

Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки

## **МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА**

на тему:

**«Інформаційна технологія GPS моніторингу та управління технічним  
забезпеченням агропромислового підприємства. Частина 1.**

**Апаратна частина**

08-23.КМКР.029.00.000 ПЗ

Виконав: студент 2 курсу, групи 1КІ-20м  
напряму підготовки (спеціальності)

123 — «Комп'ютерна інженерія»

\_\_\_\_\_ Паламарчук М. О.

Керівник: д.т.н., проф. каф. ОТ

\_\_\_\_\_ Азаров О. Д.  
(прізвище та ініціали)

Опонент: д.т.н., проф. каф. МБІС

\_\_\_\_\_ Яремчук Ю. Є.  
(прізвище та ініціали)

**Допущено до захисту**

Завідувач кафедри ОТ

\_\_\_\_\_ д.т.н., проф. Азаров О. Д.

«  » \_\_\_\_\_ 2021 р.

Вінниця 2021

Вінницький національний технічний університет  
Факультет інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки  
Освітньо-кваліфікаційний рівень — магістр  
Спеціальність 123 — «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ  
Завідувач кафедри ОТ  
Азаров О. Д.

« \_\_\_\_ »  
\_\_\_\_\_ 2021 року

## **З А В Д А Н Н Я НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Паламарчуку Миколі Олександровичу

1 Тема роботи «Інтелектуальна система GPS моніторингу та управління технічним забезпеченням агропромислового підприємства. Частина 1. Апаратна частина.», керівник роботи: д.т.н., професор Азаров О.Д. затверджені наказом Вінницького національного технічного університету від 24.09.2021 року № 277.

2 Строк подання студентом роботи 10.12.2021

3 Вихідні дані до роботи: проаналізувати агропромислову сферу на використання новітніх технологій та Інтернету речей, розробити систему GPS моніторингу, створити метод виміру рівня палива.

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): розглянути та проаналізувати сферу агропромислового виробництва; розробити структурну схему системи, реалізувати алгоритм роботи системи.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): схема електрична принципова Raspberry Pi, структурна схема системи, блок-схема алгоритму, лістинги коду та фото компонентів системи.

6 Консультанти розділів роботи приведені в таблиці 1.

Таблиця 1 — Консультанти

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1,2,3	Азаров О.Д., д.т.н., професор		
4	Кавецький В. В., к.е.н., доцент каф. ЕПВМ		

7 Дата видачі завдання 07.09.2021

Календарний план наведено в таблиці 2.

Таблиця 2 — Календарний план

№ з/п	Назва етапів дипломної магістерської роботи	Строк виконання етапів роботи	Примітка
1	Аналіз проблеми, обґрунтування актуальності розробки методів та постановка задачі	04.10.2021	Виконано
2	Розробка структурної схеми системи	29.10.2021	Виконано
3	Розробка методу вимірювання рівня палива	15.11.2021	Виконано
4	Розробка алгоритму роботи системи	26.11.2021	Виконано
5	Реалізація програмної частини системи	03.12.2021	Виконано
6	Економічна частина	10.12.2021	Виконано

Студент \_\_\_\_\_ Паламарчук М.О.  
( підпис ) (прізвище та ініціали)

Керівник магістерської кваліфікаційної роботи \_\_\_\_\_ Азаров О.Д.  
( підпис ) (прізвище та ініціали)

Консультант з економічної частини \_\_\_\_\_ Кавецький В. В.  
( підпис ) (прізвище та ініціали)

## АНОТАЦІЯ

УДК 004.3

Розроблено інформаційну технологію GPS моніторингу та управління технічним забезпеченням агропромислового підприємства. Продукт вирішує проблему недостатнього технологічного розвитку агропромислового сектору України.

Для реалізації системи розроблено алгоритм роботи, структурну схему та програмну реалізацію роботи системи.

Ключові слова: моніторинг об'єктів, програмно-апаратний засіб, GPS-моніторинг, Raspberry Pi, агропромисловість.

## ANNOTATION

UDC 004.3

The information technology of GPS monitoring and management of technical support of the agro-industrial enterprise is developed. The product solves the problem of insufficient technological development of the agro-industrial sector of Ukraine.

To implement the system, an algorithm, block diagram and software implementation of the system have been developed.

Keywords: object monitoring, software and hardware, GPS monitoring, Raspberry Pi, agro-industry.

## ЗМІСТ

Вступ.....	8	
<b>1 ОГЛЯД І АНАЛІЗ СУЧАСНИХ ТЕХНІЧНИХ РІШЕНЬ</b>		
<b>АГРОПРОМИСЛОВОЇ СФЕРИ .....</b>	<b>10</b>	
1.1 Огляд агропромислової сфери.....	10	
1.2 Технічні рішення і технології.....	13	
1.3 Аналіз існуючих аналогів систем.....	29	
1.4 Обґрунтування доцільності розробки.....	36	
<b>2 РОЗРОБКА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕЛЕКТУАЛЬНОЇ</b>		
<b>СИСТЕМИ GPS МОНІТОРИНГУ ТА УПРАВЛІННЯ .....</b>	<b>37</b>	
2.1 Вибір мікропроцесорної платформи .....	37	
2.2 Розробка структурної схеми пристрою .....	61	
2.3 Вибір компонентів пристрою .....	65	
2.4 Методи виміру об'єму палива .....	72	
<b>3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕЛЕКТУАЛЬНОЇ</b>		
<b>СИСТЕМИ GPS МОНІТОРИНГУ ТА УПРАВЛІННЯ .....</b>	<b>84</b>	
3.1 Алгоритм роботи системи.....	84	
3.2 Інтерфейси і протоколи .....	86	
3.3 Програмна реалізація алгоритму .....	91	
<b>4 ЕКОНОМІЧНА ЧАСТИНА .....</b>		<b>114</b>
4.1 Оцінювання економічного потенціалу розробки .....	114	
4.3 Прогнозування комерційних ефектів від реалізації результатів розробки .....	123	

					<i>08-23.КМКР.029.00.000 ПЗ</i>			
Змн.	Арк.	№ док.ум.	Підпис	Дата	Інформаційна технологія GPS моніторингу та управління технічним забезпеченням агропромислового підприємства. Частина 1. Апаратна частина Пояснювальна записка	Літ.	Арк.	Акрушів
Розроб.		Паламарчук М.О						
Перевір.		Азаров О.Д.					6	146
Реценз.		Яремчук Ю.Є.				<b>1КІ-20М</b>		
Н. Контр.		Швець С.І.						
Затверд.		Азаров О. Д.						

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності.....	124
Висновки .....	129
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	130
ДОДАТОК А Технічне завдання .....	131
ДОДАТОК Б Лістинг коду .....	135
ДОДАТОК В Фото компонентів системи .....	141
ДОДАТОК Г Схема електрична принципова плати Raspberry Pi.....	142
ДОДАТОК Д Структурна схема пристрою .....	143
ДОДАТОК Е Алгоритм роботи системи .....	144
ДОДАТОК И Протокол перевірки навчальної (кваліфікаційної) роботи.	146

					08-23.КМКР.029.00.000 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

## ВСТУП

Агропромислова сфера на сьогоднішній день знаходиться в не найкращому стані. Технології що використовуються в цій сфері на протязі останніх років майже не змінюються. Більшість що були винайдені не використовуються в Україні і багатьох інших країнах. Навіть в розвинених країнах незвичайні методи є скоріш рідкістю. Високотехнологічні системи для керування агропромисловими підприємствами вже існують і використовуються але все одно більшість людей продовжує використовувати застарілі методи.

Загалом більшість таких систем є дуже дорогі, не ненадійні і не зручні в використанні. Але якщо і впровадити гарну систему яка буде коштувати дорожче але її недоліки не будуть такими помітними то продуктивність підприємства значно виросте, тому що і зменшиться кількість витрат на паливо, добрива, зерно, і т.д. Також можливо вести спостереження за робітниками і за прогресом роботи що підвищить дисципліну. Використання всіх даних що система отримує допоможе пришвидшити роботу як для керівників так і для робітників.

Метою роботи є розробка інформаційної технології GPS моніторингу та управління технічним забезпеченням агропромислового підприємства, а саме система пристроїв для збору спеціалізованої інформації, її передача на мобільний додаток, а також керування з додатку цими пристроями.

У відповідності до поставленої мети потрібно виконати такі задачі:

- проаналізувати існуючі рішення;
- розробити структурну схему системи;
- розробити алгоритм роботи системи;
- здійснити апаратну реалізацію системи;
- здійснити програмну реалізацію системи.

Об'єктом дослідження є процеси відслідковування об'єктів в режимі реального часу.



Предметом дослідження є методи та засоби відслідковування об'єктів в просторі та використання Інтернету речей в агропромисловій сфері.

Було використано методи проектування програмного та апаратного забезпечення для розробки апаратної частини системи GPS моніторингу.

Наукова новизна отриманих результатів магістерської роботи полягає у реалізації розширеного функціоналу апаратного забезпечення в порівнянні з наявними аналогами на ринку, а також покращене опрацювання різнотипних даних, що дозволяє ефективно застосовувати ІТ та ІоТ технології в агропромисловій сфері.

Практичне значення роботи — це розроблений програмно-апаратний засіб, який дозволяє контролювати агропромислове підприємство, що є дуже корисно для власників і робітників такого підприємства.

За результатами магістерської роботи опубліковані тези доповіді на конференції: “Молодь в науці: дослідження, проблеми, перспективи (МН-2022)”.

Результати доповідей опубліковані у збірниках матеріалів конференцій.

# 1 ОГЛЯД І АНАЛІЗ СУЧАСНИХ ТЕХНІЧНИХ РІШЕНЬ АГРОПРОМИСЛОВОЇ СФЕРИ

## 1.1 Огляд агропромислової сфери

За останні три десятиліття сільське господарство змінилося. Виробництво зросло майже вдвічі швидше, ніж у попередні періоди, що частково сприяло значному збільшенню потенціалу країн, що розвиваються, з виробництва їжі, а частково — змінами в розвинених країнах. Перш за все, у всьому світі відбулися безпрецедентні технічні зміни в сільському господарстві. Поряд з цим чудовим і часом недооціненим досягненням стоїть «світова продовольча проблема».

Сотні мільйонів людей у країнах, що розвиваються, досі не мають достатньої кількості їжі. Зростання населення, наслідки якого часто посилюються вкрай нерівномірним розподілом доходів, різко зменшило як вигоди на душу населення від збільшення виробництва продуктів харчування, так і пов'язане з цим збільшення доходів на душу населення. Цей парадокс бідності серед достатку тривалий час переслідував народне розуміння ролі сільського господарства в економічному розвитку. З одного боку, це призвело до відчуття безнадійності щодо недоїдання у світі; з іншого, до технологічної надмірної впевненості. Надмірне занепокоєння продовольчою кризою чергувалося з прийняттям сільського господарства як належним, навіть нехтуванням ним. Жодна з цих екстремальних реакцій не є доречною або корисною.

Сільське господарство справді стикається з особливими проблемами, з яких особливо виділяються дві. Перший — це біологічна, тісна залежність від природного середовища. Це призводить до рівнів виробництва, які непередбачувано коливаються від сезону до сезону, і зростання, яке сильно варіюється від одного регіону до іншого залежно від природних джерел та ступеня, до якого їхній потенціал уже розвинений. Друга — економічна проблема радикальних перебудов, які стикаються фермерам у процесі розвитку

сільського господарства. Хоча його виробництво може продовжувати зростати, частка сільського господарства як у ВВП, так і в зайнятості незмінно зменшується в міру розвитку та індустріалізації економіки. Дійсно, на пізніх стадіях розвитку абсолютний розмір робочої сили в сільському господарстві зменшується, в кінцевому підсумку до дуже незначної частки від загальної кількості.

Однак у більшості аспектів сільське господарство не є унікальним. Фермери, як і інші працівники, реагують на заохочення. Купуючи продукти харчування, на споживачів впливають відносні ціни та доступний дохід, як і коли вони купують інші речі. Таким чином, успіх чи невдача сільського господарства багато в чому залежить від факторів, які впливають на всю економічну діяльність. Якщо він відстає, то через неадекватні інвестиції, відсутність стимулів і невідповідну політику, а також через проблеми технологій, клімату та ґрунту.

Між сільським господарством і загальним економічним зростанням існує міцний зв'язок. Небагато країн досягли стійкого економічного зростання без попереднього або одночасного розвитку свого сільського господарства. У міру свого розвитку розвиток диктує зниження ролі сільського господарства. Спочатку він домінував, але в кінцевому підсумку на нього припадає лише невелика частка чистого виробництва, зайнятості та доходу. Успішне пристосування до зміни ролі сільського господарства визначає не тільки темпи та характер розвитку в цілому, але й серйозність проблем, які виникають у сільського населення в процесі розвитку. Міжнародне економічне середовище, торгівля, технології, комерційний капітал і допомога в розвитку сприяли серйозним змінам у світовому сільському господарстві.

Виробництво динамічної агропромисловості має важливе значення для створення робочих місць і накопичення капіталу, а також для підвищення якості та попиту на сільськогосподарську продукцію. Агропромисловість може мати несприятливий вплив на міжнародний розвиток, підтри-

муючи та посилюючи економічне зростання країни та пом'якшуючи скорочення бідності як у сільській, так і в міській місцевості країн, що розвиваються.

Сільське господарство та промисловість історично розглядалися як дві різні галузі як з точки зору їх характеристик, так і їх ролі в економічному розвитку. Сільське господарство розглядалося як ознака першого етапу зростання, а ступінь індустріалізації вважається найважливішим показником прогресу країни на шляху розвитку. Сільське господарство вже стало формою промисловості, пов'язаної з технологіями, вертикальна інтеграція, маркетинг і переваги клієнтів вирости відповідно до профілю порівнянних промислових секторів.

Також завжди існує потреба у збалансованому агропромисловому зростанні сільськогосподарського виробництва, оскільки сільськогосподарський капітал стає все більш вразливим до ринкових сил і все більше інтегрується в мережу промислових взаємозалежностей. Тому індустріалізація сільського господарства та розвиток агропереробної промисловості є звичайним процесом, який породжує абсолютно нову форму промислового сектора. Таким чином, агропереробка означає перетворення товарів, отриманих від сільського, лісового та рибного господарства. Дуже велика частина сільськогосподарського виробництва зазнає певного ступеня трансформації між збиранням врожаю та кінцевим використанням. Вони варіюються від простого консервування (наприклад, сушіння на сонці) і пов'язаного збирання врожаю до виробництва, до сучасних, капіталомістких методів. Харчова промисловість набагато більш порівнянна та легша для ідентифікації, ніж нехарчова промисловість, оскільки всі їхні товари мають однакове кінцеве використання. Сьогодні, коли ми будемо намагатися тлумачити світ агропромисловості, вона продовжує переробляти основні сільськогосподарські продукти, а також інтегрувати високотехнологічні промислові ресурси, які часто є результатом значних інвестицій у науку, технології та інновації. Цій зростаючій складності вхідних даних відповідає поетапна ескалація

процесів перетворення, що характеризується як фізичними, так і хімічними змінами, які ще більше покращують прибутковість і товарність перед кінцевим використанням[1].

Сільське господарство України має значні природно-економічні можливості для збільшення виробництва та експорту агропродовольчої продукції. По-перше, це родючі, переважно чорноземні ґрунти. По-друге, сприятливі кліматичні умови для вирощування більшості сільськогосподарських культур і тварин. Нарешті, епоха хлібо-рабовласницьких традицій українських селян, досить високий професійний рівень працівників сільського господарства, харчової промисловості, аграрної науки.

Агропромисловість дуже допомагає в нашій економічній стабільності, оскільки допомагає забезпечити зайнятість у позасільськогосподарських видах діяльності, таких як переробка. Агропромисловість також додає вартість і збільшує попит на продукцію фермерів, тим самим зменшуючи бідність і відсутність продовольчої безпеки та стимулюючи економічне зростання.

Агропромисловість може мати реальний вплив на міжнародний розвиток шляхом збільшення економічного зростання та зниження рівня бідності як у сільській, так і в міській місцевості країн, що розвиваються. Однак для того, щоб обійти вплив на вразливі країни та населення, необхідні розумні політики та стратегії сприяння розвитку агропромисловості за участі спеціалістів агропромисловості, наукових експертів та технічних установ ООН, у розділах розглядаються стратегії та дії, необхідні для стимулювання розвитку конкурентоспроможності агропромислового комплексу, яка може створювати дохід і створювати робочі місця[2].

## 1.2 Технічні рішення і технології

Загальновідомо, що стійка індустріалізація відіграє важливу роль у прискоренні економічного зростання та соціального розвитку. Він перерозподіляє робочу силу та інші ресурси з трудомістких і менш продуктивних видів діяльності на більш капіталомісткі та технологічні, а також відкриває

великі можливості для країн, що розвиваються, додавати більше вартості на місцевому рівні, входячи на нижній кінець глобальних ланцюгів вартості. Різниця в рівнях життя між країнами часто пояснюється тим, що країни, що розвиваються, використовують більшу частку своєї робочої сили в сільському господарстві, а також тим, що сільське господарство в країнах, що розвиваються, є набагато менш продуктивними, ніж у розвинених країнах.

Науково-технічний прогрес призвів до появи нових проривних технологій: мобільного інтернету, штучного інтелекту, інтернету речей, хмарних технологій, прогресивних роботів, автономних та напіваавтономних транспортних засобів, генної індустрії, акумулювання та накопичення енергії, 3-D друку, матеріалів нового покоління, відновлюваної енергії, розвідки, видобутку та поповнення нафти та газу. Слід вважати перспективним використання більшості з них в агропромисловому комплексі. Вони забезпечать не лише вихід із кризи цього кластеру, а й інноваційний розвиток, кардинально перетворивши аграрну сферу. Очікується збільшення наукового, екологічного, інформаційно-комунікаційного потенціалів, підвищення технічної оснащеності, розвиток виробничої та соціальної інфраструктури, збереження природного потенціалу. Початкове зниження трудового та соціального потенціалів за рахунок відмирання низки професій, скорочення низькокваліфікованих працівників та загострення їх соціальної незахищеності має компенсуватися покращенням якості життя та праці, затребуваністю фахівців нових для агропромислового комплексу спеціальностей, підвищенням продуктивності праці. Перехід до інтенсифікації аграрного виробництва на основі створення єдиної інформаційно-комунікаційної системи дистанційного управління господарською діяльністю та соціального забезпечення, логістики, консультаційних служб, освіти, державних послуг, роботизації та автоматизації технологічних процесів, генної індустрії, використання нових технологій, матеріалів, відновлюваної енергії реалізує гармонійну взаємодію людини та біосфери. У перспективі агропромисловий кластер бачиться

інноваційним та високотехнологічним, здатним забезпечити продовольчу та національну безпеку.

Проривні технології — це досягнення, які змінюють життя, бізнес і глобальну економіку. Вони сприяють широкомасштабним економічним реформам і проривам протягом наступних років. Нині існують такі гілки технологій в агропромисловому секторі:

- мобільний Інтернет;
- штучний інтелект;
- Інтернет речей;
- хмара;
- передова робототехніка;
- автономні та напівавтономні транспортні засоби;
- геноміка наступного покоління;
- накопичувачі енергії;
- GPS.

Розробка та впровадження точного землеробства або специфічного землеробства стали можливими завдяки поєднанню глобальної системи позиціонування (GPS) та систем географічної інформації (GIS). Ці технології дозволяють поєднувати збір даних у реальному часі з точною інформацією про місцезнаходження, що веде до ефективного маніпулювання та аналізу великих обсягів геопросторових даних. Додатки на основі GPS у точному землеробстві використовуються для планування ферми, картографування полів, відбору проб ґрунту, керування трактором, розвідки врожаю, застосування змінних норм і картування врожайності. GPS дозволяє фермерам працювати в умовах низької видимості, таких як дощ, пил, туман і темрява. Раніше фермерам було важко співвідносити виробничі технології та врожайність з мінливістю землі. Це обмежило їхню здатність розробляти найефективніші стратегії обробки ґрунту/рослин, які могли б підвищити їхнє виробництво.

Сьогодні точніше застосування пестицидів, гербіцидів і добрив, а також кращий контроль над розповсюдженням цих хімікатів можливе завдяки

точному землеробству, що дозволяє знизити витрати, отримувати вищий урожай і створювати екологічно чисте господарство[3]. Точне землеробство зараз змінює погляди фермерів та агробізнесів на землю, з якої вони отримують прибутки.

Точне землеробство — це збір своєчасної геопросторової інформації про потреби ґрунту, рослини та тварин, а також призначення та застосування специфічних методів обробки для збільшення сільськогосподарського виробництва та захисту навколишнього середовища. Там, де фермери, можливо, колись обробляли свої поля однаково, тепер вони бачать переваги від мікроменеджменту своїх полів.

Точне землеробство набирає популярності значною мірою завдяки впровадженню в сільськогосподарську спільноту високотехнологічних інструментів, які є більш точними, економічно ефективними та зручними для користувача. Багато з нових інновацій покладаються на інтеграцію бортових комп'ютерів, датчиків збору даних і систем відліку часу та положення GPS. Багато хто вважає, що переваги точного землеробства можуть бути реалізовані лише на великих фермах з величезними капіталовкладеннями та досвідом роботи з інформаційними технологіями. Це не так. Існують недорогі та прості у використанні методи та методика, які можуть бути розроблені для використання всіма фермерами.

За допомогою GPS, ГІС та дистанційного зондування можна збирати інформацію, необхідну для покращення використання землі та води. Фермери можуть отримати додаткові переваги, поєднуючи краще використання добрив та інших змін ґрунту, визначаючи економічний поріг для боротьби зі шкідниками та бур'янами, а також захищаючи природні ресурси для майбутнього використання. Виробники GPS-обладнання розробили кілька інструментів, які допомагають фермерам і агробізнесу стати більш продуктивними та ефективними у точному землеробстві. Сьогодні багато фермерів використовують продукти, отримані від GPS, для покращення роботи свого



сільськогосподарського бізнесу. Інформація про місцезнаходження збирається GPS-приймачами для відображення меж полів, доріг, зрошувальних систем і проблемних зон посівів, таких як бур'яни або хвороби. Точність GPS дозволяє фермерам створювати карти ферм з точними площами для польових ділянок, розташування доріг і відстаней між цікавими точками. GPS дозволяє фермерам з року в рік точно орієнтуватися в певних місцях на полі, збирати зразки ґрунту або контролювати стан посівів.

Консультанти з сільськогосподарських культур використовують надійні пристрої збору даних із GPS для точного визначення місцезнаходження, щоб картувати зараження шкідниками, комахами та бур'янами на полі. Проблемні зони шкідників у посівах можуть бути точно визначені та нанесені на карту для майбутніх управлінських рішень та рекомендацій щодо введення. Ці ж польові дані також можуть використовуватися авіаційними обприскувачами, що дає можливість точного розкосу полів без використання людських «похибок» для їх наведення. Зерноприбиральники, оснащені GPS, здатні точно літати по полю, вносячи хімікати лише там, де це необхідно, мінімізуючи хімічний дрейф, зменшуючи кількість необхідних хімікатів, тим самим приносячи користь довкіллю. GPS також дозволяє пілотам надавати фермерам точні карти. Фермери та постачальники послуг сільського господарства можуть очікувати ще подальших покращень, оскільки GPS продовжує модернізуватися[4].

На сьогоднішній день NAVSTAR (Navigation Satellite Time and Ranging) базується на 31 супутнику (всього 32), що працюють в одній мережі та обертаються на шести різних кругових орбітах під кутом  $60^\circ$  один до одного. На кожній орбіті 4 супутники, висота орбіти приблизно 20 200 км, а період обертання кожного супутника навколо Землі 12 годин. Тому з будь-якої точки на поверхні Землі зазвичай одночасно видно від чотирьох до дванадцяти таких супутників. Супутниками керують клітини, розташовані на Землі. Ці станції розташовані в Колорадо-Спрінгс, Дієго-Гарсія, на

острові Вознесіння, на атолі Куейлайн і на Гаваях. Вся інформація, що проходила через ці станції, була записана ними і передана на головну станцію на військовій базі Falcon в Колорадо. GPS-приймач обчислює власне місцезнаходження, вимірюючи час проходження сигналу від супутника GPS. Кожен супутник постійно надсилає повідомлення, що містить інформацію про погоду, точку орбіти супутника, з якого було надіслано повідомлення (ефемериди), а також загальний стан системи та приблизні дані орбіти всіх інших супутників GPS (альманах). Ці сигнали поширюються зі швидкістю світла в космосі (і з трохи меншою швидкістю - в атмосфері). Приймач визначає час затримки сигналу та обчислює відстань до супутника, на підставі чого визначає його місцезнаходження методом трилатерації (рисунок 1.1).

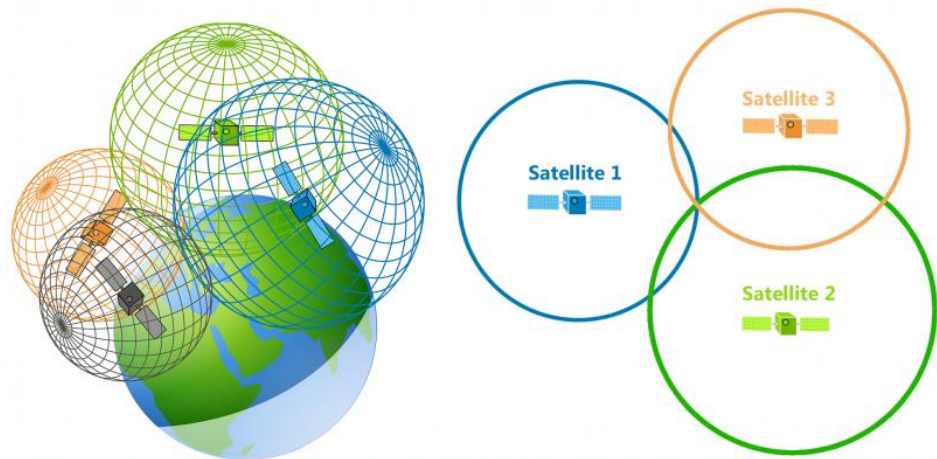


Рисунок 1.1 – Розташування за трилатерацією

Отримані координати перетворюються у наочну форму (широти і довготи або положення на карті) і відображаються користувачеві. Теоретично, щоб визначити власні координати, достатньо визначити відстань до трьох супутників. Однак, щоб розрахувати позицію, потрібно знати час з великою точністю. Щоб виключити потребу в високоточних годинниках, отримувати інформацію від 4 і більше супутників, тобто. Приймач GPS використовує чотири параметри для обчислення чотирьох невідомих:  $k$ ,  $i$ ,  $z$  і  $t$ . У деяких випадках це можна зробити з меншою кількістю супутників. Якщо одна змінна відома заздалегідь (наприклад, висота корабля в океані дорівнює 0),

приймач може обчислити позицію, використовуючи дані трьох супутників. На практиці приймач використовує різну допоміжну інформацію для розрахунку позиції з меншою точністю при відсутності одночасно чотирьох супутників.

Незважаючи на те, що проекти по створенню GPS-систем виконувались військовими відомствами, зараз, окрім спеціальних приймачів, пристрої вбудовуються в різну дрібну побутову техніку: годинники, мобільні телефони, кишенькові радіоприймачі, ноутбуки та фотоапарати, якими можна користуватися для навігації на місці або зйомки місцезнаходження користувача. Ними користуються альпіністи, рятувальники, туристи.

Споживачам також пропонують різноманітні пристрої та програмні продукти, які дозволяють їм:

- відображати своє місцезнаходження на електронній карті;
- прокладати маршрути з урахуванням дорожніх знаків, дозволених поворотів і навіть заторів;
- шукати на карті конкретні будинки та вулиці, атракціони, кафе, лікарні, заправні станції та іншу інфраструктуру.

Окремий клас обладнання — GPS-трекери — призначений для отримання інформації про рух обладнаних ними автомобілів або інших рухомих об'єктів на спостережному посту. Використання GPS-трекерів дає змогу будувати системи контролю для моніторингу та контролю руху, системи GPS-моніторингу руху.

Системи картографування GPS допомагають описати характеристики районів польоту, які перебувають у інтенсивному сільськогосподарському використанні. Ви можете точно порівняти такі характеристики, як мікроклімат, тип ґрунту, посівні площі, пошкоджені комахами чи хворобами, обсяг виробництва.

Можна виділити 2 типи системи GPS-моніторингу рухомих об'єктів.

Системи Offline використовують принцип «чорного ящика», який фіксує події, прив'язуючи кожен з них до географічних координат і реального

часу. Після прибуття об'єкта на базу вся інформація з такого «чорного ящика» зчитується, розшифровується й стає доступною для детального аналізу та агрегації. Перевагами такого підходу є велика кількість різномірної інформації, яка збирається пристроєм (обмежена тільки об'ємом власної пам'яті пристрою), відсутність абонентської плати за передачу даних (передача здійснюється або через фізичне підключення пристрою до комп'ютера диспетчера, або через локальні бездротові мережі). Недоліком такого способу є те що інформація доступна тільки після прибуття об'єкта на базу.

Системи Online використовують принцип радіомаяків — інформація про місцезнаходження не фіксується в пам'яті приладів, а передається в реальному часі диспетчеру. Каналами передачі даних в такому разі може виступати GSM, SMS-повідомлення, GPRS, радіозв'язок, мережа бездротових точок доступу. Перевагами такого підходу є те що інформація про переміщення та стан об'єкта доступні диспетчеру в реальному часі (можлива затримка обумовлена способом комунікацій). Час відгуку пристрою моніторингу залежить від каналу передачі даних (завжди є можливість підібрати компромісний варіант за співвідношенням ціна/оперативність).

Недоліком такої системи є необхідність оплачувати послуги зв'язку (розмір плати змінюється від кількох центів за SMS-повідомлення в межах мережі свого стільникового оператора до декількох доларів за GPRS-з'єднання в роумінгу).

Також варто враховувати точність вимірів так як через відстань між приймачем і супутниками точність розрахунку позиції залежить від багатьох факторів і визначається лише з певною ймовірністю. Супутникові радіосигнали можуть бути екрановані або відбиті від навколишнього середовища приймача, що збільшує часові похибки та спотворює результат вимірювання. Першорядне значення мають атмосферні явища і поточне розташування супутників відносно приймача. Похибка в розрахунку позиції буде більшою, якщо всі доступні супутники згруповані в одній півкулі відносно

приймача в порівнянні з ситуацією, коли приймач може приймати супутникові сигнали з різних напрямків. Ситуація з обмеженою видимістю супутників досить поширена в містах через екранування сигналів від будівель. Звичайна точність сучасних GPS-приймачів у горизонтальній площині становить 5-10 метрів і 10-20 метрів у висоту, але за певних умов розрахована приймачем позиція може короткочасно відрізнятись на значно більші значення. Виробники GPS-приймачів визначають значення похибки положення таким чином: не гірше 5 метрів за 50% часу спостереження і не гірше 8 метрів у 90% часу, похибка визначення швидкості не більше 0,06 PC. У США та Канаді є станції WAAS, а в Європі – станції EGNOS, які передають коригування для диференціального режиму, що дозволяє підвищити точність розрахунку позиції до 1-2 метрів. При використанні більш складних аксесуарів точність координат можна підвищити до 10 см. Наприклад, щоб GPS-приймач працював у диференціальному режимі, він повинен постійно отримувати дані від стаціонарного приймача з диференціальною корекцією.

Загальним недоліком використання будь-якої радіонавігаційної системи є те, що за певних умов сигнал може не досягати приймача або надходити зі значними спотвореннями або затримками. Наприклад, визначити своє точне розташування в глибині квартири всередині залізобетонної будівлі, в підвалі або в тунелі практично неможливо. Оскільки робоча частота GPS знаходиться в дециметровому діапазоні радіохвиль, рівень прийому сигналу від супутників може значно погіршитися під густим листям дерев або через дуже високі хмари. Перешкоди від багатьох наземних радіоджерел, а також магнітні бурі можуть заважати нормальному прийому сигналів GPS. Активне глушіння приймачів сигналів GPS ефективно використовувалося для боротьби з націлюванням крилатих ракет під час операцій США та Великобританії в Іраку та «Вирішальної сили» НАТО в Югославії. Це призвело до самознищення крилатих ракет та їх незвичайних польотів на несанкціонованих траєкторіях. Використання в системі GPS цифрових антенних

решіток, що забезпечує формування «нулів» у схемі антенної системи в напрямках джерел активних перешкод, дозволяє більш ефективно виконувати завдання супутникової навігації в умовах активних перешкод. Низький нахил орбіт супутників GPS (приблизно  $55^\circ$ ) значно погіршує точність субполярних областей Землі, оскільки супутники GPS піднімаються низько над горизонтом[5][6].

Сільське господарство вже вийшло далеко за межі того, що ми знаємо як автоматизація. Системи управління зі штучним інтелектом і супутниковим зв'язком стали частиною самодостатніх машин. Датчики в режимі реального часу надають дані про активність машини та навколишнє середовище, хмарні сховища даних доповнюють обмін даними між машинами, закриваючи останні інформаційні прогалини. Головним каталізатором еволюції в агропромисловому комплексі сьогодні є Інтернет речей. Поєднання технологій в області аналізу даних, у розробці технології датчиків і систем автокерування, а також підключених мережевих рішень, систем управління, платформ і додатків, які виводять методи вирощування рослин і тварин на новий рівень. Це проекти зі сфери Інтернету речей в агропромисловому комплексі. Впровадження IoT у рослинництві, а також автоматизованого керування тракторами, систем дистанційного збору даних з полів (вологість, температура, мінералізація), технологій диференційованого внесення добрив на основі даних з безпілотних літальних апаратів, моніторингу сільськогосподарської техніки, управління зрошенням, планування та прогнозування. Масове впровадження цифровізації та Інтернету речей у сільському господарстві обіцяє перетворити галузь, яка найменше постраждала від інформаційних технологій (ІТ), у високотехнологічний бізнес за рахунок вибухового зростання продуктивності та зниження невиробничих витрат.

Інтернет речей у сільському господарстві виник як друга хвиля зеленої революції. Переваги, які фермери отримують від адаптації IoT, подвійні.

Це допомогло фермерам знизити витрати та одночасно підвищити врожайність, покращуючи прийняття фермерами рішень за допомогою точних даних.

Розумне фермерство — це високотехнологічна та ефективна система ведення сільського господарства та вирощування продуктів харчування в екологічний спосіб. Це застосування спільного впровадження підключених пристроїв та інноваційних технологій у сільському господарстві. Розумне фермерство в основному залежить від Інтернету речей, таким чином усуваючи необхідність фізичної роботи фермерів і виробників і таким чином підвищуючи продуктивність усіма можливими способами.

Оскільки останні тенденції розвитку сільського господарства залежать від сільського господарства, Інтернет речей приніс величезні переваги, як-от ефективне використання води, оптимізація витрат та багато іншого. Різницю змінили величезні переваги, які останніми днями стали революційними в сільському господарстві.

Розумне фермерство на основі IoT покращує всю систему сільського господарства, відстежуючи поле в режимі реального часу. За допомогою датчиків і взаємозв'язку Інтернет речей у сільському господарстві не тільки заощадив час фермерів, але й зменшив екстравагантне використання таких ресурсів, як вода та електроенергія. Він контролює різні фактори, такі як вологість, температура, ґрунт тощо, і дає кришталево чітке спостереження в режимі реального часу.

Комп'ютерне зображення передбачає використання сенсорних камер, встановлених у різних куточках ферми, або дронів, обладнаних камерами для створення зображень, які піддаються цифровій обробці зображень. Цифрова обробка зображень є основною концепцією обробки вхідного зображення за допомогою комп'ютерних алгоритмів. Обробка зображень переглядає зображення різної спектральної інтенсивності, наприклад, інфрачервоного випромінювання, порівнює зображення, отримані за певний

період часу, і виявляє аномалії, таким чином аналізуючи обмежувальні фактори та допомагає краще керувати фермами(рисунок 1.2).



Рисунок 1.2 – Зображення поля з дрона

Сільське господарство є однією з головних галузей, що використовують дрони. Дрони, оснащені датчиками та камерами, використовуються для створення зображень, картографування та обстеження ферм. Існують наземні дрони та повітряні дрони. Наземні дрони — це боти, які оглядають поля на колесах. Повітряні дрони, офіційно відомі як безпілотні літальні апарати (БПЛА) або безпілотні авіаційні системи (БПАС), є літаючими роботами. Дронами можна дистанційно керувати віддалено або вони можуть літати автоматично за допомогою програмно-керованих планів польотів у своїх вбудованих системах, працюючи у координації з датчиками та GPS. На основі даних дронів можна отримати уявлення про здоров'я врожаю, зрошення, обприскування, посадки, ґрунт і поле, підрахунок рослин та прогнозування врожайності та багато іншого. Дрони можна або запланувати для обстеження ферм (дрон як послуга), або їх можна придбати та зберігати поблизу ферм, де їх можна заряджати та обслуговувати.

Датчики, розміщені вздовж ферм, контролюють посіви на предмет зміни освітленості, вологості, температури, форми та розміру(рисунок 1.3). Будь-яка аномалія виявляється датчиками, аналізується, і фермер повідомляється. Таким чином, дистанційне зондування може допомогти запобігти



поширенню хвороб і стежити за зростанням сільськогосподарських культур. Дані, зібрані датчиками щодо вологості, температури, вологості опадів і виявлення роси, допомагають визначити погодні умови у господарствах, щоб вирощувати відповідні культури. Аналіз якості ґрунту допомагає визначити поживну цінність і більш посушливу площу господарств, дренажну здатність або кислотність ґрунту, що дозволяє регулювати кількість води, необхідної для поливу, та вибрати найбільш вигідний вид вирощування.



Рисунок 1.3 – Розумний датчик вологості ґрунту

Нижче наведено переваги впровадження нової технології - Інтернету речей у сільському господарстві.

Клімат відіграє дуже важливу роль для сільського господарства. А відсутність належних знань про клімат сильно погіршує кількість та якість продукції рослинництва. Але рішення IoT дають змогу знати погодні умови в режимі реального часу. Датчики розміщуються всередині та зовні сільськогосподарських полів. Вони збирають дані про навколишнє середовище, які використовуються для вибору правильних культур, які можуть рости та підтримуватися в конкретних кліматичних умовах. Вся екосистема IoT складається з датчиків, які можуть дуже точно визначати погодні умови в режимі реального часу, такі як вологість, опади, температура тощо. Є численні ні. доступних датчиків для виявлення всіх цих параметрів і відповідної настройки відповідно до ваших вимог до розумного господарства. Ці датчики

контролюють стан посівів і погоду навколо них. У разі виявлення будь-яких тривожних погодних умов надсилається сповіщення. Що усувається, так це потреба фізичної присутності під час тривожних кліматичних умов, що в кінцевому підсумку підвищує продуктивність і допомагає фермерам отримувати більше вигод від сільського господарства.

Точне землеробство/точне землеробство є одним із найвідоміших застосувань IoT у сільському господарстві. Це робить сільськогосподарську практику більш точною та контрольованою, реалізуючи інтелектуальні додатки для сільського господарства, такі як моніторинг худоби, відстеження транспортних засобів, спостереження поля та моніторинг інвентарю. Метою точного землеробства є аналіз даних, отриманих за допомогою датчиків, і відповідна реакція. Precision Farming допомагає фермерам генерувати дані за допомогою датчиків і аналізувати цю інформацію, щоб приймати розумні та швидкі рішення. Існує безліч методів точного землеробства, таких як управління зрошенням, управління худобою, відстеження транспортних засобів та багато інших, які відіграють життєво важливу роль у підвищенні ефективності та результативності. За допомогою точного землеробства ви можете аналізувати стан ґрунту та інші пов'язані параметри для підвищення ефективності роботи. Крім того, ви також можете виявляти умови роботи підключених пристроїв у режимі реального часу для визначення рівня води та поживних речовин.

Щоб зробити наші теплиці розумними, IoT дозволив метеостанціям автоматично коригувати кліматичні умови відповідно до певного набору інструкцій. Впровадження Інтернету речей у теплицях усунуло людське втручання, що робить весь процес економічно ефективним і водночас підвищує точність. Наприклад, за допомогою датчиків IoT на сонячних батареях будуються сучасні та недорогі теплиці. Ці датчики збирають і передають дані в режимі реального часу, що допомагає дуже точно контролювати стан теплиці в режимі реального часу. За допомогою датчиків можна контролювати споживання води та стан теплиці за допомогою електронних листів або

SMS-повідомлень. Автоматичне та розумне зрошення здійснюється за допомогою IoT. Ці датчики допомагають надавати інформацію про тиск, вологість, температуру та рівні освітлення.

Звичайна система баз даних не має достатньо місця для зберігання даних, зібраних із датчиків Інтернету речей. Хмарне сховище даних і наскрізна платформа IoT відіграють важливу роль у системі розумного сільського господарства. За оцінками, ці системи відіграють важливу роль у тому, що можна виконувати кращі дії. У світі IoT датчики є основним джерелом збору даних у великих масштабах. Дані аналізуються та перетворюються на значущу інформацію за допомогою інструментів аналітики. Аналітика даних допомагає в аналізі погодних умов, умов утримання худоби та посівів. Зібрані дані використовують технологічні інновації і, таким чином, приймають кращі рішення. За допомогою пристроїв IoT ви можете дізнаватися про стан посівів у режимі реального часу, збираючи дані з датчиків. Використовуючи прогнозу аналітику, ви можете отримати уявлення, щоб приймати кращі рішення щодо збирання врожаю. Аналіз тенденцій допомагає фермерам знати майбутні погодні умови та збирання врожаю. Інтернет речей у сільськогосподарській промисловості допоміг фермерам підтримувати якість врожаю та родючість землі, таким чином підвищуючи обсяги та якість продукції.

Технологічні досягнення майже революціонізували сільськогосподарські операції, а впровадження сільськогосподарських безпілотників є тенденційним зривом. Наземні та повітряні дрони використовуються для оцінки здоров'я посівів, моніторингу посівів, посіву, обприскування посівів та аналізу поля. Завдяки належній стратегії та плануванню на основі даних у реальному часі технологія дронів дала високий підйом і змінила сільськогосподарську галузь. Дрони з тепловими або мультиспектральними датчиками визначають ділянки, які потребують змін у зрошенні. Як тільки культури починають рости, датчики показують їх здоров'я та обчислюють їх вегетацій-

ний індекс. Згодом розумні дрони зменшили вплив на навколишнє середовище. Результати були такими, що відбулося значне скорочення, і набагато менше хімічних речовин потрапляє до підземних вод.

Деякі факти про цифровізацію сільського господарства наведено знизу.

Згідно з опитуванням американських фермерів у 2015 році, лідерами серед цифрових рішень стали агрохімічний аналіз ґрунту (98% респондентів використовують його у своїй роботі), системи керування GPS для тракторів (60%) та використання супутникового картографування (30%). впроваджено в сільськогосподарський бізнес.

Перспективи досліджень точного землеробства дуже привабливі. Експерти прогнозують зростання середньої врожайності сільськогосподарських культур у світі на 70% до 2050 року.

Цифрові технології в сільському господарстві з комплексним підходом дозволяють скоротити витрати на добрива, корми, паливо та інші витрати вже зараз майже на 20%. Основні напрямки цифровізації сільського господарства Існує багато перспективних напрямків у сфері Інтернету речей і технологій точного землеробства, які сильно вплинуть на продуктивність. Ми порахуємо деякі з них.

Полювий моніторинг — найцінніша сфера цифровізації. Вона включає моніторинг здоров'я рослин, розвідку сільського господарства за допомогою високошвидкісного Інтернету та мобільних пристроїв, моніторинг продуктивності та прогноз погоди. У цій галузі також активно розвиваються аналізи ґрунтів, діагностика хвороб, шкідників і бур'янів, технології зрошення/меліорації.

Створення цифрових платформ для управління всіма техніками у сільському господарстві є важливим, якщо говорити про точне землеробство. Такі платформи, як локальні, так і хмарні, можуть значно допомогти в плануванні та аналізі даних з багатьох технічних пристроїв або машин.

Автоматизоване розкидання гербіцидів і насіння не нова ідея. Але за допомогою дронів, методів IoT і гіперспектральних камер це можна зробити більш точно.

Агротехнічна автоматизація та цифровізація. Якщо роботи досі є рідкістю на сучасних фермах, то автоматизоване рульове керування для тракторів, комбайнів чи культиваторів — реальність. Це поле включає роботизовані алгоритми для позбавлення від бур'янів, GPS-навігацію та автоматичне керування, телематику, автоматизацію висіву для оптимізації споживання насіння.

Роблячи невеликий висновок, перенесемо всі переваги цифровізації у сільському господарстві. Перший – підвищення продуктивності праці в сільському господарстві та зниження собівартості продукції, підвищення рентабельності бізнесу сільгоспвиробників за рахунок прискорення поставок продукції, підвищення рівня механізації та автоматизації. Це також допомагає виробництву з підрахованими і передбачуваними характеристиками. І, звичайно, це свідомий підхід до раціонального використання природних ресурсів та підтримки їх у збалансованості[7].

### 1.3 Аналіз існуючих аналогів систем

На сьогоднішній день існує не так багато GPS систем моніторингу, а тим більше в агропромисловій сфері. Більшість систем можна оцінити за такими критеріями:

- кількість доступних даних для обробки;
- обчислювальна спроможність системи;
- розширюваність системи;
- методи передачі даних;
- зручність у використанні;
- ціна.

На перший погляд всі GPS системи моніторингу схожі за своїм принципом дії. Це повинен бути комплекс приладів де за основу взятий мікроконтролер та GPS антена для передачі місцезнаходження.

Але в деталях — всі вони різні. Більшість систем можуть підтримувати тільки обмежені за вибором додаткові прилади, або ж мають різні методи передачі даних та стовідсотково зовсім різні у зручності використання та ціні.

Для порівняння були взяті найпопулярніші та найрозповсюдженіші системи на ринку а саме: Logist.ua – TMS система, AggreGate Fleet Manager, система моніторингу від GPS-GROUP, Teletrack-AGRO та можливо не дуже повноцінна, але розповсюджена Система паралельного водіння Raven Cruizer II.

Logist.UA — TMS система керування транспортом підприємства. Автоматизація управління доставками, планування маршрутів та моніторингу пересування автотранспорту підприємства

Програма ефективно автоматизує керування транспортом на підприємстві. Впровадження TMS системи Logist.UA від SystemGroup дозволяє скоротити час на планування, оптимізувати маршрути і знизити витрати на ПММ і доставки.

Система TMS Logist.UA автоматизує заявки на доставку (автоматизована обробка та облік), маршрути (планування оптимальних маршрутів та рейсів), контроль (GPS моніторинг пересування та ходу доставки план/факт). Присутня інтеграція та обмін даними із системами WMS, YMS, ERP.

AggreGate Fleet Manager — це система керування транспортом, розроблена для керування парком легкових та вантажних автомобілів, водними суднами, вилочними навантажувачами або іншими видами транспорту. Система здатна відстежувати розташування активів і централізовано збирати, зберігати, обробляти і візуалізувати різні телеметричні дані.

Кожен транспортний засіб оснащений апаратним пристроєм, який збирає різні дані про поведінку водія. Ці дані включають швидкість руху, максимальне прискорення і удари, що зазнають транспортні засоби при русі, а також час шляху, час на холостому ході. Всі дані зберігаються в пам'ять пристрою для вивантаження їх на комп'ютер зазвичай через бездротовий інтерфейс. Пристрій також працює як контролер доступу водіїв до транспортного засобу через коди доступу або карти RFID.

Доступ водія до автомобіля також може бути обмежений за часом, наприклад, «тільки через певні проміжки», «тільки в робочий час» та інші. Крім того, кожен вхід або вихід із транспортного засобу записується. У поєднанні із записами про керування, це надає можливості проведення детального аудиту. Система також може виконувати ряд додаткових функцій, таких як моніторинг рівня палива, заряд батарей та інші. У критичних ситуаціях автомобіль може бути негайно зупинено прямо із ситуаційного центру.

За допомогою AggreGate ви зможете керувати декількома транспортними засобами як однією групою, незважаючи на те, що кожне з них може бути обладнане різною моделлю/версією контролера. Ви можете внести одну зміну в налаштування (наприклад, поріг перевищення швидкості), яке пошириться на всі транспортні засоби при їх підключенні.

Компанія GPS-GROUP надає рішення для оптимізації бізнесу за допомогою системи навігації. Впровадження моніторингу сільгосп спецтехніки в режимі реального часу, що в свою чергу дозволяє отримати контроль:

- швидкості, місцезнаходження та напрямки пересування всіх видів спецтехніки;
- траєкторії пересування і швидкості сільгосп техніки під час виконання робіт;
- подачі інформації за планом поля та його меж, замір площі;
- розкладу, планування і виконання робіт;
- виконаної роботи, час виконання роботи об'єктом і розміщення його на межах зазначеної ділянки;

- оброблення країв і гербіцидів під час посіву;
- перевезень врожаю на місця розвантаження;
- отримання перевірених даних на простоях, лівих рейсів, зливів палива;
- отримання тривожних сигналів про відхилення маршруту або графіка виконання робіт.

Крім того дозволяє вирішити різні позаштатні ситуації, дозволяє втрутитися і запобігти їм. Є 2 варіанти:

- Двосторонній голосовий зв'язок між водієм і диспетчером, яка дозволить вмішуватися і оперативно вирішити проблему.
- Дистанційне блокування двигуна, якщо сталося викрадення, диспетчер може заблокувати двигун і викликати підкріплення.

Крім того дозволяє контролювати стан вантажів під час транспортування. Система здійснює ідентифікацію транспорту, який під'їжджає на вивантаження до комбайна. Показує пройдений маршрут пересування і точку прибуття транспорту для розвантаження, а також точку вивантаження врожаю.

З допомогою клієнтського застосунку можна контролювати наступне:

- місце розташування транспортного засобу в тому числі і спецтехніки;
- пробіг;
- час та місце стоянок;
- швидкість пересування;
- кількість пального в баці, а також заправки і сливи.

Рішення "Teletrack-AGRO" дозволяє засобами GPS моніторингу контролювати траєкторію руху техніки на полі, швидкісні режими, зупинки, простої, обсяги заправок і зливів палива, витрати палива, фактично оброблену площу, робочий час водіїв, глибину заглиблення інструмента та інші параметри.

Результати впровадження системи GPS моніторингу Teletrack-AGRO



- зниження витрат на ПММ та амортизаційні витрати;
- мінімізація втрат на етапі збору та доставки врожаю;
- отримання об'єктивних даних про хід і якість польових робіт;
- підвищення продуктивності праці та поліпшення дисципліни персоналу;
- поліпшення якості обліку виконаних робіт;
- підвищення ефективності роботи техніки за рахунок оптимального планування маршрутів.

Для ідентифікації водіїв та контролю робочого часу в кабіні транспортного засобу встановлюється спеціальний пристрій, що представляє собою RFID-зчитувач безконтактних карт. Кожному водієві видається персональна карта. В системі зберігаються дані про співробітника (прізвище, ім'я та ін.). Приступаючи до роботи, водій реєструється, вставляючи карту в зчитувач, а в кінці зміни виймає картку з зчитувача. Таким чином однозначно визначається початок, кінець і тривалість зміни.

При наявності на сільгосптехніці GPS-приймача можливе підключення до нього GPS-контролера і використання більш точних координат. Це може бути корисно для отримання більш точної траєкторії руху при визначенні перекриттів і пропусків.

Система GPS моніторингу "Teletrack-AGRO" дозволяє забезпечити поетапний контроль за переміщенням зібраного врожаю, техніки і водіїв і, таким чином, виявити нестачу і визначити порушників. Реалізується ця задача за допомогою застосування радіоміток, RFID-зчитувачів водіїв, датчиків вимірювання рівня зерна в бункері і контролю роботи шнека комбайна, перевантажувача. Система дозволяє контролювати траєкторію пересування комбайна на полі, заповнення бункера комбайна і перевантажувача зерном, місце, час перевантаження в зерновоз (перевантажувач), обсяг вивантаження, а також відстежити його маршрут прямування, несанкціоновані зупинки, відхилення від маршруту та кількість привезеного врожаю на елеватор.

Контроль зерновогоза (перевантажувача) відбувається за допомогою автономних радіоміток. Приймальна частина підключається до GPS-контролера, живиться від борт мережі і встановлюється на комбайн або перевантажувач, або в пункті прийому зерна (на елеваторі). Передавальна частина закріплюється на перевантажувач, зерновоз і забезпечує передачу сигналу з ID-кодом на задану відстань (3-10 метрів), живиться від вбудованого акумулятора, що забезпечує час безперервної роботи від 5 років. Таким чином можна простежити ланцюжок надходження врожаю від комбайна до елеватора. А вимірювання кількості зерна в комбайні (зерновозі) за допомогою ультразвукового або контактного датчиків і порівняння його з привезеним на елеватор дозволяє обчислити нестачу продукції при транспортуванні.

Система паралельного водіння Raven Cruiser II (Рисунок 1.4). При використанні навігаційної системи, збільшується робочий день за рахунок роботи вночі, при високій запиленості, затуманеності, взагалі при поганій видимості. USB-порт розширює можливості контролю за якістю та кількістю виконаної роботи. Система паралельного водіння від Raven, Cruiser II – має функцію картографування виконаної роботи. Ще однією особливістю системи паралельного водіння Cruiser II є точність 15-20 см. на стиках, що дає можливість здійснювати посів культур суцільного висіву вночі або за поганої видимості.



Рисунок 1.4 — Система паралельного водіння Raven Cruiser II

Підтримка SBAS безкоштовного сигналу: SBAS (WAAS / EGNOS) — безкоштовна диференційна поправка; Glide-поправка від Novatel на основі математичних розрахунків. Можливість підключення зовнішніх GPS-приймачів Raven S600 (Satellite GS 4 см – платний сигнал). Равен Крузер 2 має внутрішню пам'ять. Можливість збереження даних на USB – накопичувач. Збереження виконаних робіт в форматі SHP (ShapeFile), KML, BMP. Підтримка механічного автопілота (підрулювача) SmarTrax MD. Гідравлічний автопілот SmarTrax на підготовлені та не підготовлені трактори. Автоматичне відключення сікцій обприскувача AccuBoom, автоматичне регулювання висоти штанги AutoBoom. Підключення компютера виливу Raven SCS 440, SCS 450, SCS 460. Автоматичне відключення секцій на сівалці AccuRow. Відображення обробленої площі. Безпроводна передача даних, через GSM-канали зв'язку. Raven SlingShot. Система паралельного водіння Cruiser II від Raven, накопичує детальну інформацію про траєкторію руху агрегату, щільність та якість зробленої роботи. Завдяки USB роз'єму дані можуть зберігатися на зовнішню флеш пам'ять. За допомогою чого, агроном володіє інформацією наскільки якісно виконують роботу.

В результаті порівняння аналогічних систем можемо побачити що всі системи GPS відслідковування мають вагомні недоліки наведені нижче.

Відсутність розширюваності системи за рахунок модульності, а також обмеження по додаткових функціях в системах які дозволяють використовувати зовнішні розширення. Більшість систем є тільки частково підходящими для використання в агропромисловій сфері, так як загалом системи розраховані на встановлення на тільки транспорт для перевезень але не на спец техніку як наприклад комбайн.

Низька гнучкість систем так як вони заточені під роботу з окремими пристроями вибраними під вузькоспеціалізовану задачу, без можливості під'єднання інших приладів доступних на ринку.

Загалом системи можуть керувати значними об'ємами даних але майже всі системам не вистачає можливості оброблювати дані перед відправкою на клієнт що підвищує кількість даних що передаються і збільшує навантаження на кінцевий пристрій.

Відсутність новітнього методу передачі даних. Всі пристрої використовують застарілі протоколи передачі даних, що значно може погіршити швидкодію.

Висока ціна за системи з таким функціоналом і характеристиками. Всі системи є дуже дорогими для впровадження в агропромислову сферу і з наданим функціоналом займе багато часу щоб «відбити» ціну цих систем. Також системи що мали досить значний функціонал при не дуже великих недоліках мають ще більшу ціну порівняно з іншими аналогами.

#### 1.4 Обґрунтування доцільності розробки

Після огляду агропромислової сфери в першому підрозділі можна виділити що становище аграрної сфери в Україні та в світі знаходиться не в найкращому стані, тобто технологічний розвиток відбувається повільно. Вводяться нові засоби для покращення становища але введення в обіг цих технологій є повільний.

Було оглянуто новітні технології в агропромисловій сфері що вже використовуються або ж що ще знаходяться на стадії тестування і тільки починають формуватись. Після огляду цих технологій було вибрано напрямок досліджень і сформована мета розробки системи.

Було проведено аналіз альтернативних систем на ринку відповідно до визначеної системи в меті розробки. Альтернативні рішення було порівняно та виявлено низку вагомих недоліків. Виходячи з цього можна виділити низку особливостей що повинна мати розроблювана система які можна використати для створення покращеного продукту з відсутніми недоліками.

З вказаних вище досліджень можна визначити те, що створення системи є актуальним.

## 2 РОЗРОБКА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ GPS МОНІТОРИНГУ ТА УПРАВЛІННЯ

### 2.1 Вибір мікропроцесорної платформи

Мікроконтролер — це компактна інтегральна схема, призначена для керування певними операціями у вбудованій системі. Типовий мікроконтролер включає в себе процесор, пам'ять і ввід-вивод для однокристальних периферійних пристроїв. Іноді їх називають вбудованими контролерами або мікроконтролерами (MCU), серед інших пристроїв, мікроконтролери зустрічаються в транспортних засобах, роботах, офісних машинах, медичних пристроях, мобільних радіоприймачах, торгових автоматах і побутових приладах.

По суті, це прості мініатюрні персональні комп'ютери (ПК), призначені для керування невеликими можливостями більшого компонента без складної операційної системи (ОС). Мікроконтролер вбудований в систему управління однією функцією в пристрої. Це робиться шляхом інтерпретації даних, отриманих від периферійних пристроїв вводу-виводу за допомогою ЦП. Тимчасова інформація, отримана від мікроконтролера, зберігається в його пам'яті даних, де процесор отримує до неї доступ і використовує інструкції, що зберігаються в його програмній пам'яті, для розшифрування та використання вхідних даних. Потім він використовує свою периферійну мережу для спілкування та вжиття відповідних дій.

Мікропроцесори використовуються в широкому спектрі систем і пристроїв. Пристрої часто використовують кілька мікроконтролерів, які працюють разом у пристрої для вирішення пов'язаних завдань. Наприклад, автомобіль може мати багато мікроконтролерів, які керують різними окремими системами всередині, такими як антиблокувальна система, контроль тяги, контроль упорскування палива або підвіска. Усі мікроконтролери спілкуються один з одним, щоб повідомляти про правильні дії. Деякі можуть спілкуватися з більш складним центральним комп'ютером в автомобілі, а

інші можуть спілкуватися лише з іншими мікроконтролерами. Вони надсилають та отримують дані за допомогою периферійних пристроїв введення-виводу та обробляють ці дані для виконання призначених завдань.

Перелік основних елементів мікроконтролера вказаний нижче.

Процесор (CPU) — процесор можна розглядати як мозок пристрою. Він обробляє і реагує на різні інструкції, які керують функцією мікроконтролера. Це включає виконання основних арифметичних, логічних і операцій введення-виводу. Він також виконує операції передачі даних, які передають команди іншим компонентам більшої вбудованої системи.

Пам'ять мікроконтролера використовується для зберігання даних, отриманих від процесора, і використовується для відповіді на інструкції, запрограмовані на виконання. Мікроконтролер має два основних типи пам'яті:

Програма пам'яті, яка зберігає довготривалу інформацію про інструкції, що виконуються процесором. Пам'ять програм — це енергонезалежна пам'ять, тобто. зберігає інформацію протягом тривалого часу без потреби в джерелі живлення.

Пам'ять даних потрібна для тимчасового зберігання даних під час виконання інструкцій. Пам'ять даних змінна, тобто. дані, які він зберігає, є тимчасовими і зберігаються, лише якщо пристрій підключено до джерела живлення.

Периферійні пристрої введення-виводу - пристрої вводу-виводу є інтерфейсом процесора до зовнішнього світу. Вхідні порти отримують інформацію і відправляють її в процесор у двійковій формі. Процесор отримує ці дані і відправляє необхідні інструкції на вихідні пристрої, які виконують завдання за межами мікроконтролера.

Хоча центральний процесор, пам'ять і периферійні пристрої вводу-виводу є ключовими елементами мікропроцесора, є й інші елементи, які часто включають. Термін периферійні пристрої вводу-виводу просто відноситься до допоміжних компонентів, які взаємодіють з пам'яттю і процесором. Існує

багато допоміжних компонентів, які можна віднести до периферійних пристроїв. Наявність певного прояву периферійного введення-виведення є елементом мікропроцесорів, оскільки вони є механізмом, за допомогою якого використовується процесор.

Інші допоміжні елементи мікроконтролера включають:

— аналого-цифровий перетворювач (АЦП) — схема, яка перетворює аналогові сигнали в цифрові, що дозволяє процесору в центрі мікроконтролера взаємодіяти із зовнішніми аналоговими пристроями, такими як датчики;

— цифровий аналоговий перетворювач (ЦАП) — ЦАП виконує зворотну функцію АЦП і дозволяє процесору в центрі мікроконтролера передавати свої вихідні сигнали зовнішнім аналоговим компонентам;

— системна шина являє собою з'єднувальний провід, який з'єднує всі компоненти мікроконтролера між собою;

— послідовний порт — один із прикладів вводу-виводу, який дозволяє мікроконтролеру підключатися до зовнішніх компонентів, які мають функцію, подібну до USB або паралельного порту, але відрізняються способом обміну бітами.

Процесор мікроконтролера залежить від програми. Варіанти варіюються від простих 4-розрядних, 8-розрядних або 16-розрядних процесорів до більш складних 32-розрядних або 64-розрядних процесорів. Мікроконтролери можуть використовувати такі типи енергонезалежної пам'яті, як оперативна пам'ять (ОЗП) і енергонезалежна пам'ять — це включає флеш-пам'ять, стираючу пам'ять тільки для читання (EPROM) і електрично програмована пам'ять, що програмується. пам'ять тільки для читання (EEPROM).

Як правило, мікроконтролери розроблені так, щоб їх було легко використовувати без додаткових обчислювальних компонентів, оскільки вони розроблені з достатньою вбудованою пам'яттю і пропонують контакти для

звичайних операцій введення-виводу, щоб вони могли безпосередньо взаємодіяти з датчиками та іншими компонентами.

Архітектура мікроконтролерів може бути заснована на архітектурі Гарварда або фон Неймана, обидва з яких пропонують різні методи для обміну даними між процесором і пам'яттю. В архітектурі Гарварда шина даних і інструкції є окремими, що дозволяє одночасно передавати. В архітектурі фон Неймана шина використовується як для даних, так і для інструкцій.

Процесори мікроконтролера можуть базуватися на розрахунках комплексного набору команд (CISC) або скорочених обчисленнях набору інструкцій (RISC). Загалом CISC має близько 80 інструкцій, тоді як RISC має близько 30, і більше режимів адреси 12-24 порівняно з 3-5 RISC. Хоча CISC може бути легшим у реалізації та ефективнішим у використанні пам'яті, він може погіршити продуктивність через більшу кількість тактів, необхідних для виконання інструкцій. RISC, який більше зосереджується на програмному забезпеченні, часто забезпечує вищу продуктивність, ніж процесори CISC, які більше зосереджуються на апаратному забезпеченні, завдяки спрощеному набору інструкцій і, отже, збільшеній простоті дизайну, але через акцент на програмному забезпеченні, може бути складнішим. Який ISC використовується, залежить від програми.

Коли мікроконтролери вперше з'явилися, вони використовували тільки асемблер. Сьогодні мова програмування C є популярним варіантом. Інші поширені мови мікропроцесора включають Python і JavaScript. MCU мають вхідні та вихідні контакти для виконання периферійних функцій. Такі функції включають аналого-цифрові перетворювачі, РК-дисплеї (LCD-контролери), годинники реального часу (RTC), універсальний синхронний/асинхронний передавач (USART), таймери, передавач універсального асинхронного приймача (UART) і USB-з'єднання з універсальною послідовною шиною. Датчики, які збирають дані про вологість і температуру, серед іншого, також часто підключаються до мікроконтролерів.



Мікроконтролери використовуються в різноманітних галузях промисловості та додатках, включаючи будинок і бізнес, автоматизацію будівель, виробництво, робототехніку, автомобільну промисловість, освітлення, розумну енергетику, промислову автоматизацію, комунікації та Інтернет (IoT). Одним з дуже специфічних застосувань мікроконтролера є його використання як процесор цифрових сигналів. Часто аналогові вхідні сигнали надходять з певним рівнем шуму. Шум у цьому контексті означає неоднозначні значення, які неможливо легко перевести на стандартні числові значення. Мікроконтролер може використовувати АЦП і ЦАП для перетворення вхідного аналогового шумового сигналу в рівний цифровий вихідний сигнал.

Найпростіші мікроконтролери полегшують роботу з електромеханічними системами, зручними для повсякденного використання, такими як духовки, холодильники, тостери, мобільні пристрої, брелоки, системи відеоігор, телевізори та системи поливу газону. Вони також поширені в офісних пристроях, таких як копіювальні апарати, сканери, факси і принтери, а також розумні лічильники, банкомати та системи безпеки. Більш сучасні мікроконтролери виконують важливі функції в літаках, космічних кораблях, океанських кораблях, транспортних засобах, медичних системах і системах життєзабезпечення, а також роботах. У медичних сценаріях мікроконтролери можуть регулювати роботи на штучному серці, нирках або інших органах. Вони також можуть сприяти роботі протезів.

Нижче наведено список основних конкурентів на ринку плат для налагодження мікропроцесорів.

Історично недорога і легка плата для програм, що вимагають меншого шуму та вводу-виводу, ніж основна плата Raspberry Pi, Raspberry Pi Zero нещодавно отримала величезний успіх з випуском Raspberry Pi Zero 2 W (Рисунок 2.1).

Ця нова плата зберігає мініатюрну форму і більшість портів і варіантів підключення, включаючи роз'єм micro-USB 2.0 (OTG), міні-HDMI і CSI, але робить великий крок вперед з чотирьохядерним процесором Broadcom

BCM2837B0. Процесор ARM. Це ставить її на рівень з основними платами Raspberry Pi 3A+ і B+, загалом, продуктивність у кілька разів перевищує продуктивність оригінальної плати Raspberry Pi Zero.

Однак такий прогрес коштує грошей. У порівнянні з цінником попереднього Zero в 10 доларів, нова плата Zero 2 W коштує 15 доларів, якщо ви можете знайти таку.



Рисунок 2.1 — Raspberry pi Zero 2 W

Характеристики Raspberry Pi Zero 2 W:

- SBC Raspberry Pi Zero 2 W;
- processor BCM2710A1 (BCM2837), (4× 1GHz Cortex-A53);
- architecture ARM, 32-bit;
- GPU Broadcom VideoCore IV;
- memory 512 MB LPDDR2;
- inbuilt Storage None;
- expandable storage MicroSD;
- networking & connectivity 2.4GHz 802.11b/g/n Wi-Fi, Bluetooth 4.2;
- I/O Mini-HDMI, Micro-USB 2.0 OTG, CSI, 40-pin GPIO header;
- power 5V DC 2.5A;
- market price \$15.

Pine A64-LTS — це те, що ви можете вважати «традиційною» альтернативою Raspberry Pi 4 (Рисунок 2.2). Він має слот microSD, HDMI, Gigabit Ethernet, аудіороз'єм, подвійний порт USB 2.0 і порти micro-USB. Цей одно-

платний комп'ютер має розміри 133 x 80 мм і має сумісний з Pi 40-контактний роз'єм на додаток до того, що Pine називає 14-контактною «шиною Ейлера», яка пропонує додаткові можливості керування апаратним забезпеченням, крім GPIO у стилі Pi.

Технічні характеристики Topline включають чотириядерний процесор ARM Cortex A53 1,152 ГГц 64-розрядний з графічним процесором Mali-400 MP2 і 2 ГБ системної пам'яті LPDDR3. Пам'ять MicroSD використовується спільно з ОС, хоча є також додатковий модуль eMMC, який приймає до 128 ГБ додаткового сховища. Будучи довгостроковою версією, користувачі вбудованих систем повинні звернути увагу на те, що Pine зобов'язалася постачати плату принаймні до 2022 року.



Рисунок 2.2 — Pine A64-LTS

Характеристики Pine A64-LTS:

- SBC Pine A64-LTS;
- processor Allwinner A64 (4x 1.1GHz Cortex-A53);
- architecture ARM, 64-bit;
- GPU Mali-400 MP2;
- memory 2 GB LPDDR3;
- inbuilt Storage 128 MB SPI flash;
- expandable Storage eMMC (up to 128 GB), microSD;
- networking & connectivity 2.4 GHz Wi-Fi, Bluetooth 4.0, Gigabit Ethernet;

— I/O 2x USB 2.0, 3.5 mm audio/mic, DSI, CSI, TP (SPI with interrupt), RTC, VBAT, Wi-fi/BT module header, 40-pin GPIO, 34-pin Euler header, 10-pin EXP header;

— power5V @2A (via 3.5 mm jack);

— market price \$32.

PocketBeagle — це одноплатний комп'ютер з відкритим вихідним кодом від Beagle-Bone, який увійшов у список за його давню відповідність до платт для безголовних додатків – відсутність графічного процесора означає, що йому буде важко створити простий досвід використання (Рисунок 2.3).

Побудований на основі Octavo Systems OSD3358 1 ГГц ARM Cortex-A8, PocketBeagle має 512 МБ інтегрованої пам'яті DDR3. Він невеликий, лише 56 x 35 мм, що робить його приблизно такого ж розміру, як Raspberry Pi Zero.

Те, що є можливість підключити цей SBC до ноутбука, як USB-брелок, робить його надзвичайно зручним – ви можете запрограмувати плату за допомогою веб-браузера, який надає доступ до командного рядка Linux і текстового редактора. PocketBeagle також має два програмованих блоки реального часу (PRU), що надає йому додаткове використання в ситуаціях, пов'язаних із високошвидкісними подіями на основі переривань, з чим системи Linux традиційно не обладнані для роботи.



Рисунок 2.3 — PocketBeagle

### Характеристики PocketBeagle:

- SBC Pocket Beagle;
- processor Octavo Systems OSD3358 (1× 1GHz Cortex-A8, 1× Cortex-M3, 2× 200MHz PRU coprocessors);
- architecture ARM, 32-bit;
- GPU Power VR SGX530;
- memory 512 MB DDR3;
- inbuilt Storage None;
- expandable Storage microSD (bootable);
- networking & Connectivity n/a;
- I/O Micro-USB, 36-pin GPIO;
- power Unknown (via micro-USB or header);
- market price \$35.

Для багатьох, Raspberry Pi (Рисунок 2.4) — це назва, яка є синонімом швидких обчислень на одній платі. Встановлюючи планку універсальності, ціни та продуктивності, четверте покоління основних плат Raspberry Pi пропонує утиліту з повнорозмірними портами USB 3.0, гігабітним Ethernet і графічним ядром, здатним виводити два дисплеї з роздільною здатністю 4К.

Це більш ніж потужний медіа-центр, навіть із 2 ГБ оперативної пам'яті. Аналогічно, можна отримати комфортні настільні комп'ютери, хоча, природно, з опцією 4 ГБ або, доданої в 2020 році, 8 ГБ буде можливо більше.

Однією беззаперечною привабливістю Raspberry Pi 4 Model B (або будь-якої Raspberry Pi, якщо на те пішло), є велика екосистема, в якій вони живуть. Є величезні можливості через офіційні та неофіційні апаратні доповнення, розширення та модифікації, не кажучи вже про добре задокументовані проекти, які використовують їх. Якщо ви новачок у цьому практичному світі обчислень, Raspberry Pi 4 Model B — це вірний вибір.



Рисунок 2.4 — Raspberry Pi 4 Model B

Характеристики Raspberry Pi 4 Model B:

- SBC Raspberry Pi 4 Model B;
- processor Broadcom BCM2711B0, (4× 1.5GHz Cortex-A72);
- architecture ARM, 32-bit;
- GPU Broadcom VideoCore VI;
- memory 2, 4, 8 GB DDR4;
- inbuilt StorageNone;
- expandable StorageMicroSD;
- networking & connectivity Dual-band 802.11ac/b/g/n Wi-Fi, Bluetooth 5.0, Bluetooth Low Energy, Gigabit Ethernet;
- I/O 2× micro-HDMI 2.0, 2× USB 3.0, 2× USB 2.0, 3.5mm audio/video, CSI, DSI, 40-pin GPIO;
- power5V DC 2.5A;
- market price \$45.

Значно покращуючи продуктивність тепер уже застарілого Odroid-C2, Odroid-C4 є SBC (Рисунок 2.5) з подібним розміром і компоновкою до Raspberry Pi 4. Його продуктивність також порівнянна, хоча й з деякими значними апаратними відмінностями.

SoC Odroid — це Amlogic ARM Cortex-S905X3, що містить чотири-ядерний кластер Cortex-A55 і графічний процесор Mali-G31. Ядра A55

працюють на частоті 2,0 ГГц без теплового регулювання за допомогою стандартного радіатора, методом пасивного охолодження, якого не вистачає RPi 4.

В інших місцях Odroid-C4 стандартно поставляється з 4 ГБ оперативної пам'яті DDR4, як і RPi 4 середньої специфікації. Крім того, як і RPi 4, Odroid C4 може виводити відео 4К з частотою 60 Гц, хоча тут через повнорозмірний HDMI 2.0 порт. Він не пропонує подвійний вихід дисплея, як Pi, але перевершує Pi з чотирма портами USB 3.0.

На жаль, немає Wi-Fi або Bluetooth, але Gigabit Ethernet пропонує певну вбудовану мережу, якщо ви не хочете зарезервувати пару цих портів USB для бездротових ключів.

Тут представлений сумісний з Pi 40-контактний роз'єм, що завжди бажано, хоча зверніть увагу на відсутність з'єднань DSI та CSI. Це означає, що немає камер Pi або подібних візуальних периферійних пристроїв. Нарешті, Odroid C4 має слот eMMC, що дає вам можливість швидкої флеш-пам'яті.



Рисунок 2.5 — Odroid-C4

#### Характеристики Odroid-C4

- SBC Odroid-C4;
- processor Amlogic S905X3 (4x 2.0Ghz Cortex-A55);
- architecture ARM, 64-bit;
- GPU Mali-G31 MP2;
- memory 4 GB DDR4;

- inbuilt StorageNone;
- expandable StorageeMMC, microSD;
- networking & connectivity Gigabit Ethernet;
- I/O HDMI 2.0, 4x USB 3.0, micro-USB, IR receiver, UART, 47-pin GPIO;
- power 12V @2A (via 5.5/2.1 mm jack);
- market price \$54.

Rock Pi 4 Model C (Рисунок 2.6) досить вигідно поєднується з Raspberry Pi 4. Він доступний лише з 4 ГБ оперативної пам'яті, так само, як і RPi 4s для середнього класу Raspberry Pi. З точки зору потужності обробки, ви отримуете трохи більше від Rock Pi через графічний процесор Rockchip RK3399 і Mali T860.

Уважно дотримуючись стилю Raspberry Pi і використовуючи назви на носі, можна очікувати, що ввід-вивод і загальний досвід будуть здебільшого однаковими.

Для початку Rock Pi 4 C наближається до можливостей відтворення мультимедіа RPi 4 з подвійним виходом відео на дисплей через з'єднання micro-HDMI і mini Display-Port. Ви не зрівняєтеся з можливостями Rpi 4 dual-4K60, але ми підозрюємо, що мало хто буде мати проблеми з відеопотоками 4K60/2K60 Rock Pi 4 C.

Однією відмітною особливістю Rock Pi 4 C над Raspberry Pi 4 є слоти eMMC і M.2, які дозволяють додавати стабільні високошвидкісні диски для ОС і сховища.

Новачки в одноплатних обчисленнях зауважують, що готовий досвід не збігається з RPi та його майстром запуску NOOBS.

З іншого боку, ви також отримуете нейронний процесор (NPU) як частину системи Rock Pi 4 на чіпі, що корисно для розвантаження частини обробки в додатках AI.





Рисунок 2.6 — Rock Pi 4 Model C

#### Характеристики Rock Pi 4 Model C

- SBC Rock Pi 4 Model C;
- processor Rockchip RK3399 (2x 1.4Ghz Cortex-A72, 4x 1.4GHz Cortex A53);
- architecture ARM, 64-bit;
- GPU Mali T860MP4;
- memory 4 GB LPDDR4-3200;
- inbuilt StorageNone;
- expandable StorageeMMC (up to 128 GB), microSD (up to 128 GB), M.2 SSD (up to 2 TB NVME SSD) (shared with OS);
- networking & connectivity 2.4/5 GHz Wi-Fi, Bluetooth 5.0, Gigabit Ethernet;
- I/O Micro HDMI 2.0, USB 3.0 OTG, USB 3.0 host, 2x USB 2.0 host, 3.5mm audio/mic, DSI, CSI, RTC, 40-pin GPIO;
- power 9-20 V @2A (via USB C PD, Qualcomm Quick Charge compatible);
- market price \$59.

Pine64 RockPRO 64, як і Rock Pi 4 C вище в цьому списку, працює на процесорі Rockchip RK3399 з графікою Mali T860. Він має 4 ГБ оперативної пам'яті LPDDR4 (доступно також 2 ГБ) з подвійним дисплеєм і відрізняється від RPi 4 додатковою універсальністю за допомогою слотів eMMC і M.2.

Незважаючи на те, що раніше Raspberry Pi 4 був високопотужною альтернативою RPi 3, RockPRO64 (Рисунок 2.7) не менш актуальний сьогодні, причому головною відмінністю між ним і RPi (і Rock Pi 4 C, якщо на те пішло) є слот PCIe x4. Це означає, що можна ще більше розширити RockPro64, наприклад, за допомогою надшвидкісного сховища через NVMe SSD.

З фотографій ізольованих плат ніколи не зрозуміло, але RockPro64 досить великий – легко подвоюйте розмір Raspberry Pi 4, 133 x 80 мм – тому враховуйте це при прийнятті рішення про покупку, якщо ваш проект має бути невеликим.



Рисунок 2.7 — Pine64 ROCKPro 64

#### Характеристики Pine64 ROCKPro 64

- SBC Pine64 ROCKPro 64;
- processor Rockchip RK3399 (2x 1.4Ghz Cortex-A72, 4x 1.4GHz Cortex A53);
- architecture ARM, 64-bit;
- GPU Mali T860MP4;
- memory 2/4 GB LPDDR4;
- inbuilt storage 128 MB SPI flash memory boot manager;
- Expandable Storage MMC (up to 128 GB), microSD;
- networking & connectivity Gigabit Ethernet;
- I/O USB 3.0 C (video out), USB 3.0 host, 2x USB 2.0 host, 3.5 mm audio/mic, IR receiver, DSI, CSI, SCI, TP, eDP, RTC, PCIe x4, 40-pin GPIO;

- power 12V @3/5A (via 5.5/2.1 mm jack);
- market price \$59/79.

За ціною понад 80 доларів за модель 4 ГБ, Odroid-N2+ (Рисунок 2.8) далеко не найдешевший одноплатний комп'ютер у цьому списку, перевершуючи 4 ГБ Raspberry Pi 4 за 55 доларів. Однак для процесорної потужності та введення-виводу ви бачите певну віддачу від цих грошей.

Оновлення Odroid N2, N2+ може похвалитися підвищенням тактової частоти на 33%, а його швидкість робить його популярним вибором для тих, хто хоче запускати емулятори відеоігор або програми на базі ОС Android. Повний вихід HDMI (корисний для емуляційних станцій plug-and-play), а також чотири роз'єми USB 3.0, eMMC і microSD для периферійних пристроїв і швидкого зберігання. Вони, а також типовий 40-контактний роз'єм для розширення функціональних можливостей або прямого контролю над іншим обладнанням, збалансують Odroid N2+ як хороший універсальний пристрій (який добре обробляє графічні носії).

Він трохи більший за Raspberry Pi 4 з розмірами 90 x 90 мм, і велика частина цієї ваги йде на підставку Odroid N2+, яка корисно служить радіатором, щоб запобігти гальмування при інтенсивних процесорних завданнях.



Рисунок 2.8 — Odroid-N2+

Характеристики Odroid-N2+

- SBC Odroid-N2+;
- processor Amlogic S922X (4x 2.4GHz Cortex-A73, 2x 2.0GHz Cortex A53);
- architecture ARM, 64-bit;

- GPU Mali-G52 GPU;
- memory 2/4 GB DDR4;
- inbuilt storage 8 MiB SPI flash memory boot manager;
- expandable StorageeMMC (up to 128 GB), microSD;
- networking & connectivity Gigabit Ethernet;
- I/O HDMI 2.0, 4x USB 3.0 host, micro-USB, 3.5 mm audio/mic, IR receiver, UART, RTC, 40-pin GPIO;
- power 12V @2A (via 5.5/2.1 mm jack);
- market price \$66/83.

Перша плата Radxa x86, Rock Pi X, має певну репутацію, оскільки вона складна та менш зручна, ніж її попередники на базі ARM. Незважаючи на це, він має безперечно привабливий набір функцій, що робить його вартим уваги (Рисунок 2.9).

По-перше, у вас є повне середовище робочого столу Windows 10, з яким можна пограти, на додаток до вводу-виводу, який зазвичай зустрічається в Raspberry Pi, включаючи сумісний з Pi 40-контактний роз'єм, повний вихід HDMI і 3,5-мм роз'єм для навушників з мікрофоном.

Доступний з до 4 ГБ оперативної пам'яті — порівнянно з Raspberry Pi 4 середньої специфікації — Rock Pi X може мати 128 ГБ вбудованої пам'яті eMMC, яку можна розширити за допомогою microSD. Плата Model B включає Wi-Fi і Bluetooth для бездротового підключення.



Рисунок 2.9 — Rock Pi X Model B

### Характеристики Rock Pi X Model B

- SBC Rock Pi X Model B;
- processor Intel Atom x5-Z8350 Cherry Trail (4x 1.92 GHz);
- architecture x86, 64-bit;
- GPU Intel Gen8 HD Graphics;
- memory 2/4 GB LPDDR3;
- inbuilt Storage 16/32/64/128 GB eMMC;
- expandable Storage microSD (up to 128 GB);
- networking & connectivity 5 GHz Wi-Fi, Bluetooth 4.2, Gigabit Ethernet;
- I/O HDMI 2.0, USB 3.0 OTG, 3x USB 2.0 host, 3.5 mm audio/mic, RTC, 40-pin GPIO;
- power 9-20V @2A (via USB C PD 2.0, Qualcomm Quick Charge 3.0/2.0);
- market price ~\$90-110.

Сучасний одноплатний комп'ютер Asus (Рисунок 2.10) має потужний удар у своїй мініатюрній формі. Подібно до більшості плат у цьому списку, Tinker Board 2S є універсальним інструментом для створення, ідеальним для IoT, периферійних обчислень, настільних обчислень, проектів виробників або просто як медіа-центр.

Подвійний дисплей 4K (через три можливі виходи дисплея – DSI, USB-C і HDMI CEC) підходять для графічних і медіа-додатків, але велика кількість вводу/виводу та сумісність з аксесуарами та капелюхами Rpi надають апаратну універсальність.

Як і у минулих Tinker Boards, нам подобається, що користувальницький досвід добре задокументований, із простим налаштуванням і, на самій платі, безліччю зручних для користувача знаків у кольорових заголовках і етикетках із шовкографією.

Деякі невеликі переваги перед Rpi полягають у таких функціях, як заголовки RTC для ведення годинника. Asus також підкреслює захист Tinker Board 2S від перенапруги та електростатичного розряду.



Рисунок 2.10 — Asus Tinker Board 2S

#### Характеристики Asus Tinker Board 2S

- SBC Asus Tinker Board 2S;;
- processor Rockchip RK3399 (2x 1.4Ghz Cortex-A72, 4x 1.4GHz Cortex A53);
- architecture ARM, 64-bit;
- GPU Mali T860 MP4;
- memory 2 GB LPDDR4;
- inbuilt storage 16 GB eMMC;
- expandable Storage MicroSD;
- networking & Connectivity 2.4/5 GHz Wi-Fi, Bluetooth 5.0, BLE, Gigabit Ethernet;
- I/O HDMI 2.0, 3x USB 3.2, USB 3.2 C, DSI, CSI, 2-pin power on, 2-pin recovery, 2-pin DC fan, UART, RTC, 40-pin GPIO;
- power 12~19 V (via 5/2.5 mm jack);
- market price \$119.

LattePanda використовує компактність і зручність форм-фактора Raspberry Pi і його безлічі дистрибутивів і розглядає речі в зовсім іншому напрямку (Рисунок 2.11). Ви все ще маєте очікуваний вхід-вивід, повний HDMI,

USB 3.0, USB 2.0, microSD, контакти GPIO, Ethernet та бездротове з'єднання, але це в Windows. Ви отримуєте переваги підключення і відтворення міні-ПК з Windows, а також фізичний контроль, який пропонує SBC.

Звіти про використання говорять про те, що LattePanda має тенденцію працювати досить сильно, тому вам потрібно буде дослідити рішення для охолодження, щоб уникнути гальмування ЦП. Крім того, за 159 доларів, це досить дорого для версії з 4 ГБ оперативної пам'яті, якщо порівняти її з 4 ГБ Rpi4 за 55 доларів.



Рисунок 2.11 — LattePanda V1

#### Характеристики LattePanda V1

- SBC LattePanda V1;
- processor Intel Cherry Trail Z8350 + ATmega32u4 coprocessor (4x 1.8GHz);
- architecture x86, 64-bit;
- GPU Intel HD Graphics @ 200-500 MHz;
- memory 2/4 GB DDR3L;
- inbuilt Storage 32/64 GB;
- networking & connectivity 2.4 GHz Wi-Fi, Bluetooth 4.0, Ethernet;
- I/O HDMI, USB 3.0 C (video out), 2x USB 2.0, 3.5 mm audio/mic, DSI, 6-pin Intel GPIO, 20-pin Arduino GPIO, 6x Gravity sensor connectors;

- power 5V @2A (via microUSB, GPIO);
- market price \$129/159.

Розрекламований як потужне рішення для Edge Computing, Seedeed's Odyssey (Рисунок 2.12) здається, є улюбленим одноплатним комп'ютером, який пропонує чудове поєднання потужності та універсальності. Як і в інших системах x86, наведених вище, він також пропонує повний робочий стіл Windows разом із можливістю програмування для Arduino за допомогою мікроконтролера ARM Cortex M0+.

8 ГБ пам'яті LPDDR4 дає достатній запас для багатозадачності або серверних додатків, а 64 ГБ eMMC дає місце для зберігання поряд з ОС. Дешеві версії цього одноплатного комп'ютера доступні без eMMC, тому ви можете пропустити майже готовий до роботи досвід, який пропонує ця версія, і замість цього завантажитися з інших носіїв даних.

Говорячи про зберігання, Odyssey має багато портів розширення, включаючи два M.2 (B і M), SATA III, microSD, USB 2.0 і USB 3.1 (A і C). 40-контактний роз'єм у стилі Raspberry Pi і 28-контактний роз'єм Arduino означають, що ви можете підключити його до величезної кількості шилдів для розширеної функціональності. Коротше кажучи, потужний і гнучкий міні-комп'ютер, який може виконувати більшість проектів і може бути оснащений тоннами швидкої пам'яті.



Рисунок 2.12 — Seedeed Odyssey (X86J4125800)



### Характеристики Seeed Odyssey (X86J4125800)

- SBC Seeed Odyssey (X86J4125800);
- processor Intel Celeron J4125 + Microchip ATSAMD21G18 (4x 2.0GHz);
- architecture x86, 64-bit;
- GPU Intel UHD Graphics 600;
- memory 8 GB LPDDR4;
- inbuilt Storage None;
- expandable storage M.2 B, SATA III, USB 2.0, UIM, M.2 M, PCIe 2.0 x4, microSD, SIM card, SATA III;
- networking & connectivity 2.4/5 Ghz Wi-Fi, 2x Gigabit Ethernet, SIM
- I/O HDMI 2.0a, 2x USB 2.0, USB 3.1, USB 3.1 C, 3.5 mm audio/mic, TMP, UART, 4-pin fan, 4-pin power and switch, 40-pin GPIO, 28-pin Arduino;
- power 12-19V (via 5.5/2.1 mm jack, USB C PD);
- market price \$218.

Ще один одноплатний комп'ютер, який надає перевагу архітектурі x86 та перевагам Windows 10, які вона допускає, Udoо x86 II Ultra (Рисунок 2.13) поєднує в собі універсальність плата для «виробників» із потужним досвідом роботи з настільними комп'ютерами в тому, що її виробники називають «найпотужнішою платою виробника x86».

Ми не дуже впевнені в цьому, але ми впевнені, що він містить багато вводу-виводу та потужності, щоб принаймні змусити деяких людей не звернути увагу на дивовижну ціну в 294 долари.

Ви отримуєте потужний, навіть потужний SBC для настільних мультимедійних додатків, контакти та з'єднання у стилі Raspberry Pi, а також платформу, сумісну з Arduino Leonardo, завдяки вбудованому мікроконтролеру ATmega23U4. Ви можете підключити три дисплеї до Udoо x86 II Ultra завдяки повноцінному порту HDMI 2.0 і двом міні-портам DisplayPort.

Він поставляється з 8 ГБ оперативної пам'яті DDR3L в стандартній комплектації, але не має Wi-Fi – якщо ви хочете використовувати бездротовий зв'язок, вам знадобиться один із двох слотів M.2 (інший призначений для швидкого флеш-пам'яті).



Рисунок 2.13 — Udoo x86 II Ultra

#### Характеристики Udoo x86 II Ultra

- SBC Udoo x86 II Ultra;
- processor Intel Pentium N3710 (4x 2.56GHz);
- architecture x86, 64-bit;
- GPU Intel HD Graphics 405;
- memory 8 GB DDR3L;
- inbuilt Storage 32 GB eMMC;
- expandable storage SATA, M.2 B SSD, microSD;
- networking & connectivity Gigabit Ethernet, M.2 E slot;
- I/O HDMI, 3x USB 3.0, 2x miniDP++, S/PDIF out, 3.5 mm audio/mic, IR receiver, 2x UART, RTC, 25-pin Arduino, 20-pin Braswell GPIO;
- power 12V @3A (via 5.5/2.1 mm jack);
- market price \$311.

Там, де Raspberry Pi, на додаток до інтерфейсу та керування апаратним забезпеченням тугі-ад, також може використовуватися як повноцінний настільний комп'ютер, мікроконтролери на базі Arduino (Рисунок 2.14)

знаходять застосування в програмах фіксованого призначення, керуючи апаратним забезпеченням та виконуючи базове програмування.

Якщо ви хочете керувати роботом або навіть 3D-принтером, то Mega 2560 Rev 3, наразі за ціною 40 доларів, є надійним вибором.

Його ядром є чіп Atmega2560, що працює на частоті 16 МГц. Третя версія поставляється з 54 цифровими контактами входу/виходу (з яких 15 можна використовувати як виходи ШІМ), 16 аналоговими входами, 4 апаратними послідовними портами UART, USB-з'єднанням, роз'ємом ICSP і кнопкою скидання.



Рисунок 2.14 — Arduino Mega 2560

#### Характеристики Arduino Mega 2560

- SBC Arduino Mega 2560;
- processor ATmega2560 (16MHz);
- architecture RISC, 8-bit;
- GPU n/a;
- memory 8 KB SRAM, 4 KB EEPROM;
- inbuilt storage 256 KB flash ROM;
- expandable storage None;
- networking & connectivity None;
- I/O USB Type B, 54-pin digital, 16-pin analog, 4x UART;

- power 7-12 V (via USB, power jack);
- market price \$40.

Зі всіх вище зазначених варіантів Raspberry Pi (рисунок 2.15) є найкращим варіантом за вказаними нижче критеріями.



Рисунок 2.15 — Зовнішнє зображення Raspberry Pi

Цей мікрокомп'ютер корисний для малого бізнесу, який має менший бюджет, щоб використовувати свій продукт або винайти нову технологію, яка вбудовує продукт. Власники малого бізнесу можуть використовувати його для автоматизації будь-яких невеликих завдань, тобто; наприклад, використання Pi для запуску веб-сайту або використання його як невеликої бази даних та медіа-сервера.

Продукт не вимагає від користувача великого досвіду програмування, оскільки він призначений для того, щоб молоде покоління дізналося про програмування. Python, мова програмування, тобто; Pi використовує меншу кількість складних, ніж інші доступні мови. Він має кращу читабельність коду і дозволяє користувачеві вводити поняття, використовуючи меншу кількість рядків. Python також має функцію автоматичного керування пам'яттю.

Продукт дає багато простору для експериментів і перетворення його на щось інше, тобто; зовсім інший. SD-карти на платі можна легко переми-кати, тобто; що дозволяє змінювати функції пристрою, не витрачаючи багато часу на перевстановлення програмного забезпечення.

Raspberry Pi ідеально підходить для адаптивних технологій і може відображати зображення або відтворювати відео, тобто; з роздільною здатністю високої чіткості для створення систем, таких як створення прототипів вбудованих систем. Цей продукт дає можливість будувати складні та ефективні за дешевшою ціною.

Raspberry Pi поставляється з 26 контактами GPIO, які дійсно дуже корисні для вбудованих проектів та обладнання для інтерфейсу. Ці контакти дійсно корисні для вивчення інтерфейсу компонентів. Ви можете комбінувати декілька цифрових датчиків разом завдяки великій кількості контактів GPIO. Він підтримує майже всі периферійні пристрої, які підтримує Arduino.

Для нього на ринку доступно багато аксесуарів. Ви знайдете цілу велику спільноту та підтримку.

Підтримка різних мов програмування — це одна з найкращих переваг цієї плати, якщо ми порівняємо її з arduino, ви дізнаєтеся, що arduino підтримує лише C, C++. Хоча ця плата працює як одноплатний комп'ютер. Ви отримуєте середовище робочого столу Linux, у якому ви можете кодувати практично будь-якою мовою, будь то C, C++, C#, Ruby, Java, Python тощо...

В додатку Г наведено електричну принципову схему плати Raspberry Pi.

## 2.2 Розробка структурної схеми пристрою

Головною задачею магістерської роботи є розробка системи GPS моніторингу для агропромислових підприємств.

Необхідно визначити властивості, які слід враховувати, перш ніж розробляти рішення для розумного сільського господарства

Щоб створити рішення IoT для сільського господарства, потрібно вибрати датчики для свого пристрою (або створити власний). Вибір залежа-

тиме від типів інформації, яку потрібно зібрати, та від мети системи в цілому. У будь-якому випадку якість датчиків має вирішальне значення для успіху продукту: від цього буде залежати точність зібраних даних і їх надійність.

Аналітика даних має бути в основі кожного рішення для розумного сільського господарства. Зібрані дані самі по собі мало допоможуть, якщо ви не можете зрозуміти їх. Таким чином, потрібно володіти потужними можливостями аналізу даних і застосовувати алгоритми прогнозування та машинне навчання, щоб отримати корисну інформацію на основі зібраних даних.

Технічне обслуговування обладнання є проблемою, яка має основне значення для продуктів IoT у сільському господарстві, оскільки датчики зазвичай використовуються в польових умовах і їх можна легко пошкодити. Таким чином, ви повинні переконатися, що обладнання є міцним і простим у догляді. Інакше доведеться міняти датчики частіше, ніж хотілося б.

Розумні програми для сільського господарства повинні бути адаптовані для використання в полі. Власник бізнесу або керівник ферми повинні мати можливість отримати доступ до інформації на місці або віддалено через смартфон або настільний комп'ютер.

Крім того, кожен підключений пристрій має бути автономним і мати достатній діапазон бездротового зв'язку для зв'язку з іншими пристроями та надсилання даних на центральний сервер.

Щоб переконатися, що програма для розумного фермерства добре працює (і щоб переконатися, що вона справляється з навантаженням даних), потрібна надійна внутрішня інфраструктура.

Крім того, внутрішні системи мають бути захищеними. Неможливість належного захисту системи лише збільшує ймовірність того, що хтось проникне в неї, вкраде дані або навіть отримає контроль над автономними тракторами.

Потреба в передачі даних між багатьма сільськогосподарськими об'єктами все ще є проблемою для впровадження розумного сільського господарства. Зайве говорити, що з'єднання між цими об'єктами має бути достатньо надійним, щоб витримувати погані погодні умови та забезпечувати безперебійну роботу. Сьогодні пристрої IoT все ще використовують різні протоколи підключення, хоча зараз тривають зусилля щодо розробки уніфікованих стандартів у цій сфері. Сподіваємося, поява 5G і таких технологій, як космічний Інтернет, допоможе знайти рішення цієї проблеми.

Через велику різноманітність типів даних у сільськогосподарській галузі забезпечення оптимальної частоти збору даних може бути проблематичним. Дані з польових, повітряних і екологічних датчиків, програм, машин та обладнання, а також оброблені аналітичні дані можуть бути предметом обмежень і правил. Сьогодні безпечна та своєчасна доставка та обмін цими даними є однією з актуальних проблем розумного сільського господарства.

Точне землеробство та технології IoT передбачають роботу з великими наборами даних, що збільшує кількість потенційних лазівок у безпеці, які зловмисники можуть використовувати для крадіжки даних та хакерських атак. На жаль, безпека даних у сільському господарстві досі залишається значною мірою незнайомою концепцією. Багато ферм, наприклад, використовують дрони, які передають дані на сільськогосподарську техніку. Цей апарат підключається до Інтернету, але має майже нульовий захист безпеки, наприклад паролі користувачів або аутентифікацію віддаленого доступу.

Деякі з основних рекомендацій щодо безпеки IoT включають моніторинг трафіку даних, використання методів шифрування для захисту конфіденційних даних, використання інструментів безпеки на основі штучного інтелекту для виявлення слідів підозрілої активності в режимі реального часу та зберігання даних у блокчейні для забезпечення його цілісності. Щоб отримати повну користь від Інтернету речей, фермерам доведеться ознайомитися з концепцією безпеки даних, налаштувати внутрішні політики безпеки та дотримуватися їх.

Отже виходячи з вище названих критеріїв необхідно скласти структурну схему системи.

Серцем та мозком системи буде мікропроцесор, який буде керувати всією системою та обробляти вхідні дані для подальшого використання.

В системі GPS моніторингу повинен бути GPS модуль для отримання точних координат транспорту для подальшої обробки мікроконтроллером.

Всі отримані дані повинні бути відправлені на клієнтський пристрій для подальшого використання, саме тому в системі повинен бути GSM модуль для підтримки зв'язку з мережею Інтернет.

Також у непередбачуваних випадках коли система не має зв'язку необхідно мати вбудовану пам'ять для запису всіх даних та подій та подальшої передачі всіх цих даних коли зв'язок буде відновлено.

Як і будь-який електронний пристрій система GPS моніторингу потребує живлення.

Система може також використовуватись і в інших цілях окрім GPS моніторингу тому є можливість додати датчики рівня палива, рівня добрив та зерна, рівня заповненості бункеру, датчик заглиблення спецпристроїв в ґрунт. Також є можливість встановлення камер спостереження, мікрофону та динаміків для віддаленої комунікації з робітниками. Деякі з цих датчиків може бути неможливо під'єднати до системи фізично тому потрібно врахувати встановлення модулів бездротового зв'язку через WiFi або Bluetooth.

Не правильне виконання роботи може призвести до значних негативних наслідків тому системі необхідно мати реле для керування транспортом або ж для керування спецзасобами для запобігання таких випадків. Також в агропромисловій сфері не є рідкістю різні небезпечні ситуації які потребують швидкого прийняття рішень від керівництва тому є потреба для встановлення кнопки екстреної ситуації, натиск якої миттю повідомить відповідних людей про несправність або ж аварію.

Тепер після визначення всіх компонентів системи можна скласти структурну схему(рисунок 2.16).



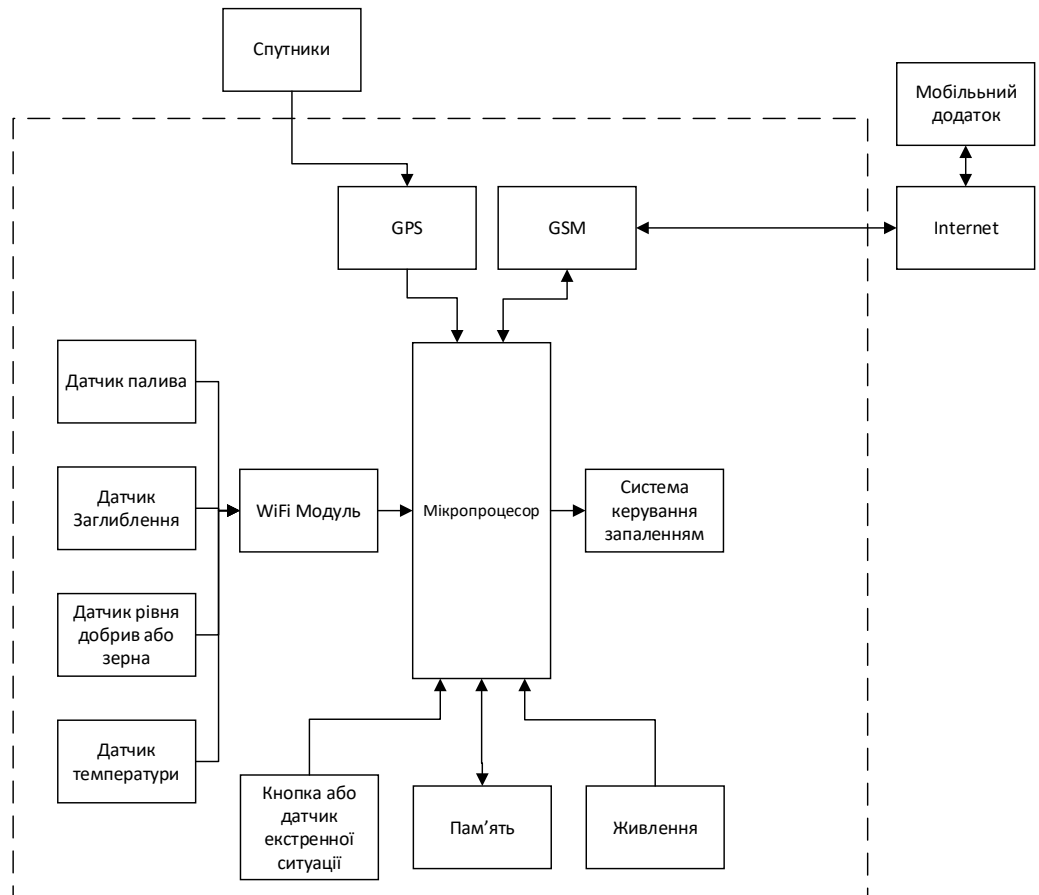


Рисунок 2.16 — Структурна схема пристрою

## 2.3 Вибір компонентів пристрою

### 2.3.1 Вибір GPS-модуля

GPS-модуль NEO-6M (рисунок 2.17) — це повноцінний GPS-приймач із вбудованою керамічною антеною розміром 25 x 25 x 4 мм, що забезпечує потужний пошук супутникового зв'язку. В основі модуля лежить GPS-чіп NEO-6M від u-blox. Розмір чіпа менший за розмір поштової марки, але він містить неймовірну кількість функцій у своїй маленькій рамці.

Він може відстежувати до 22 супутників на 50 каналах і досягає найвищого рівня чутливості в галузі, тобто відстеження -161 дБ, споживаючи всього 45 мА електроенергії.

На відміну від інших модулів GPS, він може виконувати до 5 оновлень розташування в секунду. Механізм позиціонування u-blox 6 також має час до першої корекції (TTFF) менше 1 секунди.

Однією з найкращих особливостей чіпа є режим енергозбереження (PSM). Це зменшує енергоспоживання системи за рахунок вибіркового включення і вимикання частин приймача. Це різко знижує енергоспоживання модуля лише до 11 мА, що робить його придатним для енергочутливих проєктів, таких як годинник GPS!

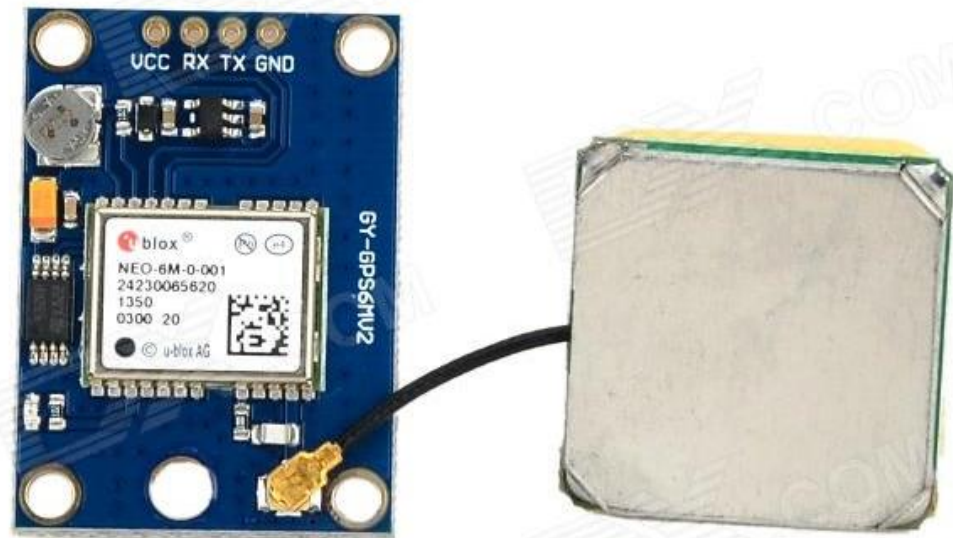


Рисунок 2.17 — Зовнішній вигляд NEO-6M

Характеристики модуля:

- вбудована керамічна активна антена розміром 25мм x 25мм x 4мм забезпечує супутниковий пошук;
- повний модуль GPS із вбудованою антеною та вбудованим EEPROM для збереження даних параметрів конфігурації;
- обладнаний індикаторами живлення та сигналу та батареєю резервного копіювання даних;
- інтерфейс: RS232 TTL;
- блок живлення: 3-5 В;
- Швидкість передачі за замовчуванням: 9600bps.

### 2.3.2 Вибір GSM/GPRS-модуля

SIM900 — широко поширений GSM/GPRS модуль з низькою вартістю від компанії SIMCOM (рисунок 2.18). Орієнтований на використання у системах автоматизації та управління.

Плата GSM модуля на стороні компонентів містить з'єднувачі для підключення антени, навушників та мікрофона. На стороні пайки плати розміщені тримач батареї CR1220 3 вольта, що підтримує роботу годинника модуля і контейнер для встановлення симкарти.

Одне із застосувань пристрою – система стеження за переміщеннями транспорту разом із пристроєм ГЛОНАСС або GPS. Відправлення SMS повідомлень дозволяє використовувати модуль у диспетчеризації, бездротової сигналізації та в охоронних системах. Модуль незамінний для контролю та управління рухомими об'єктами, що переміщуються на великі відстані. Або у разі видалення оператора на велику відстань від стаціонарного об'єкта.

GPRS/GSM SIM900 Shield з антеною може сполучатися безпосередньо з мікроконтролером через інтерфейс UART або працювати спільно з персональним комп'ютером при використанні перетворювача інтерфейсу порт ПК-UART. Це можливо завдяки програмній сумісності на рівні класу команд, що використовуються для управління модемами - AT команди.

Синтаксис системи команд серії SIM900 зберіг наступність, і переважна частина команд AT-збігається з командами SIM300. Всі модулі серії SIM900 побудовані на одній апаратно-програмній платформі і мають схожі характеристики і набір стандартних функцій, таких як голосові з'єднання, CSD-дзвінки, SMS, підключення до мережі GPRS class 10B, CMUX і т. д. Також модулі підтримують великий набір про-приетарних AT-команд для роботи з вбудованим TCP/IP-стеком та більш тонкого налаштування роботи окремих функцій модуля.

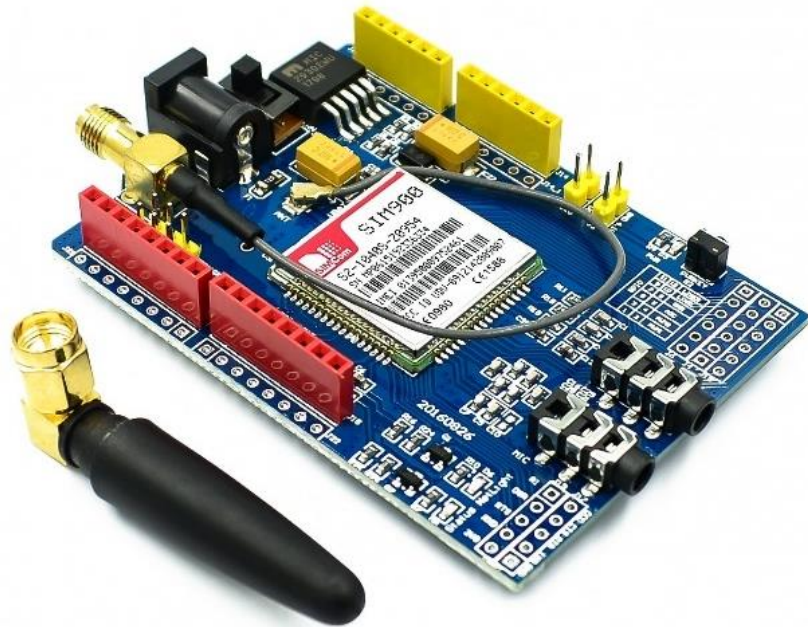


Рисунок 2.18 — Зовнішній вигляд SIM900

Загальні характеристики модулів серії SIM900:

- чотири діапазони GSM 850/900/1800/1900 МГц (900/1800 МГц для SIM900R);
- клас передачі даних GPRS multi-slot class 10/8;
- відповідність стандарту GSM фази 2/2+;
- клас потужності 4 (2 Вт у діапазонах 850/900 МГц);
- клас потужності 1 (1 Вт у діапазонах 1800/1900 МГц);
- управління AT-командами (GSM 07.07, 07.05 та фірмові AT-команди SIMCom);
- аудіокодеки HR, FR, EFR, AMR, придушення відлуння;
- CSD до 14,4 кбіт/с;
- PPP-стек;
- вбудований стек TCP/IP, UDP/IP;
- MUX (07.10);
- протоколи HTTP та FTP (крім SIM900R);
- напруга живлення 3,2-4,8 В;

— діапазон робочих температур від  $-40^{\circ}\text{C}$  до  $+85^{\circ}\text{C}$  (можливе незначне відхилення радіочастотних характеристик від специфікації GSM при збереженні функціональної працездатності).

### 2.3.3 Вибір типу живлення пристрою

Безперебійний перетворювач PiGO (Рисунок 2.19) призначений для використання в автомобілях, або в пристроях, де немає стабільного живлення. 5А. А також може використовуватися як зарядний пристрій для смартфонів. Автономна робота залежить від ємності підключеного акумулятора, а також споживання. При відключенні зовнішнього живлення (наприклад, вклучення запалювання в автомобілі), напруга, що виходить, не переривається, що забезпечує безперебійну роботу пристроїв. Блок управління напругою PiGO має характеристики:

Характеристики пристрою:

- вхідна напруга від 4,5 до 30V;
- споживаний струм 6А;
- вихідна напруга 5V;
- вихідний струм 5А;
- робочі температури від  $-40^{\circ}\text{C}$  до  $+85^{\circ}\text{C}$ ;
- час автономної роботи залежить від акумулятора (Li-Pol 3.7V);
- порти USB 2.0 для заряджання 2шт;
- напруга на USB порту 4,85...5,15V;
- вихідний струм USB порту 600mA;
- додатковий USB порт для Raspberry 1шт;
- порт для Raspberry та Arduino 3шт;
- напруга на портах 4,95V;
- струм на кожен порт 2,5А;
- розміри 64x56x12мм;
- вага 75гр.

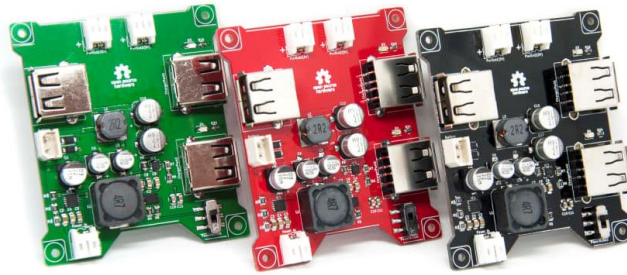


Рисунок 2.19 — Зовнішній вигляд безперебійного перетворювача PiGo

### 2.2.5 Вибір WiFi модуля

ESP8266 — мікроконтролер китайського виробника Espressif з інтерфейсом Wi-Fi (рисунок 2.20).

Окрім Wi-Fi, мікроконтролер здатен виконувати програми з зовнішньої флеш-пам'яті з інтерфейсом SPI.

Мікроконтролер привернув увагу в 2014 році у зв'язку з виходом перших продуктів на його базі за неочікувано низькою ціною.



Рисунок 2.20 — Модуль ESP8266

Характеристики модуля ESP8266:

- 80 MHz 32-bit процесор Tensilica Xtensa L106 де є можливий негарантований розгін до 160 МГц.

- IEEE 802.11 b/g/n Wi-Fi, також підтримується WEP та WPA/WPA2.

— 14 портів вводу-виводу (з них можливо використовувати 11), SPI, I<sup>2</sup>C, I<sup>2</sup>S, UART, 10-bit АЦП.

— Живлення від 2,2 до 3,6 В при споживання до 215 мА в режимі передачі, 100 мА в режимі прийому, 70 мА в режимі очікування, також підтримуються режими зниженого живлення Modem sleep (15 мА), Light sleep (0.4 мА), Deep sleep (15 мкА), всі без зберігання з'єднання з точкою доступу.

Мікроконтролер не має на кристалі енергонезалежної пам'яті для користувача. Виконання програми ведеться із зовнішнього SPI ПЗП шляхом динамічного завантаження необхідних проміжків програми в КЕШ інструкцій. Завантаження виконується апаратно, прозора для програміста. Підтримується до 16 МБ зовнішньої пам'яті програм. Можливий Standard, Dual або Quad SPI інтерфейс.

Виробник не надає документації на внутрішню периферію контролера. Замість цього він надає набір бібліотек, через API яких програміст отримує доступ до периферії. Так як ці бібліотеки інтенсивно використовують ОЗП контролера, то виробник у документах не вказує точну кількість ОЗП на кристалі, а надає лише приблизну оцінку тої кількості пам'яті, що залишається після лінкування бібліотек — близько 50 кБ. Ентузіасти, що дослідили бібліотеки ESP8266, припускають, що він має 32 кБ кешу інструкцій та 80 кБ ОЗП даних.

#### 2.3.4 Вибір датчика температури та вологості

Модуль датчика вологості та температури DHT22 (AM2302) призначений для вимірювання показань вологості та температури навколишнього середовища. Даний модуль дозволяє отримати більш точні значення і має ширший діапазон вимірюваних параметрів, ніж його аналог - датчик DHT11.

Плата модуля запроектована на основі датчика AM2302. Модуль зібраний на базі чутливого ємнісного датчика вологості, NTC-термістора та 8-

бітного АЦП. АЦП необхідний перетворення аналогових сигналів датчика в цифрові. Особливостями датчика є: низьке енергоспоживання та невеликі розміри. Затримка передачі отриманих значень становить 2 секунди. Датчик вимірює відносну вологість навколишнього середовища в діапазоні 0-100%, а температуру від  $-40\text{ }^{\circ}\text{C}$  до  $+80\text{ }^{\circ}\text{C}$ .

Підключається датчик до платформи Raspberry Pi або інших мікроконтролерів за допомогою 3 виходів (VCC, GND і OUT). Через контакт OUT на цифровий порт контролерного пристрою передаються отримані від датчика значення параметрів. Отримані від датчика дані можна переглянути у вікні монітора порту до якого він підключений.

Подавати живлення на плату датчика можна від зовнішнього джерела живлення напругою 5В, плати Raspberry Pi або іншого мікропроцесорного пристрою. Живлення підключається за допомогою контактів VCC (+5) і GND («земля»).

Технічні характеристики датчика:

- напруга живлення від 3,3В до 5,5В;
- діапазон вимірювання вологості (0...100) %;
- точність виміру вологості  $\pm 2\%$ ;
- діапазон вимірювання температури від  $-40^{\circ}\text{C}$  до  $+80^{\circ}\text{C}$ ;
- точність вимірювання температури  $\pm 0.5^{\circ}\text{C}$ ;
- час затримки 2с;
- розміри модуля 38мм x 16мм x 10мм.

## 2.4 Методи виміру об'єму палива

### 2.4.1 Розгляд існуючих методів виміру рівня палива

Як відомо, будь-який автомобіль, навіть найпростіший карбюраторний або дизель з механічним упорскуванням, в обов'язковому порядку оснащений декількома основними датчиками. До таких датчиків можна віднести датчик тиску масла, датчик температури двигуна, датчик, що показує рівень бензину або солярки в паливному баку і т.д.



Авто з електронним упорскуванням (інжекторний двигун) крім основних, також мають цілий ряд додаткових датчиків (залежно від моделі, року випуску тощо). При цьому датчик рівня палива (скорочено ДРП) у паливній системі автомобіля є обов'язковим. Далі ми розглянемо, що таке датчики рівня палива, як працює датчик бензобака, а також де стоїть датчик рівня палива в баку автомобіля.

Датчик рівня палива (ДРП) — пристрій призначений для вимірювання та контролю паливно-мастильних матеріалів (бензин, дизельне паливо, масло) на транспортних засобах та відповідних складах. Розглядати його функції в умовах складської роботи не будемо і зосередимо свою увагу на автомобілях, де він застосовується для позначення рівня палива, конвертації рівня в обсяг, передачі отриманого значення за аналоговим або цифровим шляхом.

Пристрій може використовуватися як доповнення до приладів відображення або контролерами, з можливістю програмування, що мають характеристики вхідних електричних сигналів, що відповідають його технічних вимог.

Головним призначенням датчика рівня палива транспортних засобів є вимірювання об'єму та рівня палива в баках, найчастіше бензобаках. Складається такий пристрій з металевого стрижня, що встановлюється в ємність з паливом (бак), через штатний отвір, або ж спеціально зроблений отвір, що підходить по діаметру. Підключається датчик до моніторингової системи машини, на яку і передається необхідна інформація.

Найчастіше, вони являють собою високоточні пристрої універсального типу, що дає можливість їх використання не тільки на автомобілях, але і на іншій техніці і навіть стаціонарних об'єктах (складах, заводах). Даний пристрій «співпрацює» з покажчиком рівня палива, який розміщений на панелі приладів.

У наш час існує кілька варіантів датчиків рівня палива (електронні, цифрові, поплавкові), але всі вони переслідують одну загальну мету - правильно визначити витрату вмісту бака, щоб кожен водій зміг контролювати свої витрати на пальне. Крім того, пристрій дозволить Вам стежити за обсягом зливу (особливо корисна функція для керівників підприємств, служб таксі та інших організацій, де співробітники можуть зливати казенний бензин) і заправок палива (де, коли і в якій кількості вироблялися).

Щоб мати можливість більш детально контролювати рівень палива слід встановити на транспортний засіб відповідний прилад з підвищеною чутливістю і мінімальною похибкою, що забезпечує максимально точні показники.

Помістити датчик рівня палива можна в будь-який автомобільний бак, у тому числі з дизельною системою. Виводиться показання будуть на панель приладів транспортного засобу, де для цих цілей передбачений спеціальний покажчик.

Сучасні баки мають різну форму і конструкцію, на що обов'язково варто звертати увагу, так як цей факт впливає на кількість пального. Поміщений в бак датчик, не відразу починає подавати правильні сигнали про його наповнення, так як деякий час відображає наповнення всього бака і лише потім цифри опускаються до вірного рівня. Правда, якщо вірити статистиці, жоден такий пристрій не дає абсолютно точних показань.

Принцип роботи датчика базується на відповідності значенню рівня палива певному сигналу, залежно від якого розрізняють пристрої з цифровим та аналоговим сигналом. Аналогові датчики мають високий відсоток недостовірності, через що зараз вже практично не використовуються. Цифровий датчик «переводить» аналоговий сигнал в «цифру», а потім коригує та вирівнює похибки у вимірах, спричинені нерівномірністю рівня палива та геометрії бака. Саме цей вид пристроїв забезпечує найбільш точні показники, а похибки можуть бути присутніми лише на етапі фізичного вимірювання рівня палива.

Сучасні транспортні засоби, в основному, використовують як датчик рівня палива потенціометричний датчик переміщення. Його переваги виражені в простоті, надійності та низькій вартості. Серед недоліків виділяють рухливість контактів, через що, він більшою мірою схильний до зносу і окислення.

Популярні конструкції ДРП: важільний, трубчастий та електронний. Принцип роботи у всіх однаковий. Контролер зчитує положення плавка і передає інформацію на індикатор. Точність та надійність роботи залежить від пристрою обладнання.

Важільний складається з поплавця, рухомого сполучного важеля та потенціо-метра (бігунок з резисторними пластинами).

При повному баку, поплавок розташовується на верхньому рівні. Через важіль він з'єднаний з потенціометром. Останній через індикатор з бортовою електромережею. Опір дорівнює 7 Ом., Заповнення відображається в показаннях приладу на панелі. У міру спалювання бензину рівень падає. Поплавець опускається.

Сполучений з ним бігунок рухається по резисторних пластинах і збільшує опір потенціометра, максимально до 230-240 Ом. Коли він досягає цього значення, індикатор показує, що бак порожній.

Конструкція дуже надійна. Єдиний недолік пов'язаний із зношуванням деталей, через що падає точність.

У конструкцію трубчастого входять поплавець, порожня трубка та контактна група. Датчик працює аналогічно попередньому, але є конструктивні відмінності. Поплавець має контактні кільця. У корпусі є резистивний провід та напрямна стійка. Для підключення управління, передбачений кріпильний фланець.

Бензин надходить у корпус знизу. При повному баку, поплавець розташовується нагорі. У цій точці, на нього замкнутий мінімальний відрізок

контактного проводу, опір - найменший. У міру витрачання палива, поплавки опускається, резистентність збільшується. Внизу вона буде максимальною.

Перевага системи в тому, що поплавець знаходиться в обмеженому просторі. Коливання машини згладжуються, показання набагато точніше, ніж у важільної схеми. Але, є обмеження щодо встановлення.

Електронний датчик — це сучасна модель, в якій деталі не стикаються з паливом. Вузол, що зчитує, з'єднується з важелем через магнітне поле. Похибка вимірів - не більше 1%. Подібні пристрої використовуються, коли машина заправляється бензином на основі метанолу, етанолу або біодизеля. Використання в інших варіантах малоефективні.

Виконавчий вузол датчика розташований у герметичному корпусі. У контакт з паливом вступає лише важіль та магнітний поплавець. Висота поплавка міряється по імпульсах магнітного поля. Фіксація змін ведеться за встановленими відрізками. Їхнє проходження змінює амплітуду зворотного сигналу. Індикатор оновлює показання після того, як датчик пройде чергову позначку.

Для універсального визначення обсягу/рівня рідини в конкретній В резервуарі є широкий спектр доступних технологій вимірювання, включаючи механічні, магнітні, вимірювання тиску (гідростатичні, барботажні, диференціальні), електростатичні (ємнісні, індуктивні).

Механічні датчики резервуара зазвичай визначають положення плавка, що плаває на рідині, за допомогою механічного зв'язку всередині / зовні резервуара.

Магнітні датчики резервуара зазвичай визначають положення поплавця за допомогою механічного зв'язку, прикріпленого від поплавця до магніту. Сучасні магнітні датчики засновані на ефекті Холла (явище, яке виникає, коли електричний струм, що проходить через провідник, піддається впливу зовнішнього магнітного поля, прикладеного під прямим кутом, при

якому електричний потенціал розвивається у провіднику під прямим кутом до обох провідникам). напрямом струму та магнітного поля.).

Сімейство датчиків тиску включає гідростатичні, барботажні та диференціальні датчики. Датчик гідростатичного резервуару зазвичай визначає тиск рідини на дні резервуару. Величина тиску залежить від ваги рідини над датчиком, який, у свою чергу, залежить від кількості рідини в резервуарі. Датчик барботажа ефективно заснований на тому факті, що величина тиску, необхідного для витіснення повітря з нижньої частини труби, залежить від тиску на дні резервуара - тиску, який є результатом кількості рідини в резервуарі. Датчик перепаду тиску визначає різницю в тиску між верхньою нижньою частиною резервуару і перетворює її на кількість рідини.

Датчики електростатичного типу представлені ємнісними та індуктивними. Поміщаючи рідину між електродами, здатність конденсатора накопичувати енергію змінюється, тому змінюється фактична ємність.

Досі для вимірювання рівня палива використовувалися різні методи: резистивна плівка, дискретні резистори, ємнісні та ін. Датчики на основі резисторів є одними з найчастіше використовуваних. Ці датчики механічно пов'язані з поплавком, який переміщається вгору або вниз залежно від рівня палива. У міру руху поплавця опір датчика змінюється, і положення голки змінюється пропорційно струму, що протікає в котушці.

Конденсатор з паралельними пластинами з пластинами, які щільно прилягають до зовнішньої стінки резервуару і доходять до дна резервуару. При зміні рівня палива кількість діелектричного матеріалу між пластинами змінюється, що призводить до зміни ємності. Другий ємнісний датчик, розташований біля дна, діє як опорний канал для проведення логотричних вимірювань. Ємності датчика і зразка перетворюються на цифрову форму, і дані передаються через порт I2C в головний комп'ютер чи мікропроцесор.

Ємнісні датчики досить чутливі до змін умов навколишнього середовища, до того ж вимір ємності складніший у порівнянні з виміром опору.

#### 2.4.2 Створення власного методу виміру рівня палива

Зробивши аналіз всіх наявних методів виміру палива було вирішено провести дослідження щодо додаткових методів виміру палива за допомогою цифрових приладів і без використання механічних частин що можуть зношуватись з часом. В результаті дослідження було взято ультразвукові вимірювачі відстані в якості вимірювальних приладів.

Ультразвукові хвилі виявляють об'єкт так само, як і радар. Ультразвук використовує звукові хвилі, а радар - радіохвилі. Коли ультразвуковий імпульсний сигнал спрямований на об'єкт, він відбивається від об'єкта, а луна повертається до відправника. Відстань до об'єкта визначається виходячи з розрахованого часу проходження ультразвукового імпульсу. Постійно відстежуючи час між відбитими поворотними імпульсами, можна перевірити фактичний рівень рідини.

Ультразвуковий датчик є ключовою частиною ультразвукового передавального пристрою. Цей датчик перетворює електричну енергію на ультразвукові хвилі. П'єзоелектричні кристали мають вирішальне значення для цього процесу перетворення.

Такі кристали або генерують електричні сигнали при отриманні ультразвуку або коливаються на високих частотах, коли до них прикладається електрична енергія.

Для цілей дослідження було обрано датчик відстані HC-SR04. HC-SR04 є приладом безконтактного типу і забезпечує високоточний вимір та стабільність. Діапазон дальності його вимірювання становить від 2 до 400 см. На його роботу не надає суттєвого впливу електромагнітні випромінювання та перепади температури. У комплект модуля з HC SR04 також входять ресивер та трансмітер.

Ультразвуковий далекомір HC SR04 має такі технічні параметри:

- напруга живлення 5В;
- робочий параметр сили струму — 15 мА;
- сила струму пасивному стані < 2 мА;

- оглядовий кут — 15 °;
- сенсорна роздільна здатність — 0,3 см;
- вимірювальний кут — 30 °;
- ширина імпульсу —  $10^{-6}$  с.
- датчик оснащений чотирма виходами (стандарт 2, 54 мм):
- контакт живлення позитивного типу — 5В;
- Trig (T) — вихід сигналу входу;
- Echo(R) — вихід сигналу виходу;
- GND — Земля.

Принцип роботи модуля полягає в вимірі часу, за яке ультразвуковий імпульс досягає перепони і повертається назад. Для запуску процесу вимірювання на вхід "TRIG" треба подати позитивний імпульс запуску тривалістю 10-20mkS. Після закінчення цього імпульсу внутрішній автомат виробляє пачку з 8 періодів ультразвуку частотою 40kHz, яка випромінюється п'єзоелементом "T" (Transmitter). Через приблизно 500mkS після закінчення імпульсу запуску вихід "ECHO" переходить з низького стану в високий, а підключений до цієї лінії контролер повинен почати відлік часу. У момент, коли ультразвукова пачка повернеться назад і потрапить в приймач "R" (Receiver) вихід "ECHO" повернеться в низький стан, а контролер повинен зафіксувати тривалість позитивного імпульсу на цій лінії.

Обчислення відстані покладається на мікропроцесор, в якості якого використовується Raspberry Pi. Формула розрахунку проста: ВІДСТАНЬ = ТРИВАЛІСТЬ ІМПУЛЬСУ "ECHO" \* ШВИДКІСТЬ ЗВУКУ / 2. Для довідки, імпульс тривалістю 1mS відповідає відстані близько 171mm. Слід пам'ятати, що швидкість звуку залежить від температури, вологості, тиску і складу повітря. Так, наприклад, в стандартному повітрі швидкість звуку для температури 0 ° C дорівнює 331.5, для 20 ° C — 343.1 і для 50 ° C — 360.3 m / s. Відповідно до цих значень будуть змінюватися і свідчення далекоміра

(1.7% на кожні 10 ° C). Для збільшення точності потрібно передбачити процес калібрування, який проводиться безпосередньо перед початком вимірювань.

Але для розрахунку об'єму баку потрібні ще додаткові дані окрім відстані від вершини до дна баку. У випадку якщо бак є простої форми наприклад циліндричної то нам необхідно ще знати площу дна баку. Знаючи висоту баку за допомогою далекоміру і знаючи поперечну площу можна дізнатись і об'єм за формулою:

$$V = S * h ,$$

де  $S$  — площа основи;

$h$  — висота баку.

Але в житті не завжди виходить використовувати баки такої простої форми по багатьох причинах. Тому найкращим виходом з цієї ситуації буде попередня калібровка. Підчас калібровки у бак буде поступово заливатись рідина з незмінним об'ємом за проміжок часу. Ультразвуковий датчик буде постійно вимірювати відстань до рідини до поки бак не буде повний. За допомогою знятих даних з датчика можна буде побудувати графік заповнення баку, тобто дізнатись з якою швидкістю в який момент часу рідина піднімалась в гору (рисунок 2.21). Знаючи місткість повного баку і знаючи час за який бак було заправлено можна дізнатись з якою швидкістю бак заповнювався. Ці дані можна співвіднести з даними ультразвукового вимірювача. Тобто якійсь відстані яку виміряв дальномір можна присвоїти значення заповнення баку. Ця інформація може бути записана в таблиці і збережена для подальшого використання.





Рисунок 2.21 – Графік рівня палива при постійній подачі палива бак

### 2.4.3 Випробовування методу виміру рівня палива

Випробовування пройдуть для виміру об'єму води в звичайному відрі. Площа дна відра дорівнює  $S = 2\pi R$ , де  $R$  дорівнює 20 см. Відповідно  $S$  дорівнює  $125,66 \text{ см}^2$ . Встановимо дальномір на верху відра та вимірємо відстань від дна до вершини. Заповнимо відро водою та проведемо виміри (рис. 2.22).

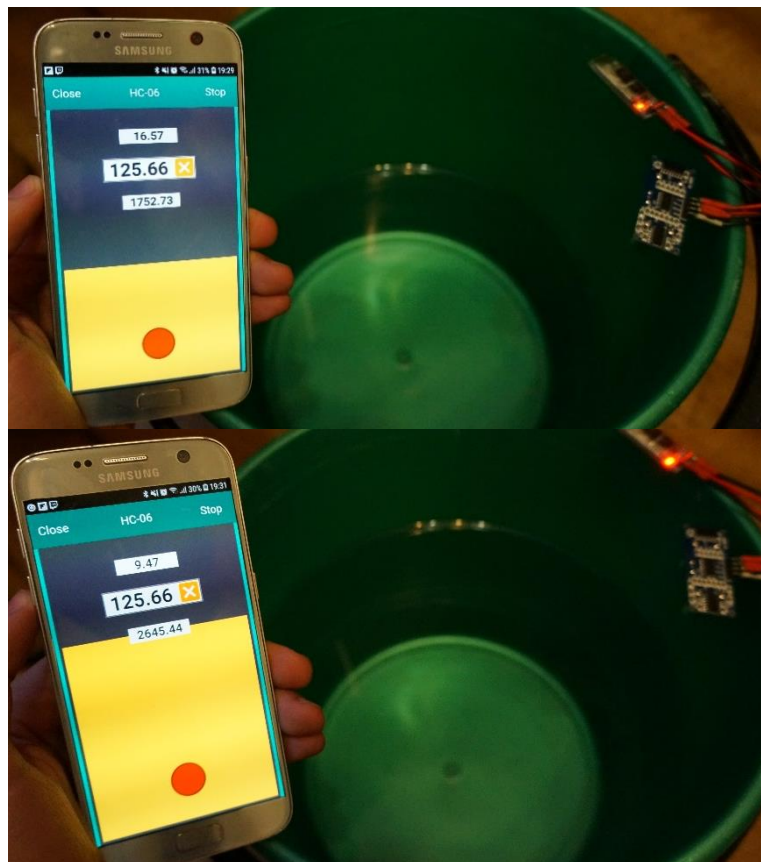


Рисунок 2.22 — Проведення випробовувань пристрою

В результаті випробовувань було перевірено здатність ультразвукового вимірювача відстані отримувати відстань до дна посудини або ж баку, та до речовини що заповнює контейнер.

#### 2.4.4 Комплексний метод виміру рівня палива з використанням різних методів

В результаті дослідження можливостей використання ультразвукового вимірювача відстані було запропоновано використання додаткового методу виміру палива. Причиною такої пропозиції було розповсюджене використання ємнісних ДРП в агропромисловій сфері. Використання комплексу датчиків для виміру рівня палива значно підвищить точність вимірів завдяки більшій кількості отриманих даних. Отримані дані від різних датчиків можуть значно відрізнитись через різний принцип роботи окремого датчика. Наприклад дані з ємнісних датчиків можуть варіюватись в залежності від зовнішньої температури, тобто при різній температурі змінюються властивості вимірюваної речовини, а саме палива. Також недоліком використання ємнісних датчиків є сам факт монтажу датчика що в процесі пошкоджує бак і може вважатись втручанням до важливих механізмів транспорту і призвести до ануляції гарантії.

Також присутня значна похибка при проведенні вимірів ультразвуковим датчиком під час руху транспорту, і відповідно руху баку. Рідина під зовнішніми впливом може хаотично рухатись по баку і створювати похибку вимірів, так як відстань від рівня рідини до датчика буде постійно змінюватись. При наявності комплексної системи виміру рівня палива ємнісний датчик буде отримувати правильні дані, через особливу будову (рисунок 2.23) таких датчиків що нівелює похибку при хаотичному русі рідини.



Рисунок 2.23 – Зовнішній вигляд ємнісного датчика

Також є можливість використання даної системи не тільки для виміру рівня палива. В агропромисловій сфері широко використовуються різні засоби в яких присутні контейнери, які використовуються для різних цілей. Використання такого комплексного методу може значно підвищити продуктивність виконання роботи через підвищену кількість даних про об'єм речовин в цих контейнерах. Наявність різних методів дозволяють використання в різних ситуаціях без втрати функціональності що значно підвищує універсальність системи в цілому.

В результаті дослідження було доведено актуальність використання комплексного методу виміру рівня палива та підтверджено його самобутність.

## **3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ GPS МОНІТОРИНГУ ТА УПРАВЛІННЯ**

### 3.1 Алгоритм роботи системи

Для повноцінної роботи системи необхідно спочатку скласти алгоритм роботи цієї ж системи.

Алгоритм являє собою впорядковану послідовність операцій. Найпростішим і найлегшим для розуміння серед різновидів презентації є блок-схема алгоритму, де всі кроки та дії програми зображені блоками. Щоб система працювала належним чином, спочатку потрібно створити словесний алгоритм, який покроково описує кроки системи. Словесний алгоритм:

Крок 1 — ініціалізація Raspberry Pi і створення змінних.

Крок 2 — підготовка GSM модуля для роботи в інтернеті.

Крок 3 — початок зняття даних з всіх під'єднаних датчиків.

Крок 4 — обрахунок отриманих даних в більш зрозуміліший формат.

Крок 5 — запис даних на пам'ять пристрою.

Крок 6 — відправка записаних даних на сервер.

Крок 7 — перевірка на сигнал вимкнення пристрою.

Блок-схема алгоритму передачі зображена на рисунку 3.1.

Так як система працює в режимах передачі та прийому необхідно скласти алгоритм роботи пристрою при прийомі даних від мобільного додатку. Складемо словесний алгоритм режиму прийому.

Крок 1 — ініціалізація Raspberry Pi і створення змінних.

Крок 2 — підготовка GSM модуля для роботи в інтернеті.

Крок 3 — очікування на команди.

Крок 4 — отримання команди.

Крок 5 — виконання команди.

Крок 6 — відправка даних на сервер про успіх.

Крок 7 — перевірка на сигнал вимкнення пристрою.

Блок-схема алгоритму прийому зображена на рисунку 3.2.



Рисунок 3.1 – Схема алгоритму роботи системи в режимі передачі



Рисунок 3.2 — Схема алгоритму роботи системи в режимі прийому

### 3.2 Інтерфейси і протоколи

UART, SPI та I2C представляють стандартні протоколи зв'язку, які доступні через контакти Raspberry Pi GPIO (загального призначення введення/виведення). Кожен має характеристики, які можуть бути кращими для конкретного проекту (рисунки 3.3). У цій статті буде порівняно ці три методи та висвітлені деякі з їхніх переваг/недоліків.

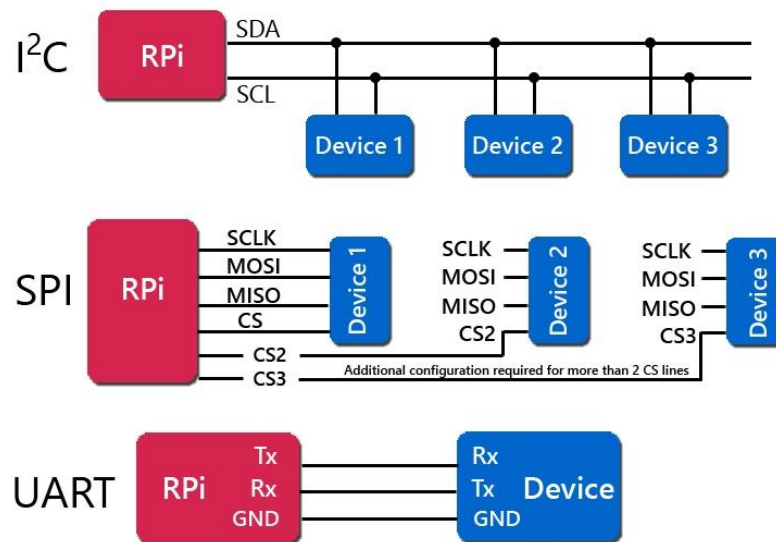


Рисунок 3.3 – Діаграма з'єднань для UART, SPI та I2C

Визначення протоколів:

- UART — універсальний асинхронний приймач/передавач.
- SPI — послідовний периферійний інтерфейс.
- I2C — міжінтегральна схема.

Напівдуплексна міжінтегральна схема I2C використовує послідовну передачу даних, що використовується для коротких відстаней між платами, модулями та периферійними пристроями. Використовує 2 піни. Шина послідовного периферійного інтерфейсу SPI — повнодуплексна послідовна передача даних, що використовується для коротких відстаней між пристроями. Використовує 4 піни. Універсальний асинхронний приймач-передавач UART — асинхронна послідовна передача даних між пристроями. Використовує 2 піни.

Перш ніж обговорити особливості та відмінності між цими методами, давайте розглянемо кілька поширених термінів та визначень комунікації.

— передача даних — це електронна передача інформації через дротові та бездротові засоби;

— протоколи зв'язку — це встановлені методи або правила, що використовуються між пристроями для надсилання або отримання передачі даних;

— шина — це електрофізичні з'єднання (штири, кабелі, роз'єми друкованої плати) між пристроями, які підтримують передачу даних;

— послідовний — це передача даних за допомогою кількох з'єднань, зазвичай 2 або 4, які надсилають/отримують 1 біт інформації за цикл;

— паралельний — це передача даних з використанням кількох з'єднань, зазвичай 8 або більше, паралельно для передачі цілих байтів (або більше) інформації за цикл;

— синхронний — зв'язок з використанням загального сигналу «годинник» для синхронізації передачі даних між пристроями;

— асинхронний — передача даних без потреби в тактовому сигналі для синхронізації передачі;

— напівдуплекс — обмежена передача даних між пристроями в одному напрямку за раз;

— повний дуплекс — можливість передачі даних в обох напрямках одночасно;

— Швидкість передачі даних — швидкість передачі даних, виражена в бітах на секунду (bps), кілобітах на секунду (kbps) або мегабітах на секунду (Mbps).

I2C — це двонаправлений, синхронний, послідовний інтерфейс зв'язку. Якщо працює на двох лініях у напівдуплексному режимі. Спочатку він був створений компанією Philips Semiconductor, яка пізніше стала NXP Semiconductors. Один головний (RPI) може спілкуватися з одним або кількома підлеглими пристроями. Кожен підключений пристрій вибирається за

допомогою 7-розрядної адреси (можлива адресація 10 або більше біт, але більш складна). Спочатку він обмежувався 100 кбіт на секунду, тепер підтримує більш високу швидкість передачі. Pi може підтримувати стандартний режим 100 Кбіт, а також «швидкий режим» 400 Кбіт, причому деякі звіти посилаються на більш високу швидкість залежно від довжини кабелю, моделі плати та програмної обробки.

Завдяки 7-розрядній адресації I2C може підтримувати до 127 пристроїв (або вузлів). Два рядки називаються SCL і SDA. SCL — це тактова лінія для синхронізації передачі. SDA — це лінія даних, через яку відправляються або отримуються біти даних. Під час передачі перший байт містить 7-бітну адресу плюс біт читання/запису. Наступні біти представляють фактичні дані.

Підключення I2C до RPi здійснюється за допомогою контактів плати GPIO 3 для SDA і 5 для SCL (режим BCM GPIO 2 і GPIO 3). RPi GPIO працює при напрузі 3,3 В, тому необхідно подбати про те, щоб підключення до підпорядкованих пристроїв було також 3,3 В. При необхідності можна використовувати перетворювач рівня напруги для інтерфейсу з іншими пристроями, наприклад, 5 В Arduino на RPi 3,3 В.

Відстань проводки I2C вважається відносно короткою, як правило, від дюймів до кількох метрів. На відстань впливають швидкість передачі даних, якість дроту та зовнішні шуми.

SPI — це двонаправлений, синхронний, послідовний інтерфейс зв'язку, як I2C. Також, як і I2C, він використовується для відносно коротких відстаней. Однак, на відміну від I2C, SPI працює в повному дуплексі, тобто дані можна надсилати та отримувати одночасно. Крім того, він може працювати з більш високою швидкістю передачі даних зі швидкістю понад 8 Мбіт або більше на RPi.

SPI може спілкуватися з кількома пристроями двома способами. Перший — це вибір кожного пристрою з рядком Chip Select. Для кожного пристрою потрібен окремий рядок Chip Select. Це найпоширеніший спосіб,



яким на даний момент RPi використовують SPI. Другий — через шлейфове з'єднання, де кожен пристрій під'єднується до іншого через його дані до даних у рядку наступного. Зображення у верхній частині цієї статті ілюструє перший метод.

Немає визначеного обмеження на кількість пристроїв SPI, які можна підключити. Однак існують практичні обмеження через обмеження кількості ліній апаратного вибору, доступних на головному пристрої в першому методі, або складність передачі даних через пристрої у другому методі послідовного ланцюга.

RPi має дві доступні лінії Chip Select (CE0 і CE1). Більше можна визначити, налаштувавши інші контакти GPIO та за допомогою програмного програмування.

Походження UART сягає десятиліть, що робить його прабатьком комунікаційного обладнання / протоколів. Його простота і широке застосування витримали випробування часом і досі є популярним методом передачі даних.

UART — це апаратна реалізація, яка підтримує двонаправлений, асинхронний послідовний зв'язок. Для цього потрібні дві лінії передачі даних - одна для передачі та одна для прийому. Лінія передачі одного пристрою з'єднана з лінією прийому другого пристрою, і навпаки для передачі в обох напрямках.

UART може підключатися лише між двома пристроями. Він може працювати між пристроями в симплексному режимі (передача даних тільки в одному напрямку), напівдуплекс (передача даних в будь-якому напрямку, але не одночасно) або повний дуплекс (передача даних в обох напрямках одночасно).

Також корисно знати що RPi має два UART. Повнофункціональний UART і другий, який називається «міні» UART з меншими можливостями. До RPi 3 повний UART був доступний на контактах 8 і 10 GPIO. Однак, щоб

підтримувати Bluetooth на RPi 3, повний UART був переміщений з GPIO на чіп Bluetooth, а міні-UART був доступний на контактах 8 GPIO. і 10.

Можна перенаправити повний UART на GPIO, але для цього потрібні зміни в конфігурації. Щоб отримати докладнішу інформацію про цю тему, перегляньте документацію UART Raspberry Pi.

Будь-який UART використовує контакт 8 GPIO (BCM GPIO 14) для передачі і контакт 10 (BCM GPIO 15) для прийому.

Швидкість передачі даних UART називається швидкістю передачі даних і за замовчуванням встановлюється на 115 200 (швидкість передачі даних базується на швидкості передачі символів, але схожа на швидкість передачі даних). Швидкість, що наближається до 1 Мбіт/с, була успішною з RPi.

Як і для будь-якого інтерфейсу до RPi GPIO, необхідно враховувати напругу. Виводи RPi працюють при нарузі 3,3 В, тоді як інші пристрої можуть працювати на 5 В або навіть вище. Для взаємодії RPi з такими пристроями необхідні перетворювачі рівня напруги.

На щастя, Raspberry Pi підтримує всі три, і реалізація відносно проста (незважаючи на Pi 3 UART). Деякі переваги / недоліки, які слід враховувати:

- першим є те, що підтримка запланованих пристроїв може обмежити або навіть диктувати ваш вибір;
- UART є найповільніший, I2C швидше і SPI найшвидший, якщо ж швидкість не так важлива, то будь-який з трох хороший.
- кількість пристроїв в UART — всього два (RPi + 1 пристрій), в SPI — багато, але більше двох пристроїв (+ Pi) стає складніше реалізувати з Pi через необхідну лінію CS для кожного пристрою, а I2C — до 127 без зайвої складності.
- UART і SPI можуть бути повним дуплексом, I2C — лише напівдуплекс.

— UART використовує 1 провід (тільки односторонній) або 2 (дво-направлений), I2C використовує 2 дроти, а SPI використовує 4 (плюс дріт заземлення для всіх).

— Жоден з цих методів не є рішенням на великі відстані — від кількох дюймів до кількох метрів залежно від швидкості передачі, якості кабелю та зовнішнього шуму.

— I2C є єдиним протоколом з цих трьох, який гарантує, що надіслані дані були отримані пристроєм, однак це може призвести до стану «заблокованого» одним пристроєм за несприятливих умов.

Передача даних важлива для пристроїв, таких як Raspberry Pi, для зв'язку та обміну «даними» з іншими пристроями. Приклади пристроїв, які може передавати RPi, включають: пристрої відображення, датчики, робототехніку, інші комп'ютери, пристрої введення, промислові засоби керування, наукові прилади.

### 3.3 Програмна реалізація алгоритму

#### 3.3.1 Налаштування Raspberry Pi

Спочатку налаштуємо Raspberry Pi:

Зайдіть через SSH у свій RPI та введіть вказану команду.

```
sudo raspi-config
```

Після цього ми повинні отримати графічний інтерфейс у терміналі. Виберемо пункти меню, виділені червоним кольором (рисунок 3.4).

Після цього виконуємо команду `sudo reboot`.

Проведемо конфігурацію I2C на RPI. Цей крок необхідний для використання бібліотеки `CircuitPython` на кроці 4, оскільки багато драйверів I2C використовуються в цій бібліотеці. По суті, це 2-провідний протокол шини, який дозволяє одному чіпу спілкуватися з іншим.

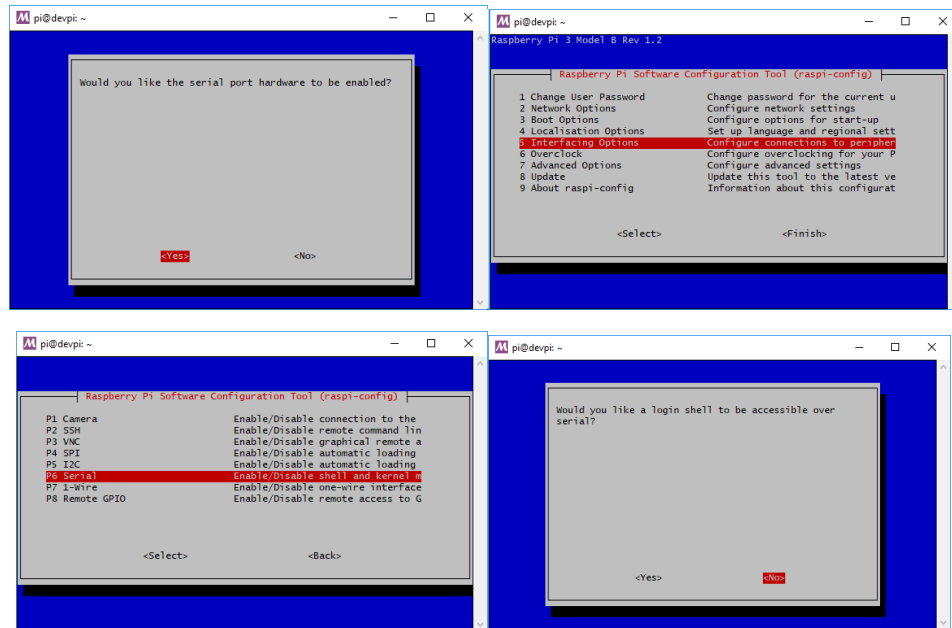


Рисунок 3.4 – Налаштування Raspberry Pi

Спочатку введіть ssh у свій RPI. Далі встановіть утиліту I2C-tools.

```
sudo apt-get install -y python-smbus
```

```
sudo apt-get install -y i2c-tools
```

Далі нам потрібно встановити Kernel Support. Як і раніше, у терміналі введемо `sudo raspi-confi`, після цього знову виконаємо `sudo reboot`.

Проведемо конфігурацію SPI на RPI. SPI (Serial Peripheral Interface) — це ще один протокол шини, який синхронізує послідовний зв'язок між мікросхемами.

Виконайте команду: `sudo raspi-config` і після неї знову `sudo reboot`.

Для перевірки на працездатність коду зайдемо в директорію проекту.

```
mkdir gps
```

```
cd gps
```

Встановимо Raspberry Pi GPIO Library.

```
pip3 install RPI.GPIO
```

Встановимо наш проект.

```
pip3 install adafruit-blinka
```

Тепер створимо скрипт `blinkatest.py`.

```
nano blinkatest.py
```

Збережемо код в тестовий скрипт і збережемо.

```
import board
import digitalio
import busio
print("Hello blinka!")
# Try to great a Digital input
pin = digitalio.DigitalInOut(board.D4)
print("Digital IO ok!")
# Try to create an I2C device
i2c = busio.I2C(board.SCL, board.SDA)
print("I2C ok!")
# Try to create an SPI device
spi = busio.SPI(board.SCLK, board.MOSI, board.MISO)
print("SPI ok!")
print("done!")
```

І нарешті запустимо тестовий скрипт командою `python3 blinkatest.py`.

Встановимо `CircuitPython`. `CircuitPython` — це мова програмування `Adafruit Technology`, яка має на меті «спростити експерименти та навчитися програмувати на дешевих платах мікроконтролера». Таким чином, це найкращий інструмент для швидкого та легкого запуску прикладів програм для різних продуктів, тому ми спрощуємо інструкції зі встановлення на цій сторінці.

Тепер встановимо необхідні залежності.

```
sudo pip3 install adafruit-circuitpython-gps
```

Запустимо тестовий проект для перевірки щоб переконатися, що ми все зробили правильно (як у апаратній, так і в програмній установці), давайте запустимо приклад сценарію в бібліотеці `circuitpython-gps`.

```
cd Adafruit_CircuitPython_GPS
cd examples
Nano gps_simpletest.py
```

Знайдіть і прокоментуйте наступні рядки, щоб переконатися, що програма працює безперебійно з нашим RPI.

```
RX = board.RX
TX = board.TX
```

Після цього знайдіть і замініть стрічку на стрічку нижче з новими даними.

```
#uart = serial.Serial("/dev/ttyS0", baudrate=9600, timeout=3000)
```

Тепер ми повинні бути готові до запуску коду. Переконайтеся, що GPS-модуль має чіткий вид на небо, і запустіть код!

```
python3 gps_simpletest.py
```

### 3.3.2 Налаштування GSM модуля

Необхідно підготувати наступне перед використанням модуля SIM900.

- GPRS-сумісна карта;
- послідовний модуль (TTL до USB, як плата CP2102 USB UART);
- живлення модуля SIM900.

Відкрийте програмне забезпечення послідовного помічника, потім виберіть відповідний послідовний порт і встановіть швидкість передачі даних

на 115200. Перевірте параметри «Новий рядок». Відкривши розширення, ви можете побачити вже встановлені AT-команди. Натисніть відповідну команду, щоб надіслати її безпосередньо.

Перш ніж виконувати кроки демонстрації мережевого зв'язку, описані в таблиці нижче, рекомендується виконати простий тест мережі, щоб підтвердити, що цільове мережеве з'єднання нормальне та працездатне.

Опис команд:

- AT AT Test Command;
- ATE ATE1 — вмикає echo, ATE0 — вимикає echo;
- AT+CSQ запит якості сигналу мережі, повертає значення сигналу;
- AT+CGMR запит версії прошивки;
- AT+CREG? запит статусу реєстрації мережі;
- AT+CGATT? перевірка статусу служби вкладення GPRS.

Програма TCP/IP послідовного модуля SIM900 має два режими підключення, які можна встановити за допомогою команди AT CIPMUX=<n>. Коли для AT CIPMUX встановлено значення 0 (AT+CIPMUX=0), він перебуває в режимі одного зв'язку; коли AT CIPMUX встановлено на 1 (AT+CIPMUX=1), він перебуває в багатоканальному режимі. За замовчуванням модуль налаштований як режим одного зв'язку.

В одноканальному режимі послідовний модуль SIM800 може працювати як у прозорому, так і в непрозорому режимах. В обох режимах передачі модуль може бути налаштований як клієнт TCP/UDP або як сервер TCP.

У багатоканальному режимі послідовний модуль SIM900 може працювати лише в непрозорому режимі.

У цьому режимі модуль може працювати як клієнт абсолютного TCP/UDP-з'єднання, який може встановити загалом 6 з'єднань; його також можна налаштувати як сервер TCP, а також можна налаштувати з 5 клієнтами TCP/UDP. SIM900 TCP/IP за замовчуванням має багатоклієнтську архітектуру, яка підтримує загалом п'ять сокетів, включаючи TCP або UDP.

Програма TCP/IP послідовного модуля SIM900 має два режими підключення, які можна встановити за допомогою команди AT+CIPMUX=<n>.

Відправлення повідомлення AT+CMGF=1: Встановлює режим SMS на ТЕКСТ AT+CMGS="номер телефону"<Enter>, встановлює номер мобільного телефону одержувача, а потім повертає: ">" Щоб надіслати необхідний вміст, наприклад «Надіслати тестове повідомлення!», не потрібно натискати enter в кінці. Після редагування короткого повідомлення надішліть «1A» у шістнадцятковому форматі для відправлення повідомлення. (1A — це ключове значення «CTRL+Z», яке використовується, щоб сповістити модуль виконати операцію «відправити». Щоб скасувати операцію, надішліть 1B або «ESC»)

Після завершення передачі модуль повертає +CMGS:15, щоб підтвердити успішність передачі.

Надішліть це повідомлення на телефон: "Це тест прийому для SIM900!" до тестового модуля.

При отриманні інформації послідовний порт розбиває звіт до певного формату, наприклад «SM», 20. Це означає, що в SM є 20 повідомлень, а надіслане повідомлення є 20-м.

Для читання повідомлень використовуйте AT+CMGR=20. Це читає 20-е повідомлення (AT+CMGL="ALL" читає всі повідомлення)

Для видалення повідомлень: AT+CMGD=20, як показано на малюнку нижче. Відображена інформація перетворюється в текст за допомогою транскодера.

Опишемо підключення GSM модуля. Забезпечте окреме джерело живлення (5 В 1 А) для GSM-модему для належного функціонування, оскільки він споживає великий струм. Вставимо SIM-карту, активовану 2G, з робочим з'єднанням для передачі даних та ввімкніть модулі. Для реєстрації SIM-карти в мережі знадобиться кілька секунд. Тепер нам потрібно перевірити зв'язок між RPi і модемом. Для цього ми використовуємо код на Python, який



передає команду AT і перевіряє, чи отримано ОК як підтвердження. Відкрийте оболонку Python за допомогою команди `sudo idle` та створіть новий файл.

```

/etc/chatscripts/gprs
ABORT BUSY
ABORT VOICE
ABORT "NO CARRIER"
ABORT "NO DIALTONE"
ABORT "NO DIAL TONE"
ABORT "NO ANSWER"
ABORT "DELAYED"
ABORT "ERROR"
ABORT "+CGATT: 0"
"" AT
TIMEOUT 12
OK ATH
OK ATE1
OK AT+CGDCONT=1,"IP","telia"
OK ATD*99#
TIMEOUT 22
CONNECT ""

```

Також створимо файл для GSM.

```

/etc/ppp/peers/a-gsm
connect "/usr/sbin/chat -v -f /etc/chatscripts/gprs -T telia"
/dev/ttyAMA0
115200
noipdefault
usepeerdns
defaultroute

```

```
persist
noauth
nocrtscts
local
```

Для під'єднання до інтернету ми просто запускаємо команду

```
sudo pon a-gsm.
```

### 3.3.3 Реалізація обрахунку відстані ультразвуковим датчиком

Спочатку імпортуємо бібліотеку Python GPIO, імпортуємо нашу бібліотеку часу і встановлюємо нумерацію контактів GPIO.

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
```

Далі нам потрібно назвати наші вхідні та вихідні контакти, щоб ми могли посилатися на них пізніше в нашому коді Python. Ми назвемо наш вихідний контакт (який запускає датчик) GPIO 23 [Pin 16] як TRIG, а наш вхідний контакт (який зчитує зворотний сигнал з датчика) GPIO 24 [Pin 18] як ECHO.

```
TRIG = 23
ECHO = 24
```

Далі встановіть свої два порти GPIO як входи або виходи, як було визначено раніше.

```
GPIO.setup(TRIG,GPIO.OUT)
GPIO.setup(ECHO,GPIO.IN)
```

Потім переконайтеся, що штифт тригера встановлений на низький рівень, і дайте датчику секунду, щоб він відкоригувався.

```
GPIO.output(TRIG, False)
print "Waiting For Sensor To Settle"
time.sleep(2)
```

Датчику HC-SR04 потрібен короткий імпульс 10 мкс для запуску модуля, що змусить датчик запустити програму визначення дальності (8 ультразвукових сигналів на 40 кГц), щоб отримати ехо-відповідь. Отже, щоб створити наш тригерний імпульс, ми встановлюємо на контакті тригера високий рівень на 10 мкс, а потім знову встановлюємо низький рівень.

```
GPIO.output(TRIG, True)
time.sleep(0.00001)
GPIO.output(TRIG, False)
```

Тепер, коли ми надіслали наш імпульсний сигнал, нам потрібно прослухати наш вхідний контакт, який підключений до ЕЧНО. Датчик встановлює значення ЕЧНО на високий час, протягом якого імпульс проходить і повертається, тому наш код повинен виміряти кількість часу, протягом якого висновок ЕЧНО залишається високим. Ми використовуємо рядок “while”, щоб гарантувати, що кожна позначка часу сигналу записується в правильному порядку.

Функція `time.time()` запише останню мітку часу для даної умови. Наприклад, якщо `pin` переходить з низького рівня на високий, і ми записуємо низький стан за допомогою функції `time.time()`, записаною міткою часу буде останній час, коли цей `pin` був низьким.

Тому першим кроком має бути запис останньої нижньої позначки часу для ЕЧНО (`pulse_start`), наприклад, безпосередньо перед тим, як буде отриманий зворотний сигнал і напруга на `pin` підвищиться.

```
while GPIO.input(ECHO)==0:
    pulse_start = time.time()
```

Після отримання сигналу значення змінюється з низького (0) на високий (1), і сигнал залишатиметься високим протягом тривалості ехо-імпульсу. Тому нам також потрібна остання висока мітка часу для ЕЧНО (pulse\_end).

```
while GPIO.input(ECHO)==1:
    pulse_end = time.time()
```

Тепер ми можемо обчислити різницю між двома записаними мітками часу, а отже, і тривалість імпульсу (pulse\_duration).

```
pulse_duration = pulse_end - pulse_start
```

Враховуючи час, необхідний для того, щоб сигнал рухався до об'єкта і назад, ми можемо обчислити відстань, використовуючи формулу що визначили в другому розділі. Ми можемо підключити це обчислення до нашого сценарію Python:

```
distance = pulse_duration x 17150
```

Тепер нам потрібно округлити нашу відстань до 2 знаків після коми

```
distance = round(distance, 2)
```

Потім друкуємо відстань. Наведена нижче команда надрукує слово «Distance:», за яким слідує змінна відстані, а потім одиниця вимірювання «см».

```
print "Distance:",distance,"cm"
```

Нарешті, ми очищаємо контакти GPIO, щоб переконатися, що всі входи/виходи скинуті

```
GPIO.cleanup()
```

Збережіть свій сценарій Python, і запустіть його за допомогою наступної команди. Запуск root (sudo) є важливим для цього сценарію (Рисунок 3.5).

```
pi@raspberrypi ~ $ sudo python range_sensor.py
Distance Measurement In Progress
Waiting For Sensor To Settle
Distance: 12.52 cm
pi@raspberrypi ~ $ █
```

Рисунок 3.5 – Результат виміру відстані

### 3.3.4 Налаштування модуля температури та вологості

Також потрібно підключити датчик виміру температури який буде використовуватись для виміру температури двигуна.

Використовуючи pip, ми встановимо бібліотеку DHT Adafruit на Raspberry Pi. Ми будемо використовувати цю бібліотеку Python для взаємодії з нашим датчиком вологості/температури DHT22. Як бонус, бібліотека також підтримує датчики вологості/температури DHT11 і AM2302, що робить її чудовою бібліотекою для вивчення способів використання. Виконаємо таку команду, щоб встановити бібліотеку DHT на ваш Raspberry Pi.

```
sudo pip3 install Adafruit_DHT
```

Тепер, коли ми встановили бібліотеку Adafruit DHT на нашому Raspberry Pi, ми можемо перейти до програмування з нею.

Спочатку ми повинні створити скрипт Python, в якому ми будемо зберігати весь наш код. Щоб почати писати цей файл, ви можете ввести наступну команду у свій Raspberry Pi.

```
nano ~/humidity.py
```

У цьому файлі нам потрібно ввести наступні рядки коду

```
import Adafruit_DHT
```

Цей рядок імпортує бібліотеку «Adafruit\_DHT», яку ми отримали за допомогою `pip` у попередньому розділі.

Ми будемо використовувати цю бібліотеку для розмови та взаємодії з датчиком DHT22. Бібліотека дозволяє нам дуже легко отримати температуру та вологість із датчика за допомогою кількох рядків коду на Python.

```
DHT_SENSOR = Adafruit_DHT.DHT22
```

У цьому рядку ми визначаємо константу для датчика, який ми хочемо використовувати для бібліотеки DHT. Бібліотека Adafruit DHT має три варіанти, з яких ви можете вибрати залежно від використовуваного датчика.

Ми використовуємо DHT22, тому ми визначили нашу константу «DHT\_SENSOR» як «Adafruit\_DHT.DHT22».

```
DHT_PIN = 4
```

Далі ми визначаємо константу під назвою «DHT\_PIN», щоб зберегти номер контакту GPIO, до якого підключений DHT22 на Raspberry Pi.

Число, яке ми використовуємо, відноситься до нумерації VCM контактів GPIO. У нашому підручнику це означає, що нам потрібно використовувати цифру «4», оскільки ми використовували контакт GPIO4 для нашого з'єднання даних.

Якщо ви використовували інший контакт GPIO, переконайтеся, що ви відповідно оновили цей номер.

```
while True:
```

```
    humidity, temperature = Adafruit_DHT.read_retry(DHT_SENSOR,
DHT_PIN)
```

Далі ми запускаємо нескінченний цикл, використовуючи код «while True:». Цей цикл буде виконуватися безперервно, доки користувач не вб'є скрипт Python.

На початку кожного циклу ми використовуємо бібліотеку Adafruit DHT, щоб отримати як вологість, так і температуру з датчика DHT22, вказуючи наші константи DHT\_SENSOR і DHT\_PIN.

Функція «read\_retry» буде постійно намагатися отримати дані з датчика вологості.

```
if humidity is not None and temperature is not None:
    print("Temp={0:0.1f}*C Humidity={1:0.1f}%" .format(temperature,
humidity))
else:
    print("Failed to retrieve data from humidity sensor")
```

Нарешті, ми перевіряємо, чи бібліотека DHT повернула корисні для нас дані.

Якщо змінні «вологість» або «температура» мають значення «Немає», ми повертаємо кінцевому користувачеві повідомлення про те, що нам не вдалося отримати дані з нашого датчика вологості.

Якщо ми мали значення в обох цих змінних, ми роздруковуємо повідомлення, яке показує кінцевому користувачеві як температуру, так і вологість, які ми щойно отримали від нашого датчика вологості.

Давайте запусимо сценарій, щоб ми могли побачити дані від датчика вологості.

Ви можете запустити цей сценарій, виконавши таку команду.

```
python3 ~/humidity.py
```

За допомогою цієї команди ви повинні побачити результати, подібні до того, що ми маємо нижче.

```
Temp=25.6*C Humidity=51.8%
```

```
Temp=25.2*C Humidity=45.4%
```

```
Temp=25.1*C Humidity=45.4%
```

```
Temp=25.0*C Humidity=45.3%
```

Тепер занесемо отримані дані в пам'ять пристрою.

Почнемо із виконання наступної команди, щоб почати писати наш новий сценарій.

```
nano ~/humidity_logger.py
```

У цьому файлі введіть наступні рядки.

```
import os
import time
import Adafruit_DHT
```

Тут ми додаємо імпорт для бібліотек «os» і «time». Ми використовуємо бібліотеку «os», щоб перевірити, чи існує наш файл CSV, перш ніж спробувати записати в нього.

Ми використовуємо бібліотеку «час», щоб ми могли позначити час кожного нового рядка з поточною датою та часом.

```
DHT_SENSOR = Adafruit_DHT.DHT22
DHT_PIN = 4
```

З цими двома рядками немає різниці, зберігайте їх, як у попередньому сценарії.

```
try:
    f = open('/home/pi/humidity.csv', 'a+')
    if os.stat('/home/pi/humidity.csv').st_size == 0:
        f.write('Date,Time,Temperature,Humidity\r\n')
except:
    pass
```

Тут ми обгортаємо все в оператор try для обробки будь-яких потенційних помилок.



Ми відкриваємо дескриптор нашого файлу «/home/pi/humidity.csv» із застосованим тегом «a+». Тег a+ означає, що будь-які дані, записані у файл, будуть додані в його кінець.

Далі ми використовуємо бібліотеку “os”, щоб перевірити, чи ми коли-небудь писали в цей файл раніше. Якщо розмір повертається як 0, ми записуємо початковий рядок у файл. Цей рядок міститиме заголовки наших стовпців, щоб вам було легше зрозуміти дані.

```
while True:
```

```
    humidity, temperature = Adafruit_DHT.read_retry(DHT_SENSOR,
DHT_PIN)
```

Ось у нас цикл while. Ми використовуємо бібліотеку DHT, щоб знову зчитувати дані з нашого датчика вологості DHT22.

Ця функція зберігатиме дані, які вона зчитує з датчиків, у наші змінні «вологість» і «температура».

```
    if humidity is not None and temperature is not None:
```

```
        f.write('{0},{1},{2:0.1f}*C,{3:0.1f}%\r\n'.format(time.strftime('%m/%d/
%y'), time.strftime('%H:%M'), temperature, humidity))
```

```
    else:
```

```
        print("Failed to retrieve data from humidity sensor")
```

```
        time.sleep(30)
```

Наша зміна в цьому розділі коду полягає в тому, що замість того, щоб друкувати температуру та вологість на консолі, ми записуємо їх у наш файл «humidity.csv». Ми робимо це за допомогою дескриптора файлу, який ми відкрили раніше.

Ми форматуємо текст, записаний у файл, щоб включати поточну дату, поточний час, нашу відформатовану температуру та вологість.

### 3.3.5 Підключення протоколу MQTT

Тепер підключимо MQTT – це свого роду легкий протокол обміну повідомленнями IoT, заснований на моделі публікації/підписки, який може надавати надійний сервіс обміну повідомленнями в режимі реального часу для пристроїв Інтернету речей, лише використовуючи дуже мало коду та пропускну здатність. Він підходить для пристроїв з обмеженими апаратними ресурсами та мережевого середовища з обмеженою пропускну здатністю. Тому протокол MQTT широко використовується в IoT, мобільному Інтернеті, IoV, електроенергетиці та інших галузях.

У цьому проекті ми будемо використовувати Python, щоб написати простий клієнт MQTT на Raspberry Pi і реалізувати підключення, підписку, скасування підписки, обмін повідомленнями та інші функції між цим клієнтом і брокером MQTT.

Для розробки цього проекту використовується Python3. Зазвичай Raspberry Pi має вбудований Python3. Якщо ви не впевнені, чи було встановлено Python3, ви можете скористатися такою командою для підтвердження.

```
python3 --version
```

Якщо відображається Python 3.x.x (x означає число), Python 3 встановлено. В іншому випадку скористайтеся командою apt для встановлення (або дотримуйтеся інструкцій із встановлення Python3).

```
sudo apt install python3
```

Встановимо клієнтську бібліотеку MQTT. Нам потрібно встановити бібліотеку paho-mqtt для легкого підключення брокера MQTT. Ви можете вибрати один із наступних двох методів встановлення.

Використовуйте вихідний код для встановлення

```
git clone https://github.com/eclipse/paho.mqtt.python
cd paho.mqtt.python
```

```
python3 setup.py install
```

Використовуйте `pip3` для встановлення

```
pip3 install paho-mqtt
```

Доступ до інформації брокера виглядає наступним чином:

— брокер: `broker.emqx.io`;

— порт TCP: 1883;

— порт WebSocket: 8083.

Якщо це необхідно, ви можете використовувати `docker` для швидкого локального встановлення брокера EMQ X.

```
docker run -d --name emqx -p 1883:1883 -p 8083:8083 -p 8084:8084 -p 18083:18083 emqx/emqx
```

Код підключення наведено в додатку Б.

Ми судимо про код відповіді у функції `on_connect`, якщо він дорівнює 0, появиться напис «Підключено успішно», щоб відобразити успішне підключення. Якщо він повертає інші числа, ми повинні оцінити це відповідно до коду відповіді нижче:

— 0 — підключення вдалось;

— 1 — з'єднання не вдалося — неправильна версія протоколу;

— 2 — не вдалося підключитися — недійсний ідентифікатор клієнта;

— 3 — з'єднання не вдалося — брокер недоступний;

— 4 — не вдалося підключитися — неправильне ім'я користувача або пароль;

— 5 — з'єднання не вдалося — неавторизоване;

— 6-255 — не визначено.

У концепції протоколу MQTT повідомлення передається через тему. Наприклад, один пристрій надсилає повідомлення на тему `T`, тільки пристрої, які підписалися на тему `T`, можуть отримати повідомлення. Тому

лише доступ до брокера MQTT не має сенсу. Якщо ви хочете повністю використовувати службу MQTT, нам все одно потрібно знати, як опублікувати та підписатися.

Код підключення на підписку наведено в додатку Б.

Виклик функції `subscribe()`, яка може дозволити Raspberry Pi підписатися на тему. В наведений нижче код, ми використовуємо його, щоб підписатися на `topic raspberry/topic` та відстежувати повідомлення.

Крім того, ми також використовуємо `will_set()` для встановлення повідомлення волі. Повідомлення заповіту є функцією MQTT, коли пристрій випадково вимикається, він надсилає повідомлення на вказану тему. Ми можемо знати, чи Raspberry Pi вимкнено, чи мережа ненормальна.

Код публікації повідомлень наведено в додатку Б.

Виклик функції `publish()` може надсилати повідомлення до теми. У наведеному вище коді ми використовуємо його для надсилання повідомлень на тему `raspberry/topic`. Параметр `QoS` є ще однією функцією MQTT, якщо ви хочете дізнатися більше про вміст `QoS`, ви можете переглянути введення в `QoS MQTT` (якість обслуговування).

Проведемо тестування за допомогою `MQTT 5.0 client tool - MQTT X`. Запустимо код і відправимо повідомлення.

```
python3 subscriber.py
```

Використаємо `MQTT X client` для з'єднання до `MQTT broker` і відправимо повідомлення до `raspberry/topic` (рисунок 3.6).

Після відправки можемо побачити повідомлення в терміналі (рисунок 3.7).

Тепер опублікуємо повідомлення через Raspberry Pi. Підпишемося до `raspberry/topic` в `MQTT X Client`. Запустимо код в терміналі для відправки повідомлень (рисунок 3.8). Первіримо `MQTT X client` на наявність відправлених повідомлень (рисунок 3.9).

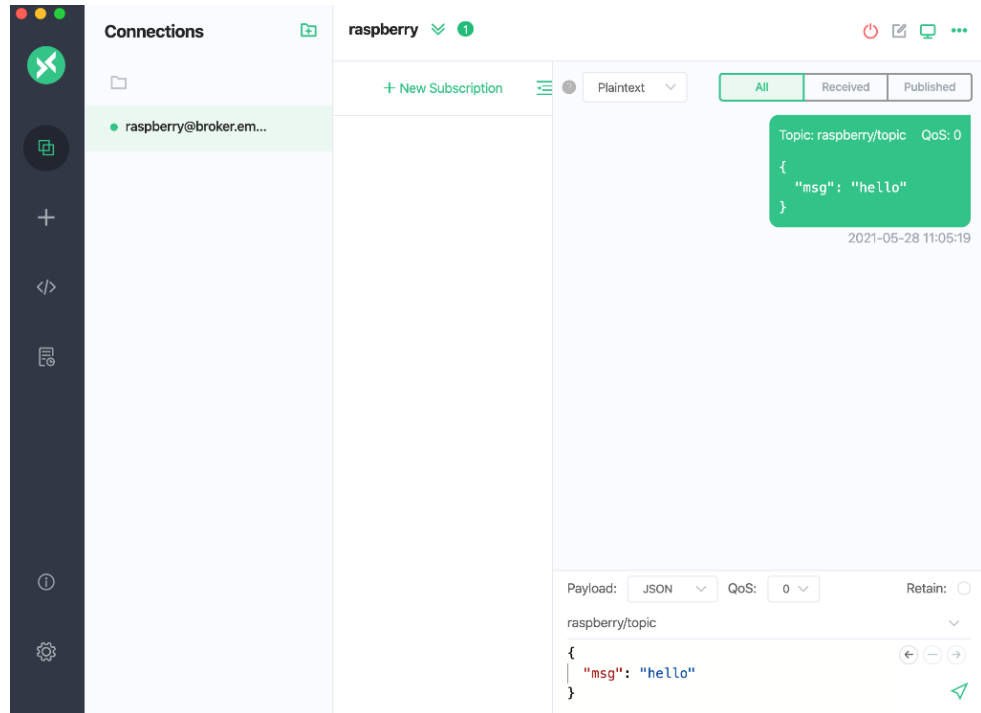


Рисунок 3.6 — Відправка повідомлення через MQTT X

```
~/Desktop $ python3 subscriber.py
Connected with result code 0
raspberry/topic b'{"msg": "hello"}'
```

Рисунок 3.7 — Повідомлення в терміналі

```
~/Desktop $ python3 publisher.py
send 0 to raspberry/topic
send 1 to raspberry/topic
send 2 to raspberry/topic
send 3 to raspberry/topic
send 4 to raspberry/topic
Connected with result code 0
```

Рисунок 3.8 — Відправка повідомлень в терміналі

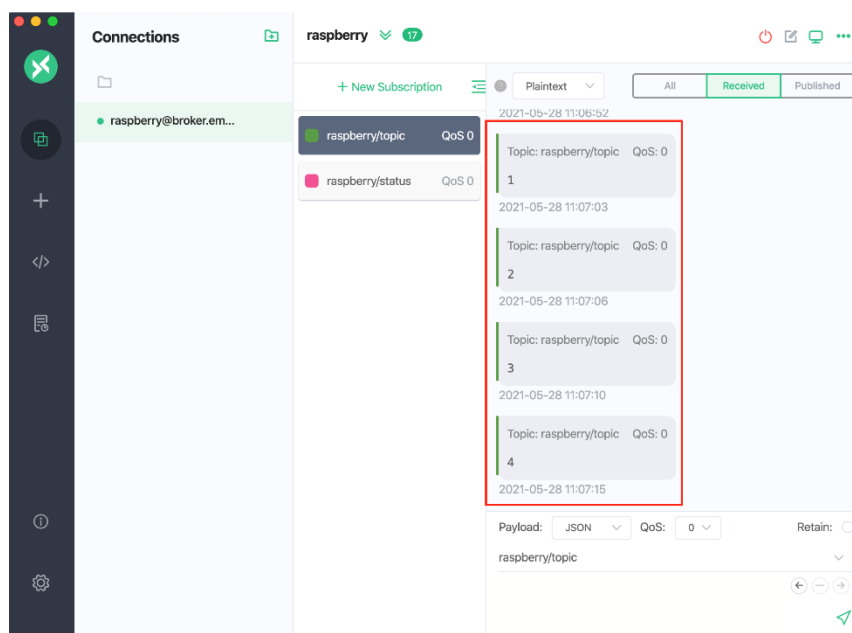


Рисунок 3.9 — Отримані повідомлення в MQTT X

### 3.3.6 Керування системою з мобільного додатку

Тепер необхідно реалізувати метод керування реле з мобільного додатку. Встановимо Pythonic. Найпростіший спосіб зробити це – вставити карту SD попередньо налаштованим образом Pythonic, доступним на [sourceforge.net](https://sourceforge.net).

Завантажте та розпакуйте образ та зашивайте його на SD-карту Raspberry Pi.

Підключіть SD-карту та перевірте, на якому пристрої вона відображається, ввівши `lsblk -p`.

Використовуйте `lsblk -p`, щоб перевірити, на якому пристрої відображається ваша SD-карта.

На знімку екрана вище пристрій SD-карти — це `/dev/sdc`, і мої системи автоматично змонтували два знайдені на ньому розділи. Якщо це так, розмонтуйте його, ввівши `umount /dev/sdc1 && umount /dev/sdc2`.

Запрограмуємо SD-карту такою командою:

```
dd if=~/.Downloads/Pythonic-1.7.img of=/dev/sdc bs=32M, conv=fsync.
```

Процес перепрошивки займе деякий час.

Після завершення процесу вставте SD-карту назад у Raspberry Pi і завантажте її.

Образ Pythonic не має попередньо встановленого робочого столу. Вся конфігурація базується на Інтернеті, тому ви повинні встановити з'єднання TCP/IP. Підключитися за допомогою звичайного інтернет-роутера нескладно. Якщо у вас немає доступу до такого маршрутизатора, ви також можете встановити з'єднання через бортовий універсальний асинхронний приймач/передавач (UART), щоб налаштувати інтерфейс Ethernet або WiFi.

За замовчуванням образ Pythonic налаштований на отримання IP-адреси за допомогою DHCP. На вашому домашньому інтернет-маршрутизаторі зазвичай працює DHCP-сервер, який роздає IP-адреси підключеним пристроям. З'єднайте вільний порт Ethernet вашого роутера з портом Ethernet на Raspberry Pi і завантажте його.

Тепер є можливість спробувати отримати доступ до веб-графічного інтерфейсу користувача з пристрою в локальній мережі. Якщо DNS у вашій локальній мережі працює належним чином, відкрийте браузер і перейдіть до <http://PythonicRPI:7000/>, щоб відкрити графічний інтерфейс програмування.

Маршрутизатор також пропонує графічний інтерфейс конфігурації. Графічний інтерфейс конфігурації надає інформацію про пристрої у локальній мережі. Ви можете знайти IP-адресу свого локального маршрутизатора, ввівши `ip route`.

У нашому випадку маршрут доступний під 192.168.188.1. Тепер увійдіть на сторінку конфігурації вашого маршрутизатора і перевірте, який IP був наданий Raspberry Pi.

У нашій мережі Raspberry Pi доступний за адресою 192.168.188.63, тому ми можемо отримати доступ до графічного інтерфейсу за адресою <http://192.168.188.63:7000/>.

Вставте SD-карту назад у пристрій для зчитування карт і змонтуйте завантажувальний розділ. Відкрийте файл `config.txt` на завантажувальному розділі та додайте в кінець наступний рядок:

```
enable_uart=1
```

Вставте SD-карту назад у Raspberry Pi та завантажте її. Тепер ви можете встановити консольне з'єднання за допомогою конвертера UART-USB, щоб налаштувати статичну IP-адресу або налаштувати з'єднання Wi-Fi.

Параметри підключення за замовчуванням:

- TxD — GPIO14;
- RxD — GPIO15;
- GND — контакт 6 або 14;
- швидкість передачі даних — 115200;
- біт даних — 8;
- біт парності — немає;
- стоп-біти — 1.

Щоб продовжити наступні кроки, завантажимо файли Pythonic з github на свій локальний жорсткий диск.

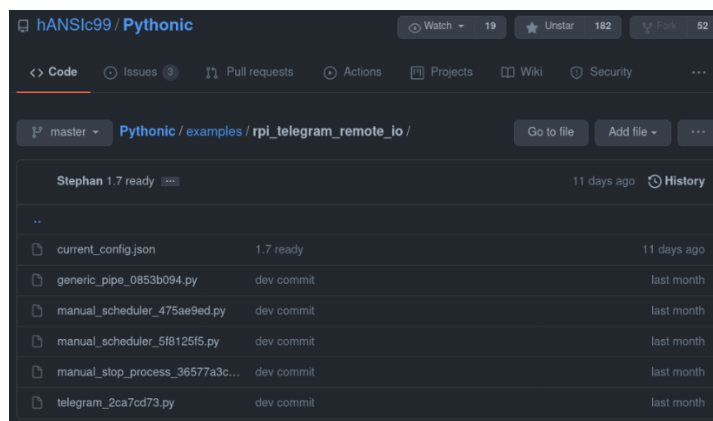


Рисунок 3.6 — Завантаження коду Pythonic з github

Тепер завантажимо конфігурацію.

За допомогою кнопки, позначеної синім кольором, ви завантажуєте current\_config.json. Ви можете завантажувати лише дійсні файли конфігурації. Після завантаження ви можете знайти файл на цільовій сторінці в /home/pythonic/Pythonic/current\_config.json.



За допомогою зеленої кнопки ви завантажуєте кожен \*.ру-файл. Після цього файли \*.ру можна знайти в /home/pythonic/Pythonic/executables.

У папку виконуваних файлів можна завантажувати будь-які файли, оскільки я планую підтримувати двійкові виконувані файли в майбутньому.

Однак, щоб конфігурація працювала, фактична реалізація повинна бути доступна для кожного елемента, описаного в current\_config.json.

Але це налаштування поки не працюватиме. Спробуйте запустити цю конфігурацію, натиснувши кнопку «Відтворити» на елементі ManualScheduler - 0x5f8125f5. Підключений елемент додатку запуститься, але потім негайно вийде. Це тому, що елемент додаток потребує додаткової конфігурації.

Ви повинні надати токен додатку для зв'язку з сервером.

Тепер ви зможете запустити елемент додатку, натиснувши кнопку відтворення на елементі ManualScheduler - 0x5f8125f5. Тепер елемент має бути активним.

Смужка, що обертається в нижньому інформаційному рядку, вказує на робоче з'єднання з бекендом.

Відкрийте нову вкладку у своєму браузері та перейдіть до <http://PythonicRPI:8000/>. Це відкриє попередньо встановлену IDE сервера коду.

Тепер ми зможемо налаштувати роботу додатку і керувати Raspberry Pi.

Ми можемо адаптувати додаток, як подобається: змінювати або додавати додаткові GPIO, використовувати аналогові функції або отримати стан введення на вимогу. Якщо ви підключите відповідне реле, ви також можете керувати запуском двигуна за допомогою Raspberry Pi.

## 4 ЕКОНОМІЧНА ЧАСТИНА

Метою економічної частини магістерської кваліфікаційної роботи є довести економічну доцільність та ефективність впровадження наукової розробки, для цього необхідно виконати такі етапи:

- оцінити комерційний потенціал розробки;
- спрогнозувати витрати на виконання наукової роботи та впровадження її результатів;
- спрогнозувати комерційний ефект від реалізації результатів розробки;
- розрахувати ефективність вкладених інвестицій та період їх окупності.

### 4.1 Оцінювання економічного потенціалу розробки

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Для проведення технологічного аудиту залучено трьох незалежних експертів. Кожен з експертів повинен ознайомитися з запропонованою розробкою, та заповнити таблицю, яка визначає рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можливу оцінку в балах. Після виконання цього, підраховується середньоарифметична сума балів та визначається який рівень комерційного потенціалу має нова розробка.

Для проведення технологічного аудиту залучено трьох незалежних експертів. Кожен з експертів повинен ознайомитися з запропонованою розробкою, та заповнити таблицю, яка визначає рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можливу оцінку в балах. Після виконання цього, підраховується середньоарифметична сума балів та визначається який рівень комерційного потенціалу має нова розробка.

Оцінювання комерційного потенціалу розробки здійснюється за критеріями, наведеними в таблиці 4.1.

Таблиця 4.1 — Рекомендовані критерії оцінювання комерційного потенціалу розробки та їх можлива бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає

Продовження таблиці 4.1

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування
10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання комерційного потенціалу наведено в табл. 4.2.

Таблиця 4.2 — Результати оцінювання комерційного потенціалу розробки

Критерії	Експерти		
	Експерт 1	Експерт 2	Експерт 3
	Бали, виставлені експертами:		
1	4	3	3
2	3	2	2
3	4	3	3
4	4	2	2
5	3	2	3
6	2	2	1
7	2	1	1
8	3	2	2
9	1	1	1
10	4	4	4
11	4	3	3
12	4	4	4
Сума балів	СБ <sub>1</sub> =38	СБ <sub>2</sub> =29	СБ <sub>3</sub> =29
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_i СБ_i}{i} = \frac{38 + 29 + 29}{3} = 32$		

Отже, з отриманих даних таблиці 4.2 видно, що середньоарифметична сума балів дорівнює 32, тобто нова розробка має рівень комерційного потенціалу вище середнього.

#### 4.2 Прогнозування витрат на виконання наукової роботи та впровадження її результатів

Прогнозування витрат на виконання науково-дослідної чи дослідно-конструкторської роботи може складатися з таких етапів:

- розрахунок витрат, які безпосередньо стосуються виконавців даного розділу роботи.
- розрахунок загальних витрат на виконання даної роботи.
- прогнозування загальних витрат на виконання та впровадження результатів даної роботи.

Розрахунок витрат можна здійснити за такими статтями та формулами:

Основна заробітна плата із розробників (дослідників)  $Z_o$ , якщо вони працюють в наукових установах бюджетної сфери:

$$Z_o = \frac{M}{T_p} \cdot t \text{ грн.}, \quad (4.1)$$

де  $M$  — місячний посадовий оклад конкретного розробника, наукового керівника;

$T_p$  — кількість робочих днів у місяці,  $T_p = 23$  дні.

$t$  — число робочих днів роботи розробника (дослідника).

Заробітна плата розробника:

$$Z_p = \frac{11000}{23} \cdot 65 = 31086,95 \text{ (грн)}$$

Заробітна плата наукового керівника проекту:

$$Z_{HK} = \frac{9000}{23} \cdot 22 = 8608,69 \text{ (грн)}$$

Витрати на оплату праці, основна заробітна плата:

$$Z_o = Z_p + Z_{HK} = 31086,95 + 8608,69 = 39695,64 \text{ (грн)}$$

Таблиця 4.3 — Розрахунки основної заробітної плати спеціалістів

Найменування посади виконавця	Місячний посадовий оклад, грн.	Оплата за робочий день, грн.	Число днів роботи, t	Витрати на оплату праці, грн.
Розробник	11000	478,26	65	31086,95
Науковий керівник проекту	9000	391,3	22	8608,69
Всього:				39696,64

Додаткова заробітна плата  $Z_d$  всіх розробників та робітників, які брали участь у виконанні даної роботи, розраховується як (10...12)% від суми основної заробітної плати всіх розробників та робітників, тобто:

$$Z_{\text{д}} = (0,1 \dots 0,12) \cdot Z_{\text{о}} \quad (4.2)$$

Додаткова заробітна плата усіх робітників та розробників:

$$Z_{\text{д}} = 0,12 \cdot 39696,64 = 4763,59 \text{ (грн)}$$

Нарахування на заробітну плату  $H_{\text{зп}}$  розробників та робітників, які брали участь у виконанні даної роботи, розраховуються за формулою:

$$H_{\text{зп}} = (Z_{\text{о}} + Z_{\text{р}} + Z_{\text{д}}) \cdot \frac{\beta}{100}, \quad (4.3)$$

де  $Z_{\text{о}}$  — основна заробітна плата розробників, грн.;

$Z_{\text{д}}$  — додаткова заробітна плата розробників, грн.;

$\beta$  — ставка єдиного внеску на загальнообов'язкове державне соціальне страхування, %.

З 01.01.2018 року ставка єдиного внеску на загальнообов'язкове державне соціальне страхування встановлено 22%.

Нарахування на заробітну плату:

$$H_{\text{зп}} = (39696,64 + 4763,59) \cdot \frac{22}{100} = 9781,25 \text{ (грн)}$$

мортизаційні відрахування обладнання можна розрахувати за формулою:

$$A = \frac{Ц \cdot H_{\text{а}}}{100} \cdot \frac{T}{12} \text{ грн.}, \quad (4.4)$$

де  $Ц$  — загальна балансова вартість обладнання;

$H_{\text{а}}$  — річна норма амортизаційних відрахувань. Для нашого випадку можна прийняти, що  $H_{\text{а}} = 10-25\%$ ;

$T$  — термін використання обладнання,  $T = 2-4$  міс.

5) Витрати на матеріали  $M$ , що були використанні під час виконання роботи, розраховуються по кожному виду матеріалів за формулою:

$$M = \sum_1^n H_i \cdot Ц_i \cdot K_i - \sum_1^n B_i \cdot Ц_{\text{в}} \text{ грн.}, \quad (4.5)$$

де  $H_i$  — витрати матеріалу  $i$ -го найменування;

$Ц_i$  — вартість матеріалу  $i$ -го найменування, грн./кг.;

$K_i$  — коефіцієнт транспортних витрат,  $K_i = (1,1 \dots 1,15)$ ;

$V_i$  — маса відходів матеріалу  $i$ -го найменування, кг;

$C_B$  — ціна відходів матеріалу  $i$ -го найменування, грн/кг.

Таблиця 4.4 — Результати розрахунків амортизаційних відрахувань

Найменування обладнання, приміщень тощо	Балансова вартість, грн.	Норма амортизації, %	Термін використання, міс.	Величина амортизаційних відрахувань, грн.
1.Ноутбук	10000	15	3	375
2. Приміщення лабораторії	58000	15	3	2175
3. Мікропроцесорна платформа Raspberry Pi.	350	25	4	29
4. Програмне забезпечення Visual Studio IDE.	4400	20	2	147,4
5. Датчики для GPS моніторингу разом з датчиком рівня палива.	9819	10	3	245,48
Всього				A = 2800

Таблиця 4.5 — Вартість матеріалів, що були використані для розробки

ПЗ

Найменування матеріалу	Одиниці виміру	Ціна, грн.	Витрачено	Вартість витрачених матеріалів, грн.
Припій	Шт.	20	1	20
Флюс	Шт.	30	1	30
Провідники	Шт.	50	1	50
Всього				100

Загальна вартість витрат матеріалів становить:

$$M = 100 \cdot 1,1 = 110 \text{ (грн)}$$

Використані для розробки нового технічного рішення комплектуючі наведенно в табл. 4.6.



Витрати на комплектуючі вироби ( $B_K$ ) розраховуються згідно з їх номенклатурою за формулою:

$$B_K = \sum_i H_{Ki} \cdot C_{Ki} \cdot K, \quad (5.8)$$

де  $H_{Ki}$  — кількість комплектуючих  $i$ -го найменування;

$C_{Ki}$  — ціна комплектуючих  $i$ -го найменування.

Таблиця 4.6 — Комплектуючі, що використовуються для розробки нового технічного рішення

Найменування комплектуючих	Кількість, шт.	Ціна за одиницю, грн.	Сума, грн.
1.Мікропроцесорна платформа Raspberry Pi. (350 грн)	1	350	350
2.Відеокамера (1092 грн).	1	1092	1092
3.GPS датчик (650 грн).	1	650	650
4. Датчик рівня пального (399 грн).	1	399	399
5.Електронні компоненти (200 грн).	1	200	200
Всього, грн.			2691

Витрати на силову електроенергію  $B_e$ , розраховується за формулою:

$$B_e = B \cdot \Pi \cdot \Phi \cdot K_{\Pi} \text{ грн.}, \quad (4.6)$$

де  $B$  — вартість 1кВт-год. електроенергії,  $B = 2,44$  грн/кВт;

$\Pi$  — установлена потужність обладнання,  $\Pi = 0,06$  кВт;

$\Phi$  — фактична кількість годин роботи обладнання,  $\Phi = 455$  год;

$K_{\Pi}$  — коефіцієнт використання потужності,  $K_{\Pi} = 0,75$ .

Витрати на силову енергію становлять:

$$B_e = 2,44 \cdot 0,06 \cdot 455 \cdot 0,75 = 49,95 \text{ (грн)}$$

Інші витрати  $I_v$  можна прийняти як (100...300%) від суми основної заробітної плати розробників, тобто:

$$V_{in} = (1...3) \cdot Z_o \quad (4.7)$$

Інші витрати становлять:

$$V_{in} = 2 \cdot 39696,64 = 79393,28(\text{грн})$$

Сума усіх витрат становить:

$$\begin{aligned} V_{in} &= 39696,64 + 4763,59 + 9781,25 + 2800 + 49,5 + 2691 + 110 \\ &+ 79393,28 = 134516,53(\text{грн}) \end{aligned}$$

Розрахунок загальних витрат на виконання роботи. Загальна вартість роботи визначається за  $V_{заг}$  формулою:

$$V_{заг} = \frac{V}{\alpha} \quad (4.8)$$

де  $\alpha$  — частка витрат, які безпосередньо здійснює виконавець даного етапу роботи, у відн. одиницях = 1.

$$V_{заг} = \frac{134516,53}{1} = 134516,53 \text{ (грн)}$$

Прогнозування загальних витрат на виконання та впровадження результатів. Прогнозування загальних витрат ЗВ на виконання та впровадження результатів здійснюється за формулою:

$$ЗВ = \frac{В_{\text{заг}}}{\beta}, \quad (4.9)$$

де  $\beta$  — коефіцієнт, який характеризує етап (стадію) виконання даної роботи, на стадії розробки промислового зразка,  $\beta \approx 0,7$ .

$$ЗВ = \frac{134516,53}{0,7} = 192166,47 \text{ (грн)}$$

#### 4.3 Прогнозування комерційних ефектів від реалізації результатів розробки

Виконання наукової роботи та впровадження її результатів буде здійснюватися протягом одного року. Позитивні результати від впровадження розробки очікуються протягом трьох років від впровадження розробки. Одним із основних позитивних результатів є зростання величини прибутку. Зростання чистого прибутку забезпечить підприємству (організації) надходження додаткових коштів, які дозволять покращити фінансові результати діяльності.

Збільшення чистого прибутку підприємства  $\Delta\Pi_i$  для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, розраховується за формулою:

$$\Delta\Pi_i = \sum_1^n (\Delta\Pi_{\text{я}} \cdot N + \Pi_{\text{я}} \cdot \Delta N)_i, \quad (4.10)$$

де  $\Delta\Pi_{\text{я}}$  — покращення основного якісного показника від впровадження результатів розробки у даному році;

$N$  — основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

$\Delta N$  — покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$P_{\text{я}}$  — основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

$n$  — кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

В результаті впровадження результатів наукової розробки витрати матеріалів на розробку алгоритму зменшаться на 250 грн (що автоматично спричинить збільшення чистого прибутку підприємства на 250 грн), а кількість користувачів збільшиться: протягом першого року — на 200 користувачів, протягом другого року — на 150 користувачів, протягом третього року — на 100 користувачів.

Реалізація продукції до впровадження результатів наукової розробки складала 600 шт., а прибуток, що його отримувало підприємство (організація) на одиницю продукції до впровадження результатів наукової розробки — 400 грн.

Спрогнозуємо збільшення чистого прибутку від впровадження результатів наукової розробки у кожному році відносно базового.

Збільшення чистого продукту  $\Delta P_1$  протягом першого року складатиме:

$$\Delta P_1 = 250 \cdot 600 + (400 + 250) \cdot 200 = 280000 \text{ (грн)}$$

Протягом другого року:

$$\Delta P_2 = 250 \cdot 600 + (400 + 250) \cdot (200 + 150) = 377500 \text{ (грн)}$$

Протягом третього року:

$$\Delta P_3 = 250 \cdot 600 + (400 + 250) \cdot (200 + 150 + 100) = 442500 \text{ (грн)}$$

4.4 Розрахунок ефективності вкладених інвестицій та періоду їх окупності

В даному підрозділі необхідно кількісно спрогнозувати, яку вигоду, зиск можна отримати у майбутньому від впровадження результатів виконаної роботи.

Розрахунок ефективності вкладених інвестицій передбачає проведення обрахунків наведених нижче.

Розрахуємо теперішню вартість інвестицій  $PV$ , що вкладаються в наукову розробку. Такою вартістю ми можемо вважати прогнозовану величину загальних витрат  $ZB$  на виконання та впровадження результатів НДДКР, розраховану раніше, тобто будемо вважати, що  $ZB = PV = 192166,47$  (грн).

Розрахуємо очікуване збільшення прибутку  $\Delta\Pi_i$ , що його отримає підприємство (організація) від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження. Таке збільшення прибутку також було розраховане нами раніше та становить:

$$\Delta\Pi_1 = 280000 \text{ (грн)}, \quad \Delta\Pi_2 = 377500 \text{ (грн)}, \quad \Delta\Pi_3 = 442500 \text{ (грн)}.$$

Для спрощення подальших розрахунків необхідно побудувати вісь часу, на яку наносять всі платежі (інвестиції та прибутки), що мають місце під час виконання науково-дослідної роботи та впровадження її результатів.

Якщо загальні витрати  $ZB$  на виконання та впровадження результатів НДДКР (або теперішня вартість інвестицій  $PV$ ) дорівнюють 192166,47 грн., а результати вкладених у наукову розробку інвестицій почнуть виявлятися вже вкінці другого року впровадження. То ці результати виявляться у тому, що у першому році підприємство отримає збільшення чистого прибутку на 280000 грн. відносно базового року, у другому році — збільшення чистого прибутку на 377500 грн. (відносно базового року), у третьому році — збільшення чистого прибутку на 442500 грн. (відносно базового року).

Тоді рисунок, що характеризує рух платежів (інвестицій та додаткових прибутків) буде мати вигляд, наведений на рис. 4.1.



Рисунок 4.1 — Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

Розрахуємо абсолютну ефективність вкладених інвестицій  $E_{\text{абс}}$ . Якщо  $E_{\text{абс}} \leq 0$  то результати від проведення наукових досліджень та їх впровадження буде збитковим і вкладати гроші в проведення цих досліджень ніхто не буде.

Для цього показника скористаємося формулою:

$$E_{\text{абс}} = (\text{ПП} - \text{PV}), \quad (4.11)$$

де ПП — приведена вартість всіх чистих прибутків, що їх отримає підприємство (організація) від реалізації результатів наукової розробки, грн.;

PV — теперішня вартість інвестицій  $PV = ZB = 192166,47$  грн.

У свою чергу, приведена вартість всіх чистих прибутків ПП розраховується за формулою:

$$\text{ПП} = \sum_1^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (4.12)$$

де  $\Delta\Pi_i$  — збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

$t$  — період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

$\tau$  — ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

$t$  — період часу (в роках) від моменту отримання чистого прибутку до точки „0”.

$$\text{ПП} = \frac{280000}{(1+0,1)^2} + \frac{377500}{(1+0,1)^3} + \frac{442500}{(1+0,1)^4} = 231404,95 + 283621,33 + 302233,45 \\ 817259,73 \text{ (грн)}$$

$$E_{\text{абс}} = (817259,73 - 192166,47) = 625093,26 \text{ (грн)}$$

Оскільки  $E_{\text{абс}} > 0$ , то вкладання коштів на виконання та впровадження результатів НДДКР може бути доцільним.

Розрахуємо відносну (щорічну) ефективність вкладених в наукову розробку інвестицій  $E_{\text{в}}$ . Для цього використаємо формулу:

$$E_{\text{в}} = \tau_{\text{ж}} \sqrt[1 + \frac{E_{\text{абс}}}{PV}]{1} - 1, \quad (4.13)$$

де  $E_{\text{абс}}$  — абсолютна ефективність вкладених інвестицій, грн;

$PV$  — теперішня вартість інвестицій  $PV = 3B$ , грн;

$\tau_{\text{ж}}$  — життєвий цикл наукової розробки, роки.

Далі, розрахована величина  $E_{\text{в}}$  порівнюється з мінімальною (бар'єрною) ставкою дисконтування  $\tau_{\text{мін}}$ , яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування  $\tau_{\text{мін}}$  визначається за формулою:

$$\tau = d + f \quad (4.14)$$

де  $d$  — середньозважена ставка за депозитними операціями в комерційних банках; в 2020 році в Україні  $d = (0,14...0,2)$ ;

$f$  — показник, що характеризує ризикованість вкладень; зазвичай, величина  $f = (0,05...0,1)$ , але може бути і значно більше.

Якщо величина  $E_{\text{в}} > \tau_{\text{мін}}$ , то інвестор може бути зацікавлений у фінансуванні даної наукової розробки. В іншому випадку фінансування наукової розробки здійснюватися не буде.

Спочатку спрогнозуємо величину  $\tau_{\text{мін}}$ . Припустимо, що за даних умов  $\tau_{\text{мін}} = 0,18 + 0,07 = 0,25$ .

Тоді відносна (щорічна) ефективність вкладних інвестицій в проведення наукових досліджень та впровадження їх результатів складе:

$$E_b = \sqrt[4]{1 + \frac{625093,26}{192166,47}} - 1 = \sqrt[4]{4,25} - 1 = 1,43 - 1 = 0,43 \text{ або } 43\%$$

Оскільки  $E_b = 43\% > \tau_{\text{мін}} = 25\%$ , то інвестор буде зацікавлений вкладати гроші в дану наукову розробку.

Розраховують термін окупності вкладених у реалізацію наукового проекту інвестицій. Термін окупності вкладених у реалізацію наукового проекту інвестицій  $T_{\text{ок}}$  можна розрахувати за формулою:

$$T_{\text{ок}} = \frac{1}{E_b} \quad (4.15)$$

Якщо  $T_{\text{ок}} < 3 \dots 5$ -ти років, то фінансування даної наукової розробки в принципі є доцільним. В інших випадках потрібні додаткові розрахунки та обґрунтування.

Термін окупності становить:

$$T_{\text{ок}} = \frac{1}{0,43} = 2,32 \quad (4.16)$$

$T_{\text{ок}} < 5$  років, що свідчить про доцільність фінансування даної наукової розробки.

Рівень комерційного потенціалу нової розробки є вище середнього. Показники ефективності показують, що даний метод є доцільним і буде цікавий для інвестора. Термін окупності розробленого проекту менше 5-ти років, що підтверджує доцільність вкладання коштів в дану розробку.



## ВИСНОВКИ

Після аналізу актуальності проблеми, було вирішено розробити інформаційну технологію GPS моніторингу та управління технічним забезпеченням агропромислового підприємства.

Під час аналізу вирішення даного питання шляхом розгляду аналогів, було визначено ряд задач, які можна вирішити з використанням програмно-апаратного засобу який розроблюється, що демонструє доцільність розробки.

Для подальшої реалізації програмно-апаратного засобу, було розроблено структурну схему, алгоритм роботи системи та створено апаратну реалізацію системи та програмну реалізацію системи. Було впроваджено новітній метод виміру рівня палива, який значно покращує точність вимірів та є універсальним у використанні.

Мікропроцесорна платформа, що була обрана за основу для програмно-апаратного засобу, дозволяє легко масштабувати систему, і є дуже простою для використання.

Також було реалізовано програмну частину, яка дозволяє збирати данні з датчиків і передавати на базу даних через мережу Інтернет. Поєднання саме таких компонентів дозволило досягти високої швидкодії, масштабованості, використання на різній техніці та стабільної роботи самої системи.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Agriculture Overview: Development news, research, data [Електронний ресурс] — <https://www.worldbank.org/en/topic/agriculture/overview#1>
2. AGRICULTURE IN UKRAINE: OVERVIEW [Електронний ресурс] — <https://dlf.ua/en/agriculture-in-ukraine-overview/>
3. Innovative development of the agro-industrial complex on the basis of disruptive technologies [Електронний ресурс] — [https://www.researchgate.net/publication/311246946\\_Innovative\\_development\\_of\\_the\\_agro-industrial\\_complex\\_on\\_the\\_basis\\_of\\_disruptive\\_tech](https://www.researchgate.net/publication/311246946_Innovative_development_of_the_agro-industrial_complex_on_the_basis_of_disruptive_tech)
4. How Digital Technology Is Changing Farming In The World [Електронний ресурс] — <https://www.fieldbee.com/blog/how-digital-technology-is-changing-farming-in-the-world/>
5. GPS, Система глобального позиціонування [Електронний ресурс] — <https://uk.wikipedia.org/wiki/GPS>
6. GPS – трекер [Електронний ресурс] — <https://pod-parusom.livejournal.com/11702.html>.
7. Smart Farming in 2020: How IoT sensors are creating a more efficient precision agriculture industry [Електронний ресурс] — <https://www.businessinsider.com/smart-farming-iot-agriculture>

**ДОДАТОК А**

Міністерство освіти і науки України  
Вінницький національний технічний університет  
Інститут інформаційних технологій та комп'ютерної інженерії  
Кафедра обчислювальної техніки

**ЗАТВЕРДЖУЮ**Завідувач кафедри ОТ

проф., д.т.н.. Азаров О.Д. \_\_\_\_\_.

(наук. ст., вч. зв., ініц. та прізви.) (підпис)

" \_\_\_\_\_ " \_\_\_\_\_ 2021 р.

**ТЕХНІЧНЕ ЗАВДАННЯ**

до магістерської кваліфікаційної роботи

**Метод та апаратно-програмні засоби визначення напрямків зву-  
кових сигналів**

08-23.КМКР.029.00.000 ПЗ

Науковий керівник: професор д.т.н.

\_\_\_\_\_ Азаров О.Д.

(підпис)

« \_\_\_ » \_\_\_\_\_ 2021 р.

Виконав: студент групи 1КІ-20м

\_\_\_\_\_ Паламарчук М. О.

(підпис)

« \_\_\_ » \_\_\_\_\_ 2021 р.

Вінниця 2021 р.

1 Підстава для виконання магістерської кваліфікаційної роботи (МКР):

— системи GPS моніторингу та методи виміру рівня палива є важливі для завдань контролю та збору статистичних даних в агропромисловій сфері;

— наказ про затвердження теми магістерської кваліфікаційної роботи.

2 Мета МКР — це спрощення апаратної реалізації та розширення можливостей застосування системи GPS моніторингу та керування агоропромисловим комплексом, а призначення МКР — визначення підходів до побудови апаратно-програмних засобів системи GPS моніторингу для керування агоропромисловим комплексом.

3 Вихідні дані для виконання МКР:

— призначення — визначення місцезнаходження за допомогою GPS засобів та вимірювання рівня пального запропонованим методом;

— джерело вхідних сигналів — GPS, GSM та модулі датчиків виміру рівня;

— обробка отриманих даних — мікропроцесорна платформа Raspberry Pi;

— режими роботи системи — прийом та передача;

— підтримка взаємодії з мобільним додатком.

4 Вимоги до виконання МКР:

— провести техніко-економічне обґрунтування доцільності розробки;

— провести аналіз сучасних методів та технологій GPS моніторингу в агропромисловій сфері та проаналізувати існуючі альтернативи на ринку;

— визначення підходів до реалізації апаратних та програмних засобів системи GPS моніторингу в агропромисловій сфері;

— оцінити комерційний потенціал розробки

5 Етапи МКР та очікувані результати наведено в таблиці А.1

Таблиця А.1 — Етапи та очікувані результати

№ етапу	Назва етапу	Термін виконання		Очікувані результати
		початок	кінець	
1	Огляд і аналіз агропромислової сфери, існуючих методів та технологій побудови систем GPS моніторингу			Аналітичний огляд літературних джерел, задачі досліджень, розділ 1 ПЗ
2	Розробка апаратних засобів системи GPS моніторингу			розділ 2
3	Розробка програмних засобів системи GPS моніторингу			розділ 3
4	Оцінка комерційного потенціалу розробки			розділ 4
5	Оформлення пояснювальної записки, графічного матеріалу і презентації			пояснювальна записка, графічний матеріал і презентація

#### 7 Матеріали, що подаються до захисту МКР

Пояснювальна записка МКР, графічні і ілюстративні матеріали, протокол попереднього захисту МКР на кафедрі, відгук наукового керівника, відгук опонента, протоколи складання державних екзаменів, анотації до МКР українською та іноземною мовами, довідка про відповідність оформлення МКР діючим вимогам.

#### 8 Порядок контролю виконання та захисту МКР

Виконання етапів графічної та розрахункової документації МКР контролюється науковим керівником згідно зі встановленими термінами. Захист МКР відбувається на засіданні Державної екзаменаційної комісії, затвердженою наказом ректора.

## 9 Вимоги до оформлення МКР

Вимоги викладені в ДСТУ 3008-2015 та «Положенні про кваліфікаційні роботи на другому (магістерському) рівні вищої освіти» Вінницького національного технічного університету; наказ ВНТУ №203 від 29.06.2021 р.

**ДОДАТОК Б**

## Лістинг коду

```
import board
import digitalio
import busio
print("Hello blinka!")
# Try to great a Digital input
pin = digitalio.DigitalInOut(board.D4)
print("Digital IO ok!")
# Try to create an I2C device
i2c = busio.I2C(board.SCL, board.SDA)
print("I2C ok!")
# Try to create an SPI device
spi = busio.SPI(board.SCLK, board.MOSI, board.MISO)
print("SPI ok!")
print("done!")

/etc/chatscripts/gprs
ABORT BUSY
ABORT VOICE
ABORT "NO CARRIER"
ABORT "NO DIALTONE"
ABORT "NO DIAL TONE"
ABORT "NO ANSWER"
ABORT "DELAYED"
ABORT "ERROR"
ABORT "+CGATT: 0"
"" AT
TIMEOUT 12
```

```
OK ATH
```

```
OK ATE1
```

```
OK AT+CGDCONT=1,"IP","telia"
```

```
OK ATD*99#
```

```
TIMEOUT 22
```

```
CONNECT ""
```

Також створимо файл для GSM.

```
/etc/ppp/peers/a-gsm
```

```
connect "/usr/sbin/chat -v -f /etc/chatscripts/gprs -T telia"
```

```
/dev/ttyAMA0
```

```
115200
```

```
noipdefault
```

```
usepeerdns
```

```
defaultroute
```

```
persist
```

```
noauth
```

```
nocrtscts
```

```
local
```

```
import RPi.GPIO as GPIO
```

```
import time
```

```
GPIO.setmode(GPIO.BCM)
```

```
TRIG = 23
```

```
ECHO = 24
```

```
GPIO.setup(TRIG,GPIO.OUT)
```

```
GPIO.setup(ECHO,GPIO.IN)
```

```
GPIO.output(TRIG, False)
```



```
print "Waiting For Sensor To Settle"

time.sleep(2)

GPIO.output(TRIG, True)

time.sleep(0.00001)

GPIO.output(TRIG, False)

while GPIO.input(ECHO)==0:

    pulse_start = time.time()

while GPIO.input(ECHO)==1:

    pulse_end = time.time()

pulse_duration = pulse_end - pulse_start

distance = pulse_duration x 17150

distance = round(distance, 2)

print "Distance:",distance,"cm

GPIO.cleanup()

import os

import time

import Adafruit_DHT

DHT_SENSOR = Adafruit_DHT.DHT22

DHT_PIN = 4

try:

    f = open('/home/pi/humidity.csv', 'a+')

    if os.stat('/home/pi/humidity.csv').st_size == 0:
```

```

        f.write('Date,Time,Temperature,Humidity\r\n')

except:

    pass

while True:

    humidity, temperature = Adafruit_DHT.read_retry(DHT_SENSOR,
DHT_PIN)

    if humidity is not None and temperature is not None:

        f.write('{0},{1},{2:0.1f}*C,{3:0.1f}%\r\n'.format(time.strftime('%m/%d/
%y'), time.strftime('%H:%M'), temperature, humidity))

    else:

        print("Failed to retrieve data from humidity sensor")

        time.sleep(30)

# test_connect.py
import paho.mqtt.client as mqtt
# The callback function. It will be triggered when trying to connect to the
MQTT broker
# client is the client instance connected this time
# userdata is users' information, usually empty. If it is needed, you can set
it through user_data_set function.
# flags save the dictionary of broker response flag.
# rc is the response code.
# Generally, we only need to pay attention to whether the response code is
0.

def on_connect(client, userdata, flags, rc):
    if rc == 0:

```

```

        print("Connected success")
    else:
        print(f"Connected fail with code {rc}")
client = mqtt.Client()
client.on_connect = on_connect
client.connect("broker.emqx.io", 1883, 60)
client.loop_forever()

# subscriber.py
import paho.mqtt.client as mqtt
def on_connect(client, userdata, flags, rc):
    print(f"Connected with result code {rc}")
    # subscribe, which need to put into on_connect
    # if reconnect after losing the connection with the broker, it will continue
to subscribe to the raspberry/topic topic
    client.subscribe("raspberry/topic")
# the callback function, it will be triggered when receiving messages
def on_message(client, userdata, msg):
    print(f"{msg.topic} {msg.payload}")
client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message
# set the will message, when the Raspberry Pi is powered off, or the network
is interrupted abnormally, it will send the will message to other clients
    client.will_set('raspberry/status', b'{"status": "Off"}')
# create connection, the three parameters are broker address, broker port
number, and keep-alive time respectively
    client.connect("broker.emqx.io", 1883, 60)
# set the network loop blocking, it will not actively end the program before
calling disconnect() or the program crash

```

```
client.loop_forever()

import paho.mqtt.client as mqtt
import time
def on_connect(client, userdata, flags, rc):
    print(f"Connected with result code {rc}")
client = mqtt.Client()
client.on_connect = on_connect
client.connect("broker.emqx.io", 1883, 60)
# send a message to the raspberry/topic every 1 second, 5 times in a row
for i in range(5):
    # the four parameters are topic, sending content, QoS and whether
retaining the message respectively
    client.publish('raspberry/topic', payload=i, qos=0, retain=False)
    print(f"send {i} to raspberry/topic")
    time.sleep(1)
client.loop_forever()
```

## ДОДАТОК В

## Фото компонентів системи

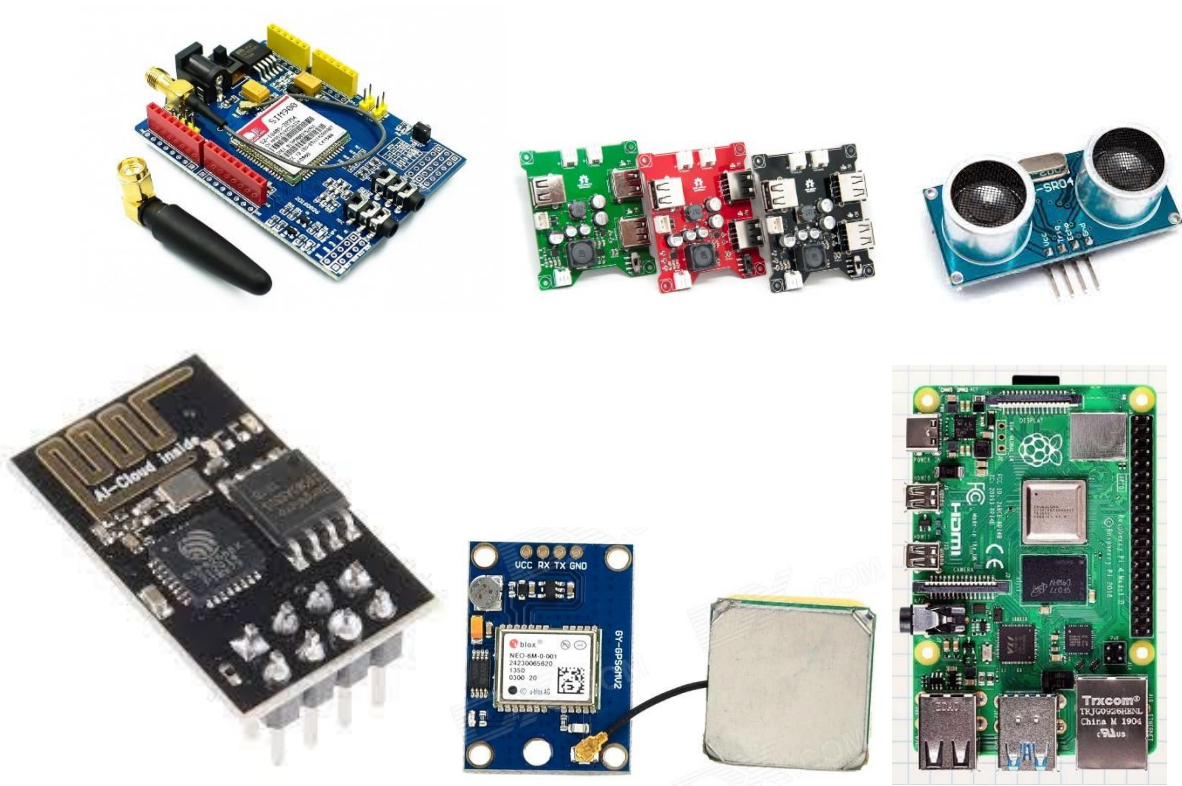


Рисунок В.1 — Зовнішнє зображення компонентів системи

# ДОДАТОК Г

## Схема електрична-принципова платформи Raspberry Pi

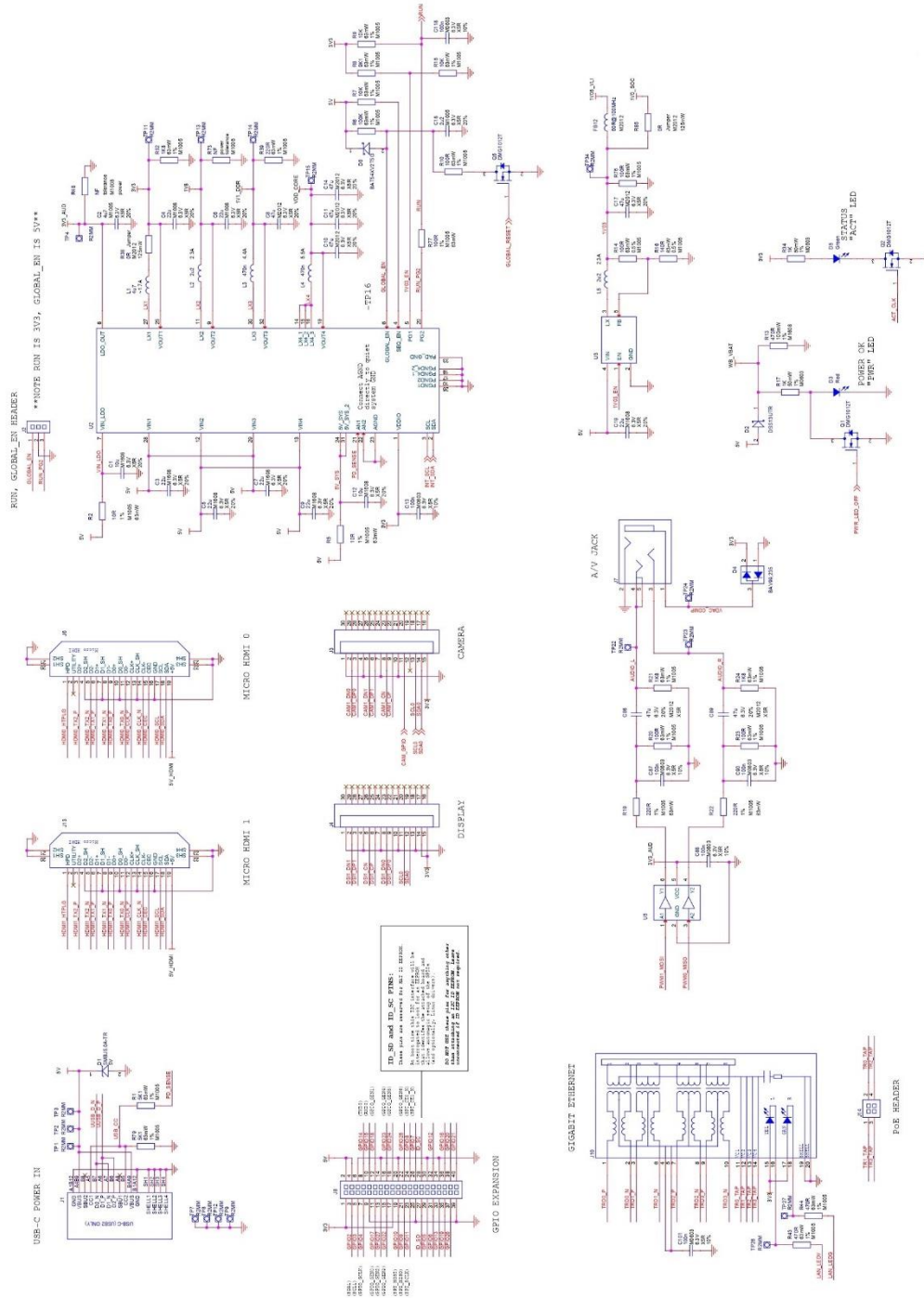


Рисунок Г.1 — Схема електрична принципова платформи Raspberry Pi

## ДОДАТОК Д

### Структурна схема системи

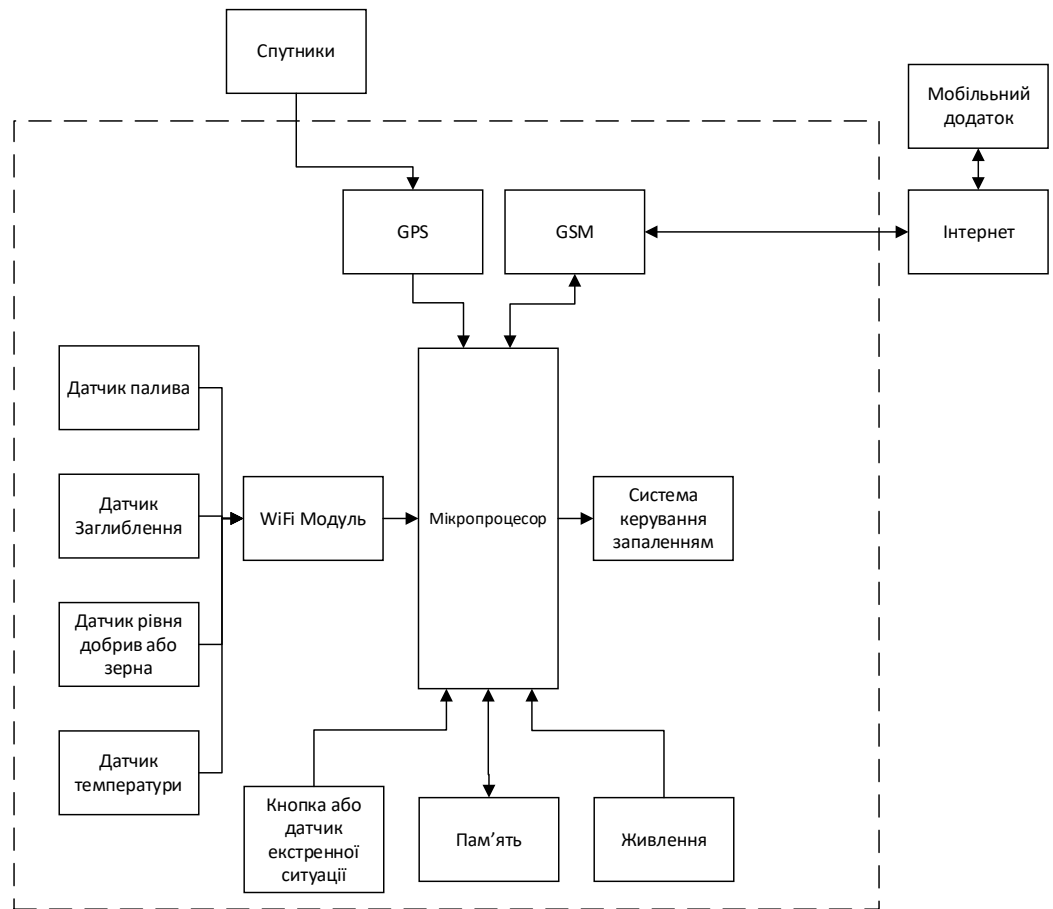


Рисунок Д.1 — Структурна схема системи

## ДОДАТОК И

### Алгоритм роботи системи



Рисунок И.1 — Блок-схема алгоритму відправки даних



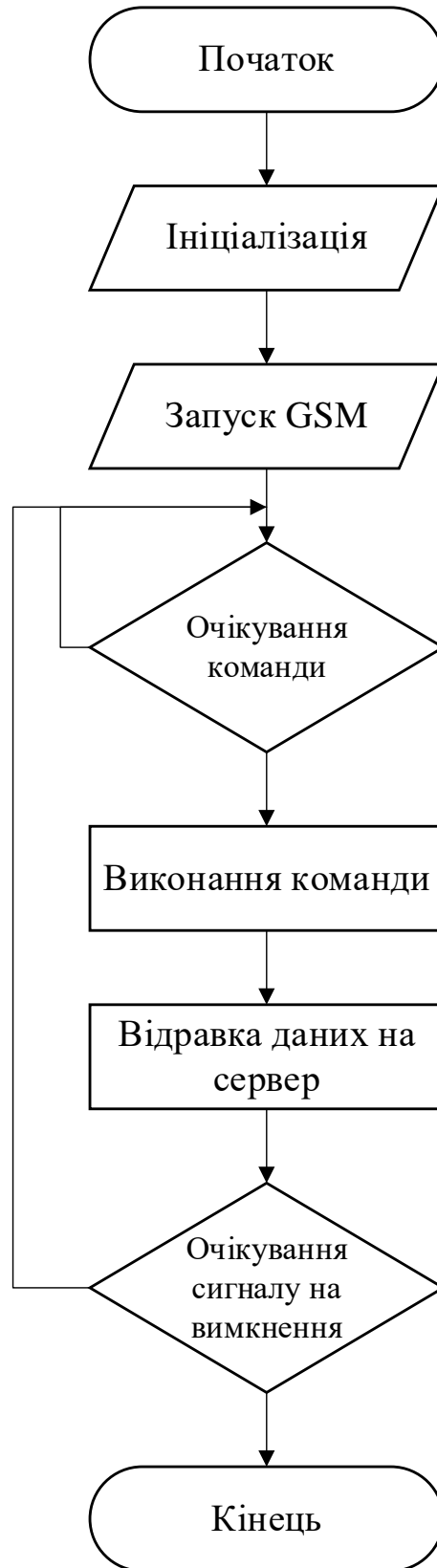


Рисунок И.2 — Блок-схема алгоритму прийому команд

## ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Назва роботи: Інтелектуальна система GPS моніторингу та управління технічним забезпеченням агропромислового підприємства. Частина 1. Апаратна частина

Тип роботи: кваліфікаційна робота

(кваліфікаційна роботи, курсовий проект (робота), реферат, аналітичний огляд, інше (вказати))

Підрозділ кафедра обчислювальної техніки

(кафедра, факультет (інститут), навчальна група)

Науковий керівник Азаров О.Д. професор, завідувач кафедри ОТ

(прізвище, ініціали, посада)

### Показники звіту подібності

Plagiat.pl (StrikePlagiarism)		Unicheck	
КП1			
КП2		Оригінальність	93,4
Тривога/Білі знаки	/	Схожість	7,6

### Аналіз звіту подібності (відмінити подібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності і відсутності самостійності її автора. Робот направити на доопрацювання.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Заявляю, що ознайомлений(-на) з повним звітом подібності, який був згенерований Системою щодо роботи (додається)

Автор \_\_\_\_\_

(підпис)

Паламарчук М. О.

(прізвище, ініціали)

### Опис прийнятого рішення

Ступінь оригінальності роботи відповідає вимогам, що висуваються до МКР

Особа, відповідальна за перевірку \_\_\_\_\_

(підпис)

Захарченко С.М.

(прізвище, ініціали)

Експерт \_\_\_\_\_

(за потреби) (підпис)

(прізвище, ініціали)