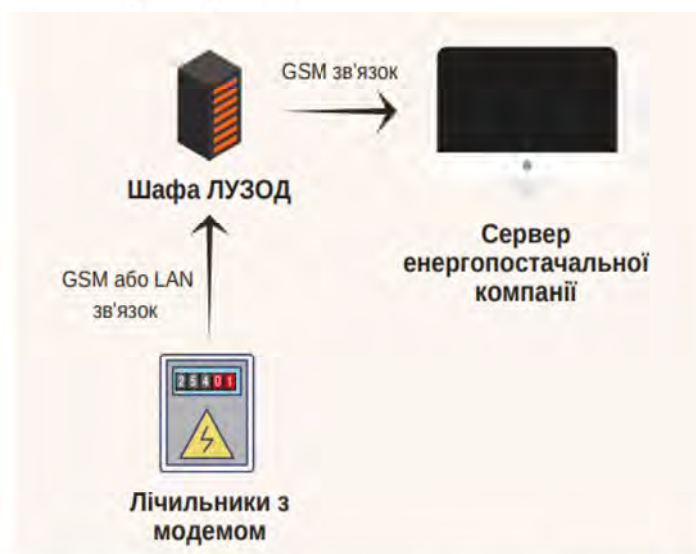


Рисунок 1.2 – Візуальна схема ЛУЗОД

Переваги які отримує споживач від використання АСКОЕ:

- Можливість доступу до даних про споживання електроенергії та потужності в будь-який момент і з будь-якою періодичністю.
- Здатність аналізувати споживання електроенергії та прогнозувати витрати на неї.
- Зберігання та надання інформації про споживання електроенергії та потужності в зручному для аналізу форматі.
- Можливість самостійного контролю за лімітами споживання електроенергії або потужності під час пікових навантажень на енергосистему.



Рисунок 1.3 – Візуальна схема АСКОЕ

Отже, АСКОЕ краще ніж ЛУЗОД, оскільки надає можливість віддаленого доступу до даних про споживання електроенергії, що дозволяє користувачам ефективніше контролювати та оптимізувати свої витрати[4].

1.3 Огляд датчиків

Датчики рівня напруги та струму: Ці датчики можуть бути встановлені на електричних панелях або окремих електроприладах для вимірювання електроенергії, що споживається.

На ринку доступні різні датчики та рішення для систем обліку споживання електроенергії в будинках. Ось кілька прикладів:

Наприклад, датчики Aeon Labs ZW095 Home Energy Meter (рис. 1.4) або Aeotec Z-Wave Home Energy Meter.



Рисунок 1.4 – Вимірювач електроенергії (3 кліпси) на 60А Aeotec Home Energy Meter Gen5

Aeon Labs Home Energy Meter може використовуватися для відстеження споживання електроенергії на трифазному вводі в будинок або квартиру через зняття показань з фазних провідників струмовим способом (одяганням кліпси струмового трансформатора на провідник).

Цей метод підключення вимірювача потужності не потребує розбирання електричних проводів і може бути виконаний власноруч, без проведення складних електромонтажних робіт. Home Energy Meter миттєво інформуватиме центральний контролер або диспетчерський пункт про значення спожитої потужності у Вт та кВт*год. Пристрій підключається безпосередньо до фаз, що дозволяє максимально точно вимірювати напругу і потужність[5].

Також, Home Energy Meter підтримує інтеграцію з іншими смарт-пристроями, що дозволяє створити комплексну систему моніторингу енерговитрат у вашому домі. Пристрій має вбудований дисплей для відображення поточних даних про споживання електроенергії, що робить його зручним для користувачів. Завдяки використанню сучасних технологій, цей

лічильник забезпечує високий рівень точності та надійності, що робить його незамінним для енергоефективного управління домом.

Системи моніторингу споживання електроенергії: Ці системи включають в себе датчики та контролери, які дозволяють вимірювати та відображати дані про споживання електроенергії. Наприклад, система Sense Energy Monitor (рис. 1.5) або Emporia Vue Smart Home Energy Monitor.



Рисунок 1.5 – Принцип роботи Sense Home Energy Monitor за допомогою програми на телефоні.

За допомогою програми на вашому телефоні монітор допоможе вам встановити та відстежувати цілі споживання енергії. Занепокоєння про те, що ви залишили праску увімкненою або зачинили двері гаража, залишилися в минулому - додаток відстежуватиме все та попереджатиме вас, щоб забезпечити вашу безпеку, ефективність і, зрештою, заощадити ваші гроші. Додаток дозволяє налаштувати спеціальні сповіщення та контролювати використання 24/7. Монітор також сумісний з пристроями Amazon Alexa та Google Home [6].

Neurio – це потужне рішення для моніторингу та управління енергоспоживанням будинку, яке повідомляє про реальне використання енергії в будинку та допомагає власникам будинків отримувати інтелектуальні дані для прийняття розумних енергетичних рішень (рис. 1.6). Neuroio може контролювати окремі електроприлади в будинку, а також генерацію сонячної енергії. Інформація, зібрана Neuroio, зберігається на хмарній платформі, і до неї можна отримати доступ через смартфон або будь-який пристрій, підключений до Інтернету.



Рисунок 1.6 – Neuroio Home Energy Monitor комплект для моніторингу та обліку електроспоживання

Блок моніторингу Neuroio встановлюється всього за 15 хвилин, при цьому базовий блок розміщується всередині основної електричної панелі або в окремому корпусі. Трансформатори струму розміщуються навколо ліній електроживлення та проводів від основного контуру сонячного виходу, а потім блок підключається через Wi-Fi до мережі власника будинку. Надається невелика зовнішня антена, яку можна розмістити за межами корпусу, покращуючи потужність сигналу.

У Neuroio є функція «навчання ваших приладів», яка фактично дозволяє вам вибирати серед різних електронних пристроїв у вашому домі, які

					КР.КІ 24.532.05.000 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

споживають щонайменше 400 Вт або потужність. Наразі він не може відрізнити прилади потужністю менше 400 Вт, до яких належать лампи, комп'ютери, телевізори та навіть деякі великі прилади, як-от холодильники. Щоб побачити, скільки енергії споживають пристрої потужністю менше 400 Вт, єдиний вихід - уважно спостерігати за монітором у реальному часі.

Shelly EM – це готовий WiFi модуль, який дозволяє вимірювати струм у двох окремих ланцюгах і управляти контактором на 230 В з максимальним навантаженням до 2 А за допомогою віддаленого керування (рис. 1.7). Цей розумний енергетичний лічильник дозволяє вам дистанційно контролювати споживання електроенергії у вашому домі або на підприємстві, а також стежити за кількістю енергії, виробленої вашою сонячною електростанцією. Для вимірювання струму використовуються спеціальні вимірювальні трансформатори.

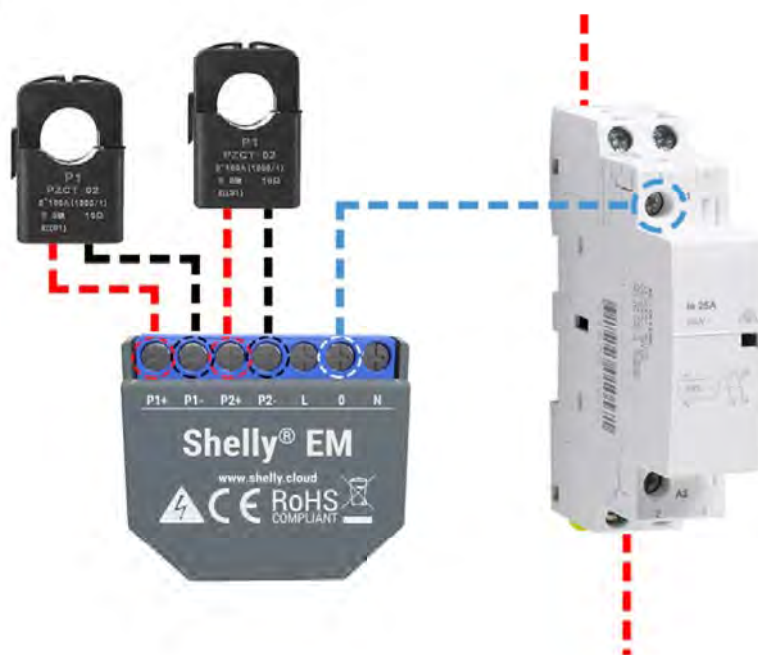


Рисунок 1.7 – Вигляд Shelly EM та приклад встановлення лічильника

Shelly EM підтримує зберігання даних вимірювань протягом цілого року (365 днів), що дозволяє аналізувати споживання електроенергії за тривалий період. Модуль може інтегруватися у додаток Shelly або популярні системи домашньої автоматизації, такі як Home Assistant, Domoticz, openHAB, через протокол MQTT. Це забезпечує широкі можливості для налаштування та управління енергоспоживанням.

					КР.КІ 24.532.05.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

Завдяки можливості віддаленого керування, ви можете вмикати або вимикати пристрої, підключені до контактора, що сприяє підвищенню енергоефективності. Пристрій також підтримує налаштування повідомлень про перевищення встановлених лімітів споживання, що допомагає уникнути перевантаження електромережі та додаткових витрат [6].

Shelly EM легко встановлюється і налаштовується, що робить його ідеальним вибором для користувачів, які прагнуть контролювати та оптимізувати своє енергоспоживання без значних витрат на обладнання та монтаж.

Системи розумного будинку з функцією енергомоніторингу такі як Nest або Ecobee, мають вбудовані функції моніторингу споживання електроенергії, які можна підключити до інших датчиків та пристроїв для більш точного контролю. Ці прилади можуть бути встановлені та інтегровані в систему управління енергозбереженням для приватного будинку, що дозволить власникам ефективно контролювати та оптимізувати своє споживання електроенергії.

1.4 Постановка завдання

Процес проектування системи обліку споживання електроенергії в приватному будинку є складним та багатоетапним завданням, яке вимагає глибокого розуміння технічних аспектів, а також потреб користувача. На першому етапі необхідно визначити основні цілі та задачі, які повинна вирішувати система.

Це включає не лише вимірювання і облік споживаної електроенергії, але й забезпечення можливості аналізу даних, прогнозування споживання, автоматизації процесів та інтеграції з іншими системами управління будинком. Важливо також врахувати такі аспекти, як точність вимірювань, надійність передачі даних, зручність користування системою та можливість її масштабування.

Перш за все, необхідно провести детальний аналіз потреб користувача.

					КР.КІ 24.532.05.000 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

Це включає збір інформації про звички та режим використання електроенергії в будинку, типи та кількість електричних приладів, пікові години споживання, середнє та максимальне навантаження на електромережу. Також важливо зазначити, які саме показники і з якою частотою мають бути виміряні та передані для подальшого аналізу. Визначення цих параметрів дозволить створити систему, яка найбільш точно відповідатиме потребам користувача і забезпечує максимальну ефективність її роботи.

Наступним кроком є визначення технічних вимог до системи. Це включає вибір типу лічильників електроенергії (наприклад, смарт-лічильники або багатотарифні лічильники), які забезпечують високу точність вимірювань і можливість передачі даних у реальному часі. Крім того, необхідно вибрати відповідні датчики потужності для моніторингу споживання окремими приладами, а також розробити систему збору, обробки і зберігання даних. Важливо також врахувати методи передачі даних, які можуть бути провідними або бездротовими, залежно від особливостей будинку і побажань користувача.

Після визначення технічних вимог необхідно розробити загальну архітектуру системи. Це включає визначення всіх компонентів системи, таких як лічильники, датчики, контролери, комунікаційні модулі та сервери для зберігання даних, а також їх взаємодію між собою. На цьому етапі важливо забезпечити надійність і безперебійність роботи системи, а також можливість її інтеграції з іншими системами управління будинком, наприклад, системами опалення, вентиляції та кондиціонування повітря.

Один з ключових етапів проектування – це вибір обладнання. Від правильного вибору лічильників, датчиків та комунікаційних модулів залежить точність вимірювань, надійність передачі даних та зручність користування системою. Необхідно врахувати не лише технічні характеристики обладнання, але й його вартість, можливість оновлення та розширення в майбутньому, а також сумісність з іншими компонентами системи.

Інтеграція всіх компонентів системи є наступним важливим кроком. Це включає підключення датчиків до лічильників або контролерів, налаштування

					КР.КІ 24.532.05.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

комунікаційних модулів для безперебійної передачі даних, встановлення програмного забезпечення для збору, обробки та аналізу даних, а також налаштування інтерфейсів користувача. Важливо забезпечити зручність користування системою, інтуїтивно зрозумілий інтерфейс та можливість віддаленого доступу до даних.

Після інтеграції системи необхідно провести її тестування для перевірки працездатності та точності вимірювань. Це включає тестування окремих компонентів, перевірку цілісності та точності переданих даних, випробування системи під різними навантаженнями для оцінки її стабільності, а також налаштування порогових значень та сповіщень для контролю за аномаліями у споживанні електроенергії.

Після успішного тестування система впроваджується в експлуатацію. Це включає установку всіх компонентів у будинку користувача, проведення початкового налаштування та введення в експлуатацію, навчання користувача основам роботи з системою, пояснення функцій інтерфейсу користувача, а також забезпечення технічної підтримки та обслуговування системи.

Після впровадження системи важливо забезпечити її регулярний моніторинг та оптимізацію. Це включає постійний аналіз даних про споживання електроенергії для виявлення можливостей підвищення ефективності, регулярне оновлення програмного забезпечення для покращення функціональності та безпеки, а також налаштування нових правил та алгоритмів для автоматизації управління енергоспоживанням.

Таким чином, процес проектування системи обліку споживання електроенергії в приватному будинку є багатоетапним і вимагає комплексного підходу, врахування потреб користувача, визначення технічних вимог, вибору обладнання, інтеграції, тестування, впровадження та постійного моніторингу. Всі ці етапи є критичними для створення ефективної та надійної системи, яка допоможе знизити витрати на електроенергію та підвищити енергоефективність.

2 ПРОЄКТУВАННЯ СИСТЕМИ

2.1 Проєктування функцій системи

Проєктування системи обліку споживання електроенергії в будинку передбачає ретельний аналіз потреб споживачів та вибір оптимальних технологій для забезпечення ефективного контролю та управління споживанням електроенергії. Цей процес включає кілька важливих етапів, кожен з яких є критичним для створення надійної та ефективної системи.

По-перше, необхідно зібрати детальну інформацію про поточне споживання електроенергії в будинку. Це включає вивчення звичок споживання мешканців, визначення пікових годин використання електроенергії, а також виявлення основних джерел споживання, таких як освітлення, опалення, кондиціонування повітря, побутова техніка та інші електричні пристрої. Такий аналіз дозволяє визначити конкретні потреби кожного споживача і налаштувати систему відповідно до цих потреб.

По-друге, вибір оптимальних технологій є ключовим аспектом проєктування системи. Це включає визначення типу лічильника електроенергії, який буде використовуватись у системі. Наприклад, смарт-лічильники надають можливість віддаленого моніторингу та керування, а також автоматичного збору даних, що значно підвищує ефективність управління енергоспоживанням. Також важливо врахувати можливості інтеграції з іншими системами управління будинком, такими як системи безпеки, опалення та кондиціонування повітря, для створення єдиної інтелектуальної екосистеми.

По-третє, необхідно розробити інтерфейс користувача, який забезпечить зручний доступ до даних про споживання електроенергії та дозволить користувачам легко керувати своїм енергоспоживанням. Інтерфейс має бути інтуїтивно зрозумілим, з візуально привабливими графіками та діаграмами, що показують споживання електроенергії в режимі реального часу, а також можливістю налаштування різних параметрів системи відповідно до індивідуальних потреб користувача.

Нарешті, визначення методів збору та аналізу даних є важливою складовою процесу проектування. Дані про споживання електроенергії повинні збиратись автоматично та зберігатись у зручному для аналізу форматі. Це дозволить користувачам та адміністраторам системи здійснювати детальний аналіз споживання, виявляти тенденції та аномалії, а також прогнозувати майбутнє споживання для оптимізації витрат на електроенергію.

Таким чином, проектування системи обліку споживання електроенергії в будинку є комплексним процесом, який включає аналіз потреб споживачів, вибір оптимальних технологій, розробку зручного інтерфейсу та визначення ефективних методів збору та аналізу даних.

2.2 Визначення компонентів системи

Проектування системи обліку споживання електроенергії в приватному будинку вимагає ретельного підбору всіх компонентів системи, які забезпечуватимуть її ефективне, надійне та економічно вигідне функціонування. Вибір компонентів базується на раніше проведеному аналізі, що дозволяє використовувати рішення з високою функціональністю, точністю та надійністю. Це є основою для створення дієвої системи обліку електроенергії, яка відповідає всім вимогам користувача та забезпечує оптимальне співвідношення ціни та якості.

Окрім датчика, який працює за принципом вимірювання напруги та генерації аналогового сигналу, система включає наступні ключові компоненти:

Мікроконтролер є основним елементом системи, відповідальним за зчитування даних з датчиків, їх обробку та передачу на сервер. Він також забезпечує можливість віддаленого моніторингу та керування системою, що є важливим для користувачів, які бажають мати повний контроль над споживанням електроенергії в режимі реального часу. Використання вбудованих модулів Wi-Fi та Bluetooth дозволяє з'єднуватися з мережею Інтернет та іншими пристроями без необхідності додаткового обладнання. Це значно спрощує установку та налаштування системи.

Мікроконтролери типу "система на кристалі" мають низьке енергоспоживання, високу швидкодію та компактні розміри, що робить їх ідеальними для використання у домашніх умовах. Вони також підтримують різноманітні протоколи зв'язку, що забезпечує їх сумісність з іншими компонентами системи

Датчик вимірювання струму використовується для безконтактного вимірювання струму в електричному колі. Вимірювальна головка працює за принципом струмового трансформатора, що дозволяє точно визначати струм без необхідності фізичного контакту з провідником. Сучасні датчики струму можуть вимірювати широкий діапазон струмів з високою точністю. Вони мають високу чутливість та стабільність, що забезпечує надійність вимірювань протягом тривалого часу. Безконтактне вимірювання мінімізує ризик пошкодження електричних мереж та забезпечує безпеку під час монтажу та експлуатації. Датчики струму легко інтегруються з мікроконтролерами та іншими елементами системи, що спрощує їх установку.

Світлодіоди використовуються для індикації справності системи. Вони забезпечують візуальний зворотний зв'язок про стан системи, що дозволяє користувачам легко визначати, чи працює система належним чином. Світлодіоди є енергоефективними джерелами світла, які мають тривалий термін служби. Вони можуть бути налаштовані на різні кольори та режими роботи, що дозволяє створювати інтуїтивно зрозумілі сигнали для користувачів

Використання світлодіодів для індикації забезпечує простий та зрозумілий спосіб контролю за станом системи. Вони споживають мінімум електроенергії, що робить їх ідеальними для постійного використання.

На основі наведеного опису вимог до системи та обраного обладнання, містить мінімальний набір компонентів для вдалого монтажу системи (табл. 2.1).

Окрім основних компонентів, система може включати ряд додаткових елементів, що забезпечують її повну функціональність.

Важливо забезпечити стабільне та надійне джерело живлення для всіх компонентів системи. Це може бути спеціальний блок живлення або батарея, яка забезпечує автономну роботу системи у випадку перебоїв з електропостачанням.

Таблиця 2.1 – Набір компонентів для мінімального монтажу

№	Назва	Кількість
1	Мікроконтролер	1
2	Датчик вимірювання напруги	1
3	Датчик вимірювання струму	1
4	Світлодіод	3

Для забезпечення бездротового зв'язку та передачі даних на сервер або в хмарне сховище, система може використовувати маршрутизатор Wi-Fi. Це дозволяє організувати віддалений доступ до даних та керування системою з будь-якої точки світу.

Дані про споживання електроенергії можуть зберігатися на локальному сервері або в хмарному сховищі. Це забезпечує доступність даних для аналізу та забезпечує їх збереження у випадку непередбачених ситуацій.

Розробка спеціалізованого програмного забезпечення для збору, обробки та аналізу даних є ключовим елементом системи. Це може бути як локальна програма, так і веб-інтерфейс або мобільний додаток, що дозволяє користувачам легко взаємодіяти з системою.

2.3 Проектування структури системи

Після успішного аналізу компонентів системи потрібно спроектувати її структуру. Загалом система складатиметься з умовних підсистем, кожна з яких виконує специфічні функції та забезпечує комплексне управління і контроль споживання електроенергії в будинку. Нижче наведено детальний опис кожної з підсистем:

Загалом система складатиметься з умовних підсистем:

1. Підсистема вимірювання напруги в мережі. Ця підсистема відповідає за постійний моніторинг напруги в електричній мережі будинку.
2. Підсистема вимірювання струму в мережі. Ця підсистема відповідає за вимірювання струму, що протікає через електричні прилади в будинку.
3. Підсистема оповіщення та індикації про стан мережі. Ця підсистема забезпечує користувача інформацією про поточний стан електричної мережі.
4. Підсистема керування вузлами системи. Ця підсистема відповідає за управління різними вузлами системи, забезпечуючи їх координацію та інтеграцію.
5. Підсистема віддаленого моніторингу та контролю. Ця підсистема дозволяє користувачам віддалено контролювати та керувати системою через інтернет або локальну мережу.

Для забезпечення безперебійної роботи системи всі підсистеми повинні бути інтегровані між собою. Це досягається через використання центрального контролера, який об'єднує дані з усіх підсистем та забезпечує їх синхронізацію. Комунікаційні модулі забезпечують надійний зв'язок між підсистемами, дозволяючи їм обмінюватись інформацією в режимі реального часу.

Проектування структури системи обліку споживання електроенергії в приватному будинку включає розробку та інтеграцію кількох підсистем, кожна з яких виконує свої специфічні функції. Підсистема вимірювання напруги в мережі, підсистема вимірювання струму, підсистема оповіщення та індикації, підсистема керування вузлами та підсистема віддаленого моніторингу та контролю працюють разом для забезпечення ефективного контролю та управління споживанням електроенергії, що дозволяє знизити витрати та підвищити енергоефективність будинку. Структурна схема системи наведена на рисунку 2.1.

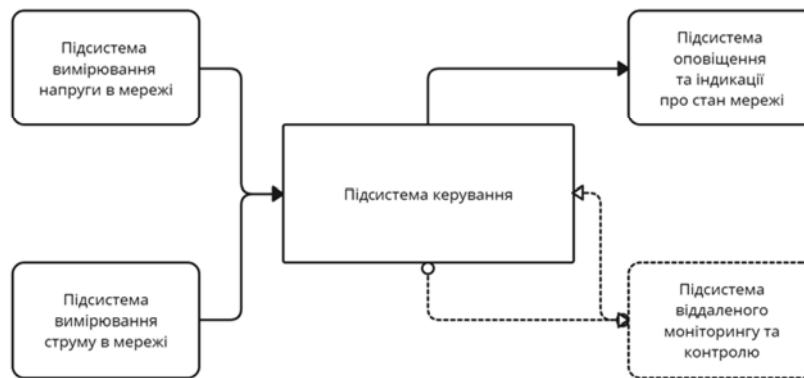


Рисунок 2.1 – Структурна схема системи

Завдяки структурній схемі можна наглядно зрозуміти на чому забезпечена система та увесь принцип її роботи завдяки налагодженим підсистемам, саме вони надсилають, збирають, сповіщають та керують. Надалі через те що підсистеми підконтрольні одному мікроконтролеру, який спарений із хмарним сервісом, нам надається можливість в реальному часі моніторити систему загалом.

2.3 Алгоритм роботи системи

Правильний алгоритм є запорукою стабільної роботи системи, тому варто детально описати кроки та дії, які відбуватимуться одразу з моменту запуску системи.

Ініціалізація починається з того, що мікроконтролер проводить початкову ініціалізацію всіх компонентів, готуючи їх до роботи

Після цього напруговий датчик здійснює зчитування рівня напруги в електричній мережі, а мікроконтролер обробляє ці дані, використовуючи алгоритми обробки, і зберігає їх у внутрішню пам'ять. Паралельно з цим процесом, датчик струму здійснює зчитування струму в електричній мережі, передаючи отримані дані на обробку мікроконтролеру.

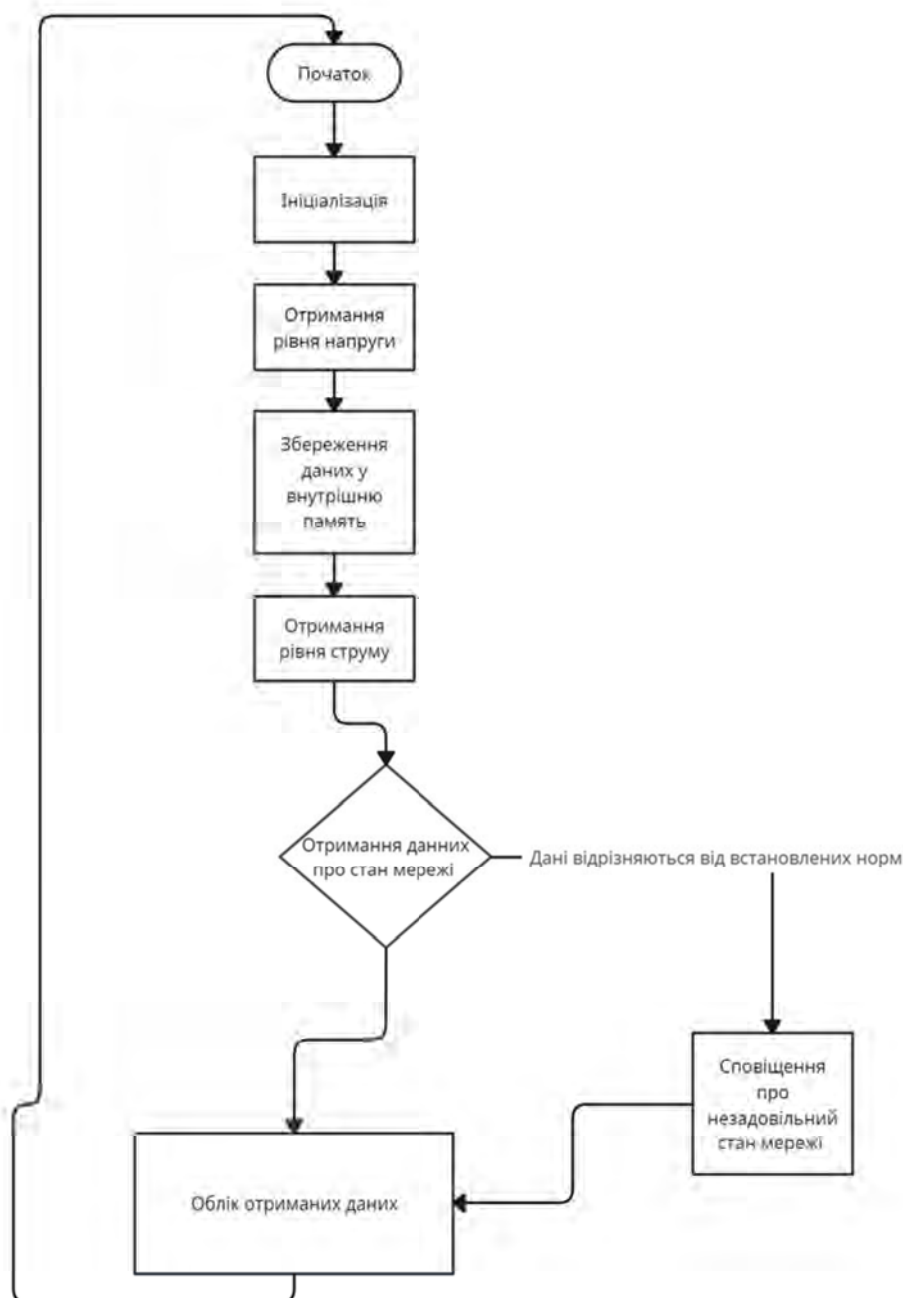


Рисунок 2.2 – Схема роботи системи

Схема алгоритму роботи системи наведена на рисунку 2.2, що візуалізує послідовність виконання всіх дій та процесів, забезпечуючи наочне розуміння принципу функціонування системи.

Далі, оброблені дані відправляються за допомогою модуля Wi-Fi на зовнішню платформу для подальшого зберігання або аналізу, забезпечуючи віддалений доступ до інформації про споживання електроенергії. У випадку, якщо отримані дані відрізняються від встановлених норм, система здійснює

відповідне сповіщення у систему моніторингу, попереджаючи користувача про можливі проблеми або аномалії в мережі.

Незалежно від позитивних чи негативних сповіщень, система веде постійний облік отриманих даних, що дозволяє користувачу у будь-який час дізнатись про стан мережі та поточне споживання електроенергії.

Таким чином, алгоритм роботи системи забезпечує її безперебійну і ефективну роботу, надаючи користувачам точну і своєчасну інформацію для управління енергоспоживанням.

					КР.КІ 24.532.05.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		27

3 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1 Вибір компонентів

Реалізація та тестування системи обліку споживання електроенергії є ключовими етапами, що визначають її надійність, точність та ефективність. На цьому етапі всі компоненти, вибрані на етапі проектування, інтегруються в єдину систему, яка потім проходить ретельне тестування для перевірки її функціональності та відповідності заданим вимогам.

Перший крок у реалізації системи полягає в монтажі всіх обраних компонентів, таких як датчики напруги і струму, мікроконтролери, комунікаційні модулі та інші необхідні елементи. Після фізичної установки, всі компоненти підключаються відповідно до розробленої структурної схеми. Особлива увага приділяється правильному підключенню датчиків та забезпеченню надійного зв'язку між ними та мікроконтролером.

Наступним етапом є налаштування програмного забезпечення. На мікроконтролер завантажується програмне забезпечення, яке буде відповідати за зчитування даних з датчиків, їх обробку, збереження та передачу на зовнішню платформу. Для цього можуть бути використані різні алгоритми обробки даних, які забезпечують високу точність та швидкість роботи системи.

Після налаштування програмного забезпечення, проводиться його інтеграція з хмарним сервісом, який забезпечує віддалений доступ до даних. Хмарний сервіс дозволяє зберігати великі обсяги даних, проводити їх аналіз, генерувати звіти та надсилати сповіщення користувачам у разі виявлення аномалій.

Після завершення етапу реалізації, система проходить кілька етапів тестування для перевірки її функціональності та відповідності технічним вимогам.

Першим кроком є тестування окремих компонентів системи. Датчики напруги та струму перевіряються на точність вимірювань, мікроконтролери на коректність обробки даних, а комунікаційні модулі – на стабільність передачі даних.

Наступний етап – це інтеграційне тестування, яке перевіряє взаємодію всіх компонентів системи між собою. Це тестування дозволяє виявити можливі проблеми в комунікації між компонентами та забезпечує їх коректну роботу у складі єдиної системи.

Після інтеграційного тестування, система проходить функціональне тестування, яке перевіряє її роботу у реальних умовах експлуатації. Це включає перевірку системи на стійкість до різних умов навантаження, тестування функціоналу віддаленого моніторингу та управління, а також перевірку алгоритмів сповіщення у випадку аномалій.

Фінальним етапом є тестування на відповідність вимогам користувача. Це тестування включає перевірку зручності користування інтерфейсом системи, точності наданих даних та швидкості реакції системи на зміни у мережі. За результатами цього тестування можуть бути внесені корективи до програмного забезпечення або налаштувань системи для досягнення максимальної відповідності потребам користувача.

Вибір компонентів обґрунтований їхньою високою функціональністю, точністю та надійністю, що дозволить створити ефективну, надійну та головне бюджетну систему обліку споживаної електроенергії в будинку.

ESP32 має двоядерний процесор та значну кількість вбудованих ресурсів, що робить його ефективним для вимірювання та обробки даних в реальному часі (рис. 3.1). Двоядерна архітектура дозволяє паралельно виконувати кілька задач, що підвищує швидкість і продуктивність системи.

Наявність вбудованих модулів Wi-Fi та Bluetooth забезпечує можливість віддаленого моніторингу та зручність з'єднання з іншими пристроями. Це дозволяє користувачам отримувати дані про споживання електроенергії з будь-якого місця, де є доступ до інтернету, а також легко інтегрувати ESP32 з іншими системами розумного будинку.



Рисунок 3.1 – Мікроконтролер ESP32 з WiFi-модулем

ESP32 легко програмується мовами Arduino та MicroPython, що дозволяє швидку розробку та розширення функціоналу. Це робить ESP32 доступним для розробників різного рівня та дозволяє створювати складні системи з мінімальними витратами часу і ресурсів.

Технічні характеристики ESP32:

- Частота процесора 160 МГц.
- Інтегрований Wi-Fi та Bluetooth модулі забезпечують бездротову комунікацію.
- Вбудована пам'ять 520 КБ RAM та 4 МБ Flash, що забезпечує достатньо простору для зберігання даних та виконання програм.

ESP32 використовується для зчитування даних зі сенсорів, таких як ZMPT101B, їх обробки та передачі на сервер чи інше зберігання. Мікроконтролер приймає аналогові сигнали від сенсорів, конвертує їх у цифрову форму, обробляє ці дані і відправляє результати на зовнішні платформи для подальшого аналізу і зберігання.

ZMPT101B є датчиком напруги, який забезпечує високоточні вимірювання напруги в електричній мережі (рис. 3.2) Цей модуль дозволяє отримувати стабільні та точні дані, що є критично важливим для моніторингу та аналізу споживання електроенергії.



Рисунок 3.2 – Датчик ZMPT101B

ZMPT101B використовує трансформатор напруги для вимірювання змінної напруги в мережі. Отримані дані передаються на мікроконтролер (ESP32) для подальшої обробки. Завдяки високій чутливості, ZMPT101B може виявляти навіть незначні зміни в напрузі, що дозволяє більш точно контролювати споживання електроенергії.

Технічні характеристики ZMPT101B:

- Діапазон вимірюваної напруги: Від 0 до 250 В.
- Висока точність вимірювань забезпечує стабільні та надійні дані.
- Аналоговий вихідний сигнал, який легко інтегрується з мікроконтролерами.

SCT-013-000 є датчиком струму, який забезпечує високоточні вимірювання струму до 100А в електричній мережі. Ця висока точність дозволяє отримувати надійні дані про споживання електроенергії, що є важливим для ефективного моніторингу та управління.

SCT-013-000 використовує безконтактний метод вимірювання струму, що забезпечує повну електричну ізоляцію від вимірювального кола. Це забезпечує безпеку під час встановлення та експлуатації датчика, а також захищає мікроконтролер від потенційних електричних перешкод та небезпек.



Рисунок 3.3 – Датчик SCT-013-000

Технічні характеристики SCT-013-000:

- Діапазон вимірюваного струму: 0-100 А.
- Вихідний сигнал: 0-1 В (залежно від моделі).
- Повна електрична ізоляція від вимірюваного кола.
- Висока точність вимірювань забезпечує стабільні та надійні дані.

Датчик SCT-013-000 працює за принципом струмового трансформатора, який безконтактно вимірює струм, що протікає через провідник. Датчик генерує аналоговий сигнал, пропорційний вимірюваному струму, який може бути зчитаний мікроконтролером ESP32. Отримані дані обробляються мікроконтролером і можуть бути передані на сервер або іншу платформу для подальшого аналізу та зберігання.

Обґрунтований вибір компонентів, таких як ESP32, ZMPT101B та SCT-013-000, базується на їхній високій функціональності, точності та надійності. Це дозволяє створити ефективну, надійну та бюджетну систему обліку споживаної електроенергії в будинку. ESP32 забезпечує високу продуктивність та можливість віддаленого моніторингу завдяки інтегрованим модулям Wi-Fi та Bluetooth, а також легкість програмування, що сприяє швидкій розробці системи.

3.2 Реалізація системи

Процес реалізації принципової електричної схеми та монтажу пристрою складається з кількох ключових етапів, кожен з яких має бути виконаний з високою точністю, щоб забезпечити надійність та ефективність роботи системи обліку споживання електроенергії. Спочатку слід розробити принципову електричну схему, використовуючи спеціалізоване програмне забезпечення (наприклад, AutoCAD Electrical або KiCad), і створити детальну електричну схему, що включає всі компоненти: SCT-013-000, ZMPT101B, ESP32 DEV KIT, резистори та конденсатор. На схемі необхідно чітко позначити всі з'єднання між компонентами, включаючи підключення живлення (VCC, GND) та сигнальні лінії (GPIO) (рис. 3.4).

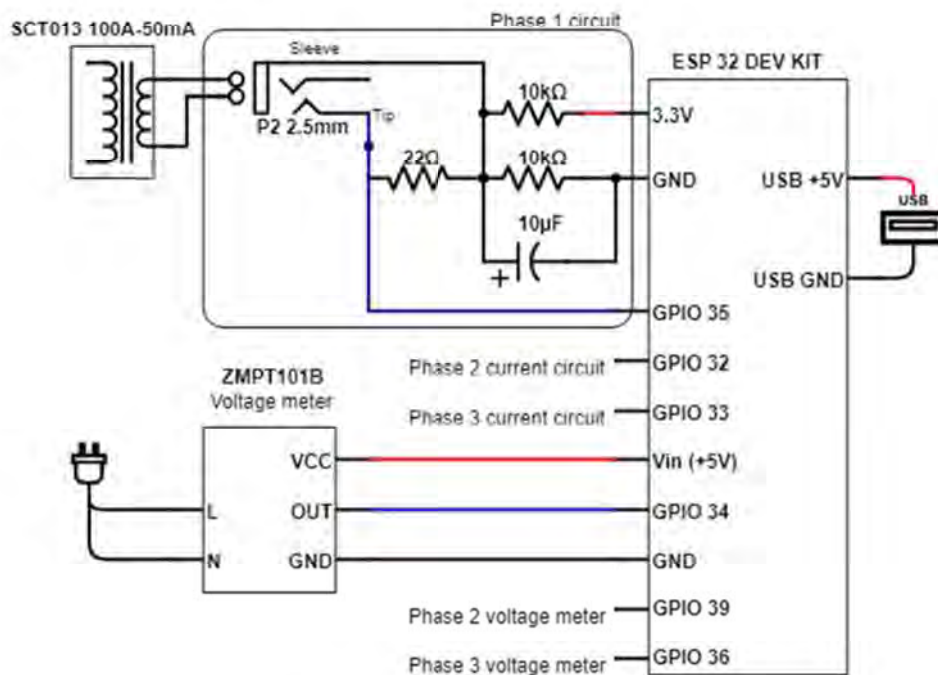


Рисунок 3.4 – Електрична схема пристрою

Для монтажу потрібно підготувати робоче місце, забезпечивши наявність всіх необхідних інструментів, таких як паяльник, провідники, тестер, пінцет тощо. Далі розміщуються всі компоненти на макетній платі відповідно до принципової електричної схеми.

Для підключення датчика до ESP32 спочатку потрібно підготувати датчик SCT-013-000. Він повинен бути розміщений на фазовому провіднику, через

який протікає струм, що вимірюється. Далі беруться два вихідних провідники від датчика SCT-013-000. Один з цих провідників підключають до заземлення (GND) на ESP32. Інший вихідний провідник підключають до одного з аналогових входів ESP32 через резистор, що забезпечить правильне вимірювання струму (рис. 3.5).

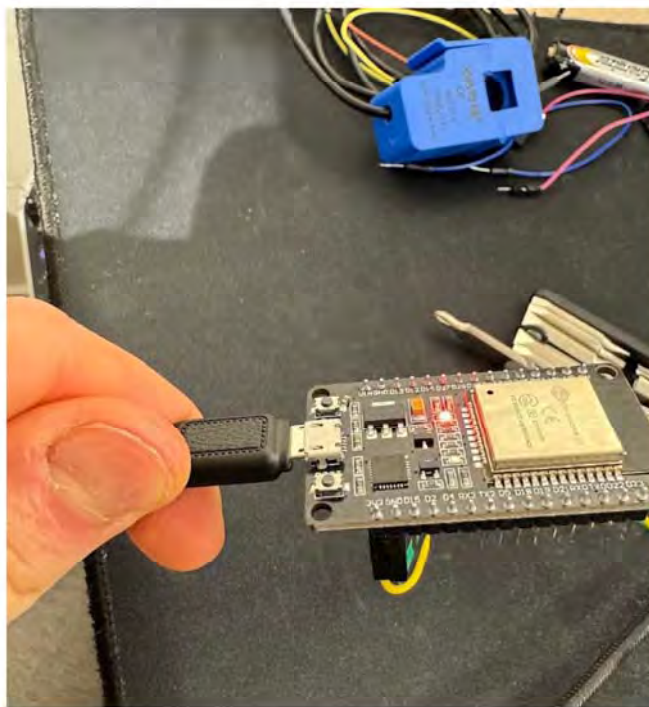


Рисунок 3.5 – Підключення ESP32 разом з датчиком SCT-013-000

Для підключення датчика ZMPT101B до ESP32 провідник від контакту живлення датчика потрібно підключити до контакту 3.3V на ESP32. Потім провідник від контакту заземлення датчика до контакту GND на ESP32. Далі вихідний провідник від датчика ZMPT101B до одного з аналогових входів ESP32.

Це дозволить датчику передавати виміряні дані про напругу до мікроконтролера для подальшої обробки та аналізу. Після усіх виконаних кроків, та дотримання норм та вимог поведження із струмом та напругою дозволяється підключення силового кабелю 220В під навантаженням (рис. 3.6).

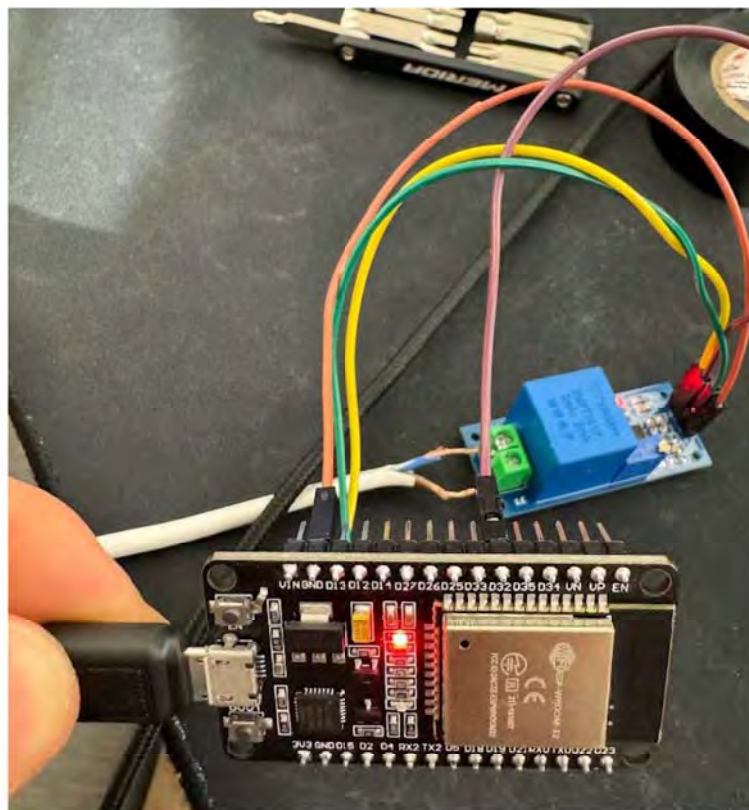


Рисунок 3.6 – Монтаж ESP32 разом з датчиком ZMPT101B

Після завершення монтажу слід перевірити всі з'єднання на правильність та відсутність коротких замикань. ESP32 підключається до комп'ютера через USB і завантажується програма для зчитування та обробки даних з датчиків, після чого перевіряється робота системи, здійснюючи вимірювання струму та напруги в реальному часі.

3.3 Реалізація програмного коду

Програмна реалізація системи виконана на мові програмування C++ за використанням мікроконтролера ESP32 та платформи Blynk для забезпечення зручного моніторингу та управління системою. Основні частини програмної реалізації включають використання бібліотек, налаштування Wi-Fi та Blynk, ініціалізацію мікроконтролера та параметрів сенсора, а також основну логіку роботи системи.

Для реалізації системи використовуються кілька важливих бібліотек:

- EmonLib використовується для реалізації алгоритмів вимірювання напруги та струму.
- WiFi.h та WiFiClient.h забезпечують роботу з Wi-Fi мережею.
- BlynkSimpleEsp32.h забезпечує інтеграцію з платформою Blynk для віддаленого моніторингу та управління.
- Esphome.h та esp_task_wdt.h: забезпечують роботу з ESPHome та контроль за виконанням завдань.

Вказуються параметри Wi-Fi мережі (назва та пароль) та токен автентифікації для Blynk. Ініціалізація підключення до Blynk здійснюється через команду Blynk.begin(auth, ssid, pass), що забезпечує з'єднання з сервером Blynk (рис. 3.7).

```
#include "esphome.h"
#include "EmonLib.h"
#include "esp_task_wdt.h" // Функції управління Watchdog
#include "BlynkSimpleEsp32.h" // Бібліотека для інтеграції з Blynk
#include <WiFi.h> // Бібліотека для роботи з Wi-Fi
#include <WiFiClient.h> // Бібліотека для клієнта Wi-Fi
#define ESP32

// Тайм-аут Watchdog у секундах
#define WDT_TIMEOUT 5

// Довжина вікна зразків визначається кількістю напівхвиль або переходів, які ми вибираємо
// 120 є досить консервативним для двох фаз
#define CROSSINGS 120

// Токен автентифікації Blynk
char auth[] = "Ваш токен Blynk";

// Назва та пароль Wi-Fi мережі
char ssid[] = "Ваш Wi-Fi SSID";
char pass[] = "Ваш Wi-Fi пароль";
```

Рисунок 3.7 – Підключення бібліотек та налаштування Wi-Fi та Blynk

Зокрема, підключаються бібліотека для управління Watchdog, яка запобігає зависанню мікроконтролера, а також бібліотеки для роботи з Wi-Fi (WiFi.h та WiFiClient.h), що дозволяють підключатися до бездротової мережі та забезпечувати зв'язок з інтернетом.

Конфігурація Watchdog є критично важливою для забезпечення надійності системи, оскільки Watchdog автоматично перезавантажує систему у випадку, якщо вона перестає відповідати. Тайм-аут Watchdog встановлений на 5 секунд, що дозволяє уникнути зависання програми та забезпечує її стабільну роботу. Якщо програма зависає, Watchdog автоматично перезапускає ESP32.

Визначення кількості напівхвиль для вимірювання є необхідним для точного аналізу споживання електроенергії. У нашому випадку ми вибираємо 120 напівхвиль, що забезпечує точний аналіз змінних електричних параметрів.

Налаштування Blynk включає в себе аутентифікаційний токен, а також параметри Wi-Fi, такі як назва мережі та пароль. Це дозволяє мікроконтролеру підключатися до Wi-Fi та Blynk, що забезпечує віддалений моніторинг даних через мобільний додаток .

Інтервал опитування визначає, як часто система буде зчитувати дані про споживання електроенергії. У нашому випадку, інтервал між вимірюваннями становить 30 секунд, що дозволяє уникнути надмірного навантаження на мікроконтролер і забезпечує оновлення даних. Встановлення інтервалу менше 5 секунд може призвести до збоїв у роботі системи (рис. 3.8).

```
// Інтервал опитування
#define POLLING_INTERVAL 30000 //Кожні 30 секунд буде виконуватися зчитування та публікація даних

// Конфігурація пінів
#define V1 34
#define V2 34 //Встановіть на V1, якщо використовується лише один вимірювач напруги
#define V3 34 //Встановіть на V1, якщо використовується лише один вимірювач напруги

#define I1 35
#define I2 32
#define I3 33
```

Рисунок 3.8 – Інтервал опитування

Конфігурація пінів включає в себе визначення пін-кодів для підключення сенсорів напруги та струму до ESP32.

Якщо використовується один вимірювач напруги, всі пін-коди для напруги встановлюються однаково ($V1 = V2 = V3 = 34$). Це дозволяє правильно підключити датчики до мікроконтролера та забезпечити коректне зчитування даних.

Наступним блоком у реалізованому коді є калібрування. Калібрування сенсорів забезпечує правильні вимірювання датчиків. У нашому випадку (рис. 3.9), встановлюються калібрувальні значення для сенсорів напруги (CV) та струму (CI). Калібрування дозволяє забезпечити точність вимірювань і може бути змінено в залежності від характеристик використовуваних сенсорів.

```
// Калібрування сенсорів напруги CV та струму CI
#define CV1 510
#define CV2 510 //Встановіть на CV1, якщо використовується лише один вимірювач напруги
#define CV3 510 //Встановіть на CV1, якщо використовується лише один вимірювач напруги

#define CI1 96.72
#define CI2 94.93
#define CI3 1
```

Рисунок 3.9 – Калібрування сенсорів напруги та струму

Далі створення класу MyPowerSensor, який наслідується від PollingComponent і Sensor, є ключовим для забезпечення періодичного опитування сенсорів та обробки даних. Цей клас відповідає за взаємодію з сенсорами напруги та струму, а також за обробку і передачу даних до Blynk (рис. 3.10).

Метод get_setup_priority() встановлює пріоритет налаштування, що гарантує правильну послідовність ініціалізації компонентів і забезпечує стабільну роботу системи.

```

class MyPowerSensor : public PollingComponent, public Sensor
{
public:
    MyPowerSensor() : PollingComponent(POLLING_INTERVAL) {}

    float get_setup_priority() const override { return esphome::setup_priority::LATE; }

    EnergyMonitor emon1; // Фаза 1
    EnergyMonitor emon2; // Фаза 2
    // EnergyMonitor emon3; // Фаза 3

    // Сенсори фази 1
    Sensor *realpower_sensor1 = new Sensor();
    Sensor *apparentpower_sensor1 = new Sensor();
    Sensor *powerfactor_sensor1 = new Sensor();
    Sensor *supplyvoltage_sensor1 = new Sensor();
    Sensor *current_sensor1 = new Sensor();

```

Рисунок 3.10 – Створення класу MyPowerSensor

Об'єкти EnergyMonitor для фаз 1 і 2 дозволяють здійснювати вимірювання електричних параметрів кожної фази, а сенсори для кожної фази включають сенсори реальної та уявної потужності, коефіцієнта потужності, напруги та струму. Це забезпечує детальний аналіз споживання електроенергії у приватному будинку.

Після усіх блоків коду, залишилось отримати повну картину споживання електроенергії у будинку. Потрібно використати загальні сенсори для об'єднання даних з усіх фаз і отримання загальних показників споживання електроенергії (рис. 3.11).

Ініціалізація Blynk є важливим етапом, який дозволяє мікроконтролеру підключитися до Wi-Fi та Blynk, що забезпечує віддалений моніторинг даних через мобільний додаток. Метод Blynk.begin(auth, ssid, pass) встановлює з'єднання з Wi-Fi мережею та Blynk-сервером.


```

// Загальні сенсори
Sensor *realpower_sensor_total = new Sensor();
Sensor *apparentpower_sensor_total = new Sensor();
Sensor *current_sensor_total = new Sensor();

void setup() override
{
    // Ініціалізація Blynk
    Blynk.begin(auth, ssid, pass);

    esp_task_wdt_init(WDT_TIMEOUT, true); // ввімкніть рапіс, щоб ESP32 перезавантажувався
    esp_task_wdt_add(NULL);               // додайте поточний потік до спостереження WDT

```

Рисунок 3.11 – Об'єднання сенсорів та ініціалізація Blynk

Ініціалізація Blynk є важливим етапом, який дозволяє мікроконтролеру підключитися до Wi-Fi та Blynk, що забезпечує віддалений моніторинг даних через мобільний додаток. Метод Blynk.begin(auth, ssid, pass) встановлює з'єднання з Wi-Fi мережею та Blynk-сервером.

Ініціалізація Watchdog забезпечує надійність системи, перезавантажуючи ESP32 у випадку зависання. Використовуються функції esp_task_wdt_init() та esp_task_wdt_add(), щоб додати поточний потік до спостереження Watchdog, що гарантує стабільну роботу. Код програмного засобу приведено в додатку А.

3.4 Підключення та налаштування Blynk.io

В даному проекті використовується сторонній сервіс Blynk для забезпечення функціональності віддаленого моніторингу та керування системою. Blynk - це платформа для створення Інтернету Речей (IoT), яка надає зручні інструменти для з'єднання різних пристроїв і мікроконтролерів з хмарним сервісом. Основні характеристики Blynk включають мобільний додаток для Android і iOS, який дозволяє користувачам моніторити та керувати своїми пристроями в реальному часі, графічний інтерфейс, який дозволяє користувачам легко створювати інтерфейс для своїх проектів без необхідності в

глибоких знаннях програмування, віддалене керування, яке забезпечує можливість віддаленого керування пристроями через Інтернет, інтеграцію з різними мікроконтролерами та платформами, такими як Arduino, Raspberry Pi, ESP8266, ESP32, і багато інших, а також сервер у хмарі, який надає хмарний сервер для зберігання та обміну даними між пристроями та мобільним додатком.

Інтерфейс налаштування пристрою у веб-версії Blynk дозволяє користувачам легко додавати нові пристрої, налаштовувати параметри та моніторити їх у режимі реального часу.

Під час реалізації додано частку програмного коду для підключення до хмарного сховища Blynk, завдяки чому можна продовжити налаштування моніторингу у веб-версії Blynk (рис. 3.12).

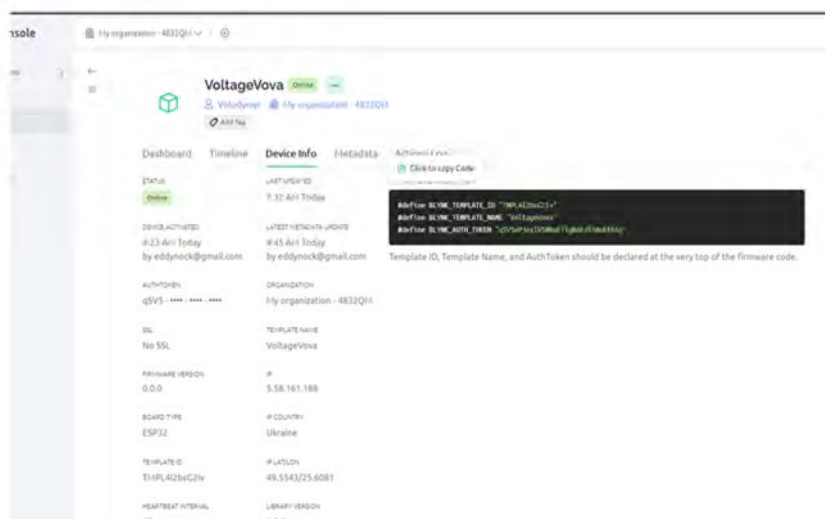


Рисунок 3.12 – Інтерфейс налаштування пристрою у веб-версії Blynk

Після успішного налаштування моніторингу у веб-версії платформи Blynk, користувачі можуть легко слідкувати за даними на своїх мобільних пристроях, використовуючи зручний мобільний додаток Blynk. Після налаштування пристрою користувачі можуть додати необхідні віджети для відображення даних, таких як графіки, індикатори, кнопки для керування та інші елементи інтерфейсу (рис. 3.13)



Рисунок 3.13 – Відстеження показників у реальному часі

Мобільний додаток дозволяє відстежувати показники у реальному часі та здійснювати віддалене керування системою.

3.5 Тестування реалізованого пристрою

План тестування включає перевірку правильності підключення всіх елементів пристрою, ініціалізацію модулів та бібліотек, підключення до Wi-Fi мережі, взаємодію з Blynk-сервером, вимірювання та обчислення даних, систему безпеки та реакцію на збої. Першим етапом є перевірка правильності підключення всіх елементів пристрою та ініціалізація модулів та бібліотек, що забезпечує коректну роботу системи. Наступним етапом є тестування можливості пристрою підключатися до бездротової мережі Wi-Fi, що включає перевірку стабільності та швидкості з'єднання.

Далі проводиться перевірка можливості пристрою встановлювати та підтримувати зв'язок із Blynk-сервером, що дозволяє здійснювати віддалений моніторинг та керування через мобільний додаток Blynk. На етапі вимірювання та обчислення даних перевіряється точність вимірювань рівня напруги. Перевірка системи безпеки включає запобігання несанкціонованому доступу та тестування стійкості до можливих атак або спроб втручання. Нарешті, тестування реакції системи на випадки втрати зв'язку або інші технічні збої дозволяє перевірити можливість відновлення роботи після відновлення зв'язку.

Тест-кейс підключення до Wi-Fi включає перевірку можливості пристрою підключитися до вказаної мережі Wi-Fi, з очікуваним результатом успішного підключення. Тест-кейс віддаленого керування перевіряє можливість вмикати/вимикати пристрій через мобільний додаток Blynk, з очікуваним результатом прискореної реакції на команди віддаленого керування. Тест-кейс вимірювання напруги включає перевірку точності вимірювань напруги, з очікуваним результатом точних вимірювань згідно з технічними характеристиками.

Тест-кейс системи безпеки передбачає спробу доступу до системи з невірними або відсутніми автентифікаційними даними, з очікуваним результатом відмови в доступі та логування спроб втручання. Тест-кейс реакції на збої зв'язку включає симуляцію втрати зв'язку та перевірку реакції системи, з очікуваним результатом адекватної поведінки системи та можливості відновлення роботи після відновлення зв'язку.

Тестування системи проводилось в контрольованих умовах, забезпечуючи безпеку під час проведення тестів та враховуючи можливі ризик.

					КР.КІ 24.532.05.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		43

4 ТЕХНІКО-ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ

4.1 Аналіз ринку

Програмне забезпечення для моніторингу та управління споживанням електроенергії є важливою складовою сучасних систем енергоефективності. Такі системи забезпечують збір, зберігання та аналіз даних про споживання електроенергії, а також надають можливість віддаленого контролю та управління електроприладами. Сучасні технології в цій сфері надають широкий спектр можливостей для оптимізації споживання електроенергії, але комерційні рішення часто мають високу вартість. Тому розробка власної системи моніторингу є економічно доцільною та виправданою.

Ефективне управління споживанням електроенергії є критичним для сучасного бізнесу та домогосподарств, оскільки дозволяє знизити витрати на електроенергію та підвищити загальну енергоефективність. Системи моніторингу та управління споживанням електроенергії потребують постійного технічного обслуговування та регулярних оновлень, щоб забезпечити їх надійність та точність. Без належного догляду та оновлень такі системи можуть втратити свою ефективність і точність, що призведе до зниження енергоефективності та зростання витрат.

4.2 Розрахунок витрат на розробку проекту

Розробка та впровадження системи моніторингу споживання електроенергії є складним проектом, що включає в себе аналіз економічних потреб. Проведення дослідження ринку, аналіз попиту, обсягу та динаміки розвитку, а також наявність потреби в таких рішеннях підтвердили доцільність цього проекту. Динамічне зростання ринку та вільна конкуренція є основними критеріями для підготовки розрахунків витрат на розробку системи.

Підрахунок витрат, які необхідно буде понести для реалізації цього проекту, наведено в таблиці 4.1. Витрати включають закупівлю обладнання,

розробку програмного забезпечення, інтеграцію та тестування системи, а також обслуговування та підтримку проекту.

Таблиця 4.1 – Список витрат на створення системи

Призначення позицій витрат	Сума, грн	Обґрунтування
Заробітна плата програміста	13500	
Відрахування на соціальні потреби	1485	ЄСП – 11% (13500 * 0.11)
Контрагентські роботи і послуги	5265	39% (13500 * 0.39)
Витрати на відрядження	0	
Інші прямі витрати	1620	13500 – 12% (13500 * 0.12)
Усього прямих витрат	6750	5265 + 1485
Накладні витрати	1417.50	6750 – 21% (6750 * 0.21)
Планові накопичення	1408.50	(5265 + 1417.50) – 21% (6682.50 * 0.21)
Усього, кошторисна вартість проекту	16408.50	13500 + 1485 + 1408.50
Податок на додану вартість	3281.70	16408.50 – 20% (16408.50 * 0.20)
Загалом, договірна ціна розробки ПЗ	19690.20	16408.50 + 3281.70

Заробітна плата робітникам калькулюється на одного учасника, розробника, як це показано в таблиці 4.2

Таблиця 4.2 – Розрахунок оплати праці

№	Посада	Оклад	Відрахування	Кількість		Сума
п/п	виконавця	грн/міс	грн/міс	чол.	місяців	з/п, грн
1	Програміст - Junior	13500	2632,5	1	1	10867,5
2	Електромонтажник	4000	780	1	1	3220
3	Тестувальник	5000	975	1	1	4025
Усього зарплати						182112,5

4.3 Обґрунтування необхідності розробки

Розвиток технологій вимагає впровадження інноваційних рішень для контролю та управління енергоспоживанням. Система обліку електроенергії в приватному будинку дозволяє не лише моніторити споживання, але й оптимізувати його, що веде до значних економій та підвищення енергоефективності. Такі системи забезпечують зручний доступ до даних про споживання в реальному часі, що дозволяє своєчасно реагувати на зміни та вживати заходів для зниження витрат.

Сучасні системи обліку електроенергії повинні бути надійними, точними та легкими у використанні, що підвищує їх привабливість для користувачів. Власна розробка системи обліку електроенергії дозволяє врахувати специфічні потреби користувача, забезпечити високу точність вимірювань та інтеграцію з іншими системами розумного будинку.

Таким чином, створення власної системи обліку електроенергії для приватного будинку є доцільним і економічно виправданим рішенням, яке дозволяє підвищити енергоефективність та знизити витрати на електроенергію.

ВИСНОВКИ

Розроблена система обліку електроенергії для приватного будинку успішно вирішує поставлені завдання, забезпечуючи користувачам можливість ефективного моніторингу та управління споживанням електроенергії. Використання сучасного апаратного забезпечення, такого як мікроконтролер ESP32, датчики напруги ZMPT101B та струму SCT-013-000, а також платформа Blynk, дозволило створити надійну, функціональну та доступну систему.

Ключові переваги системи включають можливість віддаленого моніторингу, що дає змогу користувачам у будь-який момент переглядати дані про споживання електроенергії через зручний мобільний додаток. Точність вимірювань, забезпечена використанням високоякісних датчиків, гарантує надійні результати та дозволяє користувачам робити обґрунтовані рішення щодо управління енергоспоживанням.

Економічна доступність системи робить її привабливою для широкого кола користувачів, які бажають зменшити витрати на електроенергію та підвищити енергоефективність свого будинку.

Шляхи вдосконалення системи включають розширення функціоналу, наприклад, додавання можливості виявлення інших параметрів електромережі та реагування на перевантаження, а також впровадження заходів безпеки для захисту доступу та конфіденційності даних. Інтеграція з іншими платформами та додавання системи повідомлень для сповіщення користувачів про події або аномалії в електромережі також можуть значно покращити функціональність системи.

Таким чином, розроблена система обліку електроенергії для приватного будинку є важливим кроком до підвищення енергоефективності та раціонального використання ресурсів. Вона не лише задовольняє поточні потреби користувачів, але й має значний потенціал для подальшого розвитку та вдосконалення, що зробить її ще більш корисною та надійною у майбутньому.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Посібники з використання та програмування мікроконтролера ESP32: веб-сайт. URL: <https://docs.espressif.com/projects/esp-idf/en/stable/esp32/index.html> (дата звернення: 10.03.2024)
2. Технічні специфікації та рекомендації щодо використання датчика напруги ZMPT101B: веб-сайт. URL: <https://innovatorsguru.com/wp-content/uploads/2019/02/ZMPT101B.pdf> (дата звернення 13.04.2024)
3. Технічні специфікації та рекомендації щодо використання датчика струму SCT-013-000: веб-сайт. URL: <https://bc-robotics.com/datasheets/yhdc.pdf> (дата звернення: 14.04.2024)
4. Інформація про можливості, налаштування та використання платформи Blynk для віддаленого моніторингу: веб-сайт. URL: <https://docs.blynk.io/en> (дата звернення: 25.04.2024)
5. Усе про лічильники електроенергії та методи автоматизованого обліку електроенергії в Україні: веб-сайт.
URL: <https://www.dtekdnem.com.ua/ua/metering-devices> (дата звернення: 01.05.2024)
6. Загальна інформація про мікроконтролер ESP32 в середовищі програмування Arduino IDE : веб-сайт.
URL: <https://itmaster.biz.ua/electronics/esp32/esp32-arduino.html> (дата звернення: 05.05.2024)
7. Методичні рекомендації до виконання кваліфікаційної роботи для студентів спеціальності 123 «Комп'ютерна інженерія» освітньої програми «Інженерія Інтернету речей» / Павлюс В.П., Посвятовська О.Б., Кульчинська Н.З. – Галицький фаховий коледж імені В'ячеслава Чорновола, Тернопіль, 2023. 52с.

ДОДАТКИ

Додаток А

Лістинг програмного засобу

```
#include "esphome.h"
#include "EmonLib.h"
#include "esp_task_wdt.h" //Watchdog control features
#define ESP32

// Watchdog timeout in seconds
#define WDT_TIMEOUT 5

// The Sample window length is defined by the number of half
wavelengths or crossings we choose to measure.
// 120 is quite conservative for two phases
#define CROSSINGS 120

// Polling interval
#define POLLING_INTERVAL 30000 //Every 30s will perform a read
and publish the data. Do not go under 5s to prevent crashes.

// Pin configuration
#define V1 34
#define V2 34 //Set to V1 if using only one voltage meter
#define V3 34 //Set to V1 if using only one voltage meter

#define I1 35
#define I2 32
#define I3 33

// Calibration for voltage sensors CV and current sensors CI
#define CV1 510
#define CV2 510 //Set to CV1 if using only one voltage meter
#define CV3 510 //Set to CV1 if using only one voltage meter

#define CI1 96.72
```

					КР.КІ 24.532.05.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		49


```

#define CI2 94.93
#define CI3 1

class MyPowerSensor : public PollingComponent, public Sensor
{
public:
    MyPowerSensor() : PollingComponent(POLLING_INTERVAL) {}

    float get_setup_priority() const override { return
esphome::setup_priority::LATE; }

    EnergyMonitor emon1; // Phase 1
    EnergyMonitor emon2; // Phase 2
    // EnergyMonitor emon3; //Phase 3

    // Phase 1 sensors
    Sensor *realpower_sensor1 = new Sensor();
    Sensor *apparentpower_sensor1 = new Sensor();
    Sensor *powerfactor_sensor1 = new Sensor();
    Sensor *supplyvoltage_sensor1 = new Sensor();
    Sensor *current_sensor1 = new Sensor();

    // Phase 2 sensors
    Sensor *realpower_sensor2 = new Sensor();
    Sensor *apparentpower_sensor2 = new Sensor();
    Sensor *powerfactor_sensor2 = new Sensor();
    Sensor *supplyvoltage_sensor2 = new Sensor();
    Sensor *current_sensor2 = new Sensor();

    // Phase 3 sensors
    /*
    Sensor *realpower_sensor3 = new Sensor();
    Sensor *apparentpower_sensor3 = new Sensor();
    Sensor *powerfactor_sensor3 = new Sensor();
    Sensor *supplyvoltage_sensor3 = new Sensor();
    Sensor *current_sensor3 = new Sensor();

```

```

*/

// Total sensors
Sensor *realpower_sensor_total = new Sensor();
Sensor *apparentpower_sensor_total = new Sensor();
Sensor *current_sensor_total = new Sensor();

void setup() override
{

    esp_task_wdt_init(WDT_TIMEOUT, true); // enable panic so
ESP32 restarts

    esp_task_wdt_add(NULL); // add current
thread to WDT watch

/*
    Analog attenuation: When using 0-1V sensors the
attenuation should be decreased to improve accuracy.

    ADC_0db: sets no attenuation. ADC can measure up to
approximately 800 mV (1V input = ADC reading of 1088).

    ADC_2_5db: The input voltage of ADC will be attenuated,
extending the range of measurement to up to approx. 1100 mV. (1V
input = ADC reading of 3722).

    ADC_6db: The input voltage of ADC will be attenuated,
extending the range of measurement to up to approx. 1350 mV. (1V
input = ADC reading of 3033).

    ADC_11db (default): The input voltage of ADC will be
attenuated, extending the range of measurement to up to approx.
2600 mV. (1V input = ADC reading of 1575).

*/

    analogSetPinAttenuation(V1, ADC_11db);
    analogSetPinAttenuation(V2, ADC_11db);
    analogSetPinAttenuation(V3, ADC_11db);

```



```

    analogSetPinAttenuation(I1, ADC_11db);
    analogSetPinAttenuation(I2, ADC_11db);
    analogSetPinAttenuation(I3, ADC_11db);

    // Phase 1 sensors
    emon1.voltage(V1, CV1, 1.732); // Voltage: input pin,
calibration, phase_shift
    emon1.current(I1, CI1);          // Current: input pin,
calibration.

    // Phase 2 sensors
    emon2.voltage(V2, CV2, 1.732); // Voltage: input pin,
calibration, phase_shift
    emon2.current(I2, CI2);          // Current: input pin,
calibration.

    /*
    //Phase 3 sensors
    emon3.voltage(V3, CV3, 1.732); // Voltage: input pin,
calibration, phase_shift
    emon3.current(I3, CI3);          // Current: input pin,
calibration.
    */
}

void update() override
{

    // Phase 1
    emon1.calcVI(CROSSINGS, 2000);
    float realPower1 = emon1.realPower;
    realpower_sensor1->publish_state(realPower1);
    float apparentPower1 = emon1.apparentPower;
    apparentpower_sensor1->publish_state(apparentPower1);
    float powerFactor1 = emon1.powerFactor;
    powerfactor_sensor1->publish_state(powerFactor1);

```

```

float supplyVoltage1 = emon1.Vrms;
supplyvoltage_sensor1->publish_state(supplyVoltage1);
float current1 = emon1.Irms;
current_sensor1->publish_state(current1);
esp_task_wdt_reset(); // Things can take some time... this
ensures the watchdog is aware
// Phase 2
emon2.calcVI(CROSSINGS, 2000);
float realPower2 = emon2.realPower;
realpower_sensor2->publish_state(realPower2);
float apparentPower2 = emon2.apparentPower;
apparentpower_sensor2->publish_state(apparentPower2);
float powerFactor2 = emon2.powerFactor;
powerfactor_sensor2->publish_state(powerFactor2);
float supplyVoltage2 = emon2.Vrms;
supplyvoltage_sensor2->publish_state(supplyVoltage2);
float current2 = emon2.Irms;
current_sensor2->publish_state(current2);
esp_task_wdt_reset(); // Things can take some time... this
ensures the watchdog is aware
/*
// Phase 3
emon3.calcVI(CROSSINGS, 2000);
float realPower3 = emon3.realPower;
realpower_sensor3->publish_state(realPower3);
float apparentPower3 = emon3.apparentPower;
apparentpower_sensor3->publish_state(apparentPower3);
float powerFactor3 = emon3.powerFactor;
powerfactor_sensor3->publish_state(powerFactor3);
float supplyVoltage3 = emon3.Vrms;
supplyvoltage_sensor3->publish_state(supplyVoltage3);
float current3 = emon3.Irms;
current_sensor3->publish_state(current3);
*/
/*

```



```

        // Totals 1 phase - uncomment only this block if you are
reading one phase
        float realPower_total = emon1.realPower;
        realpower_sensor_total->publish_state(realPower_total);
        float apparentPower_total = emon1.apparentPower;
        apparentpower_sensor_total-
>publish_state(apparentPower_total);
        float current_total = emon1.Irms;
        current_sensor_total->publish_state(current_total);
        */
        // Totals 2 phases - uncomment only this block if you are
reading two phases
        float realPower_total = emon1.realPower + emon2.realPower;
        realpower_sensor_total->publish_state(realPower_total);
        float apparentPower_total = emon1.apparentPower +
emon2.apparentPower;
        apparentpower_sensor_total-
>publish_state(apparentPower_total);
        float current_total = emon1.Irms + emon2.Irms;
        current_sensor_total->publish_state(current_total);
        /*
        // Totals 3 phases - uncomment only this block if you are
reading three phases
        float realPower_total = emon1.realPower + emon2.realPower
+ emon3.realPower;
        realpower_sensor_total->publish_state(realPower_total);
        float apparentPower_total = emon1.apparentPower +
emon2.apparentPower + emon3.apparentPower;
        apparentpower_sensor_total-
>publish_state(apparentPower_total);
        float current_total = emon1.Irms + emon2.Irms +
emon3.Irms;
        current_sensor_total->publish_state(current_total);
        */
    }
};

```