

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: *Комп'ютерна система інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей*

Виконав: студент 4 курсу, групи СІ-41
спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

(підпис)

Ватиляк В.А.

(прізвище та ініціали)

Керівник

(підпис)

Стадник Н.Б.

(прізвище та ініціали)

Нормоконтроль

(підпис)

Луцик Н.С.

(прізвище та ініціали)

Завідувач кафедри

(підпис)

Осухівська Г.М.

(прізвище та ініціали)

Рецензент

(підпис)

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 Ватиляку Владиславу Андрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютерна система інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей

Керівник роботи Стадник Наталія Богданівна, к.т.н.

(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

Ватиляк В.А. Комп'ютерна система інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей : робота на здобуття освітнього ступеня бакалавра: спец. 123 — комп'ютерна інженерія. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2026.

Ключові слова: Інтернет речей, супутникова навігація, GPS, транспортні засоби, система інформування, виявлення зіткнень, вбудовані системи, мікроконтролер.

Кваліфікаційна робота присвячена розробці комп'ютерної системи інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей. Метою роботи є розроблення апаратно-програмного комплексу, здатного забезпечити оперативне інформування про аварійні події та передавання відповідних даних у віддалені інформаційні сервіси.

У першому розділі виконано аналіз технічного завдання, сформульовано основні функціональні та експлуатаційні вимоги до системи, а також проведено огляд існуючих аналогів і сучасних технічних рішень у сфері моніторингу та підвищення безпеки дорожнього руху.

У другому розділі здійснено розроблення апаратного забезпечення системи. Побудовано структурну схему та електричну принципову схему пристрою, а також обґрунтовано вибір елементної бази з урахуванням вимог до енергоефективності, надійності та вартості.

Третій розділ присвячено розробленню алгоритму роботи системи та програмного забезпечення мікроконтролера. Виконано інтеграцію з IoT-платформою для передавання й візуалізації даних, а також проведено тестування працездатності та коректності функціонування розробленої системи.

ANNOTATION

Vatyliak V.A. Computer System for Vehicle Collision Notification Using Satellite Navigation and Internet of Things Technologies. Bachelor's Graduation Thesis: speciality 123 — Computer engineering. Ternopil: Ternopil Ivan Puluj National Technical University, 2026.

Keywords: Internet of Things, IoT, satellite navigation, GPS, vehicles, collision notification system, embedded systems, microcontroller.

The qualification work is devoted to the design of a computer system for vehicle collision notification using satellite navigation and Internet of Things technologies. The aim of the work is to develop a hardware and software solution capable of timely detecting collision events and transmitting relevant information to remote services.

In the first section, the technical assignment is analyzed, system requirements are defined, and a review of existing analogues and modern solutions in the field of vehicle safety and monitoring is conducted.

The second section focuses on the development of the system hardware. A structural diagram and an electrical schematic diagram are designed, and the selection of the electronic components is justified with regard to reliability, energy efficiency, and cost.

The third section describes the development of the system algorithm and microcontroller software. Integration with an IoT platform for data transmission and visualization is implemented, and system testing is performed to verify correct operation and functionality.

ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ.....	10
1.1 Огляд та аналіз сфер застосування системи інформування про зіткнення транспортних засобів	10
1.2 Аналіз вимог до комп'ютерної системи інформування про зіткнення транспортних засобів	11
1.3 Огляд існуючих засобів для інформування про зіткнення транспортних засобів.....	13
1.4 Аналіз можливих рішень поставленого завдання.....	16
РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА	18
2.1 Структура системи інформування про зіткнення транспортних засобів	18
2.2 Апаратне забезпечення системи інформування про зіткнення транспортних засобів	20
2.2.1 Модуль NodeMCU.....	20
2.2.2 Модуль акселерометра і гіроскопа MPU-6050.....	23
2.2.3 Модуль давача вібрації SW-420.....	25
2.2.4 GPS-модуль GY-NEO6MV2	28
2.2.5 GSM-модуль SIM800L.....	30
2.3 Електрична принципова схема пристрою для інформування про зіткнення транспортних засобів	33
РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА	36
3.1 Алгоритм роботи системи інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей	36
3.2 Розробка програмного забезпечення.....	39

					<i>КС КРБ 123.150.00.00 ПЗ</i>		
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>			
<i>Розробив</i>		<i>Ватиляк В.А.</i>			<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірів</i>		<i>Стадник Н.Б.</i>				5	80
<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>Осухівська Г.М.</i>					
					<i>Комп'ютерна система інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей</i>		

3.2.1 Ініціалізація апаратних модулів та налаштування системи.....	39
3.2.2 Реалізація основного циклу функціонування системи.....	40
3.2.3 Зчитування та обробка даних акселерометра.....	42
3.2.4 Зчитування та обробка координат GPS.....	43
3.2.5 Реалізація алгоритму виявлення аварійної ситуації.....	45
3.2.6 Передавання даних на IoT-платформу через GSM-мережу.....	46
3.2.7 Реалізація функції надсилання SMS-повідомлення.....	48
3.3 Налаштування хмарної IoT платформи Ubidots для реалізації системи	50
3.4 Моделювання системи.....	52
РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	55
4.1 Долікарська допомога при кровотечах	55
4.2 Вплив вібрації на організм людини і розроблення заходів щодо зниження його рівня	58
ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	62
Додаток А Технічне завдання	
Додаток Б Перелік елементів	
Додаток В Лістинг програми	

					<i>КС КРБ 123.150.00.00 ПЗ</i>	Арк.
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		6

СПИСОК СКОРОЧЕНЬ

GPRS – General Packet Radio Service

GPS – Global Positioning System

GSM – Global System for Mobile Communications

IoT – Internet of Things

JSON – JavaScript Object Notation

MPU – Motion Processing Unit

SMS – Short Message Service

UART – Universal Asynchronous Receiver Transmitter

БД – база даних

ГІС – геоінформаційна система

ДТП – дорожньо-транспортна пригода

ПП – програмна підсистема

ТЗ – транспортний засіб

ШІМ – широтно-імпульсна модуляція

					КС КРБ 123.150.00.00 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Сучасний рівень розвитку транспортної інфраструктури супроводжується постійним зростанням інтенсивності дорожнього руху, що, у свою чергу, призводить до підвищення кількості ДТП. Зіткнення транспортних засобів становлять серйозну загрозу для життя і здоров'я учасників дорожнього руху, а також спричиняють значні матеріальні збитки. Однією з ключових проблем у разі виникнення аварійних ситуацій є несвоєчасне виявлення факту зіткнення та затримка в передаванні інформації екстреним службам або відповідальним особам.

Існуючі системи безпеки транспортних засобів не завжди забезпечують автоматизоване сповіщення про аварійні події, особливо у випадках, коли водій або пасажир не мають можливості самостійно викликати допомогу. У цьому контексті актуальною є задача створення комп'ютерної системи інформування, здатної автоматично фіксувати факт зіткнення, визначати географічне місцезнаходження транспортного засобу та передавати відповідну інформацію за допомогою сучасних телекомунікаційних технологій.

Використання технологій супутникової навігації та Інтернету речей відкриває широкі можливості для побудови ефективних, масштабованих і відносно недорогих систем моніторингу та інформування. Поєднання вбудованих мікроконтролерних платформ, навігаційних модулів і хмарних IoT-сервісів дозволяє реалізувати автоматизований обмін даними в реальному часі, що зумовлює актуальність задачі розробки системи інформування про зіткнення транспортних засобів.

Метою кваліфікаційної роботи є розроблення комп'ютерної системи інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей, яка забезпечує автоматизоване виявлення аварійних подій, визначення координат транспортного засобу та передавання інформації на віддалену IoT-платформу.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

Для досягнення поставленої мети у кваліфікаційній роботі необхідно вирішити такі основні завдання:

- проаналізувати технічне завдання та сформулювати функціональні й експлуатаційні вимоги до комп'ютерної системи інформування;
- виконати огляд існуючих аналогів і сучасних технічних рішень у сфері автоматизованих систем безпеки транспортних засобів;
- розробити структурну схему комп'ютерної системи;
- спроєктувати електричну принципову схему пристрою та обґрунтувати вибір елементної бази;
- розробити алгоритм функціонування системи;
- реалізувати програмне забезпечення для мікроконтролерної платформи;
- виконати інтеграцію розробленої системи з IoT-платформою для передавання та візуалізації даних;
- провести моделювання системи та оцінити коректність її функціонування.

					<i>КС КРБ 123.150.00.00 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 інформування про зіткнення транспортних засобів

Комп'ютерні системи інформування про зіткнення транспортних засобів набувають дедалі більшого поширення у зв'язку зі зростанням інтенсивності дорожнього руху, ускладненням транспортної інфраструктури та підвищеними вимогами до безпеки перевезень. Основною сферою застосування таких систем є автомобільний транспорт, де вони використовуються для автоматизованого виявлення аварійних ситуацій і оперативного інформування відповідних служб або зацікавлених осіб про факт ДТП. В умовах сучасних міст своєчасне отримання інформації про зіткнення транспортних засобів дозволяє скоротити час реагування екстрених служб і мінімізувати наслідки аварій.

Важливою сферою застосування комп'ютерної системи інформування є комерційні та корпоративні автопарки. Для підприємств, що здійснюють вантажні або пасажирські перевезення, актуальним є постійний контроль стану транспортних засобів і безпеки водіїв. У таких умовах система інформування про зіткнення може використовуватися як складова комплексів моніторингу транспорту, забезпечуючи автоматичну реєстрацію аварійних подій, фіксацію координат місця події та передавання інформації диспетчерським службам або відповідальним особам. Це сприяє підвищенню ефективності управління автопарком і зменшенню фінансових втрат.

Ще однією сферою застосування такої системи є страхова галузь, де зростає потреба в об'єктивних даних про обставини ДТП. Використання комп'ютерних систем інформування дозволяє автоматично фіксувати факт зіткнення та його часово-просторові характеристики, що може бути

					<i>КС КРБ 123.150.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ватиліяк В.А.</i>			<i>Аналіз технічного завдання</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Стадник Н.Б.</i>					<i>10</i>	<i>8</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>						

використано для аналізу страхових випадків. У поєднанні з технологіями Інтернету речей такі системи створюють передумови для впровадження концепцій страхування, орієнтованого на реальні умови експлуатації транспортного засобу.

Значний потенціал застосування комп'ютерної системи інформування про зіткнення мають також служби екстреного реагування. Автоматизоване передавання координат місця аварії з використанням супутникової навігації дозволяє зменшити затримки, пов'язані з людським фактором, і підвищити точність визначення місця події. Це особливо актуально для аварій, що сталися в малозаселених районах або за умов обмеженої видимості, де оперативне інформування має вирішальне значення.

Окремою сферою застосування є інтеграція подібних систем у концепцію «розумного транспорту» та інтелектуальних транспортних систем. У таких рішеннях комп'ютерна система інформування може виступати джерелом даних для аналітичних платформ, що здійснюють аналіз дорожньої обстановки, виявлення небезпечних ділянок і оптимізацію організації руху. Завдяки використанню IoT-платформ з'являється можливість централізованого збору та оброблення інформації з великої кількості транспортних засобів у реальному часі.

Крім практичного використання, комп'ютерна система інформування про зіткнення транспортних засобів має значну навчальну та дослідницьку цінність. Вона може застосовуватися у закладах вищої освіти для вивчення принципів побудови вбудованих систем, використання супутникової навігації, бездротових мереж та IoT-технологій. Аналіз сфер застосування показує, що розробка такої системи є актуальною та доцільною, а її функціональні можливості можуть бути адаптовані до різних умов експлуатації та вимог користувачів.

1.2 Аналіз вимог до комп'ютерної системи інформування про зіткнення транспортних засобів

Вимоги до комп'ютерної системи інформування про зіткнення транспортних засобів базуються на положеннях технічного завдання, особливостях об'єкта

					КС КРБ 123.150.00.00 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

автоматизації та умовах експлуатації системи в реальному середовищі транспортного засобу.

Передусім система повинна забезпечувати автоматизоване та своєчасне виявлення факту зіткнення без участі користувача. Це зумовлює необхідність використання відповідних сенсорних елементів і алгоритмів оброблення даних, здатних відрізнати аварійні ситуації від звичайних режимів руху. Вимоги до точності та швидкодії системи є критичними, оскільки затримка у фіксації події або помилкові спрацювання можуть знизити ефективність інформування та призвести до недовіри користувачів.

Важливою вимогою є забезпечення коректного визначення географічних координат місця події. Застосування технологій супутникової навігації накладає вимоги до стабільності приймання сигналу, точності позиціонування та часу оновлення навігаційних даних. Отримана інформація повинна бути інтегрована з даними про аварійну подію та передана у зручному для подальшої обробки форматі.

Система також повинна відповідати вимогам до надійності та безперервності функціонування. Оскільки експлуатація відбувається в умовах вібрацій, перепадів температури та можливих перешкод зв'язку, апаратні та програмні компоненти мають бути стійкими до зовнішніх впливів. У разі тимчасової втрати зв'язку з мережею Інтернет система повинна забезпечувати збереження даних і їх подальше передавання після відновлення з'єднання.

Окрему групу вимог становлять вимоги до засобів зв'язку та інтеграції з IoT-платформою. Система повинна підтримувати сучасні протоколи передавання даних, забезпечувати сумісність з хмарними сервісами та можливість віддаленого доступу до інформації. Це дозволяє реалізувати централізований моніторинг, аналіз і візуалізацію даних, що є важливим для подальшого розвитку системи.

З точки зору апаратного забезпечення, висуваються вимоги до компактності, енергоефективності та можливості живлення від бортової мережі транспортного засобу. Обчислювальні ресурси мікроконтролерної платформи мають бути

					КС КРБ 123.150.00.00 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

достатніми для реалізації алгоритмів оброблення даних і взаємодії з периферійними модулями.

Аналіз вимог до системи інформування про зіткнення транспортних засобів дозволяє сформувати комплексне уявлення про функціональні та технічні обмеження, які визначають архітектуру системи та напрямки її подальшої реалізації.

1.3 Огляд існуючих засобів для інформування про зіткнення транспортних засобів

Сучасний ринок технологій безпеки дорожнього руху пропонує низку рішень, спрямованих на автоматичне інформування про ДТП та прискорення виклику служб порятунку. Однією з найвідоміших і найбільш стандартизованих систем є європейська система автоматичного виклику допомоги eCall, яка стала обов'язковою для нових автомобілів у країнах Європейського Союзу з 2018 року [1]. eCall активується автоматично при серйозному зіткненні або вручну через кнопку SOS, після чого передає службам номеру 112 пакет даних про місце події, час, курс руху та інші мінімально необхідні параметри, а також ініціює голосовий зв'язок між водієм і диспетчером служби порятунку (рис. 1.1).



Рисунок 1.1 – Європейська система автоматичного виклику допомоги eCall

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

Схожі рішення пропонують і виробники автотранспорту як частину власних телематичних систем. Наприклад, бренди Volvo та інші великі автовиробники інтегрують автоматичне інформування про зіткнення у свої автомобілі, що реалізує передачу сигналу у контакт-центр компанії або відповідну службу допомоги з визначенням геолокації автомобіля [2]. Аналогічно, системи екстреного виклику Intelligent Emergency Call компанії BMW можуть працювати автоматично при ДТП або бути активовані вручну, передаючи географічні дані та встановлюючи зв'язок із кол-центром підтримки (рис. 1.2).

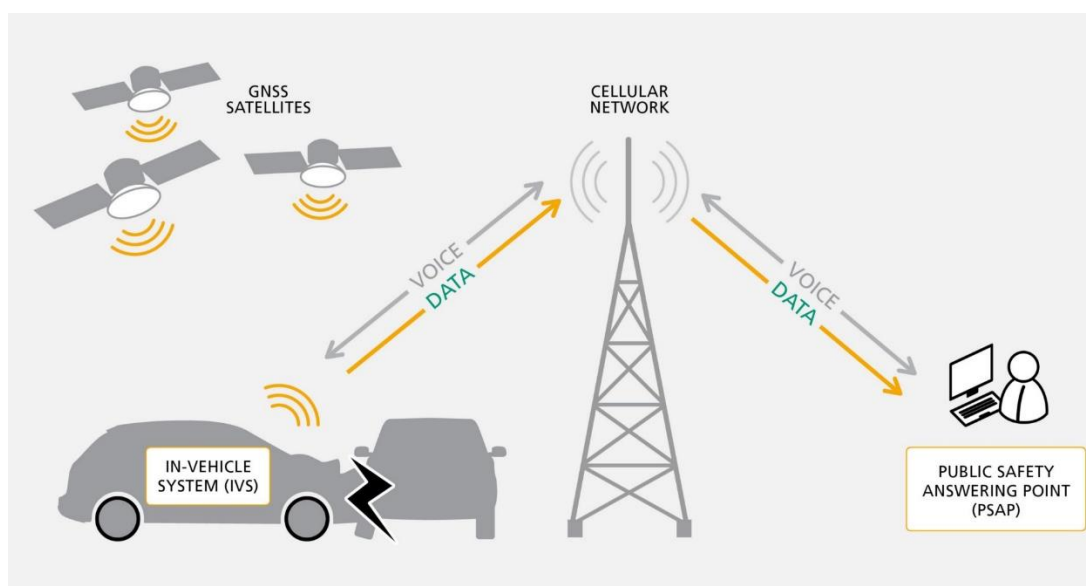


Рисунок 1.2 – Система екстреного виклику Intelligent Emergency Call

Ці рішення мають низку суттєвих переваг: автоматичне визначення аварії без участі водія, інтеграція з навігаційними сервісами для точної локалізації транспортного засобу та можливість встановлення голосового зв'язку. Такі системи здатні значно скоротити час реагування екстрених служб і потенційно зберегти життя постраждалим у ДТП.

Проте на практиці існуючі засоби мають кілька технічних, експлуатаційних і ринкових обмежень, які знижують їхню ефективність та доступність. Значна частина рішень є заводською інсталяцією, що робить їх недоступними для встановлення на старі транспортні засоби без значних витрат на модернізацію. Для eCall, наприклад, стандартно обладнуються лише нові автомобілі, що обмежує

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		14

охоплення парку транспортних засобів, особливо в країнах із високою часткою старих авто.

Багато рішень сильно залежать від телекомунікаційної інфраструктури та мобільного зв'язку, що створює ризики для їхньої роботи у регіонах зі слабким покриттям або при втраті з'єднання під час аварії. Такі обмеження можуть призводити до затримок у передаванні даних про зіткнення або до їхньої втрати, що особливо критично в умовах екстрених ситуацій, де кожна хвилина важлива.

Іншим недоліком є обмежена функціональність та інтеграція з сучасними IoT-системами у багатьох існуючих рішеннях. Стандартні реалізації eCall, як правило, передають лише мінімальний набір даних для виклику служби порятунку, включно з географічними координатами та часом події, але не забезпечують розширену телеметрію, стан транспортного засобу або біометричні дані постраждалих, які можуть бути корисними для служб реагування.

Крім того, ринкові пропозиції щодо автономних GPS-трекерів або універсальних аварійних сповіщувачів після ДТП також мають суттєві обмеження. Хоча деякі пристрої пропонують передачу сповіщень через SMS, мобільні додатки чи інші канали [3], їхні можливості щодо точного визначення факту зіткнення без помилкових спрацювань часто залишають бажати кращого. Такий функціонал зазвичай залежить від простих порогових давачів прискорення, які можуть неправильно інтерпретувати різкі маневри чи дорожні нерівності як зіткнення, що призводить до хибних тривог або, навпаки, пропуску реальних аварій.

Ще одним важливим аспектом є приватність та безпека даних, адже системи, що передають інформацію про положення транспортного засобу, постійно моніторять його координати. Це викликає занепокоєння у частини користувачів щодо збору персональних даних, що стає бар'єром для широкого впровадження таких рішень.

Також слід звернути увагу, що існуючі рішення не завжди інтегруються в єдине середовище обробки даних або IoT-екосистему, що обмежує їхню адаптивність для розширених сценаріїв використання, таких як централізований моніторинг автопарків або аналітика даних про аварії. Часто вони зосереджуються

					КС КРБ 123.150.00.00 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

лише на функції виклику допомоги, не враховуючи потреби в подальшій обробці, зберіганні та аналітиці телеметрії.

Підсумовуючи огляд існуючих засобів інформування про зіткнення транспортних засобів, можна констатувати, що сучасні рішення виконують базову функцію автоматичного виклику допомоги та мають потенціал щодо підвищення безпеки дорожнього руху. Однак вони характеризуються низькою гнучкістю щодо встановлення на різні моделі транспортних засобів, обмеженою функціональністю поза стандартним набором даних, залежністю від якості зв'язку, дорогими компонентами та потенційними ризиками приватності. Саме ці недоліки створюють передумови для розробки нової комп'ютерної системи, яка б поєднувала розширені можливості телеметрії, інтеграцію з IoT-платформою та адаптивність до різних умов експлуатації.

1.4 Аналіз можливих рішень поставленого завдання

Вирішення задачі розробки комп'ютерної системи інформування про зіткнення транспортних засобів передбачає розгляд існуючих технічних підходів до виявлення аварійних ситуацій, визначення місцезнаходження транспортного засобу та організації передавання інформації до віддалених користувачів або сервісів. На основі вимог технічного завдання та результатів огляду аналогів можна виділити кілька концептуальних напрямів реалізації такої системи, кожен з яких має свої переваги та обмеження.

Перший можливий підхід полягає у використанні штатних автомобільних систем безпеки та телематики, інтегрованих безпосередньо у конструкцію транспортного засобу. Такі рішення зазвичай характеризуються високою надійністю та оптимальною взаємодією з бортовими системами автомобіля. Проте їх застосування обмежене заводською комплектацією та потребує значних фінансових витрат, що робить даний підхід малопридатним для реалізації в межах кваліфікаційної роботи та для модернізації вже експлуатованих транспортних засобів.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

Другий підхід передбачає використання автономних комерційних GPS-трекерів або аварійних маяків, які здатні передавати повідомлення про зіткнення через мобільний зв'язок або SMS-повідомлення. Такі пристрої відзначаються відносною простотою встановлення та незалежністю від конкретної моделі транспортного засобу. Водночас їх функціональні можливості зазвичай обмежені, алгоритми виявлення зіткнення мають низьку адаптивність, а можливості інтеграції з сучасними IoT-платформами часто є мінімальними або відсутніми. Це знижує гнучкість використання таких рішень у складніших сценаріях моніторингу та аналізу даних.

Третій підхід полягає у побудові комп'ютерної системи на основі мікроконтролерної платформи з використанням окремих сенсорних, навігаційних і комунікаційних модулів. Такий варіант дозволяє гнучко формувати архітектуру системи відповідно до вимог технічного завдання, оптимізувати вибір елементної бази та реалізувати власні алгоритми оброблення даних. Застосування технологій Інтернету речей у цьому випадку забезпечує можливість передавання інформації в реальному часі, збереження даних у хмарних сервісах і подальшу їх візуалізацію.

Окремо слід розглянути програмно-орієнтовані рішення, які використовують мобільні пристрої як основний елемент системи. Смартфони оснащені давачами прискорення, GPS-модулями та засобами зв'язку, що дозволяє реалізувати функцію інформування про аварію на програмному рівні. Проте такі рішення залежать від коректної роботи користувацького програмного забезпечення, стану акумулятора та розташування пристрою, що знижує їхню надійність у критичних ситуаціях.

Порівняльний аналіз можливих рішень показує, що найбільш доцільним для досягнення поставленої мети є розроблення спеціалізованої мікроконтролерної комп'ютерної системи з інтеграцією супутникової навігації та IoT-платформи. Такий підхід забезпечує баланс між функціональністю, вартістю, гнучкістю та можливістю подальшої модернізації. Він також дозволяє реалізувати всі етапи проектування апаратного та ПЗ, що є важливим з точки зору досягнення мети кваліфікаційної роботи.

					<i>КС КРБ 123.150.00.00 ПЗ</i>	<i>Арк.</i>
						<i>17</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Структура системи інформування про зіткнення транспортних засобів

На рисунку 2.1 наведено структурну схему комп'ютерної системи інформування про зіткнення транспортних засобів. На схемі відображено основні функціональні блоки системи та інформаційні зв'язки між ними.

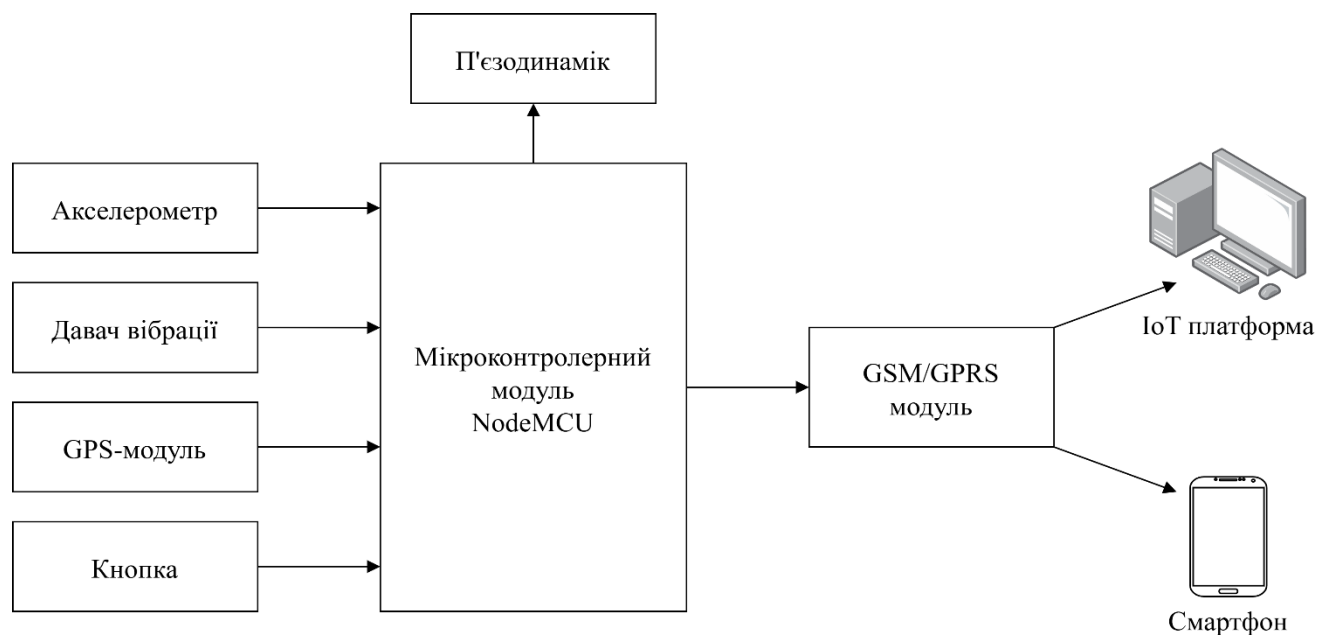


Рисунок 2.1 – Структурна схема системи інформування про зіткнення транспортних засобів

Центральним елементом системи є мікроконтролерний модуль, який виконує функції керування всіма іншими компонентами, оброблення даних сенсорів та формування повідомлень про аварійну подію. Саме цей модуль реалізує основний алгоритм роботи системи, здійснює аналіз отриманих даних і приймає рішення щодо фіксації факту зіткнення транспортного засобу. Крім того, мікроконтролер забезпечує взаємодію з навігаційним модулем, комунікаційним модулем та іншими периферійними пристроями.

					<i>КС КРБ 123.150.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ватиляк В.А.</i>			<i>Проектна частина</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
<i>Перевірив</i>		<i>Стадник Н.Б.</i>					18	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>Осужівська Г.М.</i>						

Для визначення можливого факту аварії в системі використовуються сенсорні модулі, до яких належать акселерометр з гіроскопом та сенсор вібрації. Акселерометр і гіроскоп забезпечують вимірювання прискорень і змін просторового положення транспортного засобу. На основі аналізу цих параметрів можна визначити різкі зміни руху, характерні для зіткнення або сильного удару. Сенсор вібрації виконує допоміжну функцію та дозволяє зафіксувати механічний вплив або сильну вібрацію корпусу транспортного засобу. Комбінований аналіз даних з цих двох джерел дозволяє підвищити достовірність визначення аварійної ситуації та зменшити кількість хибних спрацювань системи.

Після фіксації події зіткнення мікроконтролер ініціює отримання координат місця події за допомогою навігаційного модуля супутникового позиціонування. Даний модуль приймає сигнали супутникових навігаційних систем і визначає географічні координати транспортного засобу. Отримані координати передаються до мікроконтролера, де вони обробляються та включаються до повідомлення про аварію.

Передавання інформації до віддалених сервісів здійснюється за допомогою комунікаційного модуля стільникового зв'язку. Цей модуль забезпечує підключення системи до мобільної мережі та дозволяє передавати дані через мережу Інтернет. Сформоване повідомлення, яке містить інформацію про аварійну подію та координати місця її виникнення, передається на хмарну IoT-платформу, де здійснюється збереження, оброблення та візуалізація отриманих даних.

У структурі системи також передбачено кнопку ручного керування. У разі натискання кнопки система виконує ручне скидання аварійного стану системи. Це дозволяє запобігти повторному надсиланню повідомлень у випадку хибного спрацювання або після усунення наслідків події.

Для локальної індикації стану системи використовується активний п'єзодинамік, який може генерувати звукові сигнали. Звукове сповіщення може застосовуватися для інформування користувача про спрацювання системи.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2 Апаратне забезпечення системи інформування про зіткнення транспортних засобів

2.2.1 Модуль NodeMCU

Мікроконтролерний модуль NodeMCU є однією з найбільш поширених апаратних платформ для створення пристроїв Інтернету речей (IoT). Він поєднує у собі мікроконтролер, бездротовий Wi-Fi інтерфейс, пам'ять, схему живлення та інтерфейс програмування, що дозволяє використовувати його як центральний керуючий елемент комп'ютерних систем. Плата NodeMCU розроблена на базі мікросхеми ESP8266, яка спочатку була створена як універсальний Wi-Fi контролер для вбудованих пристроїв, а згодом отримала широке застосування у системах автоматизації, дистанційного моніторингу та IoT (рис. 2.2).



Рисунок 2.2 – Модуль NodeMCU

Конструктивно модуль NodeMCU являє собою плату розробника, на якій розміщено основний чіп ESP8266, USB-UART перетворювач CH340, стабілізатор напруги, кнопки керування та виводи для підключення периферійних пристроїв. Наявність USB-інтерфейсу значно спрощує програмування модуля, оскільки він може підключатися безпосередньо до комп'ютера через стандартний кабель micro-USB. USB-UART перетворювач виконує роль інтерфейсу між комп'ютером і

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		20

мікроконтролером, забезпечуючи передачу програмного коду та налагодження через послідовний порт.

Основним обчислювальним елементом модуля є мікроконтролер ESP8266, який містить 32-бітний процесор архітектури Tensilica L106. Процесор працює з тактовою частотою 80 МГц. Завдяки цьому модуль має достатню продуктивність для виконання алгоритмів оброблення даних сенсорів, керування периферійними пристроями та реалізації мережевих протоколів. Продуктивність мікроконтролера досягає приблизно 600 DMIPS.

Важливою особливістю NodeMCU є наявність вбудованого бездротового Wi-Fi інтерфейсу, який працює в діапазоні 2,4 ГГц. Мікроконтролер містить повний стек протоколів TCP/IP, що дозволяє реалізовувати мережеву взаємодію без використання додаткових комунікаційних мікросхем. Модуль підтримує кілька режимів роботи бездротової мережі: режим клієнта (STA), режим точки доступу (AP) або їх одночасне використання.

Плата NodeMCU також має набір універсальних цифрових входів та виходів, які використовуються для підключення сенсорів і виконавчих пристроїв. Більшість цифрових виводів можуть працювати у режимах GPIO, PWM, I2C або SPI, що забезпечує високу гнучкість при побудові електронних систем. Крім того, модуль містить один аналоговий вхід, який дозволяє вимірювати аналогові сигнали.

Принцип роботи мікроконтролерного модуля полягає у виконанні програмного коду, записаного у флеш-пам'ять мікроконтролера. Після подачі живлення або перезавантаження пристрою запускається прошивка, яка реалізує алгоритм функціонування системи. У процесі роботи мікроконтролер взаємодіє з підключеними сенсорами через цифрові або аналогові інтерфейси, виконує оброблення отриманих даних і керує периферійними модулями. У разі використання мережевих функцій модуль встановлює бездротове з'єднання та здійснює передачу даних до віддалених серверів або IoT-платформ.

NodeMCU має достатній обсяг пам'яті для виконання складніших програм. Зокрема, модуль містить приблизно 4 МБ флеш-пам'яті, яка використовується для зберігання програмного коду та файлової системи, а також оперативну пам'ять,

					КС КРБ 123.150.00.00 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

необхідну для виконання програм. Наявність апаратної підтримки мережевих протоколів дозволяє реалізувати взаємодію з веб-сервісами, MQTT-брокерами та іншими IoT-системами. Основні технічні характеристики мікроконтролерного модуля NodeMCU наведені у таблиці 2.1.

Таблиця 2.1 – Характеристики модуля NodeMCU

Характеристика	Значення
Мікроконтролер	ESP8266 (модуль ESP-12E / ESP-12F)
Архітектура процесора	Tensilica L106, 32-біт
Тактова частота	80 МГц (до 160 МГц)
Флеш-пам'ять	4 МБ
Оперативна пам'ять	~50 КБ доступно користувачу
Бездротовий інтерфейс	Wi-Fi IEEE 802.11 b/g/n (2.4 ГГц)
Режими Wi-Fi	STA, AP, STA+AP
Кількість GPIO	до 11
Аналоговий вхід	1 (ADC)
Інтерфейси	UART, SPI, I2C, PWM
Напруга живлення	3,3 В (5 В через USB)
Струм споживання	~70 мА під час передачі
Інтерфейс програмування	Micro-USB
Робоча температура	-40...+125 °С

Вибір мікроконтролерного модуля NodeMCU для реалізації комп'ютерної системи інформування про зіткнення транспортних засобів є технічно обґрунтованим. По-перше, модуль має достатню обчислювальну потужність для оброблення даних з акселерометра, сенсора вібрації та GPS-модуля. По-друге, він підтримує широкий набір інтерфейсів, що дозволяє легко підключати різноманітні сенсорні та комунікаційні модулі. По-третє, наявність вбудованого Wi-Fi та підтримки мережевих протоколів спрощує інтеграцію з IoT-платформами та віддаленими сервісами.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2.2 Модуль акселерометра і гіроскопа MPU-6050

Модуль MPU-6050 є сучасним інерційним сенсорним пристроєм, який широко використовується в системах вимірювання руху, стабілізації, навігації та контролю положення об'єктів. Даний модуль належить до класу MEMS-сенсорів і поєднує у собі два типи вимірювальних елементів: тривісний акселерометр та тривісний гіроскоп. Таким чином, сенсор забезпечує вимірювання шести параметрів руху, що дозволяє визначати як лінійні прискорення, так і кутову швидкість об'єкта в просторі (рис. 2.3).



Рисунок 2.3 – Модуль акселерометра і гіроскопа MPU-6050

Конструктивно модуль складається з мікросхеми MPU-6050, яка змонтована на невеликій платі розширення разом із допоміжними електронними компонентами. До складу плати входять стабілізатор напруги, підтягувальні резистори для інтерфейсу I²C, а також контакти для підключення до мікроконтролера. Сам сенсор виконаний у компактному корпусі типу QFN, що дозволяє використовувати його у малогабаритних електронних пристроях.

Основа модуля становить мікроелектромеханічний сенсорний елемент, у якому використовуються мікроскопічні механічні структури. Акселерометр визначає прискорення за рахунок зміщення мікромеханічної маси під дією інерційних сил. Це зміщення призводить до зміни електричної ємності між елементами сенсора, що фіксується електронною схемою та перетворюється у цифрове значення. Таким чином визначається прискорення вздовж трьох взаємно перпендикулярних осей X, Y та Z.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						23
Змн.	Арк.	№ докум.	Підпис	Дата		

Гіроскоп у складі MPU-6050 призначений для вимірювання кутової швидкості обертання об'єкта. Його робота базується на ефекті Кориоліса: під час обертання сенсорних елементів виникають додаткові сили, які спричиняють мікроскопічні коливання структури. Ці коливання реєструються електронною схемою і перетворюються на цифрові значення кутової швидкості. Поєднання акселерометра і гіроскопа дозволяє отримати значно точнішу інформацію про рух об'єкта, ніж при використанні лише одного типу сенсора.

Важливою особливістю MPU-6050 є наявність вбудованого цифрового процесора оброблення руху. Цей блок виконує обчислення, пов'язані з аналізом даних акселерометра та гіроскопа. Процесор може виконувати алгоритми об'єднання даних сенсорів і формувати інформацію про орієнтацію пристрою у просторі. Основні технічні характеристики модуля MPU-6050 наведено у таблиці 2.2.

Таблиця 2.2 – Характеристики модуля MPU-6050

Характеристика	Значення
Тип сенсора	MEMS акселерометр + гіроскоп
Кількість осей	3 осі акселерометра + 3 осі гіроскопа
Діапазон вимірювання прискорення	± 2 g, ± 4 g, ± 8 g, ± 16 g
Роздільна здатність АЦП	16 біт
Інтерфейс зв'язку	I ² C (до 400 кГц)
Напруга живлення мікросхеми	2,3–3,4 В
Типове споживання струму	$\approx 3,9$ мА
Робоча температура	$-40 \dots +85$ °C
Кількість виводів модуля	8

Для обміну даними з мікроконтролером використовується послідовний інтерфейс I²C, який підтримує швидкість передачі до 400 кГц. Сенсор передає результати вимірювань у вигляді 16-бітних цифрових значень для кожної осі акселерометра і гіроскопа. Наявність цифрового інтерфейсу спрощує інтеграцію

модуля з мікроконтролерними системами та дозволяє підключати його до більшості сучасних платформ.

Додатковою функціональною можливістю MPU-6050 є наявність вбудованого датчика температури, який дозволяє контролювати температуру мікросхеми та компенсувати можливі похибки вимірювань. Крім того, модуль підтримує систему переривань, яка дозволяє повідомляти мікроконтролер про появу нових даних або виникнення певних подій руху.

Використання модуля MPU-6050 у комп'ютерній системі інформування про зіткнення транспортних засобів є технічно обґрунтованим. По-перше, сенсор дозволяє вимірювати різкі прискорення та зміни руху, які характерні для аварійних ситуацій. У випадку зіткнення транспортного засобу виникають значні інерційні навантаження, що можуть бути зафіксовані акселерометром. По-друге, гіроскоп дає можливість визначати різкі зміни кутового положення транспортного засобу, які часто супроводжують аварії, наприклад перекидання або різкий поворот під час удару. Комбінований аналіз даних акселерометра і гіроскопа дозволяє точніше визначати факт аварії та зменшити ймовірність хибних спрацювань системи. По-третє, модуль має компактні розміри, низьке енергоспоживання та простий інтерфейс підключення, що робить його зручним для інтеграції у вбудовані системи на базі мікроконтролерів. Наявність численних програмних бібліотек і прикладів використання також значно спрощує процес розроблення ПЗ.

Модуль MPU-6050 є ефективним сенсорним елементом для реалізації функції визначення аварійних ситуацій у комп'ютерній системі інформування про зіткнення транспортних засобів, оскільки забезпечує точне вимірювання параметрів руху та дозволяє реалізувати алгоритми аналізу динаміки транспортного засобу.

2.2.3 Модуль датчик вібрації SW-420

Модуль SW-420 є простим і широко застосовуваним електронним сенсором, призначеним для виявлення механічних коливань, ударів або вібрацій різної інтенсивності. Такі модулі часто використовуються у системах безпеки, сигналізації, моніторингу роботи обладнання, а також у різноманітних IoT-

					КС КРБ 123.150.00.00 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

проектах. Завдяки простій конструкції, низькій вартості та легкій інтеграції з мікроконтролерами модуль SW-420 став одним із найпоширеніших давачів вібрації у навчальних та інженерних проєктах (рис. 2.4).

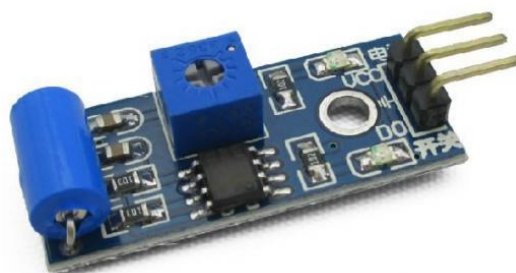


Рисунок 2.4 – Модуль давача вібрації SW-420

Конструктивно модуль складається з кількох основних елементів: сенсорного елемента SW-420, компаратора LM393, потенціометра, світлодіодного індикатора та роз'ємів для підключення до мікроконтролера. Центральним компонентом є механічний сенсор SW-420, який являє собою спеціальний вібраційний перемикач із внутрішньою пружиною. При виникненні механічних коливань або удару внутрішній контакт змінює своє положення, що призводить до короткочасної зміни електричного стану кола.

Додатково на платі розміщено компаратор LM393, який виконує функцію формування стабільного цифрового сигналу. Компаратор аналізує електричний сигнал, що надходить від сенсорного елемента, та перетворює його у чіткий логічний сигнал високого або низького рівня. Це дозволяє безпосередньо підключати модуль до цифрових входів мікроконтролерів без необхідності додаткової обробки сигналу. Крім того, на платі встановлений потенціометр, за допомогою якого можна регулювати чутливість сенсора до вібрацій, встановлюючи необхідний поріг спрацювання.

Принцип роботи давача базується на механічному перемиканні контактів під дією вібрації. У нормальному стані сенсор перебуває у замкнутому стані. Коли виникає механічна дія (струс, удар або сильна вібрація), внутрішня пружина зміщується, що призводить до короткочасного розриву електричного контакту. Ця

					КС КРБ 123.150.00.00 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

зміна фіксується електронною схемою модуля і перетворюється у цифровий сигнал, який може бути зчитаний мікроконтролером. Таким чином система отримує інформацію про факт наявності вібрації або удару.

Вихідний сигнал модуля є цифровим. У більшості реалізацій у відсутності вібрації на виході формується логічна «1», а при виникненні коливань сигнал змінюється на логічний «0». Такий принцип роботи значно спрощує використання сенсора у мікроконтролерних системах, оскільки для його роботи достатньо одного цифрового входу. Характеристики модуля SW-420 наведені у таблиці 2.3.

Таблиця 2.3 – Характеристики модуля SW-420

Характеристика	Значення
Тип вихідного сигналу	Цифровий (High/Low)
Напруга живлення	3,3–5 В
Споживаний струм	приблизно до 15 мА
Мікросхема компаратора	LM393
Регулювання чутливості	Підлаштовувальний потенціометр
Інтерфейс підключення	VCC, GND, DO
Робоча температура	–40...+85 °С

Використання модуля SW-420 у комп'ютерній системі інформування про зіткнення транспортних засобів є обґрунтованим, оскільки сенсор дозволяє оперативно зафіксувати механічний удар або сильну вібрацію, що є характерною ознакою ДТП. Крім того, він має простий цифровий інтерфейс, що значно спрощує його інтеграцію з мікроконтролерним модулем NodeMCU.

Використання цього датчика разом з акселерометром та гіроскопом дозволяє реалізувати комбінований алгоритм визначення аварійної ситуації. У такій системі акселерометр аналізує прискорення та зміну положення транспортного засобу, тоді як датчик SW-420 фіксує безпосередній факт механічного удару або різкої вібрації. Поєднання цих двох типів сенсорів підвищує надійність визначення аварійної події та зменшує ймовірність хибних спрацювань.

2.2.4 GPS-модуль GY-NEO6MV2

GPS-модуль GY-NEO6MV2 є популярним навігаційним пристроєм, який використовується для визначення географічних координат, швидкості руху та точного часу за допомогою супутникової системи глобального позиціонування. Модуль широко застосовується у системах навігації, трекінгу транспортних засобів, безпілотних апаратах, системах моніторингу та IoT-пристроях. Основою модуля є приймач u-blox NEO-6M, який забезпечує високу чутливість до супутникових сигналів і достатню точність позиціонування. (рис. 2.5).

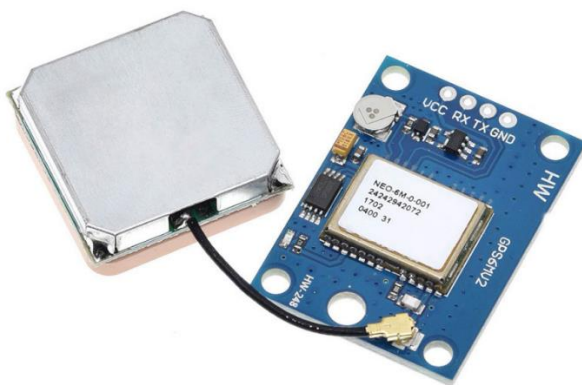


Рисунок 2.5 – GPS-модуль GY-NEO6MV2

Конструктивно модуль складається з кількох основних компонентів: навігаційної мікросхеми NEO-6M, активної керамічної GPS-антени, стабілізатора живлення, інтерфейсної електроніки та допоміжних елементів. Антена призначена для приймання сигналів супутників у діапазоні 1575,42 МГц (L1-діапазон GPS). Плата також містить світлодіодний індикатор, який сигналізує про стан прийому супутникових сигналів, а також резервну батарею, що використовується для збереження параметрів конфігурації та альманаху супутників у пам'яті модуля. Завдяки цьому час повторного запуску системи значно скорочується.

Принцип роботи GPS-модуля базується на прийманні радіосигналів від супутників навігаційної системи GPS. Кожен супутник передає інформацію про свій точний час і орбітальні параметри. Навігаційний приймач вимірює час проходження сигналу від кількох супутників і на основі цих даних визначає відстань до них. Використовуючи метод триангуляції, модуль обчислює

					КС КРБ 123.150.00.00 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

координати приймача — широту, довготу та висоту. Для точного визначення положення необхідно отримати сигнали від чотирьох супутників. Отримані дані передаються у вигляді стандартних NMEA-повідомлень через послідовний інтерфейс до мікроконтролера, який може виконувати подальшу обробку цих даних.

Модуль GY-NEO6MV2 використовує послідовний інтерфейс UART, що значно спрощує його інтеграцію з мікроконтролерними платформами. За замовчуванням обмін даними відбувається зі швидкістю 9600 бод, але цей параметр може бути змінений у налаштуваннях модуля. Технічні характеристики GPS-модуля наведено у таблиці 2.4.

Таблиця 2.4 – Технічні характеристики GPS-модуля

Характеристика	Значення
Навігаційний чіп	u-blox NEO-6M
Частота прийому	1575,42 МГц (GPS L1)
Кількість каналів прийому	до 50
Точність визначення координат	≈2,5 м
Чутливість приймача	до -161 дБм
Частота оновлення даних	1 Гц (до 5 Гц)
Інтерфейс зв'язку	UART
Типова швидкість передачі	9600 бод
Напруга живлення	3,3–5 В
Споживаний струм	≈45 мА
Час холодного старту	~27 с
Час гарячого старту	~1 с
Робоча температура	-40...+85 °С

Важливою перевагою модуля є висока чутливість приймача, яка може досягати приблизно -161 дБм, що дозволяє приймати сигнали навіть за відносно слабкого рівня сигналу. Типова точність визначення координат становить близько

2,5 м, що є достатнім для більшості прикладних систем моніторингу транспортних засобів. Частота оновлення координат зазвичай становить 1 Гц, але може бути збільшена до 5 Гц, що забезпечує швидке оновлення інформації про положення об'єкта.

Ще однією важливою особливістю модуля є наявність вбудованої пам'яті EEPROM, у якій зберігаються параметри конфігурації та навігаційні дані. Це дозволяє зменшити час ініціалізації після повторного ввімкнення. Час «холодного старту» становить приблизно 27 секунд, тоді як «гарячий старт» може виконуватися менш ніж за 1 секунду, якщо інформація про супутники вже збережена у пам'яті.

Використання модуля GY-NEO6MV2 у комп'ютерній системі інформування про зіткнення транспортних засобів є технічно обґрунтованим. Основною функцією цього модуля у системі є визначення точних координат транспортного засобу у момент аварійної події. Після виявлення зіткнення за допомогою акселерометра та сенсора вібрації мікроконтролер може отримати поточні координати від GPS-модуля та передати їх разом із повідомленням про аварію на хмарну IoT-платформу або до служби екстреної допомоги.

Крім того, модуль характеризується високою точністю позиціонування, стабільною роботою, компактними розмірами та простим інтерфейсом підключення, що робить його зручним для інтеграції з мікроконтролером NodeMCU.

2.2.5 GSM-модуль SIM800L

GSM-модуль SIM800L є компактним телекомунікаційним пристроєм, призначеним для забезпечення мобільного зв'язку у вбудованих електронних системах. Він широко використовується у проєктах Інтернету речей, системах віддаленого моніторингу, сигналізації, GPS-трекерах та інших пристроях, де необхідна передача даних через мережі стільникового зв'язку. Модуль побудований на базі мікросхеми SIM800L і підтримує роботу у чотирьох GSM-діапазонах, що дозволяє використовувати його у більшості регіонів світу (рис. 2.6).

					КС КРБ 123.150.00.00 ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

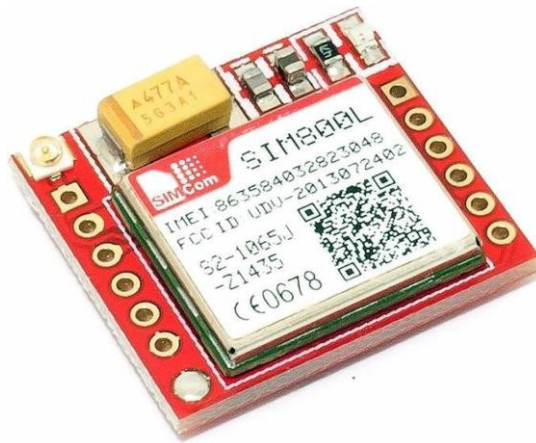


Рисунок 2.6 – GSM-модуль SIM800L

Конструктивно модуль являє собою компактну друковану плату, на якій розміщені основні елементи системи мобільного зв'язку. Центральним компонентом є мікросхема SIM800L GSM/GPRS-модему, яка реалізує всі функції оброблення сигналів мобільної мережі. На платі також розташовані роз'єм для micro-SIM карти, контакт для підключення спіральної антени, індикатор стану мережі (LED), а також контакти для підключення живлення та інтерфейсу зв'язку з мікроконтролером. Завдяки компактним розмірам плати модуль легко інтегрується у різноманітні електронні пристрої.

Принцип роботи GSM-модуля полягає у підключенні до мережі мобільного оператора за допомогою встановленої SIM-карти. Після подачі живлення модуль виконує процедуру ініціалізації та пошуку базових станцій мобільного зв'язку. У разі успішної реєстрації в мережі модуль може виконувати різні функції зв'язку: надсилання та приймання SMS-повідомлень, передачу пакетних даних через GPRS, а також здійснення голосових викликів. Управління роботою модуля здійснюється за допомогою AT-команд, які передаються через послідовний інтерфейс UART від мікроконтролера. Таким чином, мікроконтролер може ініціювати надсилання повідомлення, перевіряти стан мережі або передавати дані на віддалений сервер.

Обмін даними між модулем і мікроконтролером відбувається через послідовний інтерфейс UART, що є стандартним способом інтеграції GSM-модемів у мікроконтролерні системи. Інтерфейс підтримує автоматичне

					КС КРБ 123.150.00.00 ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

визначення швидкості передачі даних та використання стандартних команд управління, описаних у специфікаціях 3GPP. Завдяки цьому SIM800L легко інтегрується з різними платформами, такими як NodeMCU.

Особливістю модуля є його відносно високе енергоспоживання під час передачі даних. Номінальна напруга живлення становить приблизно 4 В, а піковий струм під час передачі сигналу може досягати 2 А. Тому для стабільної роботи модуля необхідно використовувати надійне джерело живлення або стабілізатор напруги, здатний забезпечити достатній струм. Технічні характеристики GSM-модуля SIM800L наведено у таблиці 2.5.

Таблиця 2.5 – Характеристики GSM-модуля SIM800L

Характеристика	Значення
Тип модуля	GSM/GPRS модем
Робочі частоти	GSM 850 / 900 / 1800 / 1900 МГц
Тип мережі	2G GSM
Напруга живлення	3,4–4,4 В (рекомендовано 4 В)
Піковий струм споживання	до 2 А
Середній струм передачі	≈350 мА
Споживання у режимі очікування	<7 мА
Споживання у режимі сну	<2 мА
Максимальна швидкість передачі GPRS	до 85,6 кбіт/с
Інтерфейс зв'язку	UART
Керування	AT-команди
Тип SIM-карти	Micro-SIM
Робоча температура	–40...+85 °С

SIM800L підтримує передачу даних через GPRS-мережу, що дозволяє використовувати стандартні мережеві протоколи, такі як TCP/IP або HTTP. Максимальна швидкість у режимі GPRS може досягати приблизно 85,6 кбіт/с. Крім того, модуль підтримує додаткові функції, зокрема USSD-запити, годинник реального часу та інші можливості мобільного зв'язку.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

Використання GSM-модуля SIM800L у комп'ютерній системі інформування про зіткнення транспортних засобів є технічно обґрунтованим. Основна функція цього модуля полягає у передачі інформації про аварійну подію через мережу мобільного зв'язку. Після виявлення зіткнення за допомогою акселерометра та датчика вібрації мікроконтролер може сформулювати повідомлення, яке містить координати місця події (отримані з GPS-модуля) та іншу службову інформацію, і передати його через GSM-мережу.

Перевагою використання GSM-модуля є велика зона покриття мобільних мереж, що дозволяє передавати дані у районах, де відсутнє WiFi підключення. Крім того, SIM800L має компактні розміри, відносно низьку вартість та простий інтерфейс взаємодії з мікроконтролером, що робить його зручним для інтеграції у вбудовані системи.

2.3 Електрична принципова схема пристрою для інформування про зіткнення транспортних засобів

На рисунку 2.7 наведено електричну принципову схему розробленої системи, яка включає мікроконтролерний модуль NodeMCU, модуль акселерометра і гіроскопа MPU-6050, сенсор вібрації SW420, GPS-модуль GY-NEO6MV2, GSM-модуль SIM800L, кнопку керування, активний п'єзодинамік, понижуючий перетворювач напруги на базі мікросхеми LM2596 та допоміжні резистори.

Центральним елементом електричної схеми є мікроконтролерний модуль NodeMCU, який виконує функції керування всіма іншими компонентами системи. До його входів і виходів підключаються сенсорні модулі, навігаційний приймач, комунікаційний модуль та пристрої індикації. Живлення мікроконтролера здійснюється від стабілізованого джерела напруги 5 В, яке формується за допомогою перетворювача LM2596. Вихід перетворювача підключається до контакту VIN плати NodeMCU, а загальний провід — до контакту GND.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

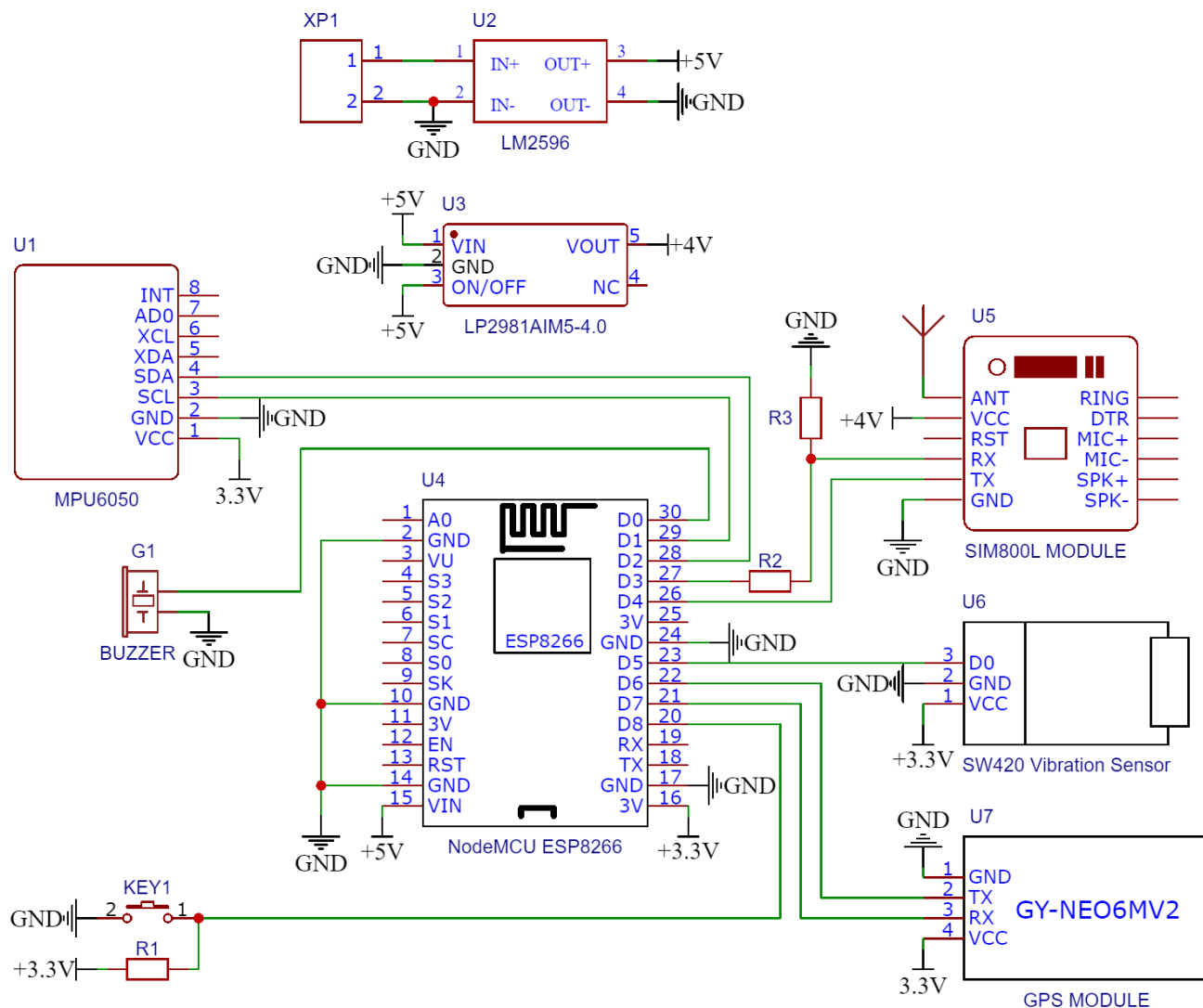


Рисунок 2.7 – Електрична принципова схема пристрою для інформування про зіткнення транспортних засобів

Для вимірювання параметрів руху транспортного засобу у системі використовується модуль акселерометра та гіроскопа MPU-6050, який підключається до мікроконтролера через інтерфейс I²C. Контакт SCL сенсора з'єднується з виводом D1 мікроконтролера, який використовується як тактовий сигнал шини I²C. Контакт SDA підключається до виводу D2, що використовується як лінія передачі даних.

Для фіксації механічних ударів або різких вібрацій у схемі використовується датчик вібрації SW420. Його цифровий вихід DO з'єднується з цифровим входом NodeMCU D5. У такій конфігурації мікроконтролер може зчитувати логічний стан сенсора та визначати факт виникнення механічної вібрації або удару.

Для визначення координат транспортного засобу використовується GPS-модуль GY-NEO6MV2, який підключається до мікроконтролера через послідовний інтерфейс UART. Контакт TX GPS-модуля підключається до цифрового входу D6 мікроконтролера, який використовується як приймальний канал програмного послідовного порту. Контакт RX може бути підключений до виводу D7, якщо передбачається можливість конфігурації модуля через мікроконтролер.

Передавання повідомлень про аварійні події здійснюється за допомогою GSM-модуля SIM800L, який також підключається через інтерфейс UART. Живлення цього модуля подається безпосередньо від виходу понижуючого перетворювача LP2981AIM5-4.0, налаштованого на напругу приблизно 4 В, що відповідає вимогам живлення SIM800L. Контакт TXD модуля підключається до цифрового входу D4 мікроконтролера, а контакт RXD — до цифрового виходу D3 через резистивний подільник напруги, який знижує логічний рівень сигналу до допустимого значення для входу модуля.

Також в схемі використовується кнопка, один контакт якої підключається до цифрового входу D8 мікроконтролера, а інший — до загальної шини. Для забезпечення стабільного рівня сигналу використовується підтягувальний резистор до напруги 3,3 В.

У схемі також передбачено активний п'єзодинамік, який використовується для формування звукових сигналів під час спрацювання системи. Позитивний контакт п'єзодинаміка підключається до цифрового виходу D0 мікроконтролера. Керування роботою п'єзодинаміка здійснюється програмно шляхом подачі логічних сигналів на відповідний вихід.

					<i>КС КРБ 123.150.00.00 ПЗ</i>	<i>Арк.</i>
						35
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Алгоритм роботи системи інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей

Алгоритм роботи комп'ютерної системи інформування про зіткнення транспортних засобів визначає послідовність виконання операцій, які забезпечують виявлення аварійної події, визначення координат місця її виникнення та передавання відповідного повідомлення до віддаленої інформаційної системи. Основою функціонування системи є мікроконтролерний модуль NodeMCU, який виконує оброблення даних сенсорів, керує роботою периферійних пристроїв і здійснює взаємодію з комунікаційним модулем.

Після ввімкнення живлення система переходить до етапу ініціалізації апаратних компонентів. На цьому етапі мікроконтролер налаштовує роботу цифрових та аналогових входів і виходів, ініціалізує інтерфейси зв'язку та перевіряє працездатність підключених модулів. Зокрема, виконується ініціалізація інтерфейсу I²C для взаємодії з модулем акселерометра та гіроскопа MPU-6050, а також налаштування послідовних інтерфейсів UART, які використовуються для обміну даними з GPS-модулем та GSM-модулем. Крім того, здійснюється конфігурація цифрових входів для зчитування стану сенсора вібрації SW420 та кнопки ручної активації системи.

Після завершення ініціалізації система переходить у режим постійного моніторингу параметрів руху транспортного засобу (рис. 3.1). На цьому етапі мікроконтролер періодично зчитує дані з акселерометра та гіроскопа MPU-6050, отримуючи інформацію про прискорення та зміну орієнтації транспортного засобу у просторі. Одночасно здійснюється контроль стану датчика вібрації SW420, який дозволяє виявити механічні удари або сильні коливання транспортного засобу.

					<i>КС КРБ 123.150.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ватиліяк В.А.</i>			<i>Практична частина</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Стадник Н.Б.</i>					<i>36</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>						

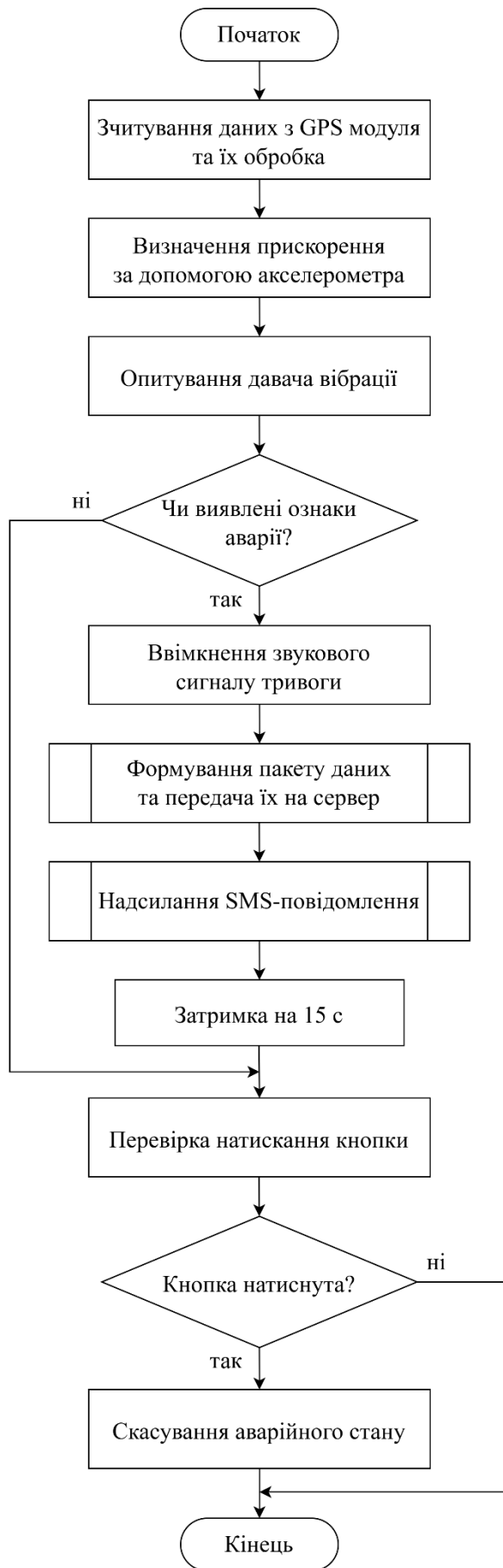


Рисунок 3.1 – Блок-схема алгоритму роботи циклічної підпрограми loop у системі дистанційного виявлення лісових пожеж

Отримані дані аналізуються програмним алгоритмом з метою виявлення характерних ознак ДТП. Якщо значення прискорення перевищує встановлений поріг або сенсор вібрації фіксує сильний механічний удар, система переходить до етапу перевірки аварійної події. На цьому етапі проводиться додатковий аналіз даних сенсорів для зменшення ймовірності хибного спрацювання системи. Аналізуються зміни прискорення по декількох осях та оцінюється різка зміна кута нахилу транспортного засобу. Якщо результати аналізу підтверджують наявність аварійної ситуації, система реєструє подію зіткнення.

Після фіксації аварійної події мікроконтролер активує процедуру визначення координат місця події. Для цього здійснюється зчитування навігаційних даних з GPS-модуля. Мікроконтролер приймає NMEA-повідомлення, що містять інформацію про широту, довготу, швидкість руху та точний час. Із отриманих даних виділяються координати поточного місцезнаходження транспортного засобу, які надалі використовуються у повідомленні про аварію.

Після отримання координат система формує інформаційне повідомлення, яке містить дані про факт зіткнення, географічні координати місця події. Для інформування користувача про спрацювання системи мікроконтролер активує п'єзодинамік, який генерує короткий звуковий сигнал.

Наступним етапом алгоритму є передавання сформованого повідомлення через мережу мобільного зв'язку. Для цього мікроконтролер надсилає відповідні AT-команди до GSM-модуля SIM800L через послідовний інтерфейс. GSM-модуль встановлює з'єднання з мережею мобільного оператора та здійснює передачу даних через GPRS-мережу до віддаленої IoT-платформи. У хмарному середовищі ці дані можуть бути використані для моніторингу аварійних подій або автоматичного інформування відповідних служб.

У системі також передбачено можливість ручного вимкнення звукового сигналу тривоги. Для цього використовується кнопка, підключена до одного з цифрових входів мікроконтролера. У разі натискання кнопки система виконує ручне скидання аварійного стану системи. Це дозволяє запобігти повторному надсиланню повідомлень у випадку хибного спрацювання або після усунення наслідків події.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

3.2 Розробка програмного забезпечення

3.2.1 Ініціалізація апаратних модулів та налаштування системи

Підпрограма `setup()` виконує початкову ініціалізацію всіх апаратних та програмних компонентів системи після подачі живлення (рис. 3.2).

```
void setup() {  
  Serial.begin(115200);  
  Wire.begin();  
  pinMode(VIBRATION_PIN, INPUT);  
  pinMode(BUTTON_PIN, INPUT_PULLUP);  
  pinMode(BUZZER_PIN, OUTPUT);  
  gpsSerial.begin(9600);  
  gsmSerial.begin(9600);  
  // MPU6050 init  
  if (!mpu.begin()) {  
    Serial.println("MPU6050 not found!");  
    while (1);  
  }  
  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);  
  mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);  
  initGSM();  
}
```

Рисунок 3.2 – Лістинг коду підпрограми `setup()`

На початку підпрограми викликається функція `Wire.begin()`, яка запускає інтерфейс I²C для взаємодії з акселерометром і гіроскопом MPU6050. Це забезпечує можливість передачі даних між мікроконтролером та сенсором по двопровідній шині.

Наступним етапом є конфігурація портів введення/виведення мікроконтролера. Контакт, до якого підключено давач вібрації, налаштовується як вхідний, що дозволяє зчитувати його логічний стан. Контакт кнопки ініціалізується в режимі `INPUT_PULLUP`. Контакт бузера конфігурується як вихідний, що дозволяє формувати керуючі сигнали для генерації звукового оповіщення.

Далі виконується ініціалізація програмних послідовних інтерфейсів для GPS- та GSM-модулів за допомогою викликів `gpsSerial.begin(9600)` та

					КС КРБ 123.150.00.00 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

gsmSerial.begin(9600). Обрана швидкість 9600 бод є стандартною для більшості подібних модулів і забезпечує надійний обмін інформацією.

Особливу увагу приділено ініціалізації акселерометра MPU6050. За допомогою виклику функції `mpu.begin()` здійснюється перевірка наявності та працездатності сенсора. У випадку, якщо пристрій не відповідає, у послідовний порт виводиться повідомлення про помилку, після чого виконання програми зупиняється за допомогою нескінченного циклу. Такий підхід дозволяє уникнути некоректної роботи системи у разі відсутності критично важливого сенсора.

Після успішної ініціалізації сенсора виконується його конфігурація. Встановлюється діапазон вимірювання прискорення `MPU6050_RANGE_8_G`, що забезпечує можливість реєстрації значних перевантажень, характерних для дорожньо-транспортних пригод. Також задається смуга пропускання цифрового фільтра `MPU6050_BAND_21_HZ`, що дозволяє зменшити вплив високочастотних шумів та підвищити точність вимірювань.

На завершальному етапі підпрограми викликається функція `initGSM()`, яка відповідає за початкове налаштування GSM-модуля, зокрема встановлення параметрів GPRS-з'єднання. Це забезпечує готовність системи до передавання даних на віддалений сервер та надсилання SMS-повідомлень у разі виникнення аварійної ситуації. Таким чином, підпрограма `setup()` забезпечує повну підготовку всіх компонентів системи до подальшої роботи в основному циклі програми.

3.2.2 Реалізація основного циклу функціонування системи

Підпрограма `loop()` реалізує безперервний цикл роботи мікроконтролерної системи та забезпечує виконання основних функціональних задач у режимі реального часу. Вона виконується циклічно після завершення підпрограми ініціалізації `setup()` і відповідає за періодичне опитування сенсорів, аналіз отриманих даних, обробку подій та ініціювання передачі інформації у випадку виявлення аварійної ситуації (рис. 3.3).

На початку кожної ітерації циклу викликається функція `readGPS()`, яка відповідає за зчитування та обробку даних з GPS-модуля. У процесі її виконання

					КС КРБ 123.150.00.00 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

відбувається декодування вхідних NMEA-повідомлень та оновлення значень географічних координат. Це дозволяє системі постійно мати актуальну інформацію про місцезнаходження транспортного засобу, яка використовується при формуванні аварійного повідомлення.

```
void loop() {  
  readGPS();  
  checkCrash();  
  if (crashDetected) {  
    Serial.println("ALERT ACTIVE");  
    sendToUbidots();  
    sendSMS();  
    delay(15000); // антиспам  
  }  
  checkButton();  
  delay(500);  
}
```

Рисунок 3.3 – Лістинг коду підпрограми loop()

Далі викликається функція checkCrash(), яка виконує ключову роль у системі, оскільки здійснює аналіз даних акселерометра та давача вібрації. У межах цієї функції визначається наявність різких змін прискорення або механічних коливань, що можуть свідчити про зіткнення. У разі перевищення заданих порогових значень встановлюється логічний прапорець crashDetected, який сигналізує про виникнення аварійної події.

У разі, якщо змінна crashDetected набуває значення істини, система переходить до виконання блоку обробки аварійної ситуації. Спочатку викликається функція sendToUbidots(), яка відповідає за передавання даних до хмарної IoT-платформи через GSM-модуль із використанням GPRS-з'єднання. Передаються ключові параметри, зокрема факт аварії та координати місцезнаходження. Після цього викликається функція sendSMS(), яка забезпечує оперативне інформування користувача або відповідних служб шляхом надсилання текстового повідомлення із зазначенням координат.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

Для запобігання надмірному навантаженню на мережу та уникнення багаторазового дублювання повідомлень передбачено затримку тривалістю 15000 мс після обробки аварійної події. Ця затримка виконує функцію антиспаму та забезпечує інтервал між повторними передаваннями даних.

Після цього виконується функція `checkButton()`, яка забезпечує можливість ручного скидання аварійного стану системи. Якщо користувач натискає кнопку, відповідний сигнал зчитується мікроконтролером, і прапорець аварії скидається. Це дозволяє запобігти повторному надсиланню повідомлень у випадку хибного спрацювання або після усунення наслідків події.

У кінці кожної ітерації циклу реалізовано додаткову затримку тривалістю 500 мс, яка визначає частоту опитування сенсорів і стабілізує роботу системи. Таким чином, підпрограма `loop()` забезпечує безперервний моніторинг стану транспортного засобу, своєчасне виявлення аварійних ситуацій та оперативне передавання відповідної інформації.

3.2.3 Зчитування та обробка даних акселерометра

Підпрограма `readAcceleration()` призначена для отримання та обробки даних з акселерометра MPU6050 і є ключовим елементом у реалізації алгоритму виявлення аварійних ситуацій. Вона забезпечує зчитування миттєвих значень прискорення за трьома осями та обчислення узагальненого показника, який використовується для оцінки динамічного стану транспортного засобу (рис. 3.4).

```
float readAcceleration() {
    sensors_event_t a, g, temp;
    mpu.getEvent(&a, &g, &temp);
    float totalAccel = sqrt(
        a.acceleration.x * a.acceleration.x +
        a.acceleration.y * a.acceleration.y +
        a.acceleration.z * a.acceleration.z
    );
    return totalAccel;
}
```

Рисунок 3.4 – Лістинг коду підпрограми `readAcceleration()`

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		42

На початку підпрограми оголошуються змінні типу `sensors_event_t`, які використовуються для зберігання даних акселерометра, гіроскопа та температурного сенсора. Далі викликається метод `mpu.getEvent()`, який здійснює зчитування поточних значень з давача та записує їх у відповідні структури. Використання цієї функції дозволяє отримувати дані у фізичних одиницях вимірювання, зокрема прискорення у метрах за секунду в квадраті, що значно спрощує подальшу обробку.

Після отримання значень прискорення за трьома координатними осями виконується обчислення повного (результуючого) прискорення. Для цього використовується формула евклідової норми вектора, яка передбачає піднесення кожної компоненти прискорення до квадрату, їх сумування та обчислення квадратного кореня. Такий підхід дозволяє отримати єдине скалярне значення, що характеризує інтенсивність руху незалежно від напрямку прикладених сил.

Обчислений показник повного прискорення є важливим параметром для визначення аварійних ситуацій, оскільки різкі зміни цього значення можуть свідчити про зіткнення або інші небезпечні події. Використання узагальненого значення замість аналізу окремих осей підвищує надійність системи та зменшує вплив орієнтації сенсора.

На завершальному етапі підпрограми обчислене значення повертається як результат функції. Це дозволяє використовувати його в інших частинах програми, зокрема у підпрограмі визначення аварійної ситуації. Таким чином, функція `readAcceleration()` реалізує ефективний механізм збору та первинної обробки даних про рух транспортного засобу, що є основою для подальшого аналізу та прийняття рішень у системі.

3.2.4 Зчитування та обробка координат GPS

Підпрограма `readGPS()` призначена для приймання, декодування та збереження даних про географічне положення транспортного засобу, отриманих від GPS-модуля. Вона є важливою складовою системи, оскільки забезпечує визначення координат місця події у випадку виникнення аварійної ситуації та їх подальше передавання до віддалених сервісів або користувачів (рис. 3.5).

					КС КРБ 123.150.00.00 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

```

void readGPS() {
    while (gpsSerial.available()) {
        | gps.encode(gpsSerial.read());
    }
    if (gps.location.isValid()) {
        | latitude = gps.location.lat();
        | longitude = gps.location.lng();
    }
}

```

Рисунок 3.5 – Лістинг коду підпрограми readGPS()

На початку підпрограми реалізовано цикл while, який перевіряє наявність доступних даних у буфері програмного послідовного інтерфейсу gpsSerial. У разі, якщо дані присутні, вони по одному байту зчитуються за допомогою функції gpsSerial.read() та передаються до методу gps.encode(). Цей метод бібліотеки TinyGPS++ виконує поетапне декодування NMEA-повідомлень, що надходять від GPS-модуля. Такий підхід дозволяє обробляти потік даних у реальному часі без необхідності накопичення повного пакета повідомлень.

Після обробки доступних даних виконується перевірка коректності отриманої інформації за допомогою методу gps.location.isValid(). Дана перевірка дозволяє визначити, чи були отримані валідні координати, що відповідають реальному положенню об'єкта. У випадку, якщо координати є достовірними, відбувається оновлення значень глобальних змінних latitude та longitude шляхом виклику методів gps.location.lat() та gps.location.lng() відповідно.

Використання такої логіки дозволяє уникнути запису некоректних або неповних даних, що можуть виникати на етапі пошуку супутників або при нестабільному сигналі. Завдяки цьому система оперує лише перевіреною інформацією про місцезнаходження, що є критично важливим для забезпечення точності інформування у разі аварії.

Підпрограма readGPS() реалізує ефективний механізм обробки даних супутникової навігації, забезпечуючи постійне оновлення координат транспортного засобу та їх готовність до використання у функціях моніторингу і передавання інформації.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

3.2.5 Реалізація алгоритму виявлення аварійної ситуації

Підпрограма `checkCrash()` реалізує основну логіку виявлення аварійної ситуації на основі аналізу даних, отриманих від акселерометра та давача вібрації. Вона є центральним елементом системи, оскільки саме в її межах приймається рішення про наявність або відсутність зіткнення транспортного засобу (рис. 3.6).

```
void checkCrash() {  
    float accel = readAcceleration();  
    readVibration();  
    // Виявлення тільки якщо ще не було зафіксовано  
    if (!crashDetected && (accel > accelThreshold || vibrationDetected)) {  
        crashDetected = true;  
        Serial.println("CRASH DETECTED!");  
        buzzerAlarm();  
    }  
}
```

Рисунок 3.6 – Лістинг коду підпрограми `checkCrash()`

На початку підпрограми викликається функція `readAcceleration()`, яка забезпечує зчитування поточного значення повного прискорення, обчисленого на основі даних акселерометра MPU6050. Отримане значення зберігається у змінній `accel` і використовується для подальшого аналізу. Використання узагальненого показника прискорення дозволяє оцінити динамічний стан об'єкта незалежно від його просторової орієнтації.

Далі викликається підпрограма `readVibration()`, яка відповідає за зчитування стану давача вібрації. У результаті її виконання оновлюється логічна змінна `vibrationDetected`, яка сигналізує про наявність або відсутність механічних коливань. Такий підхід дозволяє використовувати вібраційний сенсор як додатковий канал підтвердження аварійної події.

Після отримання даних із сенсорів виконується умовна перевірка, в якій аналізується перевищення значення прискорення заданого порогового рівня `accelThreshold` або факт виявлення вібрації. Якщо хоча б одна з цих умов виконується, система інтерпретує ситуацію як потенційне зіткнення. У цьому

					КС КРБ 123.150.00.00 ПЗ	Арк.
						45
Змн.	Арк.	№ докум.	Підпис	Дата		

випадку логічна змінна `crashDetected` встановлюється в істинне значення, що слугує сигналом для інших частин програми про виникнення аварійної події.

Додатково у межах цієї умови здійснюється виведення повідомлення "CRASH DETECTED!" у послідовний порт, що використовується для діагностики та моніторингу роботи системи під час налагодження. Також викликається функція `buzzerAlarm()`, яка активує звуковий сигнал за допомогою бузера, забезпечуючи локальне оповіщення про виявлену подію.

Підпрограма `checkCrash()` реалізує комбінований підхід до виявлення аварійних ситуацій, поєднуючи аналіз різких змін прискорення та вібраційних впливів. Це дозволяє підвищити достовірність спрацювання системи та зменшити ймовірність хибних сповіщень.

3.2.6 Передавання даних на IoT-платформу через GSM-мережу

Підпрограма `sendToUbidots()` реалізує механізм передавання даних про стан системи та параметри аварійної події до хмарної IoT-платформи із використанням GSM-модуля та GPRS-з'єднання. Вона є ключовим елементом підсистеми віддаленого моніторингу, оскільки забезпечує доставку інформації до серверної частини для подальшої обробки, зберігання та візуалізації (рис. 3.7).

На початковому етапі у підпрограмі формується HTTP-запит у вигляді JSON-об'єкта, який містить основні параметри стану системи. Для цього створюється рядок `payload`, у який послідовно додаються значення змінних `crashDetected`, `vibrationDetected`, а також координати `latitude` і `longitude`. Значення логічних змінних перетворюються у числовий формат (1 або 0), що відповідає вимогам IoT-платформи. Координати передаються з точністю до шести знаків після коми, що забезпечує достатню точність визначення місцезнаходження. Сформований JSON-пакет є тілом HTTP-запиту, який буде передано на сервер.

Далі виконується послідовність AT-команд для ініціалізації HTTP-з'єднання через GSM-модуль. Команда `AT+HTTPINIT` активує HTTP-сервіс у модулі, після чого задається ідентифікатор GPRS-контексту. Наступним кроком встановлюється адреса сервера шляхом передачі URL, який вказує на API IoT-платформи. Це дозволяє визначити кінцеву точку, куди буде надіслано дані.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

```

void sendToUbidots() {
  String url = "http://industrial.api.ubidots.com/api/v1.6/devices/esp8266";
  String payload = "{";
  payload += "\"crash\":";
  payload += crashDetected ? "1" : "0";
  payload += ",";
  payload += "\"vibration\":";
  payload += vibrationDetected ? "1" : "0";
  payload += ",";
  payload += "\"lat\":";
  payload += String(latitude, 6);
  payload += ",";
  payload += "\"lng\":";
  payload += String(longitude, 6);
  payload += "}";
  gsmSerial.println("AT+HTTPINIT");
  delay(1000);
  gsmSerial.println("AT+HTTTPARA=\"CID\",1");
  delay(500);
  gsmSerial.print("AT+HTTTPARA=\"URL\",");
  gsmSerial.print(url);
  gsmSerial.println("");
  delay(1000);
  gsmSerial.print("AT+HTTTPARA=\"USERDATA\", \"X-Auth-Token: ");
  gsmSerial.print(token);
  gsmSerial.println("");
  delay(1000);
  gsmSerial.println("AT+HTTTPARA=\"CONTENT\", \"application/json\"");
  delay(500);
  gsmSerial.print("AT+HTTPDATA=");
  gsmSerial.print(payload.length());
  gsmSerial.println(",10000");
  delay(1000);
  gsmSerial.print(payload);
  delay(1000);
  gsmSerial.println("AT+HTTPACTION=1");
  delay(5000);
  gsmSerial.println("AT+HTTPTERM");
  delay(1000);
  Serial.println("Data sent via GSM to Ubidots");
}

```

Рисунок 3.7 – Лістинг коду підпрограми sendToUbidots()

Для забезпечення авторизації запиту використовується команда AT+HTTTPARA="USERDATA", яка додає HTTP-заголовок X-Auth-Token із відповідним токеном доступу. Це дозволяє ідентифікувати пристрій у системі та забезпечує безпечний обмін даними. Додатково встановлюється тип вмісту запиту за допомогою параметра "CONTENT","application/json", що вказує на формат переданих даних.

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		47

Після налаштування параметрів виконується передавання сформованого JSON-пакета. Для цього використовується команда AT+HTTPDATA, яка визначає довжину даних та максимальний час очікування їх передачі. Після отримання підтвердження модулем у послідовний інтерфейс передається сам JSON-пакет. Далі виконується команда AT+HTTPACTION=1, яка ініціює HTTP POST-запит до сервера. У цей момент відбувається фактичне передавання даних через мобільну мережу.

Після завершення запиту виконується команда AT+HTTPTERM, яка деактивує HTTP-сесію та звільняє ресурси GSM-модуля. Це є важливим для запобігання накопиченню помилок та перевантаженню модуля при повторних викликах підпрограми. На завершення у послідовний порт виводиться повідомлення про успішне передавання даних, що може бути використано для діагностики роботи системи.

3.2.7 Реалізація функції надсилання SMS-повідомлення

Підпрограма sendSMS() призначена для реалізації механізму оперативного інформування про аварійну ситуацію шляхом надсилання текстового повідомлення через GSM-модуль. Вона забезпечує передачу ключової інформації про подію на мобільний телефон користувача або відповідних служб, що є важливим елементом підвищення швидкості реагування у випадку ДТП (рис. 3.8).

```
void sendSMS() {
    gsmSerial.println("AT+CMGF=1");
    delay(1000);
    gsmSerial.println("AT+CMGS=\"+380XXXXXXXXXX\"");
    delay(1000);
    gsmSerial.println("Accident detected!");
    gsmSerial.print("Lat: ");
    gsmSerial.println(latitude);
    gsmSerial.print("Lng: ");
    gsmSerial.println(longitude);
    gsmSerial.print("Vibration: ");
    gsmSerial.println(vibrationDetected);
    gsmSerial.write(26);
    delay(5000);
}
```

Рисунок 3.8 – Лістинг коду підпрограми sendSMS()

									Арк.
									48
Змн.	Арк.	№ докум.	Підпис	Дата					

На початковому етапі підпрограми виконується налаштування GSM-модуля у режим роботи з текстовими повідомленнями за допомогою AT-команди AT+CMGF=1. Ця команда переводить модуль у текстовий режим (Text Mode), що дозволяє передавати повідомлення у вигляді звичайного тексту без необхідності використання PDU-кодування. Після цього передбачена затримка, яка забезпечує коректну обробку команди модулем.

Далі виконується команда AT+CMGS із зазначенням номера телефону отримувача у міжнародному форматі. Ця команда ініціює процес створення нового SMS-повідомлення. Після її виконання GSM-модуль переходить у режим очікування тексту повідомлення, що передається через послідовний інтерфейс. Затримка після цієї команди необхідна для забезпечення готовності модуля до прийому текстових даних.

У наступній частині підпрограми формується текст повідомлення, який містить інформацію про виявлену аварійну подію. Спочатку передається текстове повідомлення "Accident detected!". Далі у повідомлення додаються координати місцезнаходження транспортного засобу, що формуються на основі значень змінних latitude та longitude. Також передається інформація про стан давача вібрації, що може слугувати додатковим підтвердженням події. Використання такої структури повідомлення дозволяє отримувачу швидко оцінити ситуацію та визначити місце події.

Після формування тексту повідомлення виконується команда gsmSerial.write(26), яка відповідає передачі керуючого символу Ctrl+Z. Цей символ є сигналом завершення введення тексту та ініціює безпосереднє надсилання SMS-повідомлення через мережу мобільного зв'язку. Після цього реалізується затримка, необхідна для очікування завершення процесу відправлення повідомлення.

Підпрограма sendSMS() забезпечує реалізацію простого та ефективного механізму аварійного інформування, який функціонує незалежно від доступності Інтернет-з'єднання та дозволяє оперативно передавати критично важливу інформацію користувачу або відповідним службам.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

3.3 Налаштування хмарної IoT платформи Ubidots для реалізації системи

Для реалізації функції дистанційного моніторингу та отримання повідомлень про аварійні ситуації у розробленій комп'ютерній системі було використано хмарну IoT платформу Ubidots. Дана платформа забезпечує зручні засоби для приймання, зберігання, оброблення та візуалізації даних, що надходять від віддалених пристроїв через мережу Інтернет. Крім того, платформа підтримує використання протоколів HTTP та MQTT, що дозволяє легко інтегрувати мікроконтролерні системи з хмарною інфраструктурою.

Після створення облікового запису на платформі було виконано налаштування середовища для приймання даних від розробленого пристрою. На першому етапі у веб-інтерфейсі платформи було створено новий пристрій (Device), який представляє фізичну систему інформування про зіткнення транспортних засобів. Для цього у розділі керування пристроями було додано новий елемент із відповідною назвою, що дозволило сформувати логічний контейнер для подальшого зберігання та оброблення даних. Створений пристрій отримав унікальний ідентифікатор, який використовується у програмному коді мікроконтролера під час передачі інформації до хмарного сервісу.

На наступному етапі було налаштовано змінні (Variables), які призначені для приймання та зберігання даних, що надходять від системи. Зокрема, було створено змінні для зберігання координат місця аварійної події, а саме географічної широти (latitude) та довготи (longitude). Ці змінні автоматично формуються у системі після надсилання відповідного JSON-повідомлення від мікроконтролера. Після створення змінних платформа почала зберігати отримані дані у вигляді часових рядів, що дозволяє здійснювати їх подальший аналіз та візуалізацію (рис. 3.9).

Для забезпечення авторизованого доступу до хмарного сервісу було використано спеціальний ключ доступу – токен API (Token). Цей токен було згенеровано у налаштуваннях облікового запису платформи та використано у програмному коді мікроконтролера. Під час передавання даних система надсилала HTTP-запит до серверу платформи із заголовком автентифікації, який містив

					КС КРБ 123.150.00.00 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

зазначений токен. Такий підхід забезпечує захист даних та запобігає несанкціонованому доступу до пристрою.

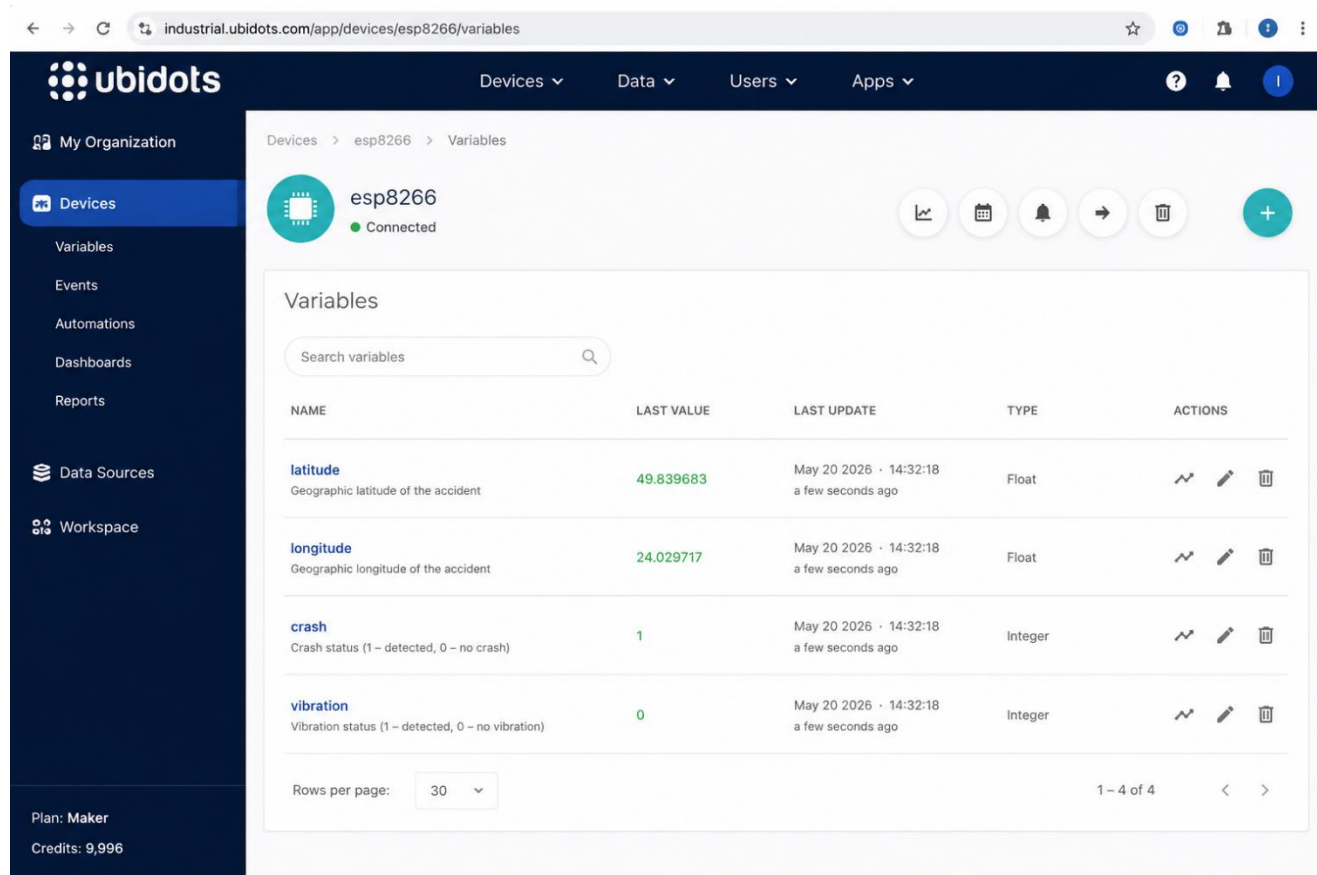


Рисунок 3.9 – Процес створення змінних в Ubidots

Після налаштування приймання даних було реалізовано їх візуалізацію. Для цього у середовищі платформи було створено інформаційну панель (Dashboard), яка дозволяє відображати поточні та історичні дані, отримані від пристрою. На даній панелі було розміщено віджети для відображення координат місця події, а також елементи графічного інтерфейсу, що дозволяють контролювати появу нових аварійних повідомлень. Використання таких віджетів забезпечує зручний доступ до інформації про стан системи та дозволяє швидко оцінити ситуацію.

Окремим етапом налаштування платформи було створення механізму автоматичних сповіщень. Для цього було використано функціональність подій та тригерів (Events). У відповідному розділі платформи було створено правило, яке спрацьовує у випадку надходження нових координат аварії або зміни стану

відповідної змінної. Після спрацювання правила система автоматично формує повідомлення та надсилає його користувачеві через обраний канал зв'язку. У якості способу інформування може використовуватися електронна пошта, SMS або інші інтегровані сервіси.

Після завершення налаштування було проведено тестування працездатності системи. Для цього мікроконтролер передавав тестові повідомлення із координатами до хмарної платформи. У результаті перевірки було підтверджено, що дані успішно приймаються сервером, відображаються на інформаційній панелі та зберігаються у базі даних. Таким чином, реалізована інтеграція мікроконтролерного пристрою з хмарною платформою забезпечила можливість дистанційного контролю та отримання інформації про аварійні ситуації. Використання платформи Ubidots дозволило значно спростити процес організації віддаленого моніторингу та підвищити ефективність функціонування розробленої комп'ютерної системи.

3.4 Моделювання системи

На етапі перевірки працездатності системи інформування про зіткнення транспортних засобів було проведено моделювання її роботи у спеціалізованому середовищі проєктування електронних пристроїв Circuit Designer.

На першому етапі було створено модель електричної схеми системи у середовищі Circuit Designer. Після додавання компонентів було виконано їх з'єднання відповідно до розробленої електричної принципової схеми.

Після завершення побудови схеми було проведено перевірку її коректності за допомогою інструментів аналізу, доступних у середовищі моделювання. Було перевірено правильність підключення живлення, відсутність коротких замикань та коректність з'єднання сигнальних ліній між мікроконтролером і периферійними модулями. Додатково було здійснено візуальний контроль структури схеми з метою переконання у відповідності моделі розробленої електричній принциповій схемі системи. У результаті проведеного моделювання було підтверджено

					КС КРБ 123.150.00.00 ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

правильність структури електронної частини системи та можливість її подальшої практичної реалізації (рис. 3.10).

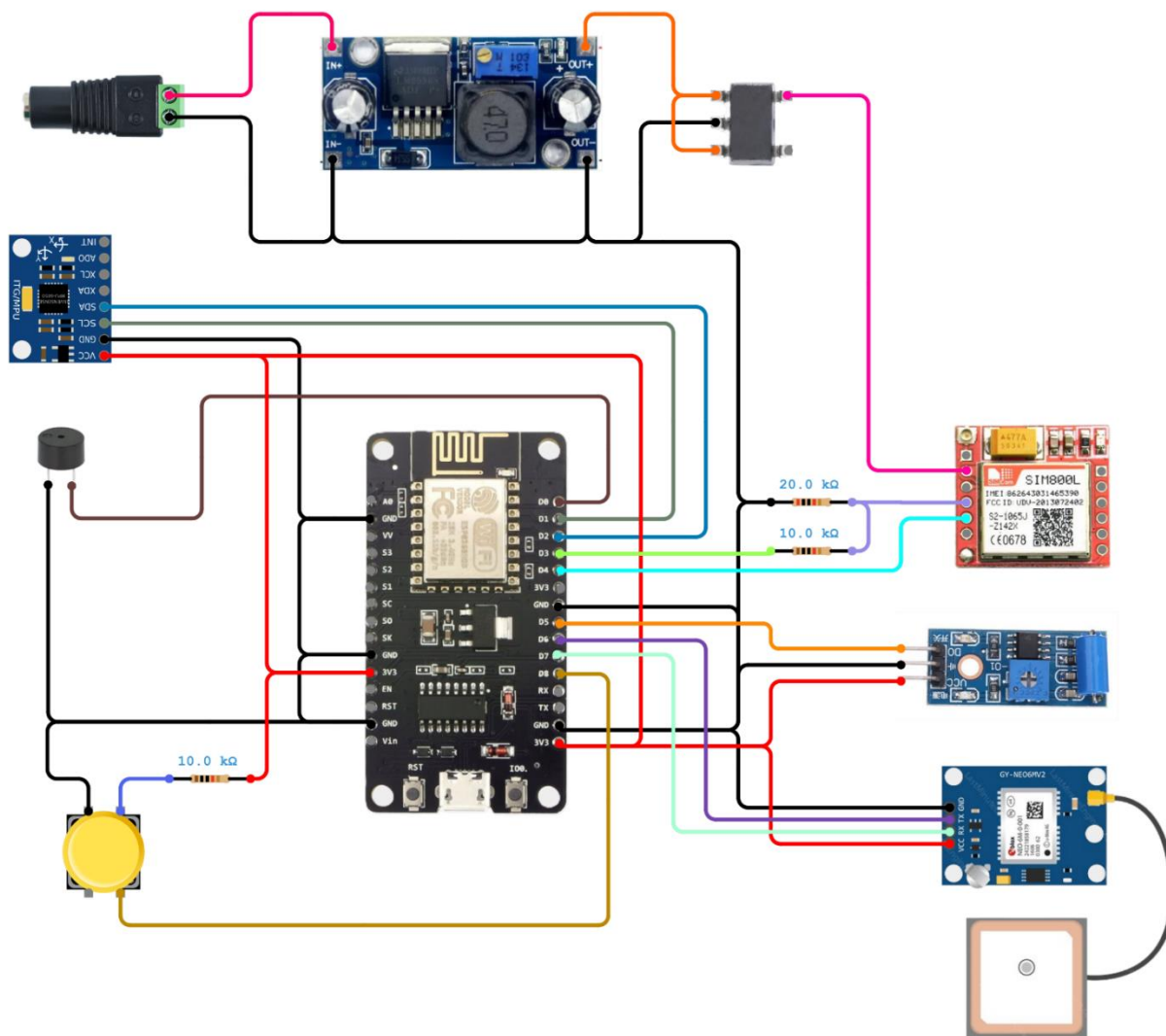


Рисунок 3.10 – Модель системи інформування про зіткнення транспортних засобів

На наступному етапі було виконано тестування програмного алгоритму роботи системи. Для цього у середовищі розробки було підготовлено програмний код для мікроконтролера, який реалізує зчитування даних з акселерометра, аналіз сигналу сенсора вібрації, визначення координат за допомогою GPS-модуля та передавання повідомлення про аварію через GSM-мережу до хмарної IoT-платформи. Після компіляції програмного коду було проведено аналіз його логіки, що дозволило переконатися у правильності реалізації алгоритму оброблення даних сенсорів та формування повідомлення про аварійну подію.

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

У результаті проведеного моделювання було підтверджено працездатність розробленої системи. Система успішно виконувала моніторинг параметрів руху транспортного засобу, виявляла аварійні ситуації, визначала координати місця події та передавала відповідні повідомлення до хмарної IoT-платформи. Отримані результати підтвердили правильність вибору елементної бази, ефективність розробленого алгоритму роботи та можливість практичного використання запропонованого технічного рішення для підвищення безпеки транспортних засобів.

					<i>КС КРБ 123.150.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		54

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Долікарська допомога при кровотечах

Кровотеча може виникнути в будь-якої людини навіть при роботі за ПК під час виконання поставлених завдань розробникам. Зазвичай це капілярна кровотеча, але може виникнути і масивніший крововилив. Необхідно вміти надавати базову допомогу при таких випадках, оскільки це може врятувати чиєсь життя або своє власне. Кровотеча виникає через порушення цілісності кровоносних судин. Загальний обсяг циркуляційної крові у дорослої людини – близько п'ять літрів. Загрозою для життя є втрата понад 30 % цього обсягу, особливо в короткі терміни. Кров зазвичай втрачається через рану. Раною називається пошкодження шкіри, слизової оболонки чи глибоких тканин, що супроводжується болем та кровотечею і має вигляд зяючого отвору [31].

Мікроби потрапляють до рани разом із предметом, яким заподіяно рану, а також із землею, шматками одягу, повітрям і при дотиках до рани руками. Найнебезпечнішими є мікроорганізми, що розвиваються у рані за умови відсутності повітря і спричиняють газову гангрену. Небезпечним ускладненням ран є зараження їх збудником правця. Рани можуть бути поверхневими або проникаючими у порожнину черепа, грудну клітину, черевну порожнину. Проникаючі рани – найнебезпечніші. Профілактика зараження рани – найшвидше накладення на неї асептичної пов'язки для запобігання доступу мікробів.

Причини кровотечі — пошкодження цілості кровоносних судин внаслідок механічного або патологічного порушення. За формою прояву кровотечі бувають:

- зовнішні – коли видно місце, звідки тече кров;
- внутрішні – коли кров виливається у внутрішні порожнини чи тканини.

Залежно від виду пошкоджених кровоносних судин кровотечі бувають артеріальні, венозні та капілярні.

					<i>КС КРБ 123.150.00.00 ПЗ</i>			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Розробив</i>		<i>Ватиліяк В.А.</i>			<i>Безпека життєдіяльності, основи охорони праці</i>	<i>Літ.</i>	<i>Арк.</i>	<i>Акрушів</i>
<i>Перевірив</i>		<i>Стадник Н.Б.</i>					55	6
<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>Осунівська Г.М.</i>						

Артеріальна кровотеча характеризується яскраво-червоним кольором крові, кров б'є сильним струменем, поштовхами. Венозна кров має темно червоне забарвлення, витікає з рани безперервно і повільно. При капілярній кровотечі кров виділяється краплями або сочиться з усієї поверхні рани.

Кровотечу необхідно якнайшвидше зупинити. Капілярна кровотеча добре зупиняється стискаючою пов'язкою, перед цим шкіру навколо рани обробляють розчином йоду, спирту, горілки, одеколону. Якщо з рани виступає сторонній предмет, його треба локалізувати і закріпити, для цього необхідно зробити у пов'язці отвір, інакше цей предмет може ще глибше проникнути всередину і викликати ускладнення [31].

Венозну кровотечу теж зупинити не дуже важко. Іноді досить підняти кінцівку, максимально зігнути її у суглобі обробити шкіру навколо рани, накласти стисну пов'язку і забинтувати. Для тимчасової зупинки артеріальної кровотечі здійснюють притискування артерії до кістки вище від місця поранення. Притискування здійснюють в тих місцях, де артерія знаходиться неглибоко, декількома пальцями однієї чи обох рук, а іноді навіть кулаком.

Скроневу артерію притискають попереду мочки вуха до виличної кістки. При кровотечі з головної шийної (сонної) артерії рану, по можливості, стискають пальцем, після чого набивають великою кількістю марлі, тобто роблять тампонування.

Для тимчасової зупинки кровотечі при пораненні передпліччя використовують різке згинання руки в ліктьовому суглобі, а у випадках пошкодження судин на нозі нижче коліна — різке згинання ноги у колінному суглобі, підклавши в ліктьову чи підколінну ямку пакунок чи згорток з марлі, вати.

Найнадійнішим методом тимчасової зупинки артеріальної кровотечі з пораними кінцівок є накладення гумового джгута. Кінцівку в місці накладання джгута обгортають марлею, рушником чи іншою тканиною, підіймають, джгут розтягують і роблять ним 2-3 оберти навколо кінцівки. Кінці джгута скріплюють за допомогою ланцюжка з крючком, а в разі їх відсутності — зв'язують. Якщо джгут накладено правильно, пульс нижче місця накладання зникає [31].

					КС КРБ 123.150.00.00 ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		

Тривалість використання джгута обмежується двома годинами, а взимку — однією годиною, у протилежному випадку кінцівка мертвіє. Необхідно залишити записку із зазначенням точного часу накладання джгута, щоб медики могли правильно оцінити ситуацію та вчасно надати необхідну допомогу. Якщо протягом цього часу немає можливості забезпечити додаткову допомогу, то через 1,5-2,0 години джгут на кілька хвилин відпускають (до почервоніння шкіри), кровотечу при цьому зменшують іншими методами (тампоном), а потім знову затягують джгут, трохи відступивши від попереднього місця його накладання.

У разі відсутності джгута накладають закрутку з паска, рушника, хустки або іншого матеріалу, який не туго зав'язують навколо кінцівки. В петлю вставляють палицю і закручують. Для того щоб не пошкодити шкіру, під закрутку необхідно підкласти бинт чи іншу тканину. Після накладання джгута чи закрутки потерпілого потрібно якомога швидше доставити в медичний заклад.

Часто кровотечі виникають через пошкодження м'яких тканин внаслідок удару і є однією з ознак цього виду травми. Іншими ознаками удару м'яких тканин є болі в місці удару, розлита припухлість (крововилив), обмежена рухомість ушкодженої частини тіла.

Перша допомога у випадку пошкодження м'яких тканин за умови відсутності в цьому місці перелому чи вивиху — холод на місце удару (рушник, змочений холодною водою, пузир з льодом чи снігом) та туга пов'язка на місце крововиливу і спокій ушкодженій частині тіла [32].

Травми легень супроводжуються відхаркуванням яскраво-червоної спіненої крові. При цьому дихання утруднене. Для надання допомоги хворого кладуть у напівлежаче положення, під спину підкладають валик, на груди кладуть холодний компрес. Потерпілому забороняється говорити і рухатись, необхідна госпіталізація.

Пошкодження органів черевної порожнини супроводжується значними болями, блюванням. У разі пошкодження печінки чи нирок з'являється внутрішня кровотеча. Кровотеча з травного тракту характеризується блюванням темно-червоною кров'ю, що зіслася. Для надання допомоги положення потерпілому

					КС КРБ 123.150.00.00 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

забезпечується те саме, що й при кровотечі з легень, але ноги згинаються в колінах. Потерпілого потрібно негайно відправити до лікарні.

В разі пошкодження м'яких тканин голови, спостерігаються припухлість, крововиливи (гематома) та болі в місці удару. Сильні удари здатні викликати і закриті пошкодження головного мозку. Поранення м'яких тканин на голові супроводжується сильними і тривалими кровотечами. Для надання допомоги потрібно прикласти холод на місце удару, а в разі сильної кровотечі потрібно накласти бинт навколо голови.

Працівники, які використовуватимуть комп'ютерну систему інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей, повинні знати правила надання долікарської допомоги при кровотечах.

4.2 Вплив вібрації на організм людини і розроблення заходів щодо зниження його рівня

Вібрація приводить тіло або його частини в коливальний рух. Розрізняють поперечні, поздовжні або крутильні коливання. Залежно від дії на людину вібрації ділять на місцеві і загальні. Загальні вібрації викликають шок в людини, місцеві залучають до коливальних рухів лише окремі частини тіла.

Загальна вертикальна вібрація викликає численні реакції в організмі людини. Ступінь дії вібрації характеризується [32]:

- станом основних нервових процесів в центральній нервовій системі (збудження і гальмування);
- реакціями з боку серцево-судинної системи (змінami серцевої діяльності);
- загальним станом: стомленням, появою болю і інших неприємних відчуттів (жару, нудоти, відчуття трясіння внутрішніх органів і т.д.).

Вібрації викликають неприємне відчуття, яке найчастіше виявляється при резонансних для організму частотах. Місцеві вібрації викликаються дією,

					КС КРБ 123.150.00.00 ПЗ	Арк.
						58
Змн.	Арк.	№ докум.	Підпис	Дата		

головним чином, ручного механізованого інструменту. Вібрації частотою 35-250 Гц і вище розвивають вібраційну хворобу із спазмом (звуженням) кровоносних судин кінцівок.

Вимірювання рівня вібрації і дослідження його характеристик є основними в рішенні питань боротьби з вібрацією будь-якого виду устаткування приладів і апаратів. Вимірювання вібрацій виконують в такій послідовності [33]:

- виявляють найбільш вібраційні вузли і вимірюють спектри на робочих місцях (в устаткуванні, пультах управління і т.п.);
- визначають тривалість дії вібрації на обслуговуючий персонал;
- значення виміряних рівня вібрації порівнюють з допустимими значеннями і з'ясовують ступінь їх відповідності.

Найнадійнішим і дешевим методом боротьби з вібрацією є використання пружинних, гумових або комбінованих амортизацій. Якщо конструктивними, технологічними і будівельно-планувальними заходами неможливо понизити рівень вібрації до гранично допустимих значень, застосовують індивідуальні засоби захисту.

Вібробезпечні умови праці забезпечують:

- застосування засобів вібраційного захисту які зменшують вібрацію, що впливає на шлях його розповсюдження;
- проєктувальне вирішення технологічних процесів і виробничих приміщень;
- організація технологічними заходами.

Носіння засобів індивідуального захисту (ЗІЗ) є додатковим, але не основним методом зниження впливу вібрацій. Спеціальні антивібраційні рукавички, виготовлені із комбінованих матеріалів (гелева амортизуюча вкладка, еластичний текстиль), здатні зменшити передачу вібрацій на кисть до 30–40 %, проте ефективність суттєво залежить від відповідності розміру рукавичок та правильності способу їх використання. Для зниження загальних вібрацій виробники пропонують сидіння з багатоступінчастою системою демпфування і пневматичними пружинами, які адаптуються до ваги оператора. Застосування

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		59

таких сидінь дозволяє зменшити вібропередачу на хребет приблизно на 50 % порівняно з жорсткими посадковими місцями [33].

Оцінювання ефективності заходів захисту від виробничої вібрації включає регулярний моніторинг як інструментальними методами, так і суб'єктивною оцінкою самопочуття працівників. Періодичні вимірювання вібраційного фону за допомогою спеціалізованих приладів дозволяють фіксувати рівень вібрації на робочому місці та порівнювати його з гранично допустимими нормами.

Крім технічного контролю, важливим є опитування персоналу щодо симптомів вібраційної хвороби, таких як оніміння кінцівок, біль у суглобах, порушення чутливості та втомлюваність. Для раннього виявлення можливих професійних захворювань рекомендовано проводити профілактичні медичні огляди працівників, які працюють з вібраційним обладнанням, не рідше ніж двічі на рік. До складу медичної комісії доцільно включати невролога, ортопеда та судинного хірурга. У разі виявлення порушень необхідно приймати відповідні медичні, реабілітаційні чи організаційні рішення, включаючи тимчасове відсторонення від роботи або переведення на інші умови праці [33].

Працівники, які використовуватимуть комп'ютерну систему інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей, повинні знати та дотримуватись заходів щодо зниження впливу вібрації на організм людини.

					<i>КС КРБ 123.150.00.00 ПЗ</i>	<i>Арк.</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		60

ВИСНОВКИ

У результаті виконання кваліфікаційної роботи розроблено систему інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей, призначену для автоматичного виявлення аварійних ситуацій і оперативного передавання інформації про місце їх виникнення до віддаленої інформаційної системи.

У ході виконання роботи проведено аналіз існуючих технічних засобів і рішень для виявлення ДТП та систем дистанційного моніторингу транспортних засобів. На основі виконаного аналізу було визначено основні функціональні вимоги до системи, обґрунтовано вибір архітектури пристрою та визначено оптимальний склад апаратних компонентів.

Розроблено структуру та електричну принципову схему комп'ютерної системи, яка включає мікроконтролерний модуль NodeMCU ESP8266, акселерометр і гіроскоп MPU-6050, сенсор вібрації SW420, GPS-модуль GY-NEO6MV2, GSM-модуль SIM800L, а також допоміжні елементи. Запропонована схема забезпечує зчитування параметрів руху транспортного засобу, визначення його координат та передавання даних через мережу мобільного зв'язку.

Розроблено алгоритм функціонування системи та програмне забезпечення для мікроконтролера, яке забезпечує зчитування даних із сенсорів, аналіз різких змін прискорення та вібрації, визначення координат місця події за допомогою супутникової навігації та передавання повідомлення про аварію на хмарну IoT-платформу Ubidots.

У процесі роботи було виконано моделювання електронної схеми системи у спеціалізованому програмному середовищі. Це підтвердило працездатність системи, коректність виявлення аварійних ситуацій, визначення координат місця події.

					<i>КС КРБ 123.150.00.00 ПЗ</i>	<i>Арк.</i>
						<i>61</i>
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Bonyár A., Géczy A., Krammer O., Sántha H., Illés B., Kámán J., Harsányi G. A review on current eCall systems for autonomous car accident detection. In 2017 40th International Spring Seminar on Electronics Technology (ISSE). 2017. P. 1-8.
2. Rifat Yıldız E., Bitirim Y. A smart emergency notification system for road accident, fire, and injury cases. Journal of Imaging Science and Technology, 2020. 64(3), P. 1-10.
3. Chowdhury A., Kaisar S., Khoda M.E., Naha R., Khoshkholghi M.A., Aiash M. IoT-based emergency vehicle services in intelligent transportation system. Sensors, 2023, 23(11). P. 5324.
4. 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.
5. Лупенко С.А., Пасічник В.В., Тиш Є.В. Комп'ютерна логіка. Навчальний посібник. Львів: Видавництво «Магнолія 2006», 2024. 354 с.
6. Буров Є., Митник М. Комп'ютерні мережі. (у 2-х томах). Львів, Магнолія, 2018. 740 с.
7. Лупенко С.А., Луцик Н.С., Лупенко А.М., Стадник Н.Б. Лінійний циклічний випадковий процес як математична модель тестових коливних сигналів у інформаційних системах діагностики, аутентифікації та прогнозування. Вісник Національного університету Львівська політехніка. Інформаційні системи та мережі, 2014 (783), С. 145-153.
8. Лупенко С.А., Литвиненко Я.В., Осухівська Г.М., Стадник Н.Б., Сверстюк А.С. Модифікація програмного комплексу для автоматизованого визначення морфологічних та ритмічних діагностичних ознак за електрокардіосигналами. Вісник Хмельницького національного університету. Технічні науки, 2020 (1), С. 137-146.
9. 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.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

10. 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.

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. 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 (ITTAP 2024), Ternopil, Ukraine, Opole, Poland, October 23-25, 2024. Vol. 3896. P. 12-24.

13. 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.

14. 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.

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 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.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						63
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 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.

20. 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.

21. 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.

22. 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.

23. Palamar A., Karpinskyy 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. 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.

					КС КРБ 123.150.00.00 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		64

25. 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.

26. 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.

27. Palamar A., Karpinskyu M. Control of an Uninterruptible Power Supply in a DC Microgrid System. 10th International 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.

28. 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.

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. Зеркалов Д.В. Безпека життєдіяльності. Навчальний посібник. К.: Основа, 2016. 267 с.

32. Ярошевська В.М. Безпека життєдіяльності. Підручник. – 2-е вид. К.: ВД „Професіонал”, 2006. 560 с.

33. Бедрій Я.І. Основи охорони праці : навч. посіб. 4-е вид. перероб. і доп. Тернопіль : Навчальна книга – Богдан, 2018. 240 с.

					КС КРБ 123.150.00.00 ПЗ	Арк.
						65
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 кафедру КС

_____ Осухівська Г.М.

“ 2 ” лютого 2026 р.

КОМП'ЮТЕРНА СИСТЕМА ІНФОРМУВАННЯ ПРО ЗІТКНЕННЯ
ТРАНСПОРТНИХ ЗАСОБІВ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ
СУПУТНИКОВОЇ НАВІГАЦІЇ ТА ІНТЕРНЕТУ РЕЧЕЙ

ТЕХНІЧНЕ ЗАВДАННЯ

на 8 листках

Вид робіт: Кваліфікаційна робота

На здобуття освітнього ступеня «Бакалавр»

Спеціальність 123 «Комп'ютерна інженерія»

«УЗГОДЖЕНО»

Керівник кваліфікаційної роботи

_____ к.т.н. Стадник Н.Б.

“ 2 ” лютого 2026 р.

«ВИКОНАВЕЦЬ»

Студент групи СІ-41

_____ Ватиляк В.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 інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей».

Умовне позначення кваліфікаційної роботи: КС КРБ 123.150.00.00.

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, Ватиляк Владислав 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 інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей. Основними регламентними документами при оформленні та пред'явленні результатів проєктування 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

Призначенням комп'ютерної системи інформування є підвищення рівня безпеки дорожнього руху шляхом автоматичного виявлення зіткнень транспортних засобів і оперативного інформування про аварійну подію. Система повинна забезпечувати безперервний моніторинг параметрів руху транспортного засобу, фіксацію критичних подій та передавання інформації про них без необхідності втручання користувача.

Проектована система може використовуватися в легкових і комерційних транспортних засобах, а також у дослідницьких або навчальних цілях для демонстрації можливостей сучасних вбудованих систем та IoT-платформ.

2.2 Мета створення системи

Метою створення системи є розроблення функціонального прототипу комп'ютерної системи інформування про зіткнення транспортних засобів, яка забезпечує автоматизоване визначення аварійних ситуацій, визначення координат місця події за допомогою супутникової навігації та передавання даних через мережу Інтернет на IoT-платформу для подальшої обробки та візуалізації.

2.3 Характеристика об'єкту

Об'єктом автоматизації є транспортний засіб як динамічний технічний об'єкт, що характеризується змінними параметрами руху, такими як прискорення, швидкість та просторове положення. Система встановлюється безпосередньо на транспортний засіб і функціонує в умовах впливу вібрацій, змін температури, вологості та інших зовнішніх факторів. Обмін даними здійснюється між бортовим пристроєм і віддаленим сервером або IoT-платформою.

3 Вимоги до системи

3.1 Вимоги до системи в цілому

Система повинна бути цілісною, автономною та придатною до тривалої безперервної роботи. Архітектура системи має забезпечувати взаємодію апаратної та програмної частин, а також можливість віддаленого доступу до зібраних даних. Рішення повинно відповідати вимогам енергоефективності, надійності та простоти експлуатації.

3.1.1 Вимоги до структури та функціонування системи

Структура системи повинна включати бортовий модуль збору та оброблення даних, навігаційний модуль супутникового позиціонування, модуль бездротового зв'язку та програмну частину, що забезпечує логіку функціонування. Система повинна функціонувати за принципом подієво-орієнтованої обробки, коли виявлення зіткнення ініціює передачу даних і формування повідомлення.

До основних функціональних компонентів системи належать:

- сенсорний модуль для фіксації різких змін параметрів руху;
- мікроконтролерний блок для оброблення даних;
- навігаційний модуль для визначення координат;
- комунікаційний модуль для передавання інформації.

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

Обмін даними між внутрішніми компонентами системи має здійснюватися за допомогою стандартних інтерфейсів, таких як I²C, SPI або UART. Для зв'язку з віддаленими сервісами система повинна використовувати бездротові канали передачі даних, зокрема Wi-Fi або мобільний інтернет. Передавання даних повинно здійснюватися з використанням протоколів, характерних для IoT-систем, із забезпеченням коректності та цілісності інформації.

3.1.3 Вимоги до режимів функціонування системи

Система повинна підтримувати декілька режимів функціонування, зокрема режим ініціалізації, робочий режим моніторингу та аварійний режим. У робочому режимі здійснюється безперервний збір і аналіз даних, тоді як в аварійному режимі активується передавання повідомлення про зіткнення.

Також має бути передбачений режим тестування для перевірки працездатності системи.

3.1.4 Перспективи розвитку та модернізації системи

Архітектура системи повинна забезпечувати можливість подальшого розширення функціональності без суттєвих змін базової структури. До перспектив розвитку належать:

- підключення додаткових сенсорів;
- інтеграція з мобільними застосунками;
- використання альтернативних IoT-платформ;
- розширення алгоритмів аналізу аварійних ситуацій.

3.1.5 Вимоги до надійності системи

Система повинна забезпечувати стабільну роботу в умовах експлуатації транспортного засобу. Програмне забезпечення має бути стійким до збоїв зв'язку та короткочасних відмов окремих компонентів. У разі втрати з'єднання з мережею Інтернет система повинна забезпечувати збереження даних для подальшої передачі.

Показники надійності системи інформування про зіткнення транспортних засобів повинні відповідати вимогам ДСТУ 50136-1. Ймовірність безвідмовної роботи системи повинна складати не менше 99,8 %.

3.1.6 Вимоги до функцій та задач, які виконує система

Система повинна виконувати такі основні функції:

- моніторинг параметрів руху транспортного засобу;
- виявлення факту зіткнення;
- визначення географічних координат місця події;
- формування та передавання повідомлень про аварію;

- збереження та візуалізація даних на IoT-платформі.

3.1.7 Вимоги до апаратного забезпечення

Апаратне забезпечення системи повинно базуватися на сучасній мікроконтролерній платформі з достатніми обчислювальними ресурсами та підтримкою необхідних інтерфейсів. Усі компоненти мають бути сумісними між собою та придатними для експлуатації в автомобільних умовах. Живлення системи повинно забезпечуватися від бортової мережі транспортного засобу або автономного джерела.

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

Поз. позначення	Найменування	Кіл.	Примітка
	<u>Перемикачі</u>		
KEY1	Вимикач кнопочний KE-011	1	
	<u>Резистори</u>		
R1, R2	0805-0,125-10 кОм±5%	2	
R3	0805-0,125-20 кОм±5%	1	
	<u>Модулі</u>		
G1	П'єзодинамік	1	
U1	Модуль акселерометра та гіроскопа MPU-6050	1	
U2	DC-DC конвертор понижувальний LM2596	1	
U3	Регулятор напруги LP2981AIM5-4.0	1	
U4	WiFi модуль NodeMCU на основі ESP8266	1	
U5	GSM-модуль SIM800L	1	
U6	Давач вібрації SW-420	1	
U7	GPS-модуль GY-NEO6MV2	1	
	<u>Роз'єми</u>		
XP1	Роз'єм ХН2.54-3Р 2-х контактний	1	

<i>КС КРБ 123.150.00.00 ПЕ</i>				
Змн.	Арк.	№ докум.	Підпис	Дата
Розробив		Ватиліак В.А.		
Перевірів		Стадник Н.Б.		
Рецензент		Деркач М.В.		
Н. Контр.		Луцик Н.С.		
Зав. каф.		Осухівська Г.М.		
Пристрій для інформування про зіткнення транспортних засобів з використанням технологій супутникової навігації та Інтернету речей Перелік елементів				
		Літ.	Арк.	Аркушів
			76	1
ТНТУ, каф. КС, гр. СІ-41				

Додаток В

Лістинг програми

Лістинг В.1 – Код програми мікроконтролера для реалізації системи інформування про зіткнення транспортних засобів.

```
#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include "secret.h"

// ----- Pins -----
#define VIBRATION_PIN D5
#define BUTTON_PIN D8
#define BUZZER_PIN D0
// ----- GPS -----
SoftwareSerial gpsSerial(D6, D7);
TinyGPSPlus gps;
// ----- GSM -----
SoftwareSerial gsmSerial(D3, D4);
// ----- MPU6050 -----
Adafruit_MPU6050 mpu;
// ----- Variables -----
float accelThreshold = 15.0;
bool crashDetected = false;
bool vibrationDetected = false;
float latitude = 0.0;
float longitude = 0.0;
unsigned long lastButtonPress = 0;
const unsigned long debounceDelay = 300;

// ----- Setup -----
void setup() {
  Serial.begin(115200);
  Wire.begin();
  pinMode(VIBRATION_PIN, INPUT);
  pinMode(BUTTON_PIN, INPUT_PULLUP);
  pinMode(BUZZER_PIN, OUTPUT);
  gpsSerial.begin(9600);
  gsmSerial.begin(9600);
  // MPU6050 init
  if (!mpu.begin()) {
    Serial.println("MPU6050 not found!");
    while (1);
  }
  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);
}
```

```

    initGSM();
}

// ----- LOOP -----
void loop() {
    readGPS();
    checkCrash();
    if (crashDetected) {
        Serial.println("ALERT ACTIVE");
        sendToUbidots();
        sendSMS();
        delay(15000); // антиспам
    }
    checkButton();
    delay(500);
}

// ----- GSM INIT -----
void initGSM() {
    delay(1000);
    gsmSerial.println("AT");
    delay(1000);
    gsmSerial.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"");
    delay(1000);
    gsmSerial.println("AT+SAPBR=3,1,\"APN\",\"internet\"");
    delay(2000);
    gsmSerial.println("AT+SAPBR=1,1");
    delay(3000);
}

// ----- MPU -----
float readAcceleration() {
    sensors_event_t a, g, temp;
    mpu.getEvent(&a, &g, &temp);
    float totalAccel = sqrt(
        a.acceleration.x * a.acceleration.x +
        a.acceleration.y * a.acceleration.y +
        a.acceleration.z * a.acceleration.z
    );
    return totalAccel;
}

// ----- GPS -----
void readGPS() {
    while (gpsSerial.available()) {
        gps.encode(gpsSerial.read());
    }
    if (gps.location.isValid()) {
        latitude = gps.location.lat();
        longitude = gps.location.lng();
    }
}
}

```

```

// ----- VIBRATION -----
void readVibration() {
  vibrationDetected = digitalRead(VIBRATION_PIN);
}

// ----- BUZZER -----
void buzzerAlarm() {
  for (int i = 0; i < 5; i++) {
    digitalWrite(BUZZER_PIN, HIGH);
    delay(200);
    digitalWrite(BUZZER_PIN, LOW);
    delay(200);
  }
}

// ----- Crash Detection -----
void checkCrash() {
  float accel = readAcceleration();
  readVibration();
  // Виявлення тільки якщо ще не було зафіксовано
  if (!crashDetected && (accel > accelThreshold ||
vibrationDetected)) {
    crashDetected = true;
    Serial.println("CRASH DETECTED!");
    buzzerAlarm();
  }
}

// ----- Button -----
void checkButton() {
  if (digitalRead(BUTTON_PIN) == LOW) {
    // антидребезг
    if (millis() - lastButtonPress > debounceDelay) {
      lastButtonPress = millis();
      if (crashDetected) {
        crashDetected = false;
        digitalWrite(BUZZER_PIN, LOW);
        Serial.println("ALARM RESET BY USER");
      }
    }
  }
}

// ----- Send to Ubidots via GSM -----
void sendToUbidots() {
  String url =
"http://industrial.api.ubidots.com/api/v1.6/devices/esp8266";
  String payload = "{";
  payload += "\"crash\":";
  payload += crashDetected ? "1" : "0";
  payload += ",";
  payload += "\"vibration\":";
  payload += vibrationDetected ? "1" : "0";
}

```

```

payload += ",";
payload += "\"lat\":";
payload += String(latitude, 6);
payload += ",";
payload += "\"lng\":";
payload += String(longitude, 6);
payload += "}";
gsmSerial.println("AT+HTTPIPINIT");
delay(1000);
gsmSerial.println("AT+HTTTPARA=\"CID\",1");
delay(500);
gsmSerial.print("AT+HTTTPARA=\"URL\",");
gsmSerial.print(url);
gsmSerial.println("");
delay(1000);
gsmSerial.print("AT+HTTTPARA=\"USERDATA\", \"X-Auth-Token: ");
gsmSerial.print(token);
gsmSerial.println("");
delay(1000);
gsmSerial.println("AT+HTTTPARA=\"CONTENT\", \"application/json\"");
delay(500);
gsmSerial.print("AT+HTTPDATA=");
gsmSerial.print(payload.length());
gsmSerial.println(",10000");
delay(1000);
gsmSerial.print(payload);
delay(1000);
gsmSerial.println("AT+HTTPACTION=1");
delay(5000);
gsmSerial.println("AT+HTTPTERM");
delay(1000);
Serial.println("Data sent via GSM to Ubidots");
}

// ----- SMS -----
void sendSMS() {
  gsmSerial.println("AT+CMGF=1");
  delay(1000);
  gsmSerial.println("AT+CMGS=\"+380XXXXXXXXXX\"");
  delay(1000);
  gsmSerial.println("Accident detected!");
  gsmSerial.print("Lat: ");
  gsmSerial.println(latitude);
  gsmSerial.print("Lng: ");
  gsmSerial.println(longitude);
  gsmSerial.print("Vibration: ");
  gsmSerial.println(vibrationDetected);
  gsmSerial.write(26);
  delay(5000);
}

```