

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень

Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi
Назва теми

КВРКІ.190243.43.19.02 ПЗ
Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія та програмування»

Назва

Виконав: студент IV курсу, група KI2-19-2

Підпис

А.М. Саїд

Ініціали, прізвище

Керівник

Підпис, дата

А.О. Нічепорук

Ініціали, прізвище

Нормоконтролер

Підпис, дата

С.М. Лисенко

Ініціали, прізвище

До захисту допускаю:

Зав. кафедри комп'ютерної інженерії та інформаційних систем

Підпис

Т.О. Говорушенко

Ініціали, прізвище

« 1 » червня 2023 р.

Хмельницький 2023

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорушенко

“ 11 ” 01 2023 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Саїду Артуру Музафаровичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

Керівник проекту (роботи) Нічепорук А.О., доцент кафедри КІС

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 01.03.2023 р. № 5

2. Строк подання студентом проекту (роботи) на кафедру 07.06.2023 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____

Аналіз відомих засобів та рішень із функцією безконтактного вимірювання температури

Програмна реалізація програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

Апаратна реалізація програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi



5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень) _____

Креслення сценаріїв серверної частини у Node-red

Схема електрична принципова

Монтажна схема

6. Консультанти розділів дипломного проекту (роботи)

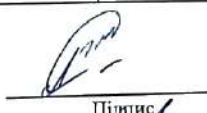
Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Лисенко С.М., професор кафедри КПС		
Антиплагіат	Нічепорук А.О., доцент кафедри КПС		

7. Дата видачі завдання « 11 » 01 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	11.01.2023	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2023	виконано
3	Робота над розділом 1 – Аналіз відомих засобів та рішень	01.03.2023	виконано
4	Робота над розділом 2 – Проектування апаратної частини програмно-технічного засобу	01.04.2023	виконано
5	Робота над розділом 3 – Проектування та реалізація програмної частини програмно-технічного засобу	30.04.2023	виконано
6	Оформлення пояснювальної записки згідно вимог	20.05.2023	виконано
7	Попередній захист ВКР	26.05.2023	виконано
8	Захист ВКР на засіданні ЕК	Червень 2023 року	

Студент


Підпис

Саїд А.М.

Ініціали, прізвище

Керівник проекту (роботи)


Підпис

Нічепорук А.О.

Ініціали, прізвище

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi».

Автор роботи: *Саїд Артур Музафаровч.*

Керівник роботи: *Нічепорук Андрій Олександрович.*

Пояснювальна записка: *56 с., 38 рис., 4 табл., 3 дод., 60 джерел.*

Графічна частина: *7 презентаційних слайдів.*

БЕЗКОНТАКТНЕ ВИМІРЮВАННЯ ТЕМПЕРАТУРИ, ЛОГУВАННЯ, СЦЕНАРІЇ АВТОМАТИЗАЦІЇ.

Мета кваліфікаційної роботи: є розробка програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi.

Тема програмно-технічного засобу безконтактного вимірювання температури з функцією логування на базі Raspberry Pi, яка включає контроль доступу користувачів за показами їх температури, є особливо актуальною сьогодні, особливо в контексті боротьби з пандемією коронавірусу COVID-19. У цьому випадку, до пристрою може бути додана RFID мітка, яку користувач зчитує. Після зчитування мітки, засіб автоматично вимірює температуру користувача без контакту. Якщо температура перевищує норму (наприклад, встановлену медичними стандартами для виявлення підозри на зараження COVID-19), то доступ користувачу може бути заборонений, наприклад, через відмову відкриття дверей або активацію аудіо-візуальних сигналів.

Підпис студента



Дата 30.05.2023

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	4
ВСТУП.....	5
1 АНАЛІЗ ВІДОМИХ ЗАСОБІВ ТА РІШЕНЬ ІЗ ФУНКЦІЄЮ БЕЗКОНТАКТНОГО ВИМІРЮВАННЯ ТЕМПЕРАТУРИ.....	7
1.1 Принципи організації ІЧ-термометрії	7
1.2 Організація ІЧ-детекторів	9
1.3 Аналіз відомих програмно-технічних засобів із функцією безконтактного вимірювання температури	12
1.5 Постановка задачі	17
2 АПАРАТНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ БЕЗКОНТАКТНОГО ВИМІРЮВАННЯ ТЕМПЕРАТУРИ ІЗ ФУНКЦІЄЮ ЛОГУВАННЯ НА БАЗІ RASPBERRY PI.....	18
2.1 Визначення вимог до програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi	18
2.2 Структура програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi	19
2.3 Схема електрична принципова	22
2.4 Монтажна схема проектованого програмно-технічного пристрою.....	24
2.5 Аналіз обраних рішень.....	26
2.5.1 Аналіз обраних апаратних рішень	27
2.5.2 Аналіз обраних програмних рішень	35
Висновки до розділу 2.....	36
3 ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ БЕЗКОНТАКТНОГО ВИМІРЮВАННЯ ТЕМПЕРАТУРИ ІЗ ФУНКЦІЄЮ ЛОГУВАННЯ НА БАЗІ RASPBERRY PI.....	37
3.1 Програмна реалізація функції безконтактного вимірювання температури	

КВРКІ. 190243.43.19.02 ПЗ				
Зм.	Арк.	№докум.	Підпис	Дата
Виконав		Саїд А.М.		
Перевір.		Нічепорук А.О.		
Н.контр.		Лисенко С. М.		
Затвер.		Говорущенко		31.09
Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi				
		Літера	Аркуш	Аркушів
		2	62	62
ХНУ, КІ2-19-2				

із функцією логування	37
3.1.1 Програмна реалізація клієнтської частини	38
3.1.2 Програмна реалізація серверної частини (сценарію у Node red)	48
Висновки за розділом 3	60
ВИСНОВКИ	60
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ	63
ДОДАТОК А Копія креслення «Креслення сценаріїв пожежної та охоронної сигналізації з фотофіксацією у Node-red»	63
ДОДАТОК Б Копія креслення «Схема електрична принципова»	70
ДОДАТОК В Копія креслення «Монтажна схема»	71

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

АЦП – Аналого-цифровий перетворювач

БВТ – Безконтактне вимірювання температури

ІТ – Інформаційна технологія

КС – Комп'ютерна система

КФС – Кібер-фізична система

ПЗ – Програмне забезпечення

ЦАП – Цифро-аналоговий перетворювач

MQTT – Message queuing telemetry transport

RPi – Raspbery Pi

					КВРКІ. 190243.43.19.02 ПЗ	Арк.
						4
Зм.	Арк.	№докум.	Підпис	Дата		

ВСТУП

Тема програмно-технічного засобу безконтактного вимірювання температури з функцією логування на базі Raspberry Pi, яка включає контроль доступу користувачів за показами їх температури, є особливо актуальною сьогодні, особливо в контексті боротьби з пандемією коронавірусу COVID-19. У цьому випадку, до пристрою може бути додана RFID мітка, яку користувач зчитує. Після зчитування мітки, засіб автоматично вимірює температуру користувача без контакту. Якщо температура перевищує норму (наприклад, встановлену медичними стандартами для виявлення підозри на зараження COVID-19), то доступ користувачу може бути заборонений, наприклад, через відмову відкриття дверей або активацію аудіо-візуальних сигналів.

Важливою функцією програмно-технічного засобу безконтактного вимірювання температури є логування. Ця функція дозволяє зберігати і аналізувати дані про вимірювану температуру з різних часових точок. Одним із головних аспектів вимірювання температури в контексті COVID-19 є відстеження можливих випадків підвищеної температури, яка може бути одним з симптомів захворювання. Логування показників температури дозволяє створити історичний запис про кожне вимірювання та зберігати цю інформацію для подальшого аналізу. Аналіз історичних даних температурних вимірів може допомогти виявити тренди та виявити випадки, коли температура підвищується або коли є аномальні зміни. Це може бути корисно для виявлення підозрілих ситуацій та прийняття відповідних заходів безпеки, наприклад, вимоги додаткових перевірок, заборони доступу або сповіщення відповідних служб.

Логування також забезпечує можливість відстеження діяльності користувачів та їхніх показників температури в разі виникнення необхідності перевірки або розслідування. Це може бути корисно для трасування контактів у випадку підозри на інфекцію та вжиття заходів для забезпечення безпеки інших осіб.

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 5
Зм.	Арк.	№докум.	Підпис	Дата		

Збирання і аналіз даних про температурні виміри також може допомогти в наукових дослідженнях та епідеміологічних дослідженнях. Збір широкомасштабних даних про температуру може допомогти у визначенні тенденцій, прогнозуванні поширення вірусу та розробці ефективних стратегій контролю захворюваннями.

Таким чином, такий програмно-технічний засіб може бути використаний у місцях збору людей, наприклад, у великих організаціях, медичних закладах, аеропортах, громадських приміщеннях тощо, де контроль температури та обмеження доступу можуть допомогти запобігти поширенню інфекційних захворювань, таких як COVID-19, а також бути використаний як основа для збору даних для наукових досліджень чи аналітики.

Метою роботи є розробка програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi.

Об'єктом дослідження є процеси реалізації безконтактного вимірювання температури.

Предметом дослідження є програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi.

					КВРКІ. 190243.43.19.02 ПЗ	Арк.
						6
Зм.	Арк.	№докум.	Підпис	Дата		

1 АНАЛІЗ ВІДОМИХ ЗАСОБІВ ТА РІШЕНЬ ІЗ ФУНКЦІЄЮ БЕЗКОНТАКТНОГО ВИМІРЮВАННЯ ТЕМПЕРАТУРИ

1.1 Принципи організації ІЧ-термометрії

Теплота передається від одного тіла до іншого шляхом провідності, конвекції або випромінювання. Випромінювання – це процес, при якому теплова енергія у вигляді електромагнітних хвиль випромінюється гарячим об'єктом і поглинається більш холодним об'єктом. Більшість цього випромінювання знаходиться в інфрачервоній (ІЧ) області електромагнітного спектру, але частина також поширюється у смузі видимого світла. Смуга довжин хвиль ІЧ-випромінювання простягається від 0,7 до 1000 мікрон, однак практичні системи вимірювання ІЧ-випромінювання використовують лише певні смуги довжин хвиль від 0,7 до 14 мікрон, оскільки випромінювання в цьому діапазоні найсильніше.

Якщо об'єкт піддається впливу ІЧ-енергії, що випромінюється джерелом тепла, таким як електричний обігрівач, лампочка, сонце чи інші джерела, енергія, що досягає об'єкта, називається падаючою енергією. Частина цієї енергії відбивається від поверхні предмета. Теоретично, коефіцієнт відбиття об'єкта може змінюватися від 0 (без відбиття) до 1,0 (100% відбиття). Шорсткі, матові поверхні мають низьку відбивну здатність. Поліровані та глянцеві поверхні, особливо метали, мають високу відбивну здатність.

Залежно від матеріалу об'єкта, товщини та довжини хвилі випромінювання частина випромінювання може проходити через об'єкт або передаватися. Коефіцієнт пропускання може варіюватися від 0 (енергія не передається через об'єкт) до 1,0 (100% енергії передається через об'єкт). Приклади високого коефіцієнта пропускання включають скло, кварц, пластикову плівку та різні гази. Матеріали, непрозорі в ІЧ-спектрі, мають близькі до нуля коефіцієнти пропускання.

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 7
Зм.	Арк.	№докум.	Підпис	Дата		

Решта енергії поглинається об'єктом і підвищує його температуру. Гіпотетичне тіло, яке не має відбиття або пропускання і поглинає всю падаючу енергію по всьому спектру, має коефіцієнт поглинання, що дорівнює 1,0, і називається чорним тілом. Реальні об'єкти, які називаються сірими тілами, мають коефіцієнти поглинання від 0 до 1,0.

Сумарна енергія, W_I , визначається як:

$$W_I = W_R + W_T + W_A \quad (1.1)$$

де:

W_I – падаюча енергія, отримана об'єктом, Вт

W_R – енергія, відбита від поверхні об'єкта, Вт

W_T – енергія, передана об'єктом, Вт

W_A – енергія, поглинена об'єктом, Вт

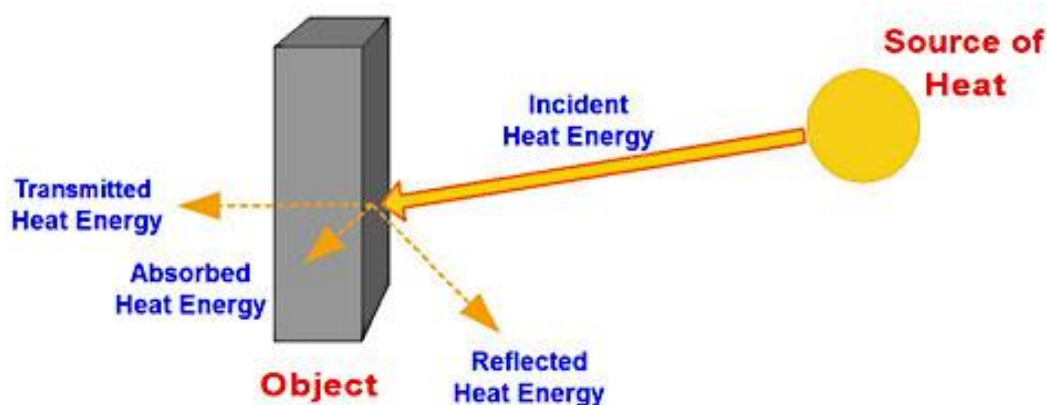


Рисунок 1.1 – Температура та ІЧ випромінювання

Оскільки об'єкт поглинає енергію та нагрівається, він також випромінює енергію. Коли об'єкт перебуває в стані теплової рівноваги, кількість енергії, яку він поглинає (W_A), дорівнює кількості енергії, яку він випромінює (W_E): $W_A = W_E$. Коли об'єкт поглинає більше енергії і його температура підвищується, кількість випромінювання, яке він випускає, також збільшується.

ІЧ-термометрія заснована на тому факті, що будь-яка речовина (тверда, рідка або газоподібна), яка має температуру вище абсолютного нуля (0оК або -

273оС), випромінює енергію випромінювання. Це явище описує закон Стефана-Больцмана.

1.2 Організація ІЧ-детекторів

Більшість ІЧ-детекторів є або однохвильовими (також називають одноколірними), або двоххвильовими (також називають двоколірними) типу. Однохвильові детектори вимірюють ІЧ-енергію в межах певного діапазону довжин хвиль, а прилад обчислює температуру об'єкта на основі вихідного сигналу детектора та попередньо встановленого коефіцієнта випромінювання. Деякі термометри мають регульований коефіцієнт випромінювання, а більшість простих приладів мають фіксований коефіцієнт випромінювання.

Двоххвильові детектори вимірюють енергію на двох різних діапазонах довжин хвиль, а прилад обчислює температуру на основі співвідношення двох показань. Якщо коефіцієнт випромінювання або енергія змінюється на однакову величину в обох діапазонах, то це не впливає на точність вимірювання. Коефіцієнт випромінювання або кількість випромінюваної енергії може змінюватися через зміну чи рух об'єкта, забруднення чи зміщення об'єктива або перешкоди огляду. Недоліком двоххвильового детектора є висока вартість і менша точність за певних умов.

Коефіцієнт випромінювання багатьох матеріалів і поверхонь залишається відносно постійним у діапазоні довжин хвиль ІЧ-випромінювання, тому вимірювання енергії в будь-якому більш вузькому діапазоні буде прийнятним. Інші матеріали мають смуги довжин хвиль із вищим і нижчим коефіцієнтом випромінювання через високу відбивну здатність або пропускання та потребують вузькосмугових детекторів, налаштованих на довжини хвиль із високим коефіцієнтом випромінювання.

Типовий інфрачервоний датчик температури складається з оптичних компонентів, ІЧ-детектора, електроніки та вихідного каскаду дисплея або інтерфейсу. Оптика фокусує ІЧ-енергію на детекторі, який перетворює ІЧ-

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 9
Зм.	Арк.	№докум.	Підпис	Дата		

енергію в електричний сигнал. Після підсилення, лінеаризації та стабілізації температури електричний сигнал перетворюється на значення, що представляє виміряну температуру (рис. 1.2).

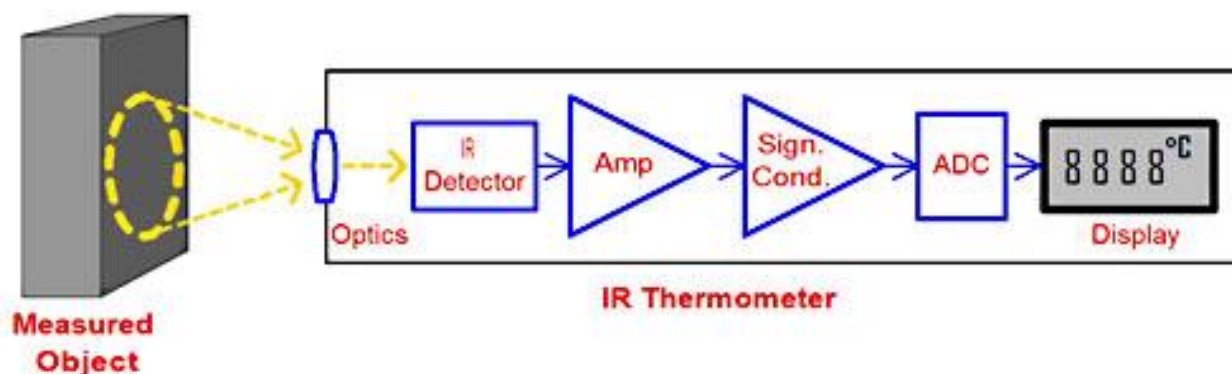


Рисунок 1.2 – Типова структура ІЧ-детектора

За принципом дії ІЧ-детектори поділяються на дві категорії: теплові детектори та фотодетектори (фотодіоди). Теплові ІЧ-детектори поглинають падаючу енергію, підвищують температуру чутливого елемента і змінюють електричні властивості детектора: термобатарей генерують термоелектричну напругу, болометри змінюють опір, а піроелектричні пристрої змінюють свою поляризацію. Загалом вони повільніші за фотодетектори.

Термобатарея виготовляється шляхом послідовного з'єднання декількох термопар і контакту їх гарячих спаїв з чорним тілом, яке поглинає падаючу ІЧ-енергію та нагріває гарячі спаї. Холодні спаї розміщені в зоні детектора з відповідним тепловідводом. Ці детектори мають швидку реакцію, широку смугу, великий динамічний діапазон і часто використовуються в термометрах загального призначення, автомобілях, кондиціонерах і людському тілі.

У болометрах (це прилад для вимірювання енергії або потужності електромагнітного випромінювання, зокрема інфрачервоного (ІЧ) випромінювання) використовується пластина матеріалу, яка змінює свій опір у відповідь на зміну температури. Схема перетворює зміну опору в зміну напруги, яка далі обробляється приладом. Болометри часто використовуються

для вимірювання ІЧ-енергії низького рівня, часто як пристосування до телескопа.

Піроелектричні прилади стають електрично зарядженими при зміні температури тіла. Щоб отримати придатний сигнал, падаюча інфрачервона енергія має «пульсувати». Вихідний амплітудний сигнал змінного струму пропорційний енергії імпульсу. Оскільки енергія, що випромінюється вимірюваними об'єктами, зазвичай постійна, термометри, які використовують піроелектричні детектори, мають механічний або оптичний подрібнювач перед датчиком. Ці датчики використовуються в багатьох системах домашньої безпеки.

Фотодетектори побудовані на кремнієвій підкладці з ІЧ-чутливою зоною, яка вивільняє вільні електрони під час впливу фотонів. Потік електронів створює електричні сигнали, пропорційні енергії падіння. Ці детектори часто використовуються як масиви в тепловізійних системах.

Детектор потребує захисту від навколишнього середовища, а вибраний матеріал вікна має пропускати правильний діапазон довжин хвиль із мінімальним загасанням. Для довгохвильових детекторів найкраще підходить вікно з сульфїду цинку або германію, для короткохвильових детекторів — скло, а для середньохвильового спектру — кварц. Деякі прилади використовують волоконно-оптичний світловод для направлення випромінювання на детектор.

Оскільки всі типи ІЧ-детекторів виробляють сигнали в діапазоні мікрровольт, слідом за детектором повинен бути підсилювач з високим коефіцієнтом посилення. Криві залежності потужності сповіщувача від температури не є лінійними та сильно коливаються зі зміною температури навколишнього середовища. Щоб усунути це, схема формування сигналу стабілізує температуру та лінеаризує сигнал. Для багатьох програм потрібен аналого-цифровий перетворювач (АЦП) для перетворення показників температури в цифровий формат.

Ручні та багато інших типів приладів мають вбудований дисплей, тоді як інші пристрої підключаються до комп'ютера, системи збору даних або системи

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 11
Зм.	Арк.	№докум.	Підпис	Дата		

контролю температури через кабель RS232 або RS-485. Деякі прилади імітують вихід термопари, інші мають контур струму 0–20 мА або 4–20 мА або вихід напруги.

1.3 Аналіз відомих програмно-технічних засобів із функцією безконтактного вимірювання температури

На сьогоднішній день існує досить велика кількість програмно-технічних засобів для безконтактного вимірювання температури. Зокрема можна виділити:

– Термометри на основі інфрачервоних сенсорів: Ці пристрої вимірюють теплове випромінювання з об'єкта і перетворюють його на температурні дані. Вони часто використовуються у промисловості та медицині.

– Термальні камери: Ці камери застосовуються для вимірювання температури на великих відстанях та в реальному часі. Вони широко використовуються для контролю температури в промислових об'єктах, будівлях та інших місцях.

– Смартфони з функцією вимірювання температури: Деякі смартфони мають вбудовані сенсори, які можуть вимірювати температуру повітря або об'єктів. Для цього можуть використовуватись спеціальні додатки.

– Безконтактні термометри для споживачів: Ці пристрої, як правило, мають форму пістолета або термометра з лазерним покажчиком. Вони широко використовуються в домашніх умовах для вимірювання температури тіла, їжі або інших поверхонь.

Важливо зазначити, що точність цих засобів може варіюватись, і їхнє використання повинно відбуватися з урахуванням вимог специфічної ситуації або за допомогою професіоналів, якщо потрібна висока точність вимірювання.

Безконтактні термометри для споживачів працюють на основі різних принципів. Основні види принципів роботи таких термометрів включають:

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 12
Зм.	Арк.	№докум.	Підпис	Дата		

– Інфрачервоні термометри: Ці термометри вимірюють теплове випромінювання з об'єкта. Вони використовуються для вимірювання температури шкіри, предметів, їжі тощо. Інфрачервоний промінь, випромінюваний об'єктом, сприймається сенсором, і на основі отриманих даних обчислюється температура.

– Термопарні термометри: Вони використовують спарену термопару для вимірювання температури. При зміні температури в точці контакту термопари змінюється напруга, що може бути виміряна і перетворена на температуру.

Що стосується характеристик безконтактних термометрів для споживачів, основні аспекти, на які слід звернути увагу, включають:

– Діапазон вимірювання: Визначає мінімальну та максимальну температуру, яку може виміряти термометр.

– Точність: Вказує, наскільки точно термометр вимірює температуру. Точність може відрізнятися залежно від моделі і вимог вимірювання.

– Час вимірювання: Вказує, як швидко термометр може здійснити вимірювання температури. Деякі моделі мають швидкий відгук, що особливо важливо для медичних додатків.

– Функціональність: Деякі термометри можуть мати додаткові функції, такі як пам'ять для збереження результатів, режими вимірювання, автоматичне вимірювання, тощо.

Розглянемо детальніше приклади безконтактних термометрів.

Одним із прикладів приладу для безконтактного вимірювання температури є термометр ArhiMED EcoTherm ST 300 (рис. 1.3). Основними характеристиками цього засобу є:

- Діапазон вимірювань: від 0 до 100°.
- Похибка вимірювання: 0,1° С.
- Температурна шкала: Цельсій та Фаренгейт.
- Кількість слотів пам'яті: 32.
- Область застосування: людина, тварини, поверхні.
- Час вимірювання: миттєво (менше 1 сек).

– Додаткові функції: автоматичне вимкнення, монохромна підсвітка, специфікація DC 3V.



Рисунок 1.3 – Безконтактний термометр ArhiMED EcoTherm ST

Ще одним приладом для безконтактного вимірювання температури є пірметр GM333 від Venetech (рис. 1.4). Головною перевагою даного пристрою є застосування кольорового інверсійного екрана, що забезпечує значене заощадження енергії й зручність експлуатації. Основними характеристиками GM333 є:

- Діапазон вимірювання: $-50\text{ }^{\circ}\text{C}$ – $400\text{ }^{\circ}\text{C}$.
- Дискретність: $0.1\text{ }^{\circ}\text{C}$ або $0.1\text{ }^{\circ}\text{F}$.
- Похибка: 1.5%.
- Час відгуку: 500 мс.
- Спектральний діапазон: 5-14 мкм.

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ. 190243.43.19.02 ПЗ

Арк.
14

- Коефіцієнт випромінювання (EMS): 0.95.
- Лазерний показчик: однопроменевий.
- Робоча температура: від 0 до +40 °С.
- Робоча вологість: 10-95%.
- Габарити: 153x100x43 мм.



Рисунок 1.4 – Venetech GM333

Іншим прикладом пристрою для безконтактного вимірювання температури є безконтактний інфрачервоний пірометр WT319B. Завдяки діапазону вимірювання від -50 ° до 600 ° С ця модель знайде своє застосування в різних галузях життєдіяльності, а саме: на машинобудівних та промислових підприємствах, для контролю температури зберігання на продуктових та овочевих складах, виявлення перегріву електрообладнання на підприємствах енергетичного комплексу, визначення витоків тепла на об'єктах ЖКП.

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ. 190243.43.19.02 ПЗ

Арк.
15



Рисунок 1.5 – Безконтактний інфрачервоний пірометр WT319B

Основними характеристиками WINTACT WT319B є:

- Роздільна здатність 0,1°C.
- Точність виміру $\pm 1,5\%$.
- Оптичний дозвіл 12:1.
- Постійний коефіцієнт EMS 0.95.
- Вибір одиниць виміру °C/°F.
- Можливість безперервного вимірювання температури.
- Рідкокристалічний екран з підсвічуванням.
- Лазерний покажчик для зручнішого прицілювання.
- Швидкість виміру 0.5 с.
- Утримання результатів виміру HOLD.

Розглянуті вище програмно-технічні засоби із функцією безконтактного вимірювання температури володіють досить високими показниками точності вимірювання, мають високий ступінь мобільності (є можливість виміряти температуру, взявши пристрій до рук), проте основним недоліком даних

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ. 190243.43.19.02 ПЗ

Арк.
16

пристроїв є те, що вони не зберігають дані про користувачів, для яких здійснено вимірювання температури (незважаючи на те, що температура зберігається, не має можливості асоціювати збережену температуру із користувачем). Таким чином розглянуті пристрої не можуть бути використанні для автоматичної системи контролю доступу за рівнем температури тіла, що є досить суттєвим недоліком.

1.5 Постановка задачі

Тема програмно-технічного засобу безконтактного вимірювання температури з функцією логування на базі Raspberry Pi, яка включає контроль доступу користувачів за показами їх температури, є особливо актуальною сьогодні, особливо в контексті боротьби з пандемією коронавірусу COVID-19. Тому для проектування даного програмно-технічного засобу слід виконати наступну послідовність кроків:

1. проаналізувати відомі засоби та системи безконтактного вимірювання температури, виявити їх недоліки та переваги;
2. скласти набір функційних та нефункційних вимог до програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi;
3. провести вибір апаратних та програмних засобів для проектування програмно-технічного засобу;
4. виконати проектування структури програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi;
5. виконати проектування схеми електричної принципової, а також монтажної схеми програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi;
6. здійснити програмну реалізацію функцій програмно-технічного засобу.

					КвРКІ. 190243.43.19.02 ПЗ	Арк. 17
Зм.	Арк.	№докум.	Підпис	Дата		

2 АПАРАТНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ БЕЗКОНТАКТНОГО ВИМІРЮВАННЯ ТЕМПЕРАТУРИ ІЗ ФУНКЦІЄЮ ЛОГУВАННЯ НА БАЗІ RASPBERRY PI

2.1 Визначення вимог до програмно-технічного засобу безконтактного вимірювання температури із функцією логуювання на базі Rasperry Pi

Визначення функціональних вимог є одним із перших етапів у процесі проектування будь-якої системи чи пристрою, оскільки ці вимоги визначають, які конкретні функції повинні бути реалізовані у цій системі. Чітко сформульовані вимоги допомагають покращити ефективність та якість процесу розробки системи.

Проектований програмно-технічний засіб безконтактного вимірювання температури із функцією логуювання на базі Rasperry Pi повинен виконувати наступні функції:

- можливість зчитування RFID мітки (у вигляді картки або брелка);
- перевірка присутності считаної RFID мітки у базі доступних RFID міток (для усунення можливості вимірювань не авторизованим користувачем);
- безконтактне вимірювання температури людини по натисканню на кнопку;
- перевірка та опрацювання виміряних значень температури (встановлення та перевірка із заданим пороговим значенням);
- індикація виміряного значення температури на LCD дисплеї;
- індикація дозволу для користувача (на основі виміряного значення температури) на LCD дисплеї та світло діодах;
- збереження даних про кожне вимірювання температури користувача у лог-файлі.

Також, виділимо набір нефункційних вимог, що будуть описувати такі характеристики програмно-технічного засобу як зручність використання,

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 18
Зм.	Арк.	№докум.	Підпис	Дата		

масштабованість, надійність, тощо. Зокрема проєктований програмно-технічний засіб повинен:

- Бути компактним, мати не великі габаритні розміри.
- Володіти простим та зручним інтерфейсом, з інтуїтивно зрозумілим порядком роботи із пристроєм.
- Мати не високу ціну.
- Бути придатним до масштабованості та розширення функцій.
- Придатним до оновлення.
- Володіти досить високою швидкістю обробки даних.
- Бути автономним та можливість під'єднуватись до зовнішніх джерел живлення.

Таким чином, було визначено набір функційних та нефункційних вимог, що дозволяють зрозуміти функції та характеристики, які повинен мати проєктований програмно-технічний засіб безконтактного вимірювання температури.

2.2 Структура програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

З метою реалізації визначених функційних та нефункційних вимог до програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi запропоновано структуру цього засобу, структурну схему якого представлено на рис. 2.1.

Представлений програмно-технічний засіб безконтактного вимірювання температури складається із таких програмних та апаратних складових:

- Одноплатна КС Raspberry Pi. Основним призначенням даної КС є прийняття рішення про отримане значення температури людини, тобто визначення чи перевищує воно задане порогове значення. Ще однією функцією, яка покладається на Raspberry Pi є перевірка отриманої RFID мітки у списку відомих (наперед визначених користувачів пристрою). Окрім того Raspberry Pi використовується як середовище у якому розгорнуто інструмент розробки

сценаріїв Node-RED та брокер mosquitto. Одноплатна КС Raspberry Pi з'єднується з ESP32 через бездротовий зв'язок Wi-Fi по протоколу MQTT, що функціонує за принципом «підписка-публікація». У системі задіяно чотири топіки:

- esp32/UID – публікація RFID мітки.
- esp32/temperature – публікація даних про температуру людини.
- esp32/PresentUI – публікація повідомлень, що визначають чи є задана RFID мітка у списку відомих міток користувачів.
- esp32/GrantedDenited – публікація повідомлень, що визначають, чи має доступ заданий користувач.

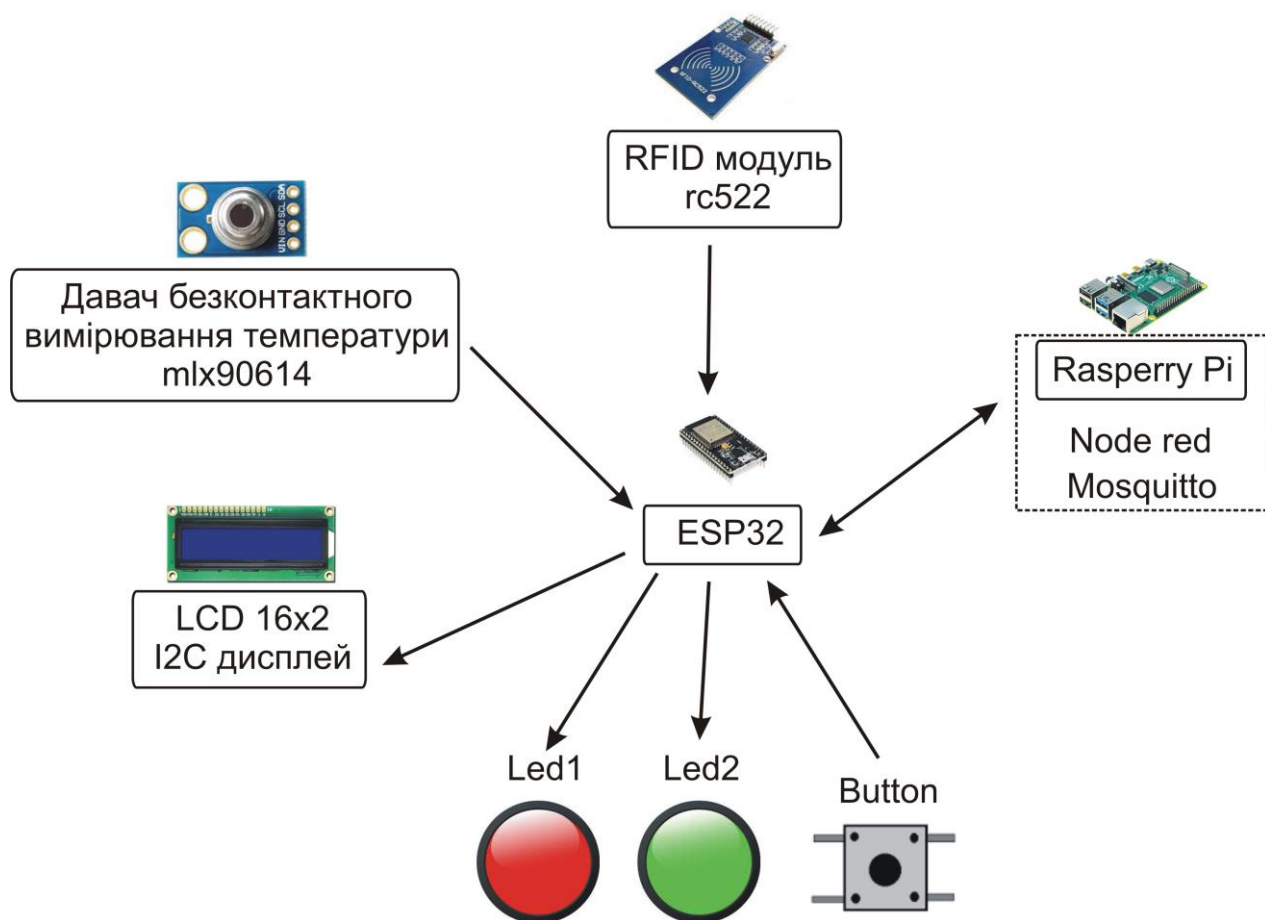


Рисунок 2.1 – Структурна схема програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

Зм.	Арк.	№докум.	Підпис	Дата

– Давач безконтактного вимірювання температури. Основне призначення вимірювання температури на відстані. Отримані дані будуть публікуватись у топик esp32/temperature. буде використаний для вимірювання температури у приміщенні. Публікує дані у топик home/temperature. Тригером, за яким здійснюється безконтактне вимірювання температури є натискання користувачем на кнопку;

– RFID Модуль. Основне призначення – зчитування мітки RFID із карти або брелка. Даний модуль під'єднується до системи на кристалі ESP32 через SPI інтерфейс. Отриманні дані у вигляді RFID мітки публікуються у топик esp32/UID;

– LCD I2C дисплей. Основне призначення – інформування користувача про виміряну температуру тіла, а також статус дозволу (Granted або Denied). Даний LCD дисплей з'єднаний із ESP32 через послідовний інтерфейс I2C. Інформація про статус дозволу отримується із топика esp32/GrantedDenied, на який підписаний ESP32;

– Мікроконтролерна плата ESP32. Основне призначення – керування набором датчиків та модулів, що під'єднуються за допомогою провідників. Комунікація ESP32 із Raspberry Pi реалізується через безпроводний зв'язок. Безпроводний характер обміну даними між Raspberry Pi та ESP32 дозволяє організувати передачу даних віддалено у межах локальної безпроводної мережі.

– Node red середовище розробки. Основне призначення – перевірка отриманої температури із граничним значенням, а також перевірку отриманої RFID мітки із списком відомих міток користувачів. Дане середовище розгортається на одноплатній комп'ютерній системі Raspberry Pi.

Таким чином означені компоненти та взаємозв'язки між ними в представленій структурі програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi дозволяють виконати реалізацію поставлених вимоги до проєктованого програмно-технічного засобу.

2.3 Схема електрична принципова

Електрична принципова схема – це графічне зображення, яке відображає взаємодію та зв'язки між основними елементами електричного пристрою. Вона допомагає зрозуміти принцип роботи пристрою та показує, як елементи підключені між собою та до джерела живлення. Схема передає інформацію про функціональні блоки пристрою та дозволяє встановити правильну послідовність елементів для оптимізації його роботи.

Проектований програмно-технічний пристрій безконтактного вимірювання температури із функцією легування на базі Raspberry Pi складається із таких компонентів як: одноплатна комп'ютерна системи Raspberry Pi, мікроконтролерна плата ESP32, давача безконтактного вимірювання температури mlx90614, RFID модуля rc522, LCD I2C дисплею із мікросхемою LCM1602, двох світлодіодів та кнопки. Давач безконтактного вимірювання температури mlx90614 та LCD дисплей під'єднуються до мікроконтролерної плати ESP32 через послідовний інтерфейс I2C, де лінії SDA та SCL підтягнуті до живлення за допомогою двох резисторів номіналом 1 кОм (на схемі позначено R3 та R4). RFID модуль під'єднується через SPI інтерфейс.

Два світлодіоди під'єднуються до мікроконтролерної плати ESP2 через гасячі резистори номіналом 220 Ом (R1 та R2). Для підключення кнопки S1 також було використано резистор номіналом 220 Ом. Для усунення явища «брязкотіння контактів» задіяно програмні механізми на основі використання таймерів, для опитуванн стану перемикача. Живлення та заземлення всіх компонентів програмно-технічного засобу організовано через загальну шину живлення та шину заземлення відповідно.

Схема електрична принципова проектованого програмно-технічного засобу безконтактного вимірювання температури на базі Raspberry Pi наведено на рис. 2.2.

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 22
Зм.	Арк.	№докум.	Підпис	Дата		

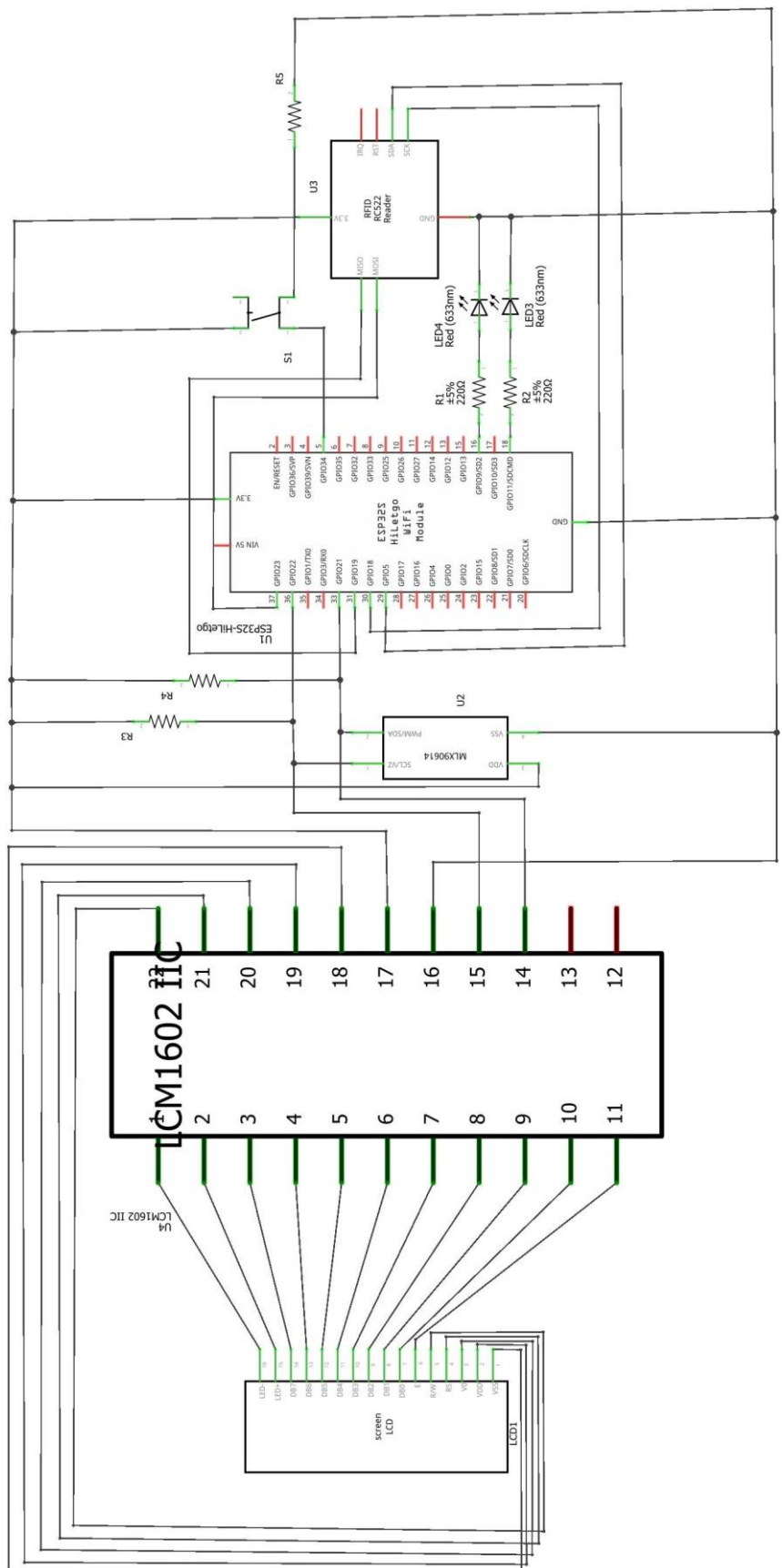


Рисунок 2.2 – Схема електрична принципова проектованого програмно-технічного засобу безконтактного вимірювання температури

Зм.	Арк.	№докум.	Підпис	Дата
-----	------	---------	--------	------

КВРКІ. 190243.43.19.02 ПЗ

2.4 Монтажна схема проектованого програмно-технічного пристрою

Для створення монтажної плати була використана система автоматизованого проектування (САПР) Fritzing. Ця САПР надає зручний графічний інтерфейс, що дозволяє створювати монтажні плати та електричні схеми. Fritzing має широку бібліотеку, яка включає в себе велику кількість компонентів, в тому числі популярні електронні компоненти, наприклад резистори, конденсатори, мікроконтролери та інші. Це спрощує процес проектування та дозволяє використовувати стандартні компоненти для швидкого створення схем та монтажних плат.

Роботу із створення монтажної плати розпочнемо із перенесення всіх необхідних компонентів на робочу область. Окрім того, планується включити макетну плату, яка буде використовуватись для створення загальної шини живлення та шини заземлення. На ці шини будуть підключені відповідні піни всіх датчиків. Давач безконтактного вимірювання температури mlx90614 під'єднується до ESP32 через I2C інтерфейс. Таким чином пін SCL mlx90614 під'єднується до GPIO 22, а пін SDA – до GPIO 21. Оскільки LCD дисплей із мікросхемою LCM1602 також під'єднуються до I2C, то піни SDA та SCL також під'єднуються до тих самих GPIO22 та GPIO21 відповідно. RFID модуль під'єднується до ESP32 через SPI інтерфейс, таким чином, пін SPI_SS rc522 з'єднується із піном ESP32 GPIO5, SPI_SCK – із GPIO18, SPI_MOSI – із GPIO23, SPI_MISO – із GPIO19, RST – із GPIO26, відповідно. Два світлодіоди під'єднуються до пінів GPIO9 та GPIO13, а кнопка – до GPIO32.

Монтажну схему проектованого програмно-технічного засобу безконтактного вимірювання температури на базі Raspberry Pi наведено на рис. 2.3.

					КВРКІ. 190243.43.19.02 ПЗ	Арк.
						24
Зм.	Арк.	№докум.	Підпис	Дата		

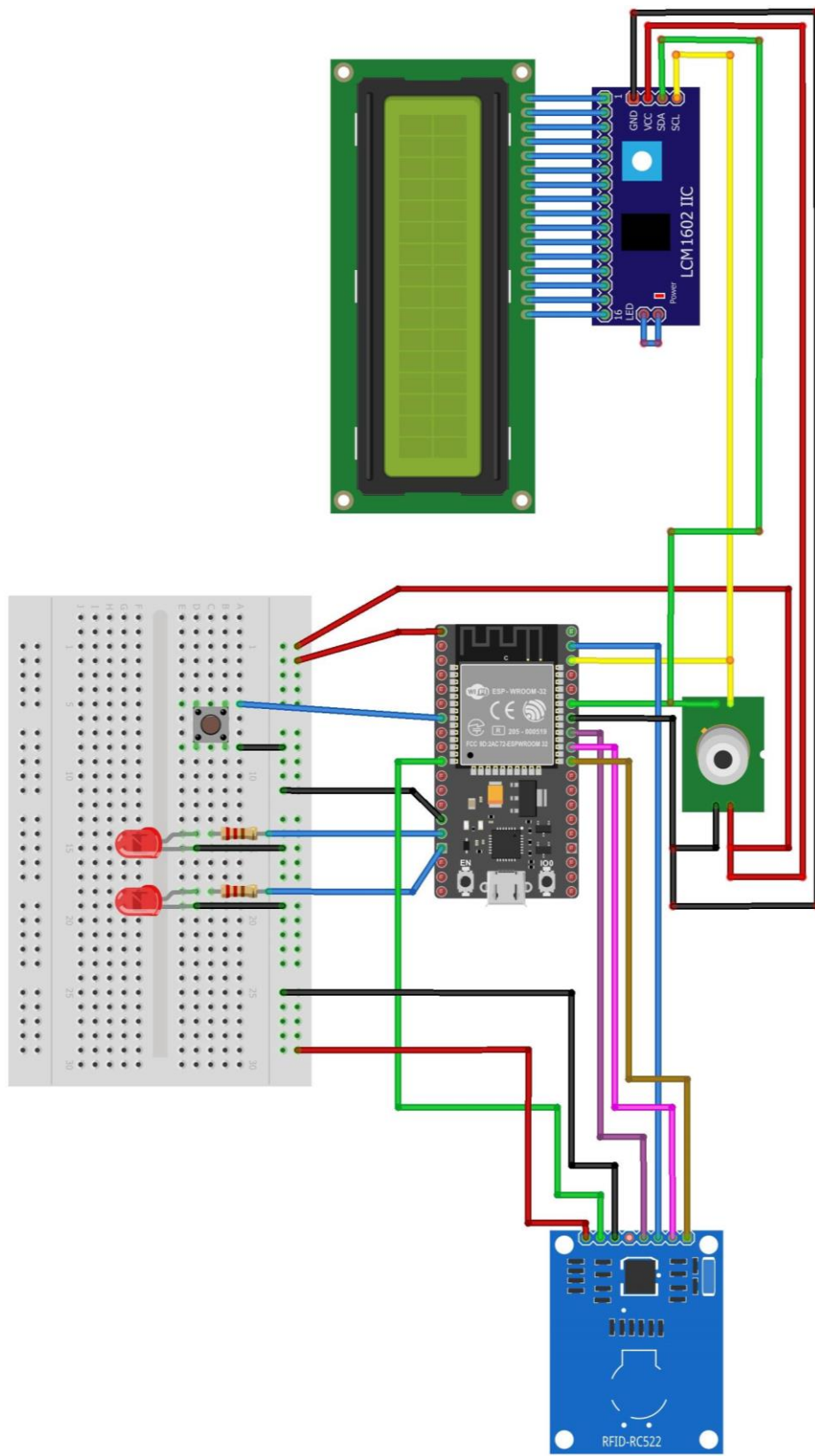


Рисунок 2.3 – Монтажна схема проектованого програмно-технічного засобу безконтактного вимірювання температури на базі Raspberry Pi

Зм.	Арк.	№докум.	Підпис	Дата

Також було спроектовано зовнішній вигляд проектованого програмно-технічного пристрою безконтактного вимірювання температури із функцією логування на базі Raspberry PI (рис. 2.4).

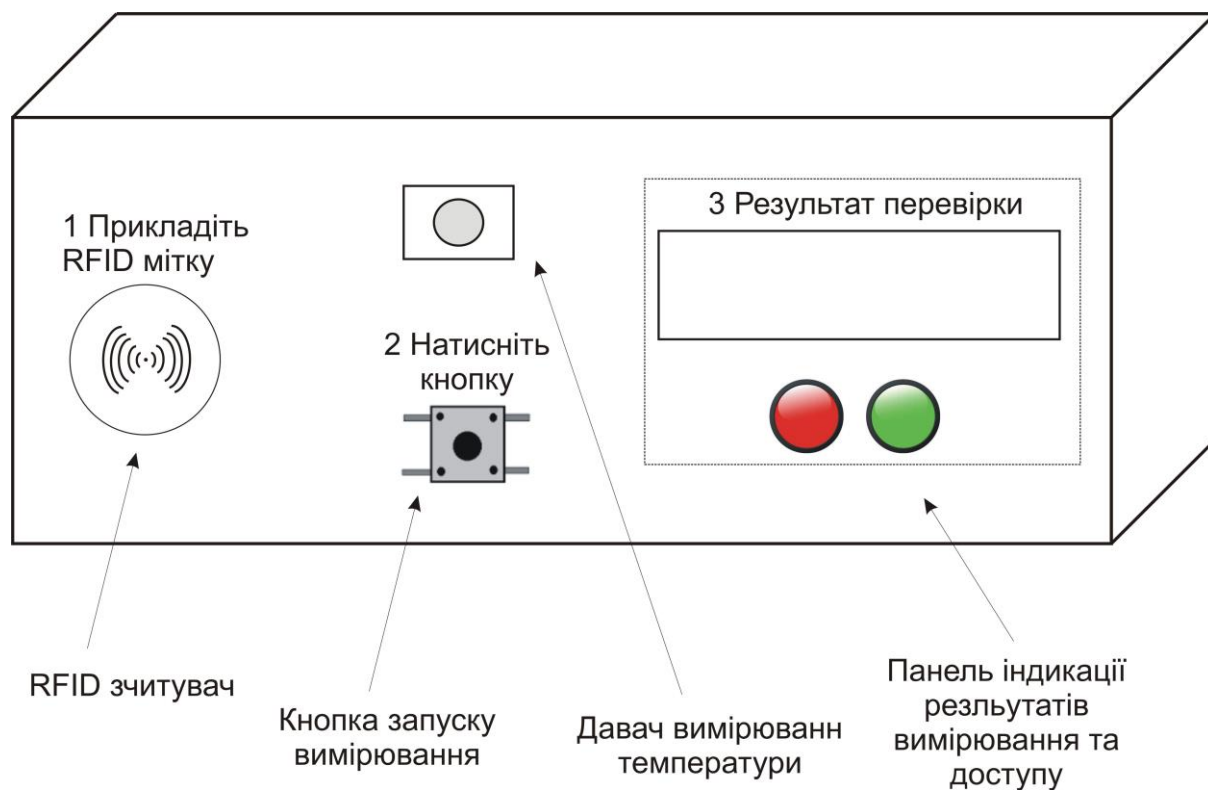


Рисунок 2.4 – Зовнішній вигляд проектованого програмно-технічного пристрою безконтактного вимірювання температури із функцією логування на базі Raspberry PI

2.5 Аналіз обраних рішень

Після завершення проектування структури та визначення принципів функціонування безконтактного пристрою вимірювання температури з функцією логування на базі одноплатної комп'ютерної системи Raspberry Pi, наступним кроком є відбір та оцінка апаратного та програмного забезпечення, які будуть використовуватись у запропонованій системі.

Зм.	Арк.	№докум.	Підпис	Дата

2.5.1 Аналіз обраних апаратних рішень

До складу запропонованого програмно-технічного пристрою безконтактного вимірювання температури із функцією логування входять наступні апаратні компоненти: комп'ютерна система Raspberry Pi, мікроконтролерна система ESP32, RFID модуль rc522, давач безконтактного вимірювання температури mlx90614, LCD I2C дисплей із мікросхемою LCM1602, два світлодіоди та одна кнопка. Розглянемо детальніше вибрані апаратні компоненти.

Модуль MLX90614 (рис. 2.5) є безконтактним вимірювачем температури, який призначений для точного вимірювання абсолютної температури об'єктів. Він забезпечує дані про температуру через цифровий інтерфейс, такий як SMBus (аналогічний I2C), або через ШІМ-вихід з вибором частоти – 10Гц або 1000Гц. Модуль MLX90614 має широкі можливості для програмування, калібрування та налаштування. Крім того, модуль має зручні отвори для кріплення, внутрішній стабілізатор, фільтруючі конденсатори для живлення та підтягуючі резистори на цифровій шині.

Дачач MLX90614 може вимірювати температуру об'єкта без фізичного контакту з ним. Це стало можливим завдяки закону під назвою Стефана-Больцмана, який стверджує, що всі об'єкти та живі істоти випромінюють ІЧ-енергію, а інтенсивність цієї випромінюваної ІЧ-енергії буде прямо пропорційна температурі цього об'єкта чи живої істоти. Тож датчик MLX90614 обчислює температуру об'єкта, вимірюючи кількість ІЧ-енергії, що випромінюється від нього. Датчик температури MLX90614 виробляється компанією Melexis. Датчик відкалібровано на заводі, тому він діє як сенсорний модуль «підключай і працюй» для прискорення процесів розробки. MLX90614 складається з двох пристроїв, вбудованих як єдиний датчик, один пристрій діє як чутливий блок, а інший пристрій діє як блок обробки. Сенсорний блок – інфрачервоний термобатарейний детектор під назвою MLX81101, який визначає температуру, а блок обробки – це єдиний кондиціонуючий ASSP під назвою MLX90302, який перетворює сигнал

від датчика в цифрове значення та обмінюється даними за допомогою протоколу I2C. MLX90302 має підсилювач з низьким рівнем шуму, 17-розрядний АЦП і DSP, який допомагає датчику мати високу точність і роздільну здатність.

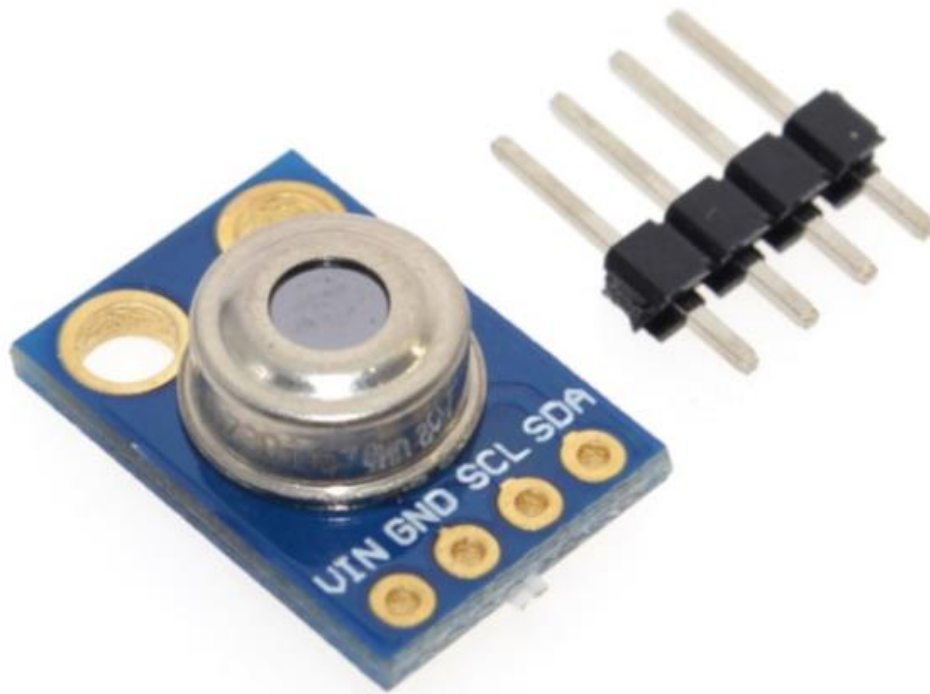


Рисунок 2.5 – Давач безконтактного вимірювання температури mlx90614

Таблиця 2.1 – Призначення контактів давача безконтактного вимірювання температури mlx90614

Вивід		Опис
1	VIN	Живлення
2	GND	GND
3	SDA	Передача даних по I2C
4	SCL	Тактуючий сигнал по I2C

Характеристиками давача безконтактного вимірювання температури mlx90614 є:

Тип давача: MLX90614ESF-BAА.

Температурний діапазон: від -40 до + 125 °С.

Калібрування датчика: заводське.

Діапазон вимірювань температури: від -70 до +380 °С.

Напруга живлення: 2.7 - 3.6 В.

Додатковий цифровий вихід: вихід ШІМ, що настраюється, для безперервного читання.

Цифровий інтерфейс: SMBus сумісний.

Точність (похибка) вимірювання: 0.5 °С для діапазону від 0 до + 50 °С.

Кут поля зору: ~ 90 °.

Роздільна здатність: 0.02 °С.

Розміри: 1.6 см x 1.1 см.

В якості RFID модуля використано RC522 (рис. 2.6), із частотою 13,56 МГц, що заснований на контролері MFRC522 від NXP semiconductors. Модуль може підтримувати такі інтерфейси як I2C, SPI та UART і зазвичай постачається з RFID-карткою та брелоком. Він зазвичай використовується в системах контролю відвідуваності та інших програмах ідентифікації осіб/об'єктів.

RC522 – це радіочастотний модуль, який складається із зчитувача RFID, картки RFID і брелока. Модуль працює на частоті 13,56 МГц, що є промисловим (ISM) діапазоном, і тому його можна використовувати без проблем з ліцензією. Модуль зазвичай працює при напрузі 3,3 В і, отже, зазвичай використовується в конструкціях 3,3 В. Брелок має 1 Кб пам'яті, яку можна використовувати для зберігання унікальних даних. Модуль зчитування RC522 може як читати, так і записувати дані в ці елементи пам'яті. Зчитувач може зчитувати дані лише з пасивних тегів, які працюють на частоті 13,56 МГц.

RC522 має робочу напругу від 2,5 В до 3,3 В і, отже, зазвичай живиться від 3,3 В і повинен використовуватися з лініями зв'язку 3,3 В. Але комунікаційні контакти цього модуля стійкі до 5 В, тому його можна використовувати з мікроконтролерами 5 В, такими як Arduino, без будь-якого додаткового обладнання. Модуль підтримує зв'язок SPI, I2C і UART, але серед них інтерфейс

SPI є найбільш часто використовуваним, оскільки він є швидким із максимальною швидкістю передачі даних 10 Мбіт/с.

Оскільки в програмі більшу частину часу модуль читання чекатиме, поки тег наблизиться, rc522 можна перевести в режим вимкнення живлення, щоб заощадити енергію в додатках, що працюють від акумулятора. Цього можна досягти за допомогою контакту IRQ на модулі. Мінімальний струм, споживаний модулем у режимі вимкнення, становитиме лише 10 мкА. Модуль можна легко використовувати з Arduino завдяки його легкодоступній бібліотеці RC522 RFID Arduino.

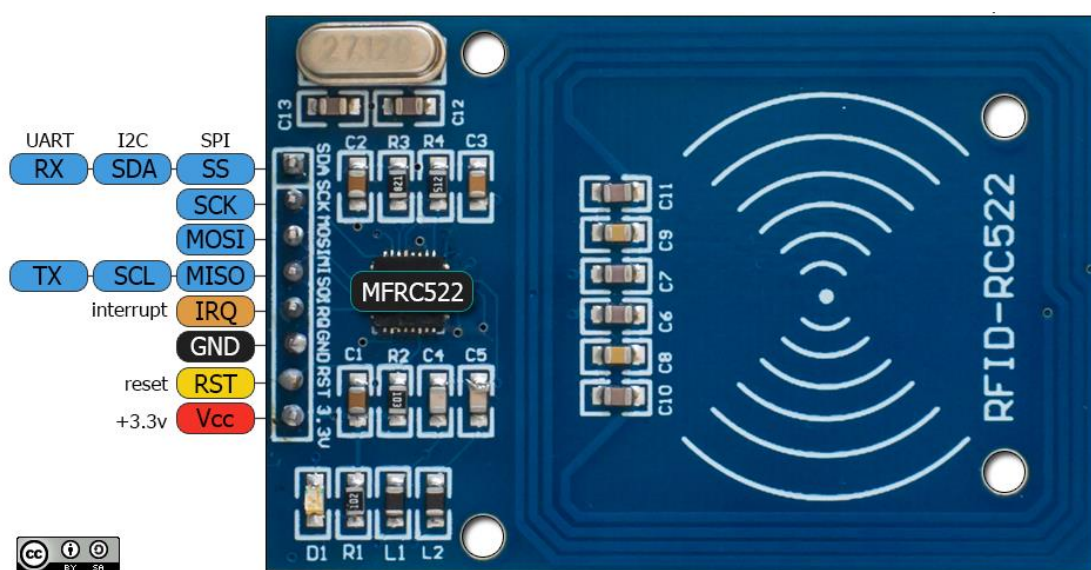


Рисунок 2.6 – RFID модуль rc522

Таблиця 2.2 – Призначення контактів RFID модуля rc522

Вивід		Опис
1	Vcc	Живлення
2	GND	GND
3	RST	Скидання
4	IRQ	Переривання
5	MISO/SCL/Tx	MISO пін при SPI комунікації, SCL для I2c і Tx при UART.

6	MOSI	Master або slave для піна для комунікації по SPI
7	SCK	Serial Clock pin, використовується для забезпечення джерела синхронізації
8	SS/SDA/Rx	Serial input (SS) для SPI комунікації, SDA для ІІС і Rx при UART

Характеристиками модуля gc522 є:

- Частота: 3,56 МГц.
- Робоча напруга: від 2,5 до 3,3 В.
- Зв'язок: SPI, протокол I2C, UART.
- Максимальна швидкість передачі даних: 10 Мбіт/с.
- Діапазон читання: 5 см.
- Споживаний струм: 13-26 мА.
- Споживання в режимі вимкнення: 10 мкА (хв.).

Для відображення інформації про зчитану температуру та статус дозволу в проєктованому програмно-технічному пристрою використано LCD дисплей (рис. 2.7).



Рисунок 2.7 – Контакти LCD дисплею

Символьний дисплей 1602 LCD модуль з можливістю відображення двох рядків по 16 символів в кожній (таблиця 2.3). Дисплей 16x2 управляється контролером HD44780. Колір вбудованої підсвітки - жовто-зелений. Колір символів - чорний (символи видні і без підсвітки). Плата сумісна зі стандартною бібліотекою LiquidCrystal під Arduino. Призначення контактів LCD модуля наведено у таблиці 2.3.

Таблиця 2.3 – Призначення контактів LCD модуля

Вивід		Опис
1	Vss (Ground)	Контакт заземлення підключений до заземлення системи
2	Vdd (+5 Volt)	Живить РК-дисплей за допомогою +5 В (4,7 В – 5,3 В)
3	VE	Визначає рівень контрастності дисплея. Заземлений для отримання максимального контрасту.
4	Register Select	Підключений до мікроконтролера для перемикавання між регістром команд/даних
5	Read/Write	Використовується для читання або запису даних. За замовчуванням заземлений для запису даних на LCD-дисплей
6	Enable	Підключено до PIN-коду мікроконтролера та перемикається між 1 і 0 для підтвердження даних
7	Data Pin 0	Виводи даних від 0 до 7 утворюють 8-бітну лінію даних. Їх можна підключити до мікроконтролера для надсилання 8-бітних даних. LCD дисплей також може працювати в 4-бітовому режимі, у такому разі контакти даних 4, 5, 6 і 7 залишаться вільними
8	Data Pin 1	
9	Data Pin 2	
10	Data Pin 3	
11	Data Pin 4	
12	Data Pin 5	
13	Data Pin 6	

14	Data Pin 7	
15	LED Positive	Позитивний пін підсвічування LED
16	LED Negative	Негативний пін підсвічування LED

Разом із LCD дисплеєм використано мікросхему LCM1602, що дозволяє організувати обмін даними через I2C шину.

Характеристиками LCD дисплею є:

- Робоча напруга від 4,7 В до 5,3 В.
- Струм споживання 1мА без підсвічування.
- Алфавітно-цифровий РК-дисплей, тобто може відображати букви та цифри.
- Складається з двох рядків, у кожному рядку можна надрукувати 16 символів.
- Кожен символ складається з сітки 5×8 пікселів.
- Може працювати як в 8-бітному, так і в 4-бітному режимі.
- Він також може відображати будь-які власноруч створені символи.
- Доступний у зеленому та синьому підсвічуванні.

В якості мікроконтролерної плати використано ESP32. ESP32 (Espressif Systems 32) є мікроконтролером з вбудованим Wi-Fi та Bluetooth, розробленим компанією Espressif Systems. Його призначення включає такі області застосування:

- Інтернет речей (IoT): ESP32 забезпечує можливості підключення до мережі Wi-Fi та Bluetooth, що робить його ідеальним вибором для реалізації проектів IoT. Він може бути використаний для побудови розумних пристроїв, систем автоматизації будинку, моніторингу та управління промисловими процесами та інших IoT-рішень.

– Розробка вбудованих систем: ESP32 є потужним мікроконтролером з багатьма входами/виходами (GPIO), що дозволяє підключати різноманітні пристрої та сенсори. Він використовується для розробки вбудованих систем, таких як системи автоматики, управління роботами, промислові контролери та інші схожі застосування.

– Розробка проектів з електроніки: ESP32 забезпечує розширені можливості з програмування та роботи з різноманітними сенсорами, дисплеями, актуаторами та іншими електронними компонентами. Він може бути використаний для створення різноманітних проектів з електроніки, таких як іграшки, прототипи, системи моніторингу та контролю тощо.

– Розробка протоколів зв'язку та мережевих додатків: ESP32 підтримує різні мережеві протоколи, такі як TCP/IP, HTTP, MQTT і багато інших. Це дозволяє розробляти різноманітні мережеві додатки, включаючи веб-сервери, клієнти IoT-платформ, системи збору даних та інші мережеві рішення.



Рисунок 2.8 – Мікроконтролерна плата ESP32

Призначення контактів ESP32 наведено на рис. 2.9.

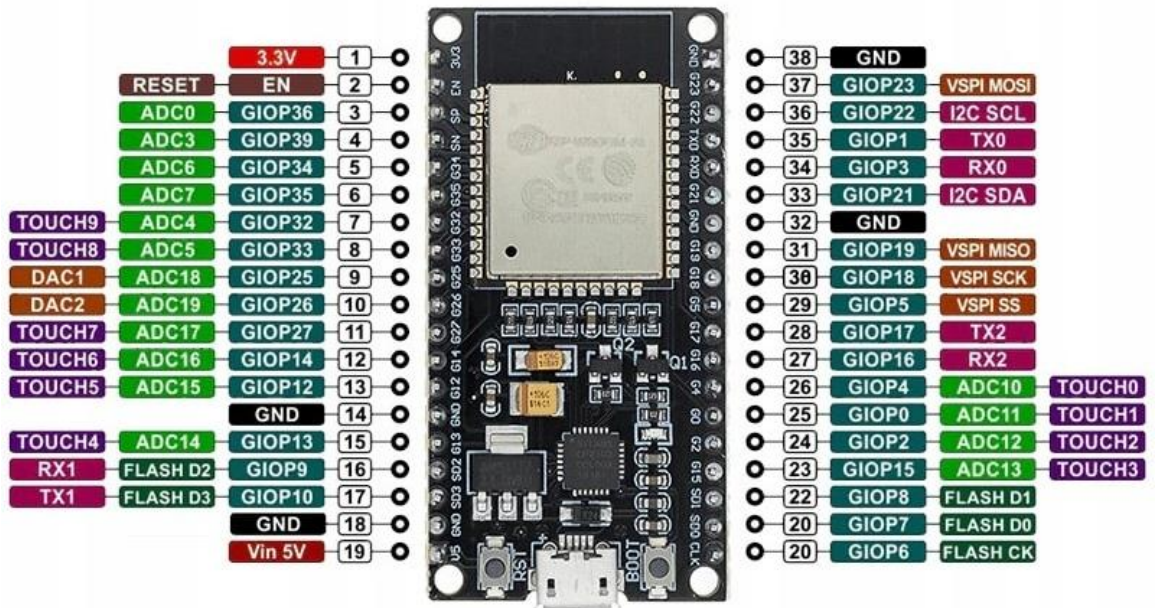


Рисунок 2.9 – Призначення контактів ESP32

2.5.2 Аналіз обраних програмних рішень

Задля реалізації функцій безконтактного вимірювання температури та аналізу виміряних даних, проєктований програмно-технічний пристрій передбачає розгортання на Raspberry Pi середовища розробки сценаріїв Node-RED. Node-RED є веб-інтерфейсом на основі потоків, що допомагає швидко створювати програми для Інтернету речей (IoT) та автоматизації процесів. Він надає зручний графічний інтерфейс, який спрощує створення та маніпулювання потоками даних. Node-RED пропонує бібліотеку вузлів, що виконують різні завдання, такі як обробка даних, комунікація з API, інтеграція з базами даних та іншими сервісами, а також збереження та обробка даних від IoT-датчиків. Його відкритий інтерфейс дозволяє розширити його функціональність за допомогою сторонніх вузлів та плагінів, що дозволяє створювати складні програмні рішення для IoT та інших додатків Інтернету.



Рисунок 2.10 – Місце Node-RED у ланцюгу розробки IoT додатків

Окрім Node-RED на Raspberry Pi потрібно встановити брокер Mosquitto, що використовуватиметься для організації обміну даними по MQTT протоколу. (Message Queuing Telemetry Transport). Mosquitto є популярним брокером MQTT і часто використовується для створення масштабованих систем IoT, включаючи збір даних з сенсорів, керування пристроями та системи віддаленого контролю. Він дозволяє клієнтам підключатись до брокера, відправляти та отримувати повідомлення, підписуватись на теми та отримувати повідомлення, що стосуються цих тем. Брокер також підтримує SSL/TLS для забезпечення безпеки обміну повідомленнями та масштабованості. Mosquitto сумісний з більшістю операційних систем, включаючи Linux, Windows і MacOS, і може бути використаний на різних пристроях, таких як Raspberry Pi, BeagleBone та інші. Він є відкритим джерелом, що дозволяє користувачам змінювати його вихідний код, розширювати його функціональність та налаштовувати його під власні потреби.

Висновки до розділу 2

Проведено опис функційних та нефункційних вимог до проєктованого програмно-технічного засобу. Розроблено структуру та схему електричну принципову для проєктованого програмно-технічного засобу. Наведено монтажну плату. Проведено огляд апаратних та програмних складових пристрою.

					КВРКІ. 190243.43.19.02 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		36

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ БЕЗКОНТАКТНОГО ВИМІРЮВАННЯ ТЕМПЕРАТУРИ ІЗ ФУНКЦІЄЮ ЛОГУВАННЯ НА БАЗІ RASPBERRY PI

3.1 Програмна реалізація функції безконтактного вимірювання температури із функцією логування

Поставлене завдання по реалізації безконтактного вимірювання температури із функцією логування передбачає виконання послідовності дій, які можна описати наступними кроками:

1. користувач підносить мітку у вигляді картки або брелка до зчитувача до модуля RFID, після чого здійснюється зчитування цієї мітки;

2. якщо зчитана мітка присутня у списку відомих міток, то на LCD дисплей виводиться повідомлення стати біля давача вимірювання температури та натиснути кнопку, після чого буде здійснено безконтактне вимірювання температури;

3. отримане значення температури виводиться на LCD дисплей та зберігається у лог файл. Кожний запис у лог файлі складається із наступних полів: Часова мітка, ідентифікатор карти (UID), значення температури, статус (дозволено або заборонено);

4. для індикації та повідомлення користувача, окрім виведення температури на LCD дисплей, здійснюється підсвічування відповідного світлодіода (червоний – доступ заборонений, зелений – доступ дозволений).

Таким чином, для реалізації даного завдання та відповідно до запропонованої структури програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi виконаємо програмну реалізацію клієнтської частини, що представляє собою систему на кристалі ESP32, та серверної частини, яка представлена Node red.

3.1.1 Програмна реалізація клієнтської частини

У запропонованій структурі програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi, мікроконтролерна система ESP32 є клієнтом, що взаємодіє із сервером (Raspberry Pi), надсилаючи і отримуючи дані. У даному випадку, ESP32 надсилає дані до Raspberry Pi (на якому функціонує Node-RED) за допомогою протоколу MQTT.

Розглянемо детальніше код, що контролює роботу системи на кристалі ESP32.

```
#include <SPI.h>
#include <MFRC522.h>
#include <WiFi.h>
#include <PubSubClient.h>
#include <LiquidCrystal_I2C.h>
#include <adafruit_mlx90614.h>
const char* ssid = "YourWiFiSSID";
const char* password = "YourWiFiPassword";
const char* mqttServer = "your_mqtt_server_ip";
const int mqttPort = 1883;
const char* mqttUser = "your_mqtt_username";
const char* mqttPassword = "your_mqtt_password";
#define RST_PIN      22
#define SS_PIN       21
#define Button_PIN   23
#define RedLED_PIN   24
#define GreenLED_PIN 25
MFRC522 mfrc522(SS_PIN, RST_PIN);
Adafruit_MLX90614 mlx = Adafruit_MLX90614();
WiFiClient espClient;
PubSubClient client(espClient);
const int lcdColumns = 16;
const int lcdRows = 2;
LiquidCrystal_I2C lcd(0x27, lcdColumns, lcdRows);
```

Основною задачею, яка ставиться до ESP32 координація показів вимірювання від датчиків (датчик безконтактного вимірювання температури та модуля RFID) та індикація результату на LCD дисплеї та двох світлодіодах.

У даному коді спочатку оголошуються необхідні бібліотеки для роботи із шиною SPI, RFID модулем MR522, датчиком безконтактного вимірювання температури MLX90614 та LCD I2C дисплеєм, а також для роботи із протоколом MQTT.

Далі визначаються змінні для аутентифікації із мережею Wi-Fi (ssid та password), параметри MQTT брокера (IP адреса сервера Raspberry Pi, назва MQTT користувача та його пароль) та набір констант, що описують залучені пінні ESP32, зокрема два пінні для RFID модуля (RST та SS), два пінні для світлодіодів та пін для кнопки. Після цього створюють три необхідні об'єкти для датчика безконтактного вимірювання температури, модуля RFID та LCD дисплея.

Після визначимо MQTT клієнта та набір топиків для підписки та публікації.

```
WiFiClient espClient;
```

```
PubSubClient client(espClient);
```

```
const char* uidTopic = "esp32/UID";
```

```
const char* presentUITopic = "esp32/PresentUI";
```

```
const char* grantedDeniedTopic = "esp32/GrantedDenied";
```

```
const char* temperatureTopic = "esp32/temperature";
```

```
enum State {
```

```
    IDLE,
```

```
    CHECKING_PERMISSION,
```

```
    READING_TEMPERATURE
```

```
};
```

```
State currentState = IDLE;
```

```
const unsigned long permissionTimeout = 5000; // Таймаут очікування дозволу
```

```
unsigned long permissionTimer = 0; // Таймер для очікування дозволу
```

```
bool isGranted = false;
```

Зм.	Арк.	№докум.	Підпис	Дата

Запропонований програмно-технічний засіб використовує чотири топіки, два з них ESP32 використовує для публікації даних (esp32/UID та esp32/temperature, мітки та температури відповідно), інші два – для підписки (esp32/PresentUI та esp32/GrantedDenited).

Також створюється змінна, яка використовується для надходження повідомлення у топик esp32/PresentUI константа, яка визначає максимальний час очікування.

Окремо слід відзначити перерахування State. Дане перерахування описує стани, у яких може перебувати ESP32. Зокрема виділимо стани IDLE, CHECKING_PERMISSION та READING_TEMPERATURE. Перший стан є початковим та використовується для отримання RFID мітки та надсилання її на сервер (публікація у топик esp32/UID). Другий стан є проміжним станом між IDLE і READING_TEMPERATURE та використовується для очікування надходження результату перевірки RFID мітки сервером. В залежності від результату перевірки RFID мітки буде здійснено перехід або до 1 стану (IDLE), якщо мітка відсутня у списку доступних міток, або у третій стан (READING_TEMPERATURE), якщо така мітка присутня у списку доступних міток. Третій стан READING_TEMPERATURE безпосередньо використовується для вимірювання температури пристроєм. Загальна схему функціонування ESP32, що представлена множиною станів подано на рис. 3.1.

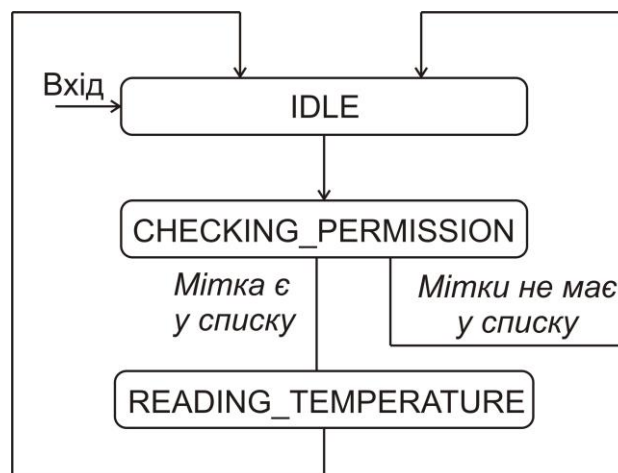


Рисунок 3.1 – Загальна схема функціонування ESP32, що представлена множиною станів

Далі визначимо дві функції connectToWiFi та connectToMQTTBroker для підключення до Wi-Fi мережі та MQTT брокера відповідно. Слід відзначити, що у функції підключення до MQTT брокера здійснюється визначення callback-функції, у якій буде здійснюватись підписка на повідомлення (client.setCallback(callback)).

```
void connectToWiFi() {
    Serial.print("Connecting to Wi-Fi...");
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.print("Connected to Wi-Fi, IP address: ");
    Serial.println(WiFi.localIP());
}

void connectToMQTTBroker() {
    client.setServer(mqttServer, mqttPort);
    client.setCallback(callback);
    while (!client.connected()) {
        Serial.print("Connecting to MQTT Broker...");
        if (client.connect("ESP32Client", mqttUser, mqttPassword)) {
            Serial.println("connected");
        } else {
            Serial.print("failed, rc=");
            Serial.print(client.state());
            Serial.println(" retrying in 5 seconds...");
            delay(5000);
        }
    }
}
```

Далі визначимо функцію для публікації у MQTT топик (publishMessage) та функцію вимірювання температури (readTemperature):

```

void publishMessage(const char* topic, const char* message) {
    client.publish(topic, message);
}

float readTemperature() {
    // Код для зчитування температури
    // Повертає значення температури (float)
    // Додаткові дії, якщо потрібно
    float temperature = mlx.readAmbientTempC();
    return temperature
}

```

Наступним кроком створимо callback функцію, яка обробляє отримані повідомлення з відповідних топиків і залежно від їх змісту встановлює відповідні значення змінних та змінює поточний стан пристрою.

Якщо топик повідомлення збігається із esp32/PresentUI, це означає, що отримано повідомлення про присутність RFID мітки. У функції перевіряється вміст повідомлення. Якщо повідомлення дорівнює «yes», це означає, що мітка присутня і дозвіл надано. Змінна isGranted встановлюється в true, а поточний стан змінюється на READING_TEMPERATURE. Якщо повідомлення дорівнює «no», це означає, що мітка відсутня і дозвіл не надано. Змінна isGranted встановлюється в false, а поточний стан функціонування ESP32 змінюється на IDLE.

У другій перевірці, якщо топик повідомлення збігається із esp32/GrantedDenied, це означає, що отримано повідомлення про надання або відмову в доступі. Всередині функції перевіряється вміст повідомлення. Якщо повідомлення дорівнює «Granted», то це означає, що доступ надано. Виконуються відповідні дії, наприклад, підсвічування зеленого світлодіоду. Якщо повідомлення дорівнює значенню «Denied», то це означає, що доступ користувачу відмовлено, і як результат, виконуються відповідні дії по його сповіщенню, наприклад, підсвічується світлодіод червоного кольору.

```

void callback(char* topic, byte* message, unsigned int length) {
    Serial.print("Message received in topic: ");

```

```

Serial.println(topic);
if (strcmp(topic, presentUITopic) == 0) {
  String presentMessage = String((char*)message);
  Serial.print("Present message: ");
  Serial.println(presentMessage);
  if (presentMessage == "yes") {
    isGranted = true;
    currentState = READING_TEMPERATURE;
  } else if (presentMessage == "no") {
    isGranted = false;
    currentState = IDLE;
  }
}
if (strcmp(topic, grantedDeniedTopic) == 0) {
  String permissionMessage = String((char*)message);
  Serial.print("Permission message: ");
  Serial.println(permissionMessage);
  if (permissionMessage == "Granted") {
    digitalWrite(GreenLED_PIN, HIGH);
    Serial.print("GreenPin is on");
    digitalWrite(RedLED_PIN, LOW);
    delay(1000)
  } else if (permissionMessage == "Denied") {
    digitalWrite(GreenLED_PIN, LOW);
    Serial.print("GreenPin is off");
    digitalWrite(RedLED_PIN, HIGH);/
  }
  PrintMessageInLCD(permissionMessage,0); }}

```

Блок-схему роботи функції зворотнього виклику (callback), яка обробляє отримані повідомлення із відповідних топіків і залежно від їх змісту змінює стан ESP32 наведено на рис. 3.2.

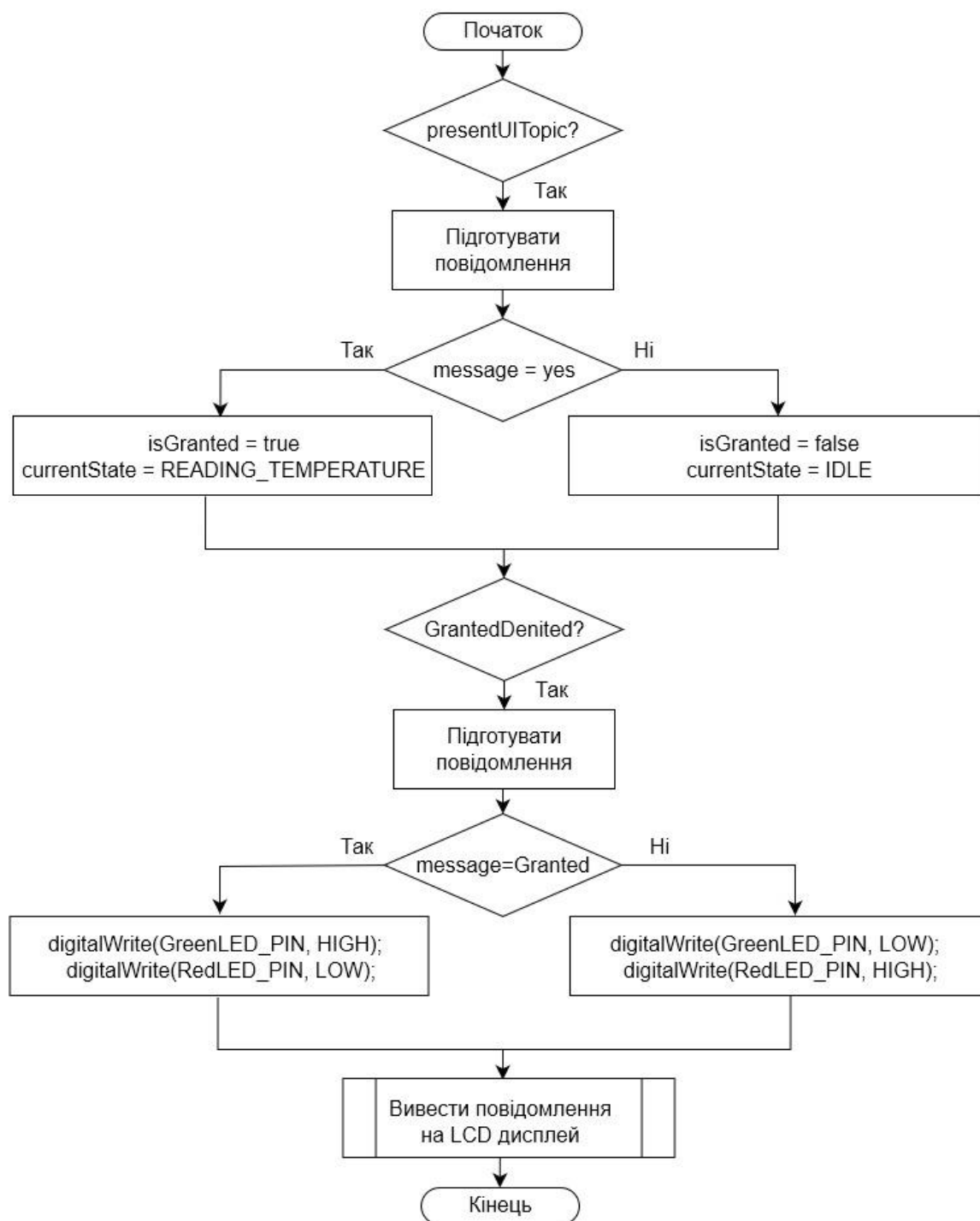


Рисунок 3.2 – Блок-схема роботи функції зворотнього виклику (callback)

Після опрацювання топіків, на які підписаний ESP32, здійснюється виведення на LCD дисплей повідомлення про статус дозволу – Granted або Denied. Код даної функції наведено у наступному лістингу.

```

void PrintMessageInLCD(String message, int choise)
{ lcd.clear();
if (choise == 0){//виведення статусу дозволу в 0-й рядок
    lcd.setCursor(0, 0);
    lcd.write(0);
    lcd.print("status ");
    lcd.setCursor(0, 7);
    lcd.print(message);
}
else{//виведення температури в 1-й рядок
    lcd.setCursor(1, 0);
    lcd.write(0);
    lcd.print("temper ");
    lcd.setCursor(1, 8);
    lcd.print(message); }}

```

На наступному етапі потрібно здійснити створення програмних об'єктів давачів, встановити режими пінів та встановити з'єднання по Wi-Fi та MQTT. Всі зазначені дії виконуються у стандартній функції setup:

```

void setup() {
Serial.begin(115200);
SPI.begin();
mfr522.PCD_Init();
connectToWiFi();
lcd.init();
mlx.begin();
pinMode(Button_PIN, INPUT);
pinMode(GreenLED_PIN, OUTPUT);
pinMode(RedLED_PIN, OUTPUT);
connectToMQTTBroker();
client.subscribe(uidTopic);}

```

Останнім кроком є реалізація стандартної функції loop. Функція loop є основною функцією, яка виконується циклічно після запуску мікроконтролера. Вона виконується безперервно, доки мікроконтролер працює. Основна мета функції loop – виконання основного коду програми та керування пристроями.

У даному програмно-технічному засобі безконтактного вимірювання температури із функцією логування на базі Raspberry Pi реалізується основна логіка функціонування пристрою та опис кожного стану. Розглянемо детальніше логіку роботи цієї функції. Залежно від поточного стану (currentState), виконується логіка для функціонування кожного стану.

У стані IDLE здійснюється зчитування RFID мітки. Якщо мітка виявлена (успішні виклик `mfr522.PICC_IsNewCardPresent()` та `mfr522.PICC_ReadCardSerial()`), отримується її UID і надсилається на MQTT у топик `uidTopic`. Після цього ESP32 зупиняє комунікацію з міткою (викликаючи функції `mfr522.PICC_HaltA()` та `mfr522.PCD_StopCrypto1()`), встановлює стан почтовим станом стан `CHECKING_PERMISSION` і запускає таймер `permissionTimer` для очікування дозволу.

У стані `CHECKING_PERMISSION` перевіряється, чи пройшов таймаут `permissionTimeout` для отримання дозволу. Якщо таймаут вичерпано (`millis() - permissionTimer >= permissionTimeout`), ESP32 повертається в стан IDLE. В іншому випадку, якщо таймаут ще не вичерпано, виконується перевірка дозволу. Якщо отримано дозвіл (`isGranted` дорівнює `true`), ESP32 переходить у стан `READING_TEMPERATURE`.

У стані `READING_TEMPERATURE` здійснюється зчитування температури за допомогою функції `readTemperature()`. Значення температури перетворюється на рядок і надсилається у MQTT топик `temperatureTopic`. Після цього ESP32 повертається в початковий стан IDLE.

Робота нового циклу розпочинається із затримкою в 1 секунду.

Таким чином результуючий код функції loop наведений у наступному лістингу:

```

void loop() {
  client.loop();
  switch (currentState) {
    case IDLE:
      if(mfrc522.PICC_IsNewCardPresent() && mfrc522.PICC_ReadCardSerial()) {
        String uid = "";
        for (byte i = 0; i < mfrc522.uid.size; i++) {
          uid += String(mfrc522.uid.uidByte[i], HEX);
        }
        publishMessage(uidTopic, uid.c_str());
        currentState = CHECKING_PERMISSION;
        permissionTimer = millis();
      }
      break;
    case CHECKING_PERMISSION:
      if (millis() - permissionTimer >= permissionTimeout) {
        currentState = IDLE;
      } else {
        if (isGranted) {
          currentState = READING_TEMPERATURE;
        } else {
          currentState = IDLE;
        }
      }
      break;
    case READING_TEMPERATURE:
      float temperature = readTemperature();
      PrintMessageInLCD(String(temperature),1);
      char temperatureString[10];
      sprintf(temperatureString, "%.2f", temperature);
  }
}

```

```

publishMessage(temperatureTopic, temperatureString);
currentState = IDLE; // Повернення у стан очікування мітки
break;
}
delay(1000)
}

```

Таким чином приведено програмну реалізацію для координації показів вимірювання від датчиків (датчик безконтактного вимірювання температури та модуля RFID) та індикація результату на LCD дисплеї та двох світлодіодах для клієнта на базі системи на кристалі ESP32.

3.1.2 Програмна реалізація серверної частини (сценарію у Node red)

Програмна реалізація серверної частини передбачала реалізацію сценаріїв перевірки присутності RFID мітки у списку доступних міток, перевірку отриманого значення температури та запис у лог-файл метаданих про вимірювання. Розподіл функцій між клієнтом та сервером наведено на рис. 3.3.

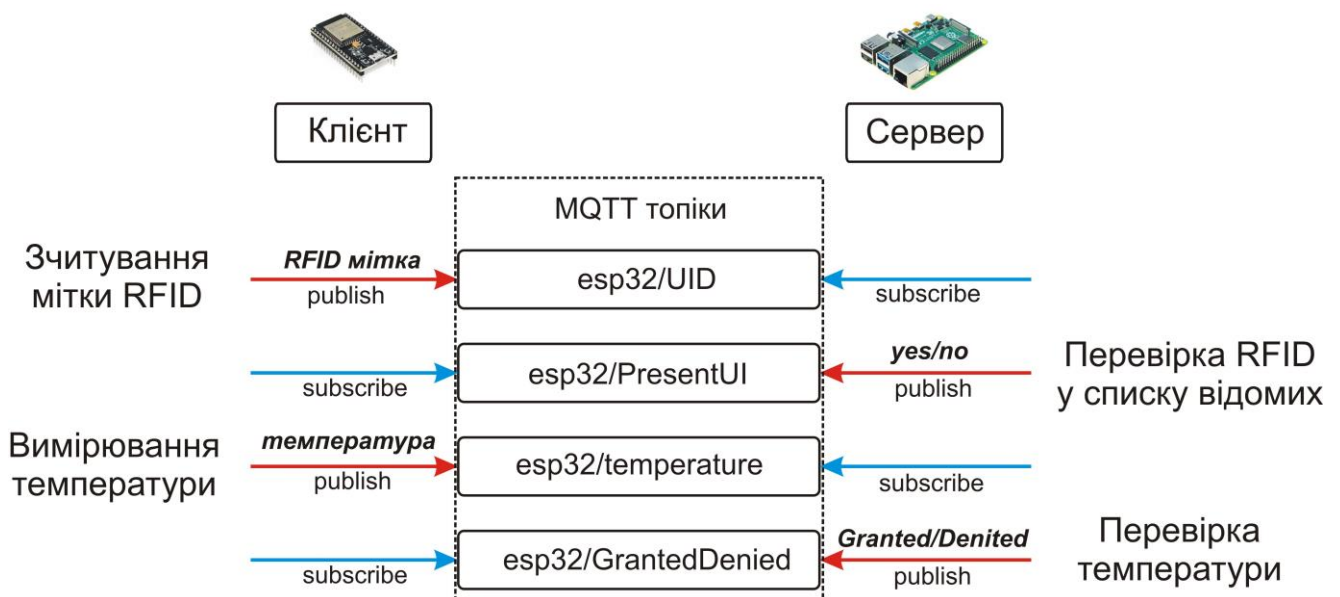
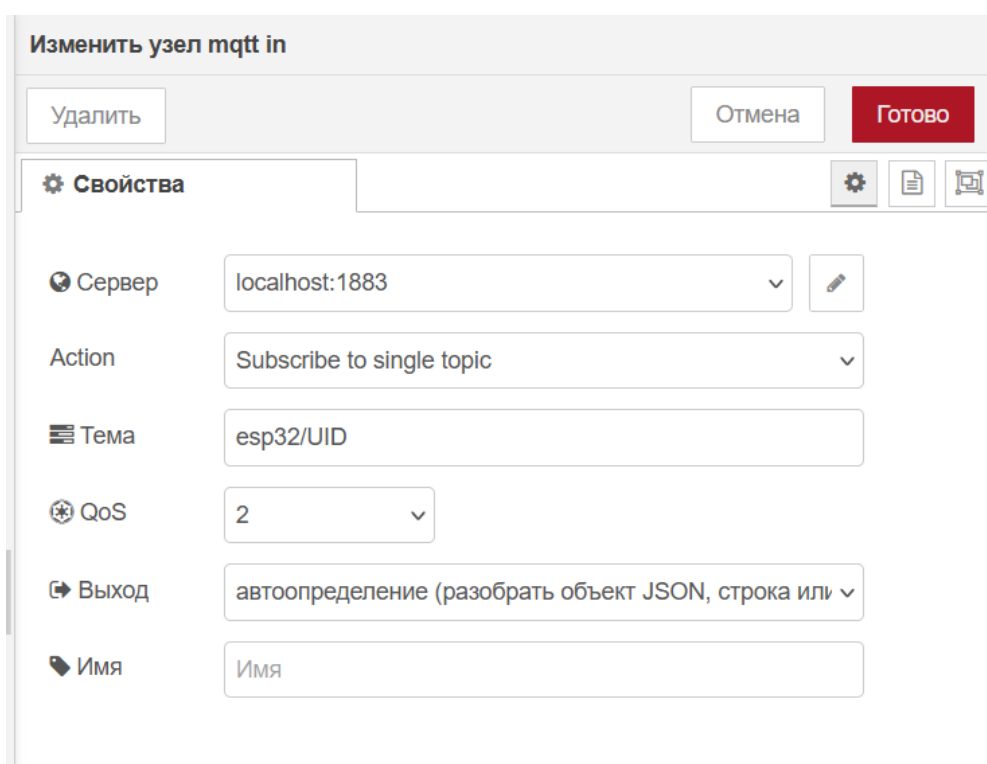


Рисунок 3.3 – Розподіл функцій між клієнтом та сервером

При взаємодії між клієнтом та сервером, зі сторони сервера здійснюється перевірка RFID міток та перевірка температури. Для реалізації цих функцій сервер, що представлений Raspberry Pi, на якому розгорнуто Node red, виконує підписку на два топіки (esp32/UID та esp32/temperature) для отримання вхідних даних та два топіки для публікації результату (esp32/PresentUI та esp32/GrantedDenited).

Розглянемо процес створення сценарію для виконання серверних функцій у Node red.

Спочатку додамо до потоку чотири вузла для взаємодії по протоколу MQTT: по два для підписки та публікації (рис. 3.4-3.7).



The image shows the configuration window for an MQTT In node in Node-RED. The window title is "Изменить узел mqtt in". At the top, there are three buttons: "Удалить" (Delete), "Отмена" (Cancel), and "Готово" (Done). Below the buttons is a "Свойства" (Properties) section with several configuration fields:

- Сервер** (Server): localhost:1883
- Action**: Subscribe to single topic
- Тема** (Topic): esp32/UID
- QoS**: 2
- Выход** (Output): автоопределение (разобрать объект JSON, строка или ...) (auto-detection (parse JSON object, string or ...))
- Имя** (Name): Имя (Name)

Рисунок 3.4 – Вузол підписки на топік esp32/PresentUI

Edit mqtt in node

Delete Cancel Done

Properties [Settings] [List] [View]

Server localhost:1883 [Edit]

Action Subscribe to single topic [v]

Topic esp32/temperature

QoS 2 [v]

Output auto-detect (parsed JSON object, string or buffer) [v]

Name Name

Рисунок 3.5 – Вузол підписки на топик esp32/temperature

Edit mqtt out node

Delete Cancel Done

Properties [Settings] [List] [View]

Server localhost:1883 [Edit]

Topic esp32/PresentUI

QoS 2 [v] Retain [v]

Name Name

Рисунок 3.6 – Вузол публікації у топик esp32/PresentUI

Зм.	Арк.	№докум.	Підпис	Дата

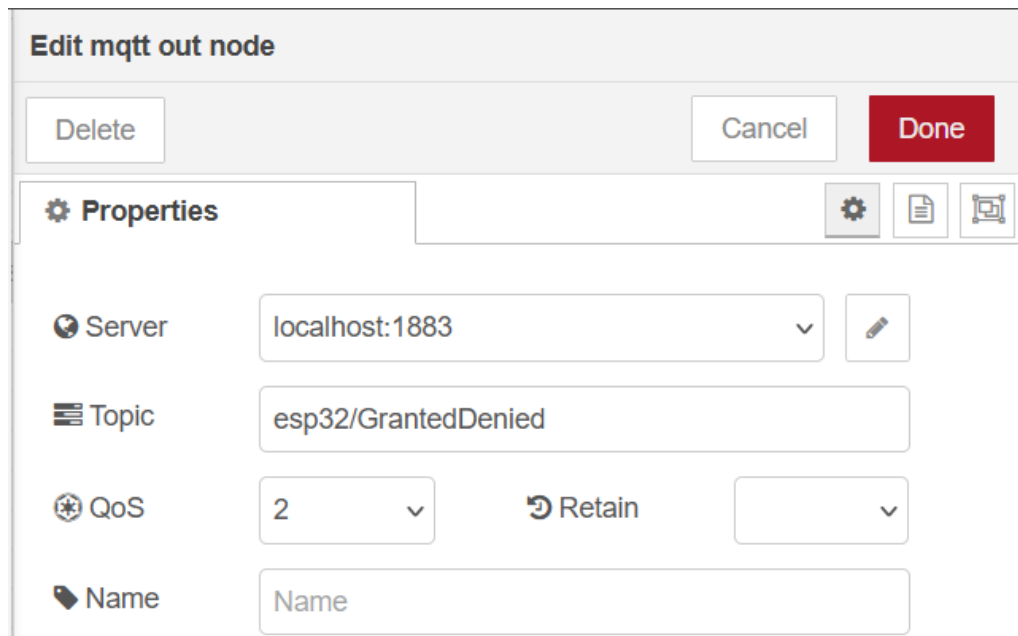


Рисунок 3.7 – Вузол публікації у топик esp32/GrantedDenied

Далі додамо вузол Change у якій оголосимо зміну потоку flow.GrantedStatus та встановимо її початкове значення – Denied, що означає на початку заборону доступу для користувача. Властивості вузла Change наведено на рис. 3.8.

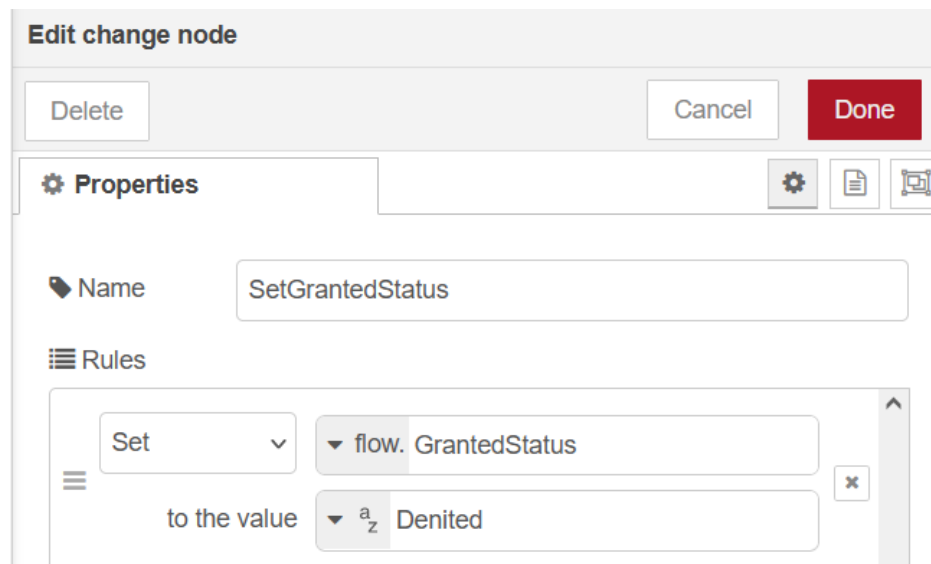


Рисунок 3.8 – Властивості вузла Change, що встановлює змінну потоку flow.GrantedStatus

Далі додамо вузол function із назвою CheckUID, у якому реалізуємо логіку перевірки UID із списком відомих міток. Програмний код вузла CheckUID

наведено на рис. 3.9. Під'єднаємо даний вузол між вузлами для роботи із MQTT esp32/UID та esp32/PresentUI.

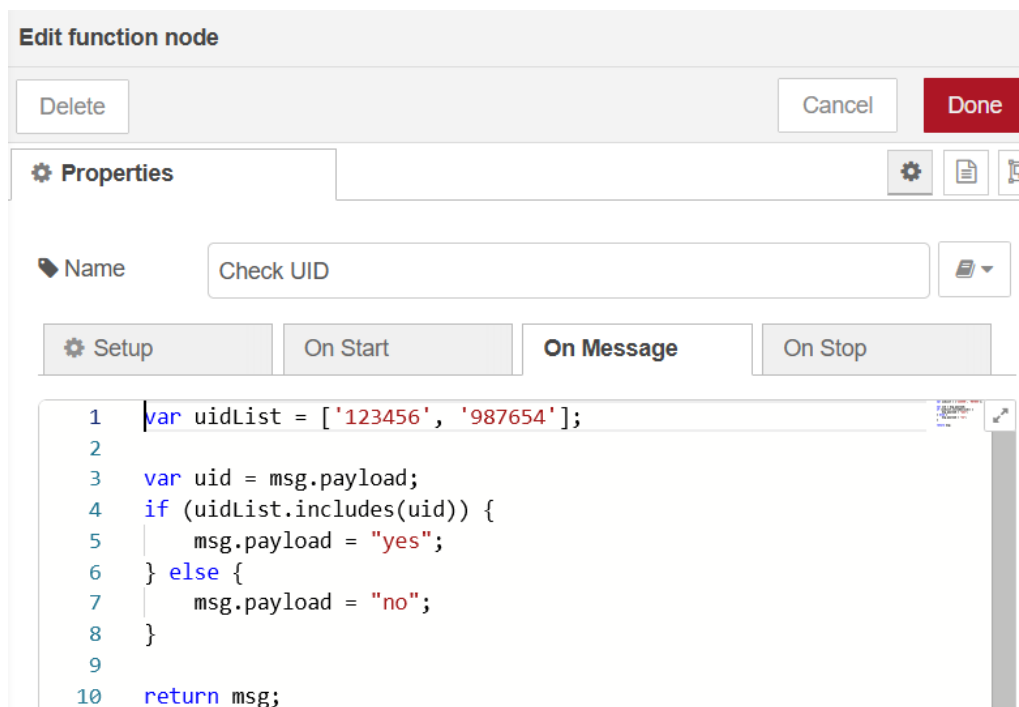


Рисунок 3.9 – Властивості вузла CheckUID

Таким чином частина потоку, у якій здійснюється підписка на топик, із якого отримуються RFID мітки, перевірка цієї мітки на предмет її наявності у списку відомих міток та публікація результату (повідомлення “yes” або “no”) у топик esp32/PresentUI наведено на рис. 3.10.

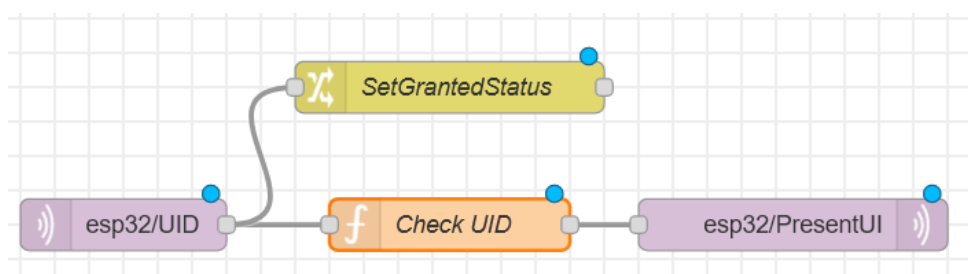


Рисунок 3.10 – Частина потоку для перевірки RFID мітки на предмет її наявності у списку відомих міток

Далі дома вузол function із назвою Check Temperature, яка здійснюватиме перевірку температури, яка була отримана із топіку esp32/temperature. Вихід вузла Check Temperature (повідомлення “Granted” або “Denied”) з’єднується із входом вузла MQTT out esp32/GrantedDenied.

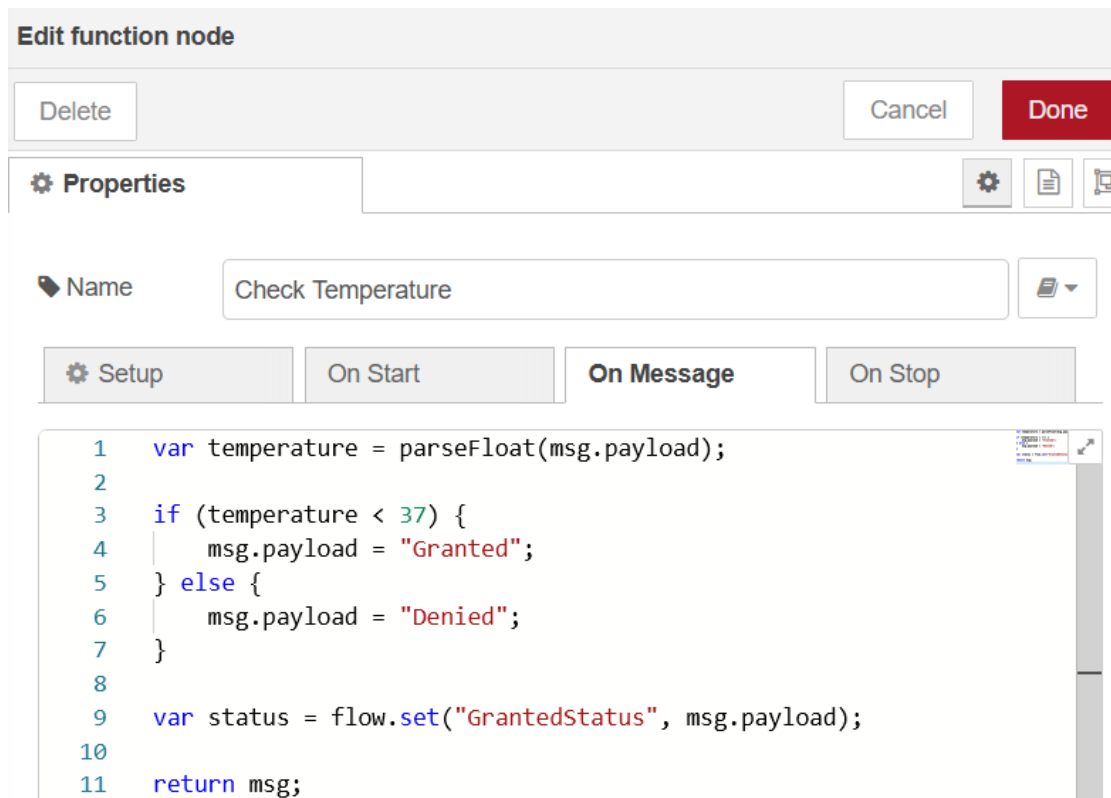


Рисунок 3.11 – Властивості вузла Check Temperature

Таким чином частина потоку визначення рівня температури та надсилання результату у топик esp32/GrantedDenied наведено на рис. 3.12.

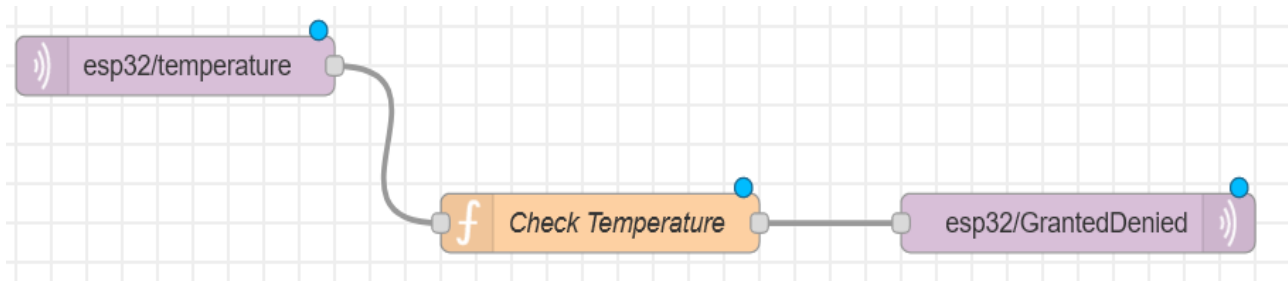


Рисунок 3.12 – Частина потоку визначення рівня температури

Наступним кроком є об'єднання двох потоків. З цією метою додамо вузол Join із назвою Merge two message. У полі о'бєднати після надходження повідомлень встановимо значення 2, що означатиме очікування двох повідомлень – RFID мітки та значення температури (рис. 3.13).

Edit join node

Delete Cancel Done

⚙️ Properties [Settings] [Save] [Reset]

Mode: manual

Combine each: msg. payload

to create: a key/value Object

using the value of msg. topic as the key

Send the message:

- After a number of message parts: 2 and every subsequent message.
- After a timeout following the first message: seconds
- After a message with the msg.complete property set

Name: Merge two message

Рисунок 3.13 – Властивості вузла Merge two message

Результатом вузла Merge two message є JSON об'єкт із двома ключами, що відповідають назвами топіків.

Далі реалізуємо логіку запису у log-файл. Відповідно до поставленого завдання порція даних для запису у файл буде виглядати наступним чином:

- Часова мітка;
- ідентифікатор карти (UID);
- значення температури;
- статус (“Granted” або “Denied”).

Для цього додамо вузол function із назвою Create data to log. Сформуємо JSON об'єкт із чотирма ключами, що відповідають необхідним полям порції даних, яку буде збережено у log-файл. Значення ключів 'Temperature' та 'UID' отримаємо від змінної msg.payload, часову мітку згенеруємо викликом new Date(), а статус надання дозволу отримаємо із змінної потоку flow.get("GrantedStatus"). Властивості вузла Create data to log наведено на рис. 3.14.

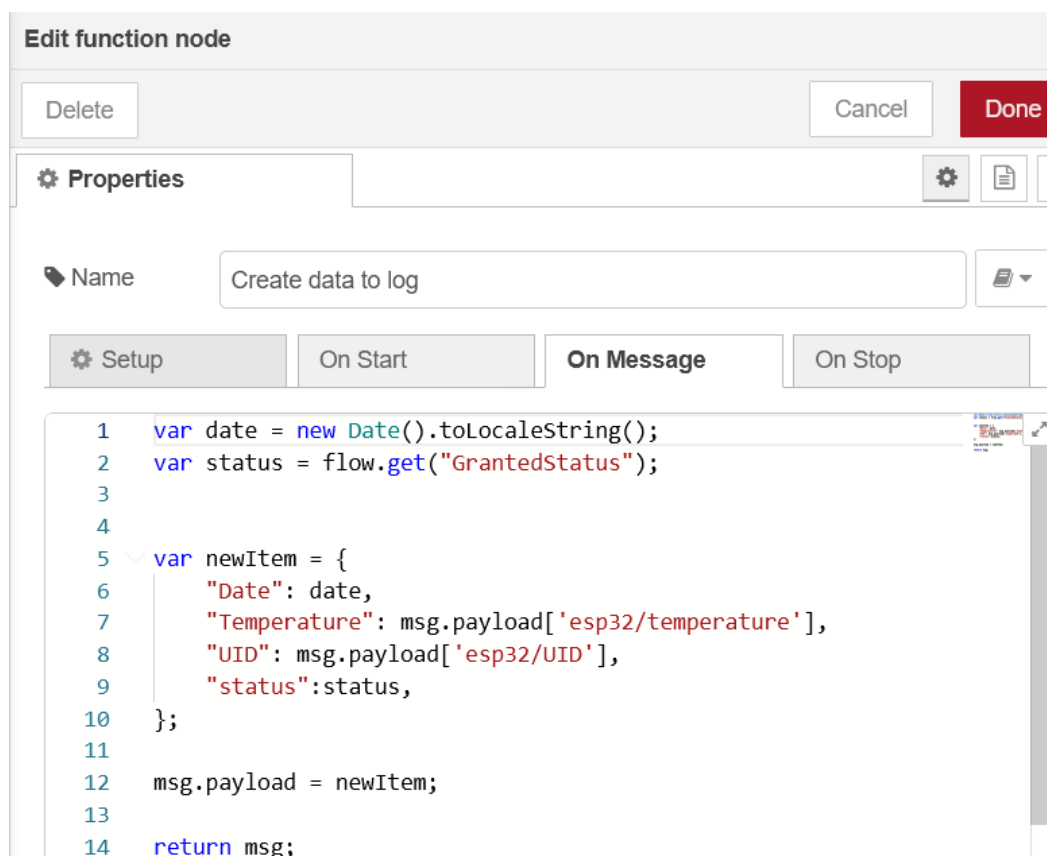


Рисунок 3.14 – Властивості вузла Create data to log

Далі додамо вузол CSV. Основне його призначення це формування із об'єкта JSON CSV представлення. Задамо стовпці як Date, Temperature, UID, satus (рис. 3.15).

Edit csv node

Delete
Cancel
Done

Properties

⚙️
📄
🖨️

☰ Columns

🔤 Separator

📁 Name

CSV to Object options

➡️ Input
Skip first lines

- first row contains column names
- parse numerical values
- include empty strings
- include null values

➡️ Output

Object to CSV options

➡️ Output

Рисунок 3.15 – Властивості вузла CSV

Останнім кроком є додавання вузла Write to file. Виконаємо його налаштування, вказавши шлях до log-файлу. Властивості вузла Write to file наведено на рис. 3.16.

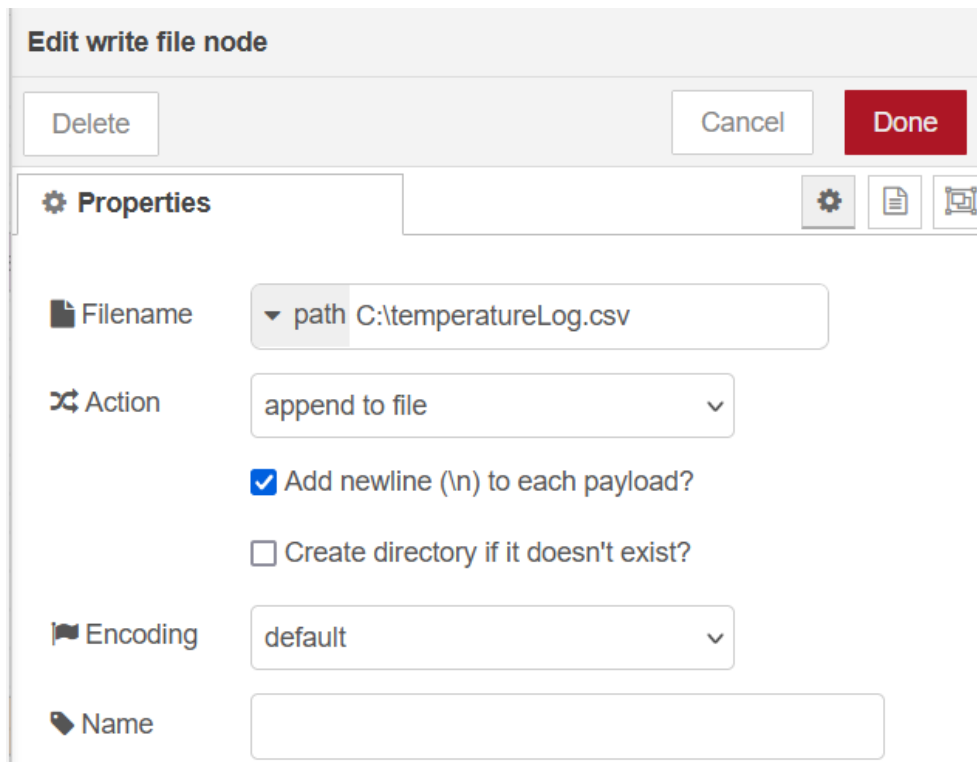


Рисунок 3.16 – Властивості вузла Write to file

Таким чином результуючий потік серверної частини наведено на рис. 3.17.

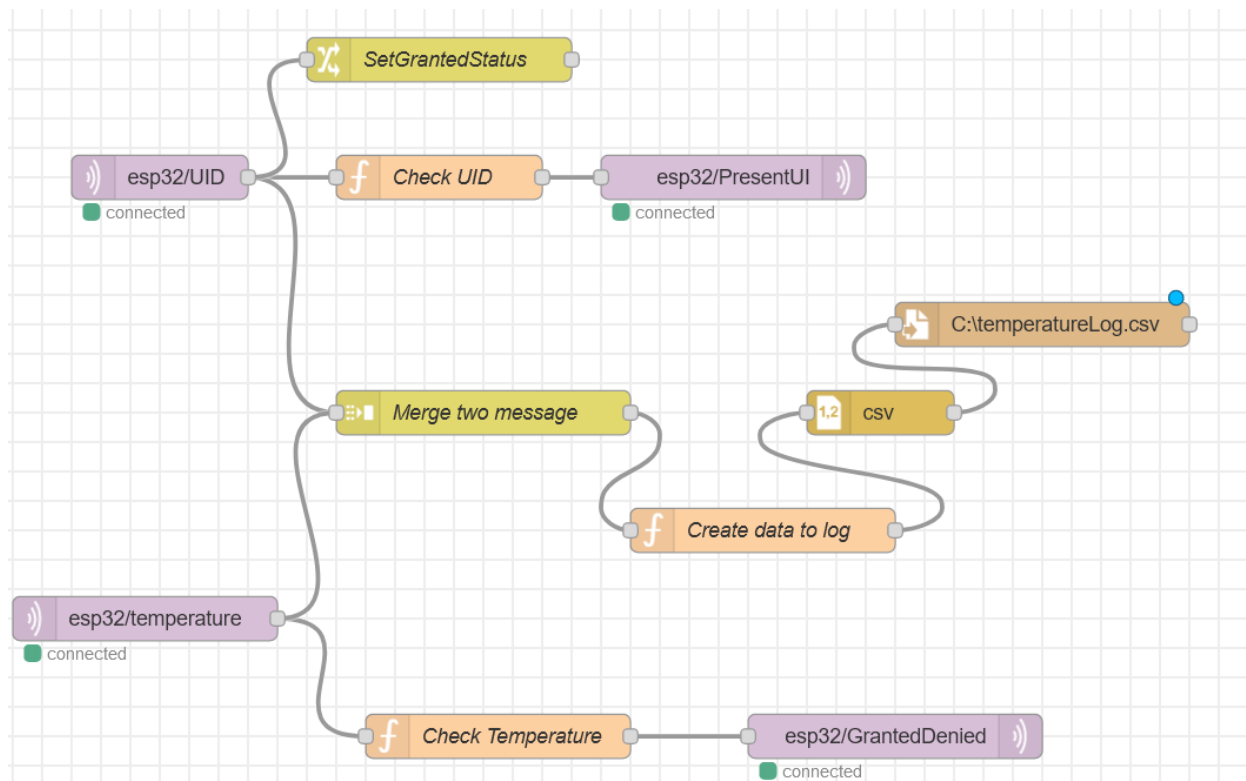


Рисунок 3.17 – Результуючий сценарій у Node red для серверної частини

Зм.	Арк.	№докум.	Підпис	Дата

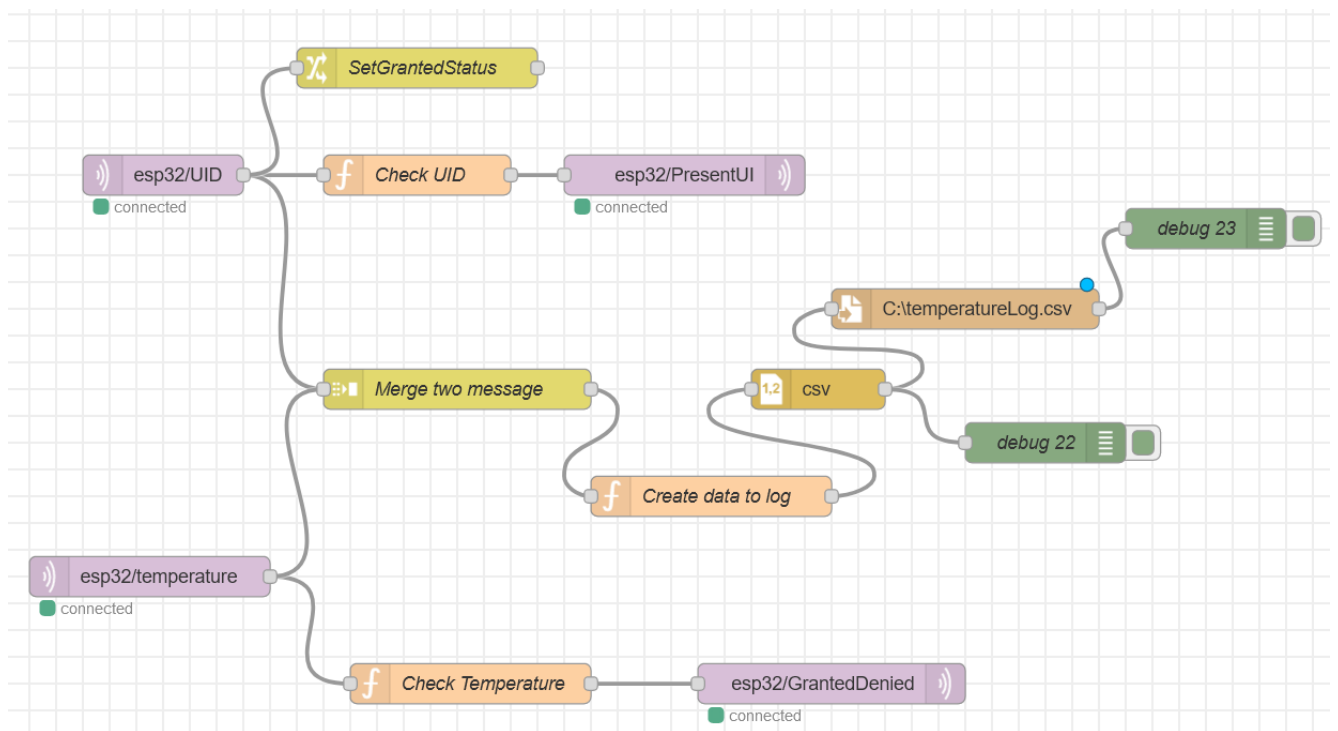


Рисунок 3.18 – Результуючий сценарій у Node red для серверної частини (із додаванням двох вузлів Debug для відлагодження)

Для тестування роботи представленого сценарію виконаємо публікацію у топик esp32/UID повідомлення “303020”, а у esp32/temperature повідомлення “35”. Та виконаємо підписку на топик esp32/GrantedDenied. Як видно із рис. 3.19, було отримано повідомлення Granted

```
C:\Program Files (x86)\mosquitto>mosquitto_pub -t "esp32/UID" -m "303020"
C:\Program Files (x86)\mosquitto>mosquitto_pub -t "esp32/temperature" -m "35"
```

Рисунок 3.19 – Публікація у топик esp32/UID повідомлення “303020”, а у esp32/temperature повідомлення “35”

```
C:\Program Files (x86)\mosquitto>mosquitto_sub -t "esp32/GrantedDenied"
Granted
_
```

Рисунок 3.20 – Підписка на топик esp32/GrantedDenied

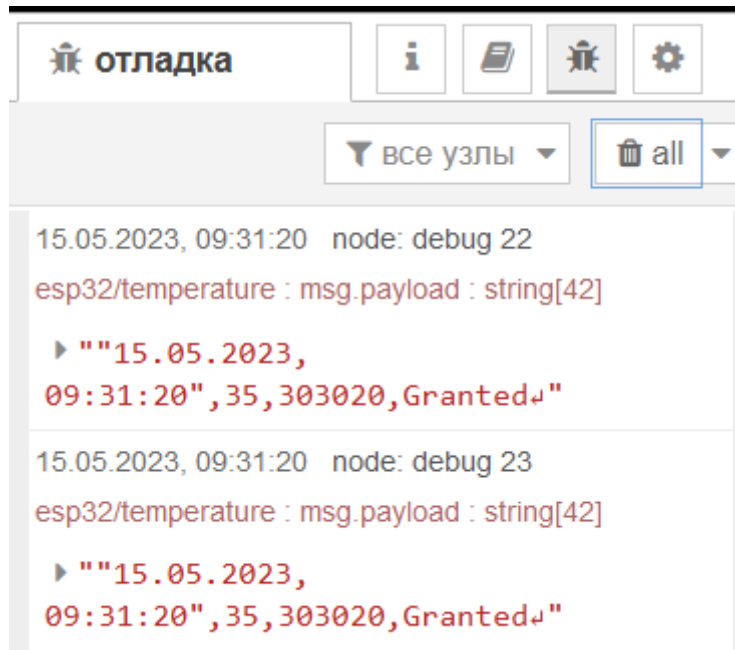


Рисунок 3.21 – Результат тестування сценарію консолі Node red



```
"15.05.2023, 09:24:53", 36, 2030, Granted
"15.05.2023, 09:24:59", 50, 3030, Denied
"15.05.2023, 09:25:38", 50, 3030, Denied
"15.05.2023, 09:30:24", 35, 303020, Denied
"15.05.2023, 09:31:20", 35, 303020, Granted
"15.05.2023, 09:59:08", 36, 2020, Granted
"15.05.2023, 09:59:37", 36, 2020, Granted
```

Рисунок 3.22 – Вміст лог-файлу програмно-технічного пристрою

Висновки за розділом 3

Таким чином було здійснено програмну реалізацію програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi. Функціонування спроектованого засобу включало розділення функцій на клієнтську та серверну частини. Реалізація клієнтської частини передбачала програмування системи на кристалі ESP32, а реалізація серверної частини передбачала написання сценарію у Node red. В результаті було реалізовано функціонал програмно-технічного засобу, що включав в себе виконання наступних кроків:

1. користувач підносить мітку до модуля RFID, після чого здійснюється зчитування цієї мітки;

2. якщо зчитана мітка присутня у списку відомих міток, то здійснюється безконтактне вимірювання температури;

3. отримане значення температури виводиться на LCD дисплей та зберігається у лог файл.

4. для індикації та повідомлення користувача, окрім виведення температури на LCD дисплей, здійснюється підсвічування відповідного світлодіода.

					КВРКІ. 190243.43.19.02 ПЗ	Арк.
						60
Зм.	Арк.	№докум.	Підпис	Дата		

ВИСНОВКИ

Створення програмно-технічного засобу безконтактного вимірювання температури з функцією логування на базі Raspberry Pi, яка включає контроль доступу користувачів за показами їх температури, є особливо актуальною сьогодні, особливо в контексті боротьби з пандемією коронавірусу COVID-19. У цьому випадку, до пристрою може бути додана RFID мітка, яку користувач зчитує. Після зчитування мітки, засіб автоматично вимірює температуру користувача без контакту. Якщо температура перевищує норму (наприклад, встановлену медичними стандартами для виявлення підозри на зараження COVID-19), то доступ користувачу може бути заборонений, наприклад, через відмову відкриття дверей або активацію аудіо-візуальних сигналів.

Важливою функцією програмно-технічного засобу безконтактного вимірювання температури є логування. Ця функція дозволяє зберігати і аналізувати дані про вимірювану температуру з різних часових точок. Одним із головних аспектів вимірювання температури в контексті COVID-19 є відстеження можливих випадків підвищеної температури, яка може бути одним з симптомів захворювання. Логування показників температури дозволяє створити історичний запис про кожне вимірювання та зберігати цю інформацію для подальшого аналізу.

В результаті було спроектовано та реалізовано програмно-технічний засіб для безконтактного вимірювання температури із функцією логування на основі одноплатної комп'ютерної системи Raspberry Pi.

У першому розділі розглянуто принципи організації ІЧ-телеметрії. Проведено аналіз програмно-технічних засобів безконтактного вимірювання температури показав, що розглянуті засоби володіють досить високими показниками точності вимірювання, мають високий ступінь мобільності, проте основним недоліком даних пристроїв є те, що вони не зберігають дані про користувачів, для яких здійснено вимірювання температури (незважаючи на те,

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 61
Зм.	Арк.	№докум.	Підпис	Дата		

що температура зберігається, не має можливості асоціювати збережену температуру із користувачем).

У другому розділі проведено опис функційних та нефункційних вимог до проєктованого програмно-технічного засобу. Розроблено структуру та схему електричну принципову для проєктованого програмно-технічного засобу. Наведено монтажну плату. Проведено огляд апаратних та програмних складових пристрою.

У третьому розділі було здійснено програмну реалізацію програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi. Функціонування спроектованого засобу включало розділення функцій на клієнтську та серверну частини. Реалізація клієнтської частини передбачала програмування системи на кристалі ESP32, а реалізація серверної частини передбачала написання сценарію у Node red. В результаті було реалізовано функціонал програмно-технічного засобу, що включав в себе виконання наступних кроків: користувач підносить мітку до модуля RFID, після чого здійснюється зчитування цієї мітки; якщо зчитана мітка присутня у списку відомих міток, то здійснюється безконтактне вимірювання температури; отримане значення температури виводиться на LCD дисплей та зберігається у лог файл; для індикації та повідомлення користувача, окрім виведення температури на LCD дисплей, здійснюється підсвічування відповідного світлодіода.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Katravath R. et al Fire Alarm Robot and Authentication System Using Raspberry Pi and Cloud, *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol. 8, Iss. 4S2, 2019. – pp. 256-259
2. Sathish P. K. et al. Smart home automation using Raspberry PI 4. *AIP Conference Proceedings*. 2463. 020012.
3. Шостак І. В., Данова М. О., Феоктистова О. І. Підхід до роботизації процесів функціонування системи «Розумний будинок» на основі Інтернету речей. *Інтегровані інтелектуальні робототехнічні комплекси* : матеріали XIII між нар. наук.-практ. конф. (19-20 трав., 2020 р., м. Київ) Київ, 2020, С.48-49.
5. Sensor tips. Understanding the Infrared Temperature Sensor, URL: <https://www.sensortips.com/temperature/infrared-temperature-sensor/>
6. Components 101. RC522 RFID Module URL: <https://components101.com/wireless/rc522-rfid-module>
7. Basri A. H. H. Ibrahim S. N., Malik N. A. and Asnawi A. L. Integrated Surveillance System with Mobile Application, *2018 7th International Conference on Computer and Communication Engineering (ICCCE)*, Kuala Lumpur, Malaysia – 2018, pp. 218-222
8. Patchava V., Kandala H. B. and Babu P. R., A Smart Home Automation technique with Raspberry Pi using IoT, *2015 International Conference on Smart Sensors and Systems (IC-SSS)*, Bangalore, India, 2015, pp. 1-4
9. Karpagam G.R., Kumar B.V., Maheswari J.U., Gao X.-Z. Smart Cyber Physical Systems Chapman and Hall, *CRC*, 2020 – 294 p.
10. Глибовець А.М., Моголівський В.О. Аналіз систем підтримки розумного будинку. *Control systems and computers*. 2019. No 5(283). С. 30–37.
11. Arduino ua, LCD 1602 символний дисплей 16x2 (синій), URL: <https://arduino.ua/prod169-lcd-1602-simvolnii-displei-16x2-sinii>
12. Arduino ua RFID модуль RC522 з картокою доступу для Arduino URL: <https://arduino.ua/prod649-rfid-modyl-rc522-s-kartochkoi-dostupa-dlya-arduino>

13. Sensor tips. Understanding the Infrared Temperature Sensor, URL: <https://www.sensortips.com/temperature/infrared-temperature-sensor/>
14. Buniyamin N. Development of Fire Alarm System using Raspberry Pi and Arduino Uno, Conference: *International Conference On Electrical, Electronics and Systems Engineering (ICEESE)*, December 2013, pp. 37-42.
15. Bhuvanewari S. Fire Detection Using Raspberry Pi, *International Journal of Electrical Engineering and Technology*, Vol. 12, № 1, 2022, pp. 73-80.
16. Конспект лекцій з дисципліни «Комп'ютерні системи» для студентів напряму підготовки «Комп'ютерна інженерія» / І. М. Лазарович. – Івано-Франківськ : Видавництво Прикарпатського національного університету імені Василя Стефаника, 2014. – 190 с.
17. Feng J., Feng Y., Ningzhao L. and Benxiang W., Design and experimental research of video detection system for ship fire, *2019 2nd International Conference on Safety Produce Informatization (ICSPI)*, 2019, pp. 367-370, doi: 10.1109/ICSPI48186.2019.9095929.
18. Jamadagni S., Sankpal P., Patil S., Chougule N. and Gurav S., Gas Leakage and Fire Detection using Raspberry Pi, *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, 2019, pp. 495-497, doi: 10.1109/ICCMC.2019.8819678.
19. Sheth M., Trivedi A., Suchak K., Parmar K. and Jetpariya D., Inventive Fire Detection utilizing Raspberry Pi for New Age Home of Smart Cities, *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 2020, pp. 724-728, doi: 10.1109/ICSSIT48917.2020.9214108.
20. Mahamudul H., Islam M., Shameem A., Rana J. and Metselaar H., Modelling of PV module with incremental conductance MPPT controlled buck-boost converter, *2013 2nd International Conference on Advances in Electrical Engineering (ICAEE)*, 2013, pp. 197-202
21. Фурман І.О., Староверов Р.М., Мельський Д.О. Огляд можливостей «розумного будинку» для покращання побутових умов та зменшення витрат на

утримання домогосподарств. *Енергетика та комп'ютерно-інтегровані технології в АПК*. 2014. No 2. С. 79–80.

22. Valov N. and Valova I., Home Automation System with Raspberry Pi, *2020 7th International Conference on Energy Efficiency and Agricultural Engineering (EE&AE)*, Ruse, Bulgaria, 2020, pp. 1-5

23. Singh N. J. and Sidhu E., Raspberry pi based smart fire management system employing sensor based automatic water sprinkler, *2017 International Conference on Power and Embedded Drive Control (ICPEDC)*, 2017, pp. 102- 106.

24. Sathyakala G., Kirthika V. and Aishwarya B., Computer Vision Based Fire Detection with a Video Alert System, *2018 International Conference on Communication and Signal Processing (ICCSP)*, 2018, pp. 0725-0727, doi: 10.1109/ICCSP.2018.8524216.

25. Bin Bahrudin M. S., Kassim R. A. and Buniyamin N., evelopment of Fire alarm system using Raspberry Pi and Arduino Uno, *2013 International Conference on Electrical, Electronics and System Engineering (ICEESE)*, 2013, pp. 43-48, doi: 10.1109/ICEESE.2013.6895040.

26. Верусь В.С., Кондратюк О.І., Ляшко С.С. Розумний будинок або автоматизована система керування житлом. *Студентський вісник Національного університету водного господарства та природокористування*. 2019. Вип. 1(11). С. 119–122.

27. Raju L., Sowmya G., Srividhya S., Surabhi S., Retika M. K. and Reshmika Janani M., Advanced Home Automation Using Raspberry Pi and Machine Learning, *2021 7th International Conference on Electrical Energy Systems (ICEES)*, Chennai, India, 2021, pp. 600-605.

28. Takahashi H. et al., Improvement of automatic fire extinguisher system for residential use, *2015 International Conference on Informatics, Electronics & Vision (ICIEV)*, 2015, pp. 1-4.

29. Swain K. B., Dash S. and Gouda S. S., Raspberry PI based Integrated Autonomous Vehicle using LabVIEW, *2017 Third International Conference on*

Sensing, Signal Processing and Security (ICSSS), 2017, pp. 69-73, doi: 10.1109/SSPS.2017.8071567.

30. Дужак І.О. Розумний будинок. *Автоматизація технологічних і бізнес-процесів*. 2013. No 13(13–14). С. 31-33.

31. Sarath Kumar R., Hariharan J., Revanth R. S., Prasanth K. R. and Lokesh J., Automatic Fire Fighting Robot using RPI, *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*, 2021, pp. 136-139.

32. Гунько Н.В. SMART HOME як інструмент оптимізації вартості житлово-комунальних послуг в Україні / Матеріали Науково-практичної Інтернет-конференції, *Проблеми регіоналістики: минуле, сучасне, майбутнє*, КНЕУ, березень 2017 р. С. 228-231

33. Jan H., Yar H., Iqbal J., Farman H., Khan Z. and Koubaa A., Raspberry Pi Assisted Safety System for Elderly People: An Application of Smart Home, *2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)*, 2020, pp. 155-160.

34. Khan M. N. A., Tanveer T., Khurshid K., Zaki H. and Zaidi S. S. I., Fire Detection System using Raspberry Pi, *2019 International Conference on Information Science and Communication Technology (ICISCT)*, 2019, pp. 1-6.

35. Тарарака В.Д. Архітектура комп'ютерних систем: навч. посіб., Житомир: *ЖДТУ*, 2018, 383 с.

36. Raspberry Pi, URL: <https://www.raspberrypi.org/>

37. Момот Т.В., Мураєв Є.В. Компаративний аналіз зарубіжних практик розвитку розумних міст та можливості їх імплементації в Україні. *Електронний науково-практичний журнал «Інфраструктура ринку»*. 2020. Вип. 42. С. 232–237.

38. Nisan N., Schocken S. The Elements of Computing Systems, second edition: Building a Modern Computer from First Principles 2nd Edition, *The MIT Press*, 2021.

39. Yadin A. Computer Systems Architecture, Chapman and Hall, *CRC*, 2016. – 467 p.

40. Null L., Lobur Y. Essentials of Computer Organization and Architecture, *Jones & Bartlett Learning*; 5th edition, 2018, 744 p.

					КВРКІ. 190243.43.19.02 ПЗ	Арк. 66
Зм.	Арк.	№докум.	Підпис	Дата		

41. Rea P., Ottaviano E., Machado J. and Antosz K. Design, Applications, and Maintenance of Cyber-Physical Systems , *Engineering Science Reference*, 2021. 314 p.
42. Гайдукевич С.В., Семенова Н.П., Леськів Я.А. Особливості SMART-технологій на прикладі автоматизації житлового будинку, *Таврійський науковий вісник*. №1. 2022. С. 12-21.
43. Li B. S. X., Wan B., Wang C., Zhou X., Chen X. Definitions of predictability for cyber physical systems, *J. of Systems Architecture*. 2016.
44. Poliakov, M., Larionova, T. Control Systems with programmable logic controllers, Remote and virtual tools in engineering: textboo, General editorship Dr.Ing.Karsten Henke, aporizhzhya: Dike Pole, 2016, 250 p.
45. Натрошвілі С. Г., Натрошвілі Г. Р., Бабина Т. Г., Злотенко Б. М., Кулік Т. І. Комп'ютерно-інтегрована система керування природним і штучним освітленням розумного будинку, *Вісник Хмельницького національного університету. Серія : Технічні науки*, 2020, № 5 (289), С. 65-71.
46. Monk S. Programming Arduino Next Steps: Going Further with Sketches, *McGraw-Hill Education TAB*, 2018. 320 p.
47. Barrett S.F. Microchip AVR® Microcontroller Primer: Programming and Interfacing , *Morgan & Claypool Publishers*, 2019, 374 p.
48. Papazoglou P. M. An Educational Guide to the AVR Microcontroller Programming: AVR Programming::Demystified (Assembly Language) (Vol.1), *CreateSpace Independent Publishing Platform*, 2018. 274 p.
49. Hu Y., Tilke D., Adams T. et al. Smart home in a box: usability study for a large scale self-installation of smart home technologies. *J Reliable Intell Environ 2.* – 2016, pp. 93-106
50. Kravets A.G., Bolshakov A.A., M.V. Shcherbakov Cyber-Physical Systems: Industry 4.0 Challenges (Studies in Systems, Decision and Control, 260) , *Springer*; 1st ed., 2020., 349 p.

51. Nisan N., Schocken S. The Elements of Computing Systems, second edition: Building a Modern Computer from First Principles 2nd Edition, *The MIT Press*, 2021. 344 p.

52. Rhee J.H., Ma J.H., Seo J., Cha S.H., Review of applications and user perceptions of smart home technology for health and environmental monitoring, *J. Comput. Des. Eng.* No 9. 2022, pp. 857–889

53. Kayastha S. and Upadhyaya P., Design and Implementation of a Cost-Efficient Smart Home System with Raspberry Pi and Cloud Services, *2019 Artificial Intelligence for Transforming Business and Society (AITB)*, Kathmandu, Nepal, 2019, pp. 1-7.

54. Бабенко О. В. Актуальність технологій розумних будинків для підвищення енергоефективності економіки держави, *Матеріали XLVIII наук.-техн. конф. підрозділів Вінниц. нац. техн. ун-ту (НТКП ВНТУ-2019)* : зб. доп., Вінниця, 2019, С. 2920-2921

55. Balta-Ozkan N., Davidson R., Bicket M., Whitmarsh L., Social barriers to the adoption of smart homes, *Energy Policy*, Vol. 63, 2013, pp. 363-374

56. Степаненко О.І. Пасивний будинок – шлях до ефективного використання енергії, *Енергетика: економіка, технології, екологія*, 2014, №3, С. 56-58.

57. Barrett S.F. Microchip AVR® Microcontroller Primer: Programming and Interfacing / S.F. Barrett, D. J. Pack, M. A. Thornton, *Morgan & Claypool Publishers*, 2019, 374 p.

58. Sharma M., Assotally A. and Bekaroo G., RaspMonitor: A Raspberry Pi Based Smart Home Monitoring System, *2022 3rd International Conference on Next Generation Computing Applications (NextComp)*, Flic-en-Flac, Mauritius, 2022, pp. 1-6.

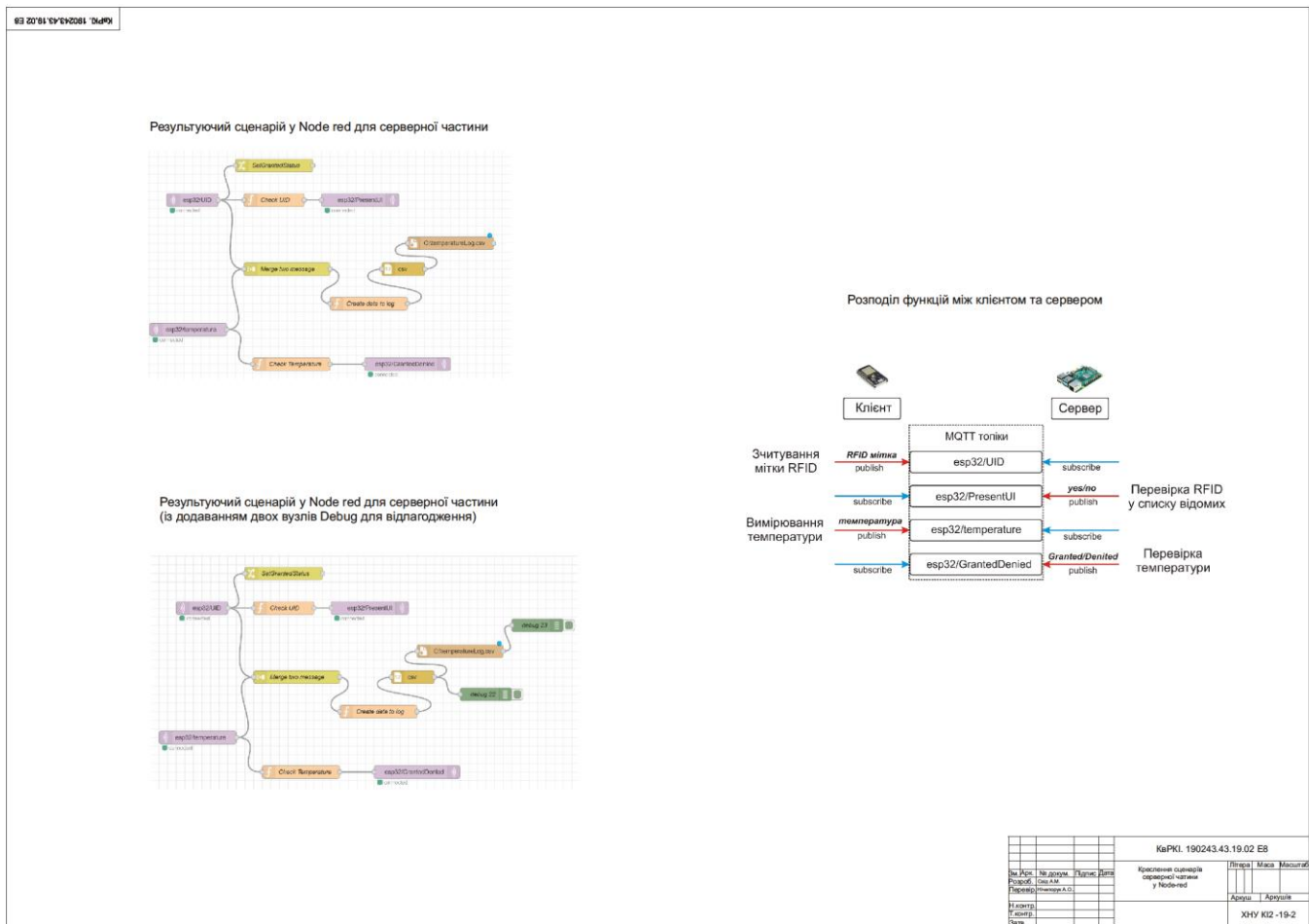
59. Singh N. J. and Sidhu E., Raspberry pi based smart fire management system employing sensor based automatic water sprinkler, *2017 International Conference on Power and Embedded Drive Control (ICPEDC)*, 2017, pp. 102- 106.

60. Poliakov, M., Larionova, T. Control Systems with programmable logic controllers, Remote and virtual tools in engineering: textboo, General editorship Dr.Ing.Karsten Henke, aporizhzhya: Dike Pole, 2016, 250 p.

					КВРКІ. 190243.43.19.02 ПЗ	Арк.
						69
Зм.	Арк.	№докум.	Підпис	Дата		

ДОДАТОК А

Копія креслення «Креслення сценаріїв серверної частини у Node-red»



Ім'я користувача:
Кафедра КІ

Дата перевірки:
22.05.2023 10:40:40 EEST

Дата звіту:
22.05.2023 10:44:30 EEST

ID перевірки:
1015171033

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100005591

Назва документа: Саїд_Програмно-технічний засіб безконтактного вимірювання температури із функцією ло...

Кількість сторінок: 69 Кількість слів: 9737 Кількість символів: 72370 Розмір файлу: 6.13 MB ID файлу: 1014850623

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

3.5% Схожість

Найбільша схожість: 2.08% з джерелом з Бібліотеки (ID файлу: 1014517653)

1.8% Джерела з Інтернету

59

Сторінка 71

2.61% Джерела з Бібліотеки

44

Сторінка 71

0.71% Цитат

Цитати

3

Сторінка 72

Посилання

1

Сторінка 72

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

5

Підозріле форматування

17
сторінок

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 8.0%

Словники перевірки: en_US, ru_RU, ua_UA. Помилки в документах: 17%

ID: 113753 Назва: БКР Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi Додано в БД: 2023-05-22 Автора: А.М. Саїд Керівники: А.О. Нічепорук Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	65378	537	5437 (8%)	47 (9%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Саїд Артур Музафарович

Тема: Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість листів креслень 3; кількість сторінок записки 56

1. Короткий зміст роботи та прийнятих рішень У роботі запропоновано програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

2. Висновок про відповідність роботи дипломному завданню _____
Дипломний проект відповідає виданому завданню _____

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі проведено аналіз предметної області та огляд відомих засобів із функцією безконтактного вимірювання температури. У другому розділі проведено проектування апаратної частини програмно-технічного засобу безконтактного вимірювання температури із функцією логування. У третьому розділі проведено реалізацію програмної частини програмно-технічного засобу безконтактного вимірювання температури із функцією логування на базі Raspberry Pi.

4. Позитивні сторони роботи: Запропоновано структуру та спроектовано програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

5. Негативні сторони роботи: В роботі не наведено оцінку вартості складових компонентів.

6. Оцінка графічного оформлення та пояснювальної записки роботи: пояснювальна записка та листи креслення виконані згідно діючих вимог

7. Відгук про роботу в цілому: В загальному робота виконана на достатньому рівні.

8. Інші зауваження: -


9. Оцінка дипломної роботи:

Розглянувши позитивні та негативні сторони представленої дипломної роботи вважаю, що робота заслуговує оцінки «добре» 3,75 (С)

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) _____

Димченко О.М., доцент кафедри МЗ
Хмельницького національного університету

“31” травня 2023р.



Завідувачу кафедри КІС
д-р.техн.наук, проф. Говорущенко Т. О.

Саїд Артур Музафарович

ПІБ здобувача вищої освіти

ФІТ, 4 курсу, групи КІ2-19-2

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

26 травня 2023 року



РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Програмно-технічний засіб безконтактного вимірювання температури із функцією логування на базі Raspberry Pi

Автор: Саїд Артур Музафарович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Нічепорук Андрій Олександрович, к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи.	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

- 1) запозичення розміщені в розділі аналізу існуючих аналогів та відомих рішень, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 3.50% і адресується до 103 першоджерела, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КПС



А.О. Нічепорук

С. М. Лисенко

Т. О. Говорущенко