

# КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

*бакалавр*

(назва освітнього ступеня)

на тему: *Комп'ютерна система IoT-моніторингу та реєстрації  
динамічних параметрів руху літального апарату*

Виконав: студент 4 курсу, групи СІ-41

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

*Верхола Б.А.*

(прізвище та ініціали)

Керівник

(підпис)

*Паламар А.М.*

(прізвище та ініціали)

Нормоконтроль

(підпис)

*Луцик Н.С.*

(прізвище та ініціали)

Завідувач кафедри

(підпис)

*Осухівська Г.М.*

(прізвище та ініціали)

Рецензент

(підпис)

*Деркач М.В.*

(прізвище та ініціали)

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.  
(підпис) (прізвище та ініціали)

«24» квітня 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 Верхолі Богдану Андрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютерна система IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

Керівник роботи Паламар Андрій Михайлович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «24» квітня 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. Проєктна частина

3. Практична частина

4. Безпека життєдіяльності, основи охорона праці

Висновки

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

1. Структурна схема системи

2. Схема електрична принципова

3. Блок-схема алгоритму роботи

4. Результати моделювання системи



## АНОТАЦІЯ

Верхола Б.А. Комп'ютерна система IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату : робота на здобуття освітнього ступеня бакалавра: спец. 123 — комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2026.

Ключові слова: IoT, літальний апарат, динамічні параметри руху, комп'ютерна система, сенсори, мікроконтролер, бездротова передача даних.

Кваліфікаційна робота присвячена розробці комп'ютерної системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату. Метою роботи є створення апаратно-програмного комплексу, здатного здійснювати збір, обробку, передачу та збереження телеметричних даних у режимі реального часу з використанням IoT-технологій.

У першому розділі виконано аналіз технічного завдання, визначено призначення та умови експлуатації системи, сформульовано функціональні вимоги. Проведено огляд і порівняльний аналіз існуючих аналогів і прототипів систем моніторингу параметрів руху літальних апаратів, що дозволило обґрунтувати доцільність розроблення власного рішення.

Другий розділ присвячено проектуванню апаратного забезпечення системи. Розроблено структурну та електричну принципову схеми пристрою, а також обґрунтовано вибір елементної бази, зокрема мікроконтролера, сенсорів та модулів бездротового зв'язку.

У третьому розділі розроблено алгоритм функціонування системи та програмне забезпечення для мікроконтролера. Виконано інтеграцію з IoT-платформою для віддаленого моніторингу та реєстрації даних. Проведено тестування розробленої системи та проаналізовано результати її роботи.

## ANNOTATION

Verkhola B.A. Computer System for IoT Monitoring and Registration of Dynamic Flight Parameters of an Aircraft. Bachelor's Graduation Thesis: speciality 123 — Computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2026.

Keywords: IoT, aircraft, dynamic motion parameters, computerized system, sensors, microcontroller, wireless data transmission.

The qualification work is devoted to the development of a computer system for IoT monitoring and registration of dynamic flight parameters of an aircraft. The aim of the work is to design a hardware–software complex capable of collecting, processing, transmitting, and storing telemetry data in real time using IoT technologies.

The first chapter presents an analysis of the technical specification, defines the purpose and operating conditions of the system, and formulates functional and non-functional requirements. A review and comparative analysis of existing analogs and prototype systems for aircraft motion parameter monitoring are carried out, which justifies the relevance of developing a dedicated solution.

The second chapter is devoted to the design of the system hardware. A structural diagram and an electrical schematic diagram are developed, and the selection of the electronic component base, including the microcontroller, sensors, and wireless communication modules, is substantiated.

In the third chapter, the system operation algorithm and microcontroller software are developed. Integration with an IoT platform for remote monitoring and data logging is implemented. Experimental testing of the developed system is performed, and the obtained results are analyzed.

## ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ.....	10
1.1 Огляд та аналіз сфер застосування системи моніторингу динамічних параметрів руху літального апарату.....	10
1.2 Аналіз вимог до комп'ютерної системи моніторингу та реєстрації динамічних параметрів руху літального апарату .....	12
1.3 Огляд існуючих засобів для моніторингу та реєстрації динамічних параметрів руху літального апарату.....	13
1.4 Аналіз можливих рішень поставленого завдання.....	17
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА .....	19
2.1 Структура комп'ютерної системи .....	19
2.2 Розроблення апаратного забезпечення системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату .....	21
2.2.1 Платформа мікроконтролера ESP32-S3 SIM7670G.....	21
2.2.2 Модуль 6-осьового акселерометра і гіроскопа BMI160.....	25
2.2.3 Давач BMP280 .....	28
2.2.4 OLED дисплей .....	31
2.3 Електрична принципова схема пристрою.....	34
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА .....	37
3.1 Алгоритм роботи системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату.....	37
3.2 Розробка програмного забезпечення.....	40
3.2.1 Ініціалізація апаратних та програмних компонентів системи.....	40
3.2.2 Реалізація основного циклу функціонування системи у підпрограмі loop...	42
3.2.3 Зчитування та фільтрація даних інерційного вимірювального модуля .....	44

					<i>КС КРБ 123.151.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Верхола Б.А.</i>			<i>Комп'ютерна система IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірів</i>		<i>Паламар А.М.</i>				5	83	
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, зр. СІ-41</i>		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>						
<i>Зав. каф.</i>		<i>Осухівська Г.М.</i>						

3.2.4 Зчитування та оброблення координат з GPS-модуля .....	45
3.2.5 Відображення телеметричної інформації на OLED-дисплеї .....	46
3.2.6 Реалізація запису телеметричних даних .....	48
3.2.7 Передача телеметричних даних та обробка керуючої кнопки .....	49
3.3 Налаштування хмарної IoT платформи для системи моніторингу .....	50
3.4 Результати моделювання та тестування системи .....	52
<b>РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ .....</b>	<b>56</b>
4.1 Стихійні лиха та їх класифікація .....	56
4.2 Розробка заходів щодо зниження рівня шуму на робочому місці .....	59
<b>ВИСНОВКИ.....</b>	<b>63</b>
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....</b>	<b>64</b>
Додаток А Технічне завдання	
Додаток Б Перелік елементів	
Додаток В Лістинг програми	

					<i>КС КРБ 123.151.00.00 ПЗ</i>	Арк.
						6
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## СПИСОК СКОРОЧЕНЬ

GNSS – Global Navigation Satellite System  
GPS – Global Positioning System  
GSM – Global System for Mobile Communications  
I2C – Inter-Integrated Circuit  
IMU – Inertial Measurement Unit  
IoT – Internet of Things  
LTE – Long Term Evolution  
OLED – Organic Light Emitting Diode  
UART – Universal Asynchronous Receiver-Transmitter  
БДС – блок давачів системи  
ГІС – гіроскопічний сенсор  
ДПР – динамічні параметри руху  
КСМ – комп'ютерна система моніторингу  
ЛА – літальний апарат  
МП – мікроконтролерна платформа  
СЗД – система збору даних  
ТІС – телеметрична інформаційна система  
ХП – хмарна платформа  
ЦОД – центр обробки даних

					КС КРБ 123.151.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		7

## ВСТУП

Стрімкий розвиток безпілотних та малогабаритних літальних апаратів, а також зростання вимог до їхньої безпеки, надійності та ефективності експлуатації зумовлюють необхідність використання сучасних засобів моніторингу параметрів руху. Контроль динамічних характеристик літального апарату, таких як прискорення, кутова швидкість, орієнтація в просторі та інші параметри, є критично важливим як під час випробувань, так і в процесі експлуатації. Недостатня або несвоєчасна інформація про ці параметри може призводити до втрати керованості, зниження точності навігації та аварійних ситуацій.

Традиційні системи реєстрації та телеметрії часто характеризуються високою вартістю, складністю інтеграції та обмеженими можливостями віддаленого доступу до даних. У свою чергу, використання технологій Інтернету речей (IoT) забезпечує нові можливості для створення компактних, масштабованих та енергоефективних систем моніторингу з підтримкою бездротової передачі даних і хмарного зберігання. Це обумовлює актуальність задачі розробки комп'ютерної системи моніторингу динамічних параметрів руху літального апарату, яка забезпечить оперативний збір, обробку та аналіз телеметричної інформації.

Метою даної кваліфікаційної роботи є розробка комп'ютерної системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату, що забезпечує надійний збір даних із сенсорів, їх обробку, передачу на віддалену IoT-платформу та можливість подальшого аналізу.

Для досягнення поставленої мети в роботі необхідно вирішити такі задачі:

- проаналізувати технічне завдання та умови експлуатації проєктованої системи;
- сформулювати функціональні та експлуатаційні вимоги до системи IoT-моніторингу;
- виконати огляд і аналіз існуючих аналогів систем моніторингу параметрів руху літальних апаратів;

					КС КРБ 123.151.00.00 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

- розробити структурну схему комп'ютерної системи;
- спроектувати електричну принципову схему апаратної частини;
- обґрунтувати вибір елементної бази системи;
- розробити алгоритм функціонування системи;
- створити програмне забезпечення для мікроконтролера;
- реалізувати інтеграцію системи з IoT-платформою;
- провести тестування розробленої системи.

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
						9
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

## РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

### 1.1 Огляд та аналіз сфер застосування системи моніторингу динамічних параметрів руху літального апарату

Сучасний етап розвитку авіаційної та аерокосмічної галузей характеризується широким упровадженням цифрових та вбудованих систем контролю, що забезпечують підвищення безпеки, ефективності та інформативності експлуатації літальних апаратів. Особливої актуальності набувають комп'ютерні системи моніторингу динамічних параметрів руху, здатні здійснювати безперервний збір і реєстрацію телеметричних даних у реальному часі. Використання IoT технологій у таких системах дозволяє забезпечити віддалений доступ до інформації, централізоване зберігання даних та їх подальший аналіз, що значно розширює сфери практичного застосування проектованої системи.

Однією з основних сфер використання комп'ютерної системи IoT-моніторингу є галузь безпілотних літальних апаратів (БПЛА). У процесі експлуатації БПЛА виникає потреба в оперативному контролі параметрів руху, таких як лінійні та кутові прискорення, просторове положення та стабільність польоту. Система моніторингу дозволяє не лише відстежувати поточний стан апарату, але й накопичувати статистичні дані для аналізу ефективності алгоритмів керування, оптимізації траєкторій руху та виявлення потенційних відмов. Це особливо важливо під час виконання місій у складних умовах або при автономному режимі польоту.

Значний інтерес до використання подібних систем спостерігається у сфері експериментальних та навчальних літальних апаратів. Під час проведення випробувань нових конструкцій або дослідних зразків виникає необхідність у детальній реєстрації динамічних параметрів руху для подальшого аналізу

					<i>КС КРБ 123.151.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Верхола Б.А.</i>			<i>Аналіз технічного завдання</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					<i>10</i>	<i>9</i>
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>						
<i>Зав. каф.</i>		<i>Осужівська Г.М.</i>						

результатів експериментів. Комп'ютерна система IoT-моніторингу дає змогу здійснювати вимірювання без застосування складних стаціонарних телеметричних комплексів, що знижує вартість досліджень та підвищує їх доступність для навчальних і наукових установ.

Ще однією важливою сферою застосування є авіаційна безпека та технічна діагностика. Дані про динаміку руху літального апарату можуть використовуватися для виявлення аномальних режимів роботи, перевантажень або нестабільності, які можуть свідчити про механічні пошкодження чи помилки в системах керування. У поєднанні з IoT-платформами це створює передумови для впровадження концепцій прогностного обслуговування, коли рішення щодо технічного стану приймаються на основі аналізу накопичених даних.

Комп'ютерні системи моніторингу динамічних параметрів також знаходять застосування у спортивній та аматорській авіації, зокрема для аналізу маневреності та оцінки якості пілотування. Реєстрація параметрів руху дозволяє виконувати післяпольотний аналіз, порівнювати різні режими керування та підвищувати рівень підготовки операторів літальних апаратів. У цьому контексті використання IoT-технологій спрощує доступ до даних і дає змогу об'єднувати інформацію з кількох апаратів у єдиній системі.

Окремо слід відзначити можливість застосування розглядуваної системи у суміжних галузях, де об'єктами контролю є рухомі платформи або транспортні засоби. Завдяки універсальності апаратно-програмної архітектури система може бути адаптована для моніторингу динамічних параметрів наземних, морських або роботизованих об'єктів. Це свідчить про високий рівень масштабованості та гнучкості запропонованого підходу.

Отже, аналіз сфер застосування системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату показує її актуальність і практичну значущість. Поєднання вбудованих систем, сучасних сенсорів та IoT-технологій забезпечує широкі можливості використання системи в авіаційній, науково-дослідній, навчальній та прикладній сферах, що підтверджує доцільність її розробки в межах даної кваліфікаційної роботи.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

## 1.2 Аналіз вимог до комп'ютерної системи моніторингу та реєстрації динамічних параметрів руху літального апарату

Вимоги до проєктованої комп'ютерної системи IoT-моніторингу формуються з урахуванням умов експлуатації літального апарату, особливостей вимірюваних параметрів, а також потреб користувачів у достовірній та оперативній інформації.

Однією з ключових вимог є забезпечення точного та стабільного вимірювання динамічних параметрів руху, зокрема прискорень, кутових швидкостей та просторової орієнтації. Система повинна забезпечувати достатню частоту дискретизації та мінімальні затримки під час обробки даних, що дозволяє коректно відображати швидкоплинні зміни руху літального апарату. Важливою вимогою є також можливість попередньої обробки та фільтрації сигналів безпосередньо на рівні мікроконтролера для зменшення впливу шумів і похибок вимірювань.

З огляду на використання IoT-технологій, суттєвими є вимоги до надійної бездротової передачі даних. Комп'ютерна система повинна підтримувати стабільний зв'язок з віддаленою IoT-платформою та забезпечувати цілісність і збереження телеметричної інформації. У випадку тимчасової втрати з'єднання система має зберігати дані локально з подальшою їх передачею після відновлення зв'язку. Це дозволяє уникнути втрати критично важливої інформації під час експлуатації.

Вагоме значення мають вимоги до апаратної частини системи. Враховуючи обмеження щодо маси, габаритів та енергоспоживання літального апарату, система повинна бути компактною та енергоефективною. Обрані електронні компоненти мають забезпечувати сумісність між собою, достатній запас обчислювальних ресурсів та стабільну роботу в умовах вібрацій і змінних зовнішніх факторів. Окремо слід врахувати вимоги до масштабованості апаратного забезпечення для можливого підключення додаткових сенсорів.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

Не менш важливими є вимоги до програмного забезпечення системи. Програмна реалізація повинна забезпечувати гнучкість налаштувань, можливість оновлення та розширення функціоналу без суттєвих змін апаратної частини. Інтерфейс взаємодії з IoT-платформою має бути інтуїтивно зрозумілим та забезпечувати зручну візуалізацію й аналіз отриманих даних.

Проаналізовані вимоги до комп'ютерної системи дозволяють сформувавши комплексний підхід до її проєктування, що забезпечує відповідність системи технічному завданню, умовам експлуатації та сучасним вимогам до IoT-рішень у сфері моніторингу динамічних параметрів руху літальних апаратів.

### 1.3 Огляд існуючих засобів для моніторингу та реєстрації динамічних параметрів руху літального апарату

Сучасний ринок рішень для моніторингу та реєстрації динамічних параметрів руху літальних апаратів охоплює широкий спектр продуктів – від промислових телеметричних систем для великої авіації до спеціалізованих модулів для БПЛА і автономних давачів. Базовою ідеєю усіх таких засобів є забезпечення збору, обробки та передачі даних про стан польоту, параметри руху й навколишнє середовище, що дозволяє підвищити інформованість операторів і систем керування польотом. Незважаючи на значну різноманітність доступних рішень, кожне з них має певні архітектурні та функціональні обмеження, які можуть бути критичними для застосування у рамкових задачах моніторингу динаміки руху, особливо в малобюджетних проєктах.

На рівні класичної авіації використовуються комплекси польотної телеметрії та системи вимірювальної апаратури, які забезпечують збір великого обсягу даних з численних бортових сенсорів та їхню передачу на наземну станцію для подальшої обробки. Такі системи базуються на стандартизованих підходах телеметрії, включно з РСМ/FM-модуляцією та високошвидкісними цифровими каналами передавання даних, які дозволяють реалізувати надійний зв'язок між літальним апаратом й наземною станцією у польоті [1]. Проте ці рішення, як правило,

					КС КРБ 123.151.00.00 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

орієнтовані на великі платформи й випробувальні програми, мають високу вартість та складну архітектуру, що ускладнює їх інтеграцію у малогабаритні або недорогі проекти моніторингу руху літальних апаратів.

На ринку існують промислові рішення для моніторингу польотних параметрів у комерційній та військовій авіації, такі як комплекси, що інтегрують інерційні давачі, барометричні й GNSS-модулі, які забезпечують багатопараметричний аналіз польоту в реальному часі та архівацію даних. Наприклад, пристрої, що відповідають рекомендаціям ED-155, можуть бути підключені до зовнішніх джерел позиціонування та реєструвати повний набір польотної інформації, включно з вимірюваннями навколишнього середовища та руху [2]. Такі системи характеризуються високою якістю даних і сертифікацією для експлуатації в галузі авіації, але вони часто мають складну інтеграцію та потребують значних знань для налаштування та обслуговування (рис. 1.1).



Рисунок 1.1 – Реєстратор польотних параметрів Acron Aviation

Для вимірювання динамічних параметрів руху в промислових та автомобільних тестових стендах широко застосовуються високоточні системи збору даних (DAQ) та GPS/IMU рішення, що дозволяють отримувати дані з частотою вимірювання до сотень герц і використовуються для тестування динамічних характеристик рухомих об'єктів. Пристрої таких виробників, як Racelogic, пропонують GPS-реєстратори та IMU-бази з підтримкою

					КС КРБ 123.151.00.00 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

диференційних GNSS-технологій, що забезпечують високу точність позиціонування та параметрів руху. Ці рішення мають демонстраційну якість і широкі можливості налаштування, але вони орієнтовані на застосування у випробуваннях у наземних та змішаних середовищах, а не на тривалу автономну роботу у польоті з обмеженими ресурсами (рис. 1.2).



Рисунок 1.2 – Система телеметрії Racelogic для реєстрації параметрів позиціонування та руху

У БПЛА основним елементом, що забезпечує моніторинг параметрів руху, є телеметричні системи, що передають дані польоту між БПЛА і наземною станцією (рис. 1.3).



Рисунок 1.3 – Система Mapple для передачі даних польоту між БПЛА і наземною станцією

Такі системи реалізують протоколи обміну, як-от MAVLink, які стандартизують передачу даних про стан апарату, навігаційну інформацію та діагностичні параметри [3]. Ці підходи забезпечують базову функціональність збору польотної інформації та передачі її у реальному часі, але вони мають низку обмежень: низька пропускна здатність каналів зв'язку, що впливає на обсяг переданих даних; залежність від стабільності зв'язку у польоті; а також обмежені можливості локального зберігання і обробки даних у разі втрати зв'язку.

Додатковим недоліком існуючих телеметричних рішень для БПЛА є те, що вони часто фокусуються на передачі вже сформованих польотних даних і не включають розширених можливостей для локальної обробки, фільтрації шумів чи адаптивної обробки сигналів. Сучасні підходи в наукових дослідженнях включають використання моделей машинного навчання або методів фільтрації для підвищення точності оцінки параметрів руху та корекції навігаційних даних на борту, але не всі комерційні рішення включають такі інтелектуальні модулі.

Ще одним напрямком є спеціалізовані рішення для вимірювання окремих параметрів руху, наприклад, системи для вимірювання кутових швидкостей і прискорень [4]. Такі системи можуть мати високу точність для окремих завдань – наприклад, моніторингу стабільності платформи або аналізу об'єкту у польоті – але вони зазвичай не пропонують повноцінної IoT-інтеграції або можливостей для віддаленого доступу до даних у реальному часі без додаткових модулів і розробки програмного забезпечення.

На базовому рівні, комерційні MEMS-сенсори й модулі IMU, як використовується в ряді відкритих проєктів, забезпечують можливість вимірювання прискорення та кутових швидкостей у компактному й енергоефективному форматі, але вони часто потребують додаткової обробки даних, калібрування й інтеграції з іншими компонентами для досягнення прийнятної точності. Це створює додаткові вимоги щодо розробки програмного забезпечення та алгоритмів обробки, що є складним завданням у рамках самостійних розробок.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

Слід зазначити, що більшість наявних рішень фокусуються або на комерційних технічних рішеннях для великої авіації, або на базовій телеметрії для БПЛА, але вони не завжди пропонують оптимальне поєднання низької вартості, автономності, можливості розширення й інтеграції з IoT-платформами, що лишає значний простір для реалізації гнучких, масштабованих і недорогих систем, призначених саме для моніторингу динамічних параметрів руху літальних апаратів у реальному часі та з віддаленим доступом до даних. Таким чином, проаналізовані на ринку рішення мають низку недоліків, серед яких висока вартість, складність інтеграції, обмежені можливості локальної обробки й адаптації під специфічні задачі моніторингу – проблеми, які повністю враховані у постановці задачі розробки комп'ютерної системи IoT-моніторингу для цієї кваліфікаційної роботи.

#### 1.4 Аналіз можливих рішень поставленого завдання

Поставлене в межах кваліфікаційної роботи завдання розробки комп'ютерної системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату може бути реалізоване різними технічними та архітектурними підходами. Вибір оптимального рішення повинен базуватися на вимогах технічного завдання, умовах експлуатації, обмеженнях щодо ресурсів, а також на доцільності використання сучасних технологій у сфері вбудованих та IoT-систем.

Одним із можливих рішень є застосування готових комерційних телеметричних систем або бортових реєстраторів, що використовуються в авіації та сфері БПЛА. Такі системи, як правило, забезпечують високу точність вимірювань, стабільність роботи та наявність сертифікації. Проте їх використання в межах даної роботи є недоцільним через високу вартість, обмежену гнучкість конфігурації та складність інтеграції з обраними IoT-платформами. Крім того, закритість апаратних і програмних рішень унеможливорює повноцінну адаптацію системи до специфічних вимог навчального або експериментального проекту.

Іншим підходом є використання стандартних систем керування польотом БПЛА, які вже містять інерційні сенсори та реалізують базові функції телеметрії.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

Такі системи дозволяють отримувати дані про динаміку руху в реальному часі та передавати їх на наземну станцію. Однак у більшості випадків ці рішення орієнтовані на задачі навігації та керування, а не на повноцінну реєстрацію й аналіз динамічних параметрів руху. Обмежені можливості локальної обробки, залежність від стабільності каналу зв'язку та недостатня інтеграція з хмарними IoT-сервісами знижують ефективність такого підходу в контексті поставленого завдання.

Перспективним рішенням є розробка власної комп'ютерної системи на базі сучасного мікроконтролера з використанням цифрових MEMS-сенсорів та бездротових модулів зв'язку. Такий підхід дозволяє реалізувати гнучку архітектуру системи, оптимізовану під конкретні вимоги моніторингу динамічних параметрів руху. Використання мікроконтролерних платформ дає змогу виконувати первинну обробку та фільтрацію даних безпосередньо на борту, зменшуючи навантаження на канали зв'язку та підвищуючи надійність системи.

Окремої уваги заслуговує вибір способу передавання та зберігання даних. Застосування IoT-платформ дозволяє реалізувати віддалений доступ до телеметричної інформації, зберігання історичних даних та їх візуалізацію у зручному вигляді. Альтернативні рішення, що базуються лише на локальному збереженні даних, не забезпечують необхідного рівня оперативності та масштабованості, тому є менш придатними для сучасних систем моніторингу.

Найбільш доцільним підходом до реалізації поставленого завдання є розробка власної комп'ютерної IoT-системи з використанням мікроконтролерної платформи, сенсорів динаміки руху та хмарної IoT-інфраструктури. Такий підхід забезпечує оптимальне поєднання функціональності, гнучкості, масштабованості та економічної доцільності, а також повністю відповідає цілям і завданням даної кваліфікаційної роботи.

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		18

## РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

### 2.1 Структура комп'ютерної системи

Проектована система призначена для реалізації функцій вимірювання, реєстрації, відображення та передавання параметрів руху літального апарату з використанням сучасних засобів вбудованої електроніки та технологій Інтернету речей. На рисунку 2.1 подано структурну схему цієї комп'ютерної системи.

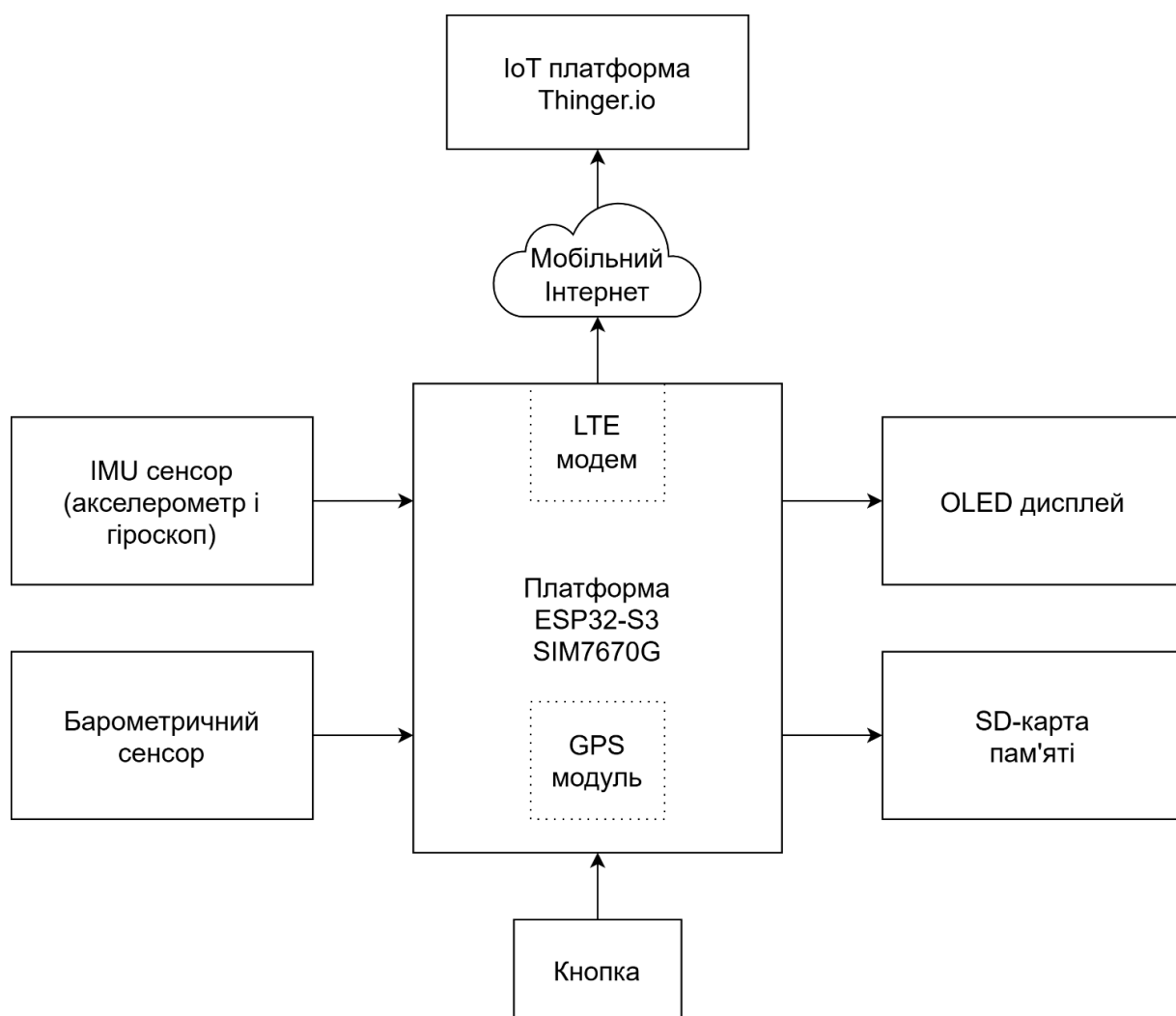


Рисунок 2.1 – Структурна схема системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

					<i>КС КРБ 123.151.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Верхола Б.А.</i>			<i>Проектна частина</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					19	18
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>						
<i>Зав. каф.</i>		<i>Осунівська Г.М.</i>						

Центральним елементом системи є мікроконтролерна платформа, яка виконує функції оброблення даних, керування роботою периферійних модулів, реєстрації вимірюваних параметрів та передавання інформації до віддаленої хмарної платформи. До складу цієї платформи входять інтегровані засоби супутникового позиціонування, модуль стільникового зв'язку та інтерфейс для під'єднання карти пам'яті, що створює можливість автономної реєстрації даних та їх подальшого передавання через мережу мобільного зв'язку.

До мікроконтролерної платформи підключено модуль інерційних вимірювань (IMU), що містить триосьовий акселерометр і триосьовий гіроскоп. Цей модуль призначений для визначення динамічних параметрів руху об'єкта, зокрема лінійних прискорень та кутових швидкостей за трьома просторовими осями. Отримані значення використовуються для аналізу руху літального апарату, оцінювання його просторової орієнтації та реєстрації динамічних режимів польоту.

Крім того, до системи підключено давач атмосферного тиску та температури, який забезпечує вимірювання параметрів навколишнього середовища. На основі отриманих значень атмосферного тиску може бути здійснене визначення відносної висоти польоту, що є важливим параметром для аналізу руху літального апарату та дослідження умов його експлуатації.

Для забезпечення взаємодії користувача із системою передбачено використання кнопки керування та графічного дисплея. Кнопка використовується для ініціювання основних режимів роботи пристрою, зокрема запуску або зупинки процесу реєстрації даних. Графічний дисплей призначений для відображення поточної службової інформації, стану системи, а також окремих вимірюваних параметрів, що дозволяє здійснювати оперативний контроль роботи пристрою без використання зовнішніх засобів.

Важливим елементом структури системи є підсистема зберігання даних, реалізована на основі карти пам'яті. Вона забезпечує локальну реєстрацію вимірюваних параметрів у вигляді файлів телеметрії, що дозволяє зберігати інформацію навіть у випадку відсутності зв'язку з мережею передачі даних.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

Збережені дані можуть бути застосовані для подальшого аналізу результатів експериментів або дослідження параметрів руху літального апарату.

Для забезпечення віддаленого моніторингу параметрів системи передбачено передавання даних до хмарної IoT платформи. Передавання інформації здійснюється через модуль стільникового зв'язку, інтегрований у мікроконтролерну платформу. Хмарна платформа забезпечує приймання, зберігання та візуалізацію отриманих даних, що дає змогу здійснювати дистанційний контроль параметрів руху об'єкта у режимі реального часу або з використанням архівних записів.

## 2.2 Розроблення апаратного забезпечення системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

### 2.2.1 Платформа мікроконтролера ESP32-S3 SIM7670G

Одним із ключових елементів проєктованої комп'ютерної системи є мікроконтролерна платформа ESP32-S3 SIM7670G, розроблена компанією Waveshare (рис. 2.2).



Рисунок 2.2 – Платформа мікроконтролера ESP32-S3 SIM7670G

					КС КРБ 123.151.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

Дана плата є універсальною апаратною платформою для створення IoT-пристроїв, систем віддаленого моніторингу та мобільних телеметричних систем. Вона поєднує в одному пристрої мікроконтролер, модуль бездротового зв'язку, засоби супутникового позиціонування, інтерфейси для підключення периферійних пристроїв та підсистему керування живленням. Завдяки інтеграції цих функціональних модулів плата дозволяє значно спростити апаратну реалізацію комп'ютерних систем збору та передавання даних.

Основу платформи становить мікроконтролер ESP32-S3, який належить до сімейства високопродуктивних мікроконтролерів компанії Espressif Systems. Даний мікроконтролер має двоядерний 32-бітний процесор архітектури Xtensa LX7. У його структурі передбачено внутрішню оперативну пам'ять SRAM обсягом 512 КБ, постійну пам'ять ROM обсягом 384 КБ, а також зовнішню PSRAM обсягом 2 МБ і флеш-пам'ять обсягом 16 МБ, що використовується для зберігання програмного коду та даних. Мікроконтролер також містить апаратні засоби забезпечення безпеки, зокрема генератор випадкових чисел, криптографічний прискорювач та модулі цифрового підпису, що дозволяє застосовувати його у захищених IoT-системах. Крім того, мікроконтролер має інтегровані модулі бездротового зв'язку Wi-Fi та Bluetooth, що забезпечує можливість локального бездротового обміну даними.

Важливою складовою платформи є інтегрований стільниковий модуль SIM7670G, який забезпечує доступ до мереж мобільного зв'язку стандарту LTE Cat-1. Даний модуль підтримує широкий спектр частотних діапазонів LTE-FDD та LTE-TDD, що дозволяє використовувати пристрій у різних регіонах світу. Максимальна швидкість передавання даних у мережі становить приблизно до 10 Мбіт/с у напрямку завантаження та до 5 Мбіт/с у напрямку передавання. Окрім функцій стільникового зв'язку, модем містить вбудований GNSS-приймач, який підтримує кілька супутникових навігаційних систем, зокрема GPS, BeiDou та Galileo. Це дозволяє визначати координати об'єкта та використовувати їх у системах моніторингу та навігації.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

Конструктивно плата містить низку допоміжних апаратних компонентів, що розширюють її функціональні можливості. Зокрема, на платі передбачено слот для TF-карти пам'яті, який використовується для локального зберігання даних, інтерфейс для підключення камери, USB-порт для програмування та налагодження, а також багатоконтактні роз'єми GPIO для підключення різноманітних сенсорів та периферійних пристроїв. Плата також оснащена системою керування живленням, що включає зарядний модуль для літієвих акумуляторів, схему контролю рівня заряду та можливість підключення акумулятора формату 18650 або зовнішнього джерела живлення. Передбачена також можливість заряджання від USB-порту або сонячної панелі, що робить платформу придатною для автономних польових систем моніторингу. Основні технічні характеристики платформи наведено у таблиці 2.1.

Таблиця 2.1 – Характеристики ESP32-S3 SIM7670G

Характеристика	Значення
Архітектура процесора	Xtensa LX7, двоядерний
Тактова частота	до 240 МГц
Оперативна пам'ять SRAM	512 КБ
ROM	384 КБ
Зовнішня PSRAM	2 МБ
Flash-пам'ять	16 МБ
Бездротові інтерфейси	Wi-Fi 2.4 ГГц, Bluetooth LE
Стільниковий зв'язок	LTE Cat-1
Швидкість передавання даних LTE	до 10 Мбіт/с (DL), до 5 Мбіт/с (UL)
Підтримка GNSS	GPS, GLONASS, BeiDou, Galileo
Інтерфейс карти пам'яті	TF / microSD
Інтерфейси підключення	GPIO, UART, SPI, I <sup>2</sup> C
Живлення	5 В (USB) або 3,7–4,2 В (акумулятор)
Логічний рівень сигналів	3,3 В

Принцип роботи платформи полягає у взаємодії мікроконтролера ESP32-S3 з периферійними модулями та комунікаційним модемом. Мікроконтролер здійснює зчитування даних із підключених сенсорів через стандартні інтерфейси (I<sup>2</sup>C, SPI, UART або GPIO), виконує їх оброблення та формує інформаційні пакети для подальшого передавання або зберігання. Зібрані дані можуть записуватися на карту пам'яті для локальної реєстрації або передаватися через модуль стільникового зв'язку до віддаленої серверної чи хмарної платформи. Передавання інформації здійснюється за допомогою стандартних мережевих протоколів, що забезпечує можливість віддаленого моніторингу та аналізу параметрів системи.

Завдяки наявності інтегрованого GNSS-приймача платформа також здатна визначати географічні координати пристрою. Дані позиціонування можуть використовуватися для прив'язки вимірних параметрів до конкретного місця розташування або для відстеження переміщення об'єкта. Така функціональність є особливо важливою для систем телеметрії, мобільних сенсорних мереж та систем моніторингу рухомих об'єктів.

Вибір даної платформи для реалізації комп'ютерної системи моніторингу параметрів руху літального апарату зумовлений рядом переваг. По-перше, вона поєднує у собі мікроконтролер, модуль мобільного зв'язку та GNSS-приймач, що дозволяє значно зменшити кількість окремих апаратних компонентів системи. По-друге, наявність слота для карти пам'яті забезпечує можливість локального зберігання великих обсягів телеметричних даних. По-третє, підтримка мереж LTE дає змогу здійснювати передавання даних у хмарні IoT-системи практично з будь-якого місця, де є покриття мобільного зв'язку.

Крім того, платформа має достатню обчислювальну продуктивність, великий обсяг пам'яті та широкий набір інтерфейсів для підключення сенсорів, що дозволяє використовувати її як центральний обчислювальний вузол системи. Наявність інтегрованої системи керування живленням та підтримка акумуляторного живлення робить її придатною для застосування у автономних або мобільних пристроях. У сукупності ці характеристики роблять платформу ESP32-S3 SIM7670G доцільним і технічно обґрунтованим вибором для реалізації розроблюваної комп'ютерної системи.

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		24

## 2.2.2 Модуль 6-осьового акселерометра і гіроскопа BMI160

Для вимірювання параметрів руху у проєктованій комп'ютерній системі використовується інерційний сенсорний модуль BMI160, який поєднує у собі акселерометр та гіроскоп. Даний сенсор розроблений компанією Bosch Sensortec і належить до класу інерційних вимірювальних модулів. Він призначений для високоточного вимірювання лінійних прискорень та кутових швидкостей руху об'єкта у тривимірному просторі. Завдяки поєднанню двох типів сенсорів в одному корпусі модуль дозволяє здійснювати комплексний аналіз динаміки руху та орієнтації об'єкта, що робить його придатним для систем телеметрії, навігації та керування рухомими об'єктами (рис. 2.3).

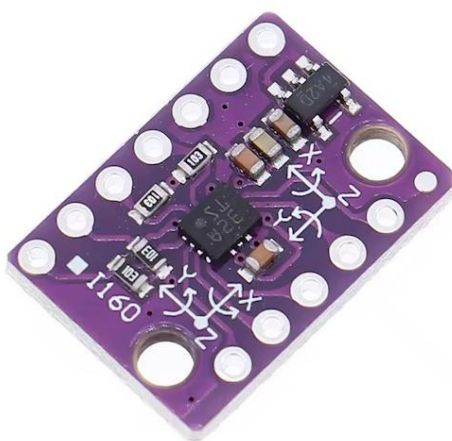


Рисунок 2.3 – Модуль 6-осьового акселерометра і гіроскопа BMI160

Конструктивно сенсор BMI160 реалізований у вигляді високорівнево інтегрованої мікросхеми, яка містить два функціональні вимірювальні блоки: триосьовий акселерометр та триосьовий гіроскоп. Обидва сенсори реалізовані за технологією мікроелектромеханічних систем (MEMS), яка передбачає використання мікроскопічних механічних структур, сформованих безпосередньо на кремнієвій підкладці. Такі структури реагують на механічні впливи, зокрема прискорення або кутове обертання, і перетворюють їх у електричні сигнали. Усередині мікросхеми також розміщені АЦП, цифрові фільтри, блок керування живленням та інтерфейсні схеми, що забезпечують взаємодію сенсора з мікроконтролером.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

Акселерометр призначений для вимірювання лінійного прискорення вздовж трьох ортогональних осей координат. Принцип його роботи базується на вимірюванні зміщення мікромеханічної інерційної маси, підвішеної на пружних елементах. При дії прискорення інерційна маса зміщується, що змінює ємність або електричні параметри вимірювальних елементів, і ці зміни перетворюються у цифровий сигнал. Сенсор дозволяє вибрати різні діапазони вимірювання прискорення, зокрема  $\pm 2g$ ,  $\pm 4g$ ,  $\pm 8g$  та  $\pm 16g$ , що забезпечує можливість використання пристрою як для вимірювання невеликих прискорень, так і для реєстрації інтенсивних динамічних процесів.

Гіроскопічний сенсор призначений для визначення кутової швидкості обертання предмета відносно трьох координатних осей. Його робота ґрунтується на ефекті Коріоліса, який виникає у мікромеханічних резонансних структурах при їх обертанні. Внаслідок цього ефекту виникає додаткове зміщення чутливих елементів, яке фіксується електронними схемами сенсора і перетворюється у цифровий сигнал. Модуль підтримує кілька діапазонів вимірювання кутової швидкості, що дозволяє використовувати його для аналізу як повільних, так і швидких обертальних рухів.

Усі вимірювальні дані формуються у цифровому вигляді з розрядністю 16 біт та передаються до мікроконтролера через стандартні цифрові інтерфейси I<sup>2</sup>C або SPI. Наявність програмованих фільтрів та регульованої частоти вибірки, що може досягати 1600 Гц для акселерометра та 800 Гц для гіроскопа, забезпечує можливість отримання високоточного потоку вимірювальних даних у реальному часі. Крім того, сенсор має вбудований буфер FIFO обсягом 1024 байти, який дозволяє накопичувати дані перед передаванням до мікроконтролера, зменшуючи навантаження на обчислювальну систему.

Додатковою перевагою сенсора є наявність вбудованого механізму апаратних переривань та функцій автономного виявлення руху. Сенсор може автоматично визначати події, такі як різкий рух, орієнтацію пристрою, падіння або удар, та формувати відповідний сигнал переривання. Це дозволяє реалізувати енергоефективні алгоритми роботи системи, оскільки мікроконтролер може перебувати у режимі зниженого енергоспоживання до моменту виникнення події.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

Важливою характеристикою сенсора є його дуже низьке енергоспоживання. У режимі повної роботи акселерометра та гіроскопа споживаний струм становить приблизно 925 мкА, тоді як у режимі очікування він може зменшуватися до кількох мікроампер. Завдяки цьому модуль добре підходить для використання у автономних пристроях з акумуляторним живленням, де важливо мінімізувати енергоспоживання електронної системи. Основні технічні характеристики сенсорного модуля наведено у таблиці 2.2.

Таблиця 2.2 – Характеристики модуля ВМІ160

Характеристика	Значення
Тип сенсора	Інерціальний вимірювальний модуль
Кількість осей	6 (3 акселерометр + 3 гіроскоп)
Розрядність вимірювань	16 біт
Діапазон вимірювання прискорення	$\pm 2g, \pm 4g, \pm 8g, \pm 16g$
Діапазон вимірювання кутової швидкості	$\pm 125^\circ/c, \pm 250^\circ/c, \pm 500^\circ/c, \pm 1000^\circ/c, \pm 2000^\circ/c$
Максимальна частота вибірки акселерометра	до 1600 Гц
Максимальна частота вибірки гіроскопа	до 800 Гц
Інтерфейси обміну даними	I <sup>2</sup> C, SPI
Напруга живлення	1,71–3,6 В
Типове енергоспоживання	$\approx 925$ мкА
Діапазон робочих температур	-40...+85 °С

Використання модуля ВМІ160 у розроблюваній комп'ютерній системі є технічно обґрунтованим з кількох причин. По-перше, модуль забезпечує одночасне вимірювання лінійних прискорень та кутових швидкостей, що дозволяє отримувати повну інформацію про динамічні параметри руху літального апарату. По-друге, сенсор характеризується високою точністю, низьким рівнем шуму та можливістю роботи з високою частотою дискретизації, що є важливим для аналізу швидкоплинних процесів руху.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

Крім того, компактні розміри сенсора, низьке енергоспоживання та підтримка стандартних цифрових інтерфейсів значно спрощують інтеграцію модуля у мікроконтролерні системи. Наявність вбудованих механізмів буферизації та апаратних переривань також дозволяє підвищити ефективність роботи програмного забезпечення та зменшити навантаження на центральний мікроконтролер. У сукупності ці характеристики роблять модуль ВМІ160 доцільним вибором для реалізації системи реєстрації та аналізу динамічних параметрів руху літального апарату.

### 2.2.3 Давач BMP280

Для вимірювання параметрів навколишнього середовища у проєктованій комп'ютерній системі використовується цифровий барометричний давач BMP280. Даний сенсор належить до класу високоточних цифрових давачів атмосферного тиску і широко використовується у системах навігації, мобільних пристроях, безпілотних літальних апаратах та IoT-системах моніторингу. Основним призначенням сенсора є визначення атмосферного тиску та температури навколишнього середовища з подальшим передаванням цих даних у цифровому вигляді до мікроконтролера. Отримані значення можуть використовуватися для визначення відносної висоти та оцінювання умов середовища (рис. 2.4).

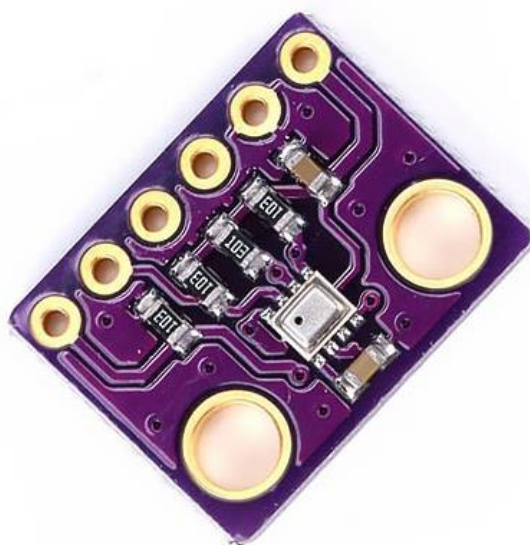


Рисунок 2.4 – Давач BMP280

					КС КРБ 123.151.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

Конструктивно сенсор BMP280 виконаний у вигляді високорівнево інтегрованої мікросхеми, що містить чутливий елемент вимірювання тиску, АЦП, блок оброблення сигналу та цифровий інтерфейс обміну даними. Чутливий елемент сенсора реалізований на основі п'єзорезистивної MEMS-технології, яка використовує мікромеханічну мембрану з інтегрованими тензорезисторами. При зміні атмосферного тиску відбувається деформація цієї мембрани, що призводить до зміни електричного опору тензорезисторів. Отримані аналогові сигнали надходять до внутрішнього АЦП, де вони перетворюються у цифрові значення, які потім обробляються вбудованим сигнальним процесором. Така конструкція забезпечує високу точність, хорошу лінійність та стабільність вимірювань протягом тривалого часу.

Сенсор BMP280 вимірює абсолютний атмосферний тиск у діапазоні приблизно від 30 до 110 кПа, що відповідає типовим умовам атмосферного тиску на поверхні Землі. Крім того, сенсор має вбудований температурний давач, який використовується як для вимірювання температури, так і для температурної компенсації вимірювань тиску. Це дозволяє підвищити точність результатів навіть при значних змінах температури навколишнього середовища. Дані вимірювань формуються у цифровому вигляді та можуть передаватися до мікроконтролера через стандартні інтерфейси I<sup>2</sup>C або SPI, що значно спрощує інтеграцію сенсора у мікроконтролерні системи.

Важливою особливістю сенсора є наявність вбудованого цифрового фільтра, який дозволяє зменшити вплив короточасних коливань атмосферного тиску та електричних завад на результати вимірювання. Крім того, сенсор підтримує різні режими роботи та параметри передискретизації, що дозволяє оптимізувати співвідношення між точністю вимірювання, швидкістю отримання даних та енергоспоживанням пристрою. У режимі низького енергоспоживання сенсор споживає лише кілька мікроампер струму, що робить його придатним для використання у портативних та автономних пристроях із живленням від акумулятора.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдяки високій роздільній здатності вимірювання тиску сенсор BMP280 може використовуватися для визначення відносної висоти з точністю до декількох метрів. Це робить його особливо корисним у системах моніторингу руху літальних апаратів, де важливо отримувати інформацію про зміну висоти польоту. Поєднання даних барометричного сенсора з інформацією, отриманою від інерціальних датчиків та GNSS-модуля, дозволяє підвищити точність визначення параметрів руху об'єкта та забезпечити більш повний аналіз його динаміки. Основні технічні характеристики сенсора наведено у таблиці 2.3.

Таблиця 2.3 – Характеристики давача BMP280

Характеристика	Значення
Тип сенсора	Цифровий барометричний сенсор тиску
Діапазон вимірювання тиску	30–110 кПа
Точність вимірювання	приблизно $\pm 0,1$ кПа
Розрядність вихідних даних	до 16–20 біт
Діапазон робочих температур	$-40 \dots +85$ °C
Напруга живлення	1,71–3,6 В
Інтерфейси обміну даними	I <sup>2</sup> C, SPI
Споживаний струм (типовий)	$\approx 2$ –3 мкА у режимі вимірювання

Використання сенсора BMP280 у розроблюваній комп'ютерній системі є технічно обґрунтованим з кількох причин. По-перше, він забезпечує достатньо високу точність вимірювання атмосферного тиску, що дозволяє визначати зміну висоти польоту та аналізувати параметри руху літального апарату. По-друге, сенсор характеризується низьким енергоспоживанням і компактними розмірами, що робить його придатним для використання у портативних пристроях. По-третє, наявність цифрових інтерфейсів I<sup>2</sup>C та SPI значно спрощує інтеграцію сенсора з мікроконтролерною платформою системи.

Крім того, сенсор BMP280 широко використовується у сучасних IoT-системах, мобільних пристроях та навігаційних модулях, що підтверджує його

					КС КРБ 123.151.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

надійність та ефективність у задачах вимірювання атмосферного тиску. Завдяки поєднанню високої точності, низького енергоспоживання та простоти інтеграції цей сенсор є доцільним вибором для реалізації проєктованої комп'ютерної системи моніторингу параметрів руху літального апарату.

#### 2.2.4 OLED дисплей

Для забезпечення взаємодії користувача з комп'ютерною системою та відображення поточних параметрів її роботи використовується графічний 0.96" OLED дисплей. Він є компактним графічним модулем, який широко застосовується у мікроконтролерних пристроях, IoT-системах, вимірювальних приладах та портативній електроніці. Основною функцією дисплея у складі проєктованої системи є відображення службової інформації, стану пристрою та окремих параметрів, отриманих від сенсорів, що дозволяє здійснювати оперативний контроль роботи системи без використання зовнішніх пристроїв (рис. 2.5).



Рисунок 2.5 – OLED дисплей

OLED-дисплей побудований на основі технології органічних світлодіодів. На відміну від рідкокристалічних дисплеїв, у яких використовується підсвітка, в OLED-матрицях кожен піксель є елементом, який світиться. Під дією

					КС КРБ 123.151.00.00 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

електричного струму органічний напівпровідниковий матеріал випромінює світло, що дозволяє формувати зображення без додаткового джерела освітлення. Завдяки цьому OLED-дисплеї характеризуються високою контрастністю, широким кутом огляду та низьким енергоспоживанням. Крім того, відсутність підсвітки дозволяє створювати дуже компактні та тонкі модулі, що є важливою перевагою для портативних електронних пристроїв.

Конструктивно модуль складається з OLED-матриці, драйверної мікросхеми керування та плати з інтерфейсними роз'ємами. У таких дисплеях використовується контролер SSD1306, який виконує функції керування пікселями матриці, формування зображення та обміну даними з мікроконтролером. Контролер містить внутрішню графічну пам'ять, у якій зберігається буфер зображення, після чого він перетворюється у сигнали керування для кожного пікселя дисплея. Це дозволяє мікроконтролеру передавати на дисплей лише графічні дані, тоді як формування сигналів керування здійснюється безпосередньо драйвером дисплея.

Дисплей дозволяє відображати як текстову, так і просту графічну інформацію. Зображення формується у монохромному режимі, при цьому залежно від модифікації дисплея може мати білий, синій або двоколірний (жовто-синій) колір світіння. Діагональ екрана становить 0,96 дюйма, що забезпечує достатню площу для відображення основних параметрів системи, водночас зберігаючи компактні габарити пристрою. Кут огляду дисплея перевищує 160°, завдяки чому інформація добре читається практично з будь-якого напрямку.

Передавання даних між дисплеєм і мікроконтролером здійснюється через послідовний інтерфейс I<sup>2</sup>C, що потребує лише двох сигнальних ліній – SDA (лінія даних) та SCL (лінія тактових імпульсів). Використання інтерфейсу I<sup>2</sup>C значно спрощує підключення дисплея до мікроконтролера та дозволяє економити виводи GPIO. Стандартна адреса пристрою на шині I<sup>2</sup>C зазвичай становить 0x3C або 0x3D, що дозволяє інтегрувати дисплей у систему разом з іншими пристроями, які використовують цю шину.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

Однією з важливих переваг OLED-дисплеїв є низьке енергоспоживання. Оскільки світяться лише активні пікселі, споживаний струм залежить від кількості відображуваних елементів і зазвичай становить близько 20–30 мА, що робить такі дисплеї придатними для використання у системах з акумуляторним живленням. Крім того, дисплей може працювати у широкому діапазоні напруг живлення (3,3–5 В), що забезпечує сумісність із більшістю сучасних мікроконтролерних платформ. Основні технічні характеристики OLED-дисплея наведено у таблиці 2.4.

Таблиця 2.4 – Характеристики OLED дисплея

Характеристика	Значення
Діагональ екрана	0,96"
Роздільна здатність	128 × 64 пікселі
Контролер дисплея	SSD1306
Тип матриці	Пасивна OLED-матриця
Інтерфейс підключення	I <sup>2</sup> C
Адреса I <sup>2</sup> C	0x3C або 0x3D
Напруга живлення	3,3–5 В
Кут огляду	понад 160°
Робоча температура	–40...+70 °С

Використання OLED-дисплея у розроблюваній комп'ютерній системі є технічно обґрунтованим з кількох причин. По-перше, дисплей має компактні розміри та невелику масу, що важливо для портативних електронних пристроїв. По-друге, висока контрастність і широкий кут огляду забезпечують хорошу читабельність інформації навіть при різних умовах освітлення. По-третє, використання інтерфейсу I<sup>2</sup>C дозволяє мінімізувати кількість використовуваних контактів мікроконтролера та спростити апаратну реалізацію системи.

Крім того, низьке енергоспоживання та простота програмної інтеграції з мікроконтролерними платформами роблять даний дисплей оптимальним засобом відображення інформації у системах моніторингу та телеметрії.

## 2.3 Електрична принципова схема пристрою

Електрична принципова схема розроблюваної системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату визначає склад апаратних компонентів системи та способи їх електричного з'єднання. На рисунку 2.3 наведено електричну принципову схему системи, яка відображає підключення основних функціональних модулів. До складу системи входять мікроконтролерна платформа, інерцій сенсорний модуль, барометричний сенсор, графічний дисплей, елементи керування, джерело живлення та допоміжні пасивні елементи. Усі модулі об'єднані навколо центрального обчислювального вузла – мікроконтролерної платформи, яка виконує функції збору, оброблення, зберігання та передавання даних.

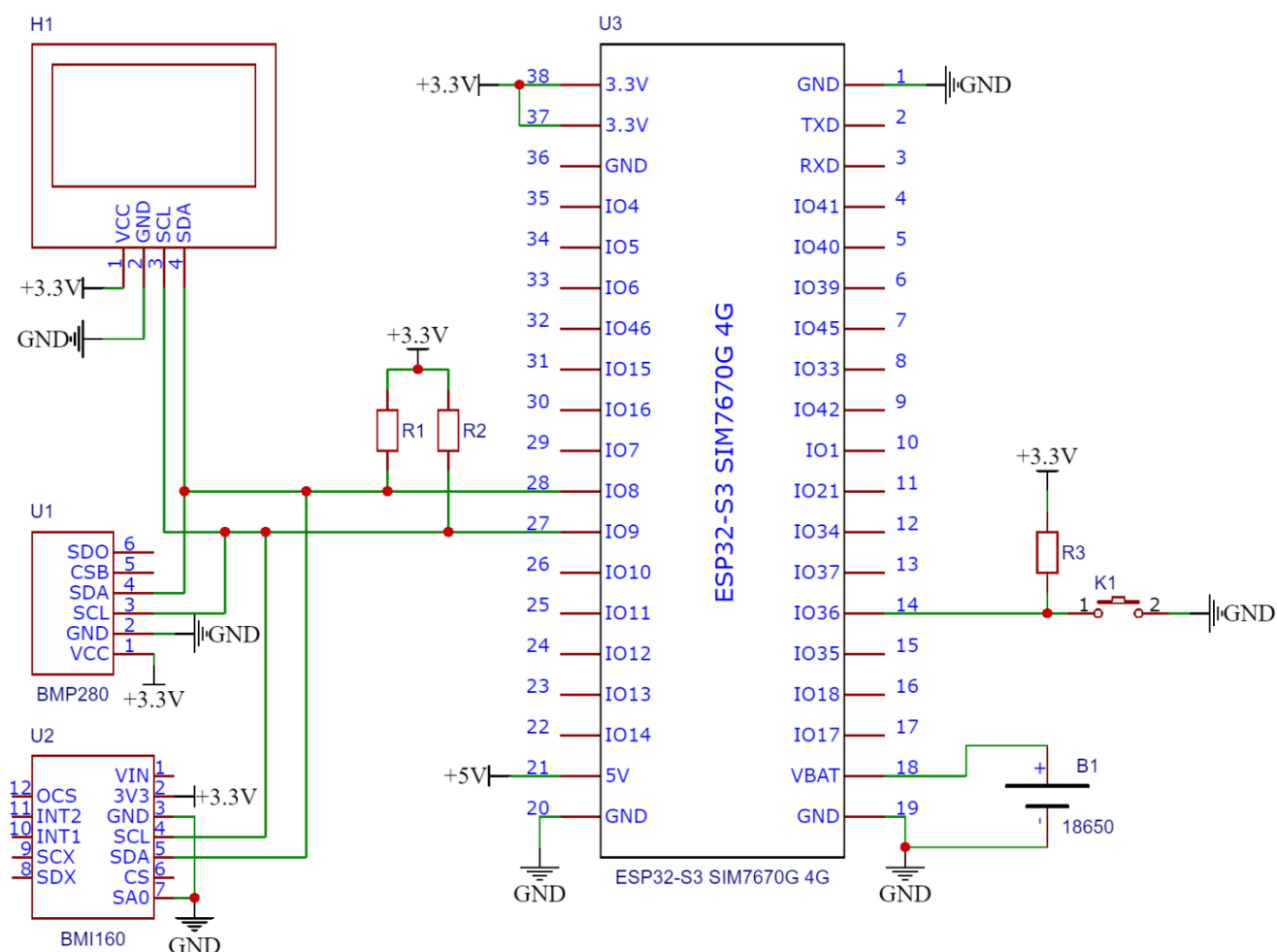


Рисунок 2.11 – Електрична принципова схема пристрою для IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

					КС КРБ 123.151.00.00 ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

Центральним елементом схеми є мікроконтролерна платформа ESP32-S3 SIM7670G, яка виконує функції керування всією системою. До її виводів підключаються сенсорні модулі, засоби відображення інформації та елементи керування. Живлення плати здійснюється від літій-іонного акумулятора з номінальною напругою 3,7 В, який підключається до відповідного роз'єму акумуляторного живлення плати. Вбудована схема керування живленням забезпечує стабілізацію напруги та заряджання акумулятора при підключенні зовнішнього джерела живлення через USB-інтерфейс. Усі периферійні модулі системи живляться від стабілізованої напруги 3,3 В, яка формується на платі мікроконтролера.

Для вимірювання динамічних параметрів руху використовується модуль 6-осьового акселерометра та гіроскопа BMI160. Даний модуль підключається до мікроконтролера через цифровий інтерфейс I<sup>2</sup>C. Лінія передачі даних SDA модуля підключається до контакту GPIO8 мікроконтролера, який використовується як лінія даних інтерфейсу I<sup>2</sup>C, тоді як лінія тактових імпульсів SCL підключається до контакту GPIO9, що виконує функцію тактового сигналу цієї шини. Для забезпечення коректної роботи інтерфейсу I<sup>2</sup>C використовуються підтягувальні резистори опором приблизно 4,7 кОм, які підключаються між лініями SDA та SCL і шиною живлення 3,3 В.

Барометричний сенсор BMP280 також підключається до мікроконтролера через інтерфейс I<sup>2</sup>C. Це дозволяє використовувати спільну шину даних разом із сенсором BMI160. Контакт SDA сенсора підключається до тієї ж лінії SDA (GPIO8) мікроконтролера, а контакт SCL – до лінії SCL (GPIO9). Завдяки використанню різних адрес на шині I<sup>2</sup>C обидва сенсори можуть працювати одночасно, передаючи дані до мікроконтролера без конфліктів на шині.

Для відображення поточної інформації про стан системи використовується графічний OLED-дисплей з інтерфейсом I<sup>2</sup>C. Лінії обміну даними дисплея також підключаються до спільної шини I<sup>2</sup>C. Таким чином, дисплей працює на тій самій шині I<sup>2</sup>C, що і сенсорні модулі, що дозволяє зменшити кількість використовуваних контактів мікроконтролера та спростити апаратну структуру системи.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

Для забезпечення керування режимами роботи пристрою у схемі передбачено кнопку керування. Один контакт кнопки підключається до входу GPIO36 мікроконтролера, а другий – до загальної шини заземлення. Для забезпечення стабільного логічного рівня на вході мікроконтролера використовується підтягувальний резистор опором приблизно 10 кОм, підключений між входом GPIO36 та шиною живлення 3,3 В. У нормальному стані на вході мікроконтролера підтримується логічна одиниця, а при натисканні кнопки відбувається з'єднання входу із землею, що формує логічний нуль і дозволяє мікроконтролеру зафіксувати подію натискання кнопки.

Мікроконтролер здійснює періодичне опитування сенсорних модулів через інтерфейс I<sup>2</sup>C, отримує дані про прискорення, кутову швидкість, атмосферний тиск та температуру, після чого виконує їх оброблення. Частина інформації відображається на OLED-дисплеї у вигляді текстових або графічних даних, а інша частина зберігається на карті пам'яті або передається через модуль мобільного зв'язку до віддаленої IoT-платформи.

Запропонована електрична принципова схема забезпечує взаємодію всіх функціональних компонентів системи, включаючи сенсорні модулі, засоби відображення інформації, елементи керування та підсистему живлення.

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		36

## РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

### 3.1 Алгоритм роботи системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

Основною метою алгоритму роботи проєктованої системи є забезпечення безперервного збору даних про динамічні параметри руху літального апарату, їх оброблення, візуалізацію та передачу до віддаленої IoT-платформи для подальшого аналізу і зберігання.

Робота системи починається з подачі живлення від акумулятора на мікроконтролерну платформу. Після ввімкнення пристрою відбувається ініціалізація апаратних ресурсів мікроконтролера, зокрема налаштування тактової частоти процесора, конфігурація портів введення-виведення, запуск внутрішніх периферійних модулів та ініціалізація програмних бібліотек, необхідних для роботи сенсорів і засобів відображення інформації. На цьому етапі також виконується перевірка стану основних компонентів системи та готовності їх до роботи. Після завершення ініціалізації система переходить у режим очікування запуску основних функцій.

Наступним етапом алгоритму є ініціалізація сенсорних модулів, підключених до мікроконтролера через інтерфейс I<sup>2</sup>C. Спочатку виконується ініціалізація модуля 6-осьового акселерометра і гіроскопа, під час якої задаються режими вимірювання прискорення та кутової швидкості, встановлюється частота вибірки та активуються внутрішні фільтри оброблення сигналів. Після цього ініціалізується барометричний сенсор, у якому встановлюється режим вимірювання атмосферного тиску та температури. Паралельно здійснюється ініціалізація OLED-дисплея, на якому відображається інформація про успішне завантаження системи та її поточний стан.

					<i>КС КРБ 123.151.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Верхола Б.А.</i>			<i>Практична частина</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					<i>37</i>	<i>19</i>
<i>Рецензент</i>		<i>Деркач М.В.</i>				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>						
<i>Зав. каф.</i>		<i>Осужівська Г.М.</i>						

Після завершення процедури ініціалізації система переходить до основного циклу роботи (рис. 3.1).

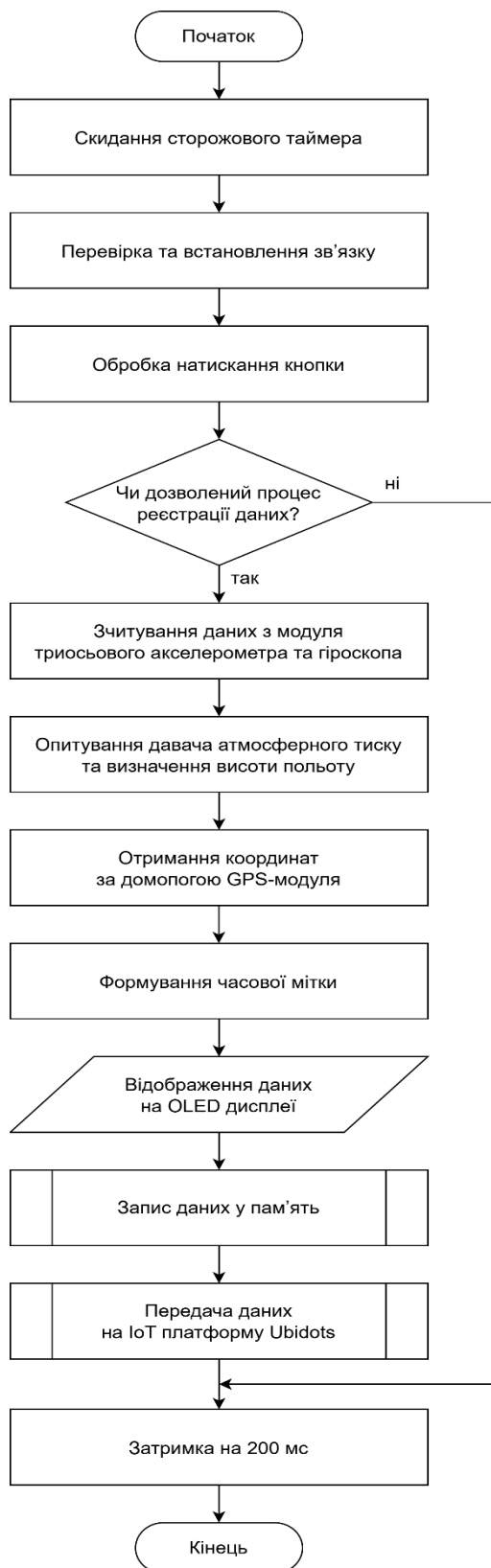


Рисунок 3.1 – Блок-схема алгоритму роботи основної підпрограми системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

У межах цього циклу мікроконтролер періодично виконує опитування сенсорних модулів для отримання вимірювальних даних. З модуля акселерометра і гіроскопа зчитуються значення прискорення по трьох осях координат та кутової швидкості обертання. Ці параметри характеризують динаміку руху літального апарату і можуть використовуватися для визначення його орієнтації, зміни швидкості та маневрів під час польоту. Одночасно з барометричного сенсора зчитуються значення атмосферного тиску та температури, на основі яких може бути визначена відносна висота польоту.

Отримані дані проходять етап первинного оброблення. На цьому етапі виконується фільтрація вимірювальних сигналів, усереднення результатів вимірювань та перетворення сирих значень у фізичні величини. Далі обчислені параметри формуються у структуру даних, яка містить часову мітку, значення прискорення, кутової швидкості, атмосферного тиску, температури та інші службові параметри системи. Ця інформація використовується для подальшої реєстрації та передавання.

Після оброблення даних система виконує їх відображення на OLED-дисплеї. На екрані можуть відображатися основні параметри стану системи, зокрема значення прискорення по трьох осях, атмосферний тиск, температура та стан з'єднання з мережею. Відображення інформації на дисплеї дозволяє оператору контролювати роботу системи в реальному часі без використання додаткових зовнішніх пристроїв.

Одночасно з візуалізацією дані зберігаються у локальній пам'яті системи або на карті пам'яті, що встановлюється у слот плати мікроконтролера. Це дозволяє реалізувати функцію реєстрації параметрів польоту навіть у випадку відсутності мережевого з'єднання. Дані записуються у вигляді структурованих записів, які можуть містити часову мітку та відповідні значення вимірюваних параметрів.

На наступному етапі алгоритму виконується передавання даних до віддаленої IoT-платформи через вбудований модуль мобільного зв'язку. Мікроконтролер формує пакет телеметричної інформації та передає його через мережу мобільного зв'язку до хмарного сервера. У хмарній платформі отримані

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
						<i>39</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

дані зберігаються у базі даних і можуть використовуватися для подальшого аналізу, побудови графіків або віддаленого моніторингу параметрів польоту.

В алгоритмі також передбачено оброблення сигналу від кнопки керування. При натисканні кнопки мікроконтролер може змінювати режим роботи системи, наприклад запускати або зупиняти процес реєстрації даних або активувати режим передавання накопиченої інформації.

## 3.2 Розробка програмного забезпечення

### 3.2.1 Ініціалізація апаратних та програмних компонентів системи

Підпрограма `setup()` виконує початкову ініціалізацію всіх апаратних і програмних компонентів комп'ютерної системи IoT-моніторингу та є критично важливим етапом запуску мікроконтролера. У цій підпрограмі здійснюється налаштування інтерфейсів обміну даними, ініціалізація периферійних пристроїв, підготовка до збору телеметричної інформації та встановлення з'єднання з хмарною платформою (рис. 3.2).

На початку виконання підпрограми виконується налаштування сторожового таймера (`Watchdog`). Це дозволяє автоматично перезапустити систему у випадку зависання або виникнення критичних помилок під час роботи, що підвищує надійність функціонування пристрою. Наступним етапом є конфігурація цифрових входів мікроконтролера. Зокрема, кнопка керування підключається у режимі `INPUT_PULLUP`. Після цього ініціалізується інтерфейс I<sup>2</sup>C за допомогою виклику `Wire.begin()`, який використовується для зв'язку з сенсорними модулями та дисплеєм.

Далі виконується ініціалізація OLED-дисплея, підключеного за інтерфейсом I<sup>2</sup>C. За допомогою функції `display.begin` задається адреса дисплея та режим живлення, після чого налаштовуються параметри відображення, зокрема розмір і колір шрифту. Після очищення екрану на дисплей виводиться службове повідомлення, що сигналізує про початок процесу ініціалізації системи.

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
						40
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

```

void setup() {
  Serial.begin(115200);
  esp_task_wdt_init(WDT_TIMEOUT, true);
  esp_task_wdt_add(NULL);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
  Wire.begin(SDA_PIN, SCL_PIN);
  display.begin(SSD1306_SWITCHCAPVCC, OLED_ADDR);
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.clearDisplay();
  display.println("System init");
  display.display();
  initSensors();
  initSD();
  initModem();
  thing["telemetry"] >> [](pson &out) {
    out["ax"] = data.ax;
    out["ay"] = data.ay;
    out["az"] = data.az;
    out["gx"] = data.gx;
    out["gy"] = data.gy;
    out["gz"] = data.gz;
    out["temperature"] = data.temperature;
    out["pressure"] = data.pressure;
    out["altitude"] = data.altitude;
    out["lat"] = data.latitude;
    out["lon"] = data.longitude;
  };
  display.println("System ready");
  display.display();
}

```

Рисунок 3.2 – Лістинг коду підпрограми setup()

Після налаштування інтерфейсів відображення викликаються підпрограми ініціалізації основних функціональних модулів системи. Функція `initSensors()` забезпечує підготовку сенсорних модулів до роботи, включаючи акселерометр, гіроскоп та барометричний давач. Функція `initSD()` відповідає за ініціалізацію карти пам'яті та підготовку файлової системи для збереження телеметричних даних. У свою чергу, функція `initModem()` виконує запуск GSM/LTE-модуля, встановлення з'єднання з мережею та підготовку до передачі даних у хмарне середовище.

Важливим етапом є налаштування обміну даними з хмарною IoT-платформою Thingier.io. У коді визначається ресурс `telemetry`, який реалізовано у вигляді лямбда-функції. У межах цього ресурсу формується структурований набір

					КС КРБ 123.151.00.00 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

даних, що включає значення прискорення, кутових швидкостей, температури, тиску, висоти, а також географічні координати. Такий підхід дозволяє організувати зручну передачу телеметричної інформації у форматі, придатному для подальшої обробки та візуалізації на стороні хмарної платформи.

На завершальному етапі підпрограми на дисплей виводиться повідомлення «System ready», що свідчить про успішне завершення ініціалізації всіх компонентів системи та готовність пристрою до виконання основного алгоритму роботи.

### 3.2.2 Реалізація основного циклу функціонування системи у підпрограмі loop

Підпрограма loop() реалізує безперервний цикл роботи мікроконтролерної системи та забезпечує виконання основного алгоритму збору, оброблення і передавання телеметричних даних. Дана підпрограма виконується циклічно протягом усього часу роботи пристрою та координує взаємодію між усіма функціональними модулями системи (рис. 3.3).

```
void loop() {  
    esp_task_wdt_reset();  
    checkButton();  
    reconnectLTE();  
  
    if(systemEnabled) {  
        readIMU();  
        readBMP();  
        readGPS();  
        data.timestamp = millis();  
        updateDisplay();  
        logData();  
        sendTelemetry();  
    }  
  
    delay(200);  
}
```

Рисунок 3.3 – Лістинг коду підпрограми loop()

На початку кожної ітерації циклу здійснюється скидання сторожового таймера за допомогою функції esp\_task\_wdt\_reset(). Це необхідно для підтвердження коректної роботи програми та запобігання автоматичному перезавантаженню мікроконтролера у випадку зависання. Таким чином

					КС КРБ 123.151.00.00 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечується підвищення надійності функціонування системи в умовах тривалої безперервної експлуатації.

Далі викликається підпрограма `checkButton()`, яка відповідає за оброблення стану кнопки керування. У межах цієї функції здійснюється перевірка натискання кнопки та, у разі її активації, змінюється логічний стан змінної `systemEnabled`. Це дозволяє користувачу вручну вмикати або вимикати основні функції системи без необхідності перезавантаження пристрою.

Наступним кроком є виклик функції `reconnectLTE()`, яка забезпечує контроль стану мережевого з'єднання та автоматичне відновлення зв'язку у разі його втрати. У випадку відсутності підключення до мережі мобільного зв'язку відбувається повторна ініціалізація модема та спроба встановлення з'єднання. Такий підхід дозволяє підвищити стійкість системи до зовнішніх факторів і забезпечити безперервність передавання даних до хмарного середовища.

Основна частина функціоналу підпрограми виконується за умови, що система знаходиться в активному стані, що визначається значенням змінної `systemEnabled`. У цьому випадку послідовно викликаються функції збору даних з сенсорів. Зокрема, підпрограма `readIMU()` зчитує дані з інерційного вимірювального модуля, `readBMP()` — з барометричного сенсора, а `readGPS()` — координати та висоту з GPS-модуля. Отримані значення зберігаються у структурі телеметричних даних, що використовується для подальшої обробки.

Після зчитування сенсорних даних формується часовий штамп за допомогою функції `millis()`, який записується у відповідне поле структури телеметрії. Це дозволяє фіксувати момент часу кожного вимірювання та забезпечує можливість подальшого аналізу динаміки зміни параметрів.

Далі виконується оновлення інформації на OLED-дисплеї за допомогою функції `updateDisplay()`. На екран виводяться актуальні значення вимірюваних параметрів, що забезпечує зручний локальний контроль стану системи. Після цього викликається функція `logData()`, яка відповідає за запис телеметричної інформації на карту пам'яті у вигляді файлу, що дозволяє накопичувати історію вимірювань для подальшого аналізу.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

Наступним етапом є передавання даних до хмарної IoT-платформи за допомогою функції `sendTelemetry()`. У процесі виконання цієї функції здійснюється відправлення структурованої телеметричної інформації, що дозволяє реалізувати віддалений моніторинг параметрів руху літального апарату.

Завершальним етапом кожної ітерації циклу є затримка виконання за допомогою функції, що визначає інтервал між послідовними циклами обробки даних. Це дозволяє оптимізувати навантаження на мікроконтролер та узгодити частоту оновлення даних з вимогами до системи.

### 3.2.3 Зчитування та фільтрація даних інерційного вимірювального модуля

Підпрограма `readIMU()` реалізує процес зчитування та оброблення даних з інерційного вимірювального модуля, який включає акселерометр і гіроскоп. Отримані дані використовуються для визначення динамічних параметрів руху літального апарату, зокрема прискорень та кутових швидкостей, що є основою для аналізу його просторового положення та руху (рис. 3.4).

```
void readIMU() {
    int16_t axRaw, ayRaw, azRaw;
    int16_t gxRaw, gyRaw, gzRaw;
    BMI160.readAccelerometer(axRaw, ayRaw, azRaw);
    BMI160.readGyro(gxRaw, gyRaw, gzRaw);
    float ax = axRaw * 0.000061;
    float ay = ayRaw * 0.000061;
    float az = azRaw * 0.000061;
    data.ax = kalmanX.update(ax);
    data.ay = kalmanY.update(ay);
    data.az = kalmanZ.update(az);
    data.gx = gxRaw * 0.00763;
    data.gy = gyRaw * 0.00763;
    data.gz = gzRaw * 0.00763;
}
```

Рисунок 3.4 – Лістинг коду підпрограми `readIMU()`

На початковому етапі підпрограми оголошуються змінні для збереження «сирих» значень, отриманих з акселерометра та гіроскопа. Ці значення представлені у вигляді 16-бітних цілих чисел і відповідають цифровому виходу

					КС КРБ 123.151.00.00 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

сенсора. Далі за допомогою функцій `BMI160.readAccelerometer()` та `BMI160.readGyro()` виконується зчитування даних по трьох осях координат для кожного типу вимірювання. Отримані значення відображають миттєві показники прискорення та кутової швидкості у вигляді цифрових кодів.

Наступним кроком є перетворення «сирих» даних акселерометра у фізичні одиниці вимірювання. Для цього використовується коефіцієнт масштабування, який враховує чутливість сенсора. Значення множаться на коефіцієнт 0,000061, що дозволяє отримати результати у одиницях прискорення (g). Аналогічно виконується перетворення даних гіроскопа, де застосовується коефіцієнт 0,00763 для отримання значень у градусах за секунду.

З метою підвищення точності вимірювань та зменшення впливу шумів у підпрограмі використовується фільтрація даних за допомогою фільтра Калмана. Для кожної осі акселерометра передбачено окремий екземпляр фільтра, який обробляє відповідні вимірювання. Отримані значення прискорення передаються до методу `update()` фільтра, у результаті чого формується згладжене значення, яке зменшує випадкові коливання та підвищує стабільність показників.

Після оброблення дані записуються у структуру телеметрії. Зокрема, відфільтровані значення прискорення по осях X, Y та Z зберігаються у відповідних полях, а значення кутових швидкостей записуються без додаткової фільтрації. Це дозволяє використовувати отриману інформацію для подальшого аналізу, відображення на дисплеї, збереження у пам'яті та передавання до хмарної платформи.

### 3.2.4 Зчитування та оброблення координат з GPS-модуля

Підпрограма `readGPS()` призначена для отримання та оброблення навігаційних даних, що надходять від GPS-модуля, інтегрованого у GSM/LTE-модем. Вона забезпечує визначення географічних координат і висоти розташування літального апарату, що є важливими параметрами для аналізу його руху та положення у просторі (рис. 3.5).

					КС КРБ 123.151.00.00 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

```

void readGPS() {
    while (SerialAT.available()) {
        | gps.encode(SerialAT.read());
    }
    if (gps.location.isValid()) {
        | data.latitude = gps.location.lat();
        | data.longitude = gps.location.lng();
    }
    if (gps.altitude.isValid()) {
        | data.gps_altitude = gps.altitude.meters();
    }
}

```

Рисунок 3.5 – Лістинг коду підпрограми readGPS()

На початковому етапі роботи підпрограми виконується перевірка наявності даних у буфері послідовного інтерфейсу зв'язку з модемом. За допомогою циклу while здійснюється поетапне зчитування байтів даних, які надходять від GPS-приймача. Кожен зчитаний символ передається до функції gps.encode(), яка виконує декодування NMEA-повідомлень. Таким чином формується структурована інформація про місцезнаходження, швидкість, висоту та інші параметри, що передаються GPS-модулем.

Після завершення оброблення вхідного потоку виконується перевірка валідності отриманих координат за допомогою методу gps.location.isValid(). У випадку, якщо дані є коректними, значення широти та довготи зчитуються з об'єкта GPS і записуються у відповідні поля структури телеметрії. Це дозволяє використовувати отримані координати для відображення на дисплеї, передачі до хмарної платформи та побудови траєкторії руху літального апарату.

Аналогічно виконується перевірка валідності значення висоти за допомогою функції gps.altitude.isValid(). Якщо дані є достовірними, висота над рівнем моря зчитується у метрах за допомогою методу gps.altitude.meters() і записується у структуру телеметричних даних. Це значення може використовуватися як додатковий параметр для аналізу руху або для порівняння з обчисленою висотою.

### 3.2.5 Відображення телеметричної інформації на OLED-дисплеї

Підпрограма updateDisplay() реалізує функцію візуалізації основних телеметричних параметрів системи на OLED-дисплеї, забезпечуючи користувачу

					КС КРБ 123.151.00.00 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

можливість оперативного контролю стану літального апарату без необхідності підключення до зовнішніх засобів моніторингу. Вона відповідає за формування текстового інтерфейсу та оновлення інформації у реальному часі (рис. 3.6).

```
void updateDisplay() {  
    display.clearDisplay();  
    display.setCursor(0,0);  
    display.print("AX:");  
    display.println(data.ax);  
    display.print("AY:");  
    display.println(data.ay);  
    display.print("AZ:");  
    display.println(data.az);  
    display.print("ALT:");  
    display.println(data.altitude);  
    display.print("LAT:");  
    display.println(data.latitude,4);  
    display.print("LON:");  
    display.println(data.longitude,4);  
    display.display();  
}
```

Рисунок 3.6 – Лістинг коду підпрограми updateDisplay()

На початковому етапі виконання підпрограми здійснюється очищення екрана за допомогою функції `display.clearDisplay()`. Це необхідно для запобігання накладанню нових даних на попередні та забезпечення коректного відображення актуальної інформації. Після цього встановлюється початкова позиція курсора у верхньому лівому куті дисплея за допомогою функції `display.setCursor(0,0)`, що визначає координати початку виведення тексту.

Далі послідовно виконується виведення значень прискорення по трьох осях координат. Для кожної осі використовується пара функцій `display.print()` та `display.println()`, що дозволяє вивести текстову мітку параметра та його числове значення з переходом на новий рядок. Аналогічним чином відображається обчислене значення висоти, отримане на основі барометричних вимірювань. Це дозволяє оцінювати зміну висоти літального апарату у процесі його руху.

Наступним кроком є відображення географічних координат, отриманих з GPS-модуля. Значення широти та довготи виводяться з обмеженням кількості

					КС КРБ 123.151.00.00 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

десяткових знаків до чотирьох, що забезпечує достатню точність позиціонування при збереженні компактності відображення. Використання такого формату дозволяє оптимально розмістити інформацію на екрані з обмеженою роздільною здатністю.

Після формування всього текстового вмісту викликається функція `display.display()`, яка передає сформований буфер даних на дисплей і здійснює фактичне оновлення зображення. Таким чином забезпечується синхронне відображення всіх параметрів, що були підготовлені у межах підпрограми.

### 3.2.6 Реалізація запису телеметричних даних

Підпрограма `logData()` відповідає за періодичний запис поточних телеметричних даних у файл. На початку її виконання реалізовано механізм обмеження частоти запису. Це дозволяє зменшити навантаження на SD-карту та оптимізувати використання ресурсів системи (рис. 3.7).

```
void logData() {
    if(millis() - lastLog < logInterval) return;
    lastLog = millis();
    logFile = SD.open("/flight_log.csv", FILE_APPEND);
    if(logFile) {
        logFile.print(data.timestamp);
        logFile.print(",");
        logFile.print(data.ax); logFile.print(",");
        logFile.print(data.ay); logFile.print(",");
        logFile.print(data.az); logFile.print(",");
        logFile.print(data.gx); logFile.print(",");
        logFile.print(data.gy); logFile.print(",");
        logFile.print(data.gz); logFile.print(",");
        logFile.print(data.temperature); logFile.print(",");
        logFile.print(data.pressure); logFile.print(",");
        logFile.print(data.altitude); logFile.print(",");
        logFile.print(data.latitude); logFile.print(",");
        logFile.println(data.longitude);
        logFile.close();
    }
}
```

Рисунок 3.7 – Лістинг коду підпрограми `logData()`

					КС КРБ 123.151.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		48

У випадку, коли інтервал часу перевищено, відбувається оновлення змінної lastLog та відкриття файлу /flight\_log.csv у режимі додавання. Далі послідовно виконується запис усіх параметрів структури data у форматі CSV із використанням розділювача у вигляді коми. Кожне значення виводиться за допомогою функцій print() або println(), що забезпечує формування коректного рядка даних. Завершення рядка здійснюється за допомогою функції println(), яка додає символ переходу на новий рядок.

Після завершення запису файл закривається, що є важливим для гарантування цілісності даних та запобігання їх втраті у разі раптового вимкнення живлення. Таким чином, реалізований механізм журналювання забезпечує надійне накопичення телеметричної інформації з можливістю її подальшої обробки та аналізу.

### 3.2.7 Передача телеметричних даних та обробка керуючої кнопки

Підпрограма sendTelemetry() реалізує важливі функціональні складові системи, а саме передавання телеметричних даних на віддалену IoT-платформу. Її використання забезпечує дистанційний моніторинг стану системи (рис. 3.8).

```
void sendTelemetry() {  
    if(millis() - lastSend < sendInterval) return;  
  
    lastSend = millis();  
  
    thing.handle();  
}
```

Рисунок 3.8 – Лістинг коду підпрограми sendTelemetry()

Підпрограма sendTelemetry() відповідає за періодичну передачу зібраних даних. На початку її виконання реалізовано перевірку інтервалу часу між послідовними відправленнями даних. Для цього застосовується функція millis(), яка повертає час роботи мікроконтролера, та змінна lastSend, що зберігає момент останньої передачі. Якщо різниця між поточним часом і lastSend є меншою за

					КС КРБ 123.151.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

заданий інтервал `sendInterval`, підпрограма завершує виконання достроково, не здійснюючи передачу. Такий підхід дозволяє уникнути надмірного мережевого навантаження та оптимізувати використання енергоресурсів.

У випадку, коли заданий часовий інтервал перевищено, змінна `lastSend` оновлюється, після чого викликається метод `thing.handle()`. Даний виклик забезпечує обробку обміну даними з IoT-платформою, включаючи відправлення телеметрії, яка попередньо була описана у відповідному ресурсі. Таким чином, передача даних здійснюється у неблокуючому режимі та інтегрується у загальний цикл виконання програми.

### 3.3 Налаштування хмарної IoT платформи для системи моніторингу

Для реалізації функцій віддаленого моніторингу, зберігання та візуалізації даних у розроблюваній комп'ютерній системі було використано хмарну IoT-платформу `Thingier.io`. Дана платформа забезпечує можливість підключення мікроконтролерних пристроїв до хмарного сервера, передачу телеметричних даних у реальному часі, їх зберігання у базі даних та подальшу візуалізацію за допомогою графіків і інформаційних панелей. Використання цієї платформи дозволило організувати віддалений доступ до параметрів руху літального апарату та реалізувати функції централізованого моніторингу роботи системи.

На початковому етапі налаштування у веб-інтерфейсі платформи було створено новий пристрій, який відповідає мікроконтролерній платформі системи. Для цього у розділі керування пристроями було сформовано запис пристрою з унікальним ідентифікатором. Під час створення пристрою система автоматично згенерувала параметри автентифікації, зокрема ідентифікатор пристрою та секретний ключ доступу. Ці параметри були використані у програмному коді мікроконтролера для встановлення захищеного з'єднання з сервером платформи. Завдяки цьому забезпечувалась ідентифікація пристрою у системі та можливість передавання даних у відповідний обліковий запис користувача (3.9).

					КС КРБ 123.151.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

Device Details

**Device Configuration**

Device Id

Device Credentials

**Device Information**

Device Name

Device Description

**Advanced Options**

Asset Type

Asset Group

Product

Enabled

Sleep Timeout  seconds

Disconnect Timeout  seconds

Рисунок 3.9 – Створення пристрою в Thinger.io

Наступним етапом було налаштування каналів обміну даними між мікроконтролером та хмарною платформою. У ПЗ мікроконтролера було визначено ресурс телеметрії, який передає структуровані дані про параметри руху літального апарату. До складу передаваних даних входили значення прискорення по трьох осях координат, кутова швидкість обертання, температура, атмосферний тиск, розрахована висота польоту, а також координати місцезнаходження, отримані з GPS-модуля. Передавання цих параметрів здійснювалось у вигляді структурованого набору даних, який платформа автоматично обробляла та зберігала у базі даних.

Після налаштування передавання телеметрії було виконано конфігурацію системи зберігання даних. У середовищі платформи було створено так званий Data Bucket, який використовується для довготривалого зберігання телеметричної інформації. Цей механізм дозволяє накопичувати історію вимірювань і забезпечує можливість подальшого аналізу параметрів руху літального апарату. До створеного сховища було прив'язано ресурс телеметрії пристрою, що дозволило автоматично записувати отримані дані у базу даних після кожного передавання.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

На наступному етапі було реалізовано засоби візуалізації отриманої інформації. Для цього у веб-інтерфейсі платформи було створено інформаційну панель моніторингу, яка відображає основні параметри роботи системи у реальному часі. До складу панелі було додано декілька віджетів, зокрема графіки для відображення прискорення по осях, значень атмосферного тиску та висоти польоту, а також інформаційні індикатори для відображення поточних координат літального апарату. Крім того, було реалізовано карту місцезнаходження, яка дозволяє відображати координати, отримані з GPS-модуля, на географічній карті.

Для перевірки коректності роботи системи було виконано тестове підключення мікроконтролера до хмарної платформи через мережу мобільного зв'язку. Після встановлення з'єднання система почала передавати телеметричні дані на сервер, де вони автоматично відображалися у веб-інтерфейсі платформи. Під час тестування було перевірено коректність відображення усіх параметрів, стабільність передавання даних та правильність їх збереження у базі даних. Отримані результати підтвердили працездатність системи віддаленого моніторингу та можливість використання платформи для довготривалого збору і аналізу телеметричної інформації.

У результаті налаштування хмарної платформи було створено повноцінне середовище для віддаленого моніторингу параметрів руху літального апарату. Реалізована конфігурація забезпечує приймання телеметричних даних від мікроконтролерної системи, їх зберігання у базі даних та наочне відображення у вигляді графіків і інформаційних панелей, що значно підвищує ефективність аналізу роботи розробленої комп'ютерної системи.

### 3.4 Результати моделювання та тестування системи

На етапі розроблення комп'ютерної системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату було виконано моделювання її апаратної структури та перевірку працездатності основних функціональних вузлів у спеціалізованому середовищі схемотехнічного проектування Circuit Designer.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

Використання цього програмного середовища дозволило створити цифрову модель системи, перевірити правильність підключення компонентів та оцінити коректність взаємодії між окремими модулями ще до виготовлення фізичного прототипу.

На першому етапі моделювання у середовищі розробки було створено структурну модель пристрою, що включала основні апаратні компоненти системи: мікроконтролерну платформу, інерційний сенсорний модуль, барометричний сенсор, графічний дисплей, кнопку керування та джерело живлення. Для кожного компонента було задано відповідні електричні з'єднання відповідно до розробленої електричної принципової схеми (рис. 3.10).

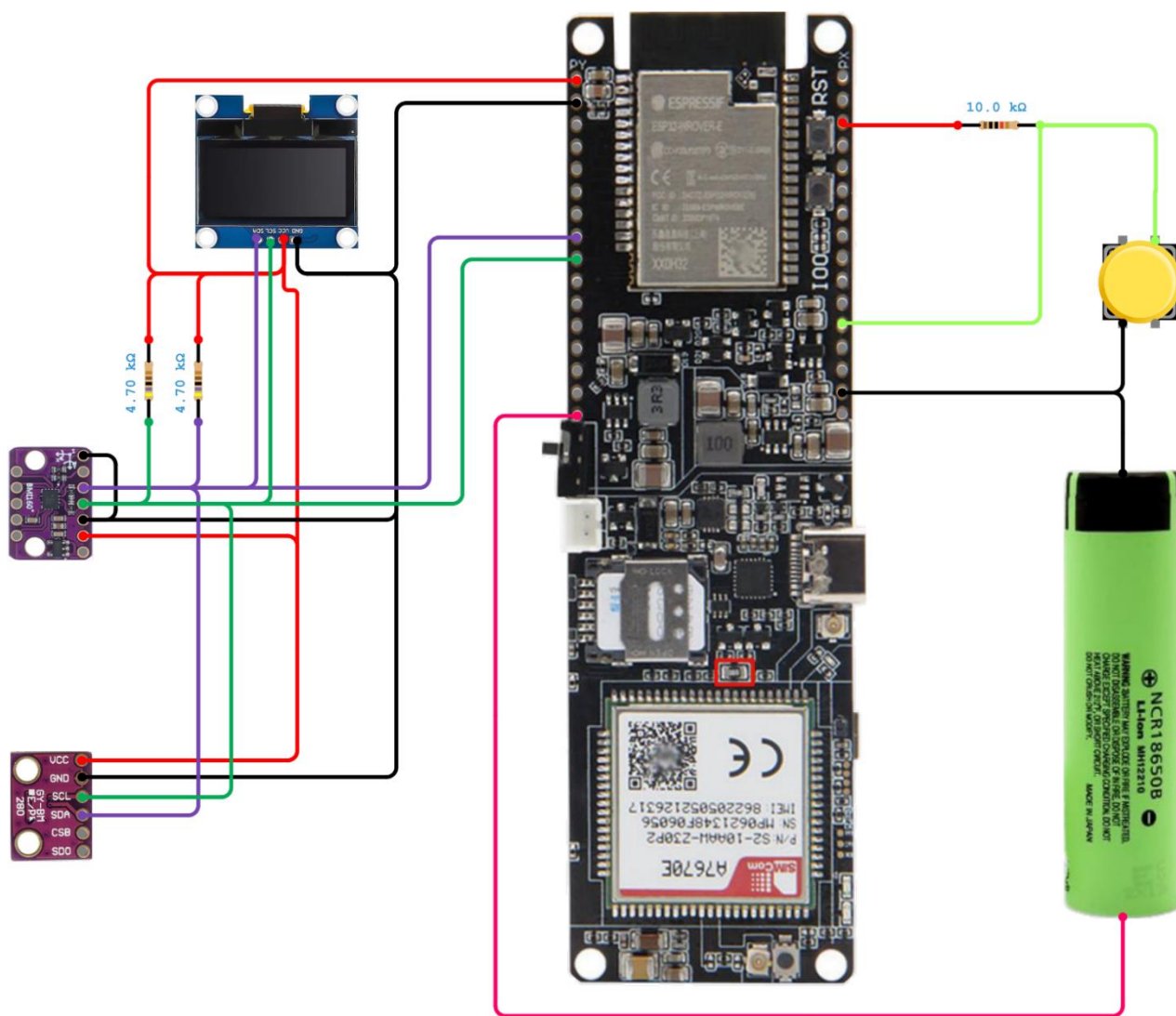


Рисунок 3.10 – Модель системи моніторингу та реєстрації динамічних параметрів руху літального апарату

Наступним етапом стало тестування взаємодії сенсорних модулів із мікроконтролером. У моделі системи було змодельовано роботу інерційного вимірювального модуля, який генерує дані про прискорення та кутову швидкість обертання. Паралельно моделювалася робота барометричного сенсора, який формує значення атмосферного тиску та температури. У середовищі моделювання було перевірено процес зчитування цих даних через інтерфейс I<sup>2</sup>C та їх подальше оброблення мікроконтролером. Результати моделювання показали, що обмін даними між пристроями здійснюється коректно, а обрана структура з'єднань забезпечує стабільну роботу сенсорних модулів.

Після перевірки роботи сенсорних компонентів було змодельовано функціонування модуля відображення інформації. У процесі тестування перевірялася передача даних до дисплея та правильність відображення інформації про стан системи. Було підтверджено, що дисплей коректно відображає значення вимірюваних параметрів, а також службову інформацію про роботу пристрою. Крім того, у процесі моделювання було перевірено роботу кнопки керування, яка використовується для перемикання режимів функціонування системи.

Після завершення етапу моделювання було виконано розроблення фізичного прототипу системи на основі вибраних апаратних компонентів. На цьому етапі було здійснено монтаж усіх елементів відповідно до розробленої електричної принципової схеми. Сенсорні модулі, дисплей та інші периферійні пристрої було підключено до мікроконтролерної платформи.

Після складання апаратної частини було виконано програмування мікроконтролера та завантаження розробленого ПЗ у пам'ять пристрою. Далі було проведено початкове тестування системи, під час якого перевірялася працездатність кожного функціонального модуля окремо. Спочатку було перевірено роботу інерційного сенсора, який вимірює прискорення та кутову швидкість. Під час тестування здійснювалося переміщення та обертання пристрою, що дозволило оцінити реакцію сенсора на зміну просторового положення. Отримані значення вимірювань успішно зчитувалися мікроконтролером і відображалися на дисплеї.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

На наступному етапі тестування було перевірено роботу барометричного сенсора. Для цього проводилися вимірювання атмосферного тиску та температури навколишнього середовища. Отримані дані використовувалися для обчислення відносної висоти, що є одним із важливих параметрів для систем моніторингу польоту. Результати вимірювань показали, що сенсор забезпечує достатню точність визначення параметрів навколишнього середовища.

Окремий етап тестування був присвячений перевірці роботи системи передавання даних. Мікроконтролер встановлював з'єднання з мережею мобільного зв'язку та передавав телеметричну інформацію на хмарну IoT-платформу (рис. 3.11).

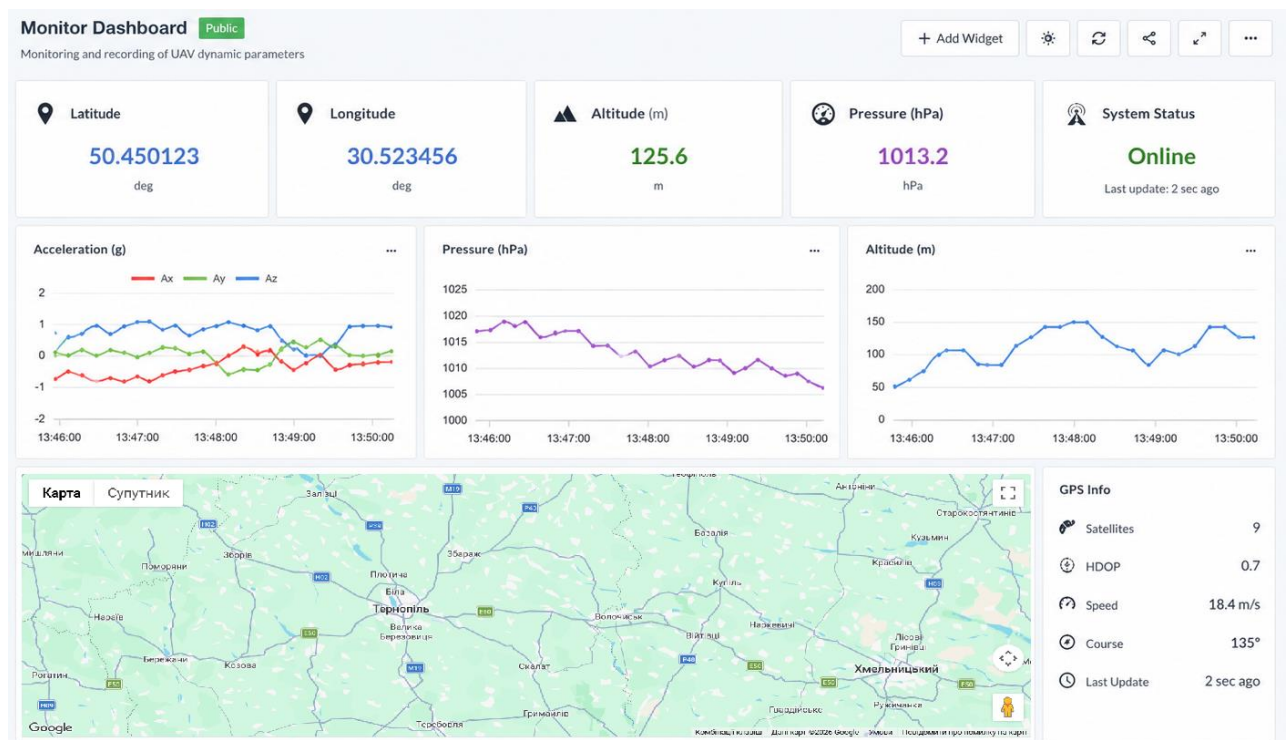


Рисунок 3.11 – Результати тестування системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату

У процесі тестування було підтверджено, що система стабільно передає дані про прискорення, кутову швидкість, атмосферний тиск, температуру та координати місцезнаходження. Отримані дані відображалися у веб-інтерфейсі хмарної платформи у вигляді графіків та інших елементів візуалізації.

## РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

### 4.1 Стихійні лиха та їх класифікація

Стихійні дії сил природи, поки що не повною мірою підвладні людині та щорічно завдають державі і населенню величезних збитків. Стихійні лиха – це такі явища природи, що викликають екстремальні ситуації, порушують нормальну життєдіяльність населення, роботу безлічі об'єктів. Стихійні лиха є трагедією для будь-якої держави. Через стихійні лиха страждає економіка країни, бо при цьому руйнуються виробничі підприємства, знищуються матеріальні цінності, гинуть люди.

Стихійні лиха – небезпечні природні явища раптового походження, хоча іноді і прогнозовані за допомогою метеорології, але на інтенсивність яких люди впливати не можуть. Їх можна класифікувати: за швидкістю переміщення – землетруси, зсуви, цунамі, снігопади, ожеледі – швидкі; підвищення рівня води в ріках через інтенсивні опади або танення снігу, льоду (повені), звільнення внутрішньої енергії Землі, виверження вулканів – повільні. Часто виникають потужні, високошвидкісні потоки повітря через швидкий перепад значень атмосферного тиску (урагани, смерчі, циклони). Стихійні лиха речовинного характеру можуть ініціювати виникнення різноманітних полів, які негативно впливають на здоров'я, самопочуття людини.

Стихійні явища часто виникають в комплексі, що значно посилює їх негативний вплив. Небезпечні природні явища визначаються трьома основними групами процесів – ендогенні, екзогенні та гідрометеорологічні. Стихійні лиха, які характерні для України, за структурою можна поділити на прості, що включають один елемент – наприклад, сильний вітер, зсув або землетрус та складні. Вони складаються з декількох процесів однієї групи або кількох груп. Найбільші збитки спричиняють повені – 40 %, на другому місці – циклони (20 %), на третьому –

					<i>КС КРБ 123.151.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Верхола Б.А.</i>			<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Паламар А.М.</i>					56	7
<i>Консульт.</i>		<i>Сенчишин В.С.</i>				<i>ТНТУ, каф. КС, гр. СІ-41</i>		
<i>Н. Контр.</i>		<i>Луцик Н.С.</i>						
<i>Зав. каф.</i>		<i>Осυχівська Г.М.</i>						

посухи та землетруси (15 %). Деякі стихійні лиха (пожежі, обвали, зсуви і навіть землетруси) можуть виникати в результаті дій самих людей, тобто мають антропогенне походження, але наслідки їх завжди є діями сил природи. Для кожного стихійного лиха характерна наявність властивих йому вражаючих чинників, що несприятливо впливають на стан здоров'я, життя людини [31].

Причинами стихійних лих можуть бути:

- швидке переміщення речовини (землетрусу, зсуви);
- вивільнення внутрішньоземної енергії (вулканічна діяльність, землетруси);
- підвищення рівня вод річок, ставків і морів (повені, цунамі);
- вплив надзвичайно сильного вітру (урагани, торнадо, циклони).

Важливо своєчасно провести роботи, спрямовані на локалізацію природного лиха, щоб зменшити зони руйнувань, звести до мінімуму кількість загиблих та постраждалих.

В Україні найчастіше спостерігаються такі надзвичайні ситуації природного характеру [31]:

- небезпечні геологічні явища (зсуви, обвали, осипки, просадки земної поверхні);
- небезпечні метеорологічні явища (зливи, урагани, сильні снігопади, сильний град, ожеледь);
- небезпечні гідрологічні явища (повені, паводки);
- природні пожежі лісових та торф'яних масивів;
- масові інфекції та хвороби людей, тварин, рослин.

В останні роки кількість стихійних лих в Україні та в світі в цілому значно збільшилася. Найчастіше в Україні виникають такі природні катастрофи як землетруси, повені, посухи (на Півдні України), лісові пожежі в літню пору року, снігові замети, зсуви поверхні.

Землетрус — це сильні коливання земної кори, викликані тектонічними причинами, які призводять до руйнування споруд, пожеж та людських жертв. Коливання земної кори передається сейсмічними хвилями. Найсильніші вони в

					КС КРБ 123.151.00.00 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

гіпоцентрі. З віддаленням від нього хвилі слабшають. Гіпоцентр, або осередок землетрусу, — місце в глибинах Землі, де зсуваються земні (гірські) породи. Епіцентр — точка на поверхні землі, що знаходиться прямо над гіпоцентром.

Основними характеристиками землетрусів є: глибина осередку, магнітуда та інтенсивність енергії на поверхні землі. Глибина осередку землетрусу зазвичай перебуває в межах від 10 до 30 км, в деяких випадках вона може бути значно більша. Магнітуда характеризує загальну енергію землетрусу і є логарифмом максимальної амплітуди зміщення ґрунту в мікронах, яка вимірюється за сейсмограмою на відстані 100 км від епіцентру. Магнітуда за Ріхтером вимірюється від 0 до 9 (найсильніший землетрус). Інтенсивність — це показник наслідків землетрусів, який характеризує розмір збитків, кількість жертв та характер сприйняття людьми психогенного впливу.

Є серйозні підстави вважати, що масштабність впливу лиха й катастроф на соціальні, економічні, політичні та інші процеси сучасного суспільства та їх драматизм вже перевищили такий рівень, який дозволяв ставитися до них як до локальних збоїв у функціонуванні державних та громадських структур [31].

Перед людиною та громадськістю в ХХІ столітті вимальовується нова мета – глобальна безпека. Досягти цього можна, в першу чергу, за допомогою зміни світогляду людини, а також покращення системи профілактичних заходів у боротьбі зі стихійними лихами, а саме: вдосконалення рятувальних служб та рятувальної техніки, проведення попереджувальних заходів та пропагандистської роботи з громадянами щодо правил поведінки та дій під час стихійних лих. Це допоможе в майбутньому зменшити кількість загиблих та постраждалих від природних катастроф, а також зменшить матеріальні збитки, що були завдані стихійним лихом.

Природні лиха з часом нікуди не зникнуть. Будуть виникати землетруси в геологічно активних районах, будуть виникати повені, а штормові припливи стануть, раз у раз затопляти морські узбережжя, не обійдеться і без пожеж. Людина безсила запобігти природним процесам, але тільки в наших силах зменшити кількість жертв і матеріальних втрат.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

## 4.2 Розробка заходів щодо зниження рівня шуму на робочому місці

Шум – будь-який несприятливий звук, який діє на людину. Це сполучення звуків різної частоти та інтенсивності. З фізичної точки зору звук являє собою механічне хвильове коливання пружного середовища, яке супроводжується виникненням надлишкового тиснення, яке сприймається людиною через слуховий орган у діапазоні частот (16 – 20) кГц.

За своїм походженням шум поділяється на [32]:

- механічний;
- аерогідродинамічний (виникає в наслідок нестаціонарних процесів в рідинах або газах);
- електромагнітний (в наслідок дії змінних електромагнітних сил, які призводять до коливань деяких вузлів та елементів машин та механізмів).

Основними фізичними характеристиками шуму є звуковий тиск (P), інтенсивність звуку (I) та частота (f). Орган слуху здатний сприймати звуки з інтенсивністю 10-12 – 102 Вт/м<sup>2</sup> (при частоті f = 1000 Гц). Ці значення є порогові (щодо звукового тиснення), але вони залежать від частоти звука і від індивідуальних особливостей людини.

Шум – загально біологічний подразник (в деяких умовах може впливати на всі органи та системи людини). Шум має вплив на різні відділи головного мозку, порушуючи нормальні процеси нервової діяльності. Характерним впливом є: стомлюваність, апатія, роздратованість, погіршення пам'яті, слабкість.

Шум великої інтенсивності призводить до змін у серцево-судинній системі, що супроводжуються порушеннями тону та ритму серцевих скорочень, та до змін артеріального кров'яного тиску.

Під впливом шуму порушується нормальне функціонування шлунка (зменшується кількість шлункового соку, змінюється кислотність, виникає гастрит та язва шлунку).

					КС КРБ 123.151.00.00 ПЗ	Арк.
						59
Змн.	Арк.	№ докум.	Підпис	Дата		

В останні роки було встановлено вплив шуму на орган зору (зменшується стійкість ясного бачення та гострота зору, погіршується кольоросприймання). Шум призводить до порушення процесів обміну [32].

Переривчастий та імпульсний шум порушують точність виконання операцій, погіршують процес сприймання та засвоєння інформації. Найбільш чутливими до шуму є такі операції: складання та збір інформації, мислення.

Під дією шуму відбувається зменшення продуктивності праці на підприємстві, збільшення кількості браку, створення небезпечних ситуацій. Тому заходи по боротьбі з шумом мають велике економічне та оздоровче значення.

Шкідливість шуму як фактору виробничого середовища і середовища життєдіяльності людини приводить до необхідності обмежувати його рівні. Санітарно-гігієнічне нормування шумів здійснюється згідно (ДСН 3.3.6.037-99), в основному, двома способами – методом граничних спектрів (ГС) і методом рівня звуку [33].

Метод граничних спектрів, який застосовують для нормування постійного шуму, передбачає обмеження рівнів звукового тиску в октавних смугах із середніми геометричними частотами 31,5; 63; 125; 250; 500; 1000; 2000; 4000 і 8000 Гц. Сукупність цих граничних октавних рівнів називають граничним спектром. Позначають той чи інший граничний спектр рівнем його звукового тиску на частоті 1000 Гц. Метод рівнів звуку застосовують для орієнтовній гігієнічній оцінки постійного шуму та визначення непостійного шуму, наприклад, зовнішнього шуму транспортних засобів, міського шуму. Вимірюють рівень звуку в децибелах А (дБА) шумоміром із стандартною коректованою частотною характеристикою, в якому за допомогою відповідних фільтрів знижена чутливість на низьких та високих частотах.

Захист від шуму досягається шумобезпечною технікою, застосуванням:

- засобів та методів колективного захисту;
- засобів індивідуального захисту;
- будівельно-акустичними методами.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Для створення шумобезпечної техніки на стадії її проектування використовуються методи, які знижують шум в самому джерелі. Їх підрозділяють на методи [34]:

- що знижують збудження шуму;
- що знижують звуковипромінюючу здатність джерела.

Зниження аеродинамічного шуму досягається покращенням аеродинамічних характеристик конструктивних елементів, наприклад – плазмотрона.

Зниження електромагнітного шуму досягається вибором оптимальних розмірів, технологій і якості виготовлення магнітопроводів, підбором значень магнітної індукції.

Для зниження звуковипромінюючої здатності джерела, його поверхню покривають демпфіруючими матеріалами, які мають велике внутрішнє тертя. Найбільш розповсюджені жорсткі покриття з пружних в'язких матеріалів (лінолеума, мастик).

До числа архітектурно-плануючих засобів із зниження шуму в цехах належать: раціональне планування територій підприємства (при яких об'єкти, які потребують захисту від шуму – лабораторії, КБ, ВЦ – максимально віддалені від шумових устаткувань і приміщень), раціональне по поверхове планування будинків і розміщення устаткування, що генерує шум, розміщення робочих місць і організація транспортних потоків, створення шумозахисних зон [34].

Акустичні засоби – це засоби захисту від шуму на шляху його розповсюдження. До них у першу чергу відносяться звукоізоляція і звукопоглинання. Метод звукоізоляції засновано на відбитті звукової хвилі, яка падає на огорожу (стіни, кожухи, екрани).

Цікавим та принципово новим методом зниження шуму є метод активного шумопригнічення. Він базується на створенні “антизвуку” тобто рівного за рівнем і протилежного по фазі звуку. В наслідок інтерференції основного звуку та антизвуку, в деяких місцях приміщення можливо створити зони тиші. Цей метод є ефективним для пригнічення тональних шумів.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						61
Змн.	Арк.	№ докум.	Підпис	Дата		

В місці, де треба зменшити шум, встановлюють мікрофон, сигнал від якого перетворюється в електричний, надходить на фазоінвертор і далі на підсилювач та динаміки, що встановлюють визначеним чином.

Для захисту працівників від шуму також використовуються засоби індивідуального захисту, які дозволяють знизити рівень сприймання звуку на 10–45 дБ, причому найбільш значні глушіння спостерігаються в області високих частот.

Засоби індивідуального захисту поділяють на:

- протишумові укладки (закривають вушну раковину зовні);
- протишумові навушники (перекривають слуховий прохід);
- протишумові каски і шоломи (закривають всю голову і застосовуються у сполученні з навушниками і протишумовими костюмами).

Під час створення системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату було забезпечено дотримання заходів щодо зниження рівня шуму на робочому місці.

					<i>КС КРБ 123.151.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		62

## ВИСНОВКИ

У результаті виконання кваліфікаційної роботи розроблено систему IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату, яка забезпечує збір, оброблення, зберігання та передавання телеметричних даних до хмарного середовища.

У процесі виконання роботи проведено аналіз існуючих підходів і технологій моніторингу параметрів руху літальних апаратів, а також обґрунтовано вибір апаратних і програмних засобів для реалізації системи. На основі цього аналізу було визначено структуру системи та сформовано її функціональні вимоги.

Розроблено структурну та електричну принципову схеми системи, що включає мікроконтролерну платформу, інерційний сенсорний модуль, барометричний давач, модуль відображення інформації, систему живлення та інші допоміжні елементи.

У межах роботи було розроблено алгоритм функціонування системи, який забезпечує ініціалізацію апаратних компонентів, періодичний збір сенсорних даних, їх оброблення, обчислення висоти польоту, відображення інформації на дисплеї, локальне збереження даних та передачу телеметрії до хмарної IoT-платформи.

Також було створено ПЗ для мікроконтролера, яке реалізує всі функції системи, включаючи оброблення даних з акселерометра, гіроскопа та барометричного сенсора, фільтрацію вимірювань, визначення координат, передачу телеметричної інформації до хмарного сервера та реєстрацію даних у локальній пам'яті.

На завершальному етапі роботи було виконано моделювання та експериментальне тестування системи. Результати тестування підтвердили коректність роботи апаратної та програмної частин, стабільність передачі телеметричних даних і можливість ефективного використання розробленої системи для моніторингу та реєстрації параметрів руху літального апарату.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						63
Змн.	Арк.	№ докум.	Підпис	Дата		

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Filipovic Z., Dikic G. Design and implementation of PCM/FM telemetry systems for aircraft and missile testing during flight test procedure. In International conference on Aerospace Sciences and Aviation Technology, 2011. Vol. 14, No. 4. P. 1-15.
2. Qin H., Wang H., Xie J., Tang Y., Li P. Development of flight data recorder for general aviation aircraft with real time data transmission. In 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT), 2021. P. 162-167.
3. Vasylenko M.P., Karpyuk I.S. Telemetry system of unmanned aerial vehicles. Electronics and control systems, 2018. No. 3, P. 95-100.
4. Lee J.K., Park E.J., Robinovitch S.N. Estimation of attitude and external acceleration using inertial sensor measurement during various dynamic conditions. IEEE transactions on instrumentation and measurement, 2012. 61(8). P. 2262-2273.
5. 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.
6. Лупенко С.А., Пасічник В.В., Тиш Є.В. Комп'ютерна логіка. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 354 с.
7. Буров Є., Митник М. Комп'ютерні мережі. (у 2-х томах). Львів, Магнолія, 2018. 740 с.
8. Yatsyshyn V., Pastukh O., Kukharska V., Palamar A., Kulikov S. Method and tool of detecting software architecture patterns in the process of computer systems development. CEUR Workshop Proceedings, 4th International Workshop on Information Technologies: Theoretical and Applied Problems (ITTAР 2024), Ternopil, Ukraine, Opole, Poland, October 23-25, 2024. Vol. 3896. P. 12-24.
9. Palamar M., Yavorska M., Palamar A., Strembitskyi M. Modeling and Research of Satellite Antenna Adjustment Process for Earth Remote Sensing. 2022 IEEE 2nd Ukrainian Microwave Week (UkrMW), Kharkiv, Ukraine, November 14-18, 2022. P. 317-320.

					КС КРБ 123.151.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		64

10. Palamar M., Nakonetchnyi Y., Palamar A., Strembitskyi M., Apostol Y. Modernization of the azimuth drive design for the antenna system. Scientific Journal of TNTU, Ternopil, Ukraine, 2025. Vol. 117, No 1, P. 54–61.

11. Strembitskyi M., Yavorska M., Palamar A., Kochan R., Yeromenko V. A comparative study of bug algorithms for robot navigation. CEUR Workshop Proceedings, 3rd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2025), Ternopil, Ukraine, June 11-12, 2025. Vol. 4057 P. 312-321.

12. Palamar A., Voloskyi V., Kramar O., Kramar T., Stankevych O., Yatsyshyn V. Information computer system with a virtual tour for cultural heritage preservation of the Zbarazh Castle Museum's exhibition hall. CEUR Workshop Proceedings, The 3rd International Workshop on Social Communication and Information Activity in Digital Humanities (SCIA 2024), Lviv, Ukraine, October 31, 2024. Vol. 3851.

13. Романов Д.В., Осухівська Г.М., Паламар А.М. Система управління зовнішнім освітленням на основі Інтернету речей. Актуальні задачі сучасних технологій : збірник тез доповідей X міжнародної науково-практичної конференції молодих учених та студентів (Тернопіль, 24-25 листопада 2021 року), Тернопіль: ТНТУ, 2021. С. 120.

14. Palamar A., Stadnyk M., Palamar M. Adaptive PID regulation method of uninterruptible power supply battery charge current based on artificial neural network. Scientific Journal of TNTU, Ternopil, Ukraine, 2022. Vol. 107, No 3. P. 5–13.

15. Оконський М.В., Лупенко С.А., Паламар А.М. Інформаційно-вимірювальна система для контролю метеорологічних параметрів на основі Інтернету речей. Матеріали ІХ науково-технічної конференції "Інформаційні моделі, системи та технології" Тернопільського національного технічного університету імені Івана Пулюя (Тернопіль, 8–9 грудня 2021 року), Тернопіль: ТНТУ, 2021. С. 118.

16. Palamar M., Pasternak Y., Palamar A., Poikhalo A. Precision tracking of the trajectory LEO satellite by antenna with induction motors in the control system. Proceedings of the 2017 IEEE 9th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS 2017), Bucharest, Romania, September 21–23, 2017. Vol. 2. P. 1051–1055.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						65
Змн.	Арк.	№ докум.	Підпис	Дата		

17. Оконський М.В., Лупенко С.А., Паламар А.М. Комп'ютерна система для моніторингу метеорологічних параметрів на основі ІоТ. Актуальні задачі сучасних технологій : збірник тез доповідей Х міжнародної науково-практичної конференції молодих учених та студентів (Тернопіль, 24–25 листопада 2021 року), Тернопіль: ТНТУ, 2021. С. 112.

18. Palamar M., Horyn T., Palamar A., Batuk V. Method of calibration MEMS accelerometer and magnetometer for increasing the accuracy determination angular orientation of satellite antenna reflector. Scientific Journal of TNTU, Ternopil, Ukraine, 2022. Vol. 108, No 4. P. 79–88.

19. Palamar A., Pettai E. Microgrid for the Department of Electrical Drives and Power Electronics. 8th International Symposium "Topical Problems in the Field of Electrical and Power Engineering" and "Doctoral School of Energy and Geotechnology II" (January 11-16, 2010), Pärnu, Estonia, 2010. P. 54-61.

20. Palamar A. Methods and means of increasing the reliability of computerized modular uninterruptible power supply system. Scientific Journal of TNTU, Ternopil, Ukraine, 2020. Vol. 99, No 3. P. 133–141.

21. Palamar A., Palamar M. Fire Safety Monitoring System Based on Internet of Things. CEUR Workshop Proceedings, 2023. 1st International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2023), Ternopil, Ukraine, June 14-16, 2023. Vol. 3468. P. 164-172.

22. Паламар М., Пастернак Ю., Паламар А. Дослідження динамічних похибок системи прецизійного керування антеною з асинхронним електроприводом. Вісник ТНТУ, Тернопіль: ТНТУ, 2014. Вип. 76, № 4. С. 164–173.

23. Palamar A., Karpinskyu M., Vodovozov V. Design and Implementation of a Digital Control and Monitoring System for an AC/DC UPS. 7th International Conference-Workshop «Compatibility and Power Electronics» (CPE 2011), June 1-3, 2011. P. 173–177.

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.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						66
Змн.	Арк.	№ докум.	Підпис	Дата		

25. Palamar A., Karpinskyy M. Control of an Uninterruptible Power Supply in a DC Microgrid System. 10th International Symposium Symposium "Topical Problems in the Field of Electrical and Power Engineering" and "Doctoral School of Energy and Geotechnology II" (January 10-15, 2011), Pärnu, Estonia, 2011. P. 80-84.

26. Stadnyk M., Palamar A. Project management features in the cybersecurity area. Scientific Journal of TNTU, Ternopil, Ukraine, 2022. Vol. 106, No 2. P. 54–62.

27. Погребенник В.Д., Клим Г.І., Бордун І.М., Пташник В.В., Паламар А.М. Системи оперативного контролю інтегральних параметрів водного середовища. Т. 2. Елементи комп'ютерних систем оперативного контролю: колективна монографія. Житомир: Видавничий дім «Бук-Друк», 2021. 180 с.

28. Voloshchuk A., Velychko D., Osukhivska H., Palamar A. Computer system for energy distribution in conditions of electricity shortage using artificial intelligence. CEUR Workshop Proceedings, 2nd International Workshop on Computer Information Technologies in Industry 4.0 (CITI 2024), Ternopil, Ukraine, June 12-14, 2024. Vol. 3742 P. 66-75.

29. Palamar A. Intelligent control and monitoring module for uninterruptible power supply system. II International Scientific and Practical Conference «Theoretical and Applied Aspects of Device Development on Microcontrollers and FPGAs» (MC&FPGA-2020), Kharkiv, Ukraine. 2020. P. 12-13.

30. Palamar A. Control system simulation by modular uninterruptible power supply unit with adaptive regulation function. Scientific Journal of TNTU, Ternopil, Ukraine, 2020. Vol. 98, No 2. P. 129–136.

31. Желібо Є.П. Безпека життєдіяльності : підручник / В. В. Зацарний. Київ : Каравела, 2023. 344 с.

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

33. ДСН 3.3.6.037-99 «Державні санітарні норми виробничого шуму, ультразвуку та інфразвуку».

34. Мелех Л.В. Безпека життєдіяльності та охорона праці: навч. посіб. / Мелех Л.В. Львів: ЛДУ внутрішніх справ. 2022. 219 с.

					КС КРБ 123.151.00.00 ПЗ	Арк.
						67
Змн.	Арк.	№ докум.	Підпис	Дата		

Додаток А  
Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Івана Пулюя

Кафедра комп'ютерних систем та мереж

«ЗАТВЕРДЖУЮ»

Завідувач кафедрою КС

\_\_\_\_\_ Осухівська Г.М.

“ 2 ” лютого 2026 р.

КОМП'ЮТЕРНА СИСТЕМА ІОТ-МОНІТОРИНГУ ТА РЕЄСТРАЦІЇ  
ДИНАМІЧНИХ ПАРАМЕТРІВ РУХУ ЛІТАЛЬНОГО АПАРАТУ

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. Паламар А.М.

“ 2 ” лютого 2026 р.

«ВИКОНАВЕЦЬ»

Студент групи СІ-41

\_\_\_\_\_ Верхола Б.А.

“ 2 ” лютого 2026 р.

Тернопіль 2026

## 1 Загальні відомості

### 1.1 Повна назва та її умовне позначення

Повна назва теми кваліфікаційної роботи бакалавра: «Комп'ютерна система IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.151.00.00.

### 1.2 Виконавець

Студент групи СІ-41, факультету комп'ютерно-інформаційних систем і програмної інженерії, кафедри комп'ютерних систем та мереж, Тернопільського національного технічного університету імені Івана Пулюя, Верхола Богдан Андрійович.

### 1.3 Підстава для виконання роботи

Підставою для виконання кваліфікаційної роботи бакалавра є наказ по університету № 4/9-188 від «24» квітня 2026 року.

### 1.4 Планові терміни початку та завершення роботи

Плановий термін початку виконання кваліфікаційної роботи бакалавра – 26.01.2026 р.

Плановий термін завершення виконання кваліфікаційної роботи бакалавра – 21.06.2026 р.

## 1.5 Порядок оформлення та пред'явлення результатів роботи

Оформлення технічної документації до кваліфікаційної роботи бакалавра здійснюється згідно діючих вимог вітчизняних та міжнародних стандартів. Технічна документація до кваліфікаційної роботи бакалавра включає в себе текст пояснювальної записки та креслення, які максимально інформативно та стисло відображають основні результати розробки комп'ютерної системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату. Основними регламентними документами при оформленні та пред'явленні результатів проектування є групи діючих стандартів ДСТУ, ISO, ЄСКД та ЕСПД. Пред'явлення результатів кваліфікаційної роботи бакалавра відбувається шляхом захисту роботи на відповідному засіданні ЕК, ілюстрацією основних досягнень за допомогою графічного матеріалу.

## 2 Призначення і цілі створення системи

### 2.1 Призначення системи

Проектована комп'ютерна система призначена для вимірювання, реєстрації та дистанційного моніторингу динамічних параметрів руху літального апарату.

Система повинна забезпечувати автоматизований збір даних із сенсорів руху, їх первинну обробку, передачу на віддалений сервер або IoT-платформу та подальше збереження для аналізу. Основне призначення системи полягає у підвищенні інформативності контролю руху літального апарату, а також у забезпеченні можливості аналізу поведінки об'єкта в різних режимах експлуатації.

## 2.2 Мета створення системи

Метою створення системи є розробка універсального IoT-рішення для моніторингу динамічних параметрів руху літального апарату, яке поєднує функції збору, реєстрації та візуалізації телеметричних даних. Система має забезпечувати надійну роботу в умовах обмежених ресурсів, підтримувати бездротову передачу даних та бути адаптованою до подальшого розширення функціональних можливостей.

## 2.3 Характеристика об'єкту

Об'єктом моніторингу є літальний апарат, зокрема безпілотний або експериментальний апарат малого та середнього класу. Характерними особливостями об'єкта є наявність динамічних рухів у тривимірному просторі, змінні режими польоту, а також обмеження щодо маси, енергоспоживання та габаритів встановлюваного обладнання. Система повинна враховувати специфіку експлуатації літального апарату, включаючи вібраційні навантаження та змінні зовнішні умови.

## 3 Вимоги до системи

### 3.1 Вимоги до системи в цілому

Система IoT-моніторингу повинна забезпечувати безперервний або періодичний збір динамічних параметрів руху, їх обробку та передачу у віддалене сховище даних. Архітектура системи має бути модульною, що дозволяє спрощувати налагодження, обслуговування та модернізацію. Програмне та апаратне забезпечення повинні функціонувати узгоджено, забезпечуючи стабільну та передбачувану роботу системи в цілому.

### 3.1.1 Вимоги до структури та функціонування системи

Структурно система повинна складатися з вимірювального пристрою, обчислювального вузла, модуля бездротового зв'язку та програмного забезпечення для обробки й візуалізації даних. Функціонування системи має ґрунтуватися на циклічному зчитуванні даних із сенсорів, їх фільтрації та нормалізації, формуванні телеметричних пакетів і передачі цих даних на IoT-платформу. Передбачається можливість локального збереження даних у разі тимчасової відсутності зв'язку.

### 3.1.2 Вимоги до способів та засобів зв'язку між компонентами системи

Обмін даними між сенсорами та мікроконтролером повинен здійснюватися з використанням стандартних цифрових інтерфейсів. Для зв'язку між системою та віддаленою IoT-платформою мають застосовуватися бездротові технології передачі даних. Допускається використання таких засобів зв'язку:

- Wi-Fi для передачі даних у зоні покриття локальної мережі;
- стільникового зв'язку для автономної роботи системи;
- стандартних мережевих протоколів IoT для обміну телеметричною інформацією.

### 3.1.3 Вимоги до режимів функціонування системи

Система повинна підтримувати кілька режимів роботи, зокрема режим ініціалізації, робочий режим, режим тестування та сервісний режим. У робочому режимі система здійснює безперервний або періодичний моніторинг параметрів руху. Режим тестування використовується для перевірки працездатності апаратних і програмних компонентів. Перехід між режимами має відбуватися автоматично або за командою користувача.

### 3.1.4 Перспективи розвитку та модернізації системи

Архітектура системи повинна передбачати можливість розширення шляхом підключення додаткових сенсорів, впровадження нових алгоритмів обробки даних та інтеграції з іншими інформаційними системами. Перспективним напрямом розвитку є використання методів інтелектуального аналізу даних та машинного навчання для прогнозування стану літального апарату.

### 3.1.5 Вимоги до надійності системи

Система повинна забезпечувати стабільну роботу в умовах тривалого навантаження та впливу зовнішніх факторів. Необхідно передбачити механізми обробки помилок, автоматичного відновлення після збоїв і захисту від втрати даних. Програмне забезпечення має бути стійким до некоректних вхідних даних і збоїв зв'язку.

Показники надійності системи моніторингу та реєстрації динамічних параметрів руху літального апарату повинні відповідати вимогам ДСТУ 50136-1. Ймовірність безвідмовної роботи системи повинна складати не менше 99,7 %.

### 3.1.6 Вимоги до функцій та задач, які виконує система

До основних функцій системи належать:

- вимірювання динамічних параметрів руху літального апарату;
- первинна обробка та фільтрація даних;
- формування та передача телеметричних повідомлень;
- збереження даних на IoT-платформі;
- візуалізація параметрів у зручному для користувача вигляді;
- забезпечення доступу до архівних даних.

### 3.1.7 Вимоги до апаратного забезпечення

Апаратна частина системи повинна базуватися на сучасному мікроконтролері з достатнім обсягом обчислювальних ресурсів. Сенсорні модулі мають забезпечувати необхідну точність і швидкодію вимірювань. Усі компоненти повинні відповідати вимогам щодо енергоспоживання, масогабаритних характеристик та сумісності між собою.

Вимоги до елементної бази розробки:

- режими роботи і умови експлуатації вибраних елементів повинні відповідати вказаним в ТЗ;
- вибрана елементна база має забезпечувати необхідні режими роботи системи;
- елементна база по можливості має бути широкоживаною, доступною і дешевою. Необхідно також враховувати можливість заміни вибраних елементів на аналогічні (вітчизняні чи імпортного виробництва).

Вимоги до мікроконтролера:

- мікроконтролер має підтримувати RISC архітектуру команд;
- мікроконтролер повинен містити необхідний набір вбудованих периферійних пристроїв (таймери, АЦП і т.п.) та потрібну кількість керованих портів введення /виведення.

## 4 Вимоги до документації

Документація повинна відповідати вимогам ЄСКД та ДСТУ.

Комплект конструкторської документації повинен складатись з:

- пояснювальної записки;
- графічного матеріалу:
  1. структурна схема системи;
  2. схема електрична принципова;
  3. блок-схема алгоритму роботи;
  4. результати моделювання системи.

\*Примітка: В комплект конструкторської документації можуть вноситися зміни та доповнення в процесі розробки.

## 5 Стадії та етапи проектування

Таблиця 1 – Стадії та етапи виконання КРБ

№ етапу	Назва етапу виконання КРБ	Термін виконання
1.	Розробка технічного завдання	26.01 – 02.02
2.	Аналіз технічного завдання, вимог до комп'ютерної системи, та можливих рішень поставленого завдання	03.02 – 15.02
3.	Розроблення структури, вибір апаратного забезпечення, проектування комп'ютерної системи	20.04 – 25.04
4.	Реалізація алгоритму, написання програмного забезпечення, моделювання комп'ютерної системи	26.04 – 05.05
5.	Робота над четвертим розділом «Безпека життєдіяльності, основи охорони праці»	07.05 – 25.05
6.	Оформлення пояснювальної записки та графічного матеріалу	26.05 – 07.06
7.	Перевірка на академічний плагіат, перевірка керівником та консультантами	08.06 – 14.06
8.	Попередній захист кваліфікаційної роботи бакалавра	15.06 – 21.06
9.	Захист кваліфікаційної роботи бакалавра	22.06.2026

## 6 Додаткові умови виконання кваліфікаційної роботи бакалавра

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

Додаток Б  
Перелік елементів



## Додаток В

### Лістинг програми

Лістинг В.1 – Код програми мікроконтролера для реалізації системи IoT-моніторингу та реєстрації динамічних параметрів руху літального апарату.

```
#include <Wire.h>
#include <SPI.h>
#include <SD.h>
#include <TinyGsmClient.h>
#include <ThingertinyGSM.h>
#include <Adafruit_BMP280.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_GFX.h>
#include <BMI160Gen.h>
#include <TinyGPSPlus.h>
#include <esp_task_wdt.h>

#define WDT_TIMEOUT 30
#define SDA_PIN 21
#define SCL_PIN 22
#define OLED_ADDR 0x3C
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define BUTTON_PIN 0
#define SD_CS 10
#define MODEM_RX 18
#define MODEM_TX 17
#define MODEM_PWRKEY 4

HardwareSerial SerialAT(1);
TinyGsm modem(SerialAT);
TinyGsmClient client(modem);
ThingertinyGSM thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL,
client);
TinyGPSPlus gps;
Adafruit_BMP280 bmp;
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
File logFile;

struct Telemetry {
  float ax;
  float ay;
  float az;
  float gx;
  float gy;
  float gz;
  float temperature;
  float pressure;
```

```

float altitude;
double latitude;
double longitude;
double gps_altitude;
unsigned long timestamp;
};
Telemetry data;

class KalmanFilter {
public:
float Q = 0.001;
float R = 0.1;
float X = 0;
float P = 1;
float K;
float update(float measurement) {
    P = P + Q;
    K = P / (P + R);
    X = X + K * (measurement - X);
    P = (1 - K) * P;
    return X;
}
};
KalmanFilter kalmanX;
KalmanFilter kalmanY;
KalmanFilter kalmanZ;

unsigned long lastSend = 0;
unsigned long lastLog = 0;
const int sendInterval = 5000;
const int logInterval = 1000;
bool systemEnabled = true;

void setup() {
    Serial.begin(115200);
    esp_task_wdt_init(WDT_TIMEOUT, true);
    esp_task_wdt_add(NULL);
    pinMode(BUTTON_PIN, INPUT_PULLUP);
    Wire.begin(SDA_PIN, SCL_PIN);
    display.begin(SSD1306_SWITCHCAPVCC, OLED_ADDR);
    display.setTextSize(1);
    display.setTextColor(WHITE);
    display.clearDisplay();
    display.println("System init");
    display.display();
    initSensors();
    initSD();
    initModem();
    thing["telemetry"] >> [](pson &out) {
        out["ax"] = data.ax;
        out["ay"] = data.ay;
        out["az"] = data.az;
    };
};

```

```

    out["gx"] = data.gx;
    out["gy"] = data.gy;
    out["gz"] = data.gz;

    out["temperature"] = data.temperature;
    out["pressure"] = data.pressure;
    out["altitude"] = data.altitude;

    out["lat"] = data.latitude;
    out["lon"] = data.longitude;
};
display.println("System ready");
display.display();
}

void loop() {
    esp_task_wdt_reset();
    reconnectLTE();
    checkButton();
    if(systemEnabled) {
        readIMU();
        readBMP();
        readGPS();
        data.timestamp = millis();
        updateDisplay();
        logData();
        sendTelemetry();
    }
    delay(200);
}

void initModem() {
    pinMode(MODEM_PWRKEY, OUTPUT);
    digitalWrite(MODEM_PWRKEY, HIGH);
    delay(2000);
    digitalWrite(MODEM_PWRKEY, LOW);
    SerialAT.begin(115200, SERIAL_8N1, MODEM_RX, MODEM_TX);
    delay(3000);
    modem.restart();
    modem.gprsConnect(APN, "", "");
}

void reconnectLTE() {
    if (!modem.isNetworkConnected()) {
        modem.restart();
        modem.gprsConnect(APN, "", "");
    }
}

void initSensors() {
    BMI160.begin(BMI160GenClass::I2C_MODE);
    bmp.begin(0x76);
}

```

```

void readIMU() {
    int16_t axRaw, ayRaw, azRaw;
    int16_t gxRaw, gyRaw, gzRaw;
    BMI160.readAccelerometer(axRaw, ayRaw, azRaw);
    BMI160.readGyro(gxRaw, gyRaw, gzRaw);
    float ax = axRaw * 0.000061;
    float ay = ayRaw * 0.000061;
    float az = azRaw * 0.000061;
    data.ax = kalmanX.update(ax);
    data.ay = kalmanY.update(ay);
    data.az = kalmanZ.update(az);
    data.gx = gxRaw * 0.00763;
    data.gy = gyRaw * 0.00763;
    data.gz = gzRaw * 0.00763;
}

void readBMP() {
    data.temperature = bmp.readTemperature();
    data.pressure = bmp.readPressure() / 100.0;
    data.altitude =
    44330 * (1 - pow(data.pressure / 1013.25, 0.1903));
}

void readGPS() {
    while (SerialAT.available()) {
        gps.encode(SerialAT.read());
    }
    if (gps.location.isValid()) {
        data.latitude = gps.location.lat();
        data.longitude = gps.location.lng();
    }
    if (gps.altitude.isValid()) {
        data.gps_altitude = gps.altitude.meters();
    }
}

void updateDisplay() {
    display.clearDisplay();
    display.setCursor(0,0);
    display.print("AX:");
    display.println(data.ax);
    display.print("AY:");
    display.println(data.ay);
    display.print("AZ:");
    display.println(data.az);
    display.print("ALT:");
    display.println(data.altitude);
    display.print("LAT:");
    display.println(data.latitude,4);
    display.print("LON:");
    display.println(data.longitude,4);
    display.display();
}

```

```

}

void initSD() {
  if(!SD.begin(SD_CS)) {
    Serial.println("SD error");
    return;
  }
  logFile = SD.open("/flight_log.csv", FILE_APPEND);
  if(logFile) {

logFile.println("time,ax,ay,az,gx,gy,gz,temp,press,alt,lat,lon");
    logFile.close();
  }
}

void logData() {
  if(millis() - lastLog < logInterval) return;
  lastLog = millis();
  logFile = SD.open("/flight_log.csv", FILE_APPEND);
  if(logFile) {
    logFile.print(data.timestamp);
    logFile.print(",");
    logFile.print(data.ax); logFile.print(",");
    logFile.print(data.ay); logFile.print(",");
    logFile.print(data.az); logFile.print(",");
    logFile.print(data.gx); logFile.print(",");
    logFile.print(data.gy); logFile.print(",");
    logFile.print(data.gz); logFile.print(",");
    logFile.print(data.temperature); logFile.print(",");
    logFile.print(data.pressure); logFile.print(",");
    logFile.print(data.altitude); logFile.print(",");
    logFile.print(data.latitude); logFile.print(",");
    logFile.println(data.longitude);
    logFile.close();
  }
}

void sendTelemetry() {
  if(millis() - lastSend < sendInterval) return;

  lastSend = millis();

  thing.handle();
}

void checkButton() {
  if(digitalRead(BUTTON_PIN) == LOW) {
    systemEnabled = !systemEnabled;
    delay(500);
  }
}

```