

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ БІЗНЕС-КОЛЕДЖ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
(повна назва випускової кафедри)

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему
СИСТЕМА КОНТРОЛЮ ЕНЕРГОСПОЖИВАННЯ НА ОСНОВІ
МІКРОКОНТРОЛЕРІВ

Виконав: студент групи 1КІ-23
спеціальності
123 - „Комп'ютерна інженерія”
(шифр і назва спеціальності)

Шуляк Антон Вікторович
(прізвище та ініціали)

Науковий керівник Розломій Інна Олександрівна
(науковий ступінь, вчене звання, прізвище та ініціали)

Черкаси 2025

ЗМІСТ

ВСТУП		4
РОЗДІЛ 1. Теоретичні основи систем контролю енергоспоживання		6
1.1	Актуальність проблеми контролю енергоспоживання	6
1.2	Огляд методів вимірювання та аналізу енергоспоживання	8
1.3	Аналіз мікроконтролерних систем в задачах енергоменеджменту	13
1.3.1	Мікроконтролери	13
1.3.2	Огляд існуючих рішень	17
РОЗДІЛ 2. Проектування системи контролю енергоспоживання		23
2.1	Постановка задачі та визначення вимог до системи	23
2.2	Вибір апаратної та програмної платформи для віртуальної реалізації	25
2.3	Розробка схеми та логіки функціонування системи у симуляторі	32
РОЗДІЛ 3. Віртуальне моделювання та тестування системи		39
3.1	Реалізація моделі системи в середовищі Wokwi	39
3.2	Віртуальне вимірювання параметрів енергоспоживання	40
3.3	Аналіз результатів моделювання та перевірка відповідності вимогам	45
ВИСНОВКИ		48
СПИСОК ВИКОРИСТАНИХ		50

ДЖЕРЕЛ		
ДОДАТКИ		52

ВСТУП

Сучасний розвиток енергетичних технологій супроводжується як зростанням загального енергоспоживання, так і підвищенням вимог до енергоефективності в усіх сферах життєдіяльності. В умовах зростання тарифів на енергоносії та глобальної потреби в ощадливому використанні ресурсів особливого значення набуває контроль та оптимізація енергоспоживання, особливо в побуті. Одним із перспективних напрямів вирішення цього завдання є використання систем моніторингу на базі мікроконтролерів, які дають змогу збирати, аналізувати та візуалізувати дані про споживання електроенергії в режимі реального часу.

З огляду на стрімкий розвиток технологій Інтернету речей (IoT) та широке впровадження мікроконтролерів у повсякденні електронні системи, зростає потреба у створенні адаптивних та доступних засобів контролю енергоспоживання. Такі пристрої можуть бути ефективно використані в житлових будинках, офісах або на малих підприємствах для моніторингу навантажень, виявлення надлишкових витрат енергії та прийняття рішень щодо підвищення енергоефективності.

Актуальність теми. На фоні постійного зростання вартості електроенергії, а також збільшення навантаження на енергосистеми, проблема раціонального використання енергії набуває критичного значення. Особливої актуальності це питання набуває в контексті децентралізованого електропостачання, розвитку відновлюваних джерел енергії та впровадження розумного дому. Пристрої контролю енергоспоживання, зокрема на базі мікроконтролерів, дозволяють реалізувати доступні засоби зворотного зв'язку між користувачем та споживаними електричними приладами, сприяючи економії та стабільній роботі системи.

Мета та завдання дослідження. Метою кваліфікаційної роботи є розробка функціональної моделі пристрою для контролю енергоспоживання

на основі мікроконтролера з можливістю візуалізації даних у реальному часі для побутового використання.

Для досягнення поставленої мети передбачено розв'язання таких завдань:

обґрунтувати актуальність та необхідність впровадження систем контролю енергоспоживання в побуті;

– провести огляд сучасних апаратних та програмних рішень для реалізації подібних систем;

– розробити концептуальну модель пристрою на базі мікроконтролера з відповідним функціоналом збору та передачі даних;

– здійснити віртуальне моделювання та тестування пристрою.

– оцінити потенційні результати застосування запропонованої системи у побутових умовах.

Об'єкт і предмет дослідження.

Об'єктом дослідження є процеси енергоспоживання в умовах житлового будинку.

Предметом дослідження є система моніторингу енергоспоживання на основі мікроконтролера.

Методи дослідження.

У роботі використано методи аналізу літературних джерел, структурно-функціонального синтезу, віртуального моделювання схемотехнічних рішень, а також порівняльного аналізу технічних засобів з метою вибору оптимальних компонентів для проєкту.

Практичне значення.

Запропонована система може бути використана як основа для побудови недорогих побутових пристроїв моніторингу енергоспоживання. Вона дозволяє відстежувати роботу електроприладів, аналізувати споживання в динаміці та приймати рішення щодо енергоощадного режиму експлуатації.

РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ КОНТРОЛЮ ЕНЕРГОСПОЖИВАННЯ

1.1. Актуальність проблеми контролю енергоспоживання

У сучасному світі проблема раціонального використання енергії стає дедалі важливішою. Стрімке зростання енергоспоживання, що обумовлене урбанізацією, цифровізацією та зростанням чисельності населення, призводить до підвищення навантаження на енергетичні системи. Це, в свою чергу, спричиняє не лише економічні витрати, але й екологічні наслідки, пов'язані зі зміною клімату, викидами парникових газів та зниженням якості навколишнього середовища. Саме тому питання контролю та оптимізації енергоспоживання набуває глобальної актуальності.

Контроль енергоспоживання — це процес моніторингу, аналізу та регулювання витрат енергії з метою досягнення максимальної енергоефективності. Він охоплює як промислові об'єкти, так і побутові та комерційні приміщення. Запровадження систем контролю енергоспоживання дозволяє не лише знизити витрати, але й сприяє підвищенню надійності енергетичних систем, зменшенню навантаження на електромережі та досягненню екологічної сталості.

Особливої актуальності ця проблема набуває в умовах енергетичної кризи, зростання цін на енергоносії та необхідності забезпечення енергетичної безпеки. В таких умовах важливо впроваджувати інноваційні технології, зокрема мікроконтролерні системи, які дозволяють реалізувати ефективні та недорогі рішення для контролю енергоспоживання.

Згідно з даними Міжнародного енергетичного агентства (ІЕА), загальне енергоспоживання дата-центрів у світі неухильно зростає. На графіку зображено динаміку споживання електроенергії у ТВт·год за типами обладнання в період з 2020 до 2030 року (включаючи прогнозовані значення з 2025 року) (див. рис 1.1). Найбільший внесок у споживання внесли сервери

та системи охолодження. Після 2024 року спостерігається стрімке зростання, що пояснюється активним впровадженням технологій хмарних обчислень, штучного інтелекту та обробки великих даних.

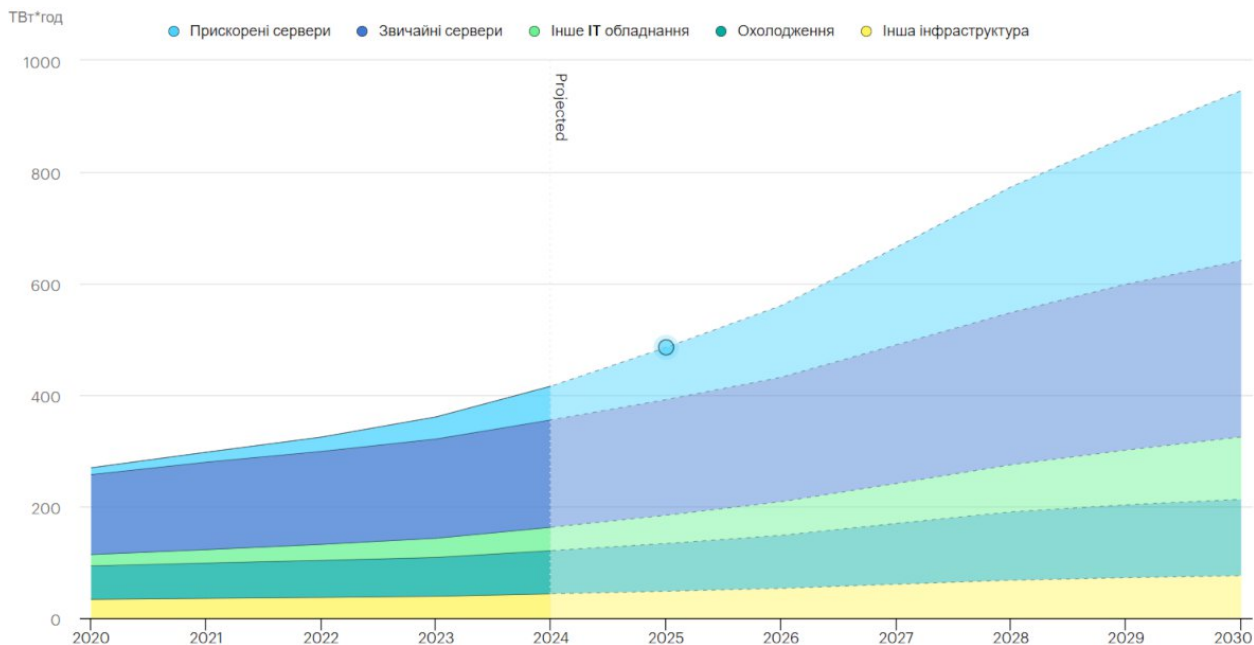


Рисунок 1.1 – Графік динаміки споживання електроенергії у ТВт·год за типами обладнання

Цей тренд підкреслює необхідність ефективного моніторингу, контролю та оптимізації енергоспоживання, зокрема через впровадження систем на базі мікроконтролерів, які дозволяють забезпечити точний облік, гнучке керування та своєчасне виявлення перевантажень чи неефективного використання енергії.

В Україні ситуація має свої особливості. Через військові дії, старіння інфраструктури та залежність від імпортованих енергоресурсів, питання енергоефективності стоїть особливо гостро. За оцінками Державного агентства з енергоефективності та енергозбереження України, потенціал скорочення споживання енергії в побутовому секторі сягає 40%, що свідчить про надзвичайну актуальність впровадження засобів контролю.

З розвитком інформаційних технологій з'явилась можливість реалізувати недорогі, компактні та енергоефективні системи моніторингу енергоспоживання. Мікроконтролери, такі як Arduino, ESP32 або STM32, дозволяють створювати автономні пристрої, здатні фіксувати параметри електричного струму, напруги, потужності, а також передавати ці дані до хмари або на локальні сервери для подальшого аналізу.

Контроль енергоспоживання сприяє зменшенню фінансового навантаження на домогосподарства та підприємства, дозволяє відмовитися від додаткових генеруючих потужностей, підвищує стабільність енергосистем, зменшує викиди CO₂, що критично важливо в умовах змін клімату.

1.2. Загальні принципи енергомоніторингу

З метою забезпечення ефективного контролю та управління електроспоживанням у побутових та промислових умовах розроблено низку методів вимірювання електричних параметрів, зокрема струму, напруги, активної та реактивної потужності, енергії, коефіцієнта потужності тощо. Залежно від технічних вимог, доступного бюджету, необхідної точності та умов експлуатації, використовуються як традиційні аналогові підходи, так і сучасні цифрові технології, інтегровані у мікропроцесорні системи.

Основу для аналізу енергоспоживання становлять первинні вимірювання миттєвих значень струму $I(t)$ та напруги $U(t)$, на основі яких розраховуються активна потужність P , реактивна Q та повна потужність S , відповідно до таких виразів:

$$P=U_{rms} \cdot I_{rms} \cdot \cos\varphi, \quad Q=U_{rms} \cdot I_{rms} \cdot \sin\varphi, \quad S=U_{rms} \cdot I_{rms}$$

де φ — фазовий зсув між струмом і напругою, а U_{rms} , I_{rms} — діюче значення напруги та струму відповідно.

Вимірювання електричного струму є ключовим етапом у системах контролю енергоспоживання, оскільки саме від точності отриманих даних

залежить коректність обчислення спожитої потужності та енергії. Існує кілька принципово різних підходів до вимірювання струму, які застосовуються залежно від конкретних умов експлуатації, вимог до точності, електробезпеки та вартості системи.

Одним із найпростіших і найбільш поширених способів є використання шунта (див. рис 1.2) — спеціального резистора з точно відомим низьким опором, через який пропускається вимірюваний струм. Внаслідок цього на резисторі виникає падіння напруги, пропорційне струму згідно з законом Ома. Цю напругу зчитує аналогово-цифровий перетворювач мікроконтролера, після чого здійснюється програмне перетворення в числове значення струму. Шунтовий метод має переваги у вигляді низької вартості та простої реалізації, однак його недоліками є відсутність гальванічної розв'язки, обмеження по допустимому струму та залежність точності від температурної стабільності елементів.



Рисунок 1.2 – Загальний вигляд шунта для вимірювання струму

Альтернативою, що забезпечує кращу електробезпеку, є використання трансформатора струму. Цей пристрій працює за принципом електромагнітної індукції: основний струм проходить через первинну обмотку, індукуючи змінну напругу у вторинній обмотці, яка пропорційна змінному струму. Пристрій надягається на дрід через який протікає струм (див. рис 1.3). Такий метод дозволяє вимірювати значні струми без прямого підключення до силового кола, що особливо актуально у промислових системах або в розподільчих щитах. Водночас цей метод непридатний для вимірювання

постійного струму, а точність залежить від частотних характеристик трансформатора та рівня магнітного насичення.



Рисунок 1.3 – Загальний вигляд трансформатора для вимірювання струму

Одним із найбільш універсальних рішень є датчики струму на ефекті Холла, які забезпечують гальванічну розв'язку та можуть працювати як з постійним, так і з змінним струмом (див. рис 1.4).



Рисунок 1.4 – Загальний вигляд датчика для вимірювання струму на ефекті Холла

У таких датчиках струм протікає через внутрішній провідник, навколо якого створюється магнітне поле. Це поле реєструється сенсором Холла, що генерує пропорційний аналоговий сигнал. До найпопулярніших представників цього класу належать інтегральні датчики типу ACS712, ACS758 тощо. Основними перевагами цього методу є простота інтеграції, універсальність, безпека, а також можливість безперервного моніторингу струму в обох напрямках (наприклад, у сонячних інверторах чи зарядних станціях). До

недоліків відносяться відносно невисока точність у нижньому діапазоні вимірювань, температурна нестабільність та необхідність калібрування.

У деяких випадках застосовуються індуктивні безконтактні методи або оптичні сенсори, однак вони характерні переважно для спеціалізованих застосувань, зокрема у високочастотних або високовольтних лініях, де прямий контакт із провідником є неможливим або небажаним.

Підсумовуючи, вибір методу вимірювання струму має базуватися на аналізі характеристик електричної системи, вимог до точності, габаритів, наявності джерел перешкод, а також пріоритетів щодо вартості та безпеки. У системах контролю енергоспоживання на базі мікроконтролерів, орієнтованих на домашнє використання, найбільш доцільним є застосування датчиків Холла як компроміс між функціональністю, простотою реалізації та електробезпекою.

Вимірювання електричної напруги є необхідною складовою будь-якої системи контролю енергоспоживання, оскільки разом із даними про струм ці величини дозволяють обчислювати активну, реактивну та повну потужність, а також енергію, спожиту навантаженням. У контексті мікропроцесорних систем, які працюють з низьковольтними аналоговими сигналами, важливо забезпечити не лише точність вимірювання, але й електробезпеку та стабільність у роботі при наявності шумів та перепадів напруги.

Найпростішим і водночас ефективним способом вимірювання змінної напруги є використання резистивного дільника напруги. Цей метод базується на принципі послідовного з'єднання двох резисторів, які знижують рівень вхідної змінної напруги до безпечного значення, придатного для оцифрування аналогово-цифровим перетворювачем мікроконтролера. Значення резисторів добираються з урахуванням максимальної напруги в мережі (зазвичай 220 В ефективного значення або близько 311 В пікового) та діапазону допустимого вхідного сигналу АЦП, який для більшості мікроконтролерів становить 0–3.3В

або 0–5В. Для уникнення пошкодження електроніки на вході часто встановлюють захисні діоди або стабілізатори напруги.

Недоліком цього методу є відсутність гальванічної розв'язки між силовим колом та мікроконтролером, що вимагає високого рівня ізоляції, правильного проектування друкованої плати та дотримання правил техніки безпеки. Також можливі неточності вимірювання через теплову нестабільність резисторів або індуктивні перешкоди з боку мережі.

Більш безпечним, хоча й складнішим методом, є використання трансформатора напруги, який забезпечує гальванічну розв'язку і знижує змінну напругу до рівня кількох вольт (див. рис 1.5). Змінний сигнал з виходу трансформатора можна далі обробляти аналогічно, як і в методі з дільником. Такий підхід особливо актуальний для систем, де пріоритетом є безпека персоналу та надійність у промисловому середовищі. Проте трансформатори мають власну похибку, пов'язану з фазовим зсувом, магнітною гістерезисом і частотною залежністю.

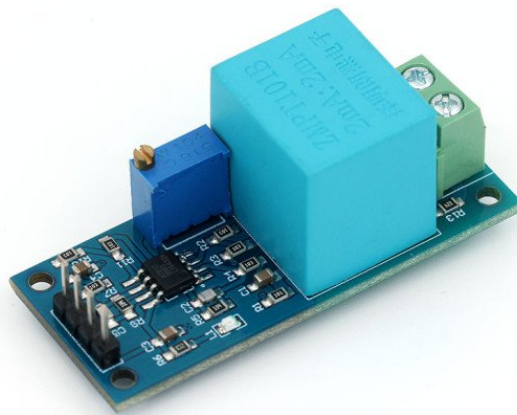


Рисунок 1.5 – Загальний вигляд модуля для вимірювання напруги на основі понижувального трансформатора напруги

Серед сучасних підходів також зустрічаються оптоізовані схеми, які поєднують в собі резистивний або трансформаторний дільник з оптронним каскадом. Такий підхід дозволяє передати сигнал про рівень напруги без прямого електричного з'єднання. Оптична розв'язка має високу стійкість до

імпульсних перенапруг та електромагнітних завад, однак ускладнює схему та потребує стабільного живлення.

Незалежно від обраного методу зниження напруги, сигнал із мережі має форму синусоїди з частотою 50 Гц, тому часто застосовується програмна фільтрація та визначення діючого значення (RMS) за вибіркою з АЦП. Також при необхідності виявлення перевищень порогових рівнів (наприклад, при перенарузі) використовуються компаратори або опорні напруги.

1.3 Аналіз мікроконтролерних систем в задачах енергоменеджменту

1.3.1 Мікроконтролери

Мікроконтролер — це мініатюрна обчислювальна система на одному кристалі, яка об'єднує в собі процесор, пам'ять, периферійні інтерфейси та засоби введення/виведення. Її ключовим завданням є обробка інформації з датчиків, прийняття рішень за заданим алгоритмом та керування виконавчими пристроями. У системах енергоменеджменту мікроконтролери забезпечують збір, фільтрацію, аналіз та передачу даних про споживання енергії, а також реалізацію алгоритмів оптимізації та автоматичного керування навантаженнями.

Одним з основних напрямів застосування мікроконтролерів у сфері енергоменеджменту є побудова інтелектуальних систем моніторингу енергоспоживання. Такі системи дозволяють вимірювати параметри електричної енергії, напругу, струм, частоту, коефіцієнт потужності, активну та реактивну енергію. На основі зібраної інформації проводиться подальший аналіз та прийняття рішень щодо зменшення навантаження, виявлення аномалій, обчислення витрат та прогнозування витрат.

Серед широкого спектру мікроконтролерних платформ особливу увагу заслуговує Arduino Nano — одна з найпоширеніших систем серед розробників-початківців (див. рис 1.6).

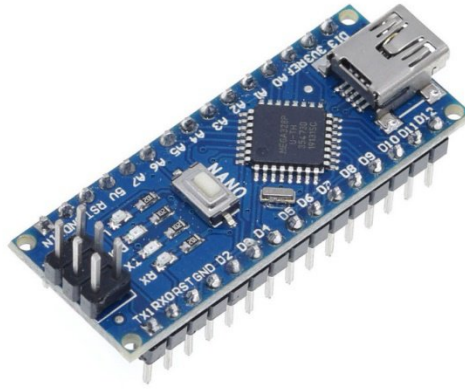


Рисунок 1.6 – Загальний вигляд Arduino Nano

Вона базується на 8-бітному мікроконтролері ATmega328P з тактовою частотою 16 МГц. Пристрій забезпечує базову функціональність: 14 цифрових входів/виходів, з яких 6 можуть бути використані для генерації ШІМ-сигналів, 6 аналогових входів, а також базову пам'ять — 32 КБ Flash, 2 КБ оперативної пам'яті (SRAM) та 1 КБ EEPROM. Завдяки великій спільноті та великій кількості бібліотек Arduino Nano є зручною платформою для розробки прототипів, хоча її обчислювальні ресурси обмежені і не дозволяють реалізовувати складні алгоритми чи обробляти великі обсяги даних у реальному часі.

Для створення бездротових енергомоніторингових систем доцільно використовувати мікроконтролери серії ESP, зокрема ESP8266 (див. рис 1.7). Цей мікроконтролер містить одне ядро Tensilica, яке працює на частоті 80 МГц, та має вбудований Wi-Fi-модуль, що дозволяє передавати дані у хмару або на локальний сервер без додаткових зовнішніх пристроїв.

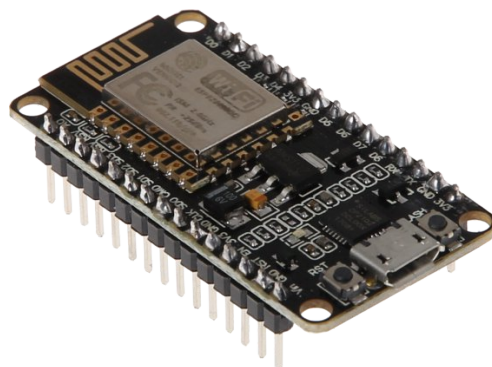


Рисунок 1.7 – Загальний вигляд ESP8266

Хоча оперативна пам'ять обмежена (близько 96 КБ), ESP8266 чудово підходить для задач з періодичним вимірюванням і надсиланням результатів. Його застосовують у зв'язці з протоколами MQTT, HTTP та з хмарними платформами типу Blynk або Thingspeak. Обмежена кількість GPIO та відсутність Bluetooth є певними недоліками, однак для багатьох побутових рішень цієї функціональності достатньо.

ESP32, наступник ESP8266, значно перевершує його за характеристиками. Це мікроконтролер з двома ядрами Xtensa LX6, тактовою частотою до 240 МГц, великим обсягом внутрішньої пам'яті (до 520 КБ SRAM) та можливістю підключення до 4 МБ Flash. Найважливішою перевагою ESP32 є його багатий набір інтерфейсів: окрім Wi-Fi, він має Bluetooth/BLE, до 18 12-бітних АЦП, 2 ЦАП, апаратне шифрування, сенсор температури та ємнісні сенсори. Це дає змогу реалізовувати складні системи аналізу споживання з багатофакторним керуванням та віддаленим контролем.

STM32F103C8T6 (див. рис 1.8), також відомий під назвою "Blue Pill", є представником потужної 32-бітної лінійки ARM Cortex-M3. З тактовою частотою 72 МГц та 64 КБ Flash-пам'яті, цей мікроконтролер пропонує широкий вибір інтерфейсів: USB, UART, SPI, I2C, ADC, PWM, що робить його придатним для високоточних вимірювань, цифрової обробки сигналів та реалізації складних алгоритмів.

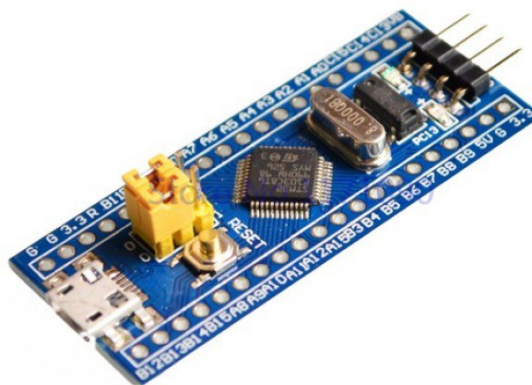


Рисунок 1.8 – Загальний вигляд STM32F103C8T6

Його перевага — це висока стабільність, велика кількість периферійних модулів та низьке енергоспоживання. Водночас робота з STM32 потребує знання архітектури ARM, що може бути викликом для початківців.

Новітньою розробкою є Raspberry Pi Pico (див. рис 1.9), побудований на базі власного мікроконтролера RP2040. Він має два ядра Cortex-M0+ з тактовою частотою до 133 МГц, 264 КБ SRAM та 2 МБ Flash. Його головна перевага — наявність програмованих інтерфейсів PIO (Programmable IO), які дозволяють створювати власні інтерфейси на апаратному рівні, що відкриває нові горизонти для нестандартних задач. Raspberry Pi Pico також підтримує розробку на MicroPython, що суттєво спрощує реалізацію логіки керування для освітніх проєктів та систем середньої складності.

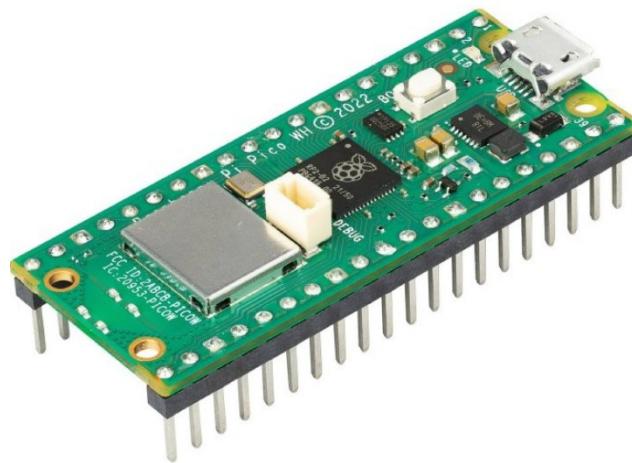


Рисунок 1.9 – Загальний вигляд Raspberry Pi Pico

Для отримання інформації про споживання енергії мікроконтролери зазвичай використовуються разом з такими сенсорними модулями, як ACS712 (вимірювання струму), ZMPT101B (вимірювання напруги), HLW8012, ADE7753 (розширене вимірювання активної, реактивної та повної енергії). Оброблені дані передаються через інтерфейси UART, SPI, I2C або Wi-Fi/Bluetooth, залежно від архітектури системи.

У більш складних системах реалізується концепція Smart Grid, що передбачає використання мікроконтролерів для збору інформації з різних

вузлів енергомережі, її централізовану обробку, формування рекомендацій користувачу або автоматичне регулювання режимів роботи пристроїв. Наприклад, система може виявити перевищення встановленого ліміту споживання й автоматично відключити частину навантаження або змінити режим роботи системи опалення.

Особливої уваги заслуговує можливість реалізації програмної логіки, що враховує часові характеристики споживання, історію використання, погодні умови та інші зовнішні фактори. Це досягається через використання алгоритмів штучного інтелекту, машинного навчання та нейромереж, що можуть бути частково реалізовані на потужніших мікроконтролерах або із залученням зовнішніх обчислювальних модулів.

Таким чином, мікроконтролерні системи є базовим інструментом для побудови енергоефективних, адаптивних та масштабованих систем керування споживанням енергії. Їх використання дозволяє не лише зменшити енергетичні витрати, а й створити інфраструктуру для подальшого розвитку розумного середовища (smart home, smart city), підвищити надійність та ефективність енергетичних систем у цілому.

1.3.2 Огляд існуючих рішень

Існує широкий спектр пристроїв та систем для контролю енергоспоживання, які реалізуються на базі мікроконтролерів. Більшість з них орієнтовані на побутове або комерційне застосування і поділяються за функціональністю на дві великі групи: пасивні системи моніторингу (тільки збирають дані) та активні системи керування енергоспоживанням (здійснюють вплив на навантаження на основі аналізу даних).

1. AP-50 — це побутовий модуль релейного типу, призначений для автономного захисту навантаження від перевищення струму. Пристрій реалізовано у вигляді компактного модуля (див. рис 1.10), що легко інтегрується у розрив ланцюга живлення та не потребує підключення до інтернету чи зовнішнього керування. AP-50 може застосовуватись як реле

захисту, так і як реле запуску навантаження після досягнення певного струму (наприклад, для виявлення споживача).



Рисунок 1.10 – Загальний вигляд реле струму AP-50

№	Показник	Величина
1	Діапазон вимірювання струму, А	1-70А
2	Програмований поріг відключення за струмом	1 – 50А
3	Час затримки на ввімкнення	1-20 мин
4	Похибка вимірювання	не більше 1%
5	Трансформатор струму	Вбудований

Таблиця 1.1 – Характеристики реле струму AP-50

2. РКТ-16/100 — реле контролю струму призначене для автономного моніторингу струму в колі змінного струму до 16 А та автоматичного вимикання навантаження у разі перевищення заданого порогу (див. рис 1.11). Воно належить до електромеханічних реле промислового або побутового застосування і монтується в розподільчі щити або окремі шафи управління.



Рисунок 1.11 – Загальний вигляд реле струму PKT-16/100

№	Показник	Величина
1	Діапазон вимірювання струму, А	1-150А
2	Програмований поріг відключення за струмом	1 – 50А
3	Час затримки на ввімкнення	0 - 4 хв.
4	Похибка вимірювання	не більше 1%
5	Трансформатор струму	Вбудований
6	Номинальний струм	16 або 100 А (залежно від метода підключення)

Таблиця 1.2 – Характеристики реле струму PKT-16/100

3. ZUBR RET I25 — Пристрій призначено для локального контролю струму в побутових електромережах, де не допускається перевищення допустимого навантаження. ZUBR RET I25 працює без підключення до інтернету, є повністю автономним, і може бути інтегрованим у щитову автоматику квартир, приватних будинків, майстерень (див. рис 1.12). Він реагує на перевищення струму споживання та автоматично розриває коло живлення, захищаючи від перегріву, пошкодження приладів або загоряння.



Рисунок 1.12 – Загальний вигляд реле струму ZUBR RET I25

№	Показник	Величина
1	Діапазон вимірювання струму, А	0,5–30 А
2	Програмований поріг відключення за струмом	1–25 А
3	Номінальний струм навантаження	25 А
4	Час затримки на вмикнення	1–600 с
5	Похибка вимірювання	не більше 1%
6	Тип реле	електромеханічне, нормально відкритий контакт

Таблиця 1.3 – Характеристики реле струму ZUBR RET I25

№	Показник	Величина
7	Живлення	100–420 В, 50 Гц (універсальне, однофазне)
8	Трансформатор струму	Вбудований
9	Тип монтажу	DIN-рейка, 1 модуль (35 мм)
10	Інтерфейс користувача	Світлодіодний дисплей + сенсорні кнопки

Таблиця 1.3 – Характеристики реле струму ZUBR RET I25

Аналіз сучасних засобів контролю енергоспоживання показав, що на ринку представлені як багатофункціональні інтелектуальні пристрої з можливістю підключення до інтернету, так і прості автономні реле, що не потребують зовнішнього керування. Значне поширення отримали автономні пристрої контролю струму — зокрема, AP-50, PКТ-16/100, ZUBR RET I25. Вони не потребують жодних зовнішніх інтерфейсів, виконують контроль та відключення навантаження виключно на основі локальних вимірювань струму, забезпечують миттєву реакцію на перевищення порогів і є незалежними від цифрової інфраструктури. Це робить їх надзвичайно ефективними для систем локального захисту та енергоконтролю, особливо в побутових умовах, де на перше місце виходять надійність, енергонезалежність та простота інтеграції.

Також варто відзначити, що більшість готових рішень мають або вузьку спеціалізацію, або фіксовані функціональні обмеження, які ускладнюють адаптацію під нестандартні задачі, наприклад, зміну логіки роботи при певних значеннях струму або комбіновані алгоритми реагування. Це створює передумови для розробки універсального та адаптованого пристрою на базі мікроконтролера, який дозволить гнучко реалізувати функції вимірювання,

порівняння з порогоми, відображення на дисплеї, збереження режимів і автономну комутацію навантаження.

Таким чином, аналіз існуючих рішень обґрунтовує доцільність проектування індивідуального мікроконтролерного пристрою, який би поєднував переваги простих автономних реле з розширеною функціональністю, адаптованою до специфічних вимог споживача. Такий підхід дозволяє не лише знизити витрати, але й підвищити гнучкість, точність та інформативність системи контролю енергоспоживання.

РОЗДІЛ 2 ПРОЄКТУВАННЯ СИСТЕМИ КОНТРОЛЮ ЕНЕРГОСПОЖИВАННЯ

2.1. Постановка задачі та визначення вимог до системи

Сучасні тенденції розвитку енергетики зумовлюють потребу у вдосконаленні механізмів обліку та контролю споживання електроенергії, особливо в контексті підвищення ефективності енергоспоживання в побутовому секторі. Більшість наявних комерційних рішень зосереджені на масштабних інсталяціях, що орієнтовані на потреби підприємств, житлових комплексів або виробничих об'єктів. У той час як серед пересічних споживачів електроенергії, які не мають спеціалізованої технічної підготовки, відчувається брак доступних, простих у налаштуванні та використанні пристроїв, здатних здійснювати точний та надійний облік енергоспоживання.

Актуальність розробки побутової системи контролю енергоспоживання зумовлена кількома факторами. По-перше, це прагнення користувача мати уявлення про структуру власного споживання електроенергії з метою оптимізації навантажень і зниження витрат. По-друге, зростання популярності концепції розумного дому, у якій контроль над енергоспоживанням відіграє ключову роль. По-третє, значний потенціал таких систем для інтеграції з мобільними додатками, хмарними сервісами, а також платформами для енергетичного моніторингу на рівні мікрогрід-систем.

У цьому контексті постає завдання розробити систему, яка надає користувачеві можливість у реальному часі отримувати інформацію про споживану електроенергію, аналізувати її зміну з часом, а також виявляти аномалії, які можуть бути пов'язані з перевантаженням мережі або несправністю окремих пристроїв. Розробка повинна спиратися на використання мікроконтролера як основного обчислювального вузла, який буде забезпечувати збір даних із сенсорів, обробку отриманих значень, їх збереження або передавання, а також взаємодію з користувачем.

Постановка задачі передбачає реалізацію функціонального комплексу, який забезпечить виконання кількох ключових функцій. Зокрема, необхідно реалізувати можливість вимірювання таких параметрів, як напруга, сила струму, активна та реактивна потужність. Також повинна бути реалізована функція обчислення добового, тижневого та місячного споживання енергії із подальшим збереженням у вбудованій пам'яті або відправкою даних у віддалене сховище. Обов'язковим є й механізм візуалізації даних – або через локальний дисплей, або за допомогою мобільного додатку чи вебінтерфейсу. Підсистема сигналізації має оперативно інформувати користувача про перевищення порогових значень споживання.

Технічні вимоги до системи мають охоплювати характеристики мікроконтролера, типи сенсорів для вимірювання електричних параметрів, методи фільтрації та калібрування сигналів, особливості зберігання даних, типи інтерфейсів для передачі інформації, а також механізми енергетичної безпеки та надійності системи. Передбачається використання безконтактних або шунтових сенсорів струму з урахуванням вимог електробезпеки та електромагнітної сумісності. Вибір мікроконтролера здійснюється на основі аналізу його продуктивності, кількості аналогових входів, наявності інтерфейсів Wi-Fi, Bluetooth або UART, енергоефективності, підтримки протоколів передачі даних, а також сумісності з наявними бібліотеками для обробки сигналів.

Функціональні вимоги до системи охоплюють можливість адаптації до різних навантажень (від малопотужних побутових приладів до енергоємних споживачів), підтримку простого налаштування через вбудований вебінтерфейс або мобільний застосунок, автономну роботу з резервним живленням та функцію збереження останніх даних у разі збоїв або відключення живлення. Також передбачається, що система буде здатна працювати з обмеженим набором ресурсів (обчислювальні можливості, об'єм пам'яті), що вимагає оптимізованих програмних рішень.

Експлуатаційні вимоги охоплюють простоту монтажу пристрою в умовах типового житла, відсутність потреби у втручанні в мережу змінного струму, сумісність з уже існуючими елементами побутової електромережі, стійкість до короткочасних стрибків напруги, а також зрозумілий інтерфейс взаємодії з користувачем. Додатково варто передбачити можливість оновлення прошивки мікроконтролера без необхідності фізичного доступу до плати.

Таким чином, поставлене завдання полягає не лише в розробці апаратної частини системи, а й у створенні програмного середовища, яке забезпечує обробку сигналів, зберігання результатів, формування аналітичних звітів, передавання даних, а також зручну візуалізацію. Система повинна відповідати принципам модульності, масштабованості, мінімізації витрат на розгортання та експлуатацію. Отримані технічні й функціональні вимоги в подальшому формуватимуть основу для розробки концептуальної, структурної та електричної схем, алгоритмів програмної обробки, а також для вибору компонентної бази, описаних у наступних розділах.

2.2. Вибір апаратної платформи для віртуальної реалізації

Виходячи з поставлених функціональних вимог до системи контролю енергоспоживання, вибір апаратної складової має базуватися на низці ключових критеріїв. Оптимальним вибором для реалізації системи є мікроконтролерна плата Arduino Nano (див. рис 1.6). Вона є однією з найбільш поширених платформ у сфері аматорських і навчальних розробок систем автоматизації, електроніки та Інтернету речей (IoT). Arduino Nano базується на мікроконтролері ATmega328P, який має 32 КБ флеш-пам'яті для збереження прошивки, 2 КБ оперативної пам'яті SRAM, 1 КБ EEPROM, а також працює з тактовою частотою 16 МГц (див. таб. 2.1).

№	Параметр	Значення
1	Мікроконтролер	ATmega328P
2	Напруга живлення (рекомендована)	7–12 В

3	Напруга логіки (рівень I/O)	5 В
4	Кількість цифрових входів/виходів	14 (з них 6 можуть бути використані як ШІМ-виходи)
5	Кількість аналогових входів	8
6	Максимальний струм на пін I/O	40 мА
7	Флеш-пам'ять	32 КБ (з яких 2 КБ використовується завантажувачем)
8	ОЗП (SRAM)	2 КБ
9	ПЗП (EEPROM)	1 КБ
10	Тактова частота	16 МГц
11	USB-інтерфейс	Mini-USB (використовується для програмування та живлення)
12	Програмування	Через USB (використовується USB-UART перетворювач CH340 або FT232)
13	Габарити	18 мм × 45 мм
14	Вага	~7 грамів

Таблиця 2.1 – Характеристики Arduino Nano

Ці характеристики забезпечують достатній обсяг ресурсів для обробки даних сенсорів, керування виконавчими механізмами та організації взаємодії з користувачем.

Однією з важливих переваг Arduino Nano є її компактність і модульність. Плата має невеликі розміри (18 x 45 мм), що дозволяє легко розміщувати її на макетних платах без потреби в додаткових перехідниках. Крім того, вона оснащена вбудованим USB-інтерфейсом, який дає змогу здійснювати програмування та моніторинг через стандартне підключення до комп'ютера. Така конфігурація значно спрощує процес розробки, особливо на етапах тестування та налагодження. Arduino Nano підтримує 14 цифрових входів/виходів (6 з яких можуть працювати в режимі ШІМ), а також 8 аналогових входів, що дозволяє підключати широкий спектр датчиків та індикаторів.

Ще однією важливою перевагою є сумісність Arduino Nano із різноманітними сенсорами, модулями індикації, комунікаційними модулями (наприклад, I2C або UART) та виконавчими пристроями. У контексті

реалізації системи контролю енергоспоживання Arduino Nano може бути інтегрована з датчиками струму, модулями вимірювання напруги, а також з релейними модулями для керування електроспоживачами.

Для вимірювання сили струму в колі споживача обрано датчик ACS712 (див. рис 2.1).

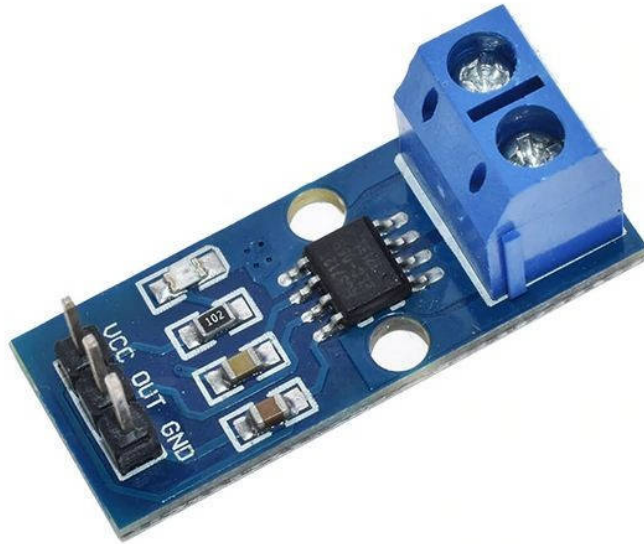


Рисунок 2.1 – Загальний вигляд ACS712

Це готовий модуль на основі ефекту Холла, який дозволяє безконтактно визначати силу змінного або постійного струму. Сигнал із шунта перетворюється в аналогову напругу, пропорційну величині струму, що зручно для зчитування через аналоговий вхід мікроконтролера Arduino Nano. ACS712 має вбудований підсилювач та фільтрацію, що забезпечує відносно стабільний та точний сигнал для обробки. Існує кілька варіантів цього датчика залежно від максимально допустимого струму: 5 А, 20 А та 30 А, що дозволяє обрати відповідний варіант для побутових навантажень. У випадку побутових приладів найчастіше доцільним є використання варіанта на 5 А (див. таб. 2.2).

№	Параметр	Значення
1	Модель	ACS712-05B

2	Діапазон вимірювання струму	± 5 А
3	Тип виходу	Аналоговий (напруга пропорційна струму)
4	Живлення	5 В
5	Напруга виходу при 0 А	≈ 2.5 В
6	Чутливість	- ACS712-05B: ~ 185 мВ/А
7	Ширина смуги пропускання	80 кГц
8	Точність	До $\pm 1.5\%$
9	Гальванічна ізоляція	Завдяки вбудованому ізольованому головному елементу
10	Робоча температура	$-40 \dots +85$ °С
11	Інтерфейс підключення до Arduino	Аналоговий вхід (наприклад, А1)

Таблиця 2.2 – Характеристики модуля ACS712-05B

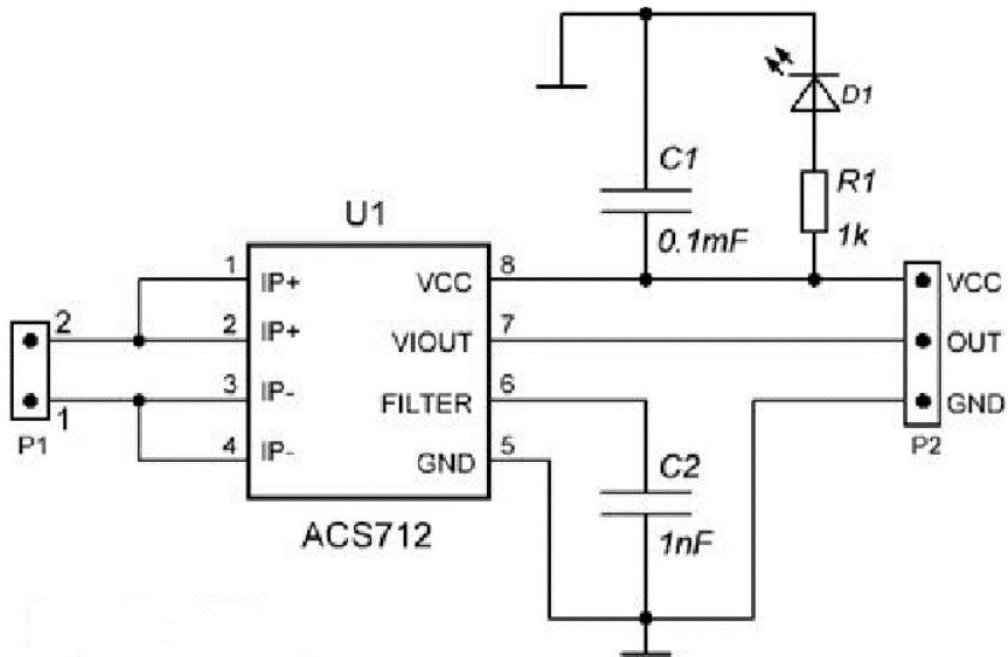


Рисунок 2.2 – Принципова схема ACS712

Для вимірювання напруги використовується модуль ZMPT101B (див. рис 2.3) — спеціалізований модуль на основі малопотужного трансформатора, що дозволяє безпечно знімати значення змінної напруги в межах побутових рівнів (до 250 В змінного струму).

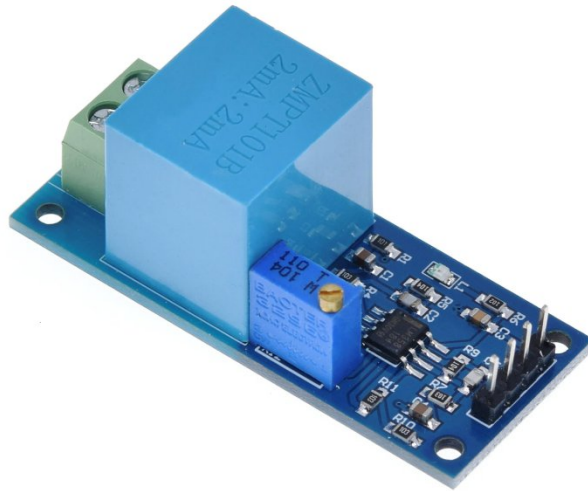


Рисунок 2.3 – Загальний вигляд ZMPT101B

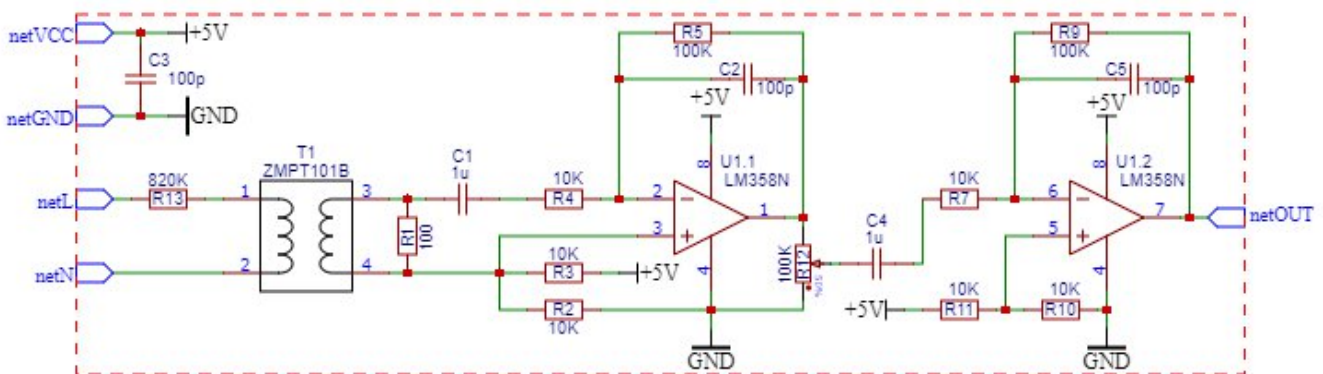


Рисунок 2.4 – Принципова схема ZMPT101B

Модуль перетворює вхідну напругу у пропорційний аналоговий сигнал, який можна зчитувати через ADC Arduino Nano. ZMPT101B має вбудований підсилювач з можливістю калібрування коефіцієнта передачі за допомогою потенціометра. Цей модуль вирізняється високою точністю і стабільністю сигналу, що дозволяє використовувати його у системах моніторингу напруги, у тому числі в реальному часі. Його простота інтеграції та підтримка у симуляторах робить його придатним для віртуального моделювання в Wokwi.

Для реалізації функцій індикації в даному проєкті обрано дисплей типу LCD1602 з інтерфейсом I2C (див. рис 2.5). Такий дисплей є дворядковим із можливістю відображення до 16 символів у кожному рядку, що є достатнім для виведення ключових параметрів: напруги, струму, потужності або стану навантаження. Інтерфейс I2C значно спрощує з'єднання дисплея з

мікроконтролером, дозволяючи використовувати лише два сигнальні дроти (SDA і SCL), що особливо актуально в умовах обмеженої кількості виводів на платі Arduino Nano. Додатково це зменшує складність монтажу й підвищує надійність з'єднань. Дисплей має вбудований контролер HD44780, підтримуваний Arduino-бібліотеками (LiquidCrystal_I2C), що полегшує процес програмної реалізації функцій виведення текстової інформації.

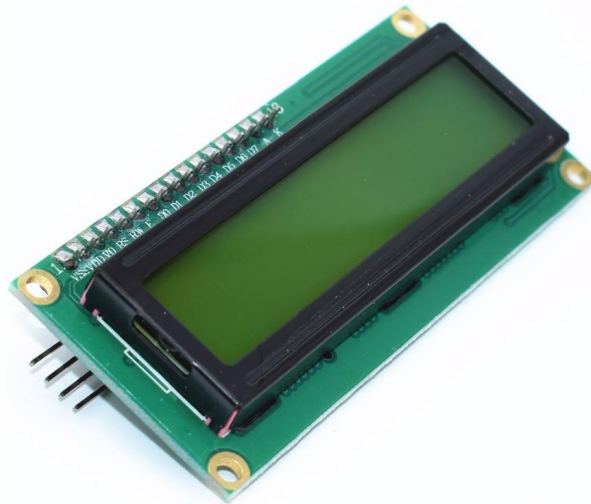


Рисунок 2.5 – Загальний вигляд дисплей типу LCD1602

Використання LCD1602 з I2C-адаптером дає змогу ефективно організувати інтерфейс користувача у віртуальному середовищі, а також дозволяє масштабувати функціонал, наприклад, додавши перемикання сторінок, вивід повідомлень або відображення помилок.

Для реалізації віртуальної моделі пристрою використовується середовище Tinkercad — популярний програмний інструмент для симуляції електронних схем та мікроконтролерних систем (див. рис 2.6). Однією з головних переваг Tinkercad є підтримка Arduino та наявність вбудованих моделей типових компонентів, зокрема LCD-дисплеїв, кнопок, реле, та інших модулів. У середовищі можна не лише будувати схеми, але й тестувати прошивку у віртуальному мікроконтролері, що дає змогу повністю змодельовати поведінку системи без фізичного збирання прототипу. Це дозволяє скоротити час на розробку, зменшити кількість помилок та провести

попередню перевірку алгоритмів роботи. Крім того, Tinkercad має графічний інтерфейс, зручний для початківців, і підтримку скриптів для складніших сценаріїв симуляції.

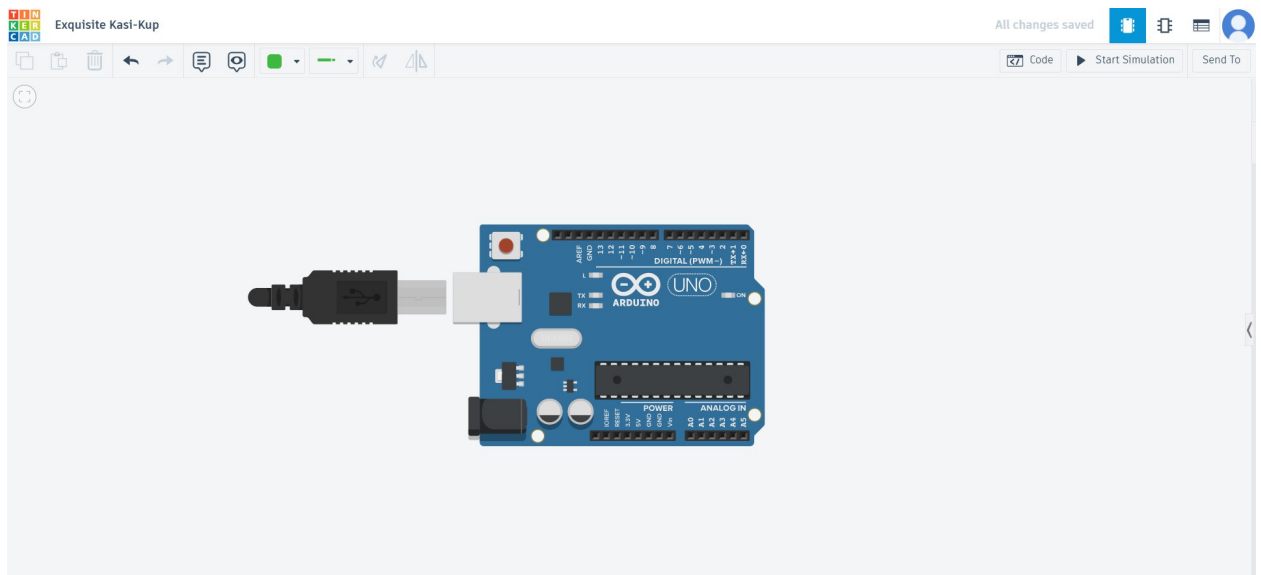


Рисунок 2.6 – Загальний вигляд інтерфейсу програми Wokwi

Arduino UNO активно підтримується в більшості середовищ моделювання, включаючи Tinkercad, що дозволяє створити повноцінну віртуальну модель системи без потреби у фізичних компонентах. Це особливо важливо на етапі прототипування, коли необхідно перевірити логіку роботи алгоритмів, протестувати взаємодію компонентів та відлагодити систему перед її фізичним виготовленням. Завдяки широкій спільноті користувачів, велика кількість бібліотек і прикладів доступна для використання, що значно скорочує час розробки.

2.3. Розробка схеми та логіки функціонування системи у симуляторі

Система контролю енергоспоживання повинна функціонувати на базі мікроконтролера Arduino UNO, який забезпечує необхідний обсяг обчислювальних ресурсів, зручне USB-підключення для програмування та достатню кількість аналогових і цифрових входів/виходів для реалізації необхідного функціоналу. Пристрій має здійснювати моніторинг електричних параметрів побутової однофазної мережі, зокрема напруги, сили струму та обчислювати миттєву потужність, а також виводити ці дані на дисплей і виконувати захисні дії у разі перевищення порогів.

Після подачі живлення пристрій повинен ініціалізувати всі необхідні модулі: I2C-шину, дисплей, входні пін-контакти кнопок керування та вихід для управління реле. На дисплей має виводитися стартове повідомлення, що сигналізує про початок роботи. Далі система повинна перейти у режим циклічного зчитування аналогових сигналів з підключених сенсорів.

Для вимірювання сили струму передбачається використання датчика ACS712, аналоговий сигнал з якого надходить на вхід A0. Для вимірювання напруги використовується модуль ZMPT101B, підключений до входу A1. Обидва сигнали повинні бути оцифровані вбудованим АЦП мікроконтролера, після чого виконується цифрова обробка (усереднення, фільтрація шумів) і перетворення у фізичні одиниці. Потужність розраховується за формулою $P=U*I$

де U — ефективне значення напруги, I — ефективне значення струму, P — миттєва потужність у ватах.

Розраховані значення повинні відображатися на LCD-дисплеї з інтерфейсом I2C, підключеному до ліній SDA (A4) та SCL (A5). У верхньому рядку дисплея мають відображатися значення напруги та струму, а в нижньому — потужність або повідомлення про спрацювання захисту.

Пристрій повинен мати реалізовану функцію автоматичного відключення навантаження. У разі перевищення напруги або струму заданих

користувачем порогових значень, мікроконтролер має активувати цифровий вихід для керування модулем реле, що розриває коло споживача. Одночасно на дисплеї повинно з'являтися відповідне повідомлення.

Для взаємодії з користувачем система має бути обладнана трьома кнопками, підключеними до цифрових входів, які дозволяють змінювати режими відображення або налаштовувати порогові значення. Також повинна бути передбачена функція автоматичного повернення до основного режиму через певний час бездіяльності.

У підсумку, пристрій має забезпечувати безперервний моніторинг параметрів електромережі, захист навантаження від перевищення граничних значень, та інтуїтивно зрозумілу індикацію для користувача.

Для забезпечення повноцінного функціонування системи контролю енергоспоживання, побудованої на мікроконтролері Arduino Nano, необхідно правильно з'єднати всі апаратні компоненти. Нижче наведено логічно структурований опис підключення кожного з них.

Перш за все, підключається датчик сили струму ACS712. Його аналоговий вихід з'єднується з входом A1 мікроконтролера. Живлення (VCC) подається від 5V Arduino, а загальна шина (GND) — до відповідного контакту на платі. Силова частина датчика (виводи IP+ та IP-) повинна бути інтегрована у фазний провід навантаження або відтворена у середовищі симуляції.

Далі підключається датчик напруги ZMPT101B. Вивід OUT приєднується до входу A0 мікроконтролера, тоді як VCC та GND — відповідно до 5V і загальної землі. Модуль вимірює змінну напругу, що подається на його силовий вхід, і видає пропорційний аналоговий сигнал.

Для виводу інформації про параметри мережі використовується LCD-дисплей з інтерфейсом I2C. Підключення здійснюється через пари контактів: SDA дисплея до A4 Arduino, SCL — до A5. Живлення дисплея (VCC) подається від 5V, а земля — до GND. I2C-інтерфейс дозволяє суттєво зменшити кількість використовуваних виводів мікроконтролера.

У системі передбачено три кнопки керування, які з'єднуються з цифровими входами D2, D3, D4. Кожна кнопка має бути підключена одним контактом до відповідного піну, а іншим — до GND. У коді Arduino для стабільної роботи застосовуються внутрішні підтягувальні резистори (режим INPUT_PULLUP).

Для управління навантаженням у схемі використовується модуль реле, керуючий сигнал якого підключається до цифрового виходу D7. Напруга живлення VCC і земля GND модуля реле відповідно приєднуються до 5V та GND мікроконтролера. Силова частина реле, що містить контакти COM і NO, підключається до ланцюга живлення навантаження. У разі спрацювання реле замикає коло та подає живлення на зовнішній пристрій.

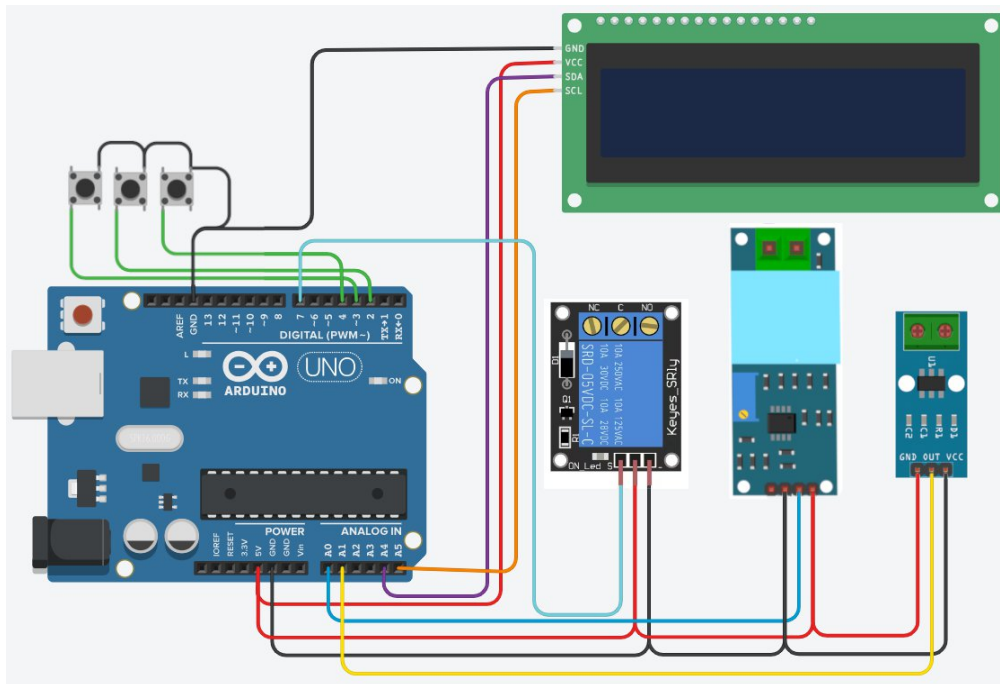


Рисунок 2.7 – Функціональна схема пристрою в Tinkercad

Після завершення підключення всіх компонентів формується функціональна схема пристрою (див. рис 2.7), яка відображає взаємозв'язок між мікроконтролером, сенсорами, дисплеєм, елементами управління та реле. Така схема слугує основою для моделювання у симуляційному середовищі та подальшої реалізації логіки керування електричним навантаженням.

На її основі розробляється також принципова електрична схема (див. рис 2.8), яка деталізує електричні з'єднання між усіма компонентами, визначає типи та номінали елементів, а також способи їх захисту. Принципова схема є технічною основою для верифікації працездатності пристрою в умовах моделювання, а також її можна використати на етапі виготовлення друкованої плати у разі подальшої фізичної реалізації пристрою.

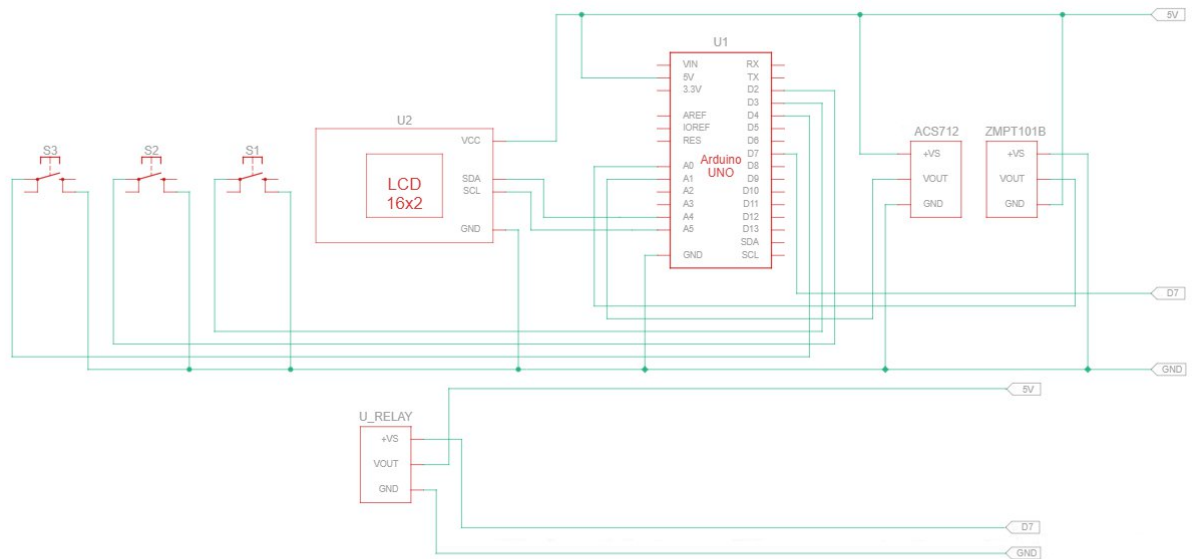


Рисунок 2.8 – Принципова схема пристрою

Для реалізації логіки системи було обрано середовище Arduino IDE (див. рис 2.9). У наведеному коді реалізовано логіку роботи пристрою для контролю енергоспоживання, який функціонує у симуляторі (наприклад, Proteus) на базі мікроконтролера Arduino (Додаток А). Основна мета цієї системи — вимірювати електричні параметри (напругу і струм), виводити їх на дисплей, а також здійснювати автоматичне керування навантаженням через реле на основі заздалегідь встановленого порогу струму.

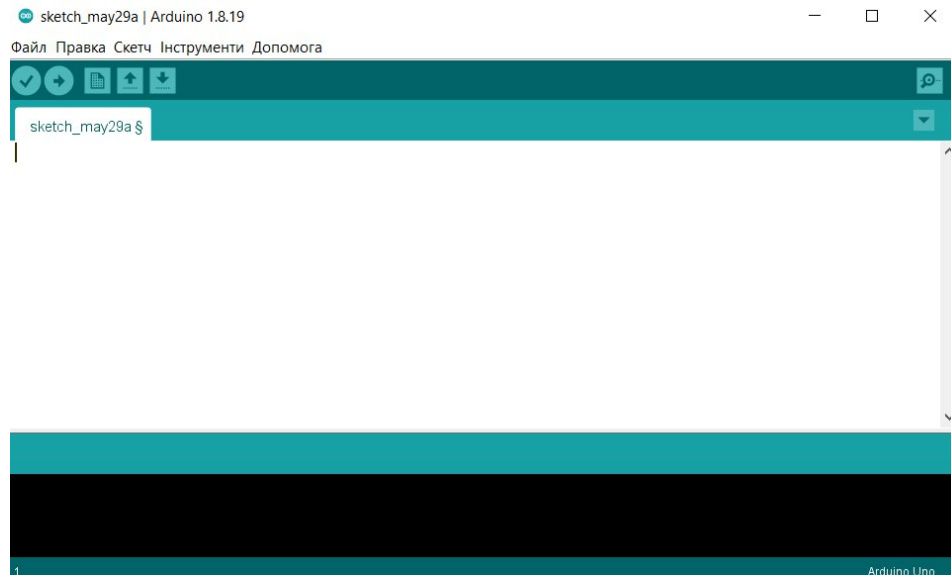


Рисунок 2.9 – Загальний вигляд програми Arduino IDE

Програма реалізує повнофункціональну систему контролю енергоспоживання, побудовану на основі мікроконтролера Arduino Uno. До складу пристрою входять сенсори напруги ZMPT101B і струму ACS712-5A, реле, LCD-дисплей з інтерфейсом I2C та три кнопки керування. Завдяки цій конфігурації пристрій виконує одночасно вимірювання, обробку, індикацію та автоматичне керування навантаженням. Основне призначення системи — забезпечити безперервний моніторинг електричних параметрів у побутовій або технічній мережі, вчасно реагувати на їх вихід за межі допустимих значень та інформувати користувача про поточний стан системи через текстові повідомлення на дисплеї. Реалізовані функції включають зчитування напруги, обчислення струму, розрахунок потужності, логіку контролю реле, інтерфейс налаштування порогів та збереження параметрів у пам'яті.

Після запуску пристрою спочатку виконується ініціалізація всіх апаратних компонентів. Функція `setup()` відповідає за налаштування дисплея, встановлення режимів пінів мікроконтролера, підготовку до роботи з кнопками та реле, а також зчитування раніше збережених налаштувань з пам'яті EEPROM. Якщо в пам'яті не виявлено коректних значень, наприклад, після першого запуску або скидання налаштувань, програма автоматично

встановлює типові параметри безпеки: мінімальну напругу 200 вольт, максимальну напругу 240 вольт та граничний струм 1 ампер. Одразу після цього виконується базова калібровка датчика струму — протягом короткого періоду вимірюються значення з входу, коли струм дорівнює нулю, після чого розраховується середнє значення. Це дозволяє врахувати внутрішній зсув сенсора (офсет) і підвищити точність усіх подальших вимірювань.

Основна логіка роботи пристрою реалізована у функції `loop()`, яка постійно виконується у циклі. Програма працює в одному з двох режимів — звичайному (режим моніторингу) або в режимі меню налаштувань. У звичайному режимі система здійснює вимірювання електричних величин. З аналогового входу A0 зчитується сигнал з сенсора ZMPT101B, який відповідає за вимірювання напруги. Програма виконує багаторазове зчитування значень за певний інтервал часу, визначає пікове значення, а потім переводить його в дійсну напругу (RMS), масштабовану відповідно до характеристик сенсора. Аналогічно, з входу A1 зчитується сигнал з ACS712, проводиться обчислення ефективного значення струму з урахуванням раніше розрахованого офсету. Ці значення використовуються для розрахунку миттєвої активної потужності, яка визначається як добуток напруги на струм. Усі три величини — напруга, струм і потужність — виводяться на екран у зручному форматі. Окрім цього, відображається стан реле: увімкнене або вимкнене.

Паралельно з вимірюваннями здійснюється логічний контроль. Програма перевіряє, чи виходить будь-яке з виміряних значень за задані межі. Якщо напруга нижча за мінімально допустиму або вища за максимально дозволена, або якщо струм перевищує встановлений ліміт, система приймає рішення вимкнути реле. Якщо всі параметри залишаються в допустимому діапазоні, реле вмикається. Щоб уникнути частого перемикання реле у випадках, коли параметри коливаються поблизу межових значень, реалізовано логіку із затримкою: стан реле змінюється лише якщо порушення триває понад

дві секунди. Такий підхід підвищує стабільність роботи системи й знижує ймовірність помилкових спрацьовувань.

Окрім автоматичного контролю, користувач має змогу вручну змінювати порогові значення. Для цього реалізовано меню налаштувань, доступ до якого відкривається шляхом утримання кнопки SET протягом однієї секунди. Після входу в меню користувач може переміщатись між пунктами за допомогою кнопок "вгору" та "вниз", змінювати значення порогів, а також скинути всі параметри до стандартних. Для підтвердження змін використовується коротке натискання SET. При цьому у режимі редагування активне значення блимає на дисплеї, що сигналізує користувачу про те, що параметр можна змінювати. Збільшення або зменшення значення відбувається натисканням відповідних кнопок. Щоб уникнути випадкових змін, вихід з меню реалізується також через довге утримання кнопки SET. У момент виходу усі змінені значення зберігаються до EEPROM, що гарантує їх збереження після вимкнення живлення або перезавантаження пристрою.

Особливу увагу приділено пункту "Reset settings?", який дозволяє повернути систему до заводських параметрів. При вході в цей пункт на екрані з'являється повідомлення з проханням утримувати кнопку SET для підтвердження скидання. Якщо користувач справді утримує кнопку довше однієї секунди, усі значення параметрів скидаються, зберігаються у пам'яті, а на дисплеї з'являється повідомлення про успішне скидання. Це дозволяє швидко виправити помилкові налаштування або підготувати систему до нового використання без потреби в перепрошивці.

Таким чином, дана програма забезпечує повноцінну роботу системи контролю енергоспоживання із широким функціоналом, інтуїтивно зрозумілим керуванням, зворотним зв'язком через дисплей, стабільним керуванням реле, гнучкими параметрами налаштування та підтримкою збереження даних. Система легко адаптується під побутові або лабораторні умови та може бути основою для більш складних рішень в енергоменеджменті.

РОЗДІЛ 3 ВІРТУАЛЬНЕ МОДЕЛЮВАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1. Реалізація моделі системи в середовищі Tinkercad

3.1 Реалізація моделі системи в середовищі Tinkercad

Для віртуальної реалізації системи контролю енергоспоживання обрано середовище Tinkercad — онлайн-симулятор для проектування, програмування та тестування мікроконтролерних систем, зокрема платформ Arduino. Основною перевагою Tinkercad є підтримка великої кількості зовнішніх модулів, можливість роботи з I2C-пристроями, зручна інтеграція коду Arduino та реалістична емуляція роботи електронної схеми.

Згідно з технічними вимогами, у системі було реалізовано такі функціональні блоки:

- Мікроконтролер Arduino UNO — обраний як центральний керуючий елемент завдяки сумісності з Tinkercad, наявності достатньої кількості входів/виходів та підтримки необхідних бібліотек.
- Дисплей 1602 з I2C-шиною — використовується для виведення поточних значень напруги, струму, повідомлень про перевищення порогів, а також меню налаштувань.
- Датчики напруги та струму — моделюються програмно, їхні значення задаються вручну або через симульовану аналогову генерацію. Для струму умовно приймається аналог виходу з ACS712, для напруги — сигнал з датчика типу ZMPT101B.
- Кнопки керування — реалізовано три кнопки: для переходу по меню, для збільшення/зменшення порогових значень.
- Електронне реле — віртуальний вихід, який активується або деактивується залежно від результатів аналізу даних датчиків.

У середовищі Tinkercad було реалізовано всі необхідні з'єднання між компонентами, включаючи живлення, цифрові та аналогові сигнали, шини

I2C, кнопки керування. Програмне забезпечення було завантажено у вигляді скетчу Arduino, який містив повну логіку обробки даних з датчиків, реалізацію меню та управління реле (див. рис 2.5). Структура програми побудована за модульним принципом, що передбачає окремі функції для ініціалізації компонентів, зчитування даних, аналізу значень та керування індикацією.

Логіка роботи системи моделювалась через завантаження скомпільованого коду у віртуальну плату Arduino. Програма забезпечує періодичне зчитування даних з датчиків через АЦП, обчислення активної потужності, а також індикацію інформації на дисплеї. Окрім цього, реалізовано функцію автоматичного вимкнення навантаження при перевищенні порогових значень, що дозволяє моделювати аварійний захист.

3.2. Реалізація моделі системи в середовищі Tinkercad

Вимірювання електричних параметрів, зокрема струму та напруги, є фундаментальним процесом для будь-якої системи контролю енергоспоживання, адже саме на основі цих величин приймаються рішення про подальші дії пристрою, зокрема увімкнення або вимкнення навантаження. У межах віртуального середовища Tinkercad, де реалізується програмне моделювання та тестування пристрою, цей процес відбувається шляхом емуляції аналогових сигналів на відповідних входах мікроконтролера Arduino. Подібне моделювання дозволяє створити умови, максимально наближені до реальної роботи пристрою в середовищі змінного електричного струму.

Для моделювання напруги у Tinkercad використовується симульований аналоговий сигнал, який подається на один із аналогових входів мікроконтролера. Цей сигнал у віртуальній моделі відповідає вихідному сигналу сенсора ZMPT101B. У реальних умовах цей сенсор виконує вимірювання змінної напруги побутової електромережі, використовуючи спеціальний трансформатор напруги. Його принцип дії базується на пропорційному зменшенні змінної напруги до безпечного рівня, який подається на аналоговий вхід Arduino для подальшої обробки. У Tinkercad ця

поведінка імітується за допомогою повзунків або змінних джерел сигналу, які користувач може регулювати вручну, задаючи необхідний рівень напруги. Це дозволяє моделювати різні ситуації, зокрема падіння напруги, її перевищення або нестабільність у мережі, і перевірити, як саме реагуватиме програма на ці зміни.

Аналогічно відбувається моделювання струму. У реальних умовах для цього використовується сенсор ACS712, який дозволяє вимірювати силу струму в провіднику, що проходить через його вхід. Сенсор видає на вихід аналоговий сигнал, який є пропорційним значенню миттєвого струму. У Tinkercad ця поведінка відтворюється аналогічно — шляхом подачі на аналоговий вхід мікроконтролера сигналу, який можна змінювати вручну або через кероване джерело, що імітує зміну навантаження. Завдяки цьому можна перевірити, як система реагує на зростання споживаного струму, перевищення порогового значення або повернення до норми після зниження навантаження.

Зчитані сигнали обробляються внутрішнім аналого-цифровим перетворювачем (ADC) мікроконтролера Arduino Nano. Цей компонент дозволяє перевести вхідну напругу з аналогового входу в цифрове значення, яке далі масштабовується у фізичні одиниці (вольти, амperi) з урахуванням технічних характеристик використовуваних сенсорів. Зокрема, для сенсора ZMPT101B використовується коефіцієнт перерахунку, який базується на амплітудному значенні сигналу та його співвідношенні до напруги мережі. Для ACS712 у розрахунках береться до уваги чутливість модуля (наприклад, 185 мВ/А для версії на 5А) та значення офсету, що визначається під час калібрування.

Після перетворення та обробки отриманих значень дані відображаються на LCD-дисплеї з інтерфейсом I2C. Виведення інформації виконується у форматі, який легко читається користувачем: у першому рядку відображаються значення напруги та струму, у другому — розрахована

потужність та стан реле. Це дає змогу в режимі реального часу бачити, які саме значення має кожен із параметрів, та оперативно реагувати на будь-які відхилення. Крім того, реалізований механізм затримки дозволяє уникнути зайвих перемикань реле при короткочасних коливаннях параметрів, що робить систему стабільною та практичною для застосування в реальних умовах.

Завдяки можливостям середовища Tinkercad розробник може протестувати поведінку системи у найрізноманітніших сценаріях, не підключаючи її до реальної електромережі. Це дозволяє швидко виявити і усунути логічні помилки, налаштувати алгоритми реагування, перевірити коректність збереження параметрів у пам'яті та оцінити зручність користування меню налаштувань. Таким чином, моделювання вимірювання струму і напруги у Tinkercad виконує не лише технічну функцію, а й виступає критично важливим інструментом під час налагодження, оптимізації та перевірки програмної частини системи контролю енергоспоживання.

У результаті моделювання було підтверджено коректність роботи пристрою в заданих умовах. Графічні результати моделювання наведено нижче (див. рис 3.1-3.3).

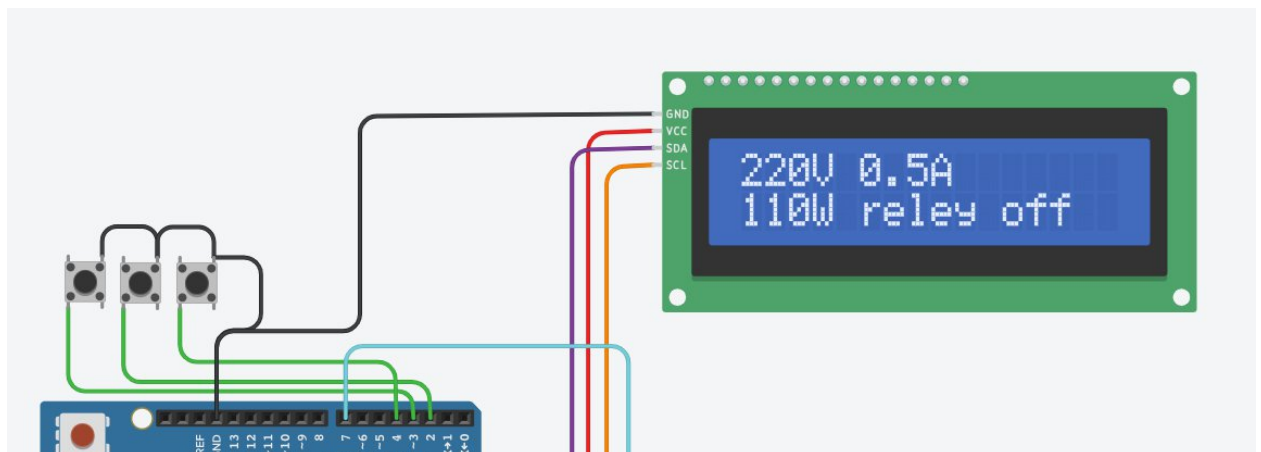


Рисунок 3.1 – Візуалізація показників струму та напруги на дисплеї

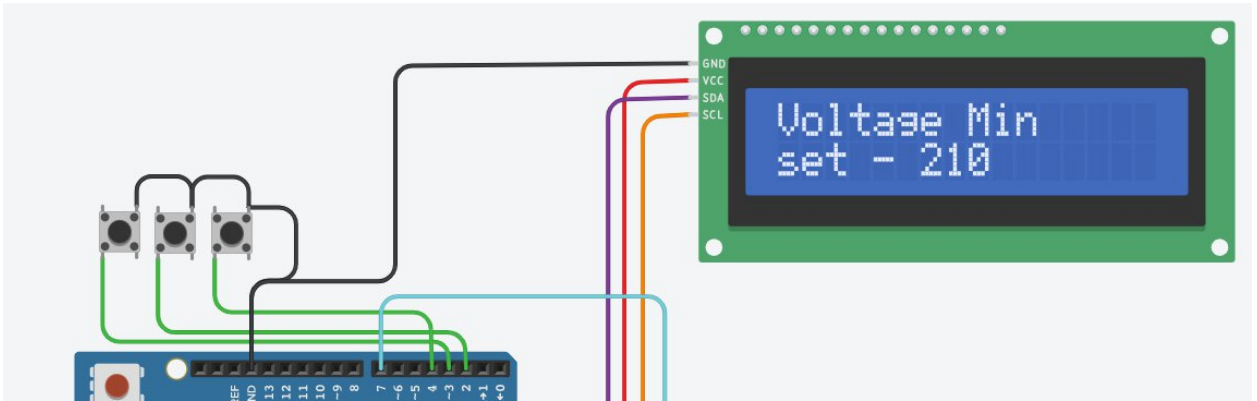


Рисунок 3.2 – Візуалізація меню налаштування напруги

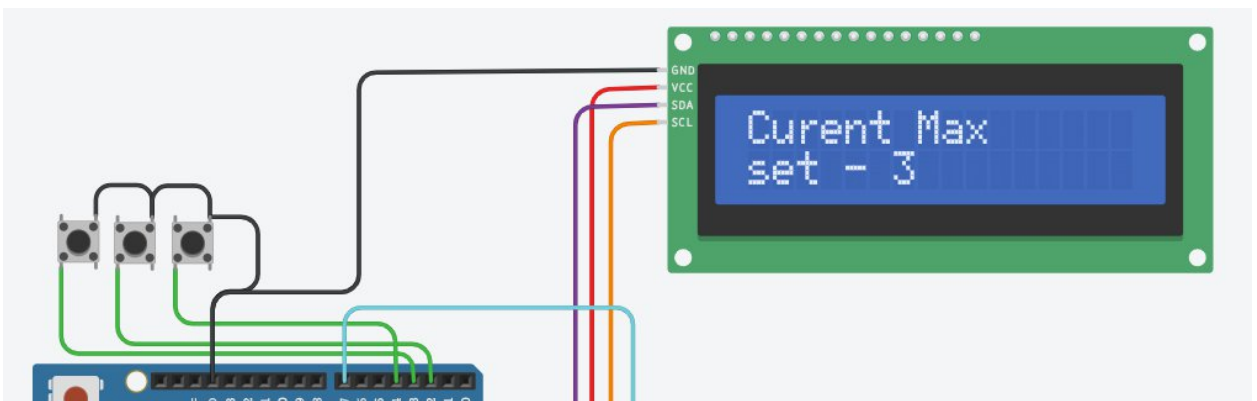


Рисунок 3.3 – Візуалізація меню налаштування струму

Однією з ключових функцій розробленої системи є автоматичне реагування на перевищення встановлених порогових значень струму або напруги, що забезпечує базовий рівень електробезпеки та захист обладнання. Для перевірки цієї функціональності в середовищі Proteus було проведено серію симуляцій із навмисною зміною вхідних параметрів.

Порогові значення було попередньо задано у прошивці Arduino, з можливістю зміни через кнопки інтерфейсу: напруга – 240 В, струм – 3 А. У ході тестування джерело живлення моделі змінювалося так, щоб зумисно викликати перевищення одного з порогів.

Тест 1: Перевищення порогу напруги (див. рис 3.4).

Напруга джерела збільшувалась до 260 В, при цьому система через датчик ZMPT101В реєструвала надлишкове значення, після чого активувала реле, що розмикало навантаження. На дисплеї з'являлося відповідне повідомлення, наприклад, «Overvoltage! Relay OFF».

Тест 2: Перевищення порогу струму (див. рис 3.5)..

Навантаження змінювалося у вигляді зменшення опору в схемі, що призводило до збільшення струму понад 3 А. Датчик ACS712 фіксував перевищення, мікроконтролер виконував аварійне відключення реле, що також супроводжувалося виведенням повідомлення на дисплей («Overcurrent!»).

Тест 3: Одночасне перевищення обох параметрів.

У цьому випадку система демонструвала коректну логіку — достатньо було перевищення одного з параметрів, щоб активувався захист. Це відповідає принципу пріоритетної реакції на будь-який фактор ризику.

Під час усіх тестів спостерігалася стабільна робота системи: затримка спрацювання не перевищувала 200–300 мс, що є прийнятним для подібних побутових задач. Скидання аварійного стану можливе через натискання кнопки або автоматично після повернення значень у нормальний діапазон (залежно від конфігурації логіки програми).

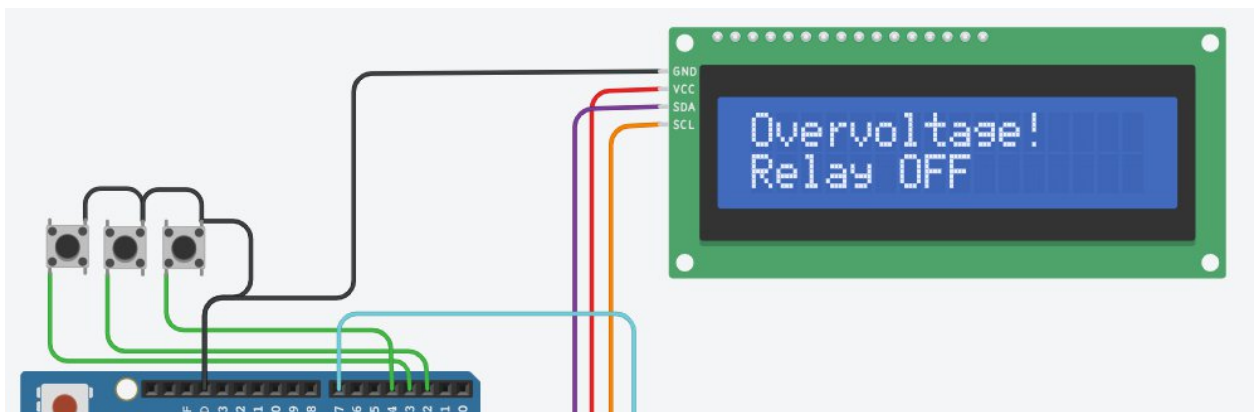


Рисунок 3.4 – Візуалізація попередження про перенапругу на дисплеї

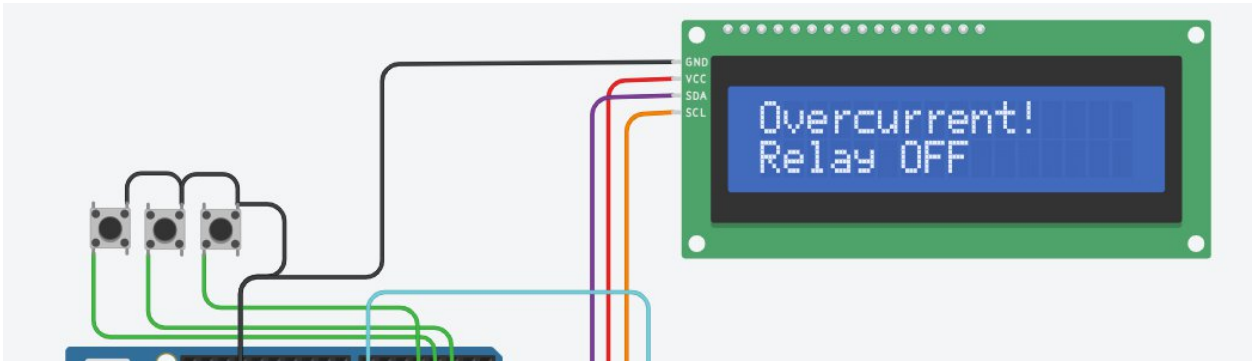


Рисунок 3.5 – Приклад спрацювання захисту при перевищенні струму

Таким чином, результати тестування підтверджують надійність реалізованого механізму захисту, що має ключове значення при використанні пристрою у побутових умовах. Це дозволяє говорити про практичну придатність системи до подальшої фізичної реалізації або розширення функціоналу.

Після завершення моделювання системи контролю енергоспоживання у середовищі Tinkercad проведено віртуальне тестування функціональних характеристик пристрою. Мета тестування полягала у верифікації коректності роботи окремих компонентів, відповідності логіки роботи заявленим вимогам та аналізі динамічної реакції системи на зміну вхідних параметрів, зокрема напруги та струму.

3.3 Аналіз результатів моделювання та перевірка відповідності вимогам

Після завершення побудови віртуальної моделі системи контролю енергоспоживання в середовищі Tinkercad було здійснено розгорнутий якісний аналіз її відповідності визначеним технічним та функціональним вимогам. Така перевірка є важливою частиною розробки системи, оскільки дозволяє своєчасно виявити можливі помилки в структурі апаратного або програмного забезпечення, оцінити взаємодію окремих компонентів, а також встановити ступінь відповідності логіки функціонування заявленим цілям і завданням, сформульованим на початку проєкту.

На етапі моделювання було відтворено повну функціональну структуру пристрою, що включала Arduino UNO, дисплей 1602 з інтерфейсом I2C, кнопки керування, віртуальне реле та аналогові входи для імітації сенсорів струму й напруги. Усі елементи були інтегровані згідно з попередньо розробленою принциповою схемою, а логіка їх взаємодії забезпечувалася відповідним програмним кодом. Аналіз починався з перевірки правильності схемотехнічного з'єднання, що включало такі аспекти:

- коректне підключення інтерфейсу дисплея до пінів SDA і SCL мікроконтролера, що забезпечувало стабільну передачу даних через шину I2C;
- призначення аналогових входів A0 та A1 для прийому даних із віртуальних датчиків ZMPT101B (напруга) та ACS712 (струм);
- підключення цифрових входів до кнопок керування, з обробкою натискань за допомогою внутрішніх підтягувальних резисторів та програмної антидребезгової фільтрації;
- використання одного з цифрових виходів перемикання реле у відповідь на перевищення заданих порогів.

Наступним етапом аналізу стала перевірка програмної логіки взаємодії компонентів. Програмний скетч, завантажений у середовище Tinkercad, містив ініціалізацію дисплея, структуру меню налаштувань, обробку сигналів з кнопок, зчитування аналогових значень та логіку ухвалення рішень щодо стану реле. Завдяки модульному принципу побудови коду, кожна функція системи могла бути ізольовано перевірена та оптимізована. Наприклад, блок обробки сигналів з датчиків містив масштабування та переведення у фізичні одиниці вимірювання, що давало змогу виводити на екран значення, наближені до реальних умов експлуатації.

Середовище Tinkercad дозволило провести детальне спостереження за поведінкою системи під час змін вхідних параметрів. При моделюванні умов

перевищення порогових значень було підтверджено, що дисплей відображає попереджувальні повідомлення, а стан цифрового виходу, що імітує реле, змінюється згідно з очікуваним алгоритмом. Крім того, виявлена стабільна робота меню дозволила підтвердити коректність навігації та збереження користувацьких налаштувань у змінних оперативної пам'яті.

Результати аналізу підтвердили, що всі апаратні компоненти віртуальної моделі взаємодіють у відповідності до заданої логіки, а програмна частина реалізує очікувану функціональність. Було досягнуто повної відповідності між структурою моделі, функціональними визначеними вимогами до системи та завданнями.

Загалом, проведене в середовищі Tinkercad моделювання та структурний аналіз дозволили переконатися у правильності архітектурного підходу, вибору елементної бази та реалізації логіки управління. Створена віртуальна система є цілісною, стабільною та готовою до масштабування, вдосконалення або практичної реалізації на фізичному рівні, що відкриває перспективи її подальшого застосування в умовах реального побутового середовища.

ВИСНОВКИ

У кваліфікаційній роботі було досліджено, розроблено та віртуально реалізовано систему контролю енергоспоживання з використанням мікроконтролерної платформи Arduino, орієнтовану на потреби побутового сектору. З огляду на актуальність проблеми енергоефективності в умовах зростаючої вартості електроенергії та потреби в енергозаощадженні, запропоноване рішення є вчасним і має практичну значущість.

У першому розділі роботи проведено огляд теоретичних аспектів проблеми контролю енергоспоживання. Було систематизовано сучасні методи вимірювання електричних параметрів, розглянуто роль мікроконтролерів у задачах енергоменеджменту, а також проаналізовано тенденції розвитку подібних систем як у промисловості, так і в побуті. Було встановлено, що більшість комерційних рішень орієнтовані на масштабні об'єкти, тоді як прості й доступні засоби контролю залишаються недостатньо поширеними серед пересічних споживачів.

У другому розділі визначено функціональні вимоги до системи, обґрунтовано вибір апаратної платформи (Arduino UNO) та необхідних периферійних модулів: датчиків напруги та струму (ZMPT101B, ACS712), екрана з інтерфейсом I²C та реле для управління навантаженням. Було розроблено логічну структуру системи, описано підключення елементів, запропоновано інтерфейс взаємодії з користувачем через дисплей та кнопки.

У третьому розділі здійснено моделювання роботи системи у середовищі Tinkercad. Продемонстровано віртуальну реалізацію пристрою, що забезпечує вимірювання струму та напруги в реальному часі, вивід даних на дисплей, сигналізацію про перевищення порогових значень та автоматичне відключення навантаження. Результати віртуального тестування підтвердили функціональність запропонованої системи. Було також реалізовано просту логіку меню налаштувань, яка дозволяє змінювати граничні параметри струму та напруги.

У процесі реалізації були досягнуті всі поставлені у вступі завдання, зокрема:

- визначено актуальні проблеми контролю енергоспоживання в побутовому секторі;
- обрано відповідні технічні засоби для побудови мікроконтролерної системи;
- розроблено структурну та функціональну схеми;
- реалізовано віртуальну модель з базовим інтерфейсом керування;
- протестовано роботу системи в умовах перевищення допустимих параметрів.

Практична цінність полягає у можливості використання розробленої моделі як основи для створення реального побутового пристрою, який зможе допомогти користувачам моніторити та регулювати власне споживання електроенергії, знижуючи витрати та підвищуючи безпеку експлуатації електроприладів.

У подальшому можлива адаптація розробки до більш складних сценаріїв, зокрема інтеграція з бездротовими мережами, хмарними платформами та реалізація мобільного застосунку для дистанційного контролю.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ефективний контроль та супровід енергоресурсів.
URL: <https://digitap.com.ua/upravlinnya-energoresursamy-cherez-systemu-avtomatyzovanogo-suprovodu-ta-kontrolyu-energiyi-askue-rushij-stalogo-rozvytku/>
2. Global data centre electricity consumption, by equipment, Base Case.
URL: <https://www.iea.org/data-and-statistics/charts/global-data-centre-electricity-consumption-by-equipment-base-case-2020-2030>
3. Що таке шунт резистор?
URL: <http://ua.alloystrip.com/news/what-is-a-shunt-resistor-5417631.html>
4. 6.1: Схеми подільника напруги - LibreTexts - Ukrayinska
URL: [https://ukrayinska.libretexts.org/Робоча_сила/Технологія_електронік_и/Книга%3A_Електричні_ланцюги_I_-_постійний_струм_\(Kirphaldt\)/06%3A_Дільникові_схеми_та_закони_Кірхгофа/6.01%3A_Схеми_подільника_напруги](https://ukrayinska.libretexts.org/Робоча_сила/Технологія_електронік_и/Книга%3A_Електричні_ланцюги_I_-_постійний_струм_(Kirphaldt)/06%3A_Дільникові_схеми_та_закони_Кірхгофа/6.01%3A_Схеми_подільника_напруги)
5. ZMPT101B Voltage Sensor Products
URL: <https://datacapturecontrol.com/articles/io-components/sensors/voltage/zmpt101b-ac-voltage-transformer-sensor-module>
6. Що таке мікропроцесор, мікроконтролер та програмований логічний контролер
URL: https://elprivod.nmu.org.ua/ua/interesting/what_is_mp_mc_plc.php
7. Arduino Nano — Arduino Official Store
URL: https://store.arduino.cc/products/arduino-nano?srsltid=AfmBOoohrRjGieHOLbF0uT6fHKHqcTPc_aMxrTEvYZXekmpF5gCowi2J
8. ESP8266.net — The Internet of Things with ESP8266
URL: <http://esp8266.net/>

9. STM32F103C8 - Mainstream Performance line, Arm Cortex-M3 MCU with 64 Kbytes of Flash memory
URL:<https://www.st.com/en/microcontrollers-microprocessors/stm32f103c8.html>
10. Pico-series Microcontrollers - Raspberry Pi Documentation
URL:<https://www.raspberrypi.com/documentation/microcontrollers/pico-series.html>
11. Реле струму | ElectroControl.com.ua
URL:<https://electrocontrol.com.ua/ua/rele/toka>
12. Реле струму AP-50A
URL:<https://digitop.ua/catalog/relestrumu/ap50a/>
13. Реле контролю струму Рубіж PKT-16/100
URL:<https://axiomplus.com.ua/ua/rele-ogranicheniya-moschnosti/product-71024/?srsltid=AfmBOop5BCWWP8znltu08yePez9JOs4nbSqoDoZ0XiCgc uNoqk4Hermh>
14. Реле контролю струму ZUBR RET I25
URL: <https://vse-e.com.ua/rele-kontrolia-toka-ret-i25>
15. Датчик струму ACS712 30A GY-712
URL: https://arduino.ua/prod618-datchik-toka-ac712-30a-gy-712?srsltid=AfmBOooqhVw_YwFFjxl4FIE8w55w6StMd5iqvLL1pQ9AWh 7bLvQaabfa
16. Acs712-datasheet.ashx
URL:<https://www.allegromicro.com/-/media/files/datasheets/acs712-datasheet.ashx>
17. LCD 1602 символний дисплей 16x2 (синій)
URL:<https://arduino.ua/prod169-lcd-1602-simvolnii-displei-16x2-sinii?srsltid=AfmBOoqX2OMMKPgKD8U1-AthLDirOHC4IvDefWNLiQHv4rA8eSSGZpr>
18. PCB Design and Circuit Simulator Software - Proteus

URL:<https://www.labcenter.com/>

19. Dashboard - Tinkercad

URL: <https://www.tinkercad.com/dashboard>

20. Arduino основи програмування

URL: https://geekmatic.in.ua/ua/arduino_osnovyi_programmirovaniya

Додатки
Додаток А
Скетч програми

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

LiquidCrystal_I2C lcd(0x27, 16, 2);

const int voltagePin = A0; // ZMPT101B
const int currentPin = A1; // ACS712-5A
const int setButtonPin = 4; // Кнопка SET
const int upButtonPin = 3; // Кнопка ввєрх
const int downButtonPin = 2; // Кнопка вниз
const int relayPin = 7;

int currentOffset = 512;

// Порогові значєння
float voltageMin;
float voltageMax;
float currentMax;

// EEPROM адреси
const int addrVoltageMin = 0;
const int addrVoltageMax = sizeof(float);
const int addrCurrentMax = 2 * sizeof(float);

// Затримка для реле
```

```
unsigned long relayTimer = 0;
bool relayState = false;
bool relayShouldBeOn = true;

// МЕНЮ
int menuState = 0;
bool inMenu = false;
bool editMode = false;
unsigned long setPressStart = 0;
bool setHoldTriggered = false;
bool waitForSetRelease = false;
bool ignoreNextRelease = false;

unsigned long blinkTimer = 0;
bool blinkState = true;

void loadSettings() {
    EEPROM.get(addrVoltageMin, voltageMin);
    EEPROM.get(addrVoltageMax, voltageMax);
    EEPROM.get(addrCurrentMax, currentMax);
    if (isnan(voltageMin)) voltageMin = 200;
    if (isnan(voltageMax)) voltageMax = 240;
    if (isnan(currentMax)) currentMax = 1.0;
}

void saveSettings() {
    EEPROM.put(addrVoltageMin, voltageMin);
    EEPROM.put(addrVoltageMax, voltageMax);
    EEPROM.put(addrCurrentMax, currentMax);
}
```

```
}
```

```
void setup() {  
  pinMode(setButtonPin, INPUT_PULLUP);  
  pinMode(upButtonPin, INPUT_PULLUP);  
  pinMode(downButtonPin, INPUT_PULLUP);  
  pinMode(relayPin, OUTPUT);  
  digitalWrite(relayPin, LOW);
```

```
  lcd.init();  
  lcd.backlight();  
  analogReference(DEFAULT);
```

```
  loadSettings();
```

```
  long sum = 0;  
  for (int i = 0; i < 500; i++) {  
    sum += analogRead(currentPin);  
    delay(2);  
  }  
  currentOffset = sum / 500;  
}
```

```
float readACVoltage(int pin) {  
  float maxV = 0;  
  unsigned long start = millis();  
  while (millis() - start < 100) {  
    int val = analogRead(pin);  
    float voltage = (val - 512) * (5.0 / 1023.0);
```

```

    if (abs(voltage) > maxV) maxV = abs(voltage);
}
return maxV * sqrt(2) * 220.0 / 2.5;
}

float readACCurrent(int pin) {
    float sum = 0;
    int samples = 200;
    for (int i = 0; i < samples; i++) {
        int val = analogRead(pin);
        float voltage = (val - currentOffset) * (5.0 / 1023.0);
        sum += voltage * voltage;
    }
    float rms = sqrt(sum / samples);
    return rms / 0.185;
}

void showMenu() {
    lcd.clear();
    switch (menuState) {
        case 1:
            lcd.setCursor(0, 0);
            lcd.print("Voltage Min");
            lcd.setCursor(0, 1);
            lcd.print("set - ");
            if (!(editMode && !blinkState)) lcd.print(voltageMin);
            break;
        case 2:
            lcd.setCursor(0, 0);

```

```

    lcd.print("Voltage Max");
    lcd.setCursor(0, 1);
    lcd.print("set - ");
    if (!(editMode && !blinkState)) lcd.print(voltageMax);
    break;
case 3:
    lcd.setCursor(0, 0);
    lcd.print("Current Max");
    lcd.setCursor(0, 1);
    lcd.print("set - ");
    if (!(editMode && !blinkState)) lcd.print(currentMax);
    break;
case 4:
    lcd.setCursor(0, 0);
    lcd.print("Reset settings?");
    lcd.setCursor(0, 1);
    lcd.print("Hold SET to yes");
    break;
}
}
// end of showMenu()

void loop() {
    static bool lastSetState = HIGH;
    static bool lastUpState = HIGH;
    static bool lastDownState = HIGH;

    bool setState = digitalRead(setButtonPin);
    bool upState = digitalRead(upButtonPin);

```

```
bool downState = digitalRead(downButtonPin);

if (millis() - blinkTimer > 600) {
  blinkTimer = millis();
  blinkState = !blinkState;
  if (inMenu && editMode) showMenu();
}

if (setState == LOW && !setHoldTriggered) {
  if (setPressStart == 0) setPressStart = millis();
  if (millis() - setPressStart >= 1000) {
    if (inMenu && menuState == 4 && editMode) {
      voltageMin = 200;
      voltageMax = 240;
      currentMax = 1.0;
      saveSettings();
      editMode = false;
      lcd.clear();
      lcd.setCursor(0, 0);
      lcd.print("Reset done");
      delay(1000);
      showMenu();
    } else if (!inMenu) {
      inMenu = true;
      editMode = false;
      if (menuState == 0) menuState = 1;
      showMenu();
    } else {
      inMenu = false;
    }
  }
}
```

```

    editMode = false;
    menuState = 0;
    lcd.clear();
    saveSettings();
}
setHoldTriggered = true;
waitForSetRelease = true;
ignoreNextRelease = true;
}
}

if (setState == HIGH) {
    setPressStart = 0;
    setHoldTriggered = false;
    if (waitForSetRelease) {
        waitForSetRelease = false;
    }
}

if (lastSetState == LOW && setState == HIGH) {
    if (ignoreNextRelease) {
        ignoreNextRelease = false;
    }
}

else if (!waitForSetRelease && inMenu && menuState != 4) {
    if (menuState == 4 && !editMode) {
        editMode = true;
    } else if (menuState == 4 && editMode) {
        // Підтвердження скидання

```

```

    voltageMin = 200;
    voltageMax = 240;
    currentMax = 1.0;
    saveSettings();
    editMode = false;
} else {
    editMode = !editMode;
}
showMenu();
}
}
lastSetState = setState;

if (inMenu) {
    if (editMode) {
        if (lastUpState == HIGH && upState == LOW) {
            switch (menuState) {
                case 1: voltageMin += 1; break;
                case 2: voltageMax += 1; break;
                case 3: currentMax += 0.1; break;
            }
            showMenu();
            delay(200);
        }
        if (lastDownState == HIGH && downState == LOW) {
            switch (menuState) {
                case 1: voltageMin -= 1; break;
                case 2: voltageMax -= 1; break;
                case 3: currentMax -= 0.1; break;
            }
        }
    }
}

```

```
    }  
    showMenu();  
    delay(200);  
  }  
} else {  
  if (lastUpState == HIGH && upState == LOW) {  
    menuState--;  
    if (menuState < 1) menuState = 4;  
    showMenu();  
    delay(200);  
  }  
  if (lastDownState == HIGH && downState == LOW) {  
    menuState++;  
    if (menuState > 4) menuState = 1;  
    showMenu();  
    delay(200);  
  }  
}  
lastUpState = upState;  
lastDownState = downState;  
return;  
}  
  
float voltage = readACVoltage(voltagePin);  
float current = readACCurrent(currentPin);  
float power = voltage * current;  
  
if (voltage < voltageMin || voltage > voltageMax || current > currentMax) {  
  relayShouldBeOn = false;
```

```
} else {  
    relayShouldBeOn = true;  
}  
  
if (relayShouldBeOn != relayState) {  
    if (millis() - relayTimer > 2000) {  
        relayState = relayShouldBeOn;  
        digitalWrite(relayPin, relayState ? HIGH : LOW);  
        relayTimer = millis();  
    }  
} else {  
    relayTimer = millis();  
}  
  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print(voltage, 0);  
lcd.print("V ");  
lcd.print(current, 1);  
lcd.print("A");  
  
lcd.setCursor(0, 1);  
lcd.print(power, 0);  
lcd.print("W ");  
lcd.print("reley ");  
lcd.print(relayState ? "on " : "off");  
  
delay(1000);  
}
```