

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему:

Комп'ютеризована система контролю температури

фарбувально-сушильної камери для автомобілів

Виконав: студент IV курсу, групи СІ-41

спеціальності 123 Комп'ютерна інженерія

(шифр і назва спеціальності)

(підпис)

Волощук І.В.

(прізвище та ініціали)

Керівник

(підпис)

Стадник Н.Б.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Луцик Н.С.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

Петрик М.Р.

(прізвище та ініціали)

Тернопіль -2026

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Осухівська Г.М.
(прізвище та ініціали)

« 25 » 04 2026 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня бакалавр
(назва освітнього ступеня)

за спеціальністю 123 Комп'ютерна інженерія
(шифр і назва спеціальності)

Студенту Волощуку Ігорю Віталійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютерна система контролю температури фарбувально-сушильної камери для автомобілів

Керівник роботи Стадник Наталія Богданівна., к.т.н.,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 24 » 04 2026 року № 4/9-188

2. Термін подання студентом завершеної роботи 15.06. 2026 р.

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ.

1. Аналіз технічного завдання.

2. Проектна частина.

3. Практична частина.

4. Безпека життєдіяльності, основи охорони праці.

Висновки

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

1. Узагальнена структура системи

2. Схема елементів

3. Налаштування компонентів на платах

4. Алгоритм роботи

АНОТАЦІЯ

Волощук І.В. Комп'ютерна система контролю температури фарбувально-сушильної камери для автомобілів: робота на здобуття кваліфікаційного ступеня бакалавра: спец. 123 — комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2026.

Ключові слова: датчик температури, мікроконтролер STM32, макетна плата, програматор, I²C, STM32CubeIDE, UART

У першому розділі роботи подано огляд існуючих контролерів температури. Серед їх різноманіття відсутні пристрої, що дозволяють користувачеві повністю визначати логіку, яка контролює сигнальні лінії. Запропоновано розробити власний контролер на основі мікроконтролера STM32, яка може бути вбудована у систему контролю температури, що функціонує у фарбувально-сушильній камері для автомобілів.

У другому розділі виконано обґрунтований вибір мікроконтролера, датчика температури, дисплея. У відповідності із поданою елементною базою розроблена електрична принципова схема, яка відтворює всі необхідні електричні з'єднання і зв'язки між компонентами. Здійснено налагодження елементів на макетній та друкованій платах.

У третьому розділі представлена блок-схема алгоритму функціонування розробки. Із застосуванням STM32CubeIDE розроблена управляюча програма на основі периферії, котра застосовується. Описано ініціалізацію периферії та портів мікроконтролера, наведено фрагменти програмного коду. Тестування розробки свідчить, що продуктивності мікроконтролера вистачає для вирішення всіх покладених на нього завдань.

У четвертому розділі роботи розглянуто важливі питання безпеки життєдіяльності та основ охорони праці.

ANNOTATION

Voloshchuk Ihor. Computer System for Temperature Control of an Automotive Paint Drying Booth: Bachelor's Graduation Thesis: speciality 123 — computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2026.

Keywords: temperature sensor, STM32 microcontroller, breadboard, programmer, I²C, STM32CubeIDE, UART

The first section of this Thesis provides an overview of existing temperature controllers. Among their variety, there are no devices that allow the user to fully define the logic that controls the signal lines. It is proposed to develop a custom controller based on the STM32 microcontroller, which can be built into the temperature control system operating in a paint and drying chamber for cars.

In the second section, a reasoned choice of the microcontroller, temperature sensor, and display is made. In accordance with the proposed element base, an electrical schematic diagram was developed that reproduces all the necessary electrical connections and connections between components. Elements were debugged on the breadboard and printed circuit boards.

In the third section, a block diagram of the development algorithm is presented. Using STM32CubeIDE, a control program was developed based on the peripherals used. The initialization of the peripherals and ports of the microcontroller is described, and fragments of the program code are given. Testing of the development shows that the microcontroller's performance is sufficient to solve all the tasks assigned to it.

The fourth section of the work considers important issues of life safety and the basics of occupational safety.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ.....	10
1.1 Мікропроцесори у вбудованих системах.....	10
1.2 Контролери температури.....	16
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА	19
2.1 Узагальнена структура системи.....	19
2.2 Вибір мікроконтролера.....	19
2.3 Вибір датчиків температури.....	24
2.4 Дисплей	30
2.5 Принципова схема.....	32
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА	34
3.1 Налаштування компонентів на платах (макетній та друкованій).....	34
3.2 Розробка алгоритму роботи.....	37
3.3 Програмна складова.....	38
3.3.1 Розробка ПЗ.....	38
3.3.2 Ініціалізація периферії та портів мікроконтролера	41
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	44
4.1 Долікарська допомога при опіках.....	44
4.2 Вимоги пожежної безпеки при гасінні електроустановок.....	46
ВИСНОВКИ.....	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	51
Додаток А Технічне завдання	
Додаток Б Код керуючої програми	

					КС КРБ 123.154.00.00 ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата					
Розроб.	Волощук І.В.				Літ.	Арк.	Аркушів		
Керівник.	Стадник Н.Б.								
Реценз.					ТНТУ, каф. КС, гр. СІ-41				
Н. Контр.	Луцик Н.С.								
Затверд.	Осухівська Г.М								

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І
ТЕРМІНІВ

EasyEDA – безкоштовна, хмарна (веб-орієнтована) система автоматизованого проектування електроніки (EDA), яка працює у браузері або як десктопний застосунок.

I²C (Inter-Integrated Circuit) – популярний протокол послідовного обміну даними

STM32CubeIDE – багатофункціональне безкоштовне середовище розробки (IDE) від компанії STMicroelectronics, створене спеціально для програмування мікроконтролерів серії STM32.

ПІД – пропорційно-інтегрально-диференційна.

ПЗ – програмне забезпечення.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Мікроконтролер, як керуючий пристрій, містить велику кількість цифрових блоків. У тому числі пам'ять, таймери, різні інтерфейси тощо.

Для управління роботою мікроконтролера пишеться програма, що визначає послідовність процесів мікроконтролера. При зміні розв'язуваного мікроконтролером завдання достатньо переписати програму. У результаті більшість специфічних завдань бере на себе мікроконтролер, а решта схема універсальна і не потребує зміни при невеликій зміні функціоналу пристрою.

Впровадження мікроконтролерів у системах керування і опрацювання інформації гарантує високу продуктивність за низької ціни, що робить їх незамінними при побудові високоякісних та недорогих систем. У багатьох простих пристроїв виявляється достатньо одного мікроконтролера.

Збільшення числа контролерів призвело до зниження цін і, як наслідок, їх широкому забезпечує використанню у різних промислових та побутових приладів, особливо 8-бітних моделей. Зростає різноманітність мікроконтролерів, які пропонуються на світовому ринку, з'являються нові високотехнологічні продукти з високим рівнем інтеграції, новими архітектурами. Це пояснює чому мікроконтролери широко використовуються в автоматизації виробництва, обладнання, засобів зв'язку, побутової техніки та багатьох інших галузях людської діяльності.

Температура вибрана через різноманіття ситуацій, у яких увімкнення або вимкнення певного пристрою залежить від температури. Важливою є тематика автоматизації виробничих процесів, зокрема автоматизованого контролю температури фарбувально-сушильної камери для автомобілів. Камера дає змогу регулювати температуру, що є обов'язковою умовою для якісного покриття фарби. Властиво підвищення температури дозволяє прискорити полімеризацію нанесених фарби і лаку, значно скорочуючи ремонтний час.

Метою кваліфікаційної роботи є розробка вбудованої системи контролю температури у невеликій майстерні для спеціальної камери, де проводиться

					<i>КС КРБ 123.154.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		8

фарбування і сушіння авто.

Завдання, необхідні для досягнення даної мети:

- визначення основного функціоналу пристрою, аналіз ринку;
- складання технічного завдання;
- вибір компонентів, створення принципової схеми;
- створення прототипу пристрою та ПЗ;
- створення готового пристрою.

.

					<i>КС КРБ 123.154.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		9

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

Основне призначення фарбувально-сушильної камери – нанесення визначеного шару фарби на поверхню кузова транспортного засобу. Для якісного проведення цього процесу у такій вбудованій системі надзвичайно важливим є дотримання температурного режиму. Контроль за виконанням цієї процедури найкраще здійснювати на основі мікроконтролера.

Варто зауважити, що загалом замість температури можна вибрати будь-яку іншу фізичну величину, значення якої можна отримати від сумісного з мікроконтролером датчика.

1.1 Мікропроцесори у вбудованих системах

Швидкий розвиток технологій та Інтернету призводить до збільшення кількості застосувань нових технологічних рішень на глобальному рівні. З появою концепції Інтернету речей мережі датчиків стають доступними і їх доцільно застосовувати у всіх сферах діяльності, забезпечуючи цим умови для створення експертних систем, які можуть працювати в будь-який час і в будь-якому місці [1].

В даний час комерційно доступні мікроконтролери різних архітектур. Загальна схема мікроконтролера наведена на рис. 1.1. Технічно мікроконтролер виконаний на одній мікросхемі, на якій об'єднані усі його компоненти (процесор, вбудована пам'ять, пристрої вводу/виводу...) [2].

Одним із найпоширеніших є мікроконтролери сімейства STM32 [3]. Популярність даних мікроконтролерів забезпечується відносно невисокою ціною, багатим набором периферії, простотою експлуатації, низьким енергоспоживанням. Ці мікроконтролери використовуються у різних сферах,

					КС КРБ 123.154.00.00 ПЗ		
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>			
<i>Розроб.</i>		<i>Волощук І.В.</i>				<i>Літ.</i>	<i>Арк.</i>
<i>Керівник.</i>		<i>Стадник Н.Б.</i>					<i>Аркушів</i>
<i>Реценз.</i>					ТНТУ, каф. КС, гр. СІ-41		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>					
<i>Затверд.</i>		<i>Осухівська Г.М</i>					

особливо при проектуванні розумних міст та розумних будинків, а також у галузі охорони здоров'я [4].

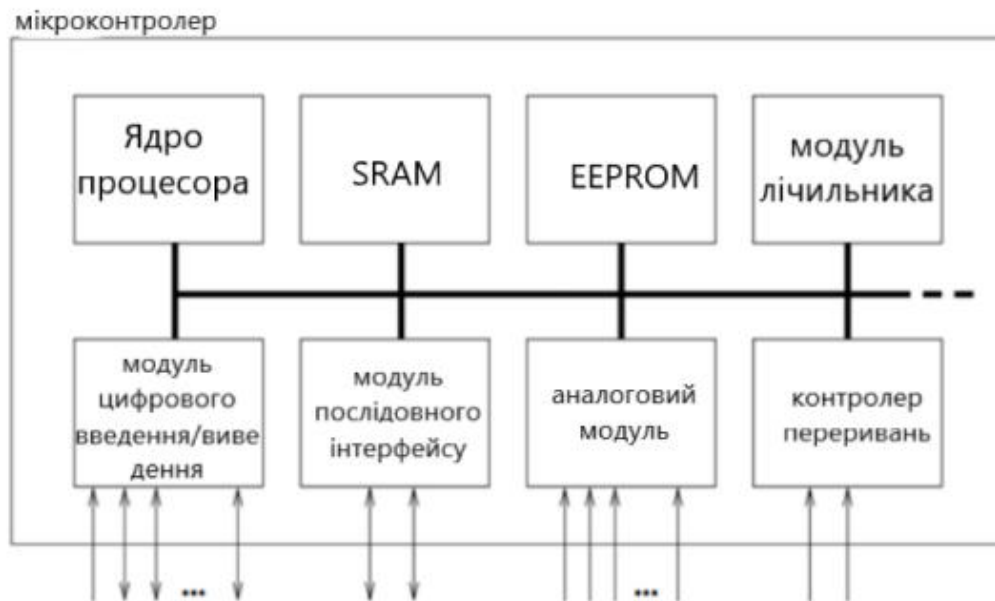


Рисунок 1.1 – Схема мікроконтролера

У роботі [5] повідомляється про розробку та впровадження вбудованої системи управління розумним будинком на базі мікропроцесора STM32F407VGT6 та операційної системи реального часу $\mu\text{C}/\text{OS-II}$. Докладно описано апаратну реалізацію та розробку ПЗ вбудованої системи управління розумним будинком. Пропонована апаратна архітектура складається з віддалених користувацьких прикладних терміналів, блоку управління терміналом та різних терміналів, розподілених у різних приміщеннях. Зв'язок між блоком керування терміналом та польовими терміналами здійснюється з використанням технології ZigBee (стандарт бездротового зв'язку). Цифрові датчики температури та вологості DHT11 використовуються для збору даних про температуру та вологість у різних приміщеннях у режимі реального часу. Для передачі інформації з датчиків та команд використовується дводіапазонний GSM/GPRS модуль SIM800A. Зазначається, що було виконано завдання дистанційного керування та отримання зворотного зв'язку від побутової техніки. Досягнуто віддалений моніторинг датчиків температури та вологості у

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		12

приміщенні, відтворення аудіо та відео та інші дії.

У роботі [6] представлена конструкція та схема реалізації домашнього бездротового шлюзу на основі мікроконтролера STM32FL03ZET6. Мікроконтролер STM32 через послідовний порт та модуль Wi-Fi був пов'язаний з модулем zigbee, розроблений набір протоколів зв'язку для кращої сумісності мереж Wi-Fi та zigbee. Пристрій здійснює контроль різноманітних датчиків та використовує камеру для централізованого керування побутовою технікою, моніторингу стану в режимі реального часу.

У роботі [7] обговорюються датчик диму та управління реле в розумному будинку. STM32 використовується як центральна частина всієї системи, а передачі даних застосовується протокол LwIP. Система використовує датчик диму MQ-2 для визначення концентрації горючих газів у повітрі та SRD-05VDC-SL-C для замикання та відключення ланцюга. У статті також обговорюється чутливість характеристики та вплив температури та вологості на датчик диму.

Датчики відіграють важливу роль і на транспорті, оскільки вже сьогодні очевидно, що транспортні засоби наступного покоління будуть не тільки сприймати навколишнє середовище за допомогою своїх датчиків, але також взаємодіяти з іншими транспортними засобами та інфраструктурою для забезпечення безпеки транспортних засобів та ефективності транспортування. Щоб реалізувати таке бачення, сучасні автомобільні електронні системи стають складнішими, ніж будь-коли, як з погляду електронних функцій, і з погляду архітектури. З функціональної точки зору існує широкий спектр нових програм, включаючи функціональні автономні та розширені системи допомоги водієві, як то адаптивний круїз-контроль та допомога при утриманні смуги руху. З метою втілення цих програм створені різні програми, які відіграють важливу роль у виявленні, обробці сигналів, управлінні, прийнятті рішень [8].

У роботі [9] продемонстровано підхід до розробки та реалізації контролера для двоколісного балансувального робота. Комбінація теорії умовного ПІД-регулятора та теорії управління з нечіткою логікою забезпечує потужний інструмент для проектування. Платформа реального часу Fuzzy-PID була

					<i>КС КРБ 123.154.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		13

реалізована на STM32F4 DISCOVERY KIT, моделювання та експериментальні результати показали переваги Fuzzy-PID Monitor.

У роботі [10] повідомляється про створення апаратної системи для контролю показників оточуючого середовища на основі мікроконтролера STM32. Система може визначати інтенсивність світла, температуру та вологість, а також концентрацію вуглекислого газу в приміщенні. Дані збираються обробляються мікроконтролером STM32 для відображення поточних значень параметрів кожної величини в приміщенні на екрані резистивному з діагоналлю 3,5 дюйма. Крім цього, реалізована можливість з персонального комп'ютера за допомогою хмарної платформи OneNET візуалізувати покази давачів.

У роботі [11] повідомляється про розробку бездротової системи визначення температури та вологості на основі мікроконтролера STM32 та модуля HC-05. Мікроконтролер STM32 використовується як ядро управління. Цифровий давач температури та вологості AM2303 з низьким енергоспоживанням та швидким відгуком використовується для отримання даних про температуру та вологість. Технологія Bluetooth використовується для віддаленого зв'язку з комп'ютером.

У роботі [12] повідомляється про розробку алгоритму прогнозування для керування тепловою системою у реальному часі з використанням високопродуктивного процесора STM32. Результати практичного використання показали, що використання мікроконтролера в тепловій системі показує кращі характеристики порівняно з ПД-регулятором.

Однією з областей, де потрібний суворий контроль температурного режиму є виноробство. Температура бродіння є визначальним чинником якості рисового вина, тому автору роботи [13] була потрібна система контролю температури в кімнаті бродіння. Система була побудована на базі мікроконтролера STM32 та цифрового давача температури. Спочатку докладно описується вся структурна схема системи, а потім описуються модуль збору температури і модуль контролю температури. Нарешті обговорюється блок-схема ПЗ. Результат показує, що розроблена система може здійснювати

					КС КРБ 123.154.00.00 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

високоточний контроль за температурою.

У роботі [14] описується реалізація швидкого перетворення Фур'є Radix-2 для STM32F746ZG мікроконтролера та вимірювання його точності. Вимірний частотний спектр порівнюється з професійним аналізатором спектру системи BitScore. У статті наводиться порівняння швидкості між швидким і дискретним перетвореннями Фур'є. Зроблено висновок про те, що розроблений пристрій видає такі ж гармоніки, як професійний спектральний аналізатор.

Основним недоліком сучасних п'єзорезистивних давачів тиску є падіння вихідної напруги із збільшенням робочої температури, що сильно знижує точність вимірів. У статті [15] подано новий пасивний метод температурної компенсації кремнієвих п'єзорезистивних давачів тиску. Вбудована технологія компенсації усуває необхідність дорогого та трудомісткого процесу калібрування, необхідного для кожного давача всередині партії. У цьому методі з метою компенсації використовуються додаткові полікремнієві резистори із від'ємним температурним параметром питомого опору. Внаслідок застосування цього методу температурний коефіцієнт чутливості звичайного некомпенсованого давача було зведено до нуля. Компенсований давач є вигідним для тиску вимірювання в суворих умовах з коливаннями температури.

У статті [16] повідомляється про розробку системи швидкого та високоточного контролю теплового режиму із застосуванням ПД-алгоритму керування з STM32 як мікроконтролер. Така система контролю може бути застосованою у тих областях, котрі ставлять високі запити щодо швидкодії і точності вимірювання власне температури. Властиво ланцюг вимірювання температури у системі використовує давач температури Pt000. Цією схемою збору даних сигнал температури здатен перетворитися на сигнал напруги і переданий на мікроконтролер. Фотоелемент TLP521-1 був узгоджений із силовим транзистором марки BD237 для управління термоелектричним охолоджувачем у компоненті FTA951. Варто зауважити, що ефективна електрична потужність охолоджувача регулювалася сигналами широтно - імпульсної модуляції (ШІМ), які генерувалися мікроконтролером.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

Характеристики сигналу ШІМ можуть бути налаштовані із використанням ПІД-регулятора. Підсумки проведеного експерименту свідчать, що система контролю температури має просту структуру, високу швидкістю відклику, адекватну стабільність і високу точністю контролю температури (за похибки $< \pm 0,5 \text{ }^\circ\text{C}$)

Profibus – це шина польового рівня німецького національного стандарту DIN19245 та європейського стандарту EN50170, це одна з найпопулярніших технологій польової шини, а також єдиний стандарт польової шини GB/T20540-2006, схвалений у Китаї. Він широко використовується в промисловості, електроенергетиці, енергетиці, транспорті та інших автоматизованих областях. Відповідно до характеристик своїх застосувань Profibus ділиться на три типи: Profibus – DP, Profibus-FMS та Profibus-PA. Profibus-DP зручний для високошвидкісного зв'язку між децентралізованими периферійними пристроями, які стають все більш популярними серед пристроїв автоматизації та інтелектуальних вимірювальних приладів. Найпоширенішим якийсь час тому був протокол Modbus з інтерфейсом RS485.

На межі переходу від Modbus до Profibus з'явилася робота [17], що повідомляє про розробку на базі мікроконтролера STM32 у поєднанні з комунікаційною мікросхемою POWERP3+C, загального лінійного мосту Modbus-Profibus. При цьому POWERP3+C реалізовано інтерфейсні протоколи Profibus-DP, а STM32 реалізує протокол зв'язку Modbus і загальну роботу лінійного мосту. Тестування показує, що шинний міст Modbus-Profibus може реалізувати функцію зв'язку Profibus з польовими пристроями та інтелектуальними приладами, забезпечуючи таким чином комунікаційні рішення Profibus з коротким циклом розробки, низькою вартістю та високою ефективністю.

Робота [18] звертає увагу на те, що в міру вдосконалення системи збору даних з датчиків орієнтації дуже важливо вивчати похибку гіроскопа. Помилки системи включають систематичну та випадкову помилку, а випадкова помилка є основним джерелом помилок системи. Алгоритм фільтра Калмана використовується для усунення випадкового шуму в помилці гіроскопа та

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

підвищення точності вимірювання. У статті розглядається конструкція платформи збору даних про орієнтацію на базі STM32, обговорюється застосування фільтра Калмана у системі збору даних та на основі технології цифрової обробки сигналів реалізовано алгоритм фільтра Калмана.

У роботі [19] повідомляється про створення на базі мікроконтролерів STM32 обладнання для освітніх цілей для підготовки фахівців у галузі обробки даних. Паралельна обробка даних – це швидке маніпулювання даними. Вхідні дані обробляються декількома пристроями через швидкість та розподіл завдань для окремих пристроїв. У цій статті описується використання паралельної обробки вхідних значень двома мікроконтролерами STM32F446RE та мікроконтролером STM32F746ZG, який є ведучим пристроєм. Інші мікроконтролери є підлеглими пристроями. Цей пристрій обробляє логічний стан уведення з 32 кнопок введення, що складає 32-розрядне слово. Існує архітектура master / slave, в якій підлеглі пристрої сканують PIN -коди та відправляють статус на головний пристрій. Основною перевагою паралельної обробки є поділ завдань між двома або більше процесорами, а це означає, що кожен процесор може працювати незалежно з меншою кількістю операцій. Це дозволяє не перевантажувати процесор, як у випадку, якщо працює лише один процесор.

1.2 Контролери температури

Об'єктом дослідження кваліфікаційної роботи є електронна система, робота якої залежить від температури. Причому ця залежність може бути неочевидною для виробника обладнання і кінцевому споживачеві доводиться самостійно реалізовувати цей функціонал.

Розглянемо типові пропозиції таких контролерів.

Найпростішим із знайдених автором є контролер температури W1209. У ньому є механічне реле для керування навантаженням 12 В, три кнопки для налаштування порогів ввімкнення та вимкнення навантаження.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

Цифровий терморегулятор W3230 випускається в кількох модифікаціях для керування ланцюгами постійного та паралельного струмів.

Цифровий регулятор температури STC-1000 призначений для організації підтримки заданої температури та має висновки для підключення нагрівального елемента та охолоджувача.

Всі перелічені пристрої використовують як давач температури NTC терморезистор 10 кОм і мають реле для управління навантаженням. Вони є готовими пристроями, що не зважають на втручання користувача в код, що виконується і не допускають розширення функціоналу пристрою.

У продажу наявний значний діапазон програмованих реле часу. Їх функціонал досить примітивний, але більшість з них мають гальванічну розв'язку управляючих та керованих ланцюгів. Будь-який із цих модулів можна використовувати як керований створений під час підготовки даної роботи контролером.

На основі викладеного вище було прийнято рішення виключити реле з пристрою, що розробляється для підвищення його універсальності: важливо дозволити користувачеві гнучко налаштувати алгоритм управління, а яким саме ланцюгом управляти - користувач вирішить сам, підключивши на вихід розробленого пристрою.

Оскільки аналогічний пристрій, що розробляється, серед комерційно доступних, знайти не вдалося, складно визначити верхню межу допустимої вартості пристрою. Крім цього, оскільки за час розробки пристрою вартість ряду компонентів збільшилася (наприклад, мікроконтролери STM32), оцінки вартості готового пристрою на початку дослідження (вересень 2025 року) та наприкінці (травень 2026 року) дещо різняться.

Пристрій, що розробляється, можна назвати вбудовуваною системою - спеціалізованою мікропроцесорною системою, на яку покладаються завдання контролю та моніторингу, яка вбудовується в більш складні системи.

Для забезпечення можливості створення модифікацій пристрою без зміни друкованої плати було вирішено додати підтримку двох давачів температури:

					КС КРБ 123.154.00.00 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

TMP36 та цифрового DS18P20. Крім цього, передбачений універсальний роз'єм для під'єднання датчиків I²C (наприклад, BMP280). Пристрій повинен штатно функціонувати за наявності будь-якого під'єданого датчика температури. За наявності кількох датчиків користувач повинен мати можливість вибору використовуваного.

На підставі викладеного вище сформулюємо технічне завдання пристрою, що розробляється. Технічні вимоги, що пред'являються до пристрою, вказані в таблиці 1.1.

Таблиця 1.1 – Технічні вимоги до пристрою

Характеристика	Значення / величина
Дисплей	Монохромний, 128x96 пікселів
Габарити (ДхВхШ) в мм, не більше	70x45x20
Напруга живлення, В	5
Роз'єм живлення	microUSB
Функціонал	<ul style="list-style-type: none"> - можливість управління двома лініями живлення незалежно від алгоритмів, що налаштовуються; - можливість під'єднання додаткових датчиків за інтерфейсом I²C; - відображення на вбудованому дисплеї актуальної інформації про відстежувану фізичну величину та статистичну інформацію; - внесені користувачем зміни до налаштувань пристрою повинні зберігатися в незалежній пам'яті; - на пристрої повинні бути виводи інтерфейсу USART для налагодження та програмування пристрою; - пристрій повинен мати світлову індикацію живлення.

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Узагальнена структура системи

Спроектована узагальнена структура вбудованої системи контролю температури для фарбувально-сушильної камери продемонстрована на рис. 2.1. Взаємодія користувача і розроблюваної системи організована за схемою відправлення запитів до системи при допомозі одного із ймовірних варіантів під'єднання. Сам мікроконтролер обробляє запит, виконуючи дії відповідно до розробленого на основі протоколів I²C та USART набору визначених правил: одержання даних із давача температури, записування показника, надсилання відповіді та відображення результату.



Рисунок 2.1 – Узагальнена структура розроблюваної системи

2.2 Вибір мікроконтролера

Мікроконтролерів з апаратною підтримкою інтерфейсу OneWire не так багато (наприклад, ARM Cortex-M4 мікроконтролери MAX32620/MAX32621).

					КС КРБ 123.154.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Волощук І.В.				Літ.	Арк.
Керівник.		Стадник Н.Б.					Аркушів
Реценз.					ТНТУ, каф. КС, гр. СІ-41		
Н. Контр.		Луцик Н.С.					
Затверд.		Осухівська Г.М					

Зазвичай, цей протокол доводиться реалізовувати програмно. Особливістю протоколу є необхідність досить точного підрахунку проміжків мікросекундної часу тривалості. Для цього може знадобитися окремий таймер або модуль USART.

Під'єднання дисплеїв можливе за інтерфейсами I²C або SPI. Другий варіант дозволяє досягати більших частот оновлення дисплея. До того ж інтерфейс I²C передбачається використовувати для під'єднання додаткових датчиків і, можливо, EEPROM пам'яті. Варіант під'єднання дисплея за паралельним інтерфейсом автору здається недоцільним через ускладнення друкованої плати та додаткові складнощі з програмною реалізацією.

Для підрахунку часових інтервалів відображення інформації на дисплеї та опитування датчиків потрібен ще один таймер.

Для можливості програмування алгоритмів роботи пристрою, котрі залежать від часу доби, знадобиться модуль годинника реального часу.

Для реалізації навігації за налаштуваннями пристрою вирішено використовувати чотири кнопки, під'єднані до виводів мікроконтролера з підтримкою зовнішніх переривань.

Таким чином, мікроконтролер повинен володіти двома таймерами, модулями аналого-цифрового перетворювача, USART, I²C, SPI, годинником реального часу, 4 незалежні лінії зовнішніх переривань. Всім цим вимогам задовольняє бюджетний мікроконтролер STM32F103C8T6, який зображений на рис. 2.2.

Символи у назві означають таке.

- STM32 (платформа STMicroelectronics);
- F1 (родина пристроїв);
- 03 (підрозділ сімейства пристроїв);
- C8T6 (фізичний прояв, що впливає на кількість SRAM, флеш-пам'яті тощо).

					КС КРБ 123.154.00.00 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

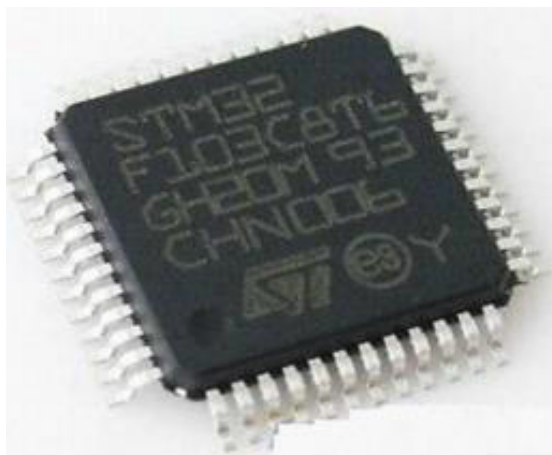


Рисунок 2.2 – Зовнішній вигляд мікроконтролера STM32F103C8T6

На рис. 2.3 наведена внутрішня будова типового STM32 на базі ядра Cortex -M4.

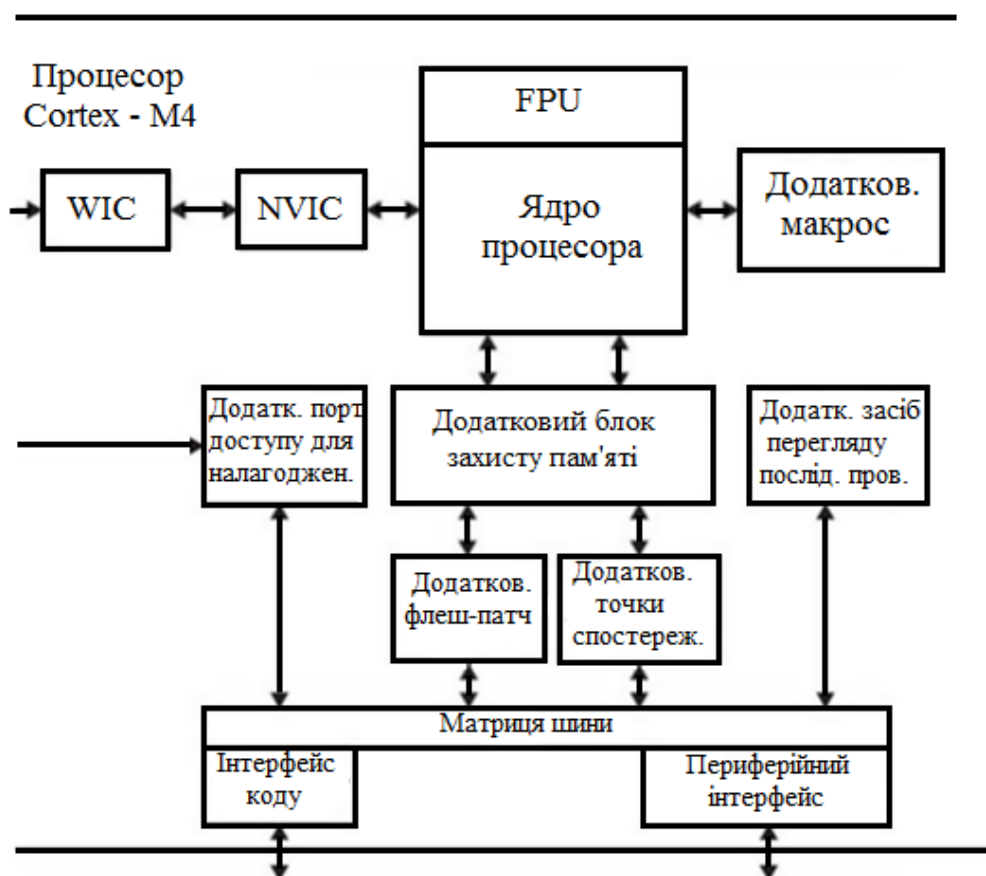


Рисунок 2.3 – Внутрішня схема типового STM32 на основі ядра Cortex-M4

Для програмування мікроконтролерів STM32F1 потрібні ще дві кнопки,

під'єднані до виводів без підтримки переривань.

Цей мікроконтролер не має вбудованої пам'яті EEPROM, але можна під'єднувати зовнішній модуль за інтерфейсом I²C. Про під'єднанні кількох пристроїв до цієї шини докладно написано в різній літературі.

Мікроконтролер STM32F103C8T6 відрізняється від інших мікроконтролерів тієї ж серії високою доступністю на платах з наявною периферією, яка потрібна на роботу мікроконтролера (плата Blue Pill). Це значно спрощує створення попередньої версії прототипу пристрою.

Існує множина мікроконтролерів STM32. Їхні останні модифікації можуть працювати на частотах понад 200 МГц. Для цієї кваліфікаційної роботи така міць не потрібна.

Живлення мікроконтролера можливе від джерела з напругою від 2,0 до 3,6В. Режими енергозбереження дозволяють розробляти рішення з низьким споживанням енергії.

Мікроконтролери серії STM32F103xx містять контролер вкладених векторних переривань, здатний обробляти до 43 маскуваних каналів переривань (крім 16 ліній переривання CortexM3) і 16 рівнів пріоритету. Адреса таблиці векторів переривань передається у ядро. Завдяки цьому досягається мала затримка до обробки переривань і обробка пізніших переривань з більш високим пріоритетом. Стан процесора зберігається автоматично. Стан процесора відновлюється при перериванні без додаткових інструкцій.

Контролер зовнішніх переривань / подій складається з 19 ліній детекторів фронту, що використовуються для створення запитів на переривання / подія. Кожну лінію можна налаштувати незалежно для вибору тригера події (наростаючий фронт, спад, обидва) і можуть маскуватися незалежно.

Вибір джерела тактових імпульсів виконується під час запуску. По дефолту при скиданні вибирається внутрішній RC-генератор 8 МГц. Можна вибрати зовнішній резонатор номіналом 4-16 МГц. Вони відстежуються щодо відмови, у якому він відключається і відбувається виклик переривання.

Під час запуску контакти використовуються для вибору одного з трьох

					КС КРБ 123.154.00.00 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

варіантів завантаження.

- із флеш-пам'яті;
- із системної пам'яті;
- із SRAM.

Завантажувач знаходиться у системній пам'яті. Він використовується для перепрограмування флеш-пам'яті за допомогою USART.

Мікроконтролери серії STM32F103xx підтримують три режими низького енергоспоживання для досягнення найкращого компромісу між низьким енергоспоживанням, коротким часом запуску та можливістю пробудження:

- режим сну. У сплячому режимі зупиняється лише ЦП. Усі периферійні пристрої продовжують працювати і можуть розбудити ЦП у разі переривання / події;

- режим зупинки. Дозволяє досягти мінімального енергоспоживання при збереженні вмісту SRAM та регістрів. Пристрій можна вивести з цього режиму за будь-якою лінією EXTI (одна з 16 зовнішніх ліній, вихід PVD, аварійний сигнал RTC або USB);

- режим очікування. Дає змогу досягти мінімального енергоспоживання. Після переходу в режим очікування режим SRAM і регістрів втрачається, за винятком регістрів в домені Backup.

Для запису файлів, що виконуються, в пам'ять мікроконтролера необхідний програматор. Найбільш доступним рішенням є клон ST-LINK V2, зображений на рис. 2.4. ST-LINK V2



Рисунок 2.4 – Програматор ST-LINK V2

					КС КРБ 123.154.00.00 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

Програматор під'єднується до USB -порту комп'ютера. До налагоджувальної плати програматор під'єднується за допомогою чотирьох проводів (рис. 2.5). Програматор дозволяє працювати з популярними середовищами розробки STM32CubeIDE, IAR, Keil та програмною для запису виконуваних файлів ST-Link Utility.



Рисунок 2.5 – Під'єднання програматора до налагоджувальної плати

2.3 Вибір датчиків температури

Станом на даний час використовуються шість типів датчиків температури. Термопара – це датчик температури найпоширенішого типу, скорочено ТС. Його можна використати на великих відстанях. На ринку є різні типи датчиків типу термопар для різних відстаней. Датчик температури термопарного типу показано рис. 2.6.



Рисунок 2.6 – Термопари

Давач температури опору (рис. 2.7) працює за принципом зміни опору при зміні температури. Опір прямо пропорційний температурі і має широкий діапазон можливостей вимірювання температури. Це більш точний давач температури, діапазон значень від -270 до 850 °С.



Рисунок 2.7 – Температурний давач опору

Термістори - це третій тип датдавачачика температури, який дешевший у порівнянні з іншими типами і дуже простий у використанні. Принцип роботи давача температури термісторного типу такий самий, як у давача попереднього типу: зміна опору при зміні температури. Вони виготовлені з нікелю або будь-якого матеріалу керамічного типу. Давач температури термісторного типу показано на рис. 2.8.



Рисунок 2.8 – Давач температури термісторного типу

Термометр - це пристрій, який використовує розширення тіл під час нагрівання для визначення температури.

ІЧ-давачі (інфрачервоні давачі) використовуються для вимірювання температури поверхні в діапазоні від -70 до 10 000 ° С. Вони перетворюють

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		26

адаптований до вимог, що стосуються точності, часу вимірювання та енергоспоживання, завдяки можливості вибору із значної кількості ймовірних налаштувань давача..

Основні властивості та особливості давача:

- 8 вивідний LGA корпус 2,0 мм x 2,5 мм x 0,95 мм;
- цифрові інтерфейси: I²C (до частоти 3,4 МГц) та SPI (до частоти 10 МГц);
- напруга живлення: від 1,71 до 3,6;
- споживаний струм: 2,7 мкА;
- оснащений вбудованим фільтром для мінімізації похибки вимірних величин, коефіцієнт фільтрації даного фільтра можна вибрати в діапазоні від 0 (фільтр вимкнено) до 16;
 - робочий діапазон: температура: -40 ..,+85 ° С; тиск: 300 ... 1100 гПа (225 - 800 мм рт. ст.);
 - точність вимірів: тиск: 0,01 гПа; температура: 0,01°С.

Вимірювання тиску та температури може бути увімкнено або вимкнено. Пропуск вимірювання тиску може бути корисним, якщо BMP280 використовується як давач температури. Аналогічно, вимірювання температури не можна виконувати, якщо давач використовується тільки для вимірювання тиску.

Цифровий давач температури DS18B20 застосовується для вимірювання температури навколишнього середовища в діапазоні -55°С до +125°. Точність даного давача становить $\pm 0,5^{\circ}\text{З}$ вимірі від -10 до +85 °С. Давач є програмованим з роздільною здатністю від 9 - 12 біт, що відповідає крокам 0,5 °С, 0,25 °С, 0,125 °С та 0,0625 °С. За замовчуванням при першому увімкненні давача роздільна здатність становить 12 біт.

Мікросхема DS18B20 застосовує інтерфейс 1-Wire, що дає змогу під'єднувати декілька давачів до однієї шини (до одного виводу мікроконтролера) та зчитувати інформацію з кожного давача окремо. Це досягається за допомогою того, що кожен давач має свій 64-розрядний код, що

					КС КРБ 123.154.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

зберігається в давачі ПЗП.

При проектуванні електронних пристроїв із давачем DS18B20 слід мати на увазі, що існує кілька версій давача. Їхні порівняльні характеристики показані в табл. 2.1.

Таблиця 2.1 – Порівняльні характеристики різних версій давача DS18B20

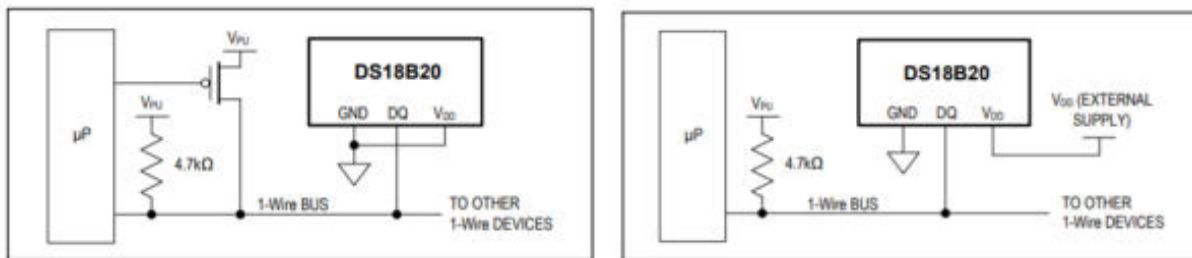
Версія давача	DS1820	DS18S20	DS18B20
Корпус	PR-35 (подовжений TO-92)	TO-92	TO-92
Розрядність	9 біт	9 біт	9 - 12 біт
Час перетворення, мс	200 (тип.) 500 (макс.)	750 (макс.)	750 (макс.)
Точність вимірювання температур $\pm 0.5\%$	0+70 °C	-10 +85 °C	-10 ... +85 °C
Напруга живлення	4,3 - 5,5 В	3,3 - 5,5 В	3,3 - 5,5 В

Мікросхему DS1820 знято з виробництва, для її заміни рекомендується мікросхема DS18S20. Проте, слід звернути увагу, що мікросхеми DS18S20 в корпусі TO-92 маркуються написом "DS1820" (без літери S). Нова мікросхема DS18S20 програмно сумісна зі старою DS1820.

Нова мікросхема DS18S20 виконана у стандартному корпусі TO-92, а стара DS1820 мала подовжений корпус. За цією ознакою можна відрізнити ці дві версії давача. Мікросхема DS18B20 завжди має відповідне маркування "DS18B20" і не може бути замінена на DS1820/DS18S20 та назад без зміни програмного коду.

DS18B20 може бути під'єднаний до зовнішнього джерела живлення з напругою від 3,3 - 5,5, або працювати в режимі «паразитного живлення». Паразитне живлення корисне при дистанційному вимірюванні температури та роботі давача в обмежених просторах. На рис. 2.10 показано схеми керування живленням давача DS18B20.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		



а)

б)

Рис. 2.10 – Способи під'єднання датчика DS18B20: а) паразитне живлення, б) живлення від зовнішнього джерела

DS18B20 під'єднується через вивід DQ до шини 1-Wire. При паразитному живленні датчик отримує живлення від мікроконтролера поки що на шині високий рівень, а конденсатор μP накопичує заряд. Коли на шині низький рівень, датчик отримує живлення від конденсатора. Використання такого способу живлення DS18B20 не рекомендується при температурах вище $+100\text{ }^\circ\text{C}$, оскільки датчик не здатний підтримувати зв'язок із шиною через високі струми витoku.

DS18B20 може житися від звичайного джерела живлення від 3,3 - 5 В. Перевага цього методу полягає в тому, що підтяжка MOSFET не потрібна, і шина 1-Wire може вільно отримувати дані з інших під'єднаних до неї датчиків під час перетворення температури.

TMP36 - це низьковольтний прецизійний датчик температури (рис. 2.11), вихідна напруга якого прямо пропорційна температурі за шкалою Цельсія.



Рисунок 2.11 – Датчик температури TMP36

TMP36 не вимагає будь-якого зовнішнього калібрування для забезпечення

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		30

типової точності $\pm 1^\circ\text{C}$ при $+25^\circ\text{C}$ і $\pm 2^\circ\text{C}$ в діапазоні температур від -40°C до $+125^\circ\text{C}$. TMP36 призначений для роботи від джерела живлення від 2,7 В до 5,5 В. Струм живлення значно нижче $0,1^\circ\text{C}$ у нерухомому повітрі. Крім того, передбачена функція відключення, що дозволяє знизити струм живлення до менш ніж 0,5 мкА.

2.4 Дисплей

Широкого поширення набули рідкокристалічні дисплеї з контролером SD1306. Існують модифікації для під'єднання до шин I²C та SPI. SD1306 - однокристальний драйвер CMOS OLED / PLED з контролером для випромінювання світла з органічних/полімерних матеріалів. На рис. 2.12 наведено фотографію модуля дисплея з 128x64 пікселів.



Рис. 2.12 – Зображення рідкокристалічного дисплея

SSD1306 вмикає контроль контрастності, оперативну пам'ять дисплея і осцилятор, що зменшує кількість зовнішніх компонентів і енергоспоживання. Він має 256-ступінчасте регулювання яскравості. Дані / команди відправляється з мікроконтролера через апаратно обраний послідовний периферійний інтерфейс. Цей дисплей підходить для багатьох компактних портативних програм.

OLED -дисплей працює без підсвічування, тому що він випромінює власне

					КС КРБ 123.154.00.00 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

світло. Завдяки цьому він володіє такою високою контрастністю, дуже широким кутом огляду та здатний відтворювати глибокі рівні чорного. Власне відсутність підсвічування суттєво зменшує енергоспоживання, котре потрібне для його функціонування. У середньому дисплей споживає струм близько 20 мА, хоча залежить від того, яка частина дисплея горить.

Безвідносно до розміру OLED модуля драйвер SSD1306 володіє вбудованою ОЗП, для відображуваних графічних даних (Graphic Display Data RAM, GDDRAM) об'ємом 1 КБ для екрану, який містить бітовий малюнок, що відображається. Ця область пам'яті 1К організована на 8 сторінок (від 0 до 7). Будь-яка сторінка має 128 стовпців/сегментів (від 0 до 127). І будь-який стовпець здатен зберігати 8 біт даних (від 0 до 7).

Кожен виробник випускає власні друковані плати з різним компонуванням електронних компонентів та інтерфейсом, оскільки контролер SSD1306 підтримує відразу три робочі протоколи:

- 8-бітний паралельний інтерфейс серії 6800/8080;
- 3/4-провідний послідовний периферійний інтерфейс;
- I²C .

Для зміни протоколу роботи є три виводи контролера (BS0, BS1 та BS2), за допомогою яких контролер дисплея визначає, з яким протоколом він працюватиме. Логічні рівні цих виводів, що відповідають кожному режиму роботи, наведені в табл. 2.2.

Таблиця 2.2 – Вибір інтерфейсу шини

Вивід SSD1306	Інтерфейс				
	I ² C	6800 (8 біт)	8080 (8 біт)	4-провідний SPI	3-провідний SPI
BS0	0	0	0	0	1
BS1	1	0	1	0	0
BS2	0	1	1	0	0

Для того, щоб дані відображалися на екрані, необхідно записати 1 та 0 у ОЗП графічних даних, яке представляє зображення. Пам'ять складається з 8 сторінок, кожна з яких має ширину 128 біт та висоту 8 біт. Це означає, що при записі 0b0101010 в позицію (0,0) будуть встановлені перші 8 пікселів у стовпці на екрані, які будуть увімкнені, вимкнені, увімкнені, вимкнені, увімкнені, вимкнені, увімкнені, вимкнені.

2.5 Принципова схема

Принципова схема пристрою наведено на рис. 2.13. Передбачається, що контролер, що розробляється, і керовані ним реле матимуть спільне джерело живлення. З цієї причини на затискні контакти не виведено шину GND (загальний провід). У разі потреби у під'єднанні до шини GND контролера може бути здійснено через виведений на плату, але такий, що не використовується вивід GND. На це рішення довелося піти через обмежені розміри корпусу пристрою.

Живлення контролера здійснюється через роз'єм шіроUSB (5 В) через стабілізатор напруги 3,3 В та діод. Таким чином напруга живлення контролера буде трохи більшою, ніж 3 В. Таким самим буде і логічний рівень керуючих вихідних контактів.

Крім цього є резервний (виведений на платі, але не використовується в готовому пристрої) вивід живлення з діодом. Цей роз'єм призначений для тестових цілей і має використовуватися з великою обережністю. Високий рівень напруги (більше 3,3) на цьому вході може вивести мікроконтролер з ладу. Опитати контролер можна і через виведення інтерфейсу I²C. Цей спосіб є ризикованим, оскільки цей вивід призначений для під'єднання зовнішнього навантаження, а не для введення живлення. Жодного захисту там немає.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Налагодження компонентів на платах (макетній та друкованій)

Перевірку працездатності основних вузлів електричної схеми було проведено з використанням налагоджувальної плати BlackPill безіменного китайського виробника. Зібрану схему зображено на рис.3.1.

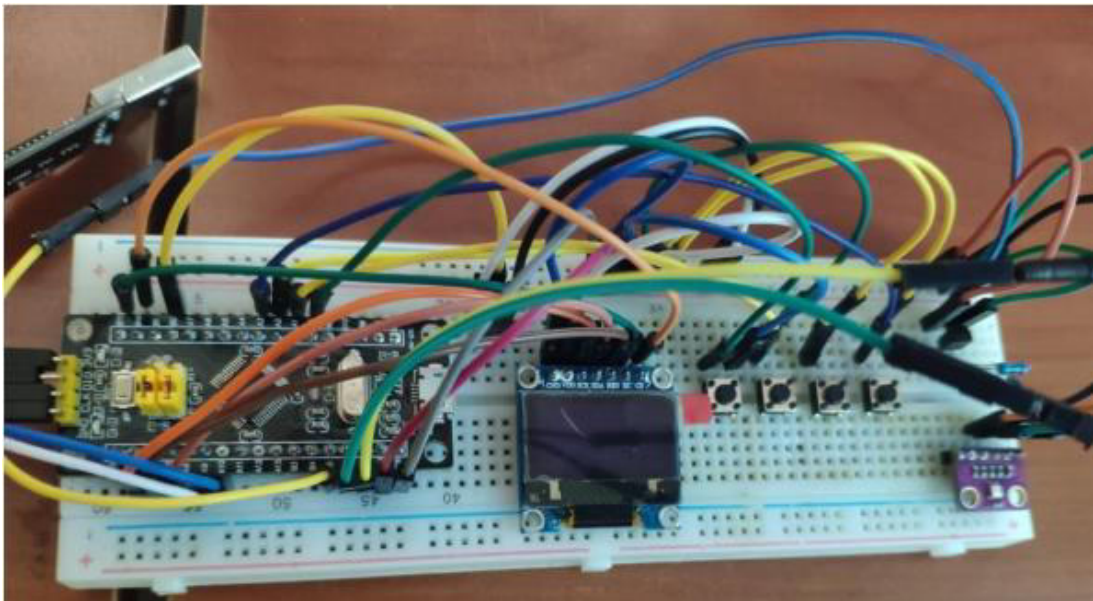


Рисунок 3.1 – Прототип пристрою на безпайковій макетній платі

Після перевірки працездатності схеми зібрано прототип пристрою на макетній платі, зображений на рис. 2.15. Цей прототип дозволив позбутися неприємних проводів і домогтися більш якісних електричних контактів. Більшість тестових запусків у ході розробки ПЗ проходила на цьому прототипі.

					КС КРБ 123.154.00.00 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Волощук І.В.				Літ.	Арк.
Керівник.		Стадник Н.Б.					Аркушів
Реценз.					ТНТУ, каф. КС, гр. СІ-41		
Н. Контр.		Луцик Н.С.					
Затверд.		Осухівська Г.М					

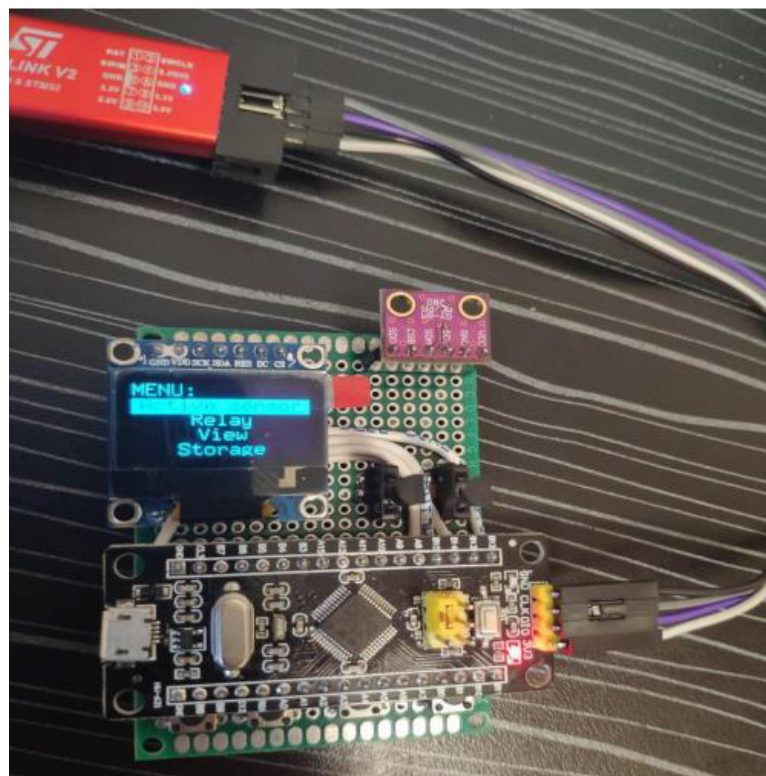


Рисунок 3.2 – Прототип пристрою на макетній платі

Для виготовлення прототипу пристрою у середовищі EaseEDA було спроектовано друковану плату (рис. 3.3). Плата була виготовлена компанією jлрсб.

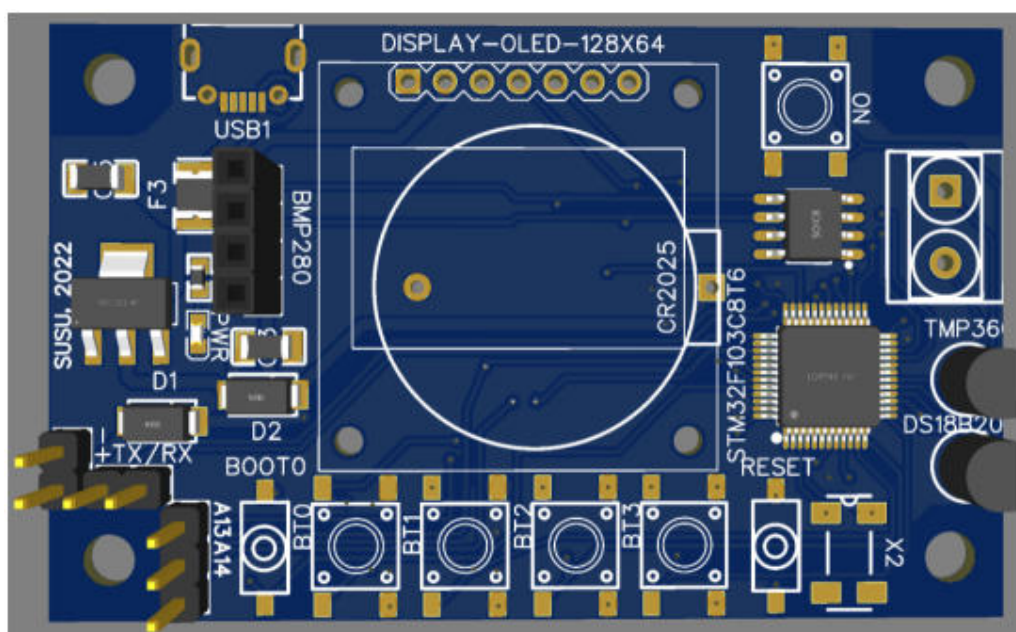


Рисунок 3.3 – Друкована плата (модель)

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		36

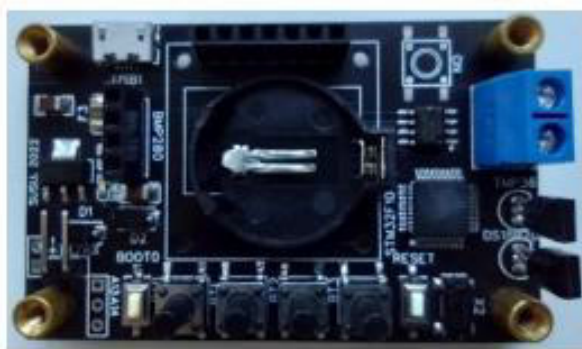
Автором роботи проведено монтаж компонентів. У комерційно доступному пластиковому корпусі було зроблено необхідні отвори. Готовий пристрій зображено на рис. 3.4. На рис.3.4 б) показаний головний екран зі показами температури та індикацією активності контрольованих каналів. Здійснюється навігація по меню та налаштування пристрою. На рис.3.4 в) і г) показана друкована плата (вид із двох сторін). Видно додаткові кнопки, необхідні для запису виконуваного файлу в пам'ять мікроконтролера і виводи інтерфейсу USART, призначені для цієї мети.



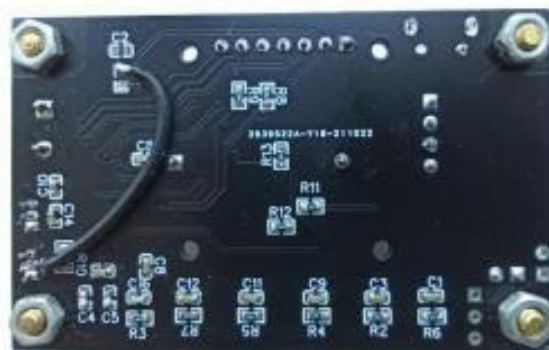
а



б



в



г

Рисунок 3.4 – Готовий пристрій

У процесі тестування було виявлено невеликі помилки у розведенні плати, виправити які не вдалося через труднощі з логістикою та оплатою на сайтах іноземних компаній. Проте вдалося досягти працездатності плати завдяки внесенню невеликих змін (провід на рис. 3.4 г) і відсутня кнопка на рис. 3.4 в).

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		37

3.2 Розробка алгоритму роботи

Перед написанням програмного коду необхідно розробити алгоритм роботи. Будь-який електронний пристрій чи система на мікроконтролері функціонує за визначеним алгоритмом, котрий проходить від старту до фінішу програмного коду. Перш за все необхідно продумати зразковий алгоритм функціонування, котрий врешті рещт призведе до досягнення необхідної мети. У подальшому, базуючись на цьому алгоритмі, потрібно написати код програми, який «перекладе» кроки алгоритму на мову, яка буде зрозумілою для мікроконтролера.

На рис. 3.5 наведена загальна блок-схема алгоритму.

Робота системи починається з оголошення змінних, ініціалізації давачів та інших складових елементів. Після чого система повинна перевірити, чи натиснута кнопка визначення температурного режиму. Якщо це так, тоді проходить процес опитування контролера, надсилання даних для перевірки температури, потім проходить обробка даних та результат виводиться на рідкокристалічний дисплей. Якщо ж результатом перевірки умови є «фальш», тоді відбувається перехід до режиму встановлення температури, проходить перевірка та обробка даних, після чого результат відображається на рідкокристалічному дисплеї

При розробленні алгоритму приймаються рішення, які саме функції будуть застосовані у програмному коді.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

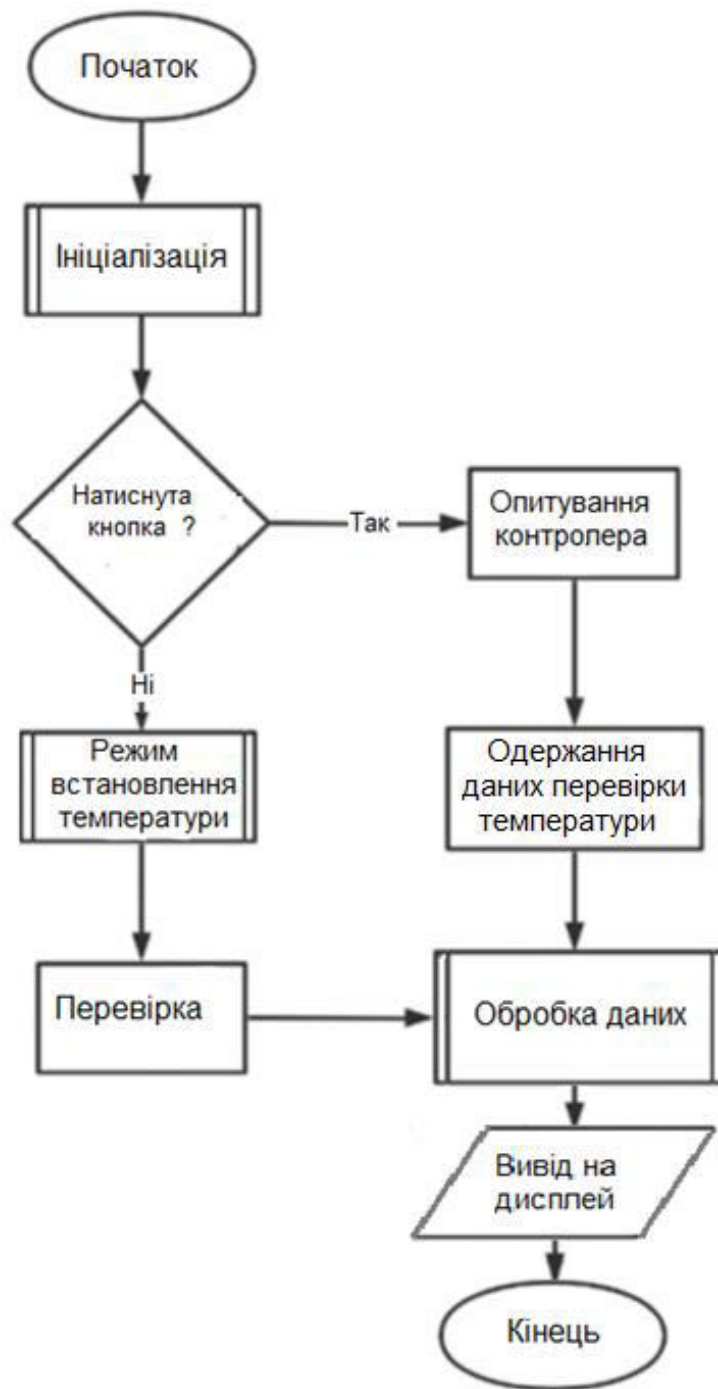


Рисунок 3.5 – Загальна блок-схема алгоритму

3.3 Програмна складова

3.3.1 Розробка ПЗ

Розробку програмного коду для мікроконтролерів STM32 можна вести з використанням низки середовищ розробки: IAR Embedded Workbench, Keil uVision, STM32CubeIDE, Arduino IDE, модулі розширення для Code::Blocks,

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		39

Visual Studio Code та Eclipse. Для реалізації програмної частини кваліфікаційної роботи було обрано середовище STM32CubeIDE, яке має одночасно низку позитивних можливостей:

- кросплатформне;
- безкоштовне;
- професійне середовище розробки;
- не потребує налаштування;
- розробляється виробником мікроконтролерів STM32.

STM32CubeIDE є універсальним інструментом для розроблення у декількох ОС. Він є складовим елементом програмної системи STM32Cube. Цей IDE призначено для розроблення коду мовами C / C++ із функціями конфігурації периферії, генерації коду, компіляції коду та налагодження для мікроконтролерів STM32. Базується на середовищі Eclipse® / CDT.

STM32CubeIDE успішно консолідує функції конфігурування STM32 і побудови проектів із STM32CubeMX для того, щоб надати універсальний засіб і заощадити час на інсталяцію та розроблення. Здійснивши вибір STM32 мікроконтролера чи завчасно сконфігурованого мікроконтролера, формується проект та створюється код ініціалізації. Будь-коли упродовж розробки користувач має можливість повернутися до ініціалізації і налагодження периферії чи проміжного ПЗ і репродукувати ініціалізаційний код, не створюючи впливу на код користувача.

STM32CubeIDE має аналізатори формування та стеку, котрі надають юзеру помічну інформацію щодо стану проекту та властиво вимог до пам'яті.

STM32CubeIDE на додачу володіє стандартними і розширеними функціями налаштування, як то переглядання регістрів ядра процесора, пам'яті та периферії. Крім того наявне миттєве відстеження змінних, інтерфейс Serial Wire Viewer чи аналізатор визначених несправностей.

STM32CubeIDE підтримується 64-розрядними версіями операційних систем Windows®, Linux® та macOS®. На рис. 3.6 та 3.7 показаний графічний інтерфейс.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

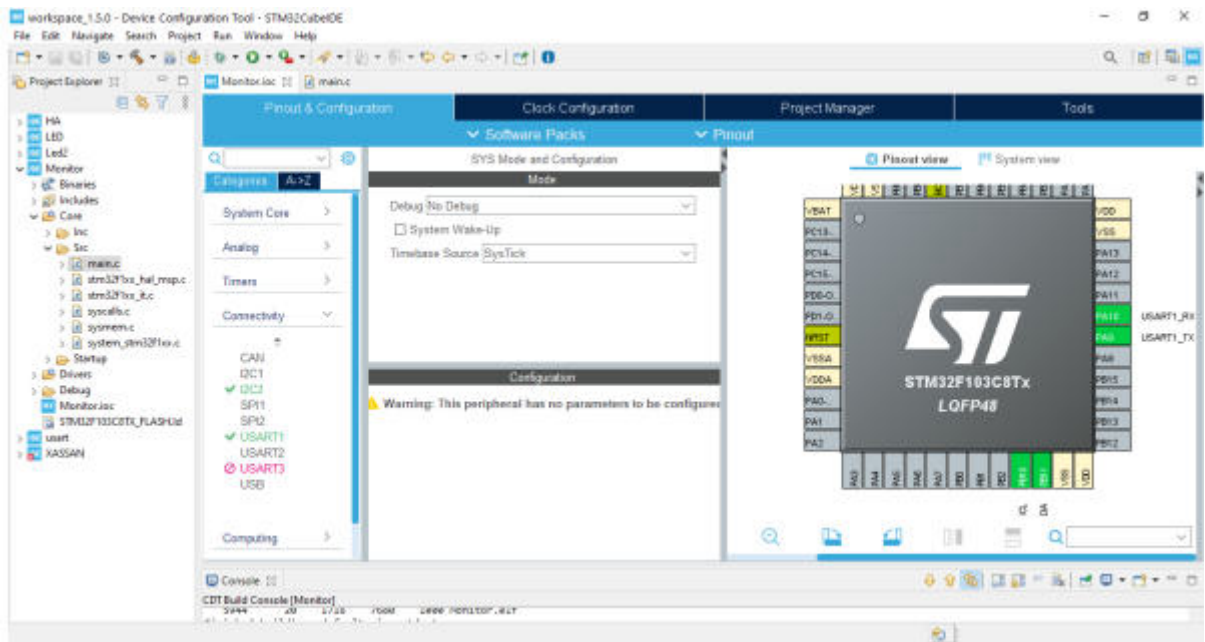


Рисунок 3.6 – Конфігурування виводів мікроконтролера

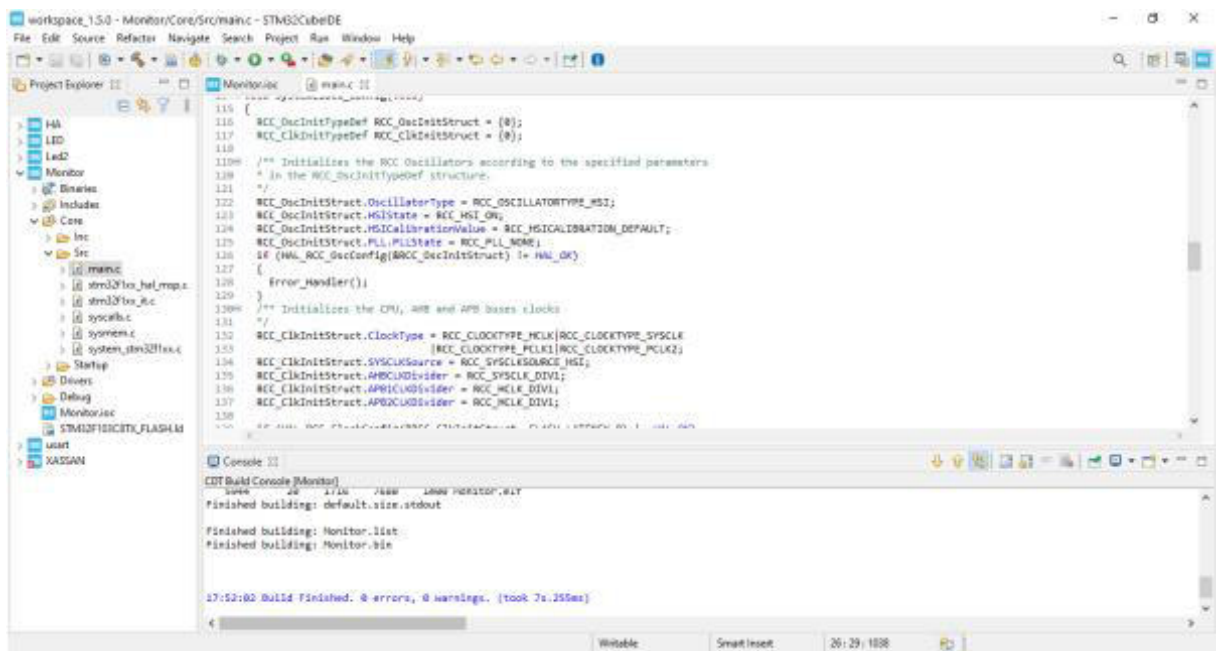


Рисунок 3.7 – Редактор коду

Для роботи з налагоджувальною інформацією UART використовувалася програма Terminal v1.9.3.b.

Для запису виконаного файлу у пам'ять мікроконтролера використовувалася програма STM32CubeProgrammer.

3.3.2 Ініціалізація периферії та портів мікроконтролера

Для налаштування портів вводу-виводу та інтерфейсів, що зв'язують модулі в одну систему, можна скористатися кількома способами. Перший з них – скористатися програмами-генераторами коду, які дають змогу за допомогою графічного інтерфейсу швидко та безпомилково налаштувати роботу всієї периферії. Таке налаштування може бути здійснено засобами STM32CubeIDE. З методичної точки зору краще було налаштувати все самостійно, вивчивши незліченну кількість регістрів мікроконтролера STM32F103, але з міркувань економії часу було вирішено піти простим шляхом.

Усього було налаштовано 19 виводів мікроконтролера:

- PA13, PA14 - для під'єднання програматора;
- PC14, PC15 - під'єднання низькочастотного резонатора;
- PB6, PB7 – для інтерфейсу I²C;
- PA9, PA10 - для інтерфейсу UART;
- PA0, PA1, PA2, PA3 - з альтернативною функцією зовнішніх переривань для під'єднання навігаційних кнопок за налаштуваннями пристрою;
- PA4 – для програмної шини OneWire;
- PB0 – для АЦП;
- PB10, PB11, PB13, PB14, PB15 – для під'єднання дисплея за інтерфейсом SPI.

Описана ініціалізація виводів наочно відображається у середовищі STM32CubeIDE як показано на рис. 3.8.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

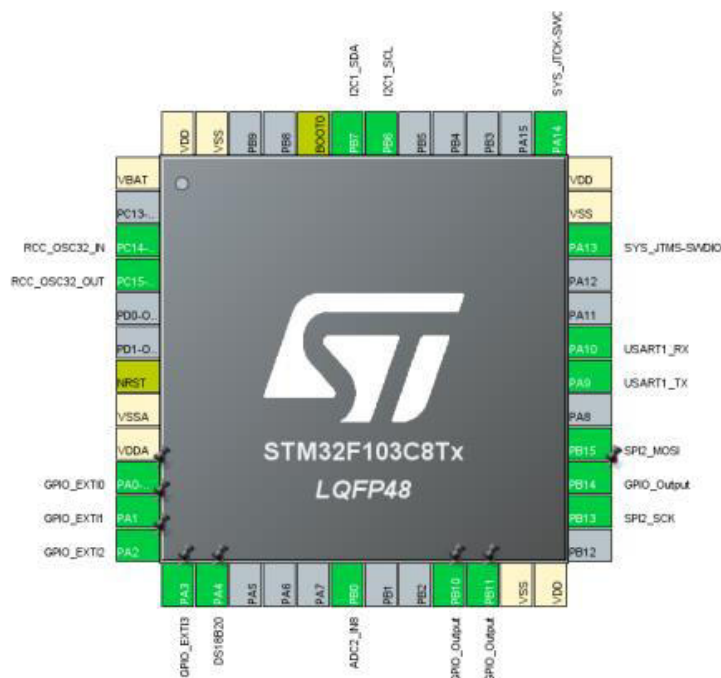


Рисунок 3.8 – Виводи мікроконтролера у середовищі STM32CubeIDE

Фрагменти вихідного коду ПЗ пристрою мовою Сі представлено на рис. 3.9, 3.10.

Повністю програмний код наведено в Додатку Б.

```

void sensor_init()
{
    uint8_t status;
    if(sensor==1)
    {
        // for DS18B20
        port_init();
        HAL_Delay(150);
        status = ds18b20_init(SKIP_ROM);
        if (status == 0)
            sensor_available = 1;
        else sensor_available = 0;
    }

    if(sensor==2)
    {
        bmp280_init_default_params(&bmp280.params);
        bmp280.addr = BMP280_I2C_ADDRESS_0;
        bmp280.i2c = &hi2c1;

        status = bmp280_init(&bmp280, &bmp280.params);
    }
}

```

Рисунок 3.9 – Лістинг фрагменту функції ініціалізації давачів

```

void get_data()
{
    int val_tem_prev = val_tem;
    if((sensor==1) && (sensor_available==1))
    {
        // for DS18B20
        if (ds18b20step==0)
        {
            ds18b20_MeasureTemperCmd(SKIP_ROM, 0);
            HAL_TIM_Base_Start_IT(&htim4);
        }
        else
        {
            HAL_TIM_Base_Stop_IT(&htim4);
            uint8_t dt[8];
            uint16_t raw_temper;
            ds18b20_ReadStratcpad(SKIP_ROM, dt, 0);
            raw_temper = ((uint16_t)dt[1]<<8)|dt[0];
            val_tem = ds18b20_Convert(raw_temper);
            if(ds18b20_GetSign(raw_temper))
                val_tem = -val_tem;
            ds18b20step=0;
            is_valid_data = true;
        }
    }
}

```

Рисунок 3.10 – Лістинг фрагменту функції одержання даних

Підсумовуючи виконану роботу хочеться відзначити, що поставлені завдання досягнуто, але пам'ять мікроконтролера використана майже повністю. У разі модернізації пристрою, що вимагає збільшення програмного коду, необхідно замінити мікроконтролер STM32F103C8T6 на мікроконтролер STM32F103CBT6, єдиною відмінністю якого є вдвічі більший об'єм flash пам'яті (128 кбайт).

Слід зауважити, що представлений у цій роботі проєкт може отримати продовження: навіть на частоті 8 МГц продуктивності мікроконтролера вистачає для вирішення всіх покладених на нього завдань. Без використання зовнішнього резонатора частота може бути збільшена до 64 МГц. Цього приросту продуктивності вистачить реалізації логування на зовнішній накопичувач інформації (flash карта), а незайняті виводи дозволять керувати ще 15 лініями навантаження.

					<i>КС КРБ 123.154.00.00 ПЗ</i>	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Долікарська допомога при опіках

Опік — це ушкодження тканин, яке спричинене дією високої температури, електроструму, хімічних речовин, рентгенівських і сонячних променів. Опіки можуть бути первинними (миттєвими) та вторинними. Вторинні опіки є результатом займання одягу та охоплення полум'ям тіла.

Перша допомога при опіках, які спричинені полум'ям передбачає гасіння палаючого одягу водою, з використанням вогнегасника або потерпілого накривають плащем, пальто, ковдрою тощо. Якщо необхідно звільнити постраждалого від одягу, після гасіння вогню в деяких випадках для звільнення ураженої ділянки тіла одяг розрізають.

При обмежених опіках, спричинених окропом, обпечену поверхню тіла охолоджують холодною водою з-під крану впродовж 10 хв. Після цього накладають асептичну суху пов'язку, застосовуючи чисту бавовняну тканину, бинт, індивідуальний пакет. У випадку поширених опіків потерпілого загортають у чисте простирадло, вкриваючи зверху, та дають пити чай [27].

Під час надання першої лікарської допомоги бригадою швидкої допомоги, крім накладання асептичної пов'язки, вводять серцеві та знеболювальні засоби, здійснюють протишокові заходи під час транспортування в медичний заклад.

Під час надання долікарської допомоги застосовують: інфільтрацію уражених тканин 0,25 % розчином новокаїну, місцеву гіпотермію, новокаїнову блокаду, накладають фурацилінові пов'язки, шкіру змащують стерильною олією.

При термічних опіках накладають стерильну пов'язку, забезпечують тепло, обмотують людину простирадлом, дають солодку каву, теплий чай, обтирають горілкою, одеколоном, спиртом чи змащують ними бинти і пов'язки. При шокові

					КС КРБ 123.154.00.00 ПЗ		
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>			
<i>Розроб.</i>		<i>Волощук І.В.</i>				<i>Лім.</i>	<i>Арк.</i>
<i>Керівник.</i>		<i>Стадник Н.Б.</i>					<i>Аркушів</i>
<i>Реценз.</i>					ТНТУ, каф. КС, гр. СІ-41		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>					
<i>Затверд.</i>		<i>Осухівська Г.М</i>					

дають пити двадцять крапель валер'янки.

При хімічних опіках 10-20 хв виконують промивання обпеченого місця струменем проточної води. У випадку отримання опіку лугом накладають марлеву пов'язку, змочену розчином борної кислоти (1 гр на склянку води) [27].

Заборонено відривати пухирі, а також відривати від них одяг, каніфоль, сургуч, бо це може спричинити інфекцію та затягнути тривалість загоєння ран. Заборонено засипати рани порошками, змащувати мазями та маслом. При обширних і важких опіках (більше 15-30% всієї поверхні тіла) виникає загальне ураження організму, яке супроводжується важким шоком (опікова хвороба), викликає інтоксикацію організму, зміни складу крові, зміни в роботі центральної нервової системи (біль). Чим більша опікова поверхня, тим більше нервових закінчень уражено і тим сильніше проявляються явища травматичного шоку [28].

При опіках з'являється велика кількість виділень крізь опікову поверхню плазми крові, виникає отруєння організму від продуктів розпаду змертвілої тканини, які із зони ушкодження всмоктуються організмом. З'являються такі симптоми як блювання, загальна слабкість, головний біль. Потерпілому потрошки і часто дають пити воду з питною сіллю, (одна чайна ложка солі + пів чайної ложки соди на один літр води).

Для пов'язок застосовують індивідуальний пакет, стерильний бинт. Опечену поверхню можна накрити чистою тканиною з бавовни, пропрасованою гарячою праскою чи змоченою перманганатом калію, горілкою або етиловим спиртом, які зменшують біль. Постраждалого тепло вкривають, для зменшення шоку вводять наркотичні речовини (морфій, промедол), дають пити гарячий чай з вином, каву, трохи горілки [28].

У випадку виникнення опіку фосфором ушкоджену частину тіла опускають в воду і там пінцетом знімають частинки фосфору, шкіру обробляють п'яти процентним розчином мідного купоросу і закривають чистою сухою пов'язкою.

При ураженні хімічною зброєю отруйні речовини всмоктуються в кров зі значно більшою швидкістю, ніж при їх потраплянні на неушкоджену ділянку

					КС КРБ 123.154.00.00 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

шкіри: ними можуть бути органи дихання і травлення, очі, заражені шкірні ділянки. Отруйні речовини можуть потрапляти на поверхню опіків і ран у вигляді газоподібних речовин, аерозолів і крапель.

Невідкладні заходи першої долікарської допомоги включають: інгаляцію кисню, дегазацію отруйних речовин, введення антидоту, введення протисудомних і серцево-судинних засобів, нейтралізація отруйних речовин на шкірі [27].

4.2 Вимоги пожежної безпеки при гасінні електроустановок

Пожежа – це неконтрольований процес горіння, який спричиняє нищення матеріальних цінностей та загибель людей. Серед причин виникнення пожеж можна виділити природні явища (посуха, блискавка), порушення правил пожежної безпеки, недбалу поведінку людей з вогнем. Відомо, що лише 7-8% пожеж спричинені блискавками і близько 90 % виникає внаслідок діяльності людини [29].

Пожежна безпека організацій та підприємств, у яких використовуються електроустановки, забезпечується шляхом здійснення організаційно-технічних та інших заходів з попередження виникнення пожеж, зменшення можливих матеріальних збитків, забезпечення безпеки людей, зниження негативних екологічних наслідків, створення умов для успішного гасіння пожеж та швидкого виклику пожежних підрозділів, а також евакуації з території виникнення та ймовірного розповсюдження пожежі людей, матеріальних цінностей і документів.

Система для контролю пожежної безпеки передбачає використання блоків живлення для подачі напруги необхідного рівня. Таке електрообладнання при неналежному нагляді, наприклад, при короткому замиканні, може стати епіцентром спалаху.

При виникненні пожежі можна виділити два методи, які застосовуються для гасіння електроустановок:

					КС КРБ 123.154.00.00 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

- гасіння електроустановок відведених від напруги мережі;
- гасіння електроустановок, які знаходяться під напругою.

Електроустановки повинні бути під'єднані до заземлення з гнучкого мідного голого провідника з поперечним перерізом не менше 25 мм², який підключається до заземлених конструкцій. Місця підключення до заземлених конструкцій, які визначаються фахівцями енергетичних об'єктів разом з представниками гарнізону пожежної охорони, вносяться до графічної частини плану пожежогасіння та позначаються знаком заземлення [28].

У разі виникнення пожежі на електроустановці особа, яка першою виявила факт загорання, повинна негайно повідомити відповідальних за пожежну безпеку осіб та керівника для уникнення подальшого загорання. Гасіння електрообладнання під напругою із застосуванням ручних стволів повинне виконуватися за умови [29]:

- застосування ефективних прийомів і способів подачі в зону горіння вогнегасних речовин;
- дотримання електробезпечних відстаней від електрообладнання, яке знаходиться під напругою, до пожежників, які використовують ручні пожежні стволи;
- забезпечення надійного заземлення пожежних автомобілів і стволів;
- використання індивідуальних ізолюючих електрозахисних засобів під час гасіння пожежі електроустановок без зняття напруги.

Як вогнегасні речовини під час гасіння електроустановок, які знаходяться під напругою, доцільно застосовувати: розпилені порошкові суміші й інертні гази, струмені води, комбіновану суміш, яка являє собою розпилену воду з порошком. При гасінні пожежі на електроустановках під напругою потрібно застосовувати прийоми та засоби подачі в зону горіння вогнегасних речовин, що забезпечують ефективне гасіння пожежі і безпечну роботу пожежників.

Компактні струмені води доцільно використовувати лише під час гасіння пожеж на електроустановках під напругою до 110 кВ, але лише в тих випадках, коли до осередку горіння не має можливості наблизитися для подачі розпиленої

					КС КРБ 123.154.00.00 ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

води. В цьому випадку пожежник повинен перебувати на безпечній відстані від найближчих струмоведучих частин електроустановок, до яких може торкнутися струмінь води [28].

Для гасіння пожеж електроустановок, що знаходяться під напругою, можна застосовувати воду з водопровідних мереж, а також із штучних і природних водойм. Забір води насосами пожежних автомобілів з водойм варто здійснювати зі спеціально обладнаних пірсів.

При гасінні пожежі на електроустановках, які перебувають під напругою до 220 кВ включно, тривалість перебування пожежників на бойових позиціях не лімітується. Заземлення ручних пожежних насосів і стволів пожежних автомобілів при гасінні пожеж на електроустановках, які знаходяться під напругою, повинно виконуватись за допомогою гнучких мідних провідників з поперечним перерізом не менше 12 мм², оснащених спеціальними струбцинами для під'єднання до заземлених конструкцій: шурфів, обсадних труб артезіанських свердловин, металевих опор повітряних ліній електропередач, гідрантів водогінних мереж.

Місця під'єднання до заземлених конструкцій повинні затверджуватись спеціалістами енергетичного об'єкта, позначатись відповідними знаками заземлення і вноситись у графічну частину плану пожежогасіння. Ручні пожежні насоси й стволи пожежних автомобілів необхідно заземлювати окремо. В процесі подачі води від внутрішнього водопроводу заземлюються лише стволи.

Індивідуальні електрозахисні ізолюючі засоби (діелектричні боти, рукавиці) потрібно застосовувати для електробезпечності пожежників та персоналу, який безпосередньо бере участь у гасінні пожежі на електроустановках, які знаходяться під напругою.

Автомобілі пожежних частин, що охороняють енергетичні об'єкти, повинні бути укомплектовані індивідуальними ізолюючими захисними засобами відповідно до чисельності бойової обслуги, яка безпосередньо бере участь у процесі гасіння пожежі [27]. Необхідна кількість індивідуальних ізолюючих захисних засобів на енергооб'єктах, у тому числі для пожежних підрозділів, що

					КС КРБ 123.154.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

залучаються до гасіння пожеж з інших частин, затверджується під час розробки планів пожежогасіння.

При пожежі на електроустановках, які знаходяться під напругою, обслуговуючий персонал зобов'язаний у першу чергу повідомити про пожежу начальника зміни (диспетчера, чергового) й пожежну охорону, а потім взяти усіх необхідних заходів відповідно до плану пожежогасіння (картки пожежогасіння).

За необхідності гасіння пожежі повітряно-механічною піною, з об'ємним заповненням приміщення (тунелю) піною виконується попереднє закріплення піногенераторів, їх заземлення, а також заземлення насосів пожежних машин. Водій пожежної машини повинен працювати в діелектричному взутті і рукавицях.

Після прибуття до місця виклику першого пожежного підрозділу старший начальник (заступник начальника частини, начальник варті і т. ін.) повинен швидко зв'язатися з начальником зміни або посадовою особою, яка відповідає за виконання робіт, з метою уточнення ситуації на пожежі, одержання інструктажу та письмового допуску на виконання гасіння пожежі на електроустановках, що перебувають під напругою [29].

Після узгодження маршрутів руху до осередку горіння та розміщення бойових позицій, із яких пожежники виконуватимуть подачу вогнегасних речовин, керівник групи повинен провести інструктаж всього особового складу, який бере участь у гасінні пожежі, й віддати розпорядження на бойове розгортання.

З метою набуття вольових якостей, удосконалення тактичних навичок, підвищення фахової майстерності і забезпечення психологічної підготовки з врахуванням дотримання правил безпеки праці особовий склад пожежних підрозділів гарнізону зобов'язаний безпосередньо на енергетичному об'єкті не рідше одного разу на рік проходити спеціальний інструктаж.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Підбиваючи підсумки виконаної роботи, можна зробити висновки про те, виконані такі завдання:

- визначено основний функціонал пристрою;
- проведено аналіз комерційно доступних аналогів;
- складено технічне завдання;
- складено електричну принципову схему, підібрано компоненти відповідно до вимог технічного завдання;
- визначено методи проектування, написано код програми мікроконтролера STM32;
- зібрано та протестовано прототип, визначено його придатність до вирішення поставлених завдань.

В цілому розроблена система на основі мікроконтролера STM32 справляється з поставленими завданнями і відповідає вимогам технічного завдання.

					<i>КС КРБ 123.154.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		51

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Maksimović M., Vujović V., Perišić B. A custom Internet of Things healthcare system // 2015 10th Iberian Conference on Information Systems and Technologies, CISTI 2015. 2015. pp. 4–9.
2. . Harish Kondoor. Kick-Start to MicroPython using ESP32 / ESP8266. 2021. 61 p.
3. Gay W. Beginning STM32: developing with FreeRTOS, libopencm3 and GCC // Beginning STM32. Apress Berkeley, 2018. 409 p.
4. Zúñiga Gómez V. S., et al. Portable Biomedical System for Acquisition, Display and Analysis (ECG, PCG, ICG, PPG). // MDPI. 2025. pp. 446-521.
5. Xiaodong Z., Jie Z. Design and Implementation of Smart Home Control System Based on STM32 // The 30th Chinese Control and Decision Conference (2018 CCDC). IEEE, 2018. p. 3023–3027.
6. Yang B. Design and implementation of intelligent home wireless gateway based on STM32 // Proceedings - 2017 4th International Conference on Information Science and Control Engineering, ICISCE 2017. 2017. pp. 258– 260.
7. Yin H. Smart Home Smoke Detection and Relay Contract based on STM32 // 5th International Conference on Environment, Materials, Chemistry and Power Electronics (EMCPE 2016) Smart. 2016. pp. 475–478.
8. Xie Y. и др. STM32-based vehicle data acquisition system for Internet-ofVehicles // Proceedings - 16th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2017. IEEE, 2017. pp. 895–898.
9. Mai T.A. и др. Fuzzy-PID Controller for Two Wheels Balancing Robot Based on STM32 Microcontroller // Proceedings - 2019 International Conference on Engineering Technologies and Computer Science: Innovation and Application, EnT 2019. 2019. p. 20–24.
10. Ren L., Yu X. Hardware Implementation of STM32 Microcontroller-Based Indoor Environment Monitoring System // Open J. Appl. Sci. 2021. V. 11. pp. 997–1008.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

11. Wang A. Bluetooth wireless temperature and humidity detection system based on STM32 // E3S Web of Conferences. 2021. V. 284. P. 04006.
12. Kouki R., Salhi H., Bouani F. Application of model predictive control for a thermal process using STM32 microcontroller // 2017 International Conference on Control, Automation and Diagnosis, ICCAD 2017. IEEE, 2017. P. 146–151.
13. Zhang S.Y., Wang X.M. The design of temperature control system for rice wine fermentation // Proceedings of the 2nd International Conference On Systems Engineering and Modeling (ICSEM-13). 2013. P. 1007–1010.
14. Jacko P., Kravets O. Spectral Analysis by STM32 Microcontroller of the Mixed Signal // Proceedings of the International Conference on Modern Electrical and Energy Systems, MEES 2019. IEEE, 2019. PP. 342–345.
15. Aryafar M., Hamed M., Ganjeh M.M. A novel temperature compensated piezoresistive pressure sensor // Meas. J. Int. Meas. Confed. Elsevier Ltd, 2015. V. 63, № December 2018. P. 25–29.
16. Zhang J. и др. Design of PID temperature control system based on STM32 // IOP Conf. Ser. Mater. Sci. Eng. 2018. V. 322, № 7. PP. 072020.
17. Jue Z., Shun Y. Design of Modbus-Profibus fieldbus bridge based on the STM32 and VPC3 + C // ICSESS 2012 - Proc. 2012 IEEE 3rd Int. Conf. Softw. Eng. Serv. Sci. IEEE, 2012. P. 411–414.
18. Liu W., Dai J. Design of attitude sensor acquisition system based on STM32 // Proc. - 5th Int. Conf. Instrum. Meas. Comput. Commun. Control. IMCCC 2015. IEEE, 2015. P. 1850–1853.
19. Jacko P. и др. The parallel data processing by nucleo board with STM32 microcontrollers // Proc. Int. Conf. Mod. Electr. Energy Syst. MEES 2017. 2017. V. 2018-Janua. P. 264–267.
20. Жаровський Р.О., Луцик Н.С., Осухівська Г.М., Паламар А.М., Тиш Є.В. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 39 с.

					КС КРБ 123.154.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

21. Паламар М.І., Стрембіцький М.О., Паламар А.М. Проектування комп'ютеризованих вимірювальних систем і комплексів. Навчальний посібник. Тернопіль: ТНТУ. 2019. 150 с.

22. Palamar A., Karpinski M., Palamar M., Osukhivska H., Mytnyk M. Remote Air Pollution Monitoring System Based on Internet of Things. CEUR Workshop Proceedings, 2nd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAР 2022), Ternopil, Ukraine, November 22–24, 2022. Vol. 3309. P. 194-204.

23. Palamar A., Palamar M., Osukhivska H. Real-time Health Monitoring Computer System Based on Internet of Medical Things. CEUR Workshop Proceedings, 3rd International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAР 2023), Ternopil, Ukraine, Opole, Poland, November 22–24, 2023. Vol. 3628. P. 106-115.

24. Voloskyi V., Leshchyshyn Y., Romanyshyn N., Palamar A., Tarasenko L. Method and algorithm for efficient cell balancing in the lithium-ion battery control system. CEUR Workshop Proceedings, The 1st International Workshop on Bioinformatics and Applied Information Technologies (BAIT 2024), Zboriv, Ukraine, October 02-04, 2024. Vol. 3842. P. 258-267.

25. Velychko D., Osukhivska H., Palaniza Y., Lutsyk N., Sobaszek L. Artificial Intelligence Based Emergency Identification Computer System. Advances in Science and Technology Research Journal, 18 no. 2, 2024, P. 296-304.

26. Луцків А., Лупенко С., Пасічник В. Паралельні та розподільнені обчислення. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 566 с.

27. Бедрій І.Я., Нечай В.Я. Безпека життєдіяльності. Навчальний посібник. Львів: Манголія 2006, 2007. 499 с.

28. Желібо Є. П. Заверуха Н.М., Зацарний В.В. Безпека життєдіяльності. Навчальний посібник. К.: Каравела, 2004. 328 с.

29. Зеркалов Д.В. Безпека життєдіяльності. Навчальний посібник. К.: Основа. 2011. 526 с.

					КС КРБ 123.154.00.00 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

Додаток А.
Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя
Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

“Затверджую”

Завідувач кафедри КС

_____ Осухівська Г.М.

“ ____ ” _____ 2026 р

**КОМП'ЮТЕРНА СИСТЕМА КОНТРОЛЮ ТЕМПЕРАТУРИ
ФАРБУВАЛЬНО-СУШИЛЬНОЇ КАМЕРИ ДЛЯ АВТОМОБІЛІВ**

ТЕХНІЧНЕ ЗАВДАННЯ

на _8_ листках

Вид робіт:

Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

Керівник кваліфікаційної роботи

_____ к.т.н. Стадник Н.Б.

« ____ » _____ 2026 р.

«ВИКОНАВЕЦЬ»

Студент групи СІ-42

_____ Волощук І.В..

« ____ » _____ 2026 р.

Тернопіль 2026

1 Загальні відомості

1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи: «Комп'ютерна система контролю температури фарбувально-сушильної камери для автомобілів».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.154.00.00

1.2 Виконавець

Студент групи СІ-41, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерної інженерії, Тернопільського національного технічного університету імені Івана Пулюя, Волощук Ігор Віталійович.

1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи є наказ по університету (№ 4/9-188 від 24.04.2026 р.)

1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи – 27.01.2026 р.

Плановий термін завершення виконання кваліфікаційної роботи – 15.06.2026 р.

1.5 Порядок оформлення та пред'явлення результатів роботи

Порядок оформлення пояснювальної записки та графічного матеріалу здійснюється у відповідності до чинних норм та правил ІСО, ЕСКД, ЕСПД та ДСТУ.

Пред'явлення проміжних результатів роботи з виконання кваліфікаційної роботи здійснюється у відповідності до графіку, затвердженого керівником роботи.

Попередній захист кваліфікаційної роботи відбувається при готовності роботи на 90% , наявності пояснювальної записки та графічного матеріалу.

Пред'явлення результатів кваліфікаційної роботи відбувається шляхом захисту на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

2 Призначення і цілі створення системи

2.1 Призначення системи

Комп'ютерна система контролю температури фарбувально-сушильної камери для автомобілів призначена для управління температурним режимом у камері для якісного нанесення лако-фарбового покриття.

До складу системи повинні входити як апаратна складова, так і програмна.

Доцільність створення системи зумовлена тим, що за допомогою вбудованого мікроконтролера можна знімати температурні показники для їх контролю без потреби перебування у камері на час відпрацювання технологічного процесу.

2.2 Мета створення системи

Основна мета розробки комп'ютерної системи контролю температури фарбувально-сушильної камери для автомобілів полягає у створенні вбудованої системи контролю температури у невеликій майстерні для спеціальної камери, де проводиться фарбування і сушіння авто.

Для того, щоб досягти поставленої мети роботи, необхідно розв'язати наступні задачі:

- визначення основного функціоналу пристрою, аналіз ринку;
- складання технічного завдання;
- вибір компонентів, створення принципової схеми;
- створення прототипу пристрою та ПЗ;
- створення готового пристрою.

2.3 Характеристика об'єкту

2.3.1 Основні задачі та функції об'єкту

Передбачається розробити прототип вбудованої системи контролю температури фарбувально-сушильної камери для автомобілів, Основні функції, що вимагають реалізації в комп'ютерній системі, що розробляється:

- встановлення температурного режиму;
- зняття даних температури;
- виведення отриманих даних на дисплей;
- передача даних з мікроконтролера.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

Система повинна бути спроектована так, щоб до її складу без особливих зусиль можна інтегрувати різні елементи управління, а також нові пристрої, не порушуючи при цьому структуру системи.

У проєктованій системі повинні бути забезпечені:

- надійність роботи апаратної частини;
- інтеграція з іншими виробниками пристроїв;
- продуктивність роботи програмного забезпечення;
- повністю реконфігурований інтерфейс;
- часова ефективність.

3.1.1 Вимоги до структури та функціонування системи

До структури та функціонування комп'ютерної системи контролю температури фарбувально-сушильної камери для автомобілів входять тактові кнопки, пов'язані з портами мікроконтролера; блок зняття температури, пов'язаний з вихідними портами контролера; рідкокристалічний дисплей для відображення інформації.

В цілому, концептуальна модель комп'ютеризованої системи повинна описувати предметну область.

Для прийому та обробки даних використовується мікроконтролер.

3.1.2 Вимоги до способів та засобів зв'язку між компонентами системи
Пропоноване рішення застосовує SPI; I²C; USART для передачі даних.

3.1.3 Вимоги по діагностуванню системи

Діагностика комп'ютерної системи контролю температури фарбувально-сушильної камери для автомобілів відбувається у відповідності до затвердженого розкладу профілактичних заходів.

3.1.4 Перспективи розвитку, модернізація системи

Перспективами розвитку та модернізації комп'ютерної системи контролю температури фарбувально-сушильної камери для автомобілів є можливість заміни мікроконтролер STM32F103C8T6 на мікроконтролер STM32F103CBT6, в якого є вдвічі більший об'єм flash пам'яті (128 кбайт).

Наявна апаратна складова системи при цьому не повинна зазнавати значних змін, а програмне забезпечення системи повинно передбачати гнучкість та здатність до масштабування.

3.1.5 Вимоги до надійності системи

Комп'ютерна система контролю температури фарбувально-сушильної камери для автомобілів повинна бути захищена на як на фізичному, так і на програмному рівнях. Фізичний рівень захисту повинен забезпечувати надійність щодо доступу до апаратного забезпечення, зокрема мікроконтролера та під'єднаних до нього компонентів.

Програмний рівень захисту повинен передбачати захист від сторонніх втручань і впливів.

3.1.6 Вимоги до функцій та задач, які виконує система

Функціональні вимоги та задачі, які повинна реалізовувати комп'ютерна система контролю температури фарбувально-сушильної камери для автомобілів полягають в наступному:

- можливість якісного визначення температури;
- передача даних від мікроконтролера;
- можливість управління двома лініями живлення незалежно від алгоритмів, що налаштовуються;
- можливість під'єднання додаткових датчиків за інтерфейсом I²C;
- відображення на вбудованому дисплеї актуальної інформації про температуру та статистичну інформацію;
- внесені користувачем зміни до налаштувань пристрою повинні зберігатися в незалежній пам'яті;
- на пристрої повинні бути виводи інтерфейсу USART для налагодження та програмування пристрою;
- пристрій повинен мати світлову індикацію живлення;
- припинення роботи у разі виявлення несправностей.

3.1.7 Вимоги до апаратного забезпечення

- плата STM32f103C8T6;
- датчі температури TMP36 та DS18P20;
- OLED дисплей SSD1306;
- низькочастотний резонатор.

3.1.8 Вимоги до програмного забезпечення

- мова програмування C/C++ (для програмування мікроконтролера);
- STM32Cube IDE (середовище програмування);
- EaseEDA (для прототипування)
- Terminal v1.9.3.b (для роботи з UART);

4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ

Комплект документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:

1 Узагальнена структура системи.

- 2 Схема елементів.
- 3 Налагодження компонентів на платах.
- 4 Алгоритм роботи

*Примітка: У комплект документації можуть вноситися міни та доповнення в процесі розробки.

- 5 Техніко-економічні показники

Планована собівартість комп'ютерної і контролю температури фарбувально-сушильної камери повинна становити не більше 2 000 грн.

*Примітка: собівартість системи може змінюватись під час розрахунку в процесі розробки.

- 6 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання кваліфікаційної роботи бакалавра

№ етапу	Назва етапу виконання кваліфікаційної роботи	Термін виконання
1.	Розробка технічного завдання	26.01 – 02.02
2.	Підбір джерел про вбудовані системи контролю температури на основі мікроконтролерів	03.02 – 04.03
3.	Опрацювання літературних джерел	05.04 – 10.05
4.	Виконання дослідження щодо розробки системи контролю температури фарбувально-сушильної камери для автомобілів	11.05 – 13.05
5.	Написання програмного коду	14.05 – 18.05
6.	Оформлення розділу «Аналіз технічного завдання»	19.05 – 22.05
7.	Оформлення розділу «Проектна частина»	23.05 – 26.05
8.	Оформлення розділу «Практична частина»	27.05 – 28.05
9.	Виконання завдання до підрозділу «Безпека життєдіяльності, основи охорони праці»	26.05 – 30.05
10.	Оформлення пояснювальної записки і графічного матеріалу	01.06 – 10.06

7 Додаткові умови виконання кваліфікаційної роботи

Під час виконання кваліфікаційної роботи у дане технічне завдання можуть вноситися зміни та доповнення.

Додаток Б

Лістинг програми

Файл main.c (рядки, додані автоматично середовищем CubeIDE, видалені)

```
#include "string.h"
#include "stdio.h"
#include "u8g.h"
#include "u8g_arm.h"
#include "bmp280.h"
#include "ds18b20.h"
#define ch1_on() ((GPIOA->ODR |= GPIO_ODR_ODR6))
#define ch1_off() ((GPIOA->ODR &= ~GPIO_ODR_ODR6))
#define ch2_on() ((GPIOA->ODR |= GPIO_ODR_ODR7))
#define ch2_off() ((GPIOA->ODR &= ~GPIO_ODR_ODR7))
RTC_TimeTypeDef RTC_Time;
char need_to_sensor_init = 1;
char need_to_pages_update = 0;
char need_to_save_settings = 0;
char time_to_draw = 0;
char time_to_get_data = 0;
int sensor = 1;
char sensor_available = 0;
u8g_t u8g;
float val_tem = 0;
float val_min = 200;
float val_max = -100;
char ds18b20step = 0;
char temperature[12];
char min[12];
char max[12];
char tmp_str[30];
struct text_page pages[7];
int active_page = -1;
bool is_ch1_on = false;
bool is_ch2_on = false;
bool is_valid_data = false;
bool is_first_step = true;
int temp1_on = 30;
int hyst1 = 2;
int temp2_on = 30;
int hyst2 = 2;
uint16_t k;
char temp1_on_str[5];
char hyst1_str[5];
char temp2_on_str[5];
char hyst2_str[5];
BMP280_HandleTypeDef bmp280;
float pressure, humidity;
uint16_t size;
char trans_str[124] = {0,};
char str[96] = {0,};
uint16_t adc = 0;
uint16_t eepronAddr = (0b1010000 << 1);
uint16_t baseMemAddr = 0x0100;
HAL_StatusTypeDef status;
```

```

void draw(void);
void get_data(void);
void text_pages_init(void);
void sensor_init(void);
void channels_logic(void);
void save_settings(void);
void read_settings(void);

int main(void)
{
    HAL_Init();
    SystemClock_Config();
    __HAL_RCC_I2C1_CLK_ENABLE();
    HAL_Delay(100);
    __HAL_RCC_I2C1_FORCE_RESET();
    HAL_Delay(100);
    __HAL_RCC_I2C1_RELEASE_RESET();
    HAL_Delay(100);

    read_settings();
    u8g_InitComFn(&u8g, &u8g_dev_ssd1306_128x64_hw_spi, u8g_com_hw_spi_fn);
    HAL_Delay(100);
    HAL_TIM_Base_Start_IT(&htim2);
    HAL_TIM_Base_Start(&htim3);
    text_pages_init();
    HAL_ADCEX_Calibration_Start(&hadc2);
    HAL_RTCEX_SetSecond_IT(&hrtc);

    while (1)
    {
        if (need_to_save_settings)
        {
            save_settings();
            need_to_save_settings = 0;
        }

        if (need_to_pages_update)
        {
            text_pages_init();
            need_to_pages_update = 0;
        }

        if (need_to_sensor_init)
        {
            sensor_init();
            need_to_sensor_init = 0;
        }

        if (time_to_get_data)
        {
            get_data();
            time_to_get_data = 0;
        }

        if (time_to_draw)
        {
            draw();
            time_to_draw = 0;
        }
    }
}

```

```

    chanel_logic();

    if (is_ch1_on) ch1_on();
    else ch1_off();

    if (is_ch2_on) ch2_on();
    else ch2_off();
}
}

void sensor_init()
{
    uint8_t status;
    if(sensor==1)
    {
        // for DS18B20
        port_init();
        HAL_Delay(150);
        status = ds18b20_init(SKIP_ROM);
        if (status == 0)
            sensor_available = 1;
        else sensor_available = 0;
    }

    if(sensor==2)
    {
        bmp280_init_default_params(&bmp280.params);
        bmp280.addr = BMP280_I2C_ADDRESS_0;
        bmp280.i2c = &hi2ci;

        status = bmp280_init(&bmp280, &bmp280.params);

        if (status == 1)
        {
            sensor_available = 1;
            bool bme280p = bmp280.id == BME280_CHIP_ID;
            size = sprintf((char *)trans_str, "BMP280: found %s\n", bme280p ?
"BME280" : "BMP280");
            HAL_UART_Transmit(&huart1, (uint8_t *)trans_str, size, 1000);
        }
        else
        {
            sensor_available = 0;
            size = sprintf((char *)trans_str, "BMP280 initialization
failed\n");
            HAL_UART_Transmit(&huart1, (uint8_t *)trans_str, size, 1000);
        }
    }

    if(sensor==3)
    {
        sensor_available = 1;
    }
}

void text_pages_init()
{
    uint8_t page = 0;
    pages[page].MENU_ITEMS = 4;
    strcpy(pages[page].item0, "MENU:");
}

```

```

strcpy(pages[page].item1, "Active sensor");
strcpy(pages[page].item2, "Relay");
strcpy(pages[page].item3, "Save");
pages[page].menu_current = 2;

page = 1;
pages[page].MENU_ITEMS = 4;
strcpy(pages[page].item0, "Active sensor:");
strcpy(pages[page].item1, "DS18B20");
strcpy(pages[page].item2, "BMP288");
strcpy(pages[page].item3, "TMP36");
pages[page].menu_current = 1;

page = 2;
pages[page].MENU_ITEMS = 5;
strcpy(pages[page].item0, "Relay:");
strcpy(pages[page].item1, "Temp1 ON ");
strcpy(pages[page].item2, "Hyster1: ");
strcpy(pages[page].item3, "Temp2 ON ");
strcpy(pages[page].item4, "Hyster2: ");
pages[page].menu_current = 1;

page = 3;
pages[page].MENU_ITEMS = 2;
strcpy(pages[page].item0, "Temp1 ON:");
sprintf(temp1_on_str, "%d", temp1_on);
strcpy(pages[page].item1, temp1_on_str);
pages[page].menu_current = 1;

page = 4;
pages[page].MENU_ITEMS = 2;
strcpy(pages[page].item0, "Hysteresis 1:");
sprintf(hyst1_str, "%d", hyst1);
strcpy(pages[page].item1, hyst1_str);
pages[page].menu_current = 1;

page = 5;
pages[page].MENU_ITEMS = 2;
strcpy(pages[page].item0, "Temp2 ON:");
sprintf(temp2_on_str, "%d", temp2_on);
strcpy(pages[page].item1, temp2_on_str);
pages[page].menu_current = 1;

page = 6;
pages[page].MENU_ITEMS = 2;
strcpy(pages[page].item0, "Hysteresis 2:");
sprintf(hyst2_str, "%d", hyst2);
strcpy(pages[page].item1, hyst2_str);
pages[page].menu_current = 1;
}

void draw_active_page(uint8_t page)
{
    uint8_t i, h;
    u8g_uint_t w, d;

    u8g_SetFont(&u8g, u8g_font_9x15);
    u8g_SetFontRefHeightText(&u8g);
    u8g_SetFontPosTop(&u8g);

    h = u8g_GetFontAscent(&u8g) - u8g_GetFontDescent(&u8g);

```

```

w = u8g_GetWidth(&u8g);

i=0;
u8g_SetDefaultForegroundColor(&u8g);
u8g_DrawStr(&u8g, 0, 1*h, pages[page].item0);

if (pages[page].MENU_ITEMS > 1)
{
    i=1;
    d = (w-u8g_GetStrWidth(&u8g, pages[page].item1))/2;
    u8g_SetDefaultForegroundColor(&u8g);
    if ( i == pages[page].menu_current )
    {
        u8g_DrawBox(&u8g, 0, 1*h+1, w, h);
        u8g_SetDefaultBackgroundColor(&u8g);
    }
    u8g_DrawStr(&u8g, d, 1*h, pages[page].item1);
}

if (pages[page].MENU_ITEMS > 2)
{
    i=2;
    d = (w-u8g_GetStrWidth(&u8g, pages[page].item2))/2;
    u8g_SetDefaultForegroundColor(&u8g);
    if ( i == pages[page].menu_current )
    {
        u8g_DrawBox(&u8g, 0, 1*h+1, w, h);
        u8g_SetDefaultBackgroundColor(&u8g);
    }
    u8g_DrawStr(&u8g, d, 1*h, pages[page].item2);
}

if (pages[page].MENU_ITEMS > 3)
{
    i=3;
    d = (w-u8g_GetStrWidth(&u8g, pages[page].item3))/2;
    u8g_SetDefaultForegroundColor(&u8g);
    if ( i == pages[page].menu_current )
    {
        u8g_DrawBox(&u8g, 0, 1*h+1, w, h);
        u8g_SetDefaultBackgroundColor(&u8g);
    }
    u8g_DrawStr(&u8g, d, 1*h, pages[page].item3);
}

if (pages[page].MENU_ITEMS > 4)
{
    i=4;
    d = (w-u8g_GetStrWidth(&u8g, pages[page].item4))/2;
    u8g_SetDefaultForegroundColor(&u8g);
    if ( i == pages[page].menu_current )
    {
        u8g_DrawBox(&u8g, 0, 1*h+1, w, h);
        u8g_SetDefaultBackgroundColor(&u8g);
    }
    u8g_DrawStr(&u8g, d, 1*h, pages[page].item4);
}
}

void draw_page_1()
{

```

```

uint8_t h;

u8g_SetFont(&u8g, u8g_font_9x15);
u8g_SetFontRefHeightText(&u8g);
u8g_SetFontPosTop(&u8g);

h = u8g_GetFontAscent(&u8g) - u8g_GetFontDescent(&u8g);

u8g.font = u8g_font_profont12;

if (sensor_available)
{
    if (sensor == 1)
        u8g_DrawStr(&u8g, 0, 0, "DS18B20:");
    if (sensor == 2)
        u8g_DrawStr(&u8g, 0, 0, "BMP280:");
    if (sensor == 3)
        u8g_DrawStr(&u8g, 0, 0, "TMP36:");

    sprintf(tmp_str, "%02u:%02u:%02u", RTC_Time.Hours, RTC_Time.Minutes,
RTC_Time.Seconds);
    u8g_DrawStr(&u8g, 80, 0, tmp_str);

    u8g_DrawStr(&u8g, 0, 1*h, temperature);

    u8g_DrawStr(&u8g, 0, 3*h, max);

    u8g_DrawStr(&u8g, 0, 4*h, min);

    sprintf(tmp_str, "t1on:%d", temp1_on);
    u8g_DrawStr(&u8g, 80, 1*h, tmp_str);

    if (is_ch1_on) sprintf(tmp_str, "ch1: %s", "on");
    else sprintf(tmp_str, "ch1: %s", "off");
    u8g_DrawStr(&u8g, 80, 2*h, tmp_str);

    sprintf(tmp_str, "t2on:%d", temp2_on);
    u8g_DrawStr(&u8g, 80, 3*h, tmp_str);

    if (is_ch2_on) sprintf(tmp_str, "ch2: %s", "on");
    else sprintf(tmp_str, "ch2: %s", "off");
    u8g_DrawStr(&u8g, 80, 4*h, tmp_str);

}
else u8g_DrawStr(&u8g, 0, 0, "Sensor not available");
}

void draw()
{
    u8g_FirstPage(&u8g);
    do
    {
        if (active_page == -1)
            draw_page_1();
        else
            draw_active_page(active_page);
    } while ( u8g_NextPage(&u8g) );
}

void data_to_string()

```

```

    (
        if ((is_first_step) && (is_valid_data))
        {
            val_min = val_tem;
            val_max = val_tem;
            is_first_step = false;
        }

        if (is_valid_data)
        {
            sprintf(temperature, "t: %.2f", val_tem);
            if (val_tem < val_min)
                val_min = val_tem;

            if (val_tem > val_max)
                val_max = val_tem;

            sprintf(max, "max: %.2f", val_max);
            sprintf(min, "min: %.2f", val_min);
        }
        else sprintf(temperature, "Error");
    )

void save_settings()
(
    char wmsg[8];
    status = HAL_I2C_IsDeviceReady(&hi2c1, eepromAddr, 1, HAL_MAX_DELAY);
    if(status == HAL_OK)
    {
        sprintf(wmsg, "%d", sensor);
        HAL_I2C_Mem_Write(&hi2c1, eepromAddr, baseMemAddr + 8,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)wmsg, sizeof(wmsg), HAL_MAX_DELAY);
        HAL_Delay(20);
        sprintf(wmsg, "%d", temp1_on);
        HAL_I2C_Mem_Write(&hi2c1, eepromAddr, baseMemAddr + 8,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)wmsg, sizeof(wmsg), HAL_MAX_DELAY);
        HAL_Delay(20);
        sprintf(wmsg, "%d", hyst1);
        HAL_I2C_Mem_Write(&hi2c1, eepromAddr, baseMemAddr + 16,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)wmsg, sizeof(wmsg), HAL_MAX_DELAY);
        HAL_Delay(20);
        sprintf(wmsg, "%d", temp2_on);
        HAL_I2C_Mem_Write(&hi2c1, eepromAddr, baseMemAddr + 24,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)wmsg, sizeof(wmsg), HAL_MAX_DELAY);
        HAL_Delay(20);
        sprintf(wmsg, "%d", hyst2);
        HAL_I2C_Mem_Write(&hi2c1, eepromAddr, baseMemAddr + 32,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)wmsg, sizeof(wmsg), HAL_MAX_DELAY);
        HAL_Delay(20);
    }
)

void read_settings()
(
    char rmsg[8];
    status = HAL_I2C_IsDeviceReady(&hi2c1, eepromAddr, 1, HAL_MAX_DELAY);

    if(status == HAL_OK)
    {
        HAL_I2C_Mem_Read(&hi2c1, eepromAddr, baseMemAddr + 8,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)rmsg, sizeof(rmsg), HAL_MAX_DELAY);
    }
)

```

```

        sensor = atoi(msg);
        HAL_I2C_Mem_Read(&hi2c1, eepromAddr, baseMemAddr + 8,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)msg, sizeof(msg), HAL_MAX_DELAY);
        temp1_on = atoi(msg);
        HAL_I2C_Mem_Read(&hi2c1, eepromAddr, baseMemAddr + 16,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)msg, sizeof(msg), HAL_MAX_DELAY);
        hyst1 = atoi(msg);
        HAL_I2C_Mem_Read(&hi2c1, eepromAddr, baseMemAddr + 24,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)msg, sizeof(msg), HAL_MAX_DELAY);
        temp2_on = atoi(msg);
        HAL_I2C_Mem_Read(&hi2c1, eepromAddr, baseMemAddr + 32,
I2C_MEMADD_SIZE_16BIT, (uint8_t*)msg, sizeof(msg), HAL_MAX_DELAY);
        hyst2 = atoi(msg);
    }
    else
    {
        sensor = 1;
        temp1_on = 30;
        hyst1 = 2;
        temp2_on = 30;
        hyst2 = 2;
    }
}

```

```
void get_data()
```

```

{
    int val_tem_prev = val_tem;
    if((sensor==1) && (sensor_available==1))
    {
        // for DS18B20
        if (ds18b20step-->0)
        {
            ds18b20_MeasureTempCnd(SKIP_ROM, 0);
            HAL_TIM_Base_Start_IT(&htim4);
        }
        else
        {
            HAL_TIM_Base_Stop_IT(&htim4);
            uint8_t dt[8];
            uint16_t raw_temper;
            ds18b20_ReadStratcpad(SKIP_ROM, dt, 0);
            raw_temper = ((uint16_t)dt[1]<<8)|dt[0];
            val_tem = ds18b20_Convert(raw_temper);
            if(ds18b20_GetSign(raw_temper))
                val_tem = -val_tem;
            ds18b20step=0;
            is_valid_data = true;
        }
    }

    if((sensor==2) && (sensor_available==1))
    {
        val_tem=0;
        while (!bmp280_read_float(&bmp280, &val_tem, &pressure))
        {
            size = sprintf((char *)trans_str, "Temperature/pressure reading
Failed\n");
            HAL_UART_Transmit(&huart1, (uint8_t *)trans_str, size, 1000);
        }
        is_valid_data = true;
    }
}

```

```

    }
    if((sensor--3) && (sensor_available--1))
    {
        HAL_ADC_Start(&hadc2);
        HAL_ADC_PollForConversion(&hadc2, 100);
        adc = HAL_ADC_GetValue(&hadc2);
        HAL_ADC_Stop(&hadc2);
        val_tem = (adc*3158/4095-500)/10;
        is_valid_data = true;
    }

    if ((val_tem_prev - val_tem) * (val_tem_prev - val_tem) > 400)
    {
        is_valid_data = false;
    }

    data_to_string();
}

void chanel_logic(void)
{
    if (is_valid_data)
    {
        if (val_tem >= temp1_on) is_ch1_on = true;
        if (val_tem < temp1_on - hyst1) is_ch1_on = false;

        if (val_tem >= temp2_on) is_ch2_on = true;
        if (val_tem < temp2_on - hyst2) is_ch2_on = false;
    }
}

```