

Тема: «Портативний пристрій контролю положення у просторі з
можливістю запису інформації у часі»

Анотація

Наукова робота складається з 26 сторінок, 1 таблиці, 17 рисунків та 6 літературних джерел.

Об'єкт дослідження в роботі: пристрій контролю положення об'єкту у просторі з можливістю реєстрації даних, спроектований на базі мікроконтролера Arduino Nano.

Мета дослідження: використання інерційного вимірюючого модуля для визначення положення у просторі.

У основній частині було спроектовано та розроблено прототип портативного мікропроцесорного пристрою контролю положення об'єкта в просторі з можливістю запису інформації з прив'язкою до часу. Обрано апаратні компоненти проєкту: мікроконтролер Arduino Nano V3 , плата датчиків Pololu ALTIMU-10v5 (гіроскоп, акселерометр, магнітометр, датчик тиску) , модуль реєстратора часу Real Time Clock на DS3231SN , модуль реєстрації даних на SD-карту AOC431 , модуль відображення інформації LCD 2004 та автономне джерело живлення на акумуляторах 18650 з модулем зарядки AOC708. Обґрунтовано вибір апаратних компонентів пристрою, описано підключення обладнання та процедури опитування датчиків з відправкою інформації , створено програмний код в середовищі Arduino IDE та проведено аналіз розробленої системи.

АКСЕЛЕРОМЕТР, ГІРОСКОП, МАГНІТОМЕТР, ARDUINO NANO, POLOLU, КОНТРОЛЬ ПОЛОЖЕННЯ, ЗАПИС ІНФОРМАЦІЇ, ДАТЧИК, МІКРОКОНТРОЛЕР

ЗМІСТ

ВСТУП.....	4
1. РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ ТА ПРОГРАМНОЇ ПРИСТРОЮ....	5
1.1. Компоненти пристрою.....	5
1.2. Вибір та обґрунтування елементної бази.....	6
1.2.1. Вибір мікроконтролера.....	6
1.2.2. Вибір датчиків.....	10
1.2.3. Вибір реєстратора часу.....	12
1.2.4. Вибір модулю реєстрації даних на SD-карту.....	12
1.2.5. Вибір модулю відображення інформації.....	14
1.2.6. Вибір автономного джерела живлення.....	15
1.3. Розробка програмно-апаратного забезпечення.....	17
1.4. Перевірка функціонування пристрою.....	22
ВИСНОВКИ.....	24
ПЕРЕЛІК ПОСИЛАНЬ.....	26

ВСТУП

Потреба у пристроях контролю положення в просторі зростає, зокрема у зв'язку з новими законами ЄС про обов'язкове встановлення "чорних скриньок" у автомобілі з липня 2024 року. Враховуючи прагнення України до європейських стандартів, подібні вимоги можуть з'явитися і в українському законодавстві. У цьому контексті, портативний пристрій контролю положення з можливістю запису даних, що працює за принципом "чорної скриньки", є актуальним і може стати незамінним завдяки своїй низькій вартості та здатності фіксувати показники акселерометра, гіроскопа, магнітометра та датчика тиску на карту MicroSD.

Дана робота представляє концепцію та розробку такого пристрою, здатного автоматично визначати положення об'єкта по трьох координатних осях, що дозволяє точно відстежувати його місцезнаходження у певний час, наприклад, під час позаштатних ситуацій, та надавати інформацію для запобігання проблем у майбутньому. Вся інформація з датчиків може бути записана на карту пам'яті за допомогою спеціального модуля.

Робота охоплює налаштування пристрою, огляд сфер його застосування, детальний опис датчиків, процес проектування та розробки (включаючи вибір компонентів, програмну реалізацію та тестування), а також результати експериментів та аналіз ефективності розробленого пристрою. Метою роботи є створення та дослідження прототипу портативного пристрою контролю положення в просторі, який може адаптуватися до різних умов та ефективно визначати положення об'єкта в будь-який час. Актуальність роботи зумовлена масовим використанням подібних пристроїв, як "чорні скриньки".

РОЗДІЛ 1. РОЗРОБКА АПАРАТНОЇ ЧАСТИНИ ТА ПРОГРАМНОЇ ПРИБРОЮ

1.1 Компоненти пристрою

Спираючись на виконання заданих функцій пропонується наступна структура системи (рис. 1.1), де складовими є:

1. Мікроконтролер Arduino Nano V3;
2. Датчики гіроскоп, акселерометр, магнітометр, та датчик тиску об'єднані на платі Pololu ALTIMU-10v5;
3. Модуль реєстратору часу Real Time Clock на DS3231SN;
4. Модуль реєстрації даних на SD-карту AOC431;
5. Модуль відображення інформації LCD 2004;
6. Автономне джерело живлення на 18650

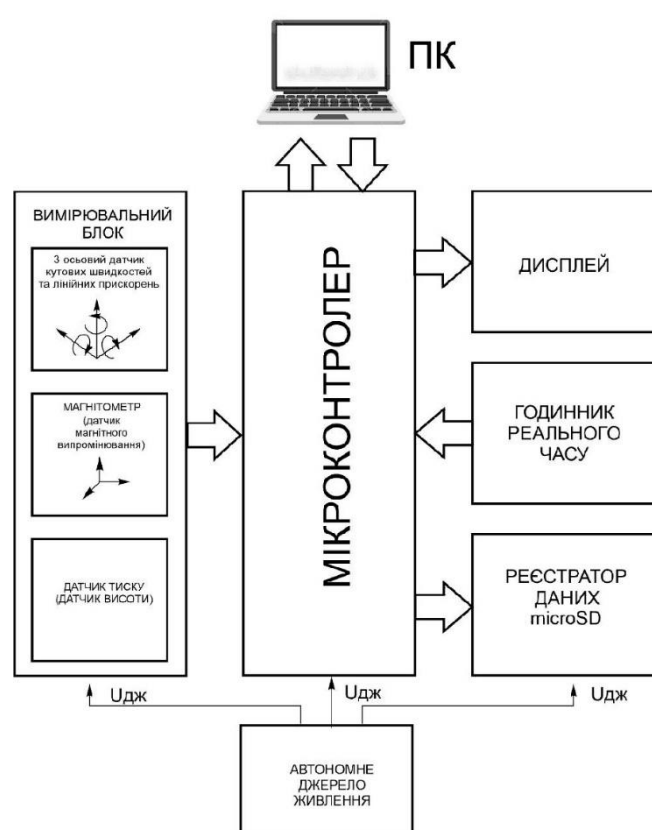


Рисунок 1.1 — Структурна схема пристрою

Вибір датчиків акселерометру, гіроскопу, магнітометру та датчику тиску обґрунтовано з метою максимально точного визначення положення об'єкту у просторі. Інформація з них надходить на мікроконтроллер Arduino, який є мозком пристрою, тобто управляє процесом. Обрано було мікроконтроллер саме Arduino, через його більш низьку вартість та простоту, на відміну від інших мікроконтролерів. Після потраплення інформації з датчиків у мікроконтроллер, одна частина інформації йде на пристрій індикації, а інша частина - на модуль реєстрації даних, котрий був обран задля можливості збереження інформації о положенні. Пристрій живлення - це акумуляторна батарея, що забезпечує портативність пристрою.

1.2 Вибір та обґрунтування елементної бази

1.2.1 Вибір мікроконтроллера

Вибір мікроконтроллера є дуже важливим, тому вибору цього компоненту пристрою слід приділяти максимальну увагу.

Arduino Nano v3 (рис 1.2) - це мініатюрна платформа для розробки, яка базується на мікроконтролері ATmega328P і використовується для створення різноманітних електронних пристроїв та прототипів.

У своїй суті, Arduino Nano - це універсальна платформа для розробки різноманітних електронних проектів. Вона широко використовується як аматорами, так і професіоналами. Дана платформа ідеально підходить для вивчення основ електроніки та програмування мікроконтролерів, використовується для створення автоматизації вдома, таких як управління освітленням, системами безпеки, клімат-контролем застосовується у створенні роботів, дронів та інших автономних пристроїв, використовується для підключення різних пристроїв до інтернету та взаємодії з ними, використовується у світлових шоу, інтерактивних виставках та інших творчих проектах і це перше що приходить на розум.



Рисунок 1.2 — Зовнішній вигляд плати Arduino Nano V3

Arduino Nano v3 має дуже компактний дизайн, що робить його ідеальним для вбудованих проектів, де обмежений простір. Плата має вбудований USB-порт, який дозволяє легко підключити плату до будь-якого пристрою, без проблем програмувати мікроконтролер і взаємодіяти з ним через USB-з'єднання. Також через USB іде живлення пристрою на базі Arduino. Плата має функцію автоматичного вибору живлення, яка дозволяє автоматично перемикає між внутрішнім живленням від USB та зовнішнім джерелом живлення в залежності від наявності живлення через USB.

Arduino Nano v3 використовує мікроконтролер ATmega328P, який має достатньо ресурсів для виконання різноманітних завдань. Помірний обсяг флеш-пам'яті (32 кБ) та ОЗП (2 кБ) дозволяє розробникам створювати складніші програми та зберігати більше даних. Arduino Nano v3 має всі необхідні периферійні пристрої, включаючи цифрові та аналогові входи/виходи, PWM виходи, UART, I2C та SPI інтерфейси. Наявність вбудованого підтримки протоколу USB-Serial дозволяє Arduino Nano v3 працювати як віртуальний COM-порт, що спрощує взаємодію з комп'ютером. Вбудований захист від перенапруги дозволяє захистити мікроконтролер від пошкоджень при введенні високих напруг або збоїв. Принципова схема Arduino Nano v3 зображено на рисунку 1.3.

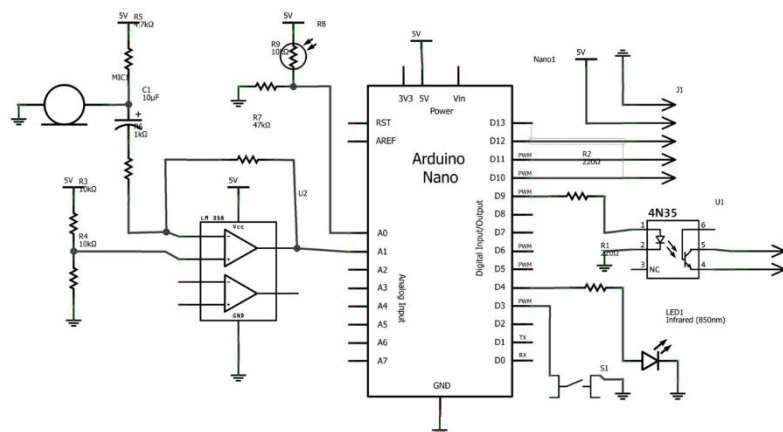


Рисунок 1.3 — Принципова схема Arduino Nano V3

Мікроконтроллер має вбудований DC-DC конвертер, який дозволяє використовувати широкий діапазон вхідних напруг (від 7 до 12 вольт) безпосередньо від джерела живлення, що робить його більш гнучким для різних умов роботи. Arduino Nano v3 може бути легко розширена за допомогою шилдів (shield), що дозволяють підключати додаткові модулі та пристрої, такі як датчики, екрани, мотори та інше. Arduino Nano v3 сумісний з великою кількістю розширень (shields), що робить його дуже універсальним та розширюваним, та має велику спільноту користувачів та підтримку ком'юніті.

Користуватись Arduino Nano до неможливості легко. Спочатку потрібно встановити на свій комп'ютер під управлінням операційної системи Windows, Mac OS або будь-якого дистрибутиву Linux програмне забезпечення Arduino IDE (рис 1.4). Далі треба підключити Arduino Nano до комп'ютера за допомогою USB-кабелю. Далі потрібно відкрити програму Arduino IDE та вибрати потрібну плату, натиснувши Tools > Board > Arduino Nano. Далі необхідно обрати потрібний порт Tools > Port > [ваш порт].

Далі все що залишиться — написати простий скетч на мові програмування C/C++. Після цього потрібно натиснути кнопку завантаження (іконка зі стрілкою праворуч) у верхньому лівому куті Arduino IDE. Програма завантажиться на плату і почне виконуватись. Це і є базове розуміння роботи з Ардуіно.

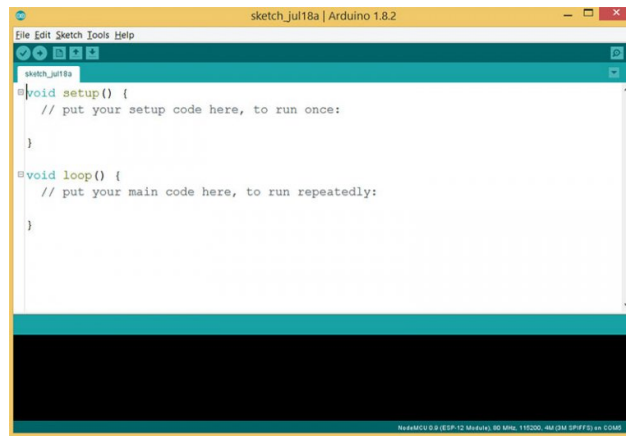


Рисунок 1.4 — Інтерфейс Arduino IDE

Arduino Nano v3 доступний великою кількістю виробників та може бути придбаний за доступну ціну, що робить його дуже привабливим для різних застосувань та проектів.

Таблиця 1.1 — Специфікації Arduino Nano v3

Мікроконтроллер	ATmega328
Напруга живлення	5В через USB або 7-12В через VIN
Цифрові входи/виходи	14 (з яких 6 можуть використовуватись як ШІМ-виходи)
Аналогові входи	8
Флеш-пам'ять	32 кб (2 кб використовуються для завантажувача)
ОЗП	2 кб
EEPROM	1 кб
Тактова частота	16 МГц
Розмір	18 мм x 45 мм
Вага	12 г

Вибір пав саме на Arduino Nano через низьку вартість даного мікроконтролеру та через свою популярність, велике ком'юніті та зручне середовище розробки

1.2.2 Вибір датчиків

Основними датчиками у пристрої є акселерометр, гіроскоп та магнітометр, але підключати кожен окремо не має практичного сенсу, тому що є варіанти плат, підключаємих до Arduino, які містять у собі усі ці 3 датчики одночас. Провівши аналіз ринку було прийнято рішення зупунитись саме на Pololu ALTIMU-10v5

Pololu ALTIMU-10v5 - це компактна (25,4 x 12,7 мм) плата. Pololu AltIMU-10 є інерційним вимірювальним пристроєм, що містить мікросхеми: ST LPS25 барометр, ST LSM6DS33 3-осьовий гіроскоп і 3-осьовий акселерометр і ST LIS3MDL 3-осьовий магнітний блок IMU та висотомір. Ці датчики є інтегральними схемами, але маленький корпус робить їх використання скрутним, особливо для навчання та хобі. Вони також працюють від напруги нижче 3,6 В, що ускладнює їх підключення до мікроконтролерів працюючих від 5 В. Ці проблеми вирішені в цій платі Pololu шляхом додавання додаткових електронних компонентів, у тому числі 3,3 В стабілізатора напруги та схеми зміщення логічного рівня, зберігаючи при цьому компактний розмір пристрою. На платі повністю встановлені компоненти SMD і в тому числі LPS25, LSM6DS33 і LIS3MDL, як це показано на фотографії продукту (рис. 1.5).



Рисунок 1.5 - Pololu ALTIMU-10v5

У LPS25, LSM6DS33 і LIS3MDL є багато параметрів, що налаштовуються, включаючи динамічний вибір чутливості для барометра, гіроскопа, акселерометра і магнітометра, а також вибір вихідної швидкості передачі даних для кожного датчика. Ці три мікросхеми можуть бути доступні через загальний I2C/TWI інтерфейс, що дозволяє окремо зчитувати дані всіх чотирьох датчиків через одну шину синхронізації і одну шину даних.

Крім того, контакт конфігурації адреси введеного пристрою дозволяє користувачам змінювати I2C адреси датчиків і отримати два Pololu AltIMU-10 підключеними до однієї шини I2C.

Плата включає стабілізатор з низьким падінням напруги, що забезпечує 3,3 В необхідні для LPS25, LSM6DS33 і LIS3MDL і дозволяє підключати датчик до зовнішнього джерела живлення 2,5 - 5,5 В.

Для підключення плати Pololu AltIMU-10 (рис. 1.6) необхідно 4 контакти: VIN (живлення), GND (земля), SCL та SDA (висновки датчиків). VIN необхідно з'єднати з джерелом живлення 2,5 - 5,5, GND до "землі", а SCL і SDA - через шину I2C (крім того, якщо використовується плата з живленням в 3,3 В, можна залишити роз'єм VIN вимкненим і обійти вбудований стабілізатор, підключивши 3,3 В безпосередньо до VDD).

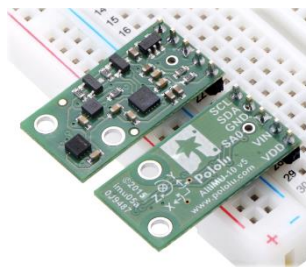


Рисунок 1.6 — Підключення Pololu ALTIMU-10v5

Плату Pololu AltIMU-10 v5 було обрано завдяки тому що в ній використовуються нові компоненти LIS3MDL 3-осьовий магнітометр та LSM6DS33 3-осьовий акселерометр і гіроскоп, що забезпечують покращену

продуктивність порівняно з попередніми версіями та пропонує ці переваги за меншою ціною.

1.2.3 Вибір реєстратора часу

Модуль реального часу (RTC) на основі DS3231SN (рис 2.7) це точний модуль годинника, який використовується для відстеження часу і дати в різних електронних проектах. Цей модуль є популярним вибором завдяки своїй високій точності та зручності використання. Модуль має вбудований кварцовий генератор з температурною компенсацією (TCXO), що забезпечує високу точність (+2 хвилини на рік). Напруга від 2.3В до 5.5В, що робить його сумісним з різними мікроконтролерами, включаючи Arduino. Інтерфейсом слугує I2C (TWI), що дозволяє легко підключити модуль до мікроконтролерів. Працює при температурах від 40°C до +85°C. Даний модуль використовує резервну батарею CR2032, яка забезпечує роботу модуля в разі відсутності основного живлення.



Рисунок 1.7 — Модуль Real Time Clock на DS3231SN

Даний реєстратор часу було обрано через його високу точність, зручний інтерфейс I2C та низьке енергоспоживання.

1.2.4 Вибір модулю реєстрації даних на SD-карту

Запис даних з датчиків у портативному пристрої контролю положення та реєстрації об'єкту у просторі з можливістю запису інформації у часу вирішено робити на SD-карту, тому потрібно обрати модуль реєстрації даних на SD-карту. Модуль microSD карти AOC431 (рис. 1.8) використовується для зберігання великих об'ємів даних, таких як текстові файли, зображення або будь-які інші дані, які потребують надійного зберігання. Цей модуль дозволяє

легко додати функціонал збереження даних до будь-яких проектів на базі мікроконтролерів, таких як Arduino.



Рисунок 1.8 — Модуль microSD карти AOC431

Модуль підтримує карти microSD та microSDHC. Напруга живлення 3.3В при необхідності, можна жити від 5В через вбудований регулятор напруги. Інтерфейс SPI (Serial Peripheral Interface), що забезпечує високу швидкість передачі даних. Даний модуль навмисне вибрано з іншим інтерфейсом, для того, щоб не перенавантажувати канал передачі даних по I2C. Пристрій має компактні розміри, що дозволяє легко інтегрувати модуль у проекти з обмеженим простором.

Перевагами пристрою є:

- Легкість використання. Завдяки стандартному інтерфейсу SPI модуль легко підключається до мікроконтролерів.
- Зберігання великих об'ємів даних. MicroSD карти можуть зберігати значні обсяги даних, що корисно для логерів даних та мультимедійних проектів.
- Надійність. Використання microSD карт забезпечує надійне і довготривале зберігання даних.

Для роботи з модулем microSD карти найчастіше використовується бібліотека SD, яка входить в стандартний набір Arduino IDE. Модуль microSD карти AOC431 був обраний через те що він є потужним і зручним інструментом для додавання функцій зберігання даних до будь-яких проектів на базі мікроконтролерів.

Його використовують для зберігання даних з сенсорів для подальшого аналізу, збереження зображень, аудіофайлів або відеофайлів, та збереження налаштувань та конфігураційних файлів для проектів.

2.2.5 Вибір модулю відображення інформації

Важливим етапом у розробці пристрою контролю положення та реєстрації об'єкту у просторі з можливістю запису інформації у часу, є вибір модулю відображення інформації.

Символьний дисплей LCD 2004 (рис 1.9) є популярним вибором для багатьох електронних проектів завдяки своїй здатності відображати значну кількість інформації одночасно. Цей дисплей може показувати до 20 символів на кожному з чотирьох рядків, що робить його ідеальним для проектів, які потребують відображення великої кількості текстової інформації.



Рисунок 1.9 — Модуль екрану LCD 2004

Роздільна здатність екрану - 20 символів на 4 рядки. У екрана синя підсвітка з білими символами. Напруга живлення складає 5В. Працює по паралельний інтерфейс (HD44780) або I²C (при використанні додаткового модуля). LCD 2004 має велику кількість символів, а саме відображає 80 символів одночасно. Екран легко інтегрується в проекти на базі Arduino та інших мікроконтролерів. Підсвітка забезпечує гарну видимість в умовах низького освітлення.

Символьний дисплей LCD 2004 було обрано тому що це потужний і зручний інструмент для додавання функцій відображення інформації. Його основною перевагою є невелика потреба в пам'яті. Екран має можливість відображення інтерфейсів користувача для налаштування параметрів, відображення поточної інформації про стан системи або пристрою, виведення поточних значень вимірювань сенсорів, відображення часу та таймерів та ін.

2.2.6 Вибір автономного джерела живлення

Для того, щоб зробити пристрій автономним, потрібно зробити акумуляторні батареї джерелом живлення пристрою.

Акумулятори 18650 — це циліндричні літій-іонні акумулятори, які отримали свою назву від своїх розмірів: 18 мм в діаметрі і 65 мм в довжину (рис 2.10). Вони стали стандартом у багатьох галузях завдяки своїй високій ємності, тривалому терміну служби та здатності забезпечувати високий струм розряду. Основною перевагою цих акумуляторів є висока щільність енергії, що робить їх ідеальними для застосування в портативних електронних пристроях, електроінструментах, електровелосипедах та навіть в електромобілях.



Рисунок 1.10 — Зовнішній вигляд акумулятору 18650

Однією з головних характеристик акумуляторів 18650 є їх ємність, яка може варіюватися від 1800 мА·год до понад 3500 мА·год. Термін служби акумуляторів 18650 зазвичай становить кілька сотень циклів зарядки-розрядки, що робить їх економічно вигідними в довгостроковій перспективі. Однак, їхній реальний термін служби залежить від умов експлуатації, зокрема від режиму зарядки та розрядки, температурних умов та частоти використання.

Акумулятори 18650 також відомі своєю стабільністю та безпекою. Багато моделей оснащені вбудованими системами захисту, які запобігають перезарядженню, перерозряду та короткому замиканню. Ще однією важливою особливістю акумуляторів 18650 є їх низький рівень саморозряду. Це означає, що вони зберігають заряд протягом тривалого часу, навіть якщо не використовуються, що робить їх зручними для резервного живлення та аварійних ситуацій. Крім того, ці акумулятори мають високу ефективність зарядки, що дозволяє швидко відновлювати їхню ємність.

Акумулятори 18650 широко використовуються у багатьох галузях завдяки своїм універсальним характеристикам та надійності. Вони стали невід'ємною частиною сучасних технологій, забезпечуючи стабільне та тривале живлення для численних пристроїв. З розвитком технологій ці акумулятори продовжують вдосконалюватися, пропонуючи ще більшу ємність, безпеку та ефективність.

Даний тип акумуляторів було обрано через її невелику ціну, велику ємність та практичність.

Окрім самих акумуляторів, ще треба обрати модуль їх живлення та зарядки.

Модуль живлення та зарядки 18650 microUSB/Type-C 5B/3A AOC708 (рис. 2.11) призначений для зарядки та живлення літій-іонних акумуляторів типу 18650, які широко використовуються в різноманітних портативних електронних пристроях. Цей модуль забезпечує надійну та стабільну роботу завдяки підтримці сучасних технологій та захистів.



Рисунок 1.11 — Модуль живлення та зарядки 18650 microUSB/Type-C 5B/3A AOC708

AOC708 має два інтерфейси для зарядки: microUSB та Type-C, що дозволяє використовувати його з різними кабелями та зарядними пристроями. Це зручно, оскільки microUSB ще досить поширений, а Type-C набуває все більшої популярності завдяки своїй зручності та високій швидкості передачі даних та енергії.

Вихідна напруга модуля становить 5 В, що є стандартом для більшості портативних пристроїв. Максимальний вихідний струм досягає 3 А, що дозволяє заряджати пристрої швидко та ефективно. Модуль здатен забезпечити стабільну подачу енергії, що особливо важливо для чутливих до коливань напруги електронних компонентів.

Однією з ключових особливостей модуля АОС708 є вбудовані захисти. Він обладнаний системами захисту від перезарядки, перерозряду, короткого замикання та перевантаження по струму. Модуль також оснащений світлодіодними індикаторами, які дозволяють візуально контролювати стан зарядки та рівень заряду акумулятора. Це зручно для користувачів, оскільки дозволяє швидко оцінити стан батареї без необхідності підключення до додаткових вимірювальних приладів.

Завдяки компактним розмірам, модуль АОС708 ідеально підходить для мікроконтролерів Arduino. Простота монтажу та використання робить цей модуль зручним для розробників та ентузіастів електроніки.

1.3. Розробка програмно-апаратного забезпечення

На основі вибраних елементів та модулів розроблена принципова схема системи, яка показана на рисунку 1.12.

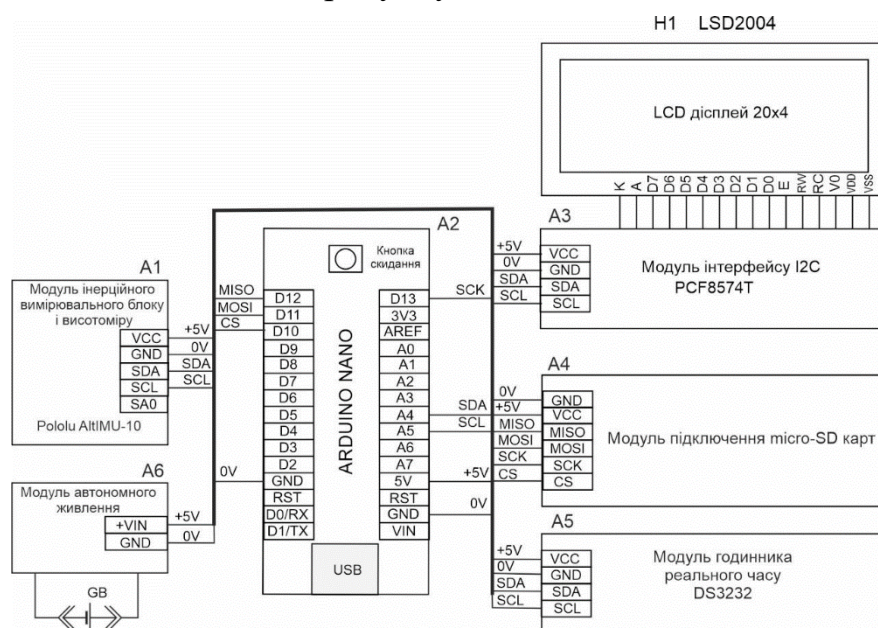


Рисунок 1.12 — Принципова схема портативного пристрою контролю положення у просторі з можливістю запису інформації у часі

Вимірювальний пристрій, який складається з комплексу датчиків, під'єднаний до мікроконтролера Arduino Nano v3. Він зчитує положення пристрою у просторі, має на собі барометр, 3-осьовий гіроскоп, магніт блок та висотомір.

Данні з датчиків передаються на мікроконтролер для обробки. Після цього, дані виводяться на ЖК екран, та дублюються на логер з microSD картою. Система живиться від пари акумуляторів типу 18650, які з'єднані послідовно, та підключені до мініатюрного лабораторного блоку живлення для макетних плат. Також цей модуль виконує роль захисту від переполюсовки батареї. Також, до мікроконтролера під'єднаний модуль часу, за допомогою якого можливо хронологічно відновити покази датчиків.

Для розробки програмного забезпечення використовувалося середовище розробки Arduino IDE. Це середовище включає в себе вбудований текстовий редактор для написання коду, область повідомлень, монітор порта для відображення повідомлень з мікроконтролера та введення даних до нього через консоль, панель інструментів з кнопками для часто використовуваних команд, а також декілька меню. Особливо важливим є те, що середовище розробки може підключатися безпосередньо до апаратної частини Arduino через USB, що дозволяє завантажувати програми та здійснювати зв'язок з мікроконтролером.

Під час розробки приладу було виконано прототипування для реалізації базової функціональності майбутнього виробу. На цьому етапі ретельно аналізувалася робота пристрою в цілому. Після етапу прототипування обов'язково слідують етапи перегляду архітектури системи, розробки, впровадження та тестування кінцевого продукту. Завантажимо скетч на Arduino Nano та відкриємо вікно серійного монітора, щоб відстежувати результати.

Скетч повинен виводити покази датчиків у вікні серійного монітора.

Підключення необхідних бібліотек до проєкту:

```
1 #include <DS3231.h>
2 #include <SPI.h>
3 #include <SD.h>
4 #include <Wire.h>
5 #include <LSM6.h>
6 #include <LPS.h>
7 #include <LiquidCrystal_I2C.h>
```

Ініціалізація датчиків та пристроїв виводу інформації:

```
9 LiquidCrystal_I2C lcd(0x27, 20, 4);
10 LPS ps;
11 LSM6 imu;
12 char report[80];
13
14 File dataFile;
15
16 const int chipSelect = 10;
17 DS3231 rtc(SDA, SCL);
```

Ініціалізація з'єднання через I2C:

```
19 void setup() {
20     rtc.begin();
21     Serial.begin(9600);
22     Wire.begin();
```

Перевірка підключення всіх датчиків:

```
24     if (!imu.init()) {
25         Serial.println("Failed to detect and initialize IMU!");
26         while (1);
27     }
28     imu.enableDefault();
29
30     if (!ps.init()) {
31         Serial.println("Failed to autodetect pressure sensor!");
32         while (1);
33     }
34     ps.enableDefault();
```

Підключення дисплея:

```
36     lcd.init();
37     lcd.backlight();
```

Налаштування роботи з картою пам'яті та створення структури запису

даних:

```
39 | Serial.print("Initializing SD card...");
40 | if (!SD.begin(chipSelect)) {
41 |     Serial.println("initialization failed!");
42 |     return;
43 | }
44 | Serial.println("initialization done.");
45 |
46 | dataFile = SD.open("LOGDATA.txt", FILE_WRITE);
47 | if (dataFile) {
48 |     Serial.println("File opened ok");
49 |
50 |     dataFile.println("    Date,    Time,    Ax,    Ay,    Az,    Gx,    Gy,    Gz,    H,    T");
51 |     dataFile.close();
52 | }
```

Налаштування для PLX-DAQ

```
54 | Serial.println("CLEARDATA");
55 | Serial.println("LABEL,Date,Logging Time,Time,AX,AY,AZ,GX,GY,GX,High,Temperature,");
56 | Serial.println("RESETTIMER");
```

Зчитування та форматування даних для виводу на серійний монітор

```
59 | void loop() {
60 |     imu.read();
71 |     dataFile = SD.open("LOGDATA.txt", FILE_WRITE);
72 |     if (dataFile) {
73 |
74 |         dataFile.print(rtc.getDateStr());
75 |         dataFile.print(",");
76 |         dataFile.print(rtc.getTimeStr());
77 |         dataFile.print(",");
78 |         dataFile.print("G: %6d %6d %6d",
79 |             imu.g.x, imu.g.y, imu.g.z);
80 |         dataFile.print(",");
81 |         dataFile.print("sure");
82 |         dataFile.println();
83 |     }
84 | }
```

Відкриття запис даних на карту пам'яті

```
79 |     dataFile.print(imu.a.x);
80 |     dataFile.print(",");
81 |     dataFile.print(imu.a.y);
82 |     dataFile.print(",");
83 |     dataFile.print(imu.a.z);
84 |     dataFile.print(",");
85 |     dataFile.print(imu.g.x);
86 |     dataFile.print(",");
87 |     dataFile.print(imu.g.y);
88 |     dataFile.print(",");
89 |     dataFile.print(imu.g.z);
90 |     dataFile.print(",");
91 |
92 |     dataFile.print(altitude);
93 |     dataFile.print(",");
94 |     dataFile.print(temperature);
95 |     dataFile.println();
96 | }
```

```
93     dataFile.print(altitude);
94     dataFile.print(",");
95     dataFile.print(temperature);
96     dataFile.println();
```

Збереження файлу, або відображення помилки

```
98     dataFile.close();
99   } else {
100     Serial.println("OOPS!! SD card writing failed");
101   }
```

Запис даних для PLX-DAQ

```
103 Serial.print("DATA, DATE, TIMER, TIME,");
104 Serial.println();
```

Приклад виводу інформації на дисплей з затримкою у 2 секунди

```

133     lcd.print(" ");
134     lcd.setCursor(14, 2);
135     lcd.print(imu.g.z);
136     lcd.setCursor(0, 3);
137     lcd.print("H=");
138     lcd.setCursor(2, 3);
139     lcd.print(altitude);
140     lcd.setCursor(10, 3);
141     lcd.print("T=");
142     lcd.setCursor(12, 3);
143     lcd.print(temperature);
        delay(2000);

```

Дані з датчиків можливо вивести не тільки на екран пристрою а і переглянути в самій програмі ArduinoIDE, що зображено на рисунку 1.13

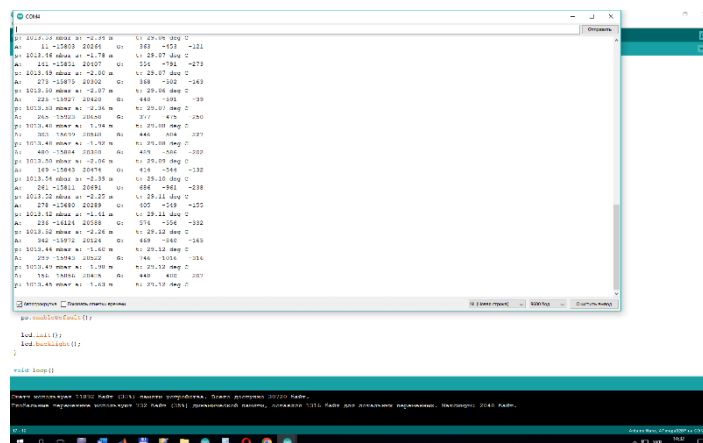
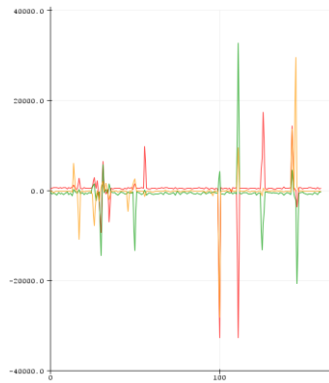
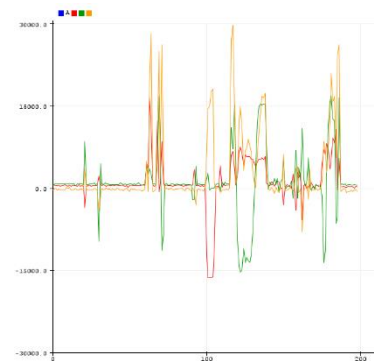


Рисунок 1.13 — Перегляд даних з датчиків через ArduinoIDE

Також, ці дані можливо вивести не тільки у числовому вигляді на SD-карту чи екран пристрою, також є можливість візуалізувати положення об'єкту за допомогою плотера у програмному забезпеченні Arduino IDE для



більш детального аналізу. Приклади графіків, зроблених за допомогою даних



з акселерометра та гіроскопа показано на рисунку 1.14 та 1.15.

Рисунок 1.14 — Дані зняті з гіроскопа у вигляді графіку

Рисунок 1.15 — Дані зняті з акселерометра у вигляді графіку

Для зчитування даних, записаних на SD-карту, потрібно підключити її до комп'ютеру та відкрити файл LOGDATA.TXT, у якому записані усі показники з датчиків. Приклад записаних даних показано на рисунку 1.16

LOGDATA - Блокнот

Файл Правка Формат Вид Справка

Date	Time	Ax	Ay	Az	Gx	Gy	Gz	H	T
23.06.2024	13:43:26	-815	1603	-15238	-908	-5741	-968	116.63	24.95
23.06.2024	13:43:30	485	1549	-14923	-6241	-2149	32	116.42	25.00
23.06.2024	13:43:33	-3158	-12496	10411	5562	-442	2130	115.82	25.29
23.06.2024	13:43:36	-4377	-15615	3065	274	25	-235	116.23	25.49
23.06.2024	13:43:39	-3731	-11787	-12266	-4616	252	3368	116.53	25.61
23.06.2024	13:43:42	3399	-1911	-16072	319	-384	-19	116.00	25.80
23.06.2024	13:43:46	-2515	-5369	-14836	-159	-2142	182	116.05	25.93
23.06.2024	13:43:49	-62	-15412	-7634	6346	-1192	1362	116.05	26.04
23.06.2024	13:43:52	-4957	-15052	3873	578	-580	-185	116.71	26.21
23.06.2024	13:43:55	-5122	-15076	4294	207	-519	-187	116.37	26.31
23.06.2024	13:43:59	-5037	-15221	3537	301	-691	-360	115.87	26.40
23.06.2024	13:44:02	-5248	-15132	3644	1081	-637	-345	116.20	26.46
23.06.2024	13:44:05	-5017	-15021	3895	329	-910	-287	115.93	26.48
23.06.2024	13:44:08	-811	3274	-16032	416	-656	-218	115.63	26.52
23.06.2024	13:44:11	-693	3609	-15801	371	-534	-199	115.99	26.35
23.06.2024	13:44:15	-652	3386	-16041	402	-831	-82	116.11	26.18
23.06.2024	13:44:18	-1867	2867	-15921	1298	-579	334	116.17	26.01
23.06.2024	13:44:21	12289	5497	-9293	2507	-1588	662	116.30	26.00
23.06.2024	13:44:24	13044	-3925	8954	-276	-1089	-400	116.67	26.07
23.06.2024	13:44:28	13464	-5028	8914	348	-468	240	116.47	26.00
23.06.2024	13:44:31	9267	-8631	10827	-1650	7757	9753	115.76	25.95
23.06.2024	13:44:34	5776	-9460	12343	587	-644	-75	116.60	25.88
23.06.2024	13:44:37	7203	-9163	12267	648	-1319	-673	116.82	25.91
23.06.2024	13:44:40	13468	5250	-8083	111	-707	1227	116.01	25.89
23.06.2024	13:44:44	11333	-8095	7879	2097	-5808	934	116.33	25.94
23.06.2024	13:44:47	15752	208	7686	-6094	10633	3295	115.74	25.91
23.06.2024	13:44:50	13346	-2594	10002	-1220	449	988	116.19	25.84
23.06.2024	13:44:53	-1379	-8247	15236	3529	-1837	958	115.52	25.78
23.06.2024	13:44:57	9322	7088	-11264	936	561	926	116.48	25.76
23.06.2024	13:45:00	15820	-2408	3811	1598	-82	-8075	115.92	25.71
23.06.2024	13:45:03	11067	2011	11606	8448	4206	-2876	115.72	25.72
23.06.2024	13:45:06	15264	4600	-3758	3430	-2272	4312	116.06	25.71

Стр 1, стр 1 100% Windows (CRLF) UTF-8

Рисунок 1.16 – Масив даних записаних з датчиків

1.4 Перевірка функціонування пристрою

На основі структурної та принципової схеми було зібрано лабораторний макет (рис. 1.17) для налагодження програмних і апаратних складових.

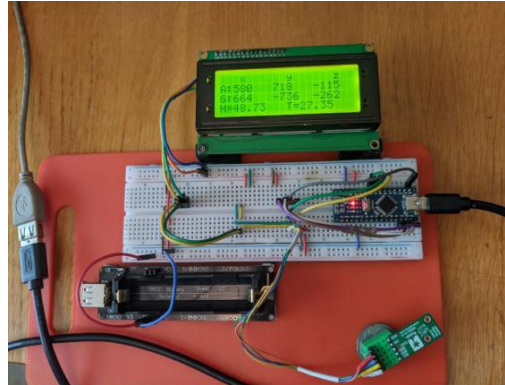


Рисунок 1.17 — Макет пристрою контролю положення у просторі з
можливістю запису інформації у часі

Макет протестовано у лабораторних умовах, після чого можна констатувати, що він повністю робочий. Приклад увімкненого пристрою зображено на рисунку 1.17.

ВИСНОВКИ

При розробці проекту описаному в даній роботі було прийнято рішення використовувати наступні компоненти: датчики — гіроскоп, акселерометр та магнітометр, об'єдані на платі Pololu ALTIMU-10v5, модуль реєстратору часу Real Time Clock на DS3231SN, модуль реєстрації даних на SD-карту AOC431, модуль відображення інформації LCD 2004, модуль живлення та зарядки 18650 microUSB/Type-C 5B/3A AOC708, та мікроконтролер Arduino Nano.

Обрана плата з датчиками акселерометру, гіроскопу та магнітометру забезпечує ефективні заміри інформації у просторі. Ці дані дозволяють точно визначити просторове положення об'єкту у який встановлено розроблений пристрій.

Модуль реєстратору часу обрано задля прив'язки даних з датчиків до часу, що дозволяє мати розуміння про те, в який момент пристрій був в тій чи іншій точці у простору.

Модуль реєстрації даних присутній для можливості запису даних з усіх датчиків на SD-карту.

Екран LCD 2004 присутній для відображення отримуємої інформації з датчиків.

Модуль живлення та зарядки слугує для можливості автономного користування приладом.

Обрані компоненти взаємодіють між собою для створення ефективної та точної системи контролю положення та реєстрації об'єкту у просторі з можливістю запису інформації у часу, яка забезпечує максимально точну інформацію о положенні конкретного об'єкту у просторі у конкретний час.

В процесі проектування виготовлено макет пристрою. Це дозволило провести налагодження апаратних та програмних засобів, а також виявити та усунути недоліки цих засобів.

У програмі ArduinoIDE було розроблено програму у вигляді коду. Також перевірено на помилки.

В роботі враховані такі виклики:

- Енергоспоживання пристрою при роботі від батареї.
- Надійність з'єднань під час вібрацій або механічних навантажень.
- Точність вимірювань, що залежить від температури, електромагнітних завад і калібрування.
- Складність програмування для синхронізації модулів.
- Обмежена пам'ять Arduino Nano.
- Сумісність модулів та конфлікти за адресами I2C.

Незважаючи на ці проблеми, портативні пристрої для контролю положення з можливістю запису інформації у часі залишаються ефективними в умовах обмеженого бюджету та необхідності портативності. Важливо забезпечити належний дизайн, обслуговування та захист пристрою для досягнення оптимальної ефективності.

У програмі ArduinoIDE було розроблено програму у вигляді коду. Також перевірено на помилки.

Важливо враховувати потенційні недоліки та забезпечити належну розробку, обслуговування та захист портативного пристрою контролю положення у просторі для досягнення оптимальної ефективності. Використання датчиків гіроскопа, акселерометра, магнітометра та тиску може допомогти покращити точність та функціональність пристрою, що дозволить максимально ефективно використовувати його можливості навіть за обмежених ресурсів.

ПЕРЕЛІК ПОСИЛАНЬ

1. V. Kaajakari, «Practical MEMS: Design of microsystems, accelerometers, gyroscopes, RF MEMS, optical MEMS, and microfluidic systems Illustrated Edition», Small Gear Publishing, 2009p, 496c.
2. E. Bekir, «Introduction to Modern Navigation Systems», World Scientific, 2007p., 240 c.
3. Tai-Ran Hsu, «MEMS and Microsystems: Design, Manufacture, and Nanoscale Engineering», John Wiley & Sons, 2020 p., 576 c.
4. Averil B. Chatfield, «Fundamentals Of High Accuracy Inertial Navigation», the American Institute of Aeronautics and Astronautics, Inc., 1997, 328c.
5. Anthony Lawrence, «Modern Inertial Technology: Navigation, Guidance, and Control», Springer Science & Business Media, 2001 p., 278 c.

6.

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]

[REDACTED]