

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені І.І.МЕЧНИКОВА

(повне найменування вищого навчального закладу)

Факультет математики, фізики та інформаційних технологій

(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних систем та технологій

(повна назва кафедри (предметної, циклової комісії))

## Дипломна робота

на здобуття ступеня вищої освіти «бакалавр»

(освітньо-кваліфікаційний рівень)

на тему

«Розробка портативного пристрою

для відстеження рухомих об'єктів»

(назва українською)

Development of a portable device for tracking moving objects

(назва англійською)

Виконав: студент денної форми навчання

спеціальності 123 Комп'ютерна інженерія.

(шифр і назва напрямку підготовки, спеціальності)

Езерович Дар'я Михайлівна

(прізвище, ім'я, по-батькові)

Керівник ст. викладач Берков Ю.М.

(науковий ступінь, вчене звання, прізвище та ініціали, підпис)

Рецензент к. ф.-м.н., доцент Якимчук В.І.

(науковий ступінь, вчене звання, прізвище та ініціали)

Рекомендовано до захисту:

Протокол засідання кафедри

№ 9 від «17» червня 2021 р.

Завідувач кафедри

Захищено на засіданні ЕК № \_\_\_\_

протокол № \_\_ від «\_\_» \_\_\_\_ 2021 р.

Оцінка \_\_\_\_ / \_\_\_\_ / \_\_\_\_

(за національною шкалою, шкалою ECTS, бали)

Голова ЕК

(підпис)

Ю.О. Гунченко

(прізвище, ініціали)

(підпис)

Н.Ф. Казакова

(прізвище, ініціали)

Одеса - 2021

## АНОТАЦІЯ

Метою роботи є розробка та створення прототипу портативного пристрою для відстеження рухомих об'єктів. Для чого був проведений аналіз існуючих і широко використовуваних в даний час подібних пристроїв в різних областях, порівняння і виділення основних вимог до їх характеристик. Зокрема розглянуті різні спеціалізовані пристрої стеження від компанії Trackit, Fixon. Окремо було досліджено сучасні мікроконтролери платформи, на основі чого було обрано елементну базу для створення прототипу. В якості бази для прототипування було обрано мікроконтролеру платформу Arduino (Arduino Nano), для відстеження місцеположення та забезпечення зв'язку застосовано модуль на чіпі SIM808 від компанії SIMCOM, що поєднує в собі блоки GPRS, GSM і GPS. В результаті сформульовані основні функції проєктованого пристрою. На підставі цього проведено вибір компонентів і побудова схеми їх підключення. Побудову схеми приладу проведено в програмі EasyEda. Для забезпечення функціонування пристрою було розроблено керуючий код, з застосуванням середовища програмування Arduino IDE.

Особливістю розробленого пристрою є те, що для визначення місцеположення об'єкту використовується дві технології GPRS та GPS. За замовченням при наявності та достатній потужності сигналу мобільного оператора використовується GPRS, що дозволяє більш точно визначити координати. В разі неможливості отримання мобільного сигналу, відбувається переключення на GPS навігацію.

## ANNOTATION

The aim of the work is to develop and create a prototype of a portable device for tracking moving objects. Why was the analysis of existing and currently widely used similar devices in different areas, comparison and selection of the basic requirements for their characteristics. In particular, various specialized tracking devices from Trackit, Fixon are considered. Modern platform microcontrollers were studied separately, on the basis of which the element base for prototype creation was selected. The Arduino platform (Arduino Nano) was chosen as the basis for prototyping for the microcontroller, and a module on the SIM808 chip from SIMCOM, which combines GPRS, GSM and GPS units, was used to track the location and provide communication. As a result, the main functions of the designed device are formulated. Based on this, the selection of components and the construction of their connection scheme. Construction of the device scheme is carried out in the EasyEda program. To ensure the operation of the device, a control code was developed using the Arduino IDE programming environment.

A feature of the developed device is that two GPRS and GPS technologies are used to determine the location of the object. By default, in the presence and sufficient signal strength of the mobile operator, GPRS is used, which allows you to more accurately determine the coordinates. If it is impossible to receive a mobile signal, switch to GPS navigation.

## АННОТАЦИЯ

Целью работы является разработка и создание прототипа портативного устройства для отслеживания движущихся объектов. Для чего был проведен анализ существующих и широко используемых в настоящее время подобных устройств в различных областях применения, сравнение и выделение основных требований их характеристик. В частности, рассмотрены различные специализированные устройства слежения от компании Trackit, Fixon. Также были исследованы современные микроконтроллерные платформы, на основе чего была избрана элементная база для создания прототипа. В качестве базы для прототипирования была избрана микроконтроллерная платформа Arduino (Arduino Nano), для отслеживания местоположения и обеспечения связи применен модуль на чипе SIM808 от компании SIMCOM, сочетающий в себе блоки GPRS, GSM и GPS. В результате сформулированы основные функции проектируемого устройства. На основании этого проведен выбор компонентов и построение схемы их подключения. Построение схемы прибора проведения в программе EasyEda. Для обеспечения функционирования устройства был разработан управляющий код, с применением среды программирования Arduino IDE.

Особенностью разработанного устройства является то, что для определения местоположения объекта используется две технологии GPRS и GPS. По умолчанию при наличии и достаточной мощности сигнала мобильного оператора используется GPRS, что позволяет более точно определить координаты. В случае невозможности получения мобильного сигнала, происходит переключение на GPS навигацию.

## ЗМІСТ

	стр.
ВСТУП .....	8
1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ .....	9
2 ПРОЕКТУВАННЯ ПРИСТРОЮ .....	15
2.1 Постановка задачі .....	15
2.2 Огляд компонентів та створення схеми пристрою.....	16
3 РОЗРОБКА ПРИСТРОЮ .....	20
4 РЕЗУЛЬТАТИ РОБОТИ ТА ІНСТРУКЦІЯ КОРИСТУВАЧА.....	30
ВИСНОВОК.....	38
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ .....	39
ДОДАТОК А.....	40
Програмний код .....	40

## ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

GSM – Глобальна система мобільного зв'язку (Global System for Mobile Communications), це міжнародний стандарт для мобільного цифрового стільникового зв'язку з розділенням каналу за принципом TDMA та високим рівнем безпеки за рахунок шифрування з відкритим ключем.

GPRS – Загальний сервіс пакетної радіопередачі (General Packet Radio Service), це стандарт, який використовує не зайняту голосовим зв'язком смугу частот для передачі інформації.

GPS – Система глобального позиціонування (Global Positioning System), це сукупність радіоелектронних засобів, що дозволяє визначати положення та швидкість руху об'єкта на поверхні Землі або в атмосфері

GNSS – Глобальна навігаційна супутникова система (ГНСС), це система, що дозволяє визначати просторове положення об'єктів місцевості шляхом опрацювання ГНСС–приймачем прийнятого супутникового сигналу.

ОЗП – це статична оперативна пам'ять з довільним доступом, використовується для зберігання і роботи змінних.

EEPROM – це енергонезалежна пам'ять, використовується для зберігання постійної інформації.

ШІМ – широко–імпульсна модуляція, це процес керування шириною (тривалістю) високочастотних імпульсів за законом, який задає низькочастотний сигнал.

UART – універсальний асинхронний приймач/передавач, це тип асинхронного приймача–передавача, компонентів комп'ютерів та периферійних пристроїв, що передає дані між паралельною та послідовною формами.

NITZ (Network Identity and Time Zone) – це механізм надання місцевого часу та дати, часового поясу та зсуву літнього часу, а також інформації про ідентифікацію постачальника послуг мобільних пристроїв через бездротову мережу.

NTP (Network Time Protocol) – це мережевий протокол для синхронізації внутрішніх годин комп'ютера з використанням мереж зі змінною латентністю.

DTMF (Dual–Tone Multi–Frequency) – це тональний набір, тональний сигнал, дво тональний багато частотний аналоговий сигнал, використовуваний для набору телефонного номера.

TCP / IP – це мережева модель передачі даних, представлених в цифровому вигляді.

З розвитком та розповсюдженням технологій в суспільстві зростає попит на пристрої відстеження місцезнаходження рухомих об'єктів. В кожній людині є велика кількість потенційних об'єктів потребуючих відстеження таких, як автомобіль, мопед або велосипед, маленька дитина на прогулянці, людини з обмеженими можливостями, домашній улюбленець, що любить втекти.

На сьогодні доступність технологій дозволяє саморуч створити безліч пристроїв, з використанням мікроконтролерів та різноманітних датчиків, що здатні робити безліч вимірювань майже будь яких даних. Тому проектування будь якого пристрою слід починати з огляду предметної області та формулювання задач користувача.

Процес створення будь якого пристрою залежить від вибору апаратних компонентів, що мають задовольняти всім функціям майбутньої системи. Коли всі компоненти підбрано, створюється схема елементів пристрою, після чого починається збірка макету, розробка та налагодження керуючого коду, що забезпечить працездатність усіх функцій приладу.

Метою роботи є розробка та створення прототипу портативного пристрою для відстеження рухомих об'єктів.

Завданнями роботи є огляд предметної області, формулювання функціональних вимог до пристрою, вибір компонентів, що задовольняють сформульованим вимогам, створення схеми підключення компонентів, розробка та налагодження керуючого коду.

## 1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

Існує велика кількість приладів відстеження рухомих об'єктів, загальні відмінності в комплектації яких залежить від предметної області приладу, але ці відмінності полягають лише в додаткових функціях пристроїв та програмного інтерфейсу.

Одна з можливих областей використання є відстеження автомобіля. Розглянемо певну комплектацію приладу відстеження пересування з цієї області в складі автомобільної сигналізації.

Автомобільні Сигналізації часто доповнюються приладами відстеження. Fixon пропонує багато різних конфігурацій приладів, що використовують наступні технології GSM/GPRS/GNSS. Розглянемо технічні характеристики однієї з конфігурацій (табл. 1.1).

Таблиця 1.1 – Технічні характеристики пристрою відстеження автомобіля

Параметр	Значення
Стандарт мобільного зв'язку	GSM 850/900/1800/1900 мГц
Канали передачі даних	GPRS, SMS для настройки, голосовий зв'язок
Клас каналу GPRS	12, TCP / IP
Супутникова навігація	GPS / ГЛОНАСС / GALILEO / BEIDOU
Антени	Вбудована GPS і GSM антени з високим коефіцієнтом посилення
Точність координат	до 5 метрів

Ця система пропонує ряд функцій:

- а) Сценарії – Безпечне водіння, Перевищення швидкості, Оповіщення про блокування GSM сигналу, Розрахунок витрати палива по GPS, Оповіщення – холостий хід, Виявлення від'єднання, Визначення

буксирування автомобіля, Визначення ДТП, Оповіщення про поїздки;

- б) Режими сну – Сплячий режим GPS, Сплячий Режим В мережі, Режим глибокого Сну, Режим Ультра глибоко Сну;
- в) Оновлення конфігурації і прошивки – FOTA Web, FOTA, TeltonikaConfigurator (USB, Bluetooth), FMBT мобільний додаток (Конфігуратор);
- г) SMS – Конфігурація, Події, Контроль цифрових виходів, Налагодження;
- д) GPRS команди – Конфігурація Контроль цифрових виходів, Налагодження;
- е) Синхронізація часу – GNSS, NITZ, NTP. [1]

Розглянувши одну з конфігурацій модулю відстеження автомобіля можна виділити ряд дуже зручних та вдалих функцій приладу. Наприклад розрахунок витрати палива використовуючи дані GPS, сплячий режим GPS, точність до 5 метрів та велика швидкість оновлення даних.

Взагалі Фіхон виділяє декілька характеристик відстежуваних пристроїв.

Для автомобіля GPS маячок виділяє наступні характеристики:

- а) Точність відстеження 3–5 метрів;
- б) Оновлення позиції автомобіля на карті кожні 5–10 секунд;
- в) Підключення до електромережі 12/24 В автомобіля;
- г) Вбудований мікрофон для прослуховування салону авто;
- д) Блокування (зупинка) двигуна в разі викрадення;
- е) Резервний акумулятор – 2–3 дня автономного живлення;
- ж) Відстеження на екрані телефону або комп'ютера, через сайт спостереження;
- з) Миттєве повідомлення на телефон, коли автомобіль рушив з місця або включився двигун;

Персональний GPS маячок виділяє наступні характеристики:

- а) Міні розміри – сірникової коробки;

- б) Точність відстеження 3–5 метрів;
- в) Вбудований мікрофон для прослуховування;
- г) Оновлення координат на карті кожні 5–10 секунд;
- д) Автономний акумулятор може тримати від 5 днів до 5 років заряд;
- е) Відстеження бачитися на екрані телефону або комп'ютера, через сайт спостереження. [2]

Окрім автомобілів можна виділити ще один транспортний засіб, який потребує відстеження є велосипед. Цей тип транспорту потребує відстеження не лише задля безпеки, але ще для відстеження спортивних досягнень.

Існує велика кількість трекерів для велосипедів, що мають велику різницю в точності вимірювання, що залежить від технічних характеристик приладів (табл. 1.2–1.3).

GPS трекер мітка прихована в ліхтарі розроблений спеціально для велосипеда. GPS трекер велосипедний ліхтар працює в мережі GSM / GPRS та GPS–супутників. Відстеження велосипеда по SMS, мобільному додатку та веб–відстеження програмного забезпечення!

Даний GPS трекер приймає самі передові технології gps та LBS, три способи позиціонування, Завдяки чому можна знайти дуже швидко і відправити точну інформацію про місцезнаходження. В GPS режимі пошуку, точність може бути менше 5 м.

GPS трекер для велосипеда ТК–906 виконаний у вигляді стоп маячка в спеціалізованому водонепроникному корпусі, що дозволить встановити його під сидіння вашого велосипеда.

Викрадач з великою ймовірністю його просто не помітить, а навіть якщо помітить, то навряд чи зможе швидко зняти, що дозволить вам швидко розшукати велосипед. Так само трекер можна встановити на дитячий велосипед, так Ви завжди будете знати де катається Ваша дитина, і де він був протягом дня. [3]

Таблиця 1.2 – Технічні характеристики пристрою відстеження велосипеда ТК–906

Параметр	Значення
Вологість	5% – 95%
Робоча температура	–20 ~ 55 °С
Мережа	2G
Тип корпусу	Пластик
Точність GPS	5–15 м
Частоти	850/900/1800 / 1900 мГц
Чутливість GPS	–159 dBm

Трекер Fixon пропонує іншу комплектацію та функції:

- а) Визначення точного місця розташування велосипеда на карті;
- б) Траєкторія руху велосипеда;
- в) Оприлюднення у момент пограбування велосипеда;
- г) Використання в якості компаса або навігатора;
- д) Вело Сигналізація з SMS оповіщенням;
- е) Онлайн відстеження з телефону;
- ж) Швидкість руху;
- з) Висота підйому і спуску;
- и) Експорт даних в GoogleEarth, Facebook, CSV, GPX. [4]

Таблиця 1.3 – Технічні характеристики пристрою відстеження велосипеда Fixon

Параметр	Значення
Стандарт мобільного зв'язку	GSM 850/900/1800/1900 мГц
Канали передачі даних	GPRS, SMS для настройки, голосовий зв'язок

Клас каналу GPRS	12, TCP / IP
Супутникова навігація	GPS / ГЛОНАСС / GALILEO / BEIDOU
Напруга електроживлення	5 В
Типовий струм	(12В) 20мА
Обсяг пам'яті для запису	2МБ (65 000 записів)
Резервний АКБ	700 мА
Температура експлуатації	від -30 ° С до +80 ° С
Клас захисту корпусу	IP64

Ще одним цікавим прикладом є GPS / GSM / GPRS трекер для велосипеда Trackit, який встановлюється в рульову стійку і нічим не відрізняється від стандартної кришки. Автономна робота – до 30 днів в режимі очікування.

Відстеження велосипеда по GPS активується магнітним ключем, вбудованим датчиком руху, або ж дистанційно за допомогою СМС.

Точність GPS – 5–25 м. Точність GSM – близько 200 м. GPRS необхідний для онлайн-відстеження через відповідні сервіси.

Інформація про локації висилає на авторизований номер (до 5 номерів одночасно) у вигляді посилання на GoogleMaps + номера осередку мережі. Ці дані надсилаються як за запитом (дзвінок на нього), так і через заданий інтервал часу або відстані. Крім координат, велотрекер може сповіщати по СМС про початок руху велосипеда, про надто високій швидкості і низькому заряді батареї.

У режимі очікування GSM трекер може перебувати до 1 місяця!

Для відстеження за даними мережі можна використовувати сервіс Cellidfinder.[5]

Окрім того, що наведені системи відрізняються інтерфейсом користувача системи, об'ємом заряду батареї. Найбільш стійким можна

виділити трекер Trackit, який може перебувати режимі очікування до місяцю, також в режимі GSM може давати найбільшу точність серед усіх наведених. Проте трекер Fixon надає більше функціоналу, що є корисним користувачу в повсякденному житті.

Тож дані порівняльні характеристики дають чітко зрозуміти, що реалізація великої кількості додаткових задач користувача може негативно вплинути на технічну характеристику пристрою.

## 2 ПРОЕКТУВАННЯ ПРИСТРОЮ

### 2.1 Постановка задачі

Після дослідження предметної області, огляду існуючих пристроїв для відстеження рухомих об'єктів визначимо загальні задачі для реалізації дипломного проекту. Для пристрою відстеження рухомих об'єктів було обрано наступні функції:

- а) Визначення місцеположення пристрою:
  - 1) Можливість запиту на отримання координат пристрою;
- б) SMS оповіщення власника пристрою:
  - 1) В разі якщо пристрій був без руху 15 хвилин;
  - 2) В разі якщо перевищено швидкість пересування;
  - 3) Про низький рівень заряду акумулятора пристрою;
- в) Можливість вмикати та вимикати SMS оповіщення;
- г) Можливість видалити телефон з пам'яті пристрою;
- д) Можливість подзвонити на пристрій, щоб знайти його за звуковим сигналом.

Приведений набір функцій дозволить користувачу розширити способи застосування пристрою. Основною функцією є передача координат пристрою на телефон власника, що дозволить визначити місцезнаходження пристрою. Функції сповіщення власника щодо нерухомості об'єкта відстеження та перевищення швидкості можуть бути корисними для відстеження своїх домашніх улюбленців. Сповіщення щодо перевищення швидкості може стати в нагоді для системи сигналізації автомобіля, чи будь якого іншого транспортного засобу. Завдяки можливості відправити звуковий сигнал на пристрій значно поліпшуються способи знайти втрачений трекер та об'єкт закріплений на ньому. Можливість додавати та видаляти номери телефонів власників з пам'яті пристрою дозволяють передавати пристрій від власника одного до іншого без перепрошивання трекера.

## 2.2 Огляд компонентів та створення схеми пристрою

Наступною важливою частиною проектування є вибір компонентів майбутнього приладу та побудова схеми підключення обраних компонентів. Побудову схеми приладу проведено в програмі EasyEda.

Розмір має велике значення для приладу даного типу, тому серед великої лінійки плат Arduino було обрано плату Arduino Nano (рис. 2.1). Обраний тип контролерів найчастіше використовується для пристроїв невеликого розміру.

Розглянемо технічні характеристики мікроконтролера Arduino Nano:

- а) Розміри 19 на 42 міліметрів;
- б) Вага 7 грамів;
- в) Флеш– пам'ять 16 або 32 Кб (в залежності від чіпу плати);
- г) ОЗУ 1 або 2 Кб (в залежності від чіпу плати)
- д) EEPROM 512 байт або 1 Кб;
- е) 14 цифрових пінів, 6 з яких можуть використовуватись як ШІМ;
- ж) 8 аналогових входів. [6]

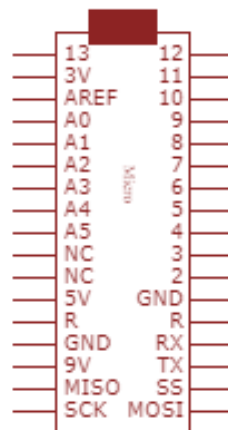


Рисунок 2.1 – Схема плати Arduino Nano

Найголовнішим компонентом проекту є модуль SIM808 (рис. 2.2). Розглянемо компактний модуль зв'язку на чіпі SIM808 від SIMCOM, який поєднує в собі GPRS, GSM модуль зв'язку і GPS навігацію. Однією з особливостей модуля є наявність на платі зарядного пристрою для 3.7 В LiPo акумулятора, що легко дозволить створювати повністю автономний пристрій.

Цей модуль обрано за найбільшу комплектацією серед усіх модулів SIM серії та широкий програмний функціонал, що забезпечує можливість створення широкого спектру дій без додавання зайвих модулів, як акселерометр, окремий GSM, GPRS та GPS модулі, годинник реального часу, та контролер заряду акумулятора. Тому що апаратно–програмна комплектація цього модулю поєдную в собі всі перелічені компоненти.

GPRS модуль працює в 4–х стандартних GSM діапазонах і суміщений в одному корпусі з навігаційним 66–канальним GPS приймачем. Управляти модулем може будь–який мікроконтролер 3.3 В–5 В з послідовним UART інтерфейсом. На цьому модулі можна побудувати автономну систему стеження за рухомим об'єктом з контролем його координат і швидкості переміщення. Micro USB роз'єм призначений тільки для підключення зовнішнього джерела живлення для зарядки акумулятора.

Технічні характеристики модулю SIM808:

- а) Діапазон: 850/900/1800/1900 мГц
- б) Параметри GPRS: GPRS mobile station class B
- в) Управління: AT командами
- г) Навігація: інтегрована GPS/CNSS з підтримкою A–GPS
- д) Режими енергозбереження: 1mA в режимі сну
- е) Підтримка протоколу: GPS NMEA
- ж) Тип SIM карти: Стандартна Micro Sim карта
- з) Передача даних: GPRS multi–slot class12. Максимальна швидкість 85.6 Кбіт/сек
- и) Індикатори: живлення і активність

- к) Роз'єми: UFL роз'єми для зовнішньої пасивної антени GPS і зовнішньої GSM антени
- л) Аудіо вхід/вихід: стандартний 4-контактний TRRS роз'єм гарнітури
- м) Розміри: 50 x 35 мм [7 – 8]

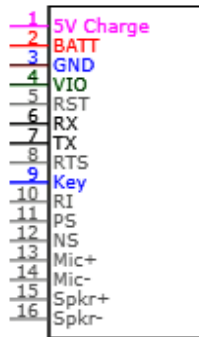


Рисунок 2.2–Схема модулю SIM808

Після того як живлення модуля SIM808 організовано, необхідно підключити його до пристрою мікро-контролеру Arduino. (рис. 2.3)

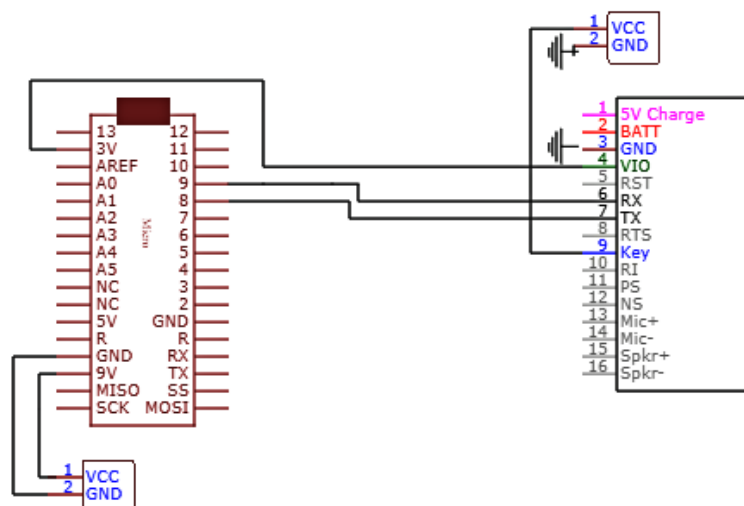


Рисунок 2.3 – Схема підключення модулю SIM808 до Arduino Nano

Для подачі звукового сигналу з пристрою, в разі потреби, додаємо п'єзо елемент.

Враховавши всі перелічені компоненти можемо побудувати схему пристрою для відстеження рухомих об'єктів. (рис. 2.4)

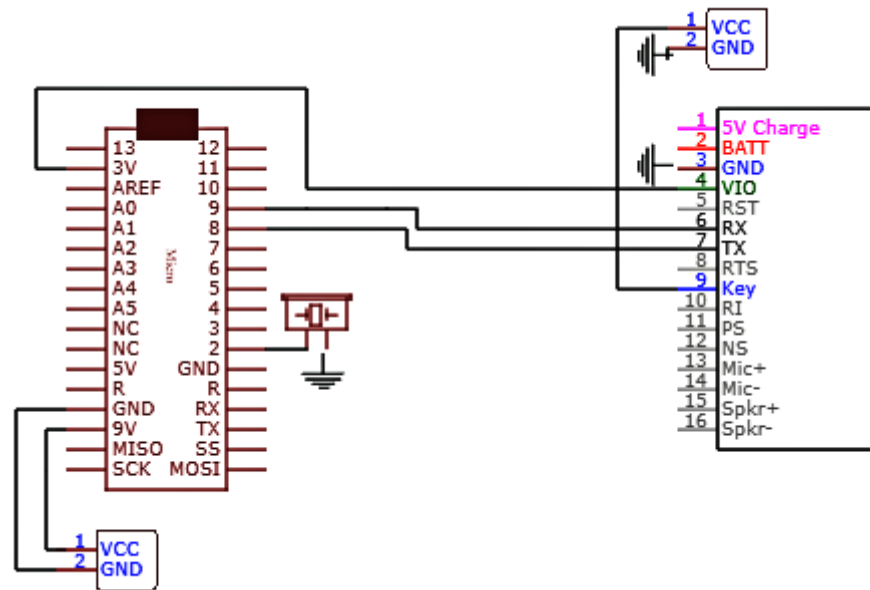


Рисунок 2.4 – Схема пристрою для відстеження рухомих об'єктів

Враховуючи все перелічене отримуємо пристрій, побудований на мікро контролері Arduino Nano, основним робочим компонентом є модуль SIM808, який забезпечить вирішення поставлених задач SMS та телефонного налаштування пристрою, та відстеження рухомих об'єктів. Живлення пристрою буде здійснюватися акумуляторами 18650, та додатковим акумулятором на 3,7V 1800 мА.

### 3 РОЗРОБКА ПРИСТРОЮ

Фінальною частиною розробки є написання програмного продукту, що забезпечить функціонування пристрою. Написання програмного коду відбуватиметься в середі програмування Arduino IDE. Скомпонувавши всі фізичні модулі пристрою, що забезпечать вирішення поставлених задач слід перейти до їх програмної реалізації.

#### Реєстрація номера телефону власника пристрою

Завдяки модулю SIM808 ми маємо можливість приймати SMS повідомлення та обробляти їх. Користуючись цією можливістю введемо реєстрацію телефону власника через SMS повідомлення. Зберігати номери телефонів краще в EEPROM (енергонезалежна пам'ять плати) Arduino Nano, щоб зберегти їх після вимкнення пристрою.

Для роботи з EEPROM необхідно підключити спеціальну бібліотеку EEPROM.h, ввести зміні phone та eepromAddress для зберігання номеру телефону та адреси запису даних до пам'яті EEPROM, та використати команду:

```
EEPROM.put(eepromAddress, msgphone);
```

Завдяки AT командам SIM модулю ми маємо можливість читати SMS повідомлення. AT команда «AT+CMGL="REC UNREAD"» дозволить перевірити непрочитані повідомлення, проте для аналізу тексту SMS необхідно створити окремий метод обробки результату запиту AT команди.

Метод аналізу тексту SMS повідомлень «parseSMS» дозволяє виділити з повідомлення та записати в окремі змінні номер телефону та текст повідомлення.

```
void parseSMS(String msg) {
  String msgheader = "";
  String msgbody = "";
  String msgphone = "";
  msg = msg.substring(msg.indexOf("+CMGR: "));
  msgheader = msg.substring(0, msg.indexOf("\r"));
```

```

msgbody = msg.substring(msgheader.length() + 2);
msgbody = msgbody.substring(0, msgbody.lastIndexOf("OK"));
msgbody.trim();
int firstIndex = msgheader.indexOf("\",\"") + 3;
int secondIndex = msgheader.indexOf("\",\"", firstIndex);
msgphone = msgheader.substring(firstIndex, secondIndex);
Serial.println("Phone: "+msgphone);
Serial.println("Message: "+msgbody);
if (msgbody.indexOf("1") > -1) {
    EEPROM.put(eepromAddress, msgphone);
}

```

Після аналізу тексту необхідно видалити всі повідомлення, щоб не перевантажувати пам'ять SIM модулю. Для видалення повідомлень слід використовувати AT команду «AT+CMGR», щоб зробити повідомлення прочитаним, а потім видалити всі прочитані повідомлення командою «AT+CMGDA=\"DEL READ\"».

Використовуючи метод читання та аналізу SMS повідомлень «parseSMS» отримуємо можливість запрограмувати налаштування приладу, за допомогою повідомлень. Додамо можливість реєстрації телефону власника приладу в EEPROM пам'ять, через відправку SMS повідомлення з цифрою 1 на номер SIM – карти пристрою.

```

if (msgbody.indexOf("1") > -1) {
    EEPROM.put(eepromAddress, msgphone);
}

```

#### Визначення місцеположення пристрою

AT команди SIM модулю мають декілька різних запитів для отримання місцеположення, що відрізняються протоколами обробки сигналів. AT команда «AT+CGPSINF=<mode>» дозволить отримати дані про місцеположення об'єкту в повному форматі: <longitude> – довгота, <latitude> – широта, <altitude> – висота, <UTC time> – час UTC, <speed> – швидкість, <course> – курс. Ще одна команда «AT+CGNSINF» дає можливість отримати більшу інформацію через те, що використовує інший протокол обробки сигналу, протокол GNSS. Змінюючи mode в команді «AT+CGPSINF=<mode>» ми маємо можливість отримувати різний обсяг даних, що допоможе в реалізації сформульованих задач. Обидві команди

використовують супутникові данні для отримання координат, проте супутникові данні мають велику похибку в координатах. Для використаного модуля середньоквадратичне відхилення точності позиціонування горизонтальних координат менш 2.5 м.

Більш точний метод є метод позиціонування за допомогою GSM та GPRS сигналів. Обраний модуль дозволяє працювати з GSM, GPRS та GPS. GSM Location – це визначення координат місця розташування за інформацією про базові станції. В обраному модулі SIM808 ця функція використовує базу Google і для роботи потрібне підключення до Інтернету. Дані про найближчих базових станціях надсилаються на сервіс Google по протоколу HTTP. У відповідь приходять дані з координатами найближчих базових станціях. Алгоритм розрахунку координат модуля заснований на відомості про основні базових станціях, до якої він зараз прив'язаний, трьох найближчих базових станціях і рівні сигналу. Всі розрахунки і запити приховані від користувача. Для отримання координат розташування модуля необхідно встановити GPRS-з'єднання і відправити команду «AT + CIPGSMLOC». Точність визначення координат залежить від кількості базових станціях в окрузі і якості сигналу, тому в місті можна отримати досить точні координати.

Через те, що GPRS забезпечує більшу точність визначення координат місцезнаходження пристрою залежить від якості сигналу та кількості базових станціях в окрузі пропонується забезпечити поєднання технологій GPRS та GPS. Залежно від якості сигналу мобільного оператора буде автоматично змінено технологію позиціонування. В разі наявності гарного сигналу мобільного оператора буде використовуватись технологія GPRS, в протилежному випадку або в разі отримання помилок від GPRS буде використано технологію GPS.

Для реалізації отримання місцеположення пристрою було введено декілька змінних location, longitude та latitude. Всі необхідні запити для

отримання координат приладу та відправки їх на номер телефону власника зібрано в один метод «getLocation».

```
void getLocation (){
    csq = sendATCommand("AT+CSQ", DEBUG);
    csq = splitString(csq, ',', 0);
    csqNumber = csq.toInt();
    sendATCommand("AT+SAPBR=3,1,CONTTYPE,GPRS", DEBUG);
    sendATCommand("AT+SAPBR=3,1,APN,internet", DEBUG);
    sendATCommand("AT+SAPBR=1,1", DEBUG);
    sendATCommand("AT+SAPBR=2,1", DEBUG);
    location = sendATCommand("AT+CIPGSMLOC=1,1", DEBUG);
    longitude = splitString(location, ',', 1);
    long = longitude.toDouble();
    latitude = splitString(location, ',', 2);
    lat = latitude.toDouble();
    if( csqNumber < 10 && (long != 0 && lat != 0)){
        message = longitude + " " + latitude;
        sendSMSAlert(phone, message);
    }
    if( csqNumber > 9 || (long == 0 && lat == 0)){
        location = sendATCommand("AT+CGPSINF=0", DEBUG);
        longitude = splitString(location, ',', 0);
        latitude = splitString(location, ',', 1);
        message = longitude + " " + latitude;
        sendSMSAlert(phone, message);
    }
}
```

SMS оповіщення власника пристрою, якщо пристрій був без руху 15 хвилин

Для відправлення SMS повідомлення з модулю слід використати AT команду «AT+CMGF» для ввімкнення текстового режиму SMS та команду «AT+CMGS=\"\" + phone + \"\"» для відправки повідомлення на збережений номер телефону. Перед виконанням описаної команди необхідно ввімкнути режим відправки повідомлень командою «AT+CMGF=1;&W». В повному обсязі відправлення повідомлень реалізовано окремими методами «sendSMS» та «sendSMSAlert», в яких ми передаємо текст повідомлення та номер телефону та обробляємо результати відправлення повідомлення.

```
void sendSMS(String phone, String messageText)
{
    sendATCommand("AT+CMGS=\"\" + phone + \"\"", DEBUG);
```

```

        sendATCommand(messageText + "\r\n" + (String)((char)26),
DEBUG);
    }
    void sendSMSAlert(String phone, String message){
        sendSMS(phone, message);
        if (SIM800.available()) {
            _response = waitResponse();
            _response.trim();
            Serial.println(_response);
            if (_response.startsWith("+CMGS:")) {
                int index = _response.lastIndexOf("\r\n");
                String result = _response.substring(index + 2,
_response.length());
                result.trim();
                if (result == "OK") {
                    Serial.println ("Message was sent. OK");
                }
                else {
                    Serial.println ("Message was not sent.
Error");
                }
            }
        }
    }
}

```

Для перевірки стану об'єкту використаємо AT команду «AT+CGPSINF=<mode>», описану вище, але з різними параметрами. Використовуючи параметр 64, ми отримаємо дані швидкості пересування об'єкту, а з параметром 128 час в UTC форматі. Проте кожна з команд видає інформацію в певному форматі, тому для того, щоб виявити потрібні дані, використаємо метод «splitString», який дозволяє взяти з результату запиту числа з певної позиції.

```

String splitString(Stringdata, charseparator, intindex)
{
    intfound = 0;
    intstrIndex[] = {0, -1};
    intmaxIndex = data.length()-1;

    for(int i=0; i<=maxIndex&&found<=index; i++){
        if(data.charAt(i)==separator || i==maxIndex){
            found++;
            strIndex[0] = strIndex[1]+1;
            strIndex[1] = (i == maxIndex) ? i+1 : i;
        }
    }
}

```

```

    }
}

```

Розглянемо частину методу «checkSpeedAlert», що відповідає за відправлення повідомлення про відсутність руху об'єкту протягом 15 хвилин. Використовуючи описані AT запити, ми маємо можливість перевіряти швидкість об'єкту, та зафіксувати час, коли швидкість впала нижче 3 кілометрів на годину. Зафіксувавши цей час падіння швидкості, ми можемо перевіряти цю швидкість протягом певного часу (15 хвилин), та в разі, якщо швидкість не підвищилась та не скинула зафіксований час протягом 15 хвилин, буде відправлено SMS повідомлення власнику, за допомогою методу «sendSMS» описаного вище.

```

void checkSpeedAlert(){
    vtg =sendATCommand( "AT+CGPSINF=64",1000,DEBUG);
    speedValue = splitString(vtg,',',5);
    speedNumberValue = speedValue.toDouble();
    zda =sendATCommand( "AT+CGPSINF=128",1000,DEBUG);
    timeValue = splitString(zda,',',1);
    timeNumberValue = timeValue.toDouble();
    if (speedNumberValue<= 3&&alertStatus == true &&timerValue
== 0) {
        timerValue = timeNumberValue + 1500;
    }
    if (speedNumberValue<= 3&&alertStatus == true &&
(timerValue - timeNumberValue) > - 10 && (timerValue -
timeNumberValue) < 10) {
        sendSMSAlert (phone,"Long motionless!");
        timerValue = 0;
    }
    if (speedNumberValue>3){
        timerValue = 0;
    }
}

```

SMS оповіщення власника пристрою, якщо перевищено швидкість пересування

Реалізація цього пункту пов'язана з пунктом «в», та є останньою частиною методу «checkSpeedAlert». В цій частині ми перевіряємо перевищення швидкості руху об'єкту понад 50 кілометрів на годину, та в

цьому випадку буде відправлено SMS повідомлення власнику, за допомогою методу «sendSMS».

```
if( speedNumberValue > 50 && alertStatus == true){
    timerValue = 0;
    sendSMSAlert (phone, "Speed limit exceeded!");
}
```

SMS оповіщення власника про низький рівень заряду акумулятора пристрою

Обраний SIM модуль підтримує спеціальний набір команд для контролю рівня заряду акумулятора. Перевірка рівня заряду акумулятора SIM модуля може бути реалізована AT командою «AT+CVC», що надає інформацію про <bc>– статус зарядки, <bc>– рівень заряду в процентах, <voltage>– напругу живлення.

Проте, завдяки описаним вище методам «splitString» та «sendSMSAlert», є можливість виділити процент рівня заряду пристрою з результату запиту та відправити повідомлення власнику з процентом заряду пристрою. Сукупність запитів для реалізації поставленої задачі скомпоновано в методі checkVoltage.

```
void checkVoltage(){
    voltage = sendATCommand("AT+CVC", DEBUG);
    voltageValue = splitString(voltage, ',', 1);
    voltageLevel = voltageValue.toInt();
    message = "Voltage = " + voltageValue;
    if(voltageLevel < 20){
        sendSMSAlert(phone, message);
    }
}
```

Можливість запиту на отримання координат пристрою

Визначення координат описано вище та реалізовано методом «getLocation». Проте відправлення координат власнику відбувається лише за запитом. Для цього додамо можливість обробляти сигнали тонального набору під час дзвінка на номер телефону SIM карти модуля. Ця можливість з'являється завдяки AT команді «AT+DDE=1», що вмикає DTMF формат.

Коли на модель надходить дзвінок модуль отримує сигнал «RING», в випадку отримання цього сигналу з номеру власника модуль відповідає на

дзвінок командою «ATA» та здатен обробляти дані тонального набору. Тому, якщо при дзвінку від власника буде введено цифру 5, буде відправлено SMS повідомлення власнику пристрою з його координатами. Описаний функціонал реалізовано методами «getDTMF» та «processingDTMF»

```

void getDTMF(){
    if (SIM800.available()) {
        _response = waitResponse();
        Serial.println(">" + _response);
        int index = -1;
        do {
            index = _response.indexOf("\r\n");
            String submsg = "";
            if (index > -1) {
                submsg = _response.substring(0, index);
                _response = _response.substring(index + 2);
            }
            else {
                submsg = _response;
                _response = "";
            }
            submsg.trim();
            if (submsg != "") {
                Serial.println("submessage: " + submsg);
                if (submsg.startsWith("+DTMF:")) {
                    String symbol = submsg.substring(7, 8);
                    processingDTMF(symbol);
                }
                else if (submsg.startsWith("RING")) {
                    sendATCommand("ATA", DEBUG);
                }
            }
        }
        while (index > -1);
    }
}

void processingDTMF(String symbol) {
    Serial.println("Key: " + symbol);
    dtmf = symbol.toInt();
    if(dtmf == 0){
        sendATCommand("ATH", DEBUG);
        phone = "";
        Serial.println("Clear EEPROM!");
    }
}

```

```

    if(dtmf == 2){
        alert();
        Serial.println("Sound Alert!");
    }
    if(dtmf == 3){
        alertStatus = !alertStatus;
        Serial.print("Change Alert Status = ");
        Serial.println(alertStatus);
    }
    if(dtmf == 5){
        getLocation();
    }
}

```

### Можливість вмикати та вимикати SMS оповіщення

Завдяки описаним вище методам «getDTMF» та «processingDTMF» є можливість змінювати значення статусу оповіщень обробкою сигналів тонального набору. Тобто якщо при дзвінку від власника буде введено цифру 3, буде оновлено значення змінної «alertStatus» на протилежне.

Для реалізації поставленої задачі необхідно створити змінну «alertStatus». При надходженні запиту про зміну статусу оповіщень буде змінено значення змінної, що відповідає за статус оповіщень, на протилежне.

### Можливість видалити телефон з пам'яті пристрою

Завдяки описаним вище методом обробки сигналів тонального набору «processingDTMF», додамо можливість видалити телефон з EEPROM пам'яті пристрою, але на відміну від налаштування повідомлень, після введення 0 в тональному наборі дзвінок буде автоматично закінчено AT командою «ATH», тому що номер має бути видалено, а приймання команд налаштування з нього буде припинено.

Можливість подзвонити на пристрій, щоб знайти його за звуковим сигналом

Завдяки наведеному вище методу обробки SMS повідомлень «parseSMS», додамо наступний пункт до налаштування приладу. При надходженні SMS з цифрою 2 з номеру власника, пристрій видаватиме звуковий сигнал, що допоможе власнику знайти пристрій.

Додатковою можливістю налаштування цих самих параметрів є обробка сигналів тонального набору «processingDTMF», що було описано вище. Тобто, при наборі цифри 2 в тональному наборі під час дзвінка, з телефону власника пристрою буде ввімкнено звуковий сигнал.

Згідно до всього вище описаного в Розділі 3 можна зробити підсумки, що завдяки набору AT команд для SIM модуля будується програмний функціонал пристрою відстеження рухомих об'єктів.

Використовуючи функції DTMF режиму отримуємо можливість налаштування наступних функцій:

- а) набравши цифру 2 буде подано звуковий сигнал з пристрою;
- б) набравши цифру 3 буде змінено статус екстреного оповіщення;
- в) набравши цифру 0 EEPROM пам'ять Arduino Nano буде очищено та дзвінок автоматично буде завершено.
- г) набравши цифру 5 буде надіслано повідомлення з координатами пристрою.

Використовуючи AT команди для обробки SMS повідомлень та методу аналізу тексту отримуємо можливість налаштування наступних функцій:

- а) повідомлення з цифрою 1 номер відправника буде збережено в EEPROM пам'ять Arduino Nano, як номер власника пристрою.

Використовуючи AT команди для отримання GPS даних отримуємо можливість:

- а) Визначати швидкість руху об'єкту;
- б) Час руху об'єкту;
- в) Керувати екстреними повідомленнями про стани руху об'єкту.

Завдяки AT командам відправлення SMS повідомлень маємо можливість відправляти повідомлення на збережений номер телефону власнику.

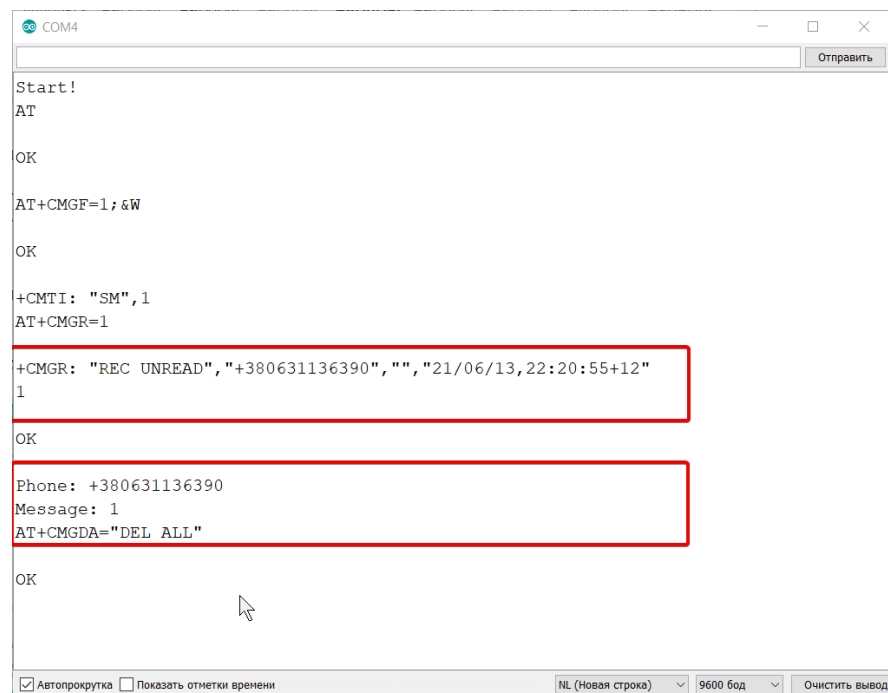
Тобто завдяки роботі з AT командами модулю SIM808 реалізовано всі сформульовані задачі проекту.

#### 4 РЕЗУЛЬТАТИ РОБОТИ ТА ІНСТРУКЦІЯ КОРИСТУВАЧА

Під час першого ввімкнення пристрою необхідно зареєструвати номер телефону власника приладу.

Реєстрація номеру телефону може відбуватися відправлення SMS повідомлення з цифрою 1 на номер телефону SIM карти модуля.

Під час надходження SMS повідомлення текст аналізується описаними вище методами та виділяється телефон та текст повідомлення в різні змінні, номер телефону додається в EEPROM пам'ять Arduino Nano (рис. 4.1).



```
COM4
Start!
AT
OK
AT+CMGF=1; &W
OK
+CMTI: "SM",1
AT+CMGR=1
+CMGR: "REC UNREAD", "+380631136390", "", "21/06/13,22:20:55+12"
1
OK
Phone: +380631136390
Message: 1
AT+CMGDA="DEL ALL"
OK
```

The screenshot shows a serial terminal window titled 'COM4'. The terminal displays the following text: 'Start!', 'AT', 'OK', 'AT+CMGF=1; &W', 'OK', '+CMTI: "SM",1', 'AT+CMGR=1', '+CMGR: "REC UNREAD", "+380631136390", "", "21/06/13,22:20:55+12"', '1', 'OK', 'Phone: +380631136390', 'Message: 1', 'AT+CMGDA="DEL ALL"', and 'OK'. Two red rectangular boxes highlight the '+CMGR' line and the 'Phone: +380631136390' line. The terminal window has a 'Отправить' button in the top right corner and a status bar at the bottom with options like 'Автопрокрутка', 'Показать отметки времени', 'NL (Новая строка)', '9600 бод', and 'Очистить вывод'.

Рисунок 4.1 – Реєстрація номеру телефона власника за SMS повідомленням

Кожним ввімкненням пристрою всі налаштування функцій екстрених повідомлень повертаються до автоматичних, тобто оповіщення про перевищення швидкості пересування, нерухомість об'єкту. Тому в разі потреби, користувач має можливість вимкнути повідомлення.

Для налаштування режимів повідомлень користувач має зателефонувати на номер телефону SIM карти модуля та ввести цифру 3 в режимі тонального набору з збереженого телефону власника (рис. 4.2).

```

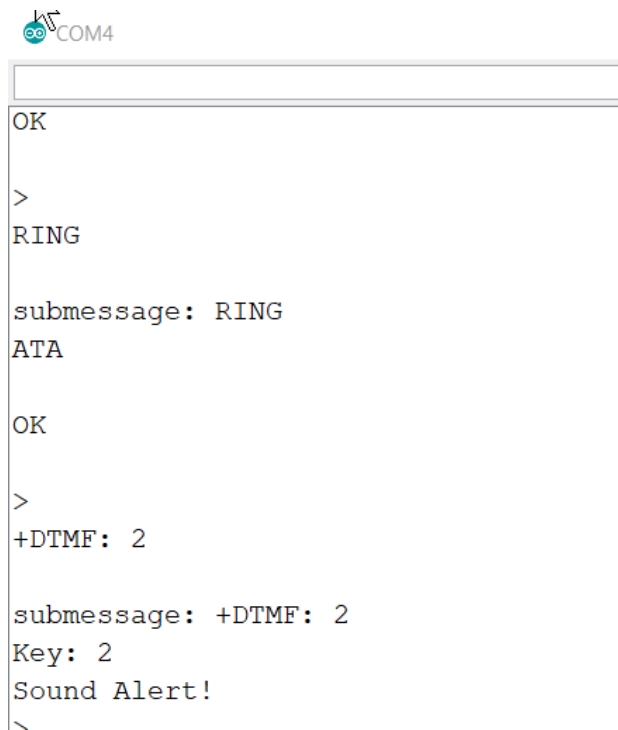
COM4
RING
submessage: RING
ATA
OK
>
+DTMF: 2
submessage: +DTMF: 2
Key: 2
Sound Alert!
>
+DTMF: 3
submessage: +DTMF: 3
Key: 3
Change Alert Status = 0
>
+DTMF: 3
submessage: +DTMF: 3
Key: 3
Change Alert Status = 1

```

Рисунок 4.2 – Налаштування екстрених повідомлень, за допомогою телефонного дзвінка

Також користувач має декілька функцій, що можуть бути ввімкнені користувачем. Наприклад, може виникнути ситуація, в якій необхідно подати звуковий сигнал з приладу для поліпшення задачі пошуку об'єкта.

Для посилення звукового сигналу з приладу користувач має зателефонувати на номер телефону SIM карти модуля та ввести цифру 2 в режимі тонального набору зі збереженого телефону власника (рис. 4.3).



```
COM4
OK
>
RING
submessage: RING
ATA
OK
>
+DTMF: 2
submessage: +DTMF: 2
Key: 2
Sound Alert!
\  

```

Рисунок 4.3 – Ввімкнення звукового сигналу на пристрої за телефонним дзвінком

Розглянемо інші функції пристрою, що вмикаються запитом від користувача. Однією з найважливіших функцій є можливість запиту на отримання координат пристрою, так само як і описаний вище функціях необхідно зателефонувати на номер телефону SIM карти модуля та ввести цифру 5 в режимі тонального набору з збереженого телефону власника (рис. 4.4).

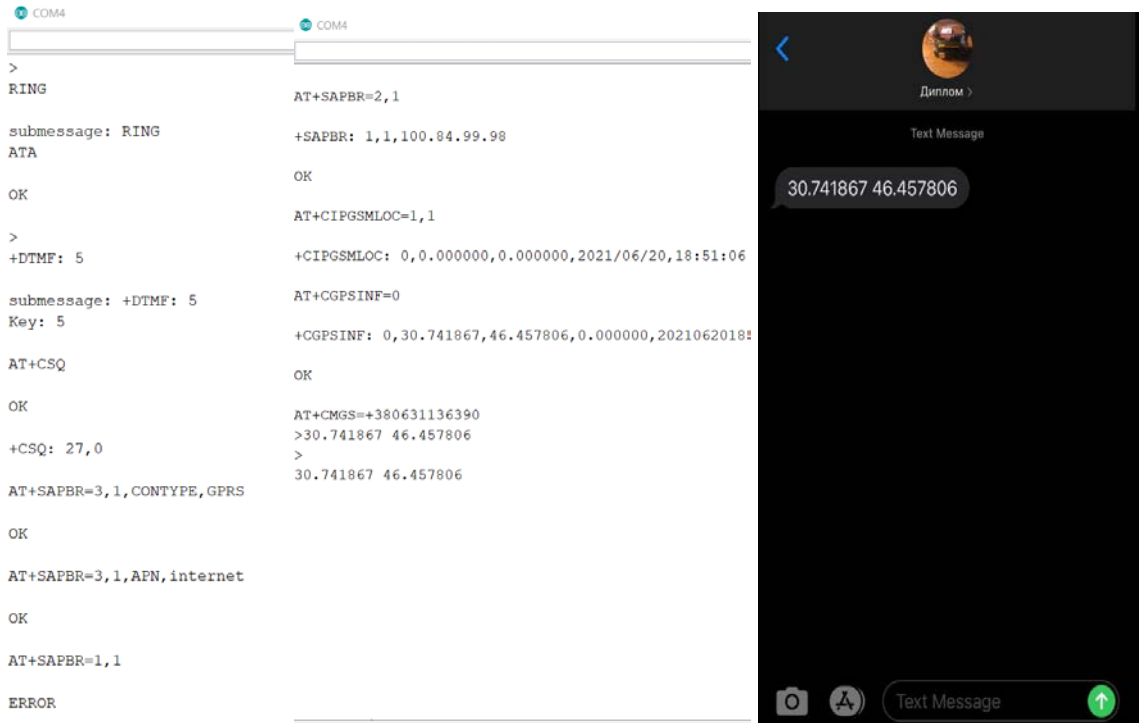
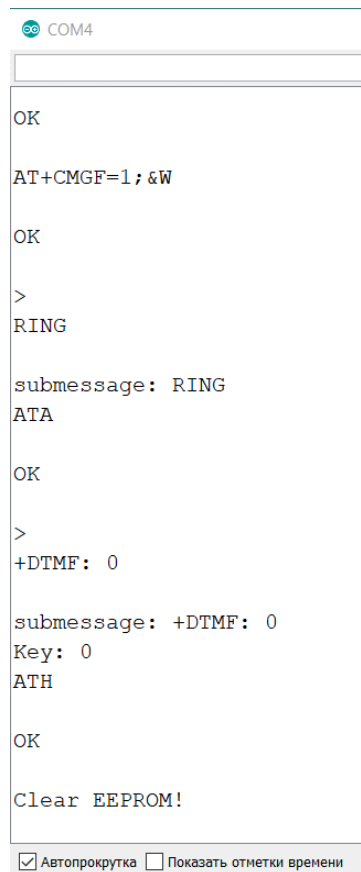


Рисунок 4.4 – Можливість запиту на отримання координат пристрою, за допомогою телефонного дзвінка

В разі потреби змінити номер телефону власника пристрою, що збережено в пам'яті, необхідно зателефонувати на номер телефону SIM карти модуля та ввести цифру 0 в режимі тонального набору з збереженого телефону власника (рис. 4.5).



```
COM4
OK
AT+CMGF=1; &W
OK
>
RING
submessage: RING
ATA
OK
>
+DTMF: 0
submessage: +DTMF: 0
Key: 0
ATH
OK
Clear EEPROM!
 Автопрокрутка  Показать отметки времени
```

Рисунок 4.5 – Видалення номеру телефона власника за телефонним дзвінком

Окрім функцій, що активуються користувачем система підтримує ряд автоматичних функцій, які будуть виконуватись в випадку, коли відправка SMS повідомлень дозволена власником приладом.

Однією з автоматичних функцій є SMS оповіщення власника про низький рівень заряду акумулятора пристрою відбувається, якщо рівень заряду акумулятора падає нижче 20 відсотків. В такому випадку власник отримає повідомлення з рівнем заряду (рис 4.6).

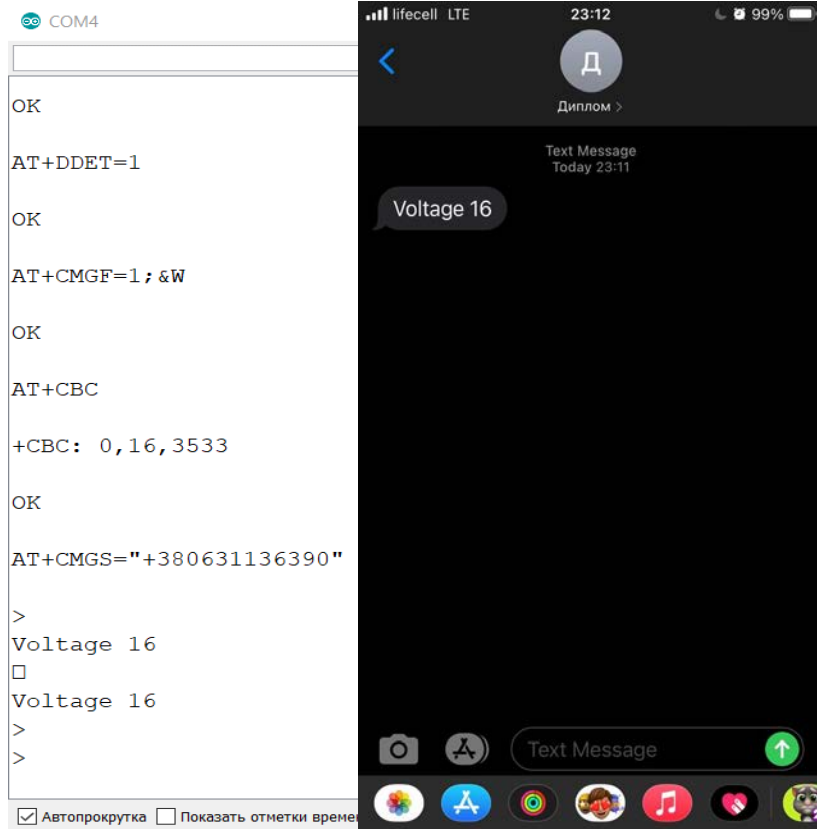


Рисунок 4.6 – SMS оповіщення власника про низький рівень заряду акумулятора пристрою

Наступною функцією є SMS оповіщення власника пристрою, якщо пристрій був без руху 15 хвилин. Для тестування функціоналу час на перевірку стану рухомості об'єкту скорочено до 2 хвилин (рис 4.7).

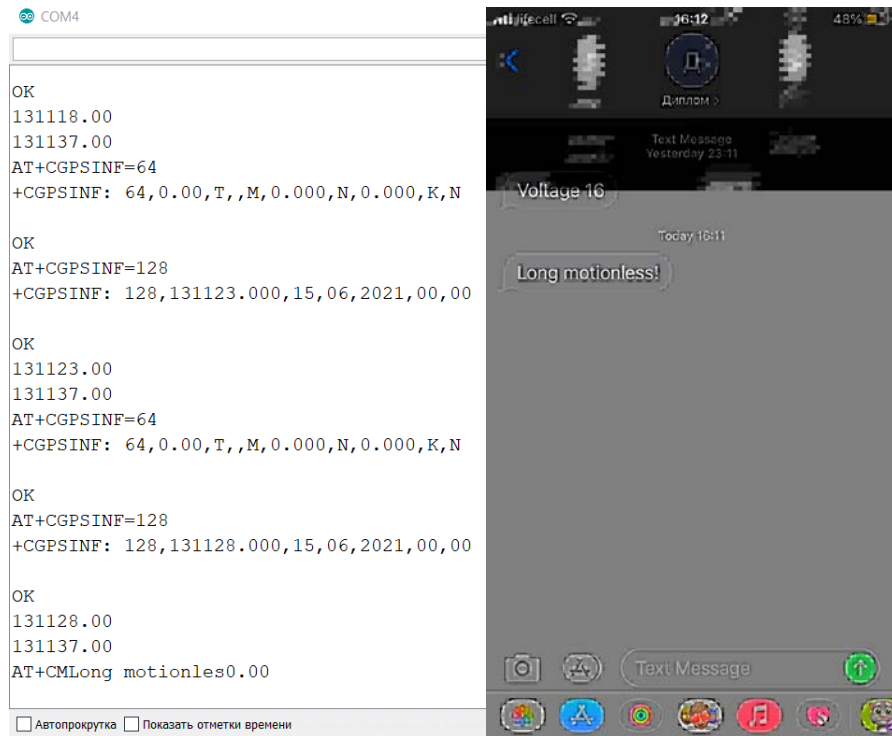


Рисунок 4.7 – SMS оповіщення власника пристрою, якщо пристрій був без руху

Останньою функцією є SMS оповіщення власника пристрою, якщо перевищено швидкість пересування. Для тестування функціоналу критерії перевірки швидкості змінено з швидкості більше 50 кілометрів на годину, до швидкості більше 10 кілометрів на годину (рис 4.8).

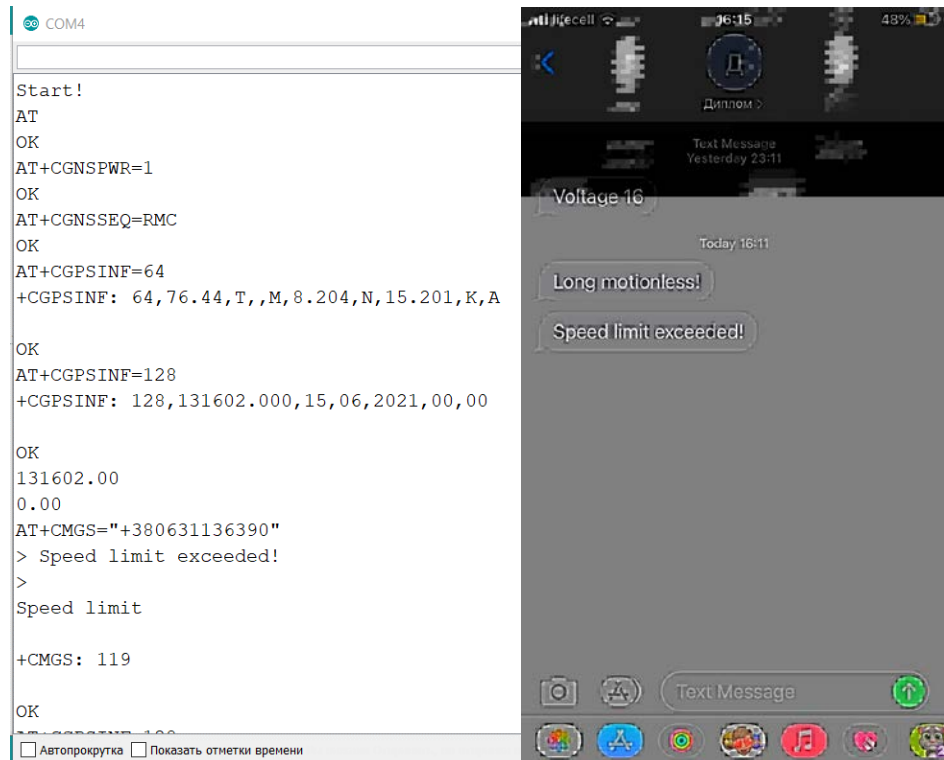


Рисунок 4.8 – SMS оповіщення власника пристрою, якщо перевищено швидкість пересування

Згідно з описаних вище функцій пристрій підтримує вмикання певних функцій за допомогою відправлення SMS повідомлень або телефонного дзвінка на номер SIM карти модулю, а саме запис номеру телефону власнику в пам'ять пристрою та видалення його з пам'яті; ввімкнення звукового сигналу на приладі; запит на отримання координат пристрою; вмикання та вимикання SMS оповіщень власника з приладу.

Окремо підтримується можливість SMS оповіщення про низький рівень заряду акумулятору, перевищення швидкості руху та перебування в нерухомому стані 15 хвилин. Такими чином перевірено функціонування всіх поставлених задач проекту.

В ході виконання кваліфікаційної роботи було досліджено предметну область, сформульовано перелік вимог до пристрою. Окремо було досліджено сучасні мікроконтролери платформи, на основі чого було обрано елементну базу для створення прототипу. В якості бази для прототипування було обрано мікроконтролеру платформу Arduino (Arduino Nano), для відстеження місцеположення та забезпечення зв'язку застосовано модуль на чіпі SIM808 від компанії SIMCOM, що поєднує в собі блоки GPRS, GSM і GPS. Для забезпечення функціонування пристрою було розроблено керуючий код, з застосуванням середовища програмування Arduino IDE.

Створений прототип дозволяє визначати координати місцеположення та швидкість руху пристрою, та передавати користувачу необхідні дані за запитом останнього, або при виконанні наперед заданих умов. Таких наприклад, як перевищення швидкості руху або його відсутністю протягом обумовленого часу та розряді батареї. Також є можливість дистанційно вмикати на пристрої звуковий сигнал для спрощення пошуку об'єкту.

Особливістю розробленого пристрою є те, що для визначення місцеположення об'єкту використовується дві технології GPRS та GPS. За замовченням при наявності та достатній потужності сигналу мобільного оператора використовується GPRS, що дозволяє більш точно визначити координати. В разі неможливості отримання мобільного сигналу, відбувається переключення на GPS навігацію.

Для більш зручного використання та розширення функціональних можливостей бажано створити програмне забезпечення з зручним інтерфейсом користувача, що забезпечить відстеження координат місцезнаходження пристрою на карті в режимі реального часу, з можливістю налаштування оповіщень про зміну стану пересування пристрою та можливістю створення соціальної мережі, для поєднання дружніх пристроїв на карті та обміну інформацією між їх власниками.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. GPS трекер FMB140; GPS мониторинг транспорта: [Электронный ресурс]. – Режим доступа: <https://fixon.com.ua/shop/avtomobilnyiy-gps-2/gps-treker-fmb140/>.
2. GPS маячок для слежения; GPS мониторинг транспорта: [Электронный ресурс]. – Режим доступа: <https://fixon.com.ua/gps-zhuchok-dlya-slezheniya/>.
3. GPS трекер метка скрытая в фонаре для велосипеда TK906 TKSTAR: [Электронный ресурс]. – Режим доступа: [http://vizir-company.com/index.php?route=product/product&product\\_id=92](http://vizir-company.com/index.php?route=product/product&product_id=92).
4. GPS маячок для велосипеда; GPS мониторинг транспорта: [Электронный ресурс]. – Режим доступа: <https://fixon.com.ua/gps-mayachok-dlya-velosipeda/>.
5. GPS трекер для велосипеда. Описание, характеристики: [Электронный ресурс]. – Режим доступа: <https://wazza.com.ua/products/gps-treker-dlya-velosipeda/>.
6. Arduino Nano 3.0 – распиновка, схема, подключение: [Электронный ресурс]. – Режим доступа: [https://arduinomaster.ru/platy-arduino/plata-arduino-nano/#\\_Arduino\\_Nano-2](https://arduinomaster.ru/platy-arduino/plata-arduino-nano/#_Arduino_Nano-2).
7. Crowtail GPRS GSM GPS модуль на SIM808 V1.1 від Elecrow: [Электронный ресурс]. – Режим доступа: <https://arduino.ua/prod1586-crowtail-gprs-gsm-gps-modyl-na-SIM808-v1-1-ot-elecrow>.
8. SIM808, Четырёхдиапазонный модуль GSM/GPRS с функцией спутниковой GPS навигации: [Электронный ресурс]. – Режим доступа: [http://catalog.gaw.ru/index.php?page=component\\_detail&id=80828](http://catalog.gaw.ru/index.php?page=component_detail&id=80828).

## Програмний код

```
#include <SoftwareSerial.h>
#include <EEPROM.h>
#define DEBUG true

SoftwareSerial SIM808(8, 9);
int buzzerPin = 2;

String gll = "";

String vtg = "";
double speedNumberValue = 0;
String speedValue = "";

String zda = "";
String timeValue = "";
double timeNumberValue = 0;
double timerValue = 0;

String voltage = "";
String voltageValue = "";
int voltageLevel = 1;

String _response = "";
String phone = "";
int eepromAddress = 0;
bool alertStatus = true;
int dtmf = 0;
String message = "";

String csq = "";
int csqNumber = 0;

String location = "";
String longitude = "";
String latitude = "";

double long = 0;
double lat = 0;

void setup() {
    Serial.begin(9600);
    SIM808.begin(9600);
```

```

Serial.println("Start!");
sendATCommand("AT", DEBUG);
_response = sendATCommand("AT+DDET=1", DEBUG);
_response = sendATCommand("AT+CMGF=1;&W", DEBUG);
pinMode(buzzerPin, OUTPUT);
getgps();
}

void loop()
{
  getDTMF();
  checkSpeedAlert();
  if (SIM808.available()) {
    _response = waitResponse();
    _response.trim();
    Serial.println(_response);
    //....
    if (_response.startsWith("+CMTI:")) {
      int index = _response.lastIndexOf(",");
      String result = _response.substring(index + 1,
_response.length());
      result.trim();
      _response=sendATCommand("AT+CMGR="+result, true);
parseSMS(_response);
      sendATCommand("AT+CMGDA=\"DEL ALL\"", true);
    }
  }
  if (Serial.available()) {
    SIM808.write(Serial.read());
  };
}

void parseSMS(String msg) {
  String msgheader = "";
  String msgbody = "";
  String msgphone = "";
  msg = msg.substring(msg.indexOf("+CMGR: "));
  msgheader = msg.substring(0, msg.indexOf("\r"));
  msgbody = msg.substring(msgheader.length() + 2);
  msgbody = msgbody.substring(0, msgbody.lastIndexOf("OK"));
  msgbody.trim();
  int firstIndex = msgheader.indexOf("\",\"") + 3;
  int secondIndex = msgheader.indexOf("\",\"", firstIndex);
  msgphone = msgheader.substring(firstIndex, secondIndex);
  Serial.println("Phone: "+msgphone);
  Serial.println("Message: "+msgbody);
}

```

```

if (msgbody.indexOf("1") > -1) {
    EEPROM.put(eepromAddress, msgphone);
}

void getLocation () {
    csq = sendATCommand("AT+CSQ", DEBUG);
    csq = splitString(csq, ',', 0);
    csqNumber = csq.toInt();
    sendATCommand("AT+SAPBR=3,1,CONTTYPE,GPRS", DEBUG);
    sendATCommand("AT+SAPBR=3,1,APN,internet", DEBUG);
    sendATCommand("AT+SAPBR=1,1", DEBUG);
    sendATCommand("AT+SAPBR=2,1", DEBUG);
    location = sendATCommand("AT+CIPGSMLOC=1,1", DEBUG);
    longitude = splitString(location, ',', 1);
    long = longitude.toDouble();
    latitude = splitString(location, ',', 2);
    lat = latitude.toDouble();
    if( csqNumber < 10 && (long != 0 && lat != 0)) {
        message = longitude + " " + latitude;
        sendSMSAlert(phone, message);
    }
    if( csqNumber > 9 || (long == 0 && lat == 0)) {
        location = sendATCommand("AT+CGPSINF=0", DEBUG);
        longitude = splitString(location, ',', 0);
        latitude = splitString(location, ',', 1);
        message = longitude + " " + latitude;
        sendSMSAlert(phone, message);
    }
}

void sendSMS(String phone, String messageText)
{
    sendATCommand("AT+CMGS=\"" + phone + "\"", DEBUG);
    sendATCommand(messageText + "\r\n" + (String)((char)26),
DEBUG);
}

void sendSMSAlert(String phone, String message) {
    sendSMS(phone, message);
    if (SIM800.available()) {
        _response = waitResponse();
        _response.trim();
        Serial.println(_response);
        if (_response.startsWith("+CMGS:")) {
            int index = _response.lastIndexOf("\r\n");
            String result = _response.substring(index + 2,
_response.length());
            result.trim();
            if (result == "OK") {

```



```

timerValue = 0;
    }
}

void checkVoltage(){
    voltage = sendATCommand("AT+CBC", DEBUG);
    voltageValue = splitString(voltage,',',1);
    voltageLevel = voltageValue.toInt();
    message = "Voltage = " + voltageValue;
    if(voltageLevel < 20){
        sendSMSAlert(phone, message);
    }
}

void getDTMF(){
    if (SIM800.available()) {
        _response = waitResponse();
        Serial.println(">" + _response);
        int index = -1;
        do {
            index = _response.indexOf("\r\n");
            String submsg = "";
            if (index > -1) {
                submsg = _response.substring(0, index);
                _response = _response.substring(index + 2);
            }
            else {
                submsg = _response;
                _response = "";
            }
            submsg.trim();
            if (submsg != "") {
                Serial.println("submessage: " + submsg);
                if (submsg.startsWith("+DTMF:")) {
                    String symbol = submsg.substring(7, 8);
                    processingDTMF(symbol);
                }
                else if (submsg.startsWith("RING")) {
                    sendATCommand("ATA", DEBUG);
                }
            }
        }
        while (index > -1);
    }
}

void processingDTMF(String symbol) {

```

```
Serial.println("Key: " + symbol);
dtmf = symbol.toInt();
if(dtmf == 0){
    sendATCommand("ATH", DEBUG);
    phone = "";
    Serial.println("Clear EEPROM!");
}
if(dtmf == 2){
    alert();
    Serial.println("Sound Alert!");
}
if(dtmf == 3){
    alertStatus = !alertStatus;
    Serial.print("Change Alert Status = ");
    Serial.println(alertStatus);
}
if(dtmf == 5){
    getLocation();
}
}

void alert(){
    tone(buzzerPin, 600);
    delay(100);
    noTone(buzzerPin);
    delay(500);
    tone(buzzerPin, 200);
    delay(100);
    noTone(buzzerPin);
    delay(900);
}
```