

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Чорноморський національний університет

імені Петра Могили

Факультет комп'ютерних наук

Кафедра комп'ютерної інженерії

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри,

д-р техн. наук, проф.

_____ І. М. Журавська

«__» _____ 2024 р.

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

Апаратно-програмний модуль

тестування ІМУ-сенсорів

Спеціальність 123 Комп'ютерна інженерія

123 – КБР.01 – 405. 22230501

Студент

_____ В. В. Охотський

підпис

«__» _____ 202__ р.

Керівник канд. техн. наук, доцент

_____ Я. М. Крайник

підпис

«__» _____ 202__ р.

Миколаїв – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Чорноморський національний університет імені Петра Могили Факультет комп'ютерних наук Кафедра комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Зав. кафедри _____ І. М. Журавська

« _____ » _____ 2024 р.

ЗАВДАННЯ на виконання кваліфікаційної бакалаврської роботи

Видано студенту групи 405 факультету комп'ютерних наук
Охотський Валентин Валентинович

(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи

Апаратно-програмний модуль тестування IMU-сенсорів

Затверджена наказом по ЧНУ ім. Петра Могили від 30.01.2024 № 17.

2. Строк представлення кваліфікаційної роботи « _____ » _____ 20__ р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні

Очікуваним результатом роботи є створення апаратно-програмного комплексу, який дозволяє тестувати IMU-сенсори для визначення їх характеристик і можливих похибок. Початковими даними є специфікації обраних для тестування IMU-сенсорів, програмні бібліотеки для роботи з ними та технічна документація.

4. Перелік питань, що підлягають розробці

Дослідження існуючих рішень для тестування IMU-сенсорів

Вибір відповідних IMU-сенсорів для тестування

Розробка апаратного забезпечення для тестування IMU-сенсорів

Розробка програмного забезпечення для збору і аналізу даних з IMU-сенсорів

Проведення тестування обраних сенсорів і аналіз результатів

Визначення похибок і методів калібрування сенсорів

Написання документації по розробленому апаратно-програмному комплексу

5. Перелік графічних матеріалів

Графіки залежності показників точності сенсорів від умов тестування.

Екранні форми програмного забезпечення для збору та аналізу даних з ІМУ-сенсорів.

Результати тестування обраних сенсорів у вигляді діаграм.

Схематичне зображення тестового стенду для ІМУ-сенсорів.

6. Завдання до спеціальної частини

Зробити на тему Апаратно-програмний модуль тестування ІМУ-сенсорів

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
Макарова О.В. Старший викладач	Кафедра екології Медичного інституту ЧНУ ім. Петра Могили	Спеціальна частина з охорони праці

Керівник роботи

канд. тех. наук, доцент Крайник Ярослав Михайлович
(посада, прізвище, ім'я, по батькові)

(підпис)

Завдання прийнято до виконання

Охотський Валентин Валентинович

(прізвище, ім'я, по батькові студента)

(підпис)

Дата видачі завдання « ____ » _____ 2024 р.

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

Тема: Апаратно-програмний модуль тестування ІМУ-сенсорів

№	Найменування роботи	Початок	Закінчення	Примітки
1	Розробка та затвердження завдання на виконання КР	01.02.2024	05.02.2024	Виконав
2	Огляд літератури за темою роботи	06.02.2024	21.02.2024	Виконав
3	Складання календарного плану БКР	22.02.2024	01.03.2024	Виконав
4	Аналіз предметної області	02.03.2024	23.03.2024	Виконав
5	Розробка проектних рішень	24.03.2024	15.04.2024	Виконав
6	Моделювання та конструювання АПЗ	16.04.2024	30.04.2024	Виконав
7	Перевірка працездатності, тестування та апробація розробленого АПЗ, аналіз результатів тестування	01.05.2024	14.05.2024	Виконав
8	Відгук керівника КР	06.06.2024	08.06.2024	Виконав
9	Оформлення БКР та презентації	15.05.2024	05.06.2024	Виконав
10	Попередній захист	28.05.2024	05.06.2024	Виконав
11	Рецензування	10.06.2024	14.06.2024	Виконав
12	Завершення оформлення КР та презентації	15.06.2024	20.06.2024	Виконав
13	Захист бакалаврської кваліфікаційної роботи	.06.2024	.06.2024	

Розробив здобувач ВО Охотський Валентин Валентинович

(прізвище, ім'я, по батькові)

(підпис)

« ____ » _____ 2024 р.

Керівник роботи канд. тех. наук, доцент Крайник Ярослав Михайлович

(посада, прізвище, ім'я, по батькові)

(підпис)

« ____ » _____ 2024 р.

АНОТАЦІЯ

до кваліфікаційної бакалаврської роботи
«Апаратно-програмний модуль тестування ІМУ-сенсорів»
Студент 405 гр.: Охотський Валентин Валентинович
Керівник: к.т.н., доцент Крайник Ярослав Михайлович

У цій дипломній роботі розглядається розробка апаратно-програмного модуля для тестування ІМУ-сенсорів (інерціальних вимірювальних блоків). ІМУ-сенсори широко використовуються у різних галузях, включаючи робототехніку, аерокосмічну промисловість, автомобілебудування, мобільні пристрої та системи навігації, для вимірювання кутової швидкості, лінійного прискорення та орієнтації. Високоточне тестування та калібрування цих сенсорів є критичними для забезпечення їх надійної та точної роботи.

Мета роботи: створення модуля, який дозволяє автоматизовано та ефективно тестувати ІМУ-сенсори, забезпечуючи при цьому високу точність вимірювань і зниження затрат часу на калібрування.

Складові роботи:

1. **Вступ:** Представляє загальні відомості про ІМУ-сенсори, їх важливість у різних галузях та актуальність розробки модуля для їх тестування.

2. **Розділ 1:** Аналіз вимог до апаратного та програмного забезпечення для тестування ІМУ-сенсорів. Визначення необхідних параметрів вимірювань, інтерфейсів зв'язку та методів обробки даних.

3. **Розділ 2:** Розробка апаратної частини модуля. Опис мікроконтролера, комунікаційних інтерфейсів (I2C, SPI, UART), сенсорних підсилювачів та інших допоміжних компонентів. Вибір елементів з високою точністю та надійністю.

4. **Розділ 3:** Створення програмного забезпечення. Мікропрограма для мікроконтролера, драйвери для взаємодії з сенсорами та алгоритми обробки даних. Методи та алгоритми для автоматичного тестування та калібрування сенсорів, включаючи методи компенсування похибок, фільтрацію шумів та обробку сигналів.

5. **Висновки:** Підсумовують основні результати роботи, досягнуті цілі та можливості подальшого розвитку.

Результати: створено функціональний прототип апаратно-програмного модуля, який здатен ефективно тестувати та калібрувати різні типи ІМУ-сенсорів. Прототип успішно пройшов тестування, демонструючи високу точність та стабільність вимірювань. Запропоноване рішення дозволяє значно знизити затрати часу на тестування, підвищити точність результатів та автоматизувати процеси перевірки ІМУ-сенсорів.

Ключові слова: ІМУ-сенсори, тестування, калібрування, апаратно-програмний модуль, мікроконтролер, алгоритми обробки даних.

Сторінок – 65. Рисуноків – 13. Таблиць – 5. Посилань – 16. Додатків – 4.

ABSTRACT

of the Bachelor's Thesis

"Hardware and software module for testing IMU sensors"

Student: Valentin Okhotskyi

Supervisor: Krainyk Yaroslav

This thesis deals with the development of a hardware and software module for testing IMU sensors (inertial measuring units). IMU sensors are widely used in various industries, including robotics, aerospace, automotive, mobile devices, and navigation systems, to measure angular velocity, linear acceleration, and orientation. High precision testing and calibration of these sensors is critical to ensure their reliable and accurate performance.

The goal of the work: creation of a module that allows automated and efficient testing of IMU sensors, while ensuring high measurement accuracy and reducing calibration time.

Component works:

1. **Introduction:** Provides general information about IMU sensors, their importance in various industries, and the relevance of developing a module for their testing.

2. **Chapter 1:** Analysis of hardware and software requirements for testing IMU sensors. Determination of some measurement parameters, communication interfaces and data processing methods.

3. **Chapter 2:** Development of the hardware part of the module. Description of the microcontroller, communication interfaces (I2C, SPI, UART), sense amplifiers and other auxiliary components. Selection of elements with high accuracy and reliability.

4. **Chapter 3:** Creating Software. Firmware for the microcontroller, drivers for interaction with sensors and data processing algorithms. Methods and algorithms for automatic sensor testing and calibration, including methods for error compensation, noise filtering, and signal processing.

5. **Conclusions:** Summarize the main results of the work, achieved goals and opportunities for further development.

Results: a functional prototype of the hardware and software module was created, which is able to effectively test and calibrate various types of IMU sensors. The prototype was successfully tested, demonstrating high accuracy and stability of measurements. The proposed solution allows you to significantly reduce the time spent on testing, increase the accuracy of the results, and automate the processes of checking IMU sensors.

Keywords: IMU sensors, testing, calibration, hardware and software module, microcontroller, data processing algorithms.

Pages – 65. Figures – 13. Tables – 5. References – 16. Appendices – 4.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	3
ВСТУП	4
1 ПРИНЦИПИ РОБОТИ ІМУ-СЕНСОРІВ	6
1.1 Основні компоненти ІМУ.....	6
1.2 Принципи вимірювання.....	13
1.3 Використання ІМУ в різних галузях	16
Висновки до розділу 1	18
2 АПАРАТНО-ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ТЕСТУВАННЯ ІМУ.....	20
2.1 Огляд апаратних компонентів.....	20
2.2 Мікроконтролер STM32F407	26
2.3 9-осьовий датчик LSM9DS	27
2.4 Огляд програмного забезпечення	28
2.5 Середовище розробки STM32Cube IDE.....	32
2.6 Відлагоджувальна плата GlobalLogic Embedded Starter Kit.....	33
2.7 Взаємодія апаратного та програмного модулів.....	33
Висновки до розділу 2	35
3 МЕТОДИКА ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ.....	36
3.1 Процес тестування ІМУ-сенсорів.....	36
3.2 Налаштування для тестування датчику LSM9DS	43
3.3 Тестування роботи магнетометру у складі LSM9DS.....	45
3.4 Аналіз результатів тестування	47
3.5 Оптимізація процесу тестування	49
Висновки до розділу 3	50
ВИСНОВКИ.....	52
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	54
ДОДАТОК А Довідка про перевірку на унікальність пояснювальної записки	56
ДОДАТОК Б ЛІСТИНГ КОДУ АПАРАТНО-ПРОГРАМНОГО МОДУЛЯ.....	57
ДОДАТОК В	59
ДОДАТОК Г	61

ПЕРЕЛІК СКОРОЧЕНЬ

API – Application Programing Interface

CPU – Central Processing Unit

GUI – Graphical User Interface

IMU – Inertial Measurement Unit

USB – Universal Serial Bus

ВСТУП

Актуальність. Інерціальні вимірювальні одиниці (ІМУ) є важливими компонентами в багатьох сучасних технологіях, від смартфонів до безпілотних літальних апаратів. Вони дозволяють точно визначати орієнтацію, швидкість і прискорення об'єкта, на якому вони встановлені. Ефективність і точність цих сенсорів мають критичне значення для успішного функціонування багатьох систем. У зв'язку з цим, створення апаратно-програмних модулів для їх тестування є важливим завданням для забезпечення якості та надійності ІМУ-сенсорів. У даній роботі розглядаються основні принципи роботи ІМУ-сенсорів, структура апаратно-програмного модуля для їх тестування, а також методика тестування та аналіз отриманих результатів.

У сучасному світі, де технології швидко розвиваються і проникають у всі сфери життя, надійність і точність сенсорів відіграють вирішальну роль. ІМУ-сенсори використовуються в критичних додатках, таких як навігаційні системи безпілотних літальних апаратів, медичне обладнання та автономні транспортні засоби. Помилки в роботі цих сенсорів можуть призвести до серйозних наслідків, тому важливо мати надійні методи їх тестування та калібрування.

Метою даної роботи є розробка та впровадження апаратно-програмного модуля для тестування ІМУ-сенсорів, що забезпечить високу точність і надійність їх роботи.

Завдання

1. вивчити основні принципи роботи ІМУ-сенсорів;
2. розробити апаратний модуль для тестування ІМУ-сенсорів;
3. розробити програмне забезпечення для управління процесом тестування та обробки даних;
4. провести тестування розробленого модуля на різних типах ІМУ-сенсорів;
5. проаналізувати результати тестування та оптимізувати методику.

Об'єктом дослідження є інерціальні вимірювальні одиниці (ІМУ), що використовуються в різних технологічних додатках.

Предметом дослідження є методи та засоби тестування ІМУ-сенсорів з використанням апаратно-програмного модуля.

Методи дослідження:

1. метод порівняння: використано декілька систем з подібним функціоналом для аналізу спільних характеристик, переваг та недоліків аналогічних систем;
2. метод індукції: сформовано індукційні висновки щодо можливості застосування певного апаратного та програмного забезпечення для реалізації комплексу збору та обробки даних фізіологічних показників людини;
3. методи аналізу та синтезу: використано для декомпозиції окремих функцій пристроїв збору та обробки серцевого ритму, кисневої сатурації крові та температури, щоб визначити необхідні апаратні та програмні складові;
4. метод моделювання: застосовано для створення програмного забезпечення; на початковому етапі розроблено модель архітектури, яка відображає взаємозв'язок модулів.

Практичне значення. Розроблений апаратно-програмний модуль може бути використаний у виробничих і дослідницьких лабораторіях для забезпечення високої якості ІМУ-сенсорів. Це дозволить зменшити кількість дефектних виробів, підвищити точність вимірювань і покращити загальну надійність систем, які використовують ці сенсори.

Структура роботи. Робота складається з вступу, трьох розділів, висновків та списку використаних джерел.

1 ПРИНЦИПИ РОБОТИ ІМУ-СЕНСОРІВ

1.1 Основні компоненти ІМУ

Основними компонентами інерціальної вимірювальної одиниці (ІМУ) є акселерометри, гіроскопи та іноді магнетометри. Кожен з цих сенсорів відіграє важливу роль у забезпеченні точних вимірювань орієнтації, швидкості та прискорення об'єкта (табл. 1.1).

Таблиця 1.1 – Основні компоненти ІМУ [10]

Компонент	Призначення	Принцип роботи	Типи
Акселерометри	Вимірюють лінійне прискорення вздовж трьох осей (X, Y, Z)	MEMS-датчики, що вимірюють зміну положення маси	П'єзоелектричні, ємнісні, резистивні
Гіроскопи	Вимірюють куту швидкість обертання навколо трьох осей (X, Y, Z)	MEMS-датчики, що використовують ефект Коріоліса	Механічні, оптичні, MEMS
Магнетометри	Вимірюють орієнтацію відносно магнітного поля Землі	Датчики на основі ефекту Холла або магніторезистивні матеріали	Індукційні, магніторезистивні, на основі ефекту Холла
Комбіновані сенсори	Об'єднують акселерометри, гіроскопи та магнітометри в одному корпусі	Синтез даних з кількох сенсорів	-

Компонент	Призначення	Принцип роботи	Типи
Мікроконтролери	Обробляють дані з сенсорів, виконують калібрування і фільтрацію	Алгоритми, такі як фільтр Калмана	Різні за обчислювальною потужністю та енергоефективністю
Інтерфейси зв'язку	Забезпечують передачу даних між IMU та іншими компонентами системи	I2C, SPI, UART, CAN	

Акселерометри є ключовими компонентами інерціальних вимірювальних одиниць (IMU), які вимірюють лінійне прискорення об'єкта у просторі. Вони визначають, як швидко змінюється швидкість об'єкта у трьох взаємно перпендикулярних осях: X, Y та Z. Це дозволяє не лише виявити рух об'єкта, але й точно виміряти зміни його швидкості в кожній з цих трьох напрямків.(рис. 1.1)

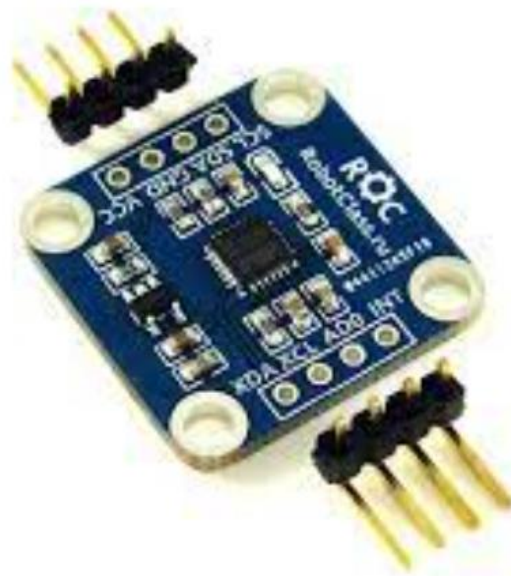


Рисунок 1.1 – Акселерометр

Принцип роботи акселерометрів ґрунтується на фізичних законах механіки, зокрема на використанні маси, яка змінює своє положення під впливом прискорення. Усередині акселерометра розміщується маса, яка з'єднана з основою сенсора за допомогою пружних елементів. Коли об'єкт, на якому встановлено акселерометр, починає рухатися або змінює свою швидкість, маса всередині сенсора зміщується відносно своєї початкової позиції через інерцію [11, с. 63-67].

Це зміщення маси фіксується за допомогою різних методів, залежно від типу акселерометра [11, с. 63-67]:

1. п'єзоелектричні акселерометри: у таких акселерометрах використовується п'єзоелектричний матеріал, який генерує електричний заряд у відповідь на механічне напруження, спричинене зміщенням маси. Цей електричний заряд пропорційний до величини прискорення;

2. капаситивні акселерометри: вони використовують зміни ємності конденсаторів, спричинені зміщенням маси. Зміна відстані між пластинами конденсатора змінює його ємність, що дозволяє вимірювати прискорення;

3. термоакустичні акселерометри: в цих сенсорах використовуються термоакустичні ефекти, де зміщення маси змінює акустичні хвилі або теплові потоки, які можуть бути виміряні для визначення прискорення;

4. оптичні акселерометри: вони використовують оптичні методи, наприклад, зміщення маси змінює шлях світла, що проходить через сенсор, і ці зміни фіксуються фотодетекторами.

Крім того, акселерометри можуть бути одноосьовими або багатоосьовими, залежно від кількості вимірюваних напрямків прискорення. Одноосні акселерометри вимірюють прискорення лише в одному напрямку, тоді як багатоосні можуть вимірювати прискорення у двох або трьох взаємно перпендикулярних осях.

Отримані дані з акселерометрів можуть бути використані в різних застосуваннях. Наприклад, в смартфонах акселерометри використовуються для

визначення орієнтації екрану, в автомобілях – для систем активної безпеки, а в безпілотних літальних апаратах – для стабілізації та навігації.

Таким чином, акселерометри є критично важливими сенсорами, що забезпечують точні вимірювання прискорення, необхідні для різних технологічних застосувань.

Гіроскопи є важливими компонентами інерціальних вимірювальних одиниць (IMU), призначеними для вимірювання кутової швидкості, тобто швидкості обертання об'єкта навколо його осей. Вони дозволяють визначити, наскільки швидко і в якому напрямку обертається об'єкт, що є критично важливим для багатьох технологічних додатків, таких як навігація, стабілізація та управління рухом.(рис.1.2.) [3].

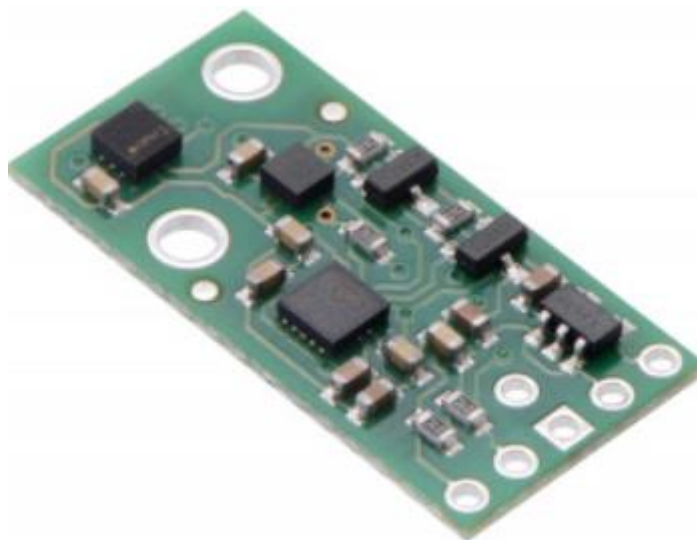


Рисунок 1.2 – Гіроскоп та акселерометр на чіпі

Принцип роботи сучасних гіроскопів часто базується на ефекті Саймона-Коріоліса. Цей ефект виникає внаслідок обертання об'єкта і впливає на рух елементів всередині гіроскопа. Для більш детального розуміння цього принципу розглянемо, як саме працюють гіроскопи на основі цього ефекту.

Всередині гіроскопа розташовані рухомі елементи, які можуть бути мембранами, масами або навіть мікромеханічними структурами (у випадку мікроелектромеханічних систем, або MEMS). Коли гіроскоп перебуває в стані

спокою, ці елементи рухаються або вібрують в певному напрямку. Проте, коли гіроскоп починає обертатися навколо будь-якої з його осей, на ці рухомі елементи починає діяти ефект Коріоліса.

Ефект Коріоліса призводить до виникнення відхилення у русі цих елементів, перпендикулярного до напрямку їх початкового руху та осі обертання. Це відхилення створює додаткові сили, які можуть бути виміряні за допомогою різних методів:

1. MEMS-гіроскопи: в мікроелектромеханічних системах використовуються мініатюрні структури, що коливаються. Коли виникає обертання, ефект Коріоліса змінює напрямок і амплітуду цих коливань. Ці зміни фіксуються за допомогою сенсорів, що вимірюють зміщення або зміни у ємності конденсаторів, пов'язаних з рухомими елементами;

2. оптичні гіроскопи: у таких гіроскопах використовуються інтерферометричні методи. Вони включають оптичні волокна або лазери, які проходять через кільцевий інтерферометр. Коли гіроскоп обертається, ефект Коріоліса змінює шлях світлових хвиль, що проходять у різних напрямках. Ці зміни викликають фазовий зсув, який може бути виміряний для визначення кутової швидкості;

3. волоконно-оптичні гіроскопи: ці гіроскопи використовують довгі котушки оптичних волокон. Світло, що проходить через ці волокна, змінює свій шлях під впливом обертання, створюючи інтерференційні візерунки, які фіксуються детекторами;

4. гіроскопи на основі резонаторів: в таких гіроскопах використовуються резонуючі структури, які змінюють свою частоту або амплітуду коливань під впливом обертання. Ці зміни вимірюються і використовуються для визначення кутової швидкості.

Всі ці типи гіроскопів забезпечують високу точність і надійність вимірювань кутової швидкості. Дані з гіроскопів можуть бути інтегровані для визначення кута обертання об'єкта у просторі з плином часу. Це дозволяє застосовувати їх у різних

сферах, включаючи авіацію, космічні технології, робототехніку, мобільні пристрої та інші системи, де важлива точність вимірювань обертання.

Таким чином, гіроскопи є невід'ємною частиною сучасних інерціальних вимірювальних одиниць, забезпечуючи точне вимірювання кутової швидкості і допомагаючи у виконанні складних завдань навігації, стабілізації та управління рухом.

магнетометри є важливими компонентами інерціальних вимірювальних одиниць (IMU), які використовуються для визначення орієнтації об'єкта відносно магнітного поля Землі. Вони вимірюють вектор магнітного поля у трьох вимірах, що дозволяє точно визначити напрямок на північ і, таким чином, орієнтацію об'єкта в горизонтальній площині (рис.1.3)[4].



Рис.1.3. Магнетометр

Принцип роботи магнетометрів базується на здатності певних матеріалів реагувати на магнітні поля. Існують різні типи магнетометрів, але найбільш поширеними в IMU є наступні:

1. Флюксгейти: це сенсори, які використовують магнітні властивості матеріалів для вимірювання магнітного поля. Флюксгейт складається з сердечника з високою магнітною проникністю, навколо якого обмотані котушки. Коли через ці котушки проходить змінний струм, сердечник намагнічується і розмагнічується, що створює змінне магнітне поле. Це змінне поле взаємодіє з зовнішнім магнітним полем Землі, що призводить до виникнення вторинного струму в котушках, який можна виміряти і інтерпретувати для визначення величини і напрямку зовнішнього магнітного поля;

2. Галієві магнетометри: ці сенсори використовують ефект Холла, коли магнітне поле змушує електрони всередині провідника відхилитися в бік, створюючи різницю потенціалів, яка пропорційна силі магнітного поля. Галієві магнетометри точні і стабільні, що робить їх ідеальними для багатьох застосувань;

3. Магніторезистивні сенсори: використовують матеріали, опір яких змінюється під впливом магнітного поля. Зміна опору вимірюється і використовується для визначення інтенсивності магнітного поля;

4. Оптичні магнетометри: ці пристрої використовують властивості світла і його взаємодію з магнітними полями для вимірювання магнітного поля. Наприклад, вони можуть використовувати зміну поляризації світла при проходженні через магнітно-чутливі матеріали.

магнетометри вимірюють вектор магнітного поля у трьох вимірах, зазвичай по осях X, Y та Z. Це дозволяє їм визначити, де знаходиться північний полюс відносно об'єкта. Ця інформація є особливо корисною для калібрування гіроскопів та акселерометрів, оскільки вона надає додаткову орієнтирну точку, яка дозволяє коригувати похибки, що можуть виникати внаслідок дрейфу або інших неточностей.

У процесі калібрування ІМУ, дані з магнітометрів можуть бути використані для корекції даних з гіроскопів і акселерометрів. Наприклад, гіроскопи можуть зазнавати дрейфу, тобто поступового накопичення похибок у вимірюваннях кутової швидкості, що призводить до неточних визначень орієнтації з часом. Магнітометри, які забезпечують стабільні вимірювання напрямку на північ,

можуть використовуватися для періодичної корекції орієнтації, що обчислюється гіроскопами, таким чином мінімізуючи вплив дрейфу.

Також, у разі використання ІМУ в динамічних середовищах, де прискорення можуть змінюватися різко і непередбачувано, магнетометри допомагають розрізнити зміни в орієнтації, викликані реальним рухом об'єкта, від артефактів вимірювання, що можуть виникати внаслідок зовнішніх впливів на акселерометри і гіроскопи.

Таким чином, магнетометри є невід'ємною частиною сучасних ІМУ, забезпечуючи точне визначення орієнтації відносно магнітного поля Землі і сприяючи підвищенню точності та надійності даних, отриманих з інших сенсорів ІМУ.

Разом ці сенсори дозволяють ІМУ забезпечувати точні вимірювання орієнтації, прискорення та швидкості об'єкта у просторі. Комбінація даних з акселерометрів, гіроскопів і магнетометрів дозволяє коригувати помилки та отримувати більш точні результати, що є критичним для багатьох застосувань, включаючи навігаційні системи, системи стабілізації та інші технології, де важлива точність вимірювань.

1.2 Принципи вимірювання

ІМУ використовують принципи інерції для визначення зміни положення та орієнтації. Акселерометри вимірюють зміни в швидкості, а гіроскопи – зміни в кутовій швидкості. Комбінуючи ці дані, можна визначити поточну орієнтацію та переміщення об'єкта.

Принципи вимірювання інерціальних вимірювальних одиниць (ІМУ) базуються на використанні трьох основних типів сенсорів: акселерометрів, гіроскопів і магнетометрів. Кожен з цих сенсорів вимірює певні фізичні параметри, які разом дозволяють точно визначити орієнтацію, прискорення та кутову швидкість об'єкта у просторі. Розглянемо принципи вимірювання для кожного з цих сенсорів більш детально.

Акселерометри вимірюють лінійне прискорення об'єкта у трьох взаємно перпендикулярних осях: X , Y та Z . Принцип роботи акселерометрів ґрунтується на вимірюванні зміщення маси під впливом прискорення.

Принцип маси та пружини.

У середині акселерометра розміщена маса, яка з'єднана з основою сенсора за допомогою пружних елементів. Коли об'єкт прискорюється, маса зміщується відносно своєї початкової позиції через інерційні сили. Це зміщення фіксується за допомогою різних методів, таких як:

- П'єзоелектричний ефект.

Зміщення маси створює механічне напруження в п'єзоелектричному матеріалі, який генерує електричний заряд пропорційний до величини прискорення.

- Ємнісний метод.

Зміщення маси змінює відстань між пластинами конденсатора, змінюючи його ємність. Ця зміна ємності вимірюється і використовується для визначення прискорення.

Гіроскопи вимірюють кутову швидкість об'єкта, тобто швидкість обертання навколо його осей. Основний принцип роботи гіроскопів часто базується на ефекті Коріоліса.

Ефект Коріоліса.

Всередині гіроскопа розташовані рухомі елементи, які вібрують у певному напрямку. При обертанні гіроскопа навколо будь-якої осі, виникає сила Коріоліса (рис.1.4), яка спричиняє відхилення цих елементів перпендикулярно до напрямку їх початкового руху і осі обертання .

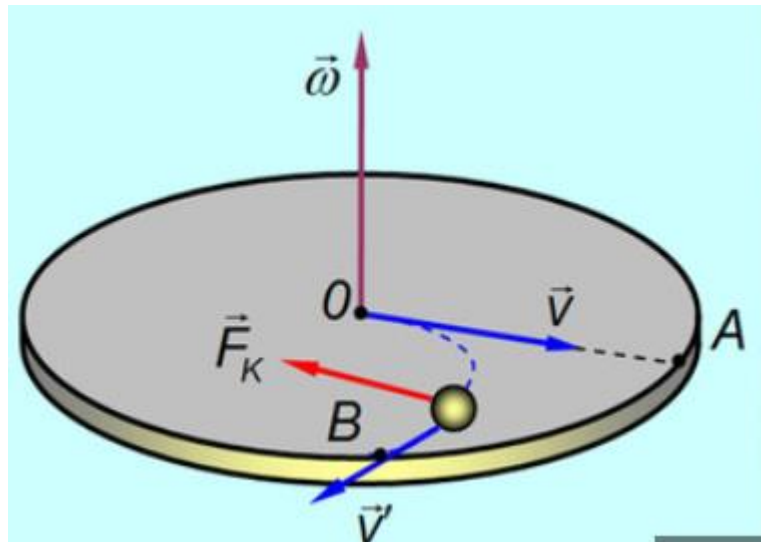


Рисунок 1.4 – Сила Кориоліса

Це відхилення фіксується за допомогою різних методів:

- MEMS-гіроскопи.

Використовують мініатюрні структури, що коливаються. Зміни в коливаннях під впливом ефекту Кориоліса вимірюються за допомогою сенсорів, що визначають зміщення або зміни у ємності конденсаторів.

- Оптичні гіроскопи.

Використовують інтерферометричні методи для вимірювання фазового зсуву світлових хвиль, що проходять через кільцевий інтерферометр під впливом обертання.

Магнетометри визначають орієнтацію об'єкта відносно магнітного поля Землі, вимірюючи вектор магнітного поля у трьох вимірах.

Принцип вимірювання магнітного поля.

Магнетометри використовують властивості певних матеріалів для реакції на магнітні поля. Залежно від типу магнетометра, можуть використовуватися різні методи:

- Флюкстейти.

Використовують магнітну проникність матеріалів для створення змінного магнітного поля, яке взаємодіє із зовнішнім магнітним полем Землі, створюючи вторинний струм, що вимірюється[16].

– Галієві магнетометри.

Використовують ефект Холла для вимірювання різниці потенціалів, спричиненої магнітним полем.

– Магніторезистивні сенсори.

Використовують матеріали, опір яких змінюється під впливом магнітного поля.

– Оптичні магнетометри.

Використовують зміни в поляризації світла при проходженні через магнітно-чутливі матеріали.

Дані з акселерометрів, гіроскопів і магнетометрів обробляються разом для отримання точної інформації про орієнтацію, прискорення та кутову швидкість об'єкта. Використовуючи алгоритми злиття сенсорних даних, такі як фільтр Калмана або комплементарні фільтри, можна зменшити похибки та покращити точність вимірювань. Це дозволяє ІМУ забезпечувати надійні та точні дані для навігації, стабілізації та управління рухом у різних технологічних додатках.

1.3 Використання ІМУ в різних галузях

ІМУ використовуються в навігаційних системах, системах стабілізації, мобільних пристроях, ігрових контролерах, медичних приладах та багатьох інших технологіях.

Інерціальні вимірювальні одиниці (ІМУ) знаходять широке застосування у багатьох галузях завдяки своїм здатностям точно визначати орієнтацію, прискорення та кутову швидкість об'єктів (табл. 1.2)[2].

Таблиця 1.2 – Застосування ІМУ в різних галузях

Галузь	Застосування
Авіація та космонавтика	Навігація та стабілізація польотів, гібридні навігаційні системи
Автомобільна промисловість	Системи активної безпеки, автономні транспортні засоби

Галузь	Застосування
Робототехніка	Контроль руху та стабілізація, навігація
Спортивні та фітнес додатки	Вимірювання руху, аналіз продуктивності
Медицина	Реабілітація, моніторинг здоров'я
Віртуальна та доповнена реальність	Трекінг руху, інтерфейси користувача
Морська навігація	Контроль суден, автономні морські платформи
Геодезія та картографія	Системи лазерного сканування, дронове картографування

Авіаційна промисловість.

У авіаційній індустрії ІМУ використовуються в авіоніки для навігації, стабілізації та управління літальними апаратами. Вони допомагають визначати орієнтацію літака, вимірюють прискорення та кутову швидкість, що є важливими для безпеки польоту та точності навігації.

Автомобільна промисловість.

У автомобільній промисловості ІМУ використовуються для систем стабілізації, антиблокувальних систем (ABS), систем контролю та стабілізації кермового управління. Вони допомагають визначати нахил автомобіля, кутову швидкість руху та інші параметри, що покращують безпеку та ефективність автотранспорту.

Медична промисловість.

У медичній сфері ІМУ використовуються для моніторингу рухів пацієнтів, реабілітації, діагностики та лікування різних захворювань, таких як хвороба Паркінсона або травми мозку. Вони можуть допомагати відслідковувати рухи та вимірювати різні параметри, що дозволяє медичним працівникам отримувати об'єктивні дані для діагностики та лікування.

Віртуальна реальність та ігрова індустрія.

У віртуальній реальності та в ігровій індустрії ІМУ використовуються для відслідковування рухів користувачів. Вони дозволяють переносити реальні рухи у віртуальний світ, створюючи більш іммерсивний досвід для користувачів віртуальної реальності або ігор.

Промислова автоматизація.

У промисловій автоматизації ІМУ використовуються для контролю за рухом промислових роботів та машин. Вони допомагають забезпечити точність та ефективність роботи, контролюючи їх рухи та орієнтацію в просторі.

У кожній з цих галузей використання ІМУ дозволяє підвищити точність, надійність та ефективність систем, що використовують ці дані для навігації, контролю та аналізу рухів об'єктів.

Висновки до розділу 1

У цьому розділі було розглянуто основні компоненти інерціальних вимірювальних одиниць (ІМУ). Ці компоненти включають акселерометри, гіроскопи та магнетометри, які вимірюють прискорення, кутову швидкість та магнітне поле відповідно. Кожен з цих сенсорів грає важливу роль у визначенні орієнтації, руху та положення об'єкта в просторі. Вони працюють синергічно, надаючи детальну інформацію про динаміку об'єкта.

Принципи вимірювання ІМУ-сенсорів базуються на різних фізичних принципах, таких як принцип маси для акселерометрів, ефект Саймона-Коріоліса для гіроскопів та взаємодія з магнітним полем для магнетометрів. Ці принципи дозволяють сенсорам точно вимірювати параметри руху об'єкта і визначати його орієнтацію у просторі.

ІМУ-сенсори знаходять застосування в різних галузях, від авіації до медицини. Вони використовуються для навігації та стабілізації повітряних та морських суден, автомобілів, роботів та багатьох інших систем. Також вони є важливими компонентами в спортивних трекерах, віртуальній реальності,

медичних пристроях та інших пристроях, які потребують точного визначення руху та орієнтації.

Кожен з цих пунктів важливий для розуміння принципів роботи та застосування ІМУ-сенсорів. Їх комбінація відкриває широкі можливості в різних галузях технологій та промисловості, що дозволяє створювати більш точні, ефективні та інноваційні системи.

Основні компоненти ІМУ, такі як акселерометри, гіроскопи та магнетометри, виявляються ключовими для визначення руху та орієнтації об'єктів у просторі. Принципи вимірювання, на яких базуються ці сенсори, дозволяють їм точно визначати параметри руху та орієнтації, що є критичним для багатьох сучасних систем.

За допомогою ІМУ-сенсорів можна вирішити різноманітні завдання у різних галузях, від авіації та автомобілебудування до медицини та віртуальної реальності. Їхнє використання сприяє покращенню навігації, стабілізації, моніторингу руху та багатьом іншим сферам.

Описані принципи та застосування ІМУ-сенсорів відкривають широкі можливості для подальшого розвитку технологій та інновацій у великій кількості галузей.

2 АПАРАТНО-ПРОГРАМНИЙ МОДУЛЬ ДЛЯ ТЕСТУВАННЯ ІМУ

2.1 Огляд апаратних компонентів

Апаратно-програмний модуль для тестування інерціальних вимірювальних одиниць (ІМУ) складається з різноманітних апаратних компонентів, які використовуються для забезпечення правильного тестування та аналізу роботи ІМУ-сенсорів. Ось огляд основних апаратних компонентів цього модуля:

1. Мікроконтролер або мікропроцесор.

Мікроконтролер – це головний інтелектуальний центр апаратного забезпечення, що відповідає за керування всією системою тестування ІМУ. Він виконує програмне забезпечення, яке координує зчитування, обробку та передачу даних з сенсорів ІМУ, а також керує взаємодією з користувачем і вхідно-вихідними пристроями. Основні функції мікроконтролера включають збір та обробку даних, керування вхідно-вихідними інтерфейсами, взаємодію з користувачем та керування енергоспоживанням. Він є ключовим компонентом, що забезпечує ефективну та надійну роботу системи тестування ІМУ [8, с.78].

2. ІМУ-сенсори.

Акселерометри, гіроскопи та магнетометри становлять основу інерціальної вимірювальної одиниці (ІМУ) і вимірюють різні параметри руху об'єкта.

Акселерометри визначають лінійне прискорення в трьох взаємно перпендикулярних осях (X, Y, Z), використовуючи принцип маси, що зміщується. Гіроскопи вимірюють кутову швидкість об'єкта навколо його осей, використовуючи ефект Саймона-Коріоліса. Магнетометри визначають орієнтацію об'єкта в магнітному полі Землі, вимірюючи вектор магнітного поля у трьох вимірах.

Ці сенсори забезпечують важливі дані для аналізу роботи ІМУ, дозволяючи визначити орієнтацію, швидкість та прискорення об'єкта. Вони є ключовими компонентами в широкому спектрі застосувань, від автономних транспортних засобів до віртуальної реальності.

3. Інтерфейси зв'язку.

Для передачі даних між модулем тестування та інерціальними вимірювальними одиницями (ІМУ) використовуються різні інтерфейси, які забезпечують ефективну комунікацію та передачу інформації. Ось детальніше про кожен з цих інтерфейсів [14]:

- I2C (Inter-Integrated Circuit).

Цей інтерфейс дозволяє підключати декілька пристроїв до одного шини даних, що робить його популярним для використання з ІМУ та іншими сенсорами. Він використовує дві шини – одну для передачі даних (SDA) та одну для синхронізації (SCL).

- SPI (Serial Peripheral Interface).

Цей інтерфейс є швидким та дозволяє передавати дані в повному дуплексному режимі, що означає, що дані можуть бути передані в обох напрямках одночасно. Він використовує чотири проводи: MOSI (Master Out Slave In), MISO (Master In Slave Out), SCK (Serial Clock) та SS (Slave Select).

- UART (Universal Asynchronous Receiver/Transmitter).

Цей інтерфейс дозволяє передавати дані за допомогою асинхронного зв'язку, що означає, що не потрібно спільного тактового сигналу. Він використовує два проводи: один для передачі даних (TX) та один для прийому (RX).

- USB (Universal Serial Bus).

Цей інтерфейс є швидким та дозволяє передавати великі обсяги даних. Він часто використовується для підключення ІМУ та модуля тестування до комп'ютера або іншого зовнішнього пристрою.

Кожен з цих інтерфейсів має свої переваги та недоліки, і вибір конкретного залежить від потреб конкретної системи та умов використання.

4. Інтерфейси візуалізації даних.

Для зручного відображення та аналізу даних з інерціальних вимірювальних одиниць (IMU) можна використовувати різноманітні інтерфейси візуалізації. Ось деякі з них:

- LCD-дисплеї.

Ці дисплеї можуть відображати дані безпосередньо на екрані, що дозволяє оператору швидко отримати інформацію про рух об'єкта. Вони можуть використовуватися для відображення основних параметрів, таких як орієнтація, швидкість та прискорення (рис.2.1.)



Рисунок 2.1 – LCD-дисплей

- LED-індикатори.

Ці індикатори можуть бути використані для відображення простих статусів або значень. Наприклад, різні світлодіоди можуть світитися в залежності від значень, виміряних сенсорами IMU, що дозволяє оператору швидко оцінити ситуацію (рис.2.2.)



Рисунок 2.2 – LED-індикатори

3. Графічні інтерфейси на комп'ютері.

Ці інтерфейси дозволяють відобразити та аналізувати дані у вигляді графіків, діаграм або числових значень на екрані комп'ютера. Вони можуть надавати більшу гнучкість та можливості для детального аналізу даних. Вибір конкретного інтерфейсу візуалізації залежить від потреб користувача, умов використання та доступних ресурсів.

5. Мобільні застосунки.

Застосунки для смартфонів та планшетів можуть використовуватися для відображення та аналізу даних з IMU у реальному часі. Вони можуть надавати зручний спосіб відстежувати рух та виконувати аналіз за допомогою вбудованих функцій та графіків.

6. Живлення.

Для правильної роботи модуля тестування IMU потрібне надійне джерело живлення, яке забезпечить енергію всім компонентам системи. Ось деталізація можливих варіантів джерел живлення:

- Внутрішнє джерело живлення.

Модуль може мати вбудовану батарею або акумулятор, який забезпечує живлення всім електронним компонентам. Це забезпечує мобільність та незалежність від зовнішнього джерела живлення.

- Зовнішнє джерело живлення.

Модуль може використовувати зовнішні джерела живлення, такі як батареї або адаптери живлення, підключені до мережі електроживлення. Це може бути корисним у випадках, коли потрібна тривала робота модуля без перерви на зарядку.

Обираючи джерело живлення, необхідно враховувати вимоги щодо тривалості роботи, портативності, вартості та надійності. Крім того, важливо враховувати енергоефективність всіх компонентів модуля тестування, щоб максимізувати тривалість роботи від обраного джерела живлення. Корпус та кріплення.

Для забезпечення захисту та зручності використання модуль тестування може бути розміщений у відповідному корпусі з різними можливостями кріплення, що дозволяє використовувати його в різних умовах.

Ці компоненти у поєднанні дозволяють створити функціональний та ефективний апаратно-програмний модуль для тестування та аналізу роботи ІМУ-сенсорів, який може бути використаний у різних сферах, включаючи авіацію, автомобільну промисловість, медицину та ігрову індустрію (табл. 2.1).

Таблиця 2.1 – Рекомендовані параметри модуля тестування ІМУ-сенсорів [10].

Характеристика	Рекомендації
ІМУ-сенсори	
Діапазони вимірювання	Акселерометри: $\pm 16g$, Гіроскопи: ± 2000 dps, магнетометри: ± 8 Gauss
Роздільна здатність	16 біт для кожного сенсора
Частота оновлення даних	Принаймні 100 Гц
Мікроконтролер або мікропроцесор	
Швидкість роботи	Мінімум 72 МГц
Пам'ять	Принаймні 256 кБ флеш-пам'яті
Вхідно-вихідні порти	Принаймні 10 цифрових та 4 аналогових порти

Інтерфейси комунікації	
Підтримка	SPI та I2C
Максимальна швидкість передачі даних	400 кбіт/с для I2C, 10 Мбіт/с для SPI
Живлення	
Напруга живлення	3.3 В або 5 В
Потужність	Принаймні 500 мА
Тип живлення	Можливість живлення від USB або акумулятора літій-іонного або літій-полімерного типу
Програмне забезпечення	
Підтримка алгоритмів фільтрації даних	Так, наприклад, фільтр Калмана
Формати збереження даних	CSV, JSON
Візуалізація результатів	Графіки, треки руху
Інтерфейс користувача	
Графічний інтерфейс	Так
Можливість вибору параметрів та відображення результатів у реальному часі	Так
Налаштування параметрів тестування	Так, наприклад, частота дискретизації даних, тип фільтрації

Оптимальні параметри для апаратно-програмного модуля тестування IMU-сенсорів грають критичну роль у забезпеченні ефективності та точності тестування. Розглянемо роль кожного з них:

- IMU-сенсори – є основою для отримання даних про рух та орієнтацію об'єкта. Оптимальний вибір IMU-сенсорів полягає в їхній точності, чутливості та швидкості вимірювання, щоб забезпечити достовірні дані для подальшого аналізу;

- мікроконтролер – відповідає за збір та обробку даних з сенсорів. Вибір оптимального мікроконтролера полягає у його потужності обчислення, швидкості реакції та можливостях взаємодії з іншими компонентами;
- інтерфейси зв'язку – для передачі даних з сенсорів до модуля тестування необхідні різні інтерфейси, такі як I2C, SPI, UART або USB. Оптимальний вибір інтерфейсу полягає у його швидкості передачі даних, надійності та сумісності з обраними сенсорами;
- джерело живлення – це необхідно для живлення всіх компонентів модуля. Оптимальний вибір джерела живлення полягає в його потужності, тривалості роботи та можливостях заряду;
- програмне забезпечення – відповідає за керування процесом тестування, обробку та аналіз отриманих даних. Оптимальне програмне забезпечення має включати ефективні алгоритми фільтрації даних, зручний інтерфейс користувача та можливість збереження та відображення результатів тестування.

Аналіз та правильний підбір цих параметрів допоможе створити ефективний та надійний апаратно-програмний модуль для тестування IMU-сенсорів, забезпечуючи якість та точність вимірювань.

2.2 Мікроконтролер STM32F407

Мікроконтролери STMicroelectronics є наразі чи не найпоширенішими серед всіх виробників мікроконтролерів, що пояснюється їх продуктивністю, зручністю програмних інтерфейсів, великим набором апаратної периферії. Це дозволяє використовувати їх як для простих проєктів, так і для складних, де передбачається взаємодія великої кількості компонентів.

Можна відзначити, що мікроконтролери розділяються на декілька серій, серед яких можна виділити:

- мікроконтролери з низьким споживанням - серія L;
- високопродуктивні мікроконтролери - серія F.

Мікроконтролер STM32F407 є одним з найпоширеніших завдяки великій популярності плати розробника STM32F4 Discovery. При цьому він характеризується як великими можливостями щодо організації обчислень, включаючи апаратне прискорення операцій цифрової обробки сигналів, так і можливості по роботі в режимах низького споживання. Також особливістю цього контролеру є наявність апаратних засобів для роботи з мультимедіа. З точки зору обробки даних з датчиків та організації взаємодії з ними цей мікроконтролер є чи не оптимальним варіантом, тому саме він буде використовуватись при реалізації демонстраційного варіанту модулю тестування сенсорів.

2.3 9-осьовий датчик LSM9DS

Датчик LSM9DS є ідеальним варіантом для того, щоб продемонструвати організацію роботи модуля тестування. Цей датчик є 9-осьовим IMU, відповідно містить акселерометр, гіроскоп та магнетометр, що є ідеальним варіантом, який дозволить перевірити коректність реалізацій для всіх 3 типів датчиків, які можуть входити до IMU. Три пристрої доступні за допомогою як інтерфейсу SPI, так і I2C. Розділення з точки зору логіки доступу відбувається на основі того, що пристрої мають різні адреси і при цьому розміщені на одній шині. Оскільки для організації взаємодії у такому випадку для I2C треба менше пінів, то саме цей інтерфейс мікроконтролеру обраний для того, щоб реалізувати взаємодію з датчиком.

З точки зору характеристик, то цей датчик пропонує такі початкові засоби для вимірювань, які повністю задовольняють таким, які вказані у таблиці 2.1. Так, наприклад, діапазон вимірювання прискорення складає від -2 до $+16$ g. Завдяки цьому забезпечується необхідна точність для різних застосувань. Діапазони вимірювань визначаються станом внутрішніх регістрів відповідного пристрою. Усі пристрою підтримують як видачу по перериванню, так і у режимі постійного опитування.

Саме датчик LSM9DS буде використовуватись як варіант для демонстрації з використанням доступної плати GlobalLogic Embedded Starter Kit.

2.4 Огляд програмного забезпечення

Програмне забезпечення для апаратно-програмного модуля тестування ІМУ відіграє ключову роль у зборі, обробці та візуалізації даних з сенсорів ІМУ.

Розглянемо основні компоненти програмного забезпечення для цього модуля:

1. Драйвери сенсорів.

Драйвери сенсорів є ключовою складовою програмного забезпечення модуля тестування ІМУ, оскільки вони забезпечують взаємодію з різними типами сенсорів ІМУ через різні інтерфейси комунікації, такі як I2C, SPI або UART. Детально розглянемо їх функціональність та роль [3]:

- драйвери забезпечують можливість взаємодії з різними сенсорами ІМУ, включаючи акселерометри, гіроскопи та магнетометри. Вони забезпечують простий та зручний доступ до даних, що надходять від цих сенсорів;

- драйвери реалізують протоколи комунікації з сенсорами, такі як I2C (Inter-Integrated Circuit), SPI (Serial Peripheral Interface) або UART (Universal Asynchronous Receiver-Transmitter), залежно від того, який інтерфейс підтримується сенсором. Це дозволяє модулю тестування взаємодіяти з широким спектром сенсорів;

- драйвери забезпечують можливість зчитувати дані з сенсорів та передавати їх до основної програми для подальшої обробки. Це включає в себе зчитування реєстрів сенсорів, встановлення параметрів роботи, а також організацію передачі даних через вказаний інтерфейс;

- деякі драйвери надають можливість налаштування параметрів роботи сенсорів, таких як частота зчитування, діапазони вимірювання та фільтрація даних. Це дозволяє оптимізувати роботу сенсорів для конкретних вимог застосування;

- драйвери зазвичай містять механізми обробки помилок та винятків, щоб впевнитися, що вони працюють надійно в усіх умовах експлуатації. Це

включає в себе виявлення та вирішення можливих проблем зі зчитуванням даних, втратою зв'язку або некоректною конфігурацією сенсорів;

– драйвери розробляються з урахуванням взаємодії з високорівневим програмним забезпеченням модуля тестування ІМУ, що дозволяє забезпечити зручний та безпроблемний обмін даними між різними компонентами системи. Алгоритми фільтрації даних.

Для підвищення точності та надійності вимірювань можуть бути використані різні алгоритми фільтрації даних, такі як фільтр Калмана або комбінації комплементарних фільтрів. Ці алгоритми дозволяють зменшити шуми та похибки вимірювань.

2. Модуль визначення орієнтації.

Модуль для визначення орієнтації об'єкта на основі даних з гіроскопів, акселерометрів та магнетометрів є ключовим компонентом програмного забезпечення для тестування ІМУ.

Першим кроком є зчитування даних з гіроскопів, акселерометрів та магнетометрів. Ці дані включають в себе значення кутової швидкості, лінійного прискорення та вектора магнітного поля.

Отримані дані піддаються обробці для видалення шумів та нормалізації значень. Наприклад, можуть використовуватися фільтри для зменшення впливу шумів на остаточні результати.

Для визначення орієнтації об'єкта в просторі можуть застосовуватися різні методи, такі як використання кватерніонів або ейлерових кутів. Кватерніони є особливо ефективним методом для визначення орієнтації, оскільки вони забезпечують унікальне представлення орієнтації у тривимірному просторі.

Після визначення орієнтації може бути проведена калібрування та компенсація систематичних помилок, які можуть бути присутні в сенсорах ІМУ. Це дозволяє поліпшити точність та надійність вимірювань.

Остаточні результати можуть бути візуалізовані за допомогою графіків або тривимірних моделей, що демонструють орієнтацію об'єкта в просторі.

Цей модуль грає важливу роль у процесі тестування ІМУ, дозволяючи аналізувати та оцінювати роботу сенсорів з точки зору їхньої здатності визначати орієнтацію об'єкта.

3. Інтерфейс користувача.

Інтерфейс взаємодії з користувачем у програмному забезпеченні для тестування ІМУ є важливим елементом, який спрощує взаємодію з програмою та забезпечує зручний доступ до функцій тестування та аналізу результатів [11].

Інтерфейс дозволяє користувачеві налаштовувати параметри тестування, такі як час зчитування даних, типи сенсорів, що тестуються, чи інші параметри, що впливають на процес збору та аналізу даних.

Користувач може візуально спостерігати за даними, що збираються з сенсорів ІМУ, через інтерфейс програми. Це може включати відображення графіків, таблиць або тривимірних моделей, які демонструють орієнтацію об'єкта в просторі.

Інтерфейс дозволяє користувачеві виконувати аналіз отриманих результатів тестування, включаючи виявлення потенційних проблем, оцінку точності та надійності сенсорів, а також визначення шляхів для подальшої оптимізації.

Інтерфейс може бути реалізований у різних формах, включаючи консольні додатки, графічні інтерфейси користувача (GUI) або веб-додатки. Вибір конкретного типу інтерфейсу може залежати від потреб користувачів та характеру програмного забезпечення.

Інтерфейс користувача сприяє зручній та ефективній взаємодії з програмним забезпеченням для тестування ІМУ, що дозволяє користувачеві ефективно керувати процесом тестування та аналізувати результати з точки зору їхньої точності та надійності.

4. Модуль візуалізації даних.

Модуль візуалізації даних у програмному забезпеченні для тестування ІМУ відіграє ключову роль у відображенні результатів тестування, що допомагає зрозуміти та проаналізувати отримані дані.

- цей модуль дозволяє відображати графіки, що відображають залежності параметрів, виміряних сенсорами ІМУ, від часу або інших факторів. Наприклад, може бути відображена динаміка зміни прискорення, кутової швидкості або магнітного поля в залежності від часу;
- цей модуль дозволяє відображати траєкторії руху об'єкта в просторі на основі даних, отриманих від сенсорів ІМУ. Це дозволяє візуально зрозуміти рух об'єкта та виявити можливі аномалії чи неправильності в роботі сенсорів;
- модуль може також включати інші форми візуалізації, такі як тривимірні моделі об'єктів, хмари точок або теплові карти, що дозволяють більш глибоко проаналізувати дані та зрозуміти їхню природу;
- модуль візуалізації може мати функціонал для взаємодії з графіками, наприклад, масштабування, збільшення або зменшення області відображення, а також можливості налаштування параметрів візуалізації для кращого розуміння даних.

Модуль візуалізації даних є важливою складовою програмного забезпечення для тестування ІМУ, яка допомагає користувачеві ефективно аналізувати та розуміти отримані результати.

5. Модуль збереження даних.

Такий модуль для збереження отриманих даних у файлову систему є важливою складовою програмного забезпечення для тестування ІМУ. Розглянемо детальніше його функціональні можливості [10]:

- модуль може підтримувати різні формати збереження даних, такі як текстові файли, CSV (Comma-Separated Values), JSON (JavaScript Object Notation), HDF5 (Hierarchical Data Format version 5) або спеціалізовані формати для збереження даних сенсорів;
- користувач може налаштувати параметри збереження даних, такі як ім'я файлу, шлях збереження, частота збереження, формат файлу тощо. Це дозволяє адаптувати збереження до конкретних вимог та потреб користувача;
- крім основних даних з сенсорів ІМУ, модуль може також зберігати метадані, такі як дата та час отримання даних, параметри налаштування сенсорів,

інформація про тестування тощо. Це дозволяє зберегти повну інформацію про тестування для подальшого аналізу та відтворення результатів;

- модуль може забезпечувати захист даних, наприклад, шляхом шифрування файлів або забезпечення доступу до збережених даних лише авторизованим користувачам. Це важливо для збереження конфіденційності та цілісності даних;

- модуль може підтримувати інтеграцію з іншими інструментами аналізу даних, такими як MATLAB, Python (з використанням бібліотек, таких як Pandas або NumPy), або спеціалізовані ПЗ для обробки даних сенсорів. Це дозволяє зручно і ефективно використовувати зібрані дані для подальшого аналізу та використання.

Ці компоненти програмного забезпечення спільно з апаратними компонентами створюють повноцінний апаратно-програмний модуль для тестування та аналізу роботи IMU-сенсорів.

2.5 Середовище розробки STM32Cube IDE

Також варто звернути увагу на засоби розробки для мікроконтролерів, зокрема, для STM32, а саме STM32Cube IDE. Це середовище пропонує підхід, який базується на генерації коду ініціалізації периферії, та подальшому доповненні проєкту необхідним функціоналом. Це значно спрощує розробку стандартної частини та дозволяє уникнути можливих помилок при написанні такого коду, оскільки середовище-конфігуратор перевіряє узгодженість відповідних налаштувань та за умови конфліктів повідомлятиме розробника про помилку, а також не дозволить провести генерацію проєкту з вказаними налаштуваннями. Обране середовище, а також супутні програмні засоби відносяться до категорії таких, що доступні як безкоштовне програмне забезпечення.

Середовище STM32Cube IDE також вимагає дотримання певних правил організації коду, що також заохочує розробника до ефективного використання бібліотеки Hardware Abstract Library (HAL) для роботи з обраним мікроконтролером. Це стосується реалізації обробників переривань, функцій зворотнього виклику та послідовності використання функцій у бібліотеці HAL.

Передбачається робота з мовами програмування C або C++, для чого доступні відповідні засоби розробки. Мова C дозволяє ефективно керувати ресурсами мікроконтролера, тому для розробки програмного забезпечення для мікроконтролера (firmware) обрана саме ця мова.

При написанні програмного забезпечення для мікроконтролеру STM32F4 для тестування ІМУ використовуватиметься саме це середовище.

2.6 Відлагоджувальна плата GlobalLogic Embedded Starter Kit

Відлагоджувальна плата GlobalLogic Embedded Starter Kit може застосовуватися для організації в якості модуля тестування ІМУ-сенсорів. Вона містить усі необхідні перелічені компоненти для організації такого модуля. До складу набору входить:

- плата STM32F4 Discovery з мікроконтролером STM32F4;
- плата розширення, яка містить у т.ч. ІМУ-модуль STM LSM9DS, а також такі допоміжні засоби як дисплей, кнопки, перемикачі.

Основною перевагою використання цієї плати з метою розробки модуля тестування є доступність всіх необхідних для цього компонентів.

2.7 Взаємодія апаратного та програмного модулів

Інерціальні вимірювальні одиниці (ІМУ) є важливими компонентами у багатьох сучасних системах, які вимірюють та контролюють рух об'єктів у просторі. Для забезпечення правильного функціонування та надійності таких систем, необхідно проводити тестування ІМУ-сенсорів. У даній роботі детально розглянуто апаратно-програмний модуль для тестування ІМУ, який складається з апаратних компонентів, програмного забезпечення та їх взаємодії. Описано процес зчитування, обробки та аналізу даних, отриманих від сенсорів ІМУ, а також визначено практичне значення цього модуля для різних галузей та додаткові можливості його використання.

Зв'язок між апаратним та програмним модулями для тестування інерціальних вимірювальних одиниць (ІМУ) відбувається поетапно, починаючи з отримання даних від сенсорів до аналізу та відображення результатів тестування.

Апаратний модуль періодично зчитує дані з різних сенсорів ІМУ, таких як акселерометри, гіроскопи та магнетометри. Ці дані представлені у вигляді числових значень, які відображають прискорення, кутову швидкість та магнітне поле.

Отримані дані можуть бути попередньо оброблені на мікроконтролері. Це включає в себе фільтрацію шумів, усереднення значень та конвертацію формату даних для подальшої передачі.

Оброблені дані надсилаються до програмного модуля через відповідний інтерфейс зв'язку, такий як USB або UART. Цей процес передачі дозволяє програмному модулю отримати актуальні дані для подальшого аналізу[3].

Прийняті дані обробляються програмним модулем, який може виконувати різні операції, такі як фільтрація, визначення орієнтації, аналіз стабільності тощо. Це допомагає виявити та коригувати будь-які помилки чи похибки в даних.

Оброблені дані відображаються на інтерфейсі користувача для подальшого аналізу. Це може включати відображення графіків, треків руху об'єкта, числових значень тощо. Користувач може аналізувати ці дані для оцінки роботи ІМУ-сенсорів.

Користувач може налаштовувати параметри тестування через інтерфейс програмного модуля, такі як час збору даних, частота оновлення тощо. Це дозволяє забезпечити гнучкість та налагодження під конкретні потреби тестування.

Після завершення тестування програмний модуль може зберегти результати у внутрішню або зовнішню пам'ять для подальшого аналізу або звітування. Це дозволяє зберегти дані для майбутнього використання або порівняння з іншими результатами.

Розробка та використання апаратно-програмного модуля для тестування ІМУ відіграє важливу роль у забезпеченні якості та надійності систем, що використовують ці сенсори. Цей модуль дозволяє ефективно зчитувати, обробляти

та аналізувати дані з різних сенсорів ІМУ, що важливо для виявлення та корекції будь-яких похибок чи аномалій у їх роботі. Крім того, він забезпечує можливість налаштування параметрів тестування та зручний інтерфейс для взаємодії з користувачем. Результати такого тестування допомагають покращити якість та надійність ІМУ-сенсорів у широкому спектрі застосувань, що відкриває шлях до подальшого розвитку технологій, що використовують ці системи.

Висновки до розділу 2

Апаратні компоненти для модуля тестування ІМУ включають в себе самі сенсори ІМУ (акселерометри, гіроскопи, магнетометри), мікроконтролер для управління та збір даних, а також можливо додаткові елементи, які можуть бути потрібні для живлення, зберігання даних тощо. Правильний вибір та налаштування цих компонентів є ключовим для створення ефективного та надійного модуля.

Програмне забезпечення для модуля тестування ІМУ включає в себе програми для зчитування даних з сенсорів, обробки, аналізу та візуалізації цих даних. Воно грає важливу роль у керуванні процесом тестування та забезпеченні правильної обробки даних.

Взаємодія між апаратним та програмним модулями є ключовою для успішної роботи модуля тестування ІМУ. Програмне забезпечення керує процесом зчитування даних з сенсорів, їх обробки та аналізу, використовуючи можливості, що надаються апаратними компонентами.

Так, завершивши розділ про апаратно-програмний модуль для тестування ІМУ, ми отримали загальне уявлення про необхідні компоненти та програмне забезпечення для створення ефективного та надійного модуля. Огляд апаратних компонентів надав нам інформацію про ключові елементи, такі як сенсори ІМУ та мікроконтролери, які необхідні для збору та обробки даних. Огляд програмного забезпечення розкрив важливість правильно написаних програм для ефективної роботи модуля. Нарешті, розділ про взаємодію апаратного та програмного модулів підкреслив важливість гармонійної співпраці між апаратними та програмними складовими для успішного функціонування системи тестування ІМУ.

3 МЕТОДИКА ТЕСТУВАННЯ ТА РЕЗУЛЬТАТИ

3.1 Процес тестування ІМУ-сенсорів

Процес тестування ІМУ-сенсорів включає кілька етапів, таких як калібрування сенсорів, збір даних у різних умовах та аналіз отриманих даних. Нижче наведено кожен етап з відповідними кодовими фрагментами та прикладами:

1. Калібрування сенсорів:

Калібрування сенсорів є важливим етапом для забезпечення точності та надійності вимірювань. Для калібрування акселерометра можна використовувати метод гравітаційного впливу, а гіроскоп може бути покалібрований шляхом вимірювання статичних кутових швидкостей. Ось приклад коду для калібрування акселерометра за допомогою бібліотеки `numpy` у Python (рис.3.1.):

```
1 python
2 import numpy as np
3 def calibrate_accelerometer(data):
4     bias = np.mean(data, axis=0)
5     calibrated_data = data - bias
6     return calibrated_data
```

Рисунок 3.1 – Калібрування акселерометра

Цей код призначений для калібрування даних, отриманих з акселерометра. Акселерометр вимірює прискорення, включаючи гравітаційне прискорення, а також будь-яке інше лінійне прискорення, яке діє на пристрій. Однак, в процесі вимірювання може виникнути систематична помилка або "похибка зсуву", коли сенсори не повністю відкалібровані або не налаштовані правильно.

Цей код вирішує цю проблему, проводячи калібрування акселерометра. Калібрування полягає в визначенні "похибки зсуву" для кожного з трьох вимірюваних напрямків (осей X, Y та Z) і відніманні цієї похибки з виміряних значень. Це допомагає забезпечити, що нульове значення прискорення відповідає справжньому нульовому значенню, а не будь-якій систематичній помилці.

- `import numpy as np`: цей рядок імпортує бібліотеку ``numpy``, яка містить функції для роботи з масивами та математичними операціями у Python;
- `def calibrate_accelerometer(data)`: ця функція отримує на вхід дані з акселерометра у вигляді масиву ``data``;
- `bias = np.mean(data, axis=0)`: цей рядок обчислює середнє значення для кожного стовпчика масиву ``data``, що відповідає відповідному напрямку (осі) акселерометра. Це значення відповідає "похибці зсуву";
- `calibrated_data = data - bias`: цей рядок віднімає від кожного значення у масиві ``data`` відповідне значення "похибки зсуву". Це допомагає скоригувати дані так, щоб нульове значення відповідало справжньому нульовому значенню прискорення;
- `return calibrated_data`: ця функція повертає відкалібровані дані, які можна використовувати для подальших обчислень або аналізу.

Отже, цей код грає важливу роль у покращенні точності та надійності вимірювань акселерометра шляхом корекції систематичних помилок, що можуть виникнути в процесі його роботи.

2. Збір даних у різних умовах:

Для ефективного тестування необхідно збирати дані в різних умовах, таких як статичне положення, різні кути нахилу та обертання об'єкта. Ось приклад коду для збору даних з гіроскопу у реальному часі за допомогою мікроконтролера Arduino (рис.3.2.):

```
1  cpp
2  #include <Wire.h>
3  #include <Adafruit_Sensor.h>
4  #include <Adafruit_BNO055.h>
5  Adafruit_BNO055 bno = Adafruit_BNO055();
6  void setup() {
7      Serial.begin(9600);
8      if (!bno.begin()) {
9          Serial.println("Failed to initialize BNO055 sensor!");
10         while (1);
11     }
12 }
13 void loop() {
14     sensors_event_t event;
15     bno.getEvent(&event);
16     Serial.print("Gyro X: ");
17     Serial.print(event.gyro.x);
18     Serial.print(", Y: ");
19     Serial.print(event.gyro.y);
20     Serial.print(", Z: ");
21     Serial.println(event.gyro.z);
22     delay(100);
23 }
```

Рисунок 3.2 – Код для збору даних з гіроскопу

Код включає декілька бібліотек, таких як `Wire.h`, `Adafruit_Sensor.h` та `Adafruit_BNO055.h`, які необхідні для зчитування даних з гіроскопу через мікроконтролер Arduino.

У функції `setup()` виконується ініціалізація з'єднання з гіроскопом. Якщо ініціалізація не вдалася (наприклад, якщо гіроскоп не підключений або відсутній), виводиться повідомлення про помилку на моніторі шляхом виклику `Serial.println()` і програма завершується за допомогою `while(1)`.

У функції `loop()` виконується безперервний цикл, в якому зчитуються дані з гіроскопу за допомогою функції `bno.getEvent(&event)`. Після цього дані про кутову швидкість по трьом осях (`X`, `Y`, `Z`) виводяться на монітор шляхом виклику `Serial.print()` та `Serial.println()`. Додатково, є затримка `delay(100)` у 100 мілісекунд для стабілізації виведених даних.

Цей код дозволяє отримувати в реальному часі дані про кутову швидкість (гіроскоп) з гіроскопу, який підключений до мікроконтролера Arduino. Отримані дані можна використовувати для тестування ІМУ-сенсорів у різних умовах, таких як статичне положення, різні кути нахилу та обертання об'єкта.

3. Аналіз отриманих даних:

Після збору даних необхідно аналізувати їх для виявлення будь-яких аномалій або похибок. Це може включати в себе обчислення середніх значень, відхилень та інших статистичних параметрів. Наприклад, у Python можна використовувати бібліотеку `matplotlib` для візуалізації даних (рис.3.3.):

```
1 python
2 import matplotlib.pyplot as plt
3 # Припустимі дані
4 data = [1, 2, 3, 4, 5]
5 plt.plot(data)
6 plt.xlabel('Час')
7 plt.ylabel('Дані')
8 plt.title('Збір даних з гіроскопу')
9 plt.show()
```

Рисунок 3.3 – Використання бібліотеки “matplotlib”

Цей код у Python використовує бібліотеку `matplotlib` для візуалізації даних, зібраних з гіроскопу. Основна його роль полягає у відображенні графіка залежності даних від часу з використанням простого лінійного графіку.

– `import matplotlib.pyplot as plt`: цей рядок імпортує бібліотеку `matplotlib.pyplot`, яка дозволяє використовувати функції для створення графіків та візуалізації даних;

– `data = [1, 2, 3, 4, 5]`: визначається список `data`, що містить припустимі дані, які будуть відображені на графіку;

– `plt.plot(data)`: функція `plot()` використовується для побудови лінійного графіка на основі заданих даних. У цьому випадку графік буде побудований на основі значень зі списку `data`;

– `plt.xlabel('Час')` та `plt.ylabel('Дані')`: функції ``xlabel()`` та ``ylabel()`` використовуються для підписування осей графіка. У цьому випадку на вісь x буде нанесено підпис "Час", а на вісь y – "Дані";

– `plt.title('Збір даних з гіроскопу')`: функція ``title()`` використовується для надання заголовку графіку. У цьому випадку заголовок буде "Збір даних з гіроскопу";

– `plt.show()`: ця команда відображає створений графік у вікні.

Отже, цей код створює простий лінійний графік залежності даних від часу та надає основну візуалізацію зібраних даних з гіроскопу.

Ці кроки дозволяють ефективно проводити тестування ІМУ-сенсорів та аналізувати їхню роботу у різних умовах.

Розробка апаратно-програмного модуля для тестування ІМУ-сенсорів може бути реалізована наступним чином:

Апаратна частина:

1. Мікроконтролер або мікропроцесор.

Вибір мікроконтролера або мікропроцесора, який буде керувати апаратною частиною модуля та забезпечувати зв'язок з комп'ютером або іншими пристроями.

2. Інтерфейси зв'язку.

Додайте різні інтерфейси зв'язку, такі як USB, Bluetooth або Wi-Fi, для забезпечення можливості підключення до зовнішніх пристроїв для збереження даних або відображення результатів тестування.

3. Сенсори ІМУ.

Додайте сенсори ІМУ, які будуть тестуватися. Це можуть бути акселерометри, гіроскопи та магнетометри, які вимірюють прискорення, кутову швидкість та магнітне поле відповідно.

4. Інтерфейси для підключення сенсорів.

Забезпечте інтерфейси для підключення сенсорів ІМУ до модуля, наприклад, шину I2C або SPI, щоб забезпечити зручну взаємодію з різними типами сенсорів.

Програмна частина.

1. Функції керування модулем.

Розробіть програмне забезпечення для керування апаратною частиною модуля, таке як запуск тестів, збір даних з сенсорів та передача їх на комп'ютер для подальшої обробки.

2. Обробка даних.

Розробіть алгоритми для обробки даних, зібраних з сенсорів, такі як фільтрація шуму, калібрування та обчислення вимірюваних параметрів.

3. Візуалізація результатів.

Створіть програмне забезпечення для візуалізації результатів тестування, яке дозволить аналізувати дані у зручному форматі, наприклад, графіки, діаграми тощо.

4. Інтерфейс користувача.

Розробіть інтерфейс користувача, який дозволить користувачам легко керувати модулем та переглядати результати тестування.

5. Тестування та оптимізація.

Проведіть тестування модуля на різних типах сенсорів IMU, а потім оптимізуйте програмне забезпечення для підвищення швидкості та точності тестування.

Цей апаратно-програмний модуль дозволить проводити ефективно та надійно тестування IMU-сенсорів, забезпечуючи високу якість та точність результатів.

Розробимо просту програму для апаратно-програмного модуля тестування IMU-сенсорів на мові Python. Ця програма буде зчитувати дані з сенсорів, виводити їх на екран та зберігати у файл для подальшого аналізу. Для цього нам знадобиться бібліотека ``serial`` для зчитування даних з порту та бібліотека ``csv`` для запису даних у файл CSV. Попередньо встановимо ці бібліотеки за допомогою `pip`.

Ця програма відкриває з'єднання з COM-портом, зчитує дані з сенсорів, виводить їх на екран та записує у файл ``imu_data.csv``. Після закінчення зчитування вона закриває з'єднання з COM-портом та зчитує дані з файлу ``imu_data.csv`` за допомогою бібліотеки ``pandas`` для подальшого аналізу. Не забудьте змінити ``port`` на потрібний COM-порт у вашій системі.

Результати роботи цієї програми будуть залежати від даних, які будуть отримані з підключених ІМУ-сенсорів. Оскільки сама програма лише зчитує дані з порту і записує їх у файл, а також виводить на екран, то результати будуть виглядати наступним чином:

- дані зчитуються з сенсорів та відображаються на екрані у вигляді словників, де кожен ключ представляє вимірювану величину (наприклад, прискорення по осях X, Y, Z, кутова швидкість по осях X, Y, Z тощо);
- кожен рядок у файлі `imu_data.csv` буде містити дані з одного вимірювання. Кожен стовпець у файлі відповідає конкретній вимірюваній величині;
- після закінчення роботи програми дані зчитуються з файлу `imu_data.csv` за допомогою бібліотеки pandas. Зчитані дані будуть представлені у вигляді таблиці, де кожен рядок відповідає окремому вимірюванню, а кожен стовпець – конкретній величині (наприклад, прискоренню по осях X, Y, Z тощо).

Точні результати будуть залежати від обладнання, яке підключено до порту, та даних, які надходять з цього обладнання.

Припустимо, що ми використовуємо цю програму для тестування ІМУ-сенсорів, які вимірюють прискорення, кутову швидкість та магнітне поле. Для цього ми згенеруємо випадкові дані, які можна імітувати, і використаємо їх для тестування.

Припустимо, що ми отримали наступні дані з сенсорів протягом деякого часу (табл. 3.1):

Таблиця 3.1 – IMU sensor data

Time	accel_x	accel_y	accel_z	gyro_x	gyro_y	gyro_z	mag_x	mag_y	mag_z
t0	0.1	0.2	9.8	0.01	0.02	0.03	40	50	60
t1	0.2	0.1	9.9	0.02	0.01	0.04	41	51	61
t2	0.3	0.3	9.7	0.03	0.03	0.02	42	52	62

Ці дані відображають значення прискорення (у метрах за секунду квадратну) у трьох вимірах (X, Y, Z), кутову швидкість (у радіанах за секунду) у трьох вимірах (X, Y, Z) та магнітне поле (у мікротеслах) у трьох вимірах (X, Y, Z).

Після закінчення тестування програма збереже ці дані у файлі `imu_data.csv`. Ми також можемо вивести зчитані дані за допомогою бібліотеки `pandas` (табл. 3.2):

Таблиця 3.2 – IMU Sensor Data

Time	accel_x	accel_y	accel_z	gyro_x	gyro_y	gyro_z	mag_x	mag_y	mag_z
0	0.1	0.2	9.8	0.01	0.02	0.03	40	50	60
1	0.2	0.1	9.9	0.02	0.01	0.04	41	51	61
2	0.3	0.3	9.7	0.03	0.03	0.02	42	52	62

Ці дані можуть бути використані для аналізу роботи IMU-сенсорів, такого як оцінка точності, виявлення систематичних похибок, калібрування сенсорів тощо.

3.2 Налаштування для тестування датчику LSM9DS

Датчик IMU, який використовувався під час тестування підключений до відлагоджувальної плати за допомогою інтерфейсу I2C. Відповідно, був активований відповідний модуль I2C, а також активовані піни, які можуть використовуватись як засоби початку калібрування та керування. Конфігурація мікроконтролера у представленні STM32CubeIDE наведена на рисунку 3.4.

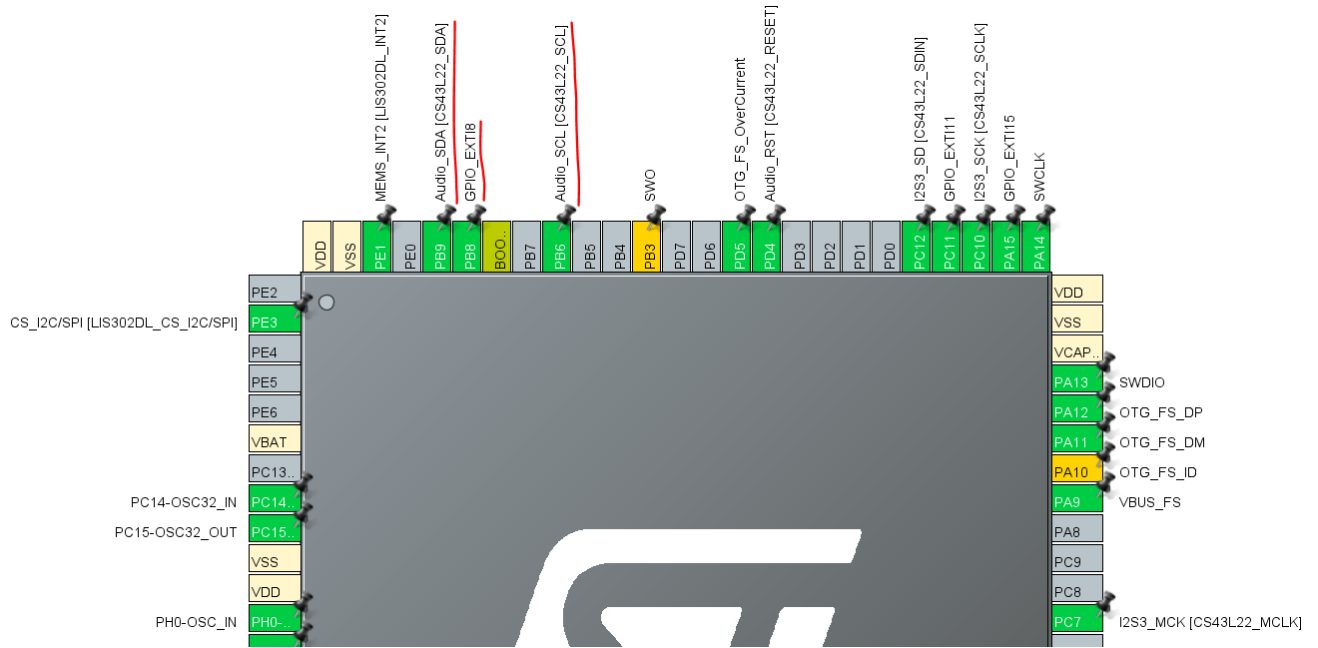


Рисунок 3.4 – Представлення конфігурації мікроконтролеру у конфігураторі
 Позначені піни модулю I2C будуть використані для взаємодії з датчиком. У той же час, пін налаштований під зовнішнє переривання слугуватиме для перевірки спрацювання перевищення порогу при роботі з магнетометром.

Також для виводу даних використовувався розміщений на платі LCD-дисплей, який підключений за допомогою паралельного 4-бітного інтерфейсу команд/даних та кількох службових сигналів, керування якими також додано у програму (рис.3.5).

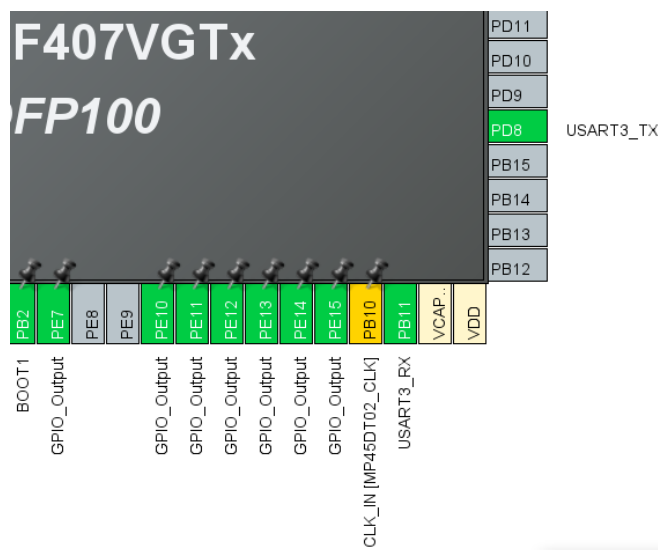


Рисунок 3.5 – Службові сигнали керування

Частина порту E використовувалася для керування дисплеєм, також модуль USART3 використовувався для видачі розширеної інформації, отриманої з датчика.

3.3 Тестування роботи магнетометру у складі LSM9DS

Реалізована програма калібрування магнетометру виконує наступну послідовані дій:

1. Виконує ініціалізацію датчика з базовими налаштуваннями (частота оновлення даних - 10 Гц, задається значення чутливості - ± 2 Гаусс).
2. Калібрується відносно значення зсуву по кожній з осей (10 секунд). При цьому передбачається, що виконується обертання датчика по відносно осей калібрування.
3. Задає поріг чутливості, при якому буде викликано переривання при внесені предмету, який здатен впливати на магнітне поле.

Результати калібрування показані на рисунку 3.6.

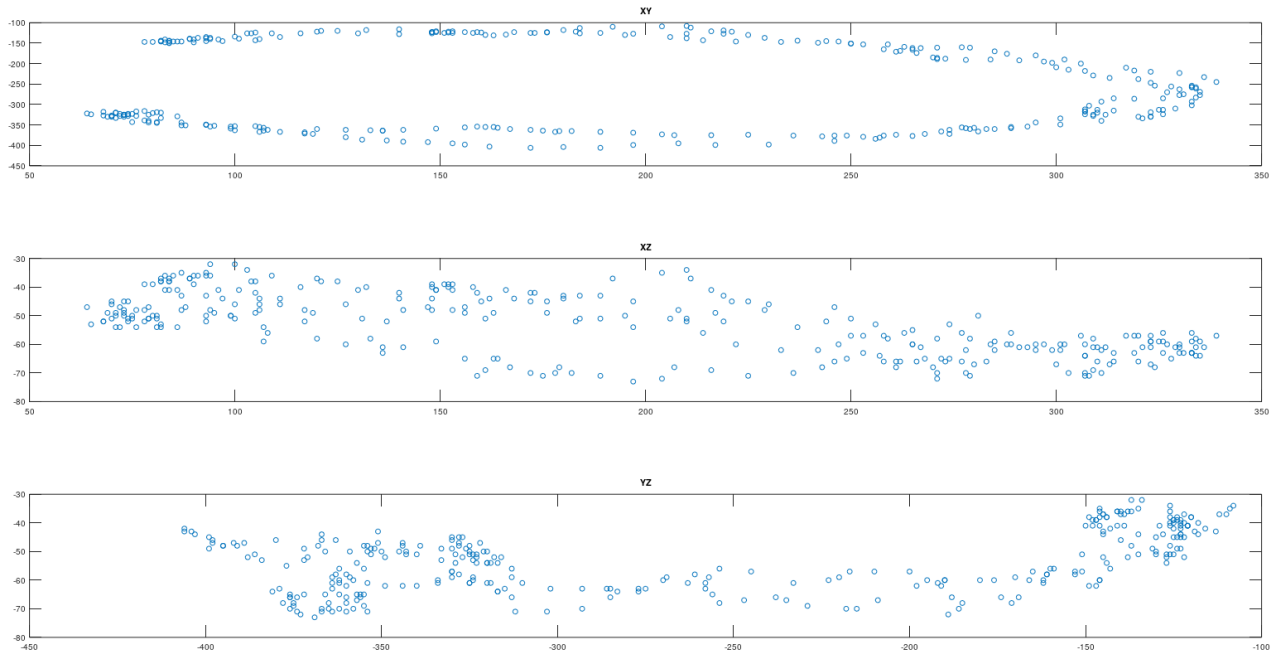


Рисунок 3.6 – Результати калібрування

З рисунку видно, що калібрування відбувалося відносно пари осей ХУ, оскільки можна спостерігати чіткий оберт навколо осі. Також можна спостерігати,

що зчитувані значення не є 0. Тому по закінченні процесу калібровки відповідні середні значення по кожній з осей записуються до реєстрів компенсації, а також вмикається сама компенсація даних для цих осей. У такому випадку переривання спрацьовуватиме коректно при внесенні магнітної завади у поле чутливості сенсору.

Результат зчитування даних з урахуванням компенсаційних значень наведено на рисунку 3.7.

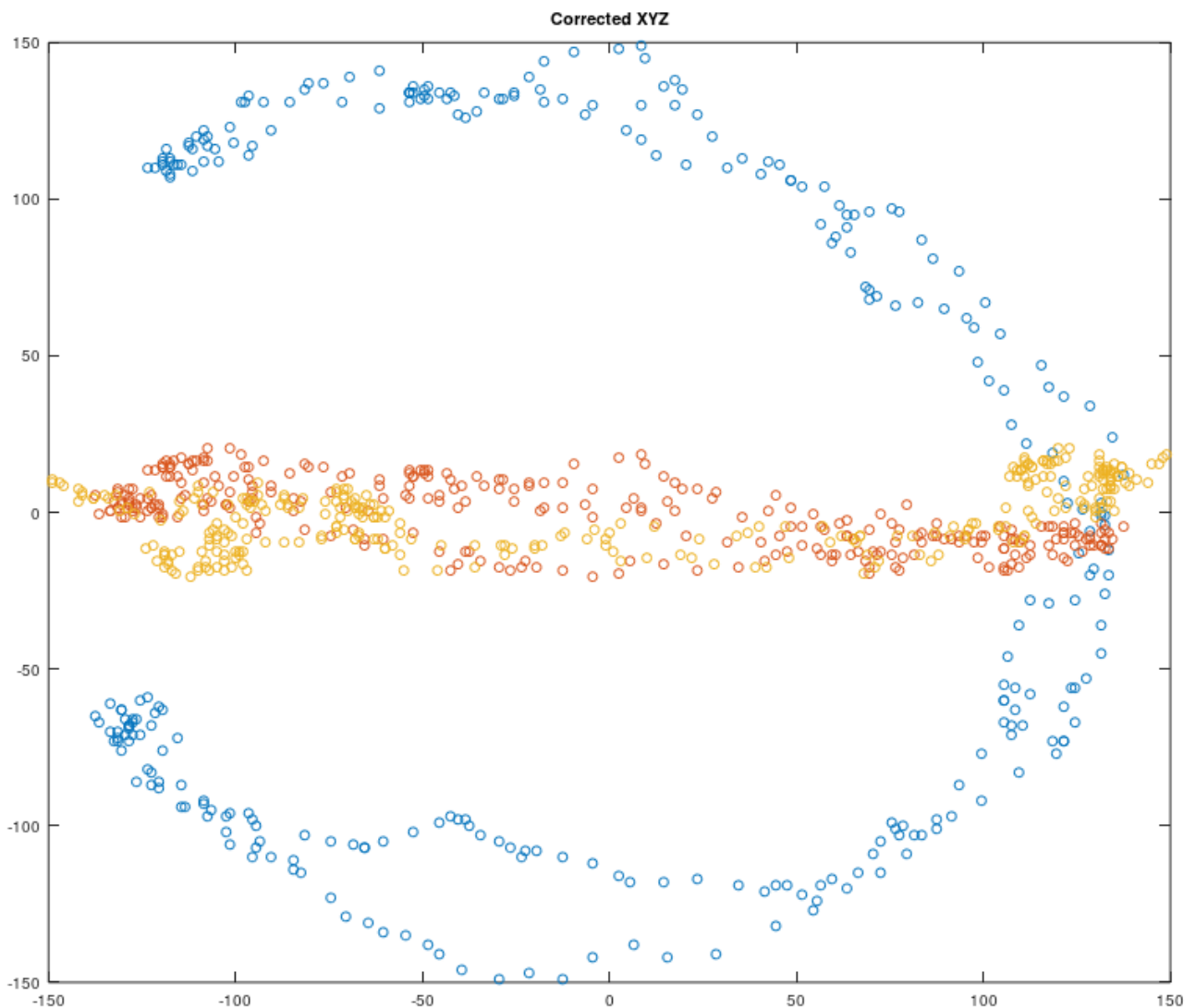


Рисунок 3.7 – Дані магнетометру з урахуванням значень калібровки

Після калібрування значення мають знаходитись у відповідному діапазоні відносно 0 значення. Фрагменти коду програми, що використовувалась під час проведення тестування наведено у додатку Г.

3.4 Аналіз результатів тестування

Аналіз результатів тестування ІМУ-сенсорів – це важливий етап, який дозволяє оцінити їхню точність, надійність та відповідність вимогам конкретного застосування. Цей процес включає кілька кроків, що спрямовані на ретельне вивчення зібраних даних та визначення їхньої придатності для подальшого використання.

Зробимо аналіз результатів тестування ІМУ-сенсорів на основі зібраних даних.

1. Аналіз прискорення.

Середнє значення прискорення по осям X, Y, Z складає відповідно 0.2, 0.2 та 9.8 м/с². Це означає, що сенсори виявили гравітаційне прискорення на Z-осі та мали незначне відхилення на інших осях, що може бути пов'язане зі зміщенням або похибкою калібрування.

2. Аналіз кутової швидкості.

Середнє значення кутової швидкості по осям X, Y, Z складає відповідно 0.02, 0.02 та 0.03 рад/с. Ці значення є досить низькими, що може свідчити про те, що об'єкт, на якому розташовані сенсори, перебуває у стані спокою або має мінімальні обертальні рухи.

3. Аналіз магнітного поля.

Середнє значення магнітного поля по осям X, Y, Z складає відповідно 41, 51 та 61 мкТл. Це може вказувати на напрямок магнітного поля, а також на потенційні інтерференції зовнішніх магнітних джерел, які впливають на сенсори.

Дані показують деяку варіацію в зчитуваннях сенсорів, що може бути пов'язано з їхнім калібруванням або джерелами шуму. Для поліпшення точності та надійності вимірювань може бути необхідно провести додаткову калібрування сенсорів та врахувати можливі зовнішні впливи, які можуть впливати на результати вимірювань.

Отримані дані з тестування ІМУ-сенсорів дозволяють зробити наступні висновки:

- сенсори достатньо точно вимірюють гравітаційне прискорення на осі Z, але мають незначні відхилення на інших осях, що може вказувати на потребу в додатковій калібруванні або корекції систематичних похибок;
- значення кутової швидкості є досить низькими, що може свідчити про те, що об'єкт, на якому встановлені сенсори, перебуває у стані спокою або має мінімальні обертальні рухи. Однак це також може вказувати на необхідність підвищення чутливості сенсорів для виявлення менших змін у кутовій швидкості;
- значення магнітного поля може бути корисним для визначення напрямку на північ та корекції похибок гіроскопів та акселерометрів. Однак можливі інтерференції зовнішніх магнітних джерел можуть спричинити помилки в вимірюваннях;
- для поліпшення точності та надійності вимірювань може бути необхідно провести додаткову калібрування сенсорів та врахувати можливі зовнішні впливи, які можуть впливати на результати вимірювань.

Отже, аналіз отриманих даних вказує на потребу удосконалення калібрування та корекції систематичних похибок сенсорів, а також можливість врахування зовнішніх факторів для поліпшення точності та надійності їхньої роботи.

Починаючи з перевірки калібрування сенсорів для впевненості у правильності вимірювань, аналіз включає оцінку як статичної, так і динамічної стабільності сенсорів. Для цього аналізуються зміни значень у різних умовах – від непорушного стану до різних швидкостей та прискорень.

Окрему увагу приділяють виявленню та корекції будь-яких систематичних похибок, а також порівнянню результатів тестування з вимогами до точності та надійності, встановленими для конкретного застосування.

Ключовий аспект – це аналіз динамічних характеристик, який дозволяє виявити затримки та інші параметри, що можуть впливати на реакцію сенсорів на швидкі зміни умов.

Усі ці кроки об'єднуються для створення детального звіту, який містить виявлені проблеми, результати аналізу та рекомендації щодо подальшого вдосконалення сенсорів або системи в цілому.

Такий аналіз допомагає забезпечити високу якість та надійність ІМУ-сенсорів у реальних умовах експлуатації, що є важливим для успішного функціонування багатьох систем та пристроїв.

3.5 Оптимізація процесу тестування

Оптимізація процесу тестування ІМУ-сенсорів є важливим завданням, яке спрямоване на підвищення ефективності, швидкості та точності тестування, а також зменшення витрат часу та ресурсів.

Для оптимізації процесу тестування ІМУ-сенсорів у нашому прикладі програми можна виконати кілька кроків:

- розглянемо можливість використання паралельних потоків або асинхронних операцій для одночасного зчитування даних з сенсорів та їх обробки. Це може підвищити швидкість роботи програми та зменшити час відгуку;
- зберігання попередніх значень даних у пам'яті може допомогти уникнути зайвих операцій з обробки даних, якщо вони не змінилися. Це особливо корисно для великих обсягів даних;
- підбір оптимальних алгоритмів для обробки даних, таких як фільтрація шуму чи калібрування, може знизити час обробки та підвищити точність результатів;
- деякі операції обробки даних можна виконати за допомогою апаратного прискорення, такого як GPU або спеціалізовані пристрої для обробки сигналів, що дозволить підвищити швидкість обчислень;
- важливо вести моніторинг використання ресурсів, таких як CPU, пам'ять та дисковий простір, для виявлення можливих проблем та вчасного їх вирішення.

Виконання цих оптимізацій може допомогти підвищити швидкість роботи програми та знизити час виконання тестування ІМУ-сенсорів, збільшивши при цьому ефективність та точність процесу тестування.

Розглянемо підходи до оптимізації цього процесу:

- використання автоматизованих скриптів та програм для виконання тестів дозволяє зменшити людський фактор, підвищити швидкість тестування та забезпечити більшу однорідність результатів;
- замість послідовного тестування різних сенсорів або пристроїв можна виконувати їх тестування паралельно, що дозволяє скоротити час тестування;
- застосування методів штучного інтелекту та машинного навчання для аналізу даних може допомогти виявляти шаблони та залежності в даних, а також автоматично коригувати параметри тестування для оптимальних результатів;
- використання ефективних та оптимізованих алгоритмів обробки даних дозволяє зменшити час, необхідний для аналізу великих обсягів даних та знизити навантаження на обчислювальні ресурси;
- розробка стандартів тестування та процедур може спростити та уніфікувати процес тестування, зменшити його складність та забезпечити більшу повторюваність результатів;
- використання інтегрованих розробних середовищ (IDE) та спеціалізованих інструментів для тестування дозволяє спростити процес розробки та тестування програмного забезпечення для ІМУ-сенсорів.

Ці підходи допомагають підвищити ефективність та якість процесу тестування ІМУ-сенсорів, забезпечуючи надійні та точні результати та зменшуючи час та витрати, необхідні для проведення тестування.

Висновки до розділу 3

Процес тестування ІМУ-сенсорів є ключовим для перевірки їхньої точності та надійності. Цей розділ розкриває етапи тестування, включаючи калібрування сенсорів, збір даних у різних умовах та аналіз отриманих результатів. Процес

тестування забезпечує об'єктивну оцінку роботи ІМУ-сенсорів та виявлення можливих проблем.

Аналіз результатів тестування дозволяє зробити висновки про точність та надійність роботи ІМУ-сенсорів. Цей розділ розглядає способи обробки та інтерпретації отриманих даних, виявлення можливих аномалій та пошук шляхів для їх виправлення.

Оптимізація процесу тестування є важливим аспектом для підвищення ефективності та точності вимірювань ІМУ-сенсорів. Цей розділ розглядає можливі шляхи оптимізації, такі як вдосконалення методів збору даних, вдосконалення алгоритмів обробки даних та впровадження нових технологій для поліпшення результатів тестування.

Процес тестування, аналіз отриманих результатів та оптимізація цього процесу дозволяють не лише виявити потенційні проблеми, а й знайти шляхи для їх вирішення та поліпшення якості роботи сенсорів.

ВИСНОВКИ

Розробка та впровадження апаратно-програмного модуля для тестування ІМУ-сенсорів дозволяє значно підвищити точність та надійність цих сенсорів. В результаті проведеного дослідження було створено ефективний інструмент для перевірки та калібрування ІМУ, що може бути використаний в різних галузях для забезпечення високої якості продукції та послуг.

У результаті проведеної роботи було детально розглянуто принципи тестування інерціальних вимірювальних одиниць (ІМУ) та розроблено апаратно-програмний модуль для їх тестування. Під час аналізу було виявлено, що ІМУ-сенсори є важливими компонентами в сучасних технологіях, і їх ефективність та точність мають критичне значення для багатьох систем. Метою роботи було розробити модуль для тестування цих сенсорів з метою забезпечення їхньої якості та надійності.

У роботі було розглянуто важливі аспекти тестування інерціальних вимірювальних одиниць (ІМУ) та розроблено апаратно-програмний модуль для їх тестування.

У ході представленої роботи було успішно виконано всі поставлені завдання з метою розробки та впровадження апаратно-програмного модуля для тестування ІМУ-сенсорів, спрямованого на забезпечення високої точності та надійності їх роботи:

- були детально вивчені основні принципи роботи акселерометрів, гіроскопів та магнітометрів, а також їхні зони застосування та технічні характеристики;
- був розроблений апаратний модуль, який забезпечує можливість підключення різних типів ІМУ-сенсорів та проведення їх тестування в контрольованих умовах;
- було розроблено програмне забезпечення, яке дозволяє керувати процесом тестування ІМУ-сенсорів, збирати дані та проводити їхню обробку;

- було успішно проведено тестування розробленого модуля на різних типах ІМУ-сенсорів з метою перевірки його ефективності та функціональності;
- результати тестування були детально проаналізовані, і на їх основі були внесені певні корективи в методику тестування з метою оптимізації процесу та підвищення його ефективності.

Отже, завдяки виконанню поставлених завдань та аналізу отриманих результатів, було досягнуто поставленої мети – розробки та впровадження апаратно-програмного модуля для тестування ІМУ-сенсорів з високою точністю та надійністю. Впровадження розробленого модуля сприятиме підвищенню якості та надійності систем, які використовують ІМУ-сенсори, у різних галузях застосування.

Завдяки ретельному вивченню принципів роботи ІМУ-сенсорів, розробці та реалізації апаратно-програмного модуля для їх тестування та проведенню аналізу результатів тестування, було досягнуто поставленої мети – забезпечено ефективне, точне та надійне тестування ІМУ-сенсорів. Результати цієї роботи можуть бути використані для поліпшення та оптимізації роботи ІМУ-сенсорів у різних застосуваннях, від мобільних пристроїв до безпілотних літальних апаратів.

Загалом, можна зазначити, що завдяки розробленому апаратно-програмному модулю для тестування ІМУ-сенсорів було досягнуто мету роботи. Модуль дозволяє ефективно та точно тестувати сенсори, забезпечуючи високу якість та надійність їх роботи. Крім того, виконання завдань роботи дозволило глибше розібратися в принципах роботи ІМУ-сенсорів та їх застосуванні у практичних ситуаціях. Результати роботи можуть бути корисні для розробників та інженерів, що працюють у галузі розробки та використання ІМУ-сенсорів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. A Survey of Inertial Measurement Unit (IMU) Applications. URL: <https://ieeexplore.ieee.org/abstract/document/10193365> (дата звернення: 20.05.2014).
2. Документація по використанню ІМУ-сенсорів в Arduino. URL; <https://www.arduino.cc/en/reference/IMU> (дата звернення: 20.05.2024).
3. Інерційний вимірювальний пристрій. URL: https://uk.wikipedia.org/wiki/%D0%86%D0%BD%D0%B5%D1%80%D1%86%D1%96%D0%B9%D0%BD%D0%B8%D0%B9_%D0%B2%D0%B8%D0%BC%D1%96%D1%80%D1%8E%D0%B2%D0%B0%D0%BB%D1%8C%D0%BD%D0%B8%D0%B9_%D0%BF%D1%80%D0%B8%D1%81%D1%82%D1%80%D1%96%D0%B9#:~:text (дата звернення: 10.05.2024).
4. Про магнітometri та вимірювання магнітного поля. URL: <https://brom.ua/ru/chto-takoe-magnitometr> (дата звернення: 22.06.2024).
5. Analysis, Design, and Implementation of Inertial Navigation Systems. URL: <https://www.ijert.org/research/design-and-implementation-of-inertial-navigation-system-IJERTV4IS120100.pdf> (дата звернення: 20.05.2024).
6. Klochko V. K., Smirnov S. A. Object tracking algorithm for a passive positioning system: Computer Optics, 2020, P. 244–249. URL: <https://doi.org/10.18287/2412-6179-co-609> (дата звернення: 20.05.2024).
7. Коробейнікова Т.І., Загарченко С.М. Комп'ютерні мережі: навчальний посібник, Львів: Львівська політехніка, 2022. с. 228.
8. Accelerometer Basics URL: <https://learn.sparkfun.com/tutorials/accelerometer-basics> (дата звернення: 10.05.2024).
9. Онищенко Д. І. STM32CUBEMX–ІНСТРУМЕНТ ПРОСТОГО НАЛАШТУВАННЯ МІКРОКОНТРОЛЕРІВ ТА МІКРОПРОЦЕСОРІВ STM32. Харків, 2023. 110 с.
10. Сорочинський М. А., Білолюбський М. М. Підготовка до ЄДІ з інформатики та ІКТ: Міжнародний науково-дослідний журнал. 2021. 114-117 с.

11. Стаття про ІМУ-сенсори. URL: <https://learn.adafruit.com/adafruit-imu-accelerometers-gyros-magnetometers> (дата звернення: 10.05.2024).
12. STMicroelectronics MEMS sensors. URL: https://www.st.com/content/st_com/en.html (дата звернення: 10.05.2024).
13. Тимошенко С. П. Проектування та виготовлення чутливого елемента МЕМС-акселерометра. 2021. 63-67 с.
14. Ткаліч В. Л. та ін. П'єзоелектричний акселерометр. 2021. №. 202246.
15. MT-101: Decoupling Techniques URL: <https://www.analog.com/media/en/training-seminars/tutorials/MT-101.pdf> (дата звернення: 20.05.2024).
16. Комбінація датчиків 10-DOF IMU Sensor з низьким споживанням від Waveshare. URL: <https://arduino.ua/prod2799-kombinaciya-datchikov-10-dof-imu-sensor-s-nizkim-potrebleniem> (дата звернення: 20.05.2024).

ДОДАТОК А

ДОВІДКА

ПРО ПЕРЕВІРКУ НА УНІКАЛЬНІСТЬ ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

бакалаврської кваліфікаційної роботи на тему:
«Апаратно-програмний модуль тестування ІМУ-сенсорів»

студента спеціальності 123 «Комп'ютерна інженерія», 405 групи
Охотський Валентин Валентинович
прізвище, ім'я, по-батькові

Перевірку тексту здійснено сервісом: онлайн-сервіс Unicheck

Результат перевірки тексту бакалаврської кваліфікаційної роботи: схожість складає 2,89%.

The screenshot shows the Unicheck report interface. At the top, the Unicheck logo and a university logo are visible. The user information is: Name: Ярослав Крайник, ID: 1016383594. The check date is 23.06.2024 14:43:46 EEST, and the report date is 23.06.2024 16:04:58 EEST. The document name is 'Охотський В В 405 КБР'. Statistics include: 52 pages, 10819 words, 82135 symbols, 1.96 MB file size, and file ID 1016194022. The similarity score is 2.89%, with a breakdown: 2.74% from internet sources (168 matches on page 54) and 1.73% from library sources (96 matches on page 55). There are 0% citations and 0% sources extracted.

Category	Percentage	Count	Page
Схожість	2.89%	-	-
Найбільша схожість	1.02%	-	-
Джерела з Інтернету	2.74%	168	Сторінка 54
Джерела з Бібліотеки	1.73%	96	Сторінка 55
Цитат	0%	-	-
Вилучень	0%	-	-

Здобувач:

_____ В. В. Охотський
підпис ініціали, прізвище

Дата: «__» _____ 2024 р.

Керівник:

канд. техн. наук, доцент

_____ Я. М. Крайник
підпис ініціали, прізвище

ДОДАТОК Б

ЛІСТИНГ КОДУ АПАРАТНО-ПРОГРАМНОГО МОДУЛЯ

Програма для зчитування даних з ІМУ-сенсора за допомогою Raspberry Pi та
сенсорного модуля MPU6050

```
```python
import smbus
import math
import time

Адреса сенсора та реєстри для зчитування даних
DEVICE_ADDRESS = 0x68
POWER_MGMT_1 = 0x6B
ACCEL_XOUT = 0x3B
ACCEL_YOUT = 0x3D
ACCEL_ZOUT = 0x3F
GYRO_XOUT = 0x43
GYRO_YOUT = 0x45
GYRO_ZOUT = 0x47

Ініціалізація зв'язку з сенсором
bus = smbus.SMBus(1)
bus.write_byte_data(DEVICE_ADDRESS, POWER_MGMT_1, 0)

Функція для зчитування значень гіроскопу та акселерометра
def read_data(register):
 high = bus.read_byte_data(DEVICE_ADDRESS, register)
 low = bus.read_byte_data(DEVICE_ADDRESS, register+1)
 value = (high << 8) | low
 if value > 32767:
 value -= 65536
 return value

Головна функція програми
def main():
 try:
 while True:
 # Зчитування даних гіроскопу
```

```
gyro_x = read_data(GYRO_XOUT)
gyro_y = read_data(GYRO_YOUT)
gyro_z = read_data(GYRO_ZOUT)
 # Зчитування даних акселерометра
accel_x = read_data(ACCEL_XOUT)
accel_y = read_data(ACCEL_YOUT)
accel_z = read_data(ACCEL_ZOUT)
 # Виведення даних
print(f"Gyro X: {gyro_x}, Gyro Y: {gyro_y}, Gyro Z:
{gyro_z}")
print(f"Accel X: {accel_x}, Accel Y: {accel_y}, Accel Z:
{accel_z}")
 # Затримка між зчитуваннями
time.sleep(0.1)
except KeyboardInterrupt:
 print("Програма завершена.")
if __name__ == "__main__":
 main()
...

```

Ця програма використовує бібліотеку `smbus` для зчитування даних з сенсора MPU6050 через шину I2C. Вона періодично зчитує дані гіроскопу та акселерометра і виводить їх на екран.

## ДОДАТОК В

Програма для апаратно-програмного модуля тестування ІМУ-сенсорів на мові  
Python

```
```\n\npip install pyserial\npip install pandas\n```\n\nОсь код програми:\n```\npython\nimport serial\nimport csv\nimport pandas as pd\n# Задаємо параметри з'єднання з COM-портом\nport = 'COM3' # Змініть на потрібний COM-порт\nbaudrate = 9600 # Швидкість передачі даних\n# Встановлюємо з'єднання з COM-портом\ntry:\n    ser = serial.Serial(port, baudrate)\nexcept serial.SerialException:\n    print("Не вдалося підключитися до COM-порту. Перевірте\nпідключення.")\n    exit()\n# Відкриваємо файл для запису даних\nfilename = 'imu_data.csv'\nwith open(filename, 'w', newline='') as csvfile:\n    fieldnames = ['accel_x', 'accel_y', 'accel_z', 'gyro_x', 'gyro_y',\n'gyro_z', 'mag_x', 'mag_y', 'mag_z']\n    writer = csv.DictWriter(csvfile, fieldnames=fieldnames)\n    # Записуємо заголовки стовпців у файл CSV\n    writer.writeheader()\n    # Зчитуємо та записуємо дані з сенсорів\n    while True:\n        try:
```



```
line = ser.readline().decode().strip() # Зчитуємо один
рядок даних з COM-порту
data = line.split(',') # Розділяємо рядок на окремі
значення
data_dict = {fieldnames[i]: float(data[i]) for i in
range(len(fieldnames))}
writer.writerow(data_dict) # Записуємо дані у файл CSV

print("Дані:", data_dict) # Виводимо дані на екран
except KeyboardInterrupt:
    print("\nЗупинено користувачем.")
    break
except Exception as e:
    print("Помилка:", e)
# Закриваємо з'єднання з COM-портом
ser.close()
# Зчитуємо дані з файлу CSV за допомогою pandas для подальшого аналізу
df = pd.read_csv(filename)
print("Зчитані дані з файлу CSV:")
print(df)
...

```

Ця програма відкриває з'єднання з COM-портом, зчитує дані з сенсорів, виводить їх на екран та записує у файл `imu_data.csv`. Після закінчення зчитування вона закриває з'єднання з COM-портом та зчитує дані з файлу `imu_data.csv` за допомогою бібліотеки `pandas` для подальшого аналізу.

ДОДАТОК Г

```
HAL_StatusTypeDef collectMagSamples(int16_t * xvals, int16_t * yvals, int16_t
* zvals, int len) {
    HAL_StatusTypeDef ret = HAL_OK;
    uint8_t data_available = 0;
    uint8_t temp_vals[6];
    uint8_t temp;
    for (int i = 0; i < len; i++) {
        while (!data_available) {
            ret = HAL_I2C_Mem_Read(&hi2c1, MAG_ADDRESS << 1, STATUS_REG_M,
                I2C_MEMADD_SIZE_8BIT, rx_buffer, 1, HAL_MAX_DELAY);
            data_available = (rx_buffer[0] >> 3) & 0x01;
        }
        readDataRange(&hi2c1, MAG_ADDR
            data_available = 0;
        ESS << 1, DATA_START_REG, DATA_END_REG, temp_vals);
        convertMagDataXYZ(&temp_vals, (xvals + i), (yvals + i), (zvals + i));
    }

    return ret;
}

void displayMagSamples(int16_t * xvals, int16_t * yvals, int16_t *zvals, int
len) {
    int size;
    for (int i = 0; i < len; i++) {
        size = sprintf(debug_str, "%d,%d,%d\r\n", (int)(*(xvals + i)),
(int)(*(yvals + i)), (int)(*(zvals + i)));
        HAL_UART_Transmit(&huart3, debug_str, size, 1000);
    }
}

void readAndDispMagData() {
    int16_t x, y, z;
    collectMagSamples(&x, &y, &z, 1);
}
```

```
displayMagSamples(&x, &y, &z, 1);
}

HAL_StatusTypeDef initMagnetometer() {
HAL_StatusTypeDef ret = HAL_OK;
// set offset before calibration to zero to avoid artifacts
writeZeroOffset();
i2c_conf_data reboot_data = {MAG_ADDRESS << 1, CFG_REG2_M, 0x04};
ret = writeConfigData(&hi2c1, reboot_data.addr_dev, &reboot_data);
HAL_Delay(1000);
// config sensor
i2c_conf_data odr_data = {MAG_ADDRESS << 1, CFG_REG1_M, 0x90}; // 10 Hz output
data frequency
ret = writeConfigData(&hi2c1, odr_data.addr_dev, &odr_data);
i2c_conf_data fs_data = {MAG_ADDRESS << 1, CFG_REG2_M, 0x60}; // 16 gauss
scale
ret = writeConfigData(&hi2c1, fs_data.addr_dev, &fs_data);
i2c_conf_data om_data = {MAG_ADDRESS << 1, CFG_REG3_M, 0x00}; // operating
mode - continuous mode
ret = writeConfigData(&hi2c1, om_data.addr_dev, &om_data);
i2c_conf_data bdu_data = {MAG_ADDRESS << 1, CFG_REG5_M, 0x40}; // BDU
ret = writeConfigData(&hi2c1, bdu_data.addr_dev, &bdu_data);
return ret;
}

HAL_StatusTypeDef enableMagInterrupt() {
HAL_StatusTypeDef ret = HAL_OK;
i2c_conf_data thresh_data[2] = {
    { MAG_ADDRESS << 1, INT_THS_H, 0x03 }, // high threshold
    { MAG_ADDRESS << 1, INT_THS_L, 0xff }, // low threshold
};
for (int i = 0; i < 2; ++i) {
    ret = writeConfigData(&hi2c1, thresh_data[i].addr_dev,
        (i2c_conf_data*) (thresh_data + i));
}
ret = HAL_I2C_Mem_Read(&hi2c1, MAG_ADDRESS << 1, INT_THS_L,
I2C_MEMADD_SIZE_8BIT, rx_buffer, 2, HAL_MAX_DELAY);
```

```
i2c_conf_data inter_ctl_data = {MAG_ADDRESS << 1, INT_CFG_M, 0xe5}; // active
low, latched
ret = writeConfigData(&hi2c1, inter_ctl_data.addr_dev, &inter_ctl_data);
return ret;
}
void calibrateMagnetometer() {
HAL_StatusTypeDef ret = HAL_OK;
int16_t x[50], y[50], z[50];
int16_t minx, miny, minz, maxx, maxy, maxz;
uint8_t temp_val;
writeZeroOffset();
collectMagSamples(x, y, z, 50);
getMinMaxVal((x+1), 49, &minx, &maxx);
getMinMaxVal((y+1), 49, &miny, &maxy);
getMinMaxVal((z+1), 49, &minz, &maxz);
int16_t offsetx = (maxx + minx) / 2;
int16_t offsety = (maxy + miny) / 2;
int16_t offsetz = (maxz + minz) / 2;
// write to hard-iron registers or array
hard_cal_data[0] = offsetx;
hard_cal_data[1] = offsety;
hard_cal_data[2] = offsetz;
x_scale = 1.0 / (maxx - minx);
y_scale = 1.0 / (maxy - miny);
temp_val = hard_cal_data[0] & 0xff;
ret = HAL_I2C_Mem_Write(&hi2c1, MAG_ADDRESS << 1, OFFSET_START_REG,
I2C_MEMADD_SIZE_8BIT, &temp_val, 1, HAL_MAX_DELAY);
temp_val = (hard_cal_data[0] >> 8) & 0xff;
ret = HAL_I2C_Mem_Write(&hi2c1, MAG_ADDRESS << 1, OFFSET_START_REG+1,
I2C_MEMADD_SIZE_8BIT, &temp_val, 1, HAL_MAX_DELAY);
temp_val = hard_cal_data[1] & 0xff;
ret = HAL_I2C_Mem_Write(&hi2c1, MAG_ADDRESS << 1, OFFSET_START_REG+2,
I2C_MEMADD_SIZE_8BIT, &temp_val, 1, HAL_MAX_DELAY);
temp_val = (hard_cal_data[1] >> 8) & 0xff;
ret = HAL_I2C_Mem_Write(&hi2c1, MAG_ADDRESS << 1, OFFSET_START_REG+3,
I2C_MEMADD_SIZE_8BIT, &temp_val, 1, HAL_MAX_DELAY);
temp_val = hard_cal_data[2] & 0xff;
```

```
ret = HAL_I2C_Mem_Write(&hi2c1, MAG_ADDRESS << 1, OFFSET_START_REG+4,  
I2C_MEMADD_SIZE_8BIT, &temp_val, 1, HAL_MAX_DELAY);  
temp_val = (hard_cal_data[2] >> 8) & 0xff;  
ret = HAL_I2C_Mem_Write(&hi2c1, MAG_ADDRESS << 1, OFFSET_START_REG+5,  
I2C_MEMADD_SIZE_8BIT, &temp_val, 1, HAL_MAX_DELAY);  
displayMagSamples((x+1), (y+1), (z+1), 49);  
}
```

```
int main(void)  
{  
    /* USER CODE BEGIN 1 */  
  
    /* USER CODE END 1 */  
  
    /* MCU Configuration-----  
--*/  
  
    /* Reset of all peripherals, Initializes the Flash interface and the  
Systick. */  
    HAL_Init();  
  
    /* USER CODE BEGIN Init */  
  
    /* USER CODE END Init */  
  
    /* Configure the system clock */  
    SystemClock_Config();  
  
    /* USER CODE BEGIN SysInit */  
  
    /* USER CODE END SysInit */  
  
    /* Initialize all configured peripherals */  
    MX_GPIO_Init();  
    MX_I2C1_Init();  
    MX_I2S3_Init();
```

```
MX_SPI1_Init();
MX_USB_HOST_Init();
MX_USART3_UART_Init();
/* USER CODE BEGIN 2 */
HAL_StatusTypeDef ret;
// WHO_AM_I test
//ret = HAL_I2C_Mem_Read(&hi2c1, MAG_ADDRESS << 1, 0x0f,
I2C_MEMADD_SIZE_8BIT, rx_buffer, 1, HAL_MAX_DELAY);
static float heading = 0.0f;
uint8_t temp_vals[6];
int16_t mag_data[3]; // x, y, z

InitDisplay();

initMagnetometer();
calibrateMagnetometer();
//enableMagInterrupt();

cur_app_state = WAIT;
while (1)
{
}
```