

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр  
Освітній рівень


Програмно-технічний пристрій розумного поливу на базі Raspberry Pi  
Назва теми


КвРКІ.2001128.20.01.07 ПЗ  
Шифр

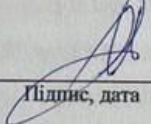
Галузь знань 12 «Інформаційні технології»  
Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»  
Шифр, назва

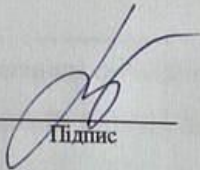
Освітня програма «Комп'ютерна інженерія та програмування»  
Назва

Виконав: студент III курсу, група КІ2с-20-1  Д.І. Борщевський  
Підпис Ініціали, прізвище

Керівник  В.В. Яцків  
Підпис, дата Ініціали, прізвище

Нормоконтролер  С.М. Лисенко  
Підпис, дата Ініціали, прізвище

До захисту допускаю:  
Зав. кафедри комп'ютерної  
інженерії та інформаційних  
систем

 Т.О. Говорушенко  
Підпис Ініціали, прізвище

« 5 » червня 2023 р.

Хмельницький 2023

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ ТА ПРОГРАМУВАННЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорущенко

“ 11 ” 01 2023 р.

ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Борщевський Денис Ігорович

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-технічний пристрій розумного поливу на базі Raspberry Pi

Керівник проекту (роботи) Яцків В.В., професор кафедри КІПС

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 01.03.2023 р. № 5

2. Строк подання студентом проекту (роботи) на кафедру 02.06.2023 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз відомих засобів й рішень автоматизованого поливу

Проектування програмно-технічного пристрою розумного поливу на базі Raspberry Pi

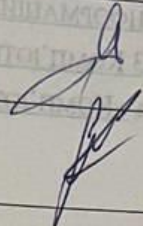

Реалізація програмно-технічного пристрою розумного поливу на базі Raspberry Pi

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Блок-схеми алгоритмів функціонування програмно-технічного пристрою

Схема електрична принципова

Монтажна схема

Нормоконтроль	Підпис, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Антиплагиат	Лисенко С.М., професор кафедри КПС		
	Нічепорук А.О., доцент кафедри КПС		

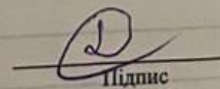
7. Дата видачі завдання « 11 » 01 2023 р.

### КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітки
1	Вибір напрямку дослідження та узгодження тематики кваліфікаційної роботи з керівником	11.01.2023	ВИКОНАНО
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2023	ВИКОНАНО
3	Робота над розділом 1 – Аналіз відомих засобів й рішень автоматизованого поливу	01.03.2023	ВИКОНАНО
4	Робота над розділом 2 – Проектування програмно-технічного пристрою розумного поливу на базі Raspberry Pi	01.04.2023	ВИКОНАНО
5	Робота над розділом 3 – Реалізація програмно-технічного пристрою розумного поливу на базі Raspberry Pi	30.04.2023	ВИКОНАНО
6	Оформлення пояснювальної записки згідно вимог	20.05.2023	ВИКОНАНО
7	Попередній захист ВКР	26.05.2023	ВИКОНАНО
8	Захист ВКР на засіданні ЕК	Червень 2023 року	

Студент

Керівник проекту (роботи)

  
Підпис

  
Підпис

Бошевський Д.І.  
Ініціали, прізвище

Яцків В.В.  
Ініціали, прізвище



## АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-технічний пристрій розумного поливу на базі Raspberry Pi».

Автор роботи: *Борщевський Денис Ігорович.*

Керівник роботи: *Яків Василь Васильович.*

Пояснювальна записка: *55 с., 25 рис., 2 табл., 3 дод., 60 джерел.*

Графічна частина: *3 креслення.*

**РОЗУМНИЙ ПОЛИВ, ВИМІРЮВАННЯ ВОЛОГОСТІ ҐРУНТУ,  
RASPBERRY PI.**

Мета кваліфікаційної роботи: розробка програмно-технічного пристрою розумного поливу на базі Raspberry Pi.

Глобальне потепління впливає на кліматичні зміни, які можуть викликати кризи в різних галузях, включаючи сільське господарство. Однією з головних проблем, пов'язаних з глобальним потеплінням, є зменшення кількості опадів в деяких регіонах та збільшення їх в інших. Це може призвести до нестачі води для зрошення рослин або до затоплення полів.

Щоб боротися з протидіяти та пом'якшувати цій проблемі, з'явилися автоматизовані системи поливу ґрунту. Вони дозволяють ефективніше використовувати воду та зменшити її витрати. Використання таких систем забезпечує рівномірний та ефективний полив ґрунту, що забезпечує зростання рослин та підтримує врожайність. Тому створення автоматизованих систем розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi є актуальним завданням.

Підпис студента



Дата 05.06.2023 р.

## ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ .....	3
ВСТУП.....	4
1 АНАЛІЗ ВІДОМИХ ЗАСОБІВ Й РІШЕНЬ АВТОМАТИЗОВАНОГО ПОЛИВУ	6
1.1 Прилади для вимірювання вологості ґрунту .....	6
1.2 Відомі підходи до створення систем автоматизованого поливу .....	9
1.3 Постановка задачі .....	16
2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ПРИСТРОЮ РОЗУМНОГО ПОЛИВУ НА БАЗІ RASPBERRY PI.....	17
2.1 Вимоги до програмно-технічного пристрою розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi.....	17
2.2 Структура програмно-технічного пристрою розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi.....	18
2.3 Аналіз обраних рішень.....	20
2.3.1 Аналіз обраних апаратних рішень.....	21
2.3.2 Аналіз обраних програмних рішень .....	33
2.4 Висновки до розділу 2.....	33
3 РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ПРИСТРОЮ РОЗУМНОГО ПОЛИВУ НА БАЗІ RASPBERRY PI.....	34
3.1 Підготовка та встановлення необхідних бібліотек для програмної реалізації пристрою.....	34
3.2 Монтажна схема програмно-технічного пристрою розумного поливу на базі Raspberry Pi .....	37
3.3 Схема електрична принципова програмно-технічного пристрою розумного поливу на базі Raspberry Pi.....	41

КвРКІ. 2001128.20.01.07 ПЗ				
Зм.	Арк.	№докум.	Підпис	Дата
Виконав		Борщевський Д.І.		
Перевір.		Яцків В.В.		
Н.контр.		Лисенко С.М.		
Затвер.		Говорущенко Т.О.		25.06
Програмно-технічний пристрій розумного поливу на базі Raspberry Pi				
		Літера	Аркуш	Аркушів
		2	62	62
ХНУ, КІ2с-20-1				

КвРКІ. 2001128.20.01.07 ПЗ				
Зм.	Арк.	№докум.	Підпис	Дата
Виконав		Борщевський Д.І.		
Перевір.		Яцків В.В.		
Н.контр.		Лисенко С.М.		
Затвер.		Говорущенко Т.О.		
Програмно-технічний пристрій розумного поливу на базі Raspberry Pi				
		Літера	Аркуш	Аркушів
		2	62	62
ХНУ, КІ2с-20-1				

3.4 Реалізація функцій програмно-технічного пристрою розумного поливу на базі Raspberry Pi .....	43
3.5 Висновки за розділом 3 .....	57
ВИСНОВКИ .....	58
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ .....	61
ДОДАТОК А Блок-схеми алгоритмів функціонування програмно-технічного пристрою.....	<b>Error! Bookmark not defined.</b>
ДОДАТОК Б Копія креслення «Схема електрична принципова»	<b>Error! Bookmark not defined.</b>
ДОДАТОК В Копія креслення «Монтажна схема».....	70

					КВРКІ. 2001128.20.01.07 ПЗ			
Зм.	Арк.	№докум.	Підпис	Дата				
Виконав		Борщевський Д.І.			Програмно-технічний пристрій розумного поливу на базі Raspberry Pi	Літера	Аркуш	Аркушів
Перевір.		Яцків В.В.					3	62
Н.контр.		Лисенко С.М.			ХНУ, КІ2с-20-1			
Затвер.		Говорущенко Т.О.						

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

АЦП – Аналого-цифровий перетворювач

АП – Автоматизований полив

КС – Комп'ютерна система

ПЗ – Програмне забезпечення

GPIO – General Purpose Input/Output

LDR – Light-Dependent Resistor

RPi – Raspberry Pi

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						3
Зм.	Арк.	№докум.	Підпис	Дата		

## ВСТУП

Глобальне потепління впливає на кліматичні зміни, які можуть викликати кризи в різних галузях, включаючи сільське господарство. Однією з головних проблем, пов'язаних з глобальним потеплінням, є зменшення кількості опадів в деяких регіонах та збільшення їх в інших. Це може призвести до нестачі води для зрошення рослин або до затоплення полів.

Щоб боротися з протидіяти та пом'якшувати цій проблемі, з'явилися автоматизовані системи поливу ґрунту. Вони дозволяють ефективніше використовувати воду та зменшити її витрати. Використання таких систем забезпечує рівномірний та ефективний полив ґрунту, що забезпечує зростання рослин та підтримує врожайність.

Однією з головних переваг автоматичних систем поливу ґрунту є економія води. Це досягається завдяки тому, що системи поливу можуть бути налаштовані на автоматичне регулювання часу та кількості води, яка використовується для поливу. Це дає змогу оптимізувати процес поливу, зменшити кількість води, яка витрачається на полив та, як результат, зменшити витрати на воду.

Також автоматичні системи поливу дозволяють зберегти час та зусилля фермерів та господарів приватних ділянок, оскільки системи можуть працювати без необхідності постійного контролю та управління. Це дає змогу використовувати час та ресурси більш ефективно, зосереджуючись на інших аспектах діяльності.

Нарешті, автоматичні системи поливу можуть бути використані для покращення естетичного вигляду ландшафту, наприклад, для поливу газонів та інших рослинних насаджень, що вимагають регулярного поливу. Це допомагає підтримувати здоров'я та красу рослин, що в свою чергу відображається на загальному вигляді ділянки.

На сьогоднішній день розвиток технологій дозволяє реалізовувати складні системи із використанням мікроконтролерів та мікропроцесорів. Одним

					КВРКІ. 2001128.20.01.07 ПЗ	Арк. 4
Зм.	Арк.	№докум.	Підпис	Дата		

із таких рішень є використання в якості основи одноплатної комп'ютерної системи Raspberry Pi. Таке рішення дозволить забезпечити не тільки реалізацію складних алгоритмів планування та контролю за вологістю ґрунту, а й забезпечить зв'язок програмно-технічного пристрою із Інтернетом, що дозволить тим самим здійснювати моніторинг та управління цим пристроєм з будь-якої точки світу.

Таким чином, задача створення автоматизованих систем розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi є актуальним завданням.

Метою роботи є розробка програмно-технічного пристрою розумного поливу на базі Raspberry Pi.

Об'єктом дослідження є процеси моніторингу вологості та автоматичного поливу ґрунту.

Предметом дослідження є програмно-технічний пристрій розумного поливу бази Raspberry Pi.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						5
Зм.	Арк.	№докум.	Підпис	Дата		

# 1 АНАЛІЗ ВІДОМИХ ЗАСОБІВ Й РІШЕНЬ АВТОМАТИЗОВАНОГО ПОЛИВУ

## 1.1 Прилади для вимірювання вологості ґрунту

Визначення необхідності поливу рослин без приладів є складним, тому для забезпечення оптимального водного режиму рослин доцільно використовувати вимірювальні прилади для визначення вмісту вологи у ґрунті.

Контроль вологості ґрунту є важливим моментом у визначенні необхідності поливу. Для правильного вибору контрольного приладу, який найбільше підходить для застосування у конкретних умовах господарства, доцільно мати уяву про їх принципи дії та особливості застосування. Слід не забувати, що водний режим рослин залежить не лише від вологості ґрунту, але й від інших факторів, зокрема, інших характеристик самого ґрунту, температури повітря, рівня сонячної радіації, фізіологічного стану самих рослин і т.п. Самі прилади потребують правильного встановлення для отримання надійних даних.

Найпростіший і один з найбільш точних методів є ваговий, при якому вміст вологи у зразку ґрунту визначається за різницею ваги початкового зразка та того ж зразка після того, як під дією нагрівання зі зразка буде видалено вологу. Класичний ваговий метод полягає у сушінні проби ґрунту в термостаті при 105 градусах до постійного значення маси протягом 8 годин. Прискорений ваговий метод сушіння йде з використання спиртового випалювання ґрунту. При цьому проба ґрунту змочується спиртом і обпалюють при дефіциті кисню в спеціальних бюксах. Органіка з ґрунту при згорянні спирту практично не вигоряє, а про вміст вологи судять за різницею мас проби до і після випалювання. Основними недоліками цього методу є неможливість проведення аналізу на місці, а також необхідність окремого відбору проб.

Метод визначення вологості ґрунту, який використовує тензіометри, базується на здатності ґрунту поглинати вологу з навколишнього середовища до повного насичення. Сам тензіометр складається з замкнутої посудини у вигляді

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		6

трубки з водою, яка пов'язана з ємністю, де знаходиться проба ґрунту. Мембрана, розташована на одній зі стінок приладу, відхиляється під дією сили всмоктування ґрунту, що індикатором вологості зразка ґрунту. Однак, такі прилади мають недоліки, зокрема потребують заливання рідиною та встановлення з ретельністю, їх міцність низька, а також покази можуть бути зчитані тільки через деякий час. Вартість тензіометрів коливається від 45-80 у.о. за одиницю і залежить від їх довжини. Є також "сухі" тензіометри, але вони дорожчі.

З іншого боку, рефлектометри використовують метод визначення вологості в залежності від ступеня відображення електромагнітної хвилі вологою поверхнею. Датчик вимірює ступінь відображення електромагнітної хвилі, який змінюється в залежності від ступеня вологості матеріалу, а процесор обчислює показник в залежності від виду вимірюваного матеріалу. Такий прилад можна застосовувати у польових умовах і зчитувати дані практично без затримки. Однак, вартість рефлектометрів досить висока: від 500-4500 у.о. і більше.



Рисунок 1.1 – Рефлектометр

Зм.	Арк.	№докум.	Підпис	Дата

Прилади, які вимірюють зміну електрорушійної сили постійного струму, коли поміщені між двома металевими електродами на ділянці ґрунту, можуть бути відносно дешевими, але менш точними. Залежно від типу приладу, вартість може варіюватися від 5 до 35 доларів за одиницю. Прилади з гіпсовим блоком є найдешевшими, а ті, що базуються на зернистій матриці, коштують трохи більше. Однак, для ефективної роботи таких приладів потрібно використовувати пристрої реєстрації даних, які можуть коштувати від 150 до 600 доларів за кожен прилад реєстрації та включати кілька датчиків. Строк служби гіпсових приладів становить близько 1 року, а зернистих матриць - 5-7 років.

## 1.2 Відомі підходи до створення систем автоматизованого поливу

Зрошення визначається як штучне нанесення води на землю або ґрунт. Процес зрошення можна використовувати для вирощування сільськогосподарських культур у період недостатніх опадів та для підтримки ландшафтів. Автоматична система поливу забезпечує роботу системи, не вимагаючи ручної участі людей. Кожна система поливу, така як крапельна, спринклерна та поверхнева, автоматизується за допомогою електронних приладів і детекторів, таких як комп'ютер, таймери, датчики та інші механічні пристрої.

На сьогоднішній день існують різні системи, що використовують широкий спектр сучасних технологій для вирішення задачі автоматизованого поливу ділянок ґрунту.

Однією із систем є автоматична зрошувальна система за визначенням вмісту вологи в ґрунті (рис. 1.2). Порівняно зі звичайним методом зрошення, автоматизована система зрошення на основі датчиків вологості ґрунту може визначати, коли та скільки зрошувати. Автоматична система зрошення може вмикати та вимикати воду автоматично або дистанційно відповідно до потреб культури, більше не вимагаючи відвідування певної ділянки.

Проект автоматичної системи зрошення по датчику вологості ґрунту призначений для розробки системи зрошення, яка вмикає або вимикає насоси за

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
Зм..	Арк.	№докум.	Підпис	Дата		8

допомогою реле. Основна перевага використання цієї зрошувальної системи полягає у зменшенні втручання людини та забезпеченні належного поливу. Мікроконтролер діє як основний блок всього проекту, а блок живлення використовується для подачі живлення 5 В на всю схему за допомогою трансформатора, схеми мостового випрямляча та регулятора напруги. В якості мікроконтролера може бути використаний наприклад Intel 8051.

Мікроконтролер 8051 запрограмований таким чином, що він отримує вхідний сигнал від чутливого матеріалу, який складається з компаратора, щоб знати різні умови вологи в ґрунті. ОР-АМР, який використовується як компаратор, діє як інтерфейс між чутливим матеріалом і мікроконтролером для передачі умов вологості ґрунту, а саме вологості, сухості тощо.



Рисунок 1.2 – Автоматична зрошувальна система за визначенням вмісту вологи в ґрунті

Зм.	Арк.	№докум.	Підпис	Дата

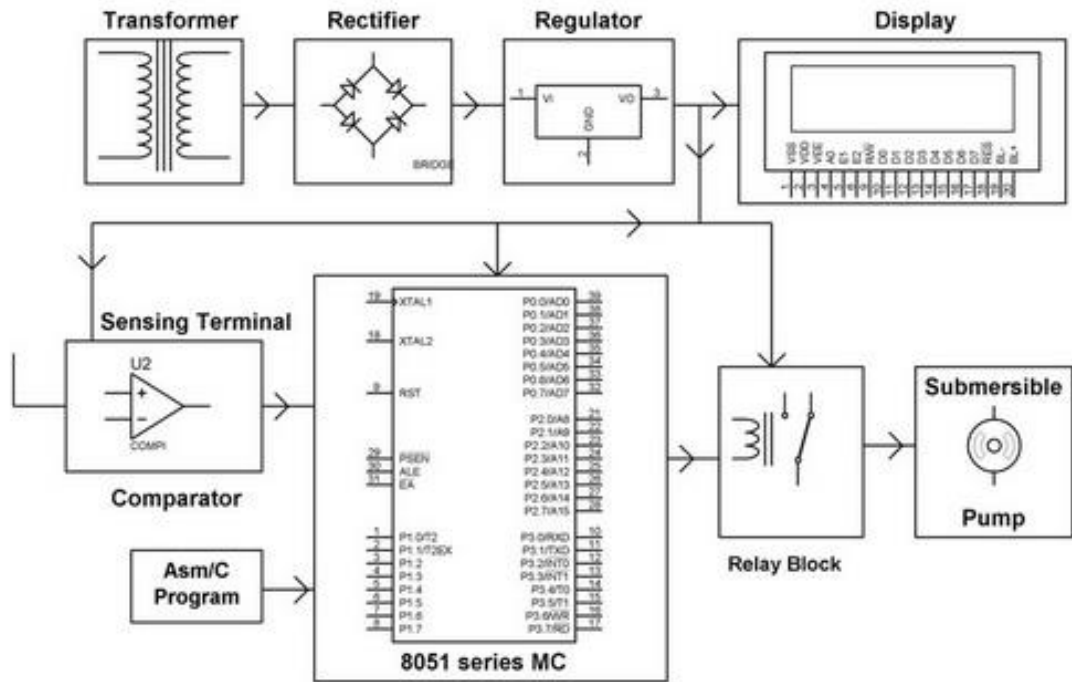


Рисунок 1.3 – Електрична принципова схема зрошувальною системи за визначенням вмісту вологи в ґрунті

Після того, як мікроконтролер отримує дані від чутливого матеріалу, він порівнює дані та генерує вихідні сигнали та активує реле для роботи водяного насоса. Сенсорна установка реалізується за допомогою двох жорстких металевих стрижнів, які вставляються в сільськогосподарське поле на деякій відстані. Необхідні з'єднання цих металевих стрижнів з'єднані з блоком керування для керування роботою насоса відповідно до вологості ґрунту.

Цю автоматичну систему зрошення можна додатково вдосконалити за допомогою передової технології, яка споживає сонячну енергію від сонячних панелей.

На рис. 1.3 наведено систему для роботи якої необхідне живлення від мережі. Як розширення вищезгаданої системи, можна використати сонячні панелі для живлення схеми. У сфері сільського господарства правильне використання методу автоматичного зрошення є дуже важливим через деякі недоліки реального світу, такі як дефіцит води в резервуарах землі та дефіцит опадів. Рівень води (ґрунтових вод) знижується внаслідок постійного

вилучення води з землі, що поступово призводить до дефіциту води в сільськогосподарських зонах, повільно перетворюючи їх на безплідні землі.

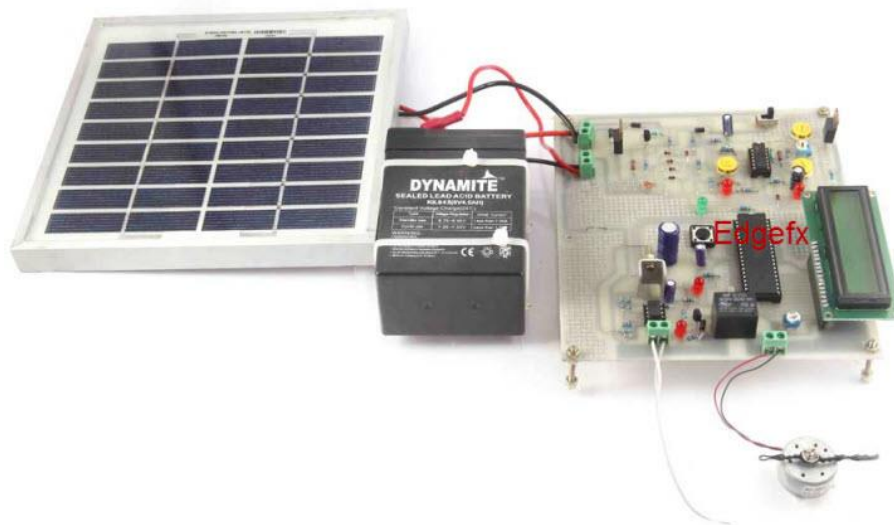


Рисунок 1.4 – Автоматична зрошувальна система за визначенням вмісту вологи в ґрунті із живленням від сонячних панелей

Окремим аспектом в цій зрошувальній системі є використання сонячних панелей для отримання сонячної енергії. Сонячні панелі збирають сонячне випромінювання та перетворюють його на електричну енергію. Ця енергія потім використовується для живлення зрошувального насоса, що дозволяє ефективно здійснювати зрошення рослин. Крім того, система включає датчики вологості, які визначають рівень вологості ґрунту. Вони працюють за допомогою IC OP-AMP, що використовується як компаратори для аналізу вимірюваних значень. Додатково, схема контролера заряду використовується для заряджання фотоелементів, забезпечуючи постачання стабільної сонячної енергії усій системі зрошування. Таке поєднання сонячних панелей з контролером заряду дозволяє забезпечити ефективну та сталу роботу зрошувальної системи за допомогою енергії з відновлювального джерела. Електричну принципову схему зрошувальною системи за визначенням вмісту вологи в ґрунті із живленням від сонячних панелей наведено на рис. 1.5

Зм.	Арк.	№докум.	Підпис	Дата

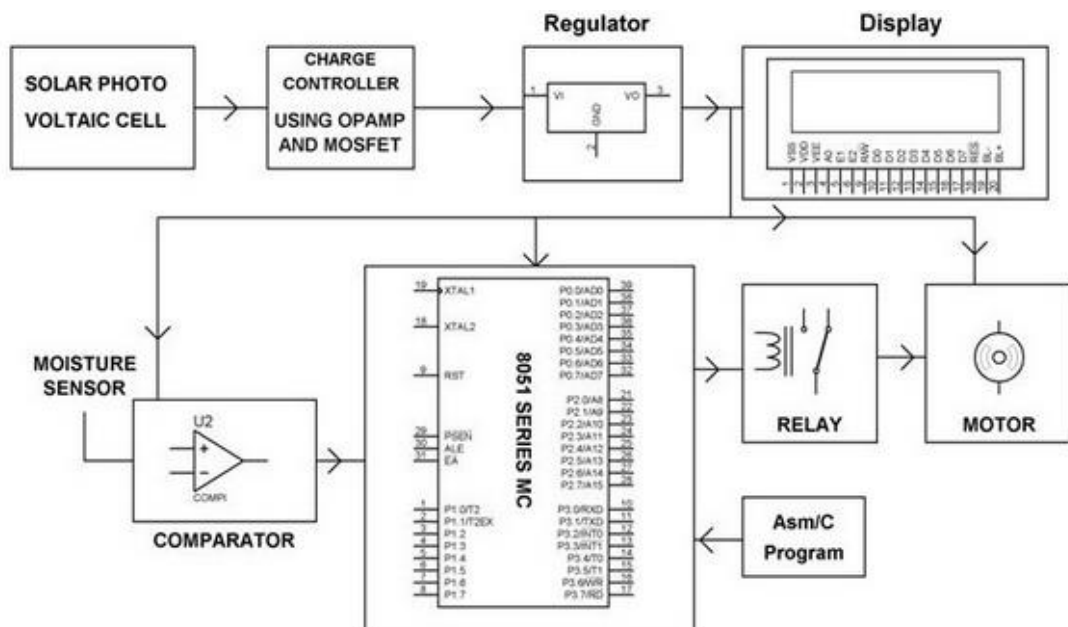


Рисунок 1.5 – Електрична принципова схема зрошувальною системи за визначенням вмісту вологи в ґрунті із живленням від сонячних панелей

Сигнали, які надсилаються від датчиків до мікроконтролера через вихід компаратора, працюють під керуванням програмного забезпечення, яке зберігається в ПЗУ мікроконтролера. На РК-дисплеї відображається стан насоса (увімкнено або вимкнено), підключеного до мікроконтролера. Цю автоматичну систему зрошення можна додатково вдосконалити за допомогою технології GSM, щоб отримати контроль над роботою перемикання двигуна.

Завдання зрошення полів стає досить складним для фермерів через недостатню регулярність у їхній роботі та недбалість з їхнього боку, оскільки іноді вони вмикають двигун, а потім забувають вимкнути, що може призвести до втрати води. Так само забувають навіть увімкнути систему поливу, що знову ж таки призводить до шкоди посівам. Щоб подолати цю проблему, одним із рішень є використання технології GSM.

Система автоматичного поливу на базі GSM – це система, у якій отримується статус поточної операції, що виконується на сільськогосподарських полях через SMS за допомогою модему GSM. Є можливість додати також можемо додати інші системи, такі як РК-дисплеї, веб-камери та інші розумні керовані пристрої.

Зм..	Арк.	№докум.	Підпис	Дата

Як і у попередніх схемах датчик вимірювання вологості ґрунту з'єднаний із мікроконтролером (рис. 1.6). Сигнали вхідних даних від датчика вологи надсилаються на мікроконтролер, і на основі цього він активує двигун постійного струму та вмикає двигун за допомогою драйвера двигуна. Після зволоження ґрунту двигун автоматично вимикається. Статус поливу сільськогосподарських полів можна дізнатися з індикації світлодіода (LED) або через повідомлення, надіслане на GSM-модем, що розміщується на полі. Одночасно можна відправляти повідомлення через мобільний комплект через модем GSM. Таким чином, двигуном поливу можна керувати за допомогою мобільного та GSM модему.

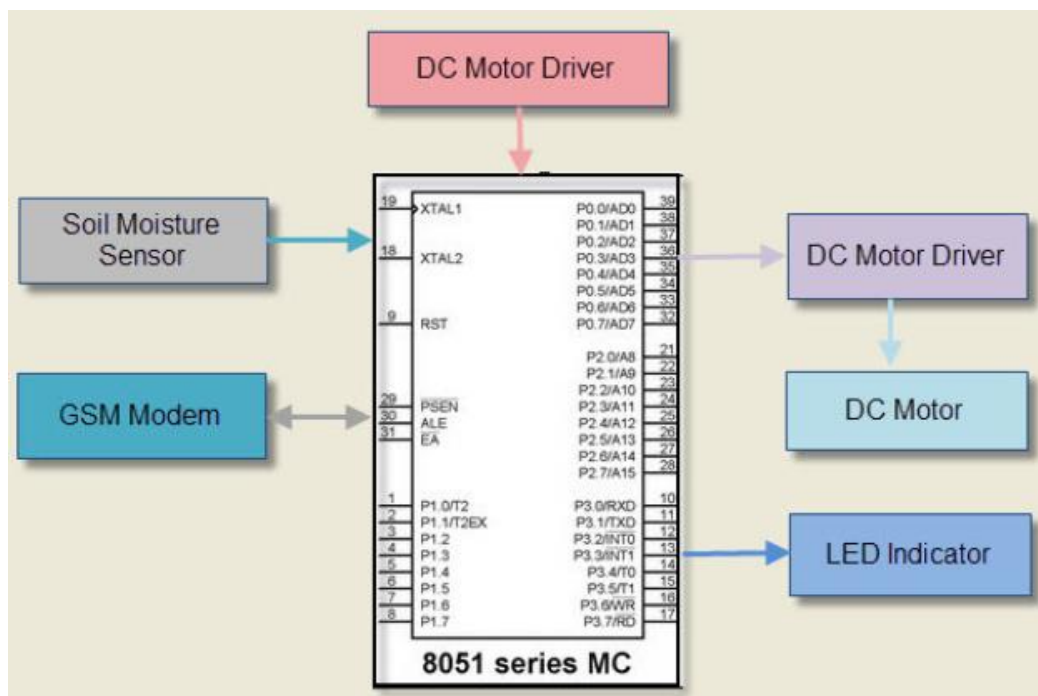


Рисунок 1.6 – Автоматична зрошувальна система за визначенням вмісту вологи в ґрунті та використанням GSM технології

Іншим прикладом реалізації автоматизованої системи поливу є використання технологій IoT, зокрема LoRa [2]. Схематична конструкція системи автоматичного зрошення IoT показана на рис. 1.7.

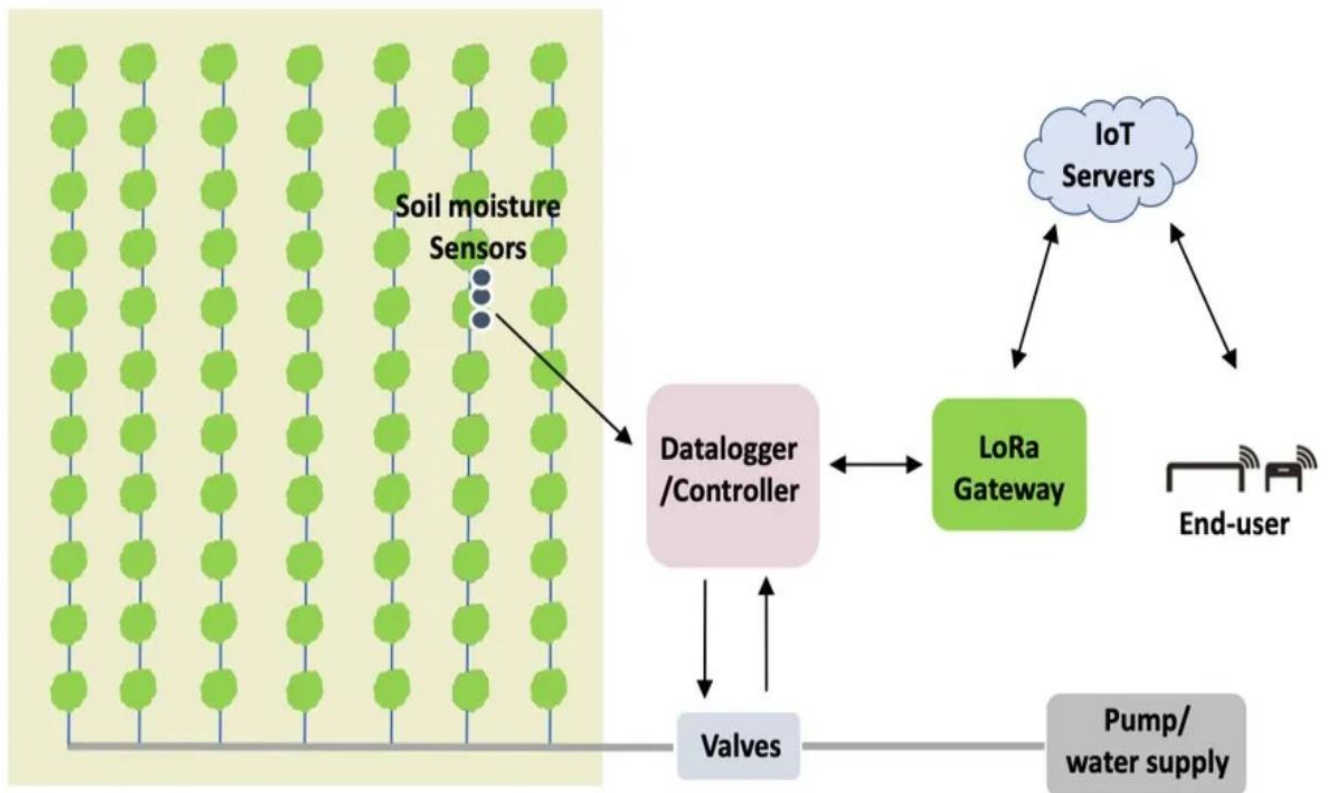


Рисунок 1.7 – Система автоматичного зрошення IoT показана

Система зрошення IoT є системою двостороннього зв'язку. Одна з них полягає в тому, щоб надати кінцевим користувачам дані датчика та стан клапана (увімкнено/вимкнено). Інший – керування клапаном за допомогою команди кінцевого користувача або автоматично за допомогою попередньо встановленого порогу вологості ґрунту. Датчики вологості ґрунту розміщують у саду та підключають до реєстратора/контролера даних. Електромагнітний клапан також підключено до реєстратора/контролера даних. Шлюз LoRa® (Long Range) використовується як комунікаційний центр і підключений до Інтернету в офісі. Існує можливість використовувати різні сервери IoT для передачі даних, наприклад Things Network і Allthingstalk.

Складовими компонентами запропонованої системи є водяний насос, давачі вимірювання вологості ґрунту, соленоїдний клапан. Вода, що подається в зрошувальну систему, має бути доступною без втручання людини (наприклад, ручного запуску насоса), щоб подавати воду автоматично. У більшості випадків потрібен електричний насос із контролером насоса на основі тиску води.

Зм.	Арк.	№докум.	Підпис	Дата

Наприклад, якщо тиск води на виході встановлено на 40 psi, насос увімкнеться, коли тиск знизиться до нижчого значення (зазвичай регулюється). Під час зрошення, коли відкривається клапан, тиск води падає, що призводить до запуску водяного насоса для підтримки тиску води 40 psi.

В запропонованій системі було використано три датчики вологості ґрунту (Watermark 200SS-5, Irrrometer Company, Inc.), що були встановлені на 1, 2 та 3 фути відповідно. Цей тип датчика широко використовується для вимірювання рівнів вологості ґрунту для різноманітних культур. Кожен датчик був встановлений на одному кінці ½-дюймової ПВХ труби. Для кожного датчика була викопана яма біля крапельної лінії відповідної глибини для розміщення у ній датчика.

Електромагнітний клапан (соленоїди) з блокуванням постійного струму використовувався для контролю зрошення шляхом підключення та відключення подачі води. Клапан управлявся двостороннім реле з 9-вольтовою батареєю. Для кожної зони/блоку зрошення потрібен електромагнітний клапан для автоматичного зрошення.

Також у системі присутній датчик тиску. Датчик тиску, встановлений за електромагнітним клапаном у трубопроводі, вказує на стан зрошення (увімкнено чи вимкнено). В найкращому випадку значення 0 psi вказує на вимкнення поливу, тобто закриття клапана, а додатне число (наприклад, 20 psi) означає, що зрошення ввімкнено, тобто клапан відкрито. Цей індикатор є критично важливим для перевірки будь-яких ненормальних умов у полі, таких як неправильна реакція клапана або витік.

Таким чином зазначені вище системи зрошення, які використовують різні технології, є досить актуальними та корисними в області сільського та домашнього господарства, проте основним їх недоліком є відсутність моніторингу стану ґрунту (та відповідно зрошення) на декількох ділянках. Іншим недоліком відомих систем є їх відносно висока вартість.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						15
Зм..	Арк.	№докум.	Підпис	Дата		

### 1.3 Постановка задачі

Створення автоматизованих систем розумного поливу, заснованих на одноплатній комп'ютерній системі Raspberry Pi, є важливим завданням, яке пов'язане з потребою протистояти кліматичним змінам та збільшенням ефективності використання водних ресурсів. Тому для вирішення цього завдання пропонується виконання наступних етапів:

1. аналізу відомих пристроїв, засобів та систем для автоматизованого поливу;
2. складання вимог програмно-технічного пристрою розумного поливу на базі Raspberry Pi;
3. проектування структури програмно-технічного пристрою розумного поливу на базі Raspberry Pi та вибір програмних та апаратних компонентів;
4. проектування монтажною схеми та схеми електричної принципової програмно-технічного пристрою розумного поливу на базі Raspberry Pi;
5. реалізація функцій розумного поливу.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						16
Зм..	Арк.	№докум.	Підпис	Дата		

## 2 ПРОЄКТУВАННЯ ПРОГРАМНО-ТЕХНІЧНОГО ПРИСТРОЮ РОЗУМНОГО ПОЛИВУ НА БАЗІ RASPBERRY PI

2.1 Вимоги до програмно-технічного пристрою розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi

Один з ключових етапів проектування будь-якої системи або програмно-технічного пристрою – визначення їх функціональних вимог. Ці вимоги визначають, які завдання повинні бути виконані системою або програмно-технічним пристроєм, тобто які функції мають бути реалізовані. Чітко сформульовані вимоги допомагають покращити ефективність та якість розробки системи.

Сформулюємо набір функційних вимог до програмно-технічного пристрою розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi наступним чином:

1. Нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту. Дана функція повинна виконуватись на регулярній основі, постійно нагадуючи власнику програмно-технічного засобу про доцільність перевірки рівня води. Слід відзначити, що безпосередньо значення рівня води не вимірюється, а контроль за наявністю води покладений на користувача програмно-технічного пристрою.

2. Визначення рівня вологості ґрунту. Реалізація даного функціоналу передбачає постійний моніторинг рівня вологості ґрунту на декількох ділянках (під ділянкою мається на увазі окремий горщик для рослини або грядка). У випадку зниження рівня вологості ґрунту (нижче заданого порогового значення) на ділянці програмно-технічний пристрій повинен виконати полив відповідної ділянки ґрунту.

Таким чином зазначені функції дозволяють проводити моніторинг вологості ґрунту та здійснювати полив на декількох ділянках ґрунту, а не на одній, що

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						17
Зм.	Арк.	№докум.	Підпис	Дата		

значно розширює сферу застосування проектованого програмного-технічного пристрою.

Окрім функціональних вимог, визначення нефункційних вимог є також важливим етапом проектування будь-якої системи. Нефункційні вимоги описують такі характеристики системи, як вартість, безпека, надійність, масштабованість, зручність використання, та інші.

Основними нефункційними вимогами до проектованого програмно-технічного пристрою є:

- Низька вартість програмно-технічного пристрою;
- Стабільна та безперебійна робота пристрою;
- Масштабованість;
- Зручність використання;
- Доступність;
- Мобільність.

Таким чином, описані вище функційні та нефункційні вимоги описують функції, яким повинен задовольняти проектований програмно-технічний пристрій розумного поливу.

## 2.2 Структура програмно-технічного пристрою розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi

За для реалізації визначених вимог до програмно-технічного пристрою розумного поливу на базі одноплатної комп'ютерної системи Raspberry Pi запропоновано структуру даного пристрою, структурну схему якого представлено на рис. 2.1. Особливістю запропонованого рішення є здатність програмно-технічного пристрою до моніторингу вологості ґрунту та поливу одночасно на трьох ділянках ґрунту, що значно розширює сферу застосування даного пристрою, у порівнянні із відомими рішеннями. Запропонований програмно-технічний пристрій розумного поливу складається із одноплатної комп'ютерної

системи Raspberry Pi, АЦП ADS1115, водяних насосів, модулів реле та датчиків вимірювання вологості ґрунту.

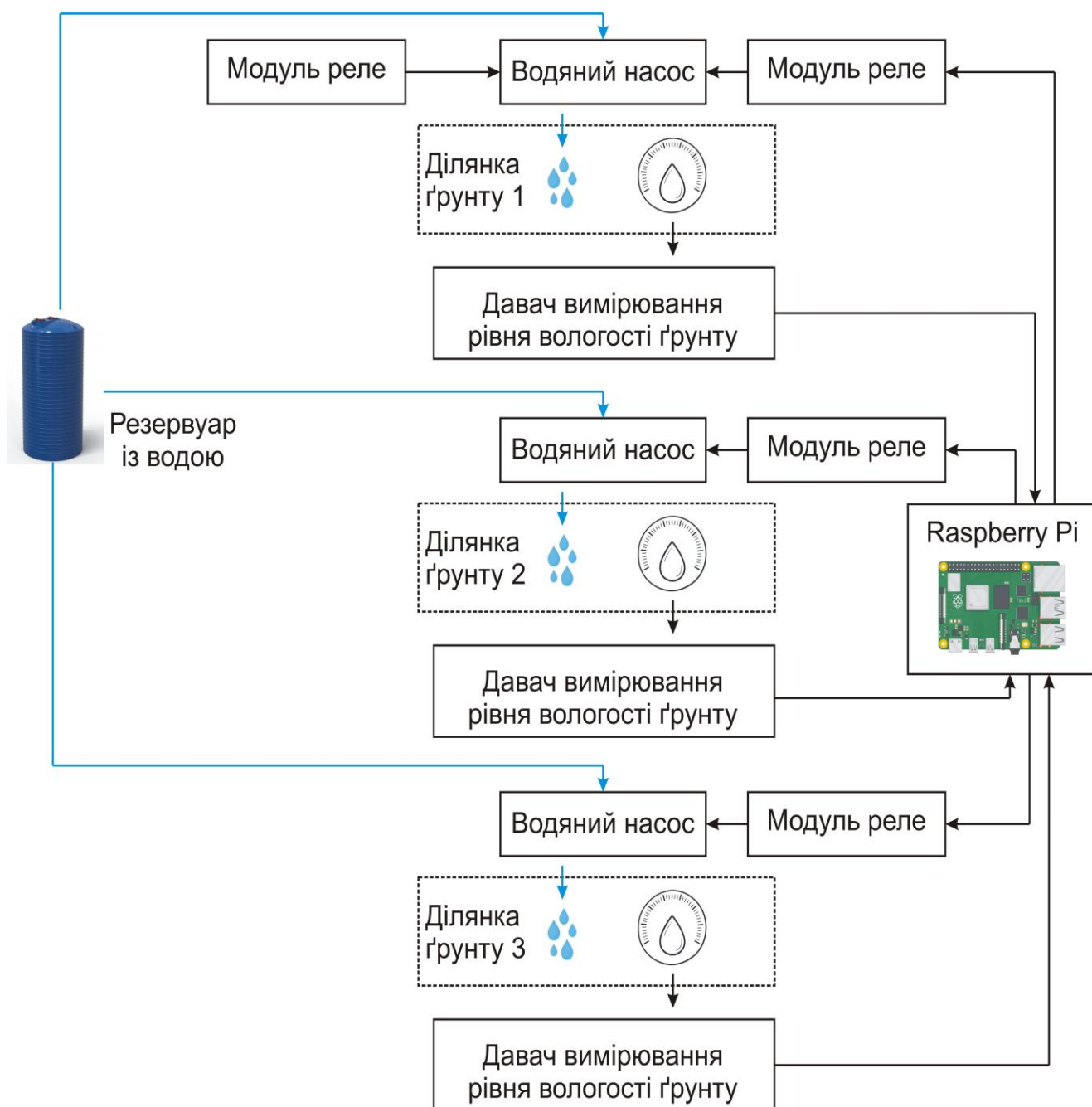


Рисунок 2.1 – Структурна схема програмно-технічного пристрою розумного поливу на базі одноплатної комп’ютерної системи Raspberry Pi

Розглянемо детальніше компоненти із яких складається запропонований програмно-технічний пристрій:

Зм.	Арк.	№докум.	Підпис	Дата

– Одноплатна комп'ютерна система Raspberry Pi, основними функціями якої є контроль рівня вологості ґрунту та полив ділянок ґрунту. Виконання цих функцій реалізується за допомогою водяних насосів та датчиків визначення вологості ґрунту. На Raspberry Pi встановлено додатково програмне забезпечення Schedule для виконання періодичних завдань та модуль smtplib для відправки сповіщення на електронну пошту користувача про здійснений полив ділянки ґрунту, а також повідомлення-нагадування про необхідність доливу води у резервуар.

– Датчик вимірювання рівня вологості ґрунту для вимірювання показнику вологості ґрунту;

– Модулі реле, що використовуються для керування водяними насосами. Необхідність застосування модулів реле продиктовано потребою керування навантаженням у вигляді насосу;

– Аналогово-цифровий перетворювач ADS1115 використовується для перетворення рівнів напруги від датчиків вимірювання вологості ґрунту;

– Програмний модуль schedule, що використовується для виконання завдань із заданою періодичністю;

– Програмний модуль smtplib, що використовується для відправки повідомлень на електронну пошту.

Таким чином, за допомогою визначення компонентів та їх зв'язків в структурі програмно-технічного пристрою розумного поливу на базі Raspberry Pi можна успішно втілити всі вимоги до цієї системи.

### 2.3 Аналіз обраних рішень

Після виконання проектування структури й визначення принципів функціонування програмно-технічного пристрою розумного поливу на базі Raspberry Pi, наступним етапом є відбір та оцінка можливостей апаратного та програмного забезпечення, що будуть задіяні у запропонованій системі.

Розглянемо детальніше характеристики використаних програмних та апаратних компонентів.

### 2.3.1 Аналіз обраних апаратних рішень

Запропонований програмно-технічний пристрій розумного поливу на базі Raspberry Pi складається із таких апаратних компонентів: одноплатної комп'ютерної системи Raspberry Pi, датчиків вимірювання вологості рівня ґрунту, модулів реле, водяних насосів та мікросхеми АЦП.

Головним апаратним компонентом у проектуваному програмно-технічному пристрою є одноплатна комп'ютерна система Raspberry Pi 4 (рис. 2.2). Raspberry Pi, який має розмір банківської картки, спочатку був розроблений як доступна система для навчання інформатики, але згодом став дуже популярним і знайшов застосування в багатьох інших галузях. Сьогодні існує безліч версій цього одноплатного комп'ютера, зокрема Raspberry Pi 4, який використовує чотирьохядерний процесор ARM Cortex A72 з тактовою частотою 1,5 ГГц та новий стандарт оперативної пам'яті LPDDR4, що забезпечує значно більшу продуктивність порівняно з попередніми версіями, де використовувалася LPDDR2. В даній одноплатній комп'ютерній системі, SD карти продовжують використовуватися як засіб зберігання даних, але тепер є можливість завантажувати дані з USB, а також є порти USB 3.0 і Type-C. Швидкість читання/запису для слоту SD тепер досягає до 50MBps, що в два рази більше в порівнянні з попередньою версією. За допомогою доступного SATA SSD і USB 3.0 перехідника можна створити невеликий, але продуктивний комп'ютер з ОС Linux.

Для живлення Raspberry Pi 4 використовується блок живлення на 5.2В і 3А, який підключається через роз'єм Type-C. Незважаючи на більш потужний процесор і високе споживання енергії, Raspberry Pi не генерує багато тепла. Після двох хвилин бездіяльності температура процесора становить 55 градусів за Цельсієм, а в тестах температура піднімається не вище 66 градусів. Якщо

					КВРКІ. 2001128.20.01.07 ПЗ	Арк. 21
Зм.	Арк.	№докум.	Підпис	Дата		

необхідно підтримувати Raspberry Pi під навантаженням, можна встановити охолоджувальні радіатори.

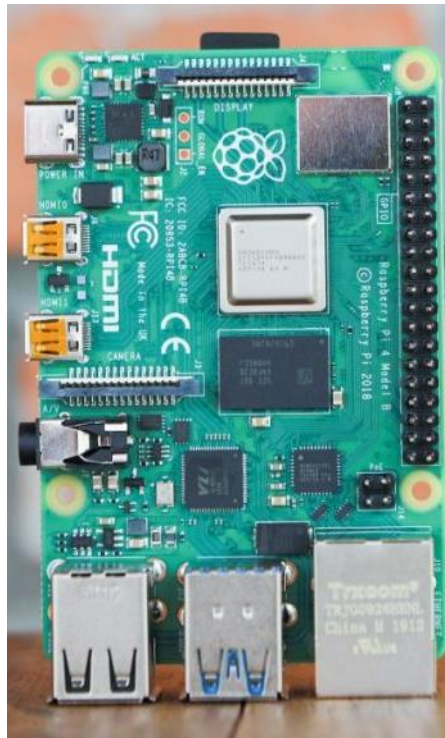


Рисунок 2.2 – Одноплатна комп'ютерна система Raspberry Pi

Призначення та номери портів GPIO Raspberry Pi 4 наведено на рис. 2.3.

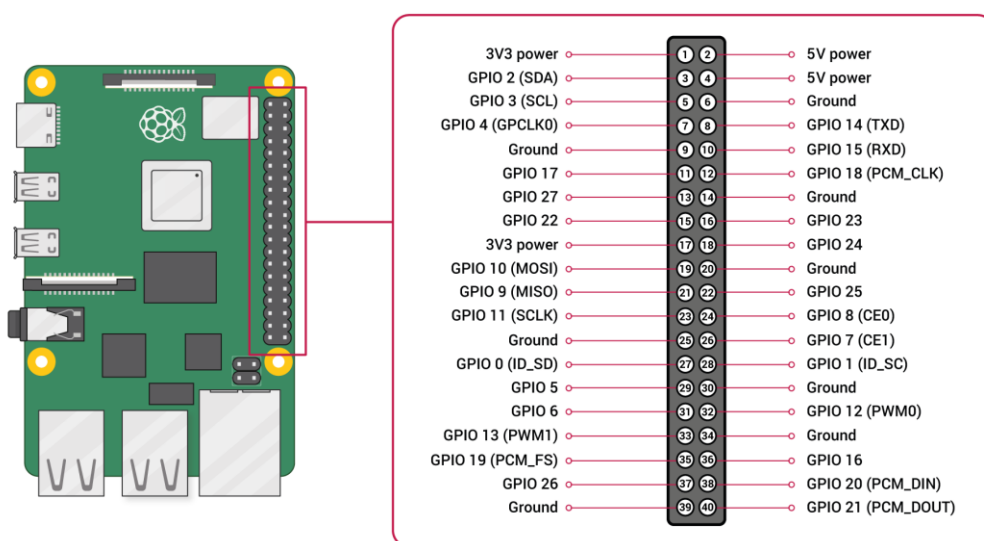


Рисунок 2.3 – Призначення портів GPIO Raspberry Pi 4

Незважаючи на те, що Raspberry Pi має значну кількість контактів GPIO (general purpose input/output), він не має жодного вбудованого або бортового аналого-цифрового перетворювача. Тому Raspberry Pi сам по собі не може збирати дані з аналогових датчиків. У вбудованих програмах аналогові датчики часто використовуються. Фактично, багато датчиків, таких як світлозалежні резистори (LDR) та давачі вимірювання вологості ґрунту, недоступні в цифровій формі. У цьому випадку аналогові датчики можна підключати до Raspberry Pi двома способами. Один – використовувати зовнішній аналого-цифровий перетворювач; другий – вибірка даних з аналогових датчиків за допомогою мікроконтролера та передача всіх отриманих даних на Raspberry Pi через послідовний інтерфейс.

Комунікація одноплатної комп'ютерної системи Raspberry Pi з мікроконтролером лише для отримання аналогових даних не є оптимальним рішенням. Зовнішній АЦП є найкращим рішенням, коли Raspberry Pi працює як окремий вбудований мікрокомп'ютер і потрібен аналоговий збір даних.

Збір аналогових даних на Raspberry Pi має кілька переваг. Набагато легше застосувати алгоритми машинного навчання та моделі глибокого навчання в мікрокомп'ютері, такому як Raspberry Pi, ніж у будь-якому мікроконтролері. Мікроконтролери занадто обмежені в ресурсах і мають обмежені програмні засоби та API для застосування алгоритмів машинного навчання або реалізації мереж глибокого навчання. Доступні програмні засоби машинного та глибокого навчання для мікроконтролерів також обмежені у своїх можливостях. Наприклад, програми комп'ютерного бачення зазвичай потребують великої оперативної пам'яті та ефективних датчиків камери для будь-якого практичного застосування. Інструменти машинного навчання для мікроконтролерів не дуже ефективні для таких ресурсомістких вбудованих програм.

У проєктованому-програмно технічному засобі пропонується використати 16-розрядний АЦП із 4 аналоговими входами ADS1115 (рис.2.4). ADS1115 підтримує два типи збору даних – односторонній і диференціальний. При односторонньому перетворенні аналогові дані з кожного з чотирьох аналогових

вхідних контактів незалежно відбираються. Різниця між напругами на двох аналогових вхідних контактах вимірюється в диференціальному режимі роботи. ADS1115 має роздільну здатність 16 біт, з яких один біт залишається зарезервованим для знака плюс/мінус різниці напруг у диференціальному режимі, а решта 15 біт використовуються для зберігання перетворених значень напруги. Мікросхема має інтерфейс I2C для послідовного обміну даними з мікрокомп'ютером/мікроконтролером.



Рисунок 2.4 – Мікросхема аналогово-цифрового перетворювача ADS1115

Частота дискретизації в ADS1115 може бути встановлена в діапазоні від 8 до 860 вибірок в секунду. Чим вища частота дискретизації тим менше часу, необхідно АЦП для захоплення та перетворення аналогового сигналу. Мікросхема також має вбудований підсилювач сигналу, що посилює сигнали слабкої напруги від 2-3 до 16 разів. Перетворення (одностороннє або диференціальне) можна здійснити двома способами. Перетворення ініціюється Raspberry Pi (або керуючим мікрокомп'ютером/мікроконтролером) одним методом. Перетворене значення зберігається в регістрі на ADS1115. Потім Raspberry Pi зчитує вміст регістру. У другому методі аналогово-цифрове перетворення виконується ADS1115 безперервно. Сигнал дискретизується з попередньо запрограмованою швидкістю (8~860 sps), а перетворене значення багаторазово оновлюється в

Зм..	Арк.	№докум.	Підпис	Дата



подається живлення. На платі є червоний світлодіод, що горить, коли датчик передає дані в контролер.

Таблиця 2.2 – Призначення контактів АЦП ADS1115

Вивід		Опис
1	VDD	Живлення
2	GND	Заземлення
3	SCL	Керуючий сигнал SCL по шині I2C
4	SDA	Керуючий сигнал SDA по шині I2C
5	ADDR	Пін адреси
6	ALRT	Alert/ready
7	A0	Аналоговий вхід 0
8	A1	Аналоговий вхід 1
9	A2	Аналоговий вхід 2
10	A3	Аналоговий вхід 3



Рисунок 2.5 – Давач вимірювання рівня вологості ґрунту FC-28

Управління датчиком здійснюється або від контролера Arduino, або від іншого керуючого мікропроцесорного пристрою за допомогою спеціальних програм. На платі розташований змінний резистор, який використовується для налаштування порога спрацьовування датчика (чутливості).

Датчик вологості ґрунту має два інтерфейси для підключення до живлення та мікроконтролера, для підключення елемента, за допомогою якого здійснюється вимірювання:

– Для підключення до живлення та мікроконтролера використовується 4-контактний штирьовий інтерфейс. Позначення контактів: VCC – напруга живлення; GND - загальний контакт; D0 – цифровий вихід; A0 – аналоговий вихід.

– Для підключення чутливого елемента використовується 2-контактний штирьовий інтерфейс.

Живлення датчика здійснюється або від Arduino контролера (або від іншого керуючого мікропроцесорного пристрою) або зовнішнього джерела живлення (блоку живлення, батареї). Напруга живлення датчика складає 3,3 - 5В.

Характеристики вимірювання рівня вологості ґрунту використано давач вологості ґрунту FC-28:

- Вимірювальний елемент – металізований щуп (винесений окремий).
- Виконання датчика – на базі мікросхеми LM393 із налаштуванням чутливості.
- Напруга живлення датчика: 3,3 - 5В.
- Виходи датчика: цифровий та аналоговий.
- Габарити датчика: 63,8 x 20 x 8 мм.
- Габарити модуля: 38 x 15,6 x 8 мм.
- Довжина кабелю: 20 см.
- Вага давача: 11 г.

Ще одним апаратним компонентом, що використовуватиметься у даному програмно-технічному пристрої є реле. Основне призначення реле це замикання і розмикання електричного кола за допомогою електромагніту. При подачі напруги

на електромагнітну котушку, в ній виникає електромагнітне поле, яке притягує металевий контакт і контакти потужного навантаження замикаються, що тим самим дозволяє комутувати ланцюги із потужним навантаженням.

В даній роботі в якості модуля реле пропонується використати модуль Songle SRD-05VDC-SL-C, схему електричну принципову якого наведено на рис. 2.6.

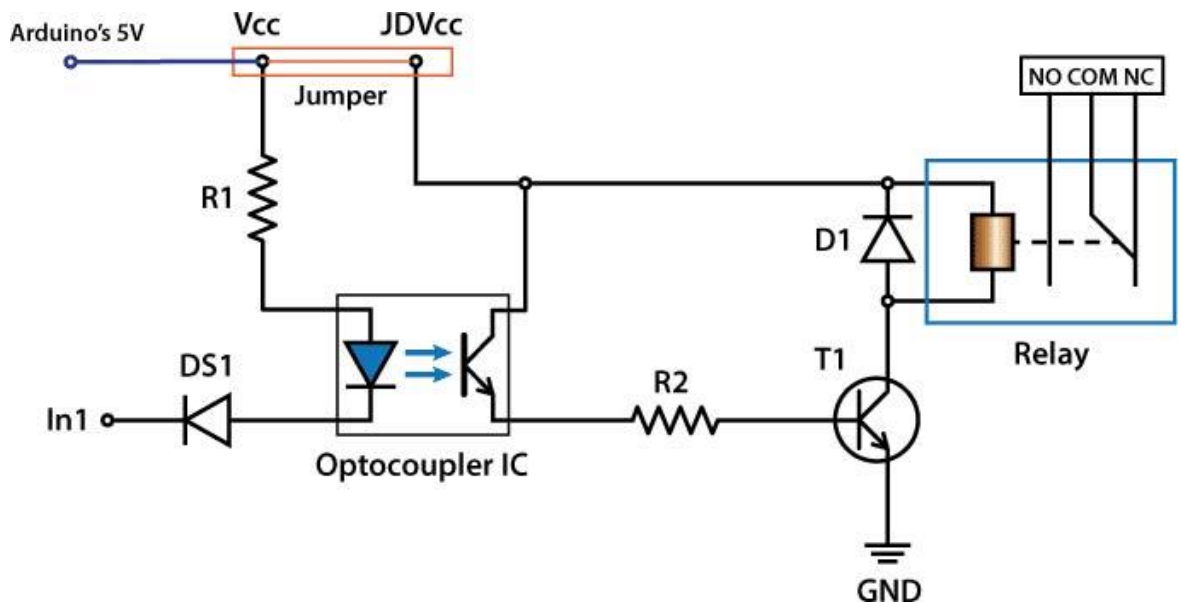


Рисунок 2.6 – Схема електрична принципова модуля реле Songle SRD-05VDC-SL-C

При підключенні до напруги, вивід In1 перебуває у високому стані (логічна 1), для перемикавання першого реле необхідно вивід In1 перевести в низький рівень (логічний 0), тобто закоротити ланцюг на землю. Узагальнюючи можна сказати, що реле керується логічним нулем. Через світлодіод, які знаходиться в оптопарі почне протікати струм, після засвічення якого відкриється фототранзистор, через який так само почне текти струм на базу транзистора, який відкриється і реле спрацює. Таким чином сигнал з мікроконтролера відділяється від навантаження через промінь світла, що дуже важливо при комутації високовольтних або інших чутливих ланцюгів. Слід відзначити, що модуль реле може працювати і від окремого джерела живлення. Для цього необхідно прибрати перемичку і підключити живлення до контактів JD-VCC і GND.

Для захисту виходів мікроконтролерів від стрибків напруги в котушці реле, застосовується транзистор і оптрон. Оптопара або оптрон – елемент, що дозволяє виконувати дві функції: комутувати навантаження, а також повністю фізично розв'язувати мікроконтролер із навантаженням. Оптрони можна використовувати для імітації натискання кнопок у інших зовнішніх пристроях, тобто замикати чисто логічний сигнал.

Також слід відзначити, що реле в якості виконавчого елемента використовує котушку. Котушка має досить високу індуктивність, що не дозволяє різко обірвати струм в ній. Якщо це спробувати зробити, то потенційна енергія, накопичена в електромагнітному полі, з'явиться в іншому місці. При різкому перериванні струму, на котушці буде потужний сплеск напруги, в сотні вольт. Якщо струм обривається механічним контактом, то буде повітряний пробій – іскра. Для згладжування цього негативного явища в схемі використовується діод. При нормальній роботі діод включений зустрічно напрузі і струм через нього не йде. А при вимкненні (фізичного розриву) напруга на індуктивності буде вже в іншу сторону і струм пройде через діод, тим самим повільно вивільняючи накопичену енергію.

Потужність навантаження, яку можуть вмикати/вимикати модулі, обмежується встановленими на платі реле. Характеристики реле Songle SRD-05VDC-SL-C:

- Робочий струм котушки: 71 мА.
- Робоча напруга: 5 В.
- Максимальне комутоване постійна напруга: 28 В.
- Максимальне комутоване змінну напругу: 250 В.
- Максимальний комутований струм: 10А.
- Робочий температурний режим: від -25 до + 70 °С.

Реле Songle SRD-05VDC-SL-C має 5 контактів (рис. 2.7). Де 1 і 2 контакт використовуються для подачі живлення на реле. Група контактів 3 і 4 являють собою нормально розімкнуті контакти (NO), група контактів 3 і 5 – нормально замкнуті (NC). Подібні реле бувають на різні напруги живлення: 3, 5, 6, 9, 12, 24,

48 В. В даному випадку використовується 5-вольтовий варіант, що дозволяє жити реле-модуль безпосередньо від Arduino або Raspberry Pi.

Розглянемо розташування контактів на релейному модулі та варіанти підключення цього модуля до мікроконтролера (рис. 2.7).

На платі є перемичка (JDVcc), для живлення реле або від мікроконтролера, або від окремого джерела живлення. Пінами In1, In2, In3, In4 модуль підключається до цифрових виходів Arduino або Raspberry Pi.

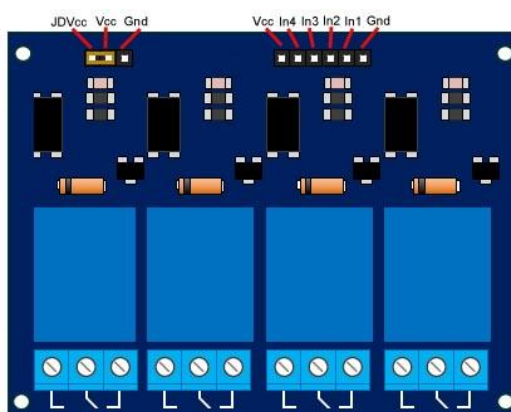


Рисунок 2.7 – Розташування та позначення контактів чотирьохканального релейного модуля Songle SRD-05VDC-SL-C

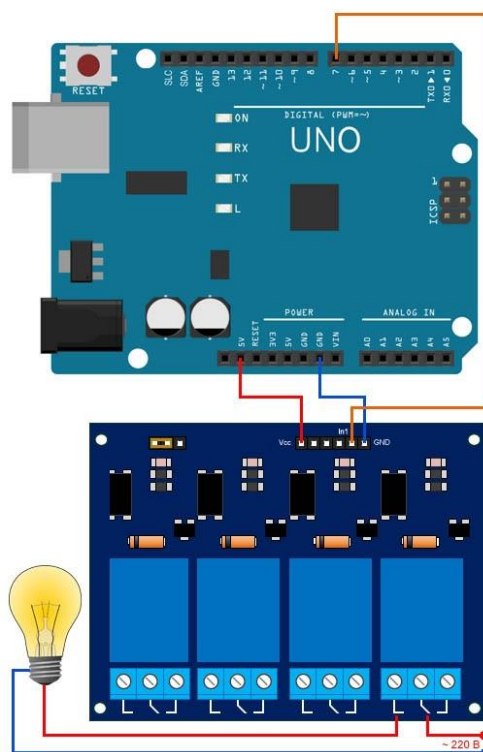


Рисунок 2.8 – Підключення модуля SRD-05VDC-SL-C до Arduino

Зм.	Арк.	№докум.	Підпис	Дата

Якщо модуль реле розрахований на живлення 5 В, то живлення для його підключення можна взяти від самого мікроконтролера. У прикладі на рис. 6 наведено підключення одного реле. В якості навантаження буде використано лампочку на 220 в.

Для живлення реле модуля від Arduino, перемичка повинна замикати піни «Vcc» і «JDVcc», зазвичай по-замовчуванню вона там і встановлена. Якщо використовується реле не на 5 вольт, жити від Ардуіно модуль не можна, живлення потрібно брати від окремого джерела.

Наведені нижче схема (рис. 7) показує, як підключити живлення модуля від окремого джерела. За такою схемою потрібно підключати реле, розраховане на живлення більше ніж на 5 В. Для 5 В реле ця схема так само буде більш краща.

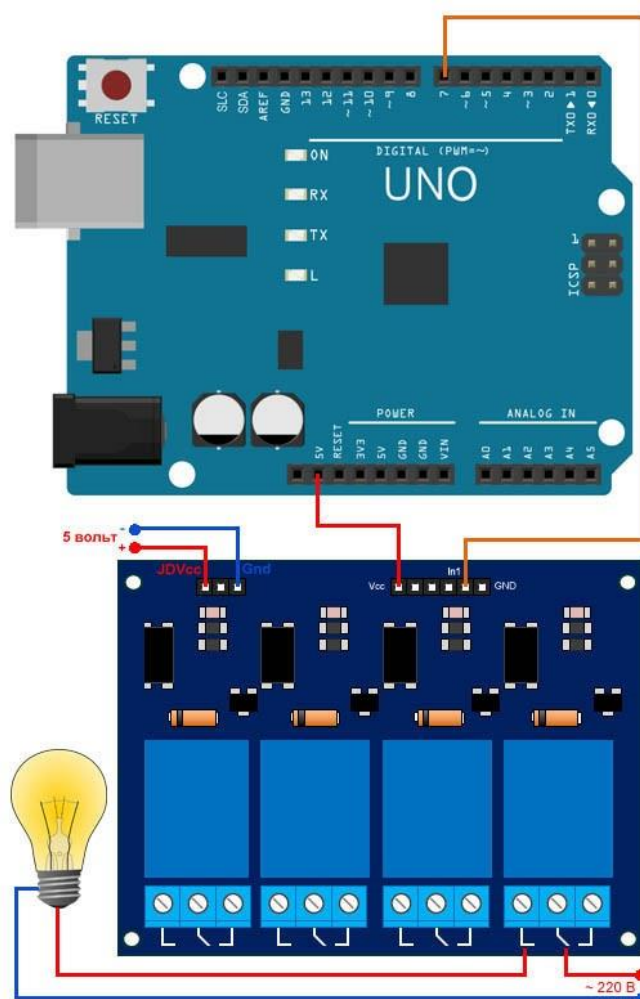


Рисунок 2.9 – Підключення модуля SRD-05VDC-SL-C до Arduino із використанням зовнішнього живлення для реле

Зм..	Арк.	№докум.	Підпис	Дата

При такому підключенні потрібно прибрати перемичку між пінами «Vcc» і «JDVcc». Далі пін «JDVcc» підключити до «+» зовнішнього джерела живлення, пін «Gnd» підключити до «-» джерела живлення. Пін «Gnd», який в попередній схемі підключався до пину «Gnd» Ардуіно, в даній схемі не підключається. При цьому живлення на вивід VCC керуючого роз'єму також подається (від +5 В Arduino). У наведеному прикладі, зовнішнє джерело живлення 5 В, якщо реле розраховане на іншу напругу (3, 12, 24 В), вибирається відповідне зовнішнє живлення.

Узагальнюючи можна відзначити, при використанні електромагнітного реле повинно враховувати наступні особливості:

- Реле має обмежену кількість перемикачів: механічний контакт зношується, особливо при великому і / або індуктивному навантаженні;
- Електромагнітне реле при комутації контактів створює звук.
- При великому навантаженні реле може "залипнути", тому для великих струмів потрібно використовувати більш потужні реле, які доведеться включати за допомогою менш потужних реле або транзисторів.
- Необхідні додаткові ланцюги для управління реле, так як котушка є індуктивним навантаженням, і навантаження самого по собі реле є занадто великим для піна мікроконтролера (вирішується використанням готових модулів реле, розрахованих для Arduino та Raspberry Pi).
- Великі наведення на всю лінію живлення при комутації індуктивного навантаження.

### 2.3.2 Аналіз обраних програмних рішень

За для реалізації визначених функцій програмно-технічного пристрою розумного поливу на базі Raspberry Pi, на одноплатній комп'ютерній системі було розгорнуто програмне забезпечення для виконання періодичних завдань (schedule) та відправки електронних повідомлень (smtplib).

Модуль Schedule – це планувальник у процесі для періодичних завдань, які використовують для конфігурації шаблону конструктора. Розклад дозволяє запускати функції Python (або будь-які інші виклики) періодично через заздалегідь визначені проміжки часу за допомогою простого, зручного для людини синтаксису. Бібліотека розкладів використовується для планування завдання на певний час кожного дня або певного дня тижня. Є можливість також встановити час у 24-годинному форматі. Фактично бібліотека Schedule узгоджує системний час із запланованим часом, що був встановлений. Коли запланований час і системний час збігаються, викликається функція завдання (командна функція, яка запланована). Даний модуль використовуватиметься для періодичного відтворення завдань перевірки необхідності поливу ділянок ґрунту та відправки повідомлень-нагадувань на електронну пошту користувача.

Для відправки електронних повідомлень на Raspberry Pi також встановлено модуль smtplib. Даний модуль надає можливість надсилати електронні листи за допомогою протоколу SMTP (Simple Mail Transfer Protocol). Він дозволяє з'єднатися з SMTP-сервером, аутентифікуватися зі своїми обліковими даними (якщо потрібно) та відправляти електронні повідомлення.

## 2.4 Висновки до розділу 2

Таким чином було визначено функційні та нефункційні вимоги до проєктованого програмно-технічного пристрою, що передбачає виконання функцій нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту, а також визначення рівня вологості ґрунту та здійснення автоматичного поливу. При проєктуванні пристрою основна увага була приділена можливості виконання функцій вимірювання і поливу на декількох (трьох) ділянках, а не на одній. Також запропоновано структуру програмно-технічного пристрою та проведено аналіз апаратних та програмних компонентів.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		33

### 3 РЕАЛІЗАЦІЯ ПРОГРАМНО-ТЕХНІЧНОГО ПРИСТРОЮ РОЗУМНОГО ПОЛИВУ НА БАЗІ RASPBERRY PI

3.1 Підготовка та встановлення необхідних бібліотек для програмної реалізації пристрою

Важливою складовою будь-якого програмно-технічного пристрою на основі Raspberry Pi є його комунікація (або обмін даними) із іншими пристроями (давачі, виконавчі пристрої). Одним із можливих рішень є використання шини I2C.

Виконаємо встановлення необхідних засобів для роботи із I2C:

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

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

Далі перейдемо у налаштування, виконавши команду:

```
sudo raspi-config
```

Та перейдемо у розділ Interfacing Options (рис. 3.1).

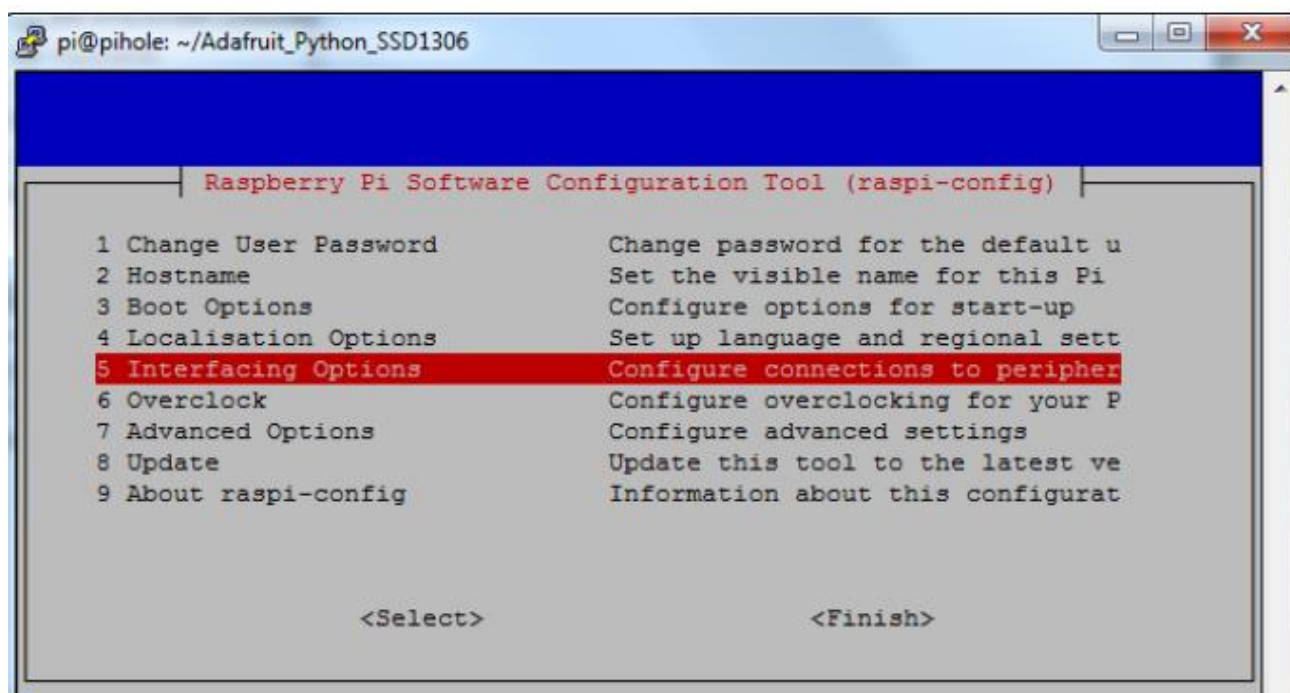


Рисунок 3.1 – Вибір вкладки Interfacing Options

Далі виберемо I2C (рис. 3.2 та 3.3).

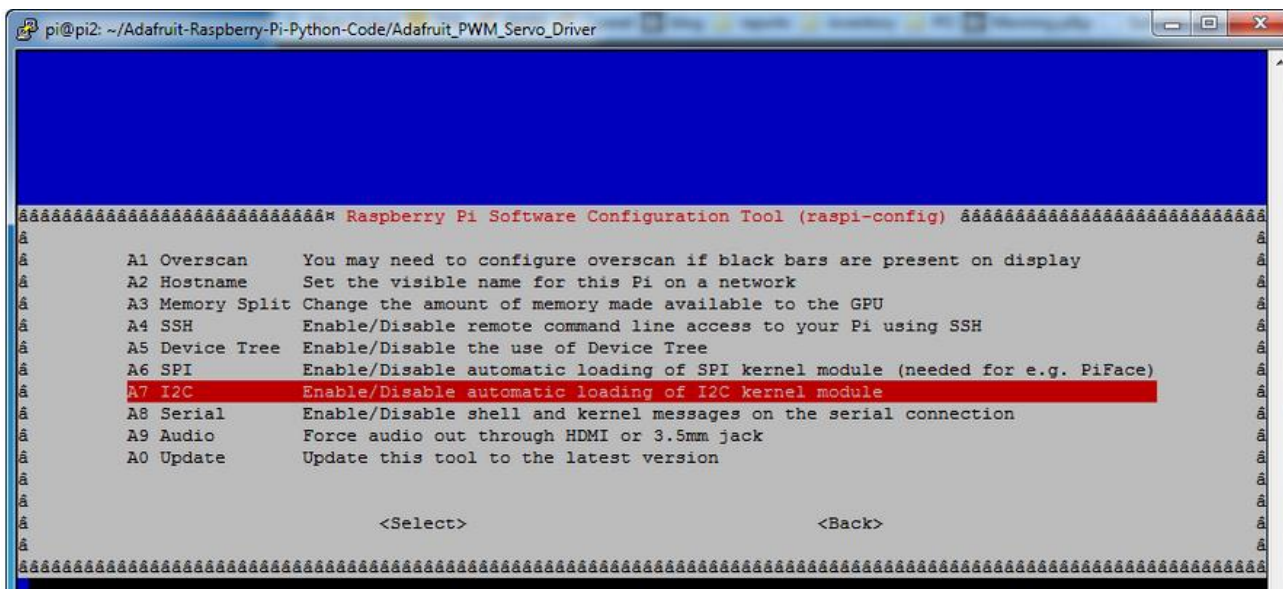


Рисунок 3.2 – Увімкнення I2C

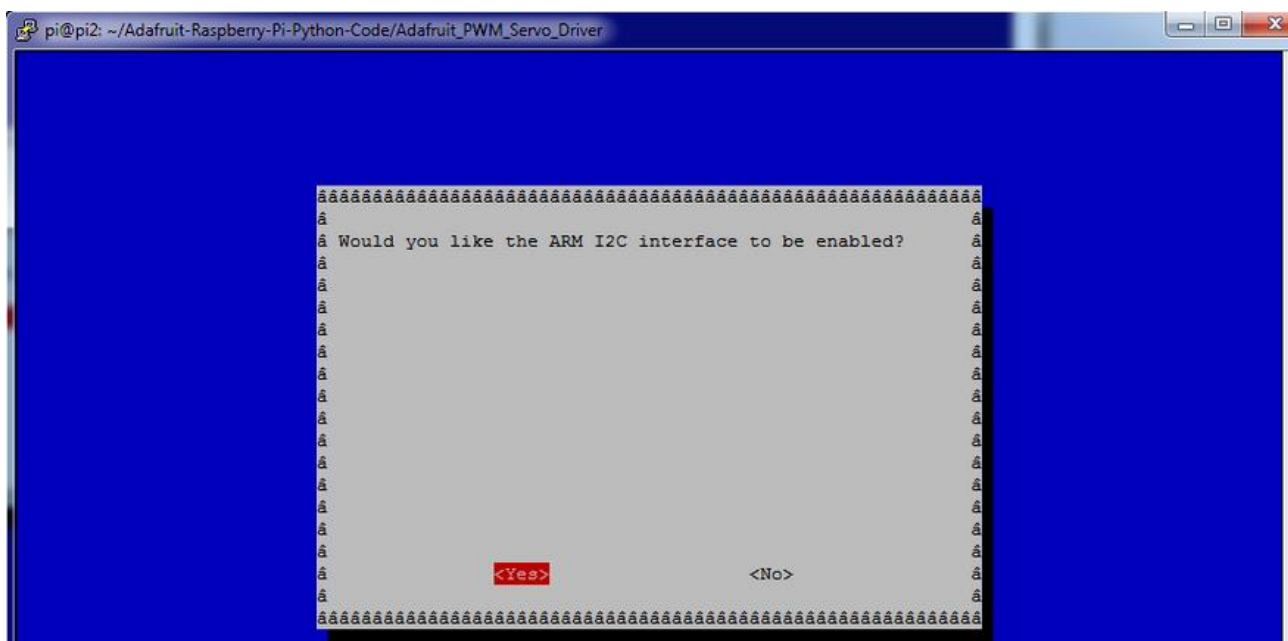


Рисунок 3.3 – Підтвердження увімкнення I2C

Після чого виконаємо перезавантаження Raspberry Pi:

```
sudo reboot
```

Далі виконаємо під'єднання АЦП ADS1115 до Raspberry Pi та перевіримо його підключення по I2C шині, виконавши команду:

```
sudo i2cdetect -y 1
```

Зм..	Арк.	№докум.	Підпис	Дата

В результаті її виконання буде виведено адреси пристроїв, що під'єднанні по I2C шині. На рис. 3.4 відображено адресу 0x48, яка представляє АЦП ADS1115. Слід звернути увагу, що підключення ADS1115 було виконано таким чином, що пін ADDR було підключено до піна GND (рис.3.5).

```
File Edit Tabs Help
pi@raspberrypi ~ $ sudo i2cdetect -y 1
 0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  48  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
pi@raspberrypi ~ $
```

Рисунок 3.4 – Адреси під'єднаних пристроїв по I2C шині

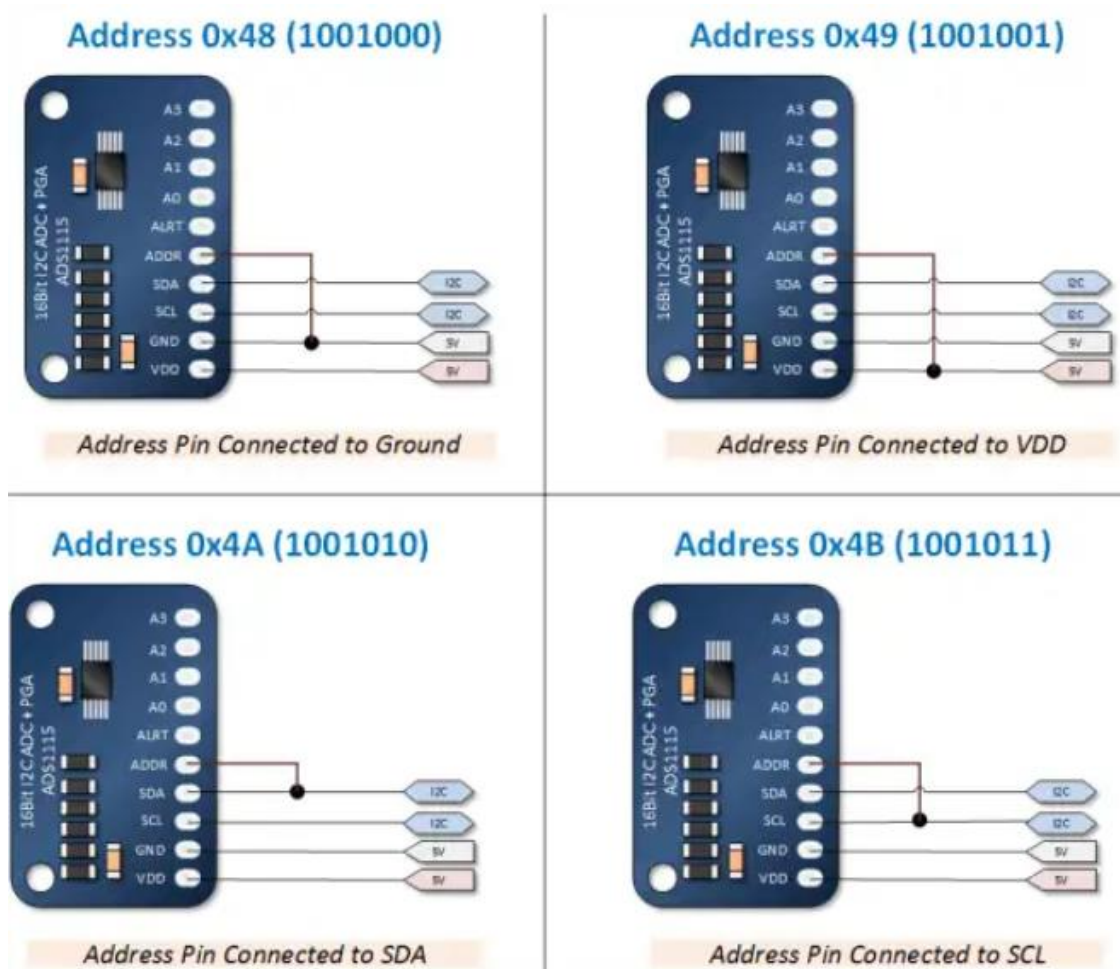


Рисунок 3.5 – Варіації підключення піна ADDR

Зм..	Арк.	№докум.	Підпис	Дата

Настпним кроком є встановлення бібліотеки для роботи із АЦП ADS1115.

Для цього у терміналі виконаємо наступні команди:

```
sudo apt-get update
sudo apt-get install build-essential python-dev
python-smbus git
cd ~
git clone
https://github.com/adafruit/Adafruit_Python_ADS1x15.git
cd Adafruit_Python_ADS1x15
sudo python setup.py install
```

Таким чином в результаті виконання даного етапу було проведено налаштування обміну даними по I2C шині та встановлено необхідні бібліотеки для роботи із АЦП ADS1115.

### 3.2 Монтажна схема програмно-технічного пристрою розумного поливу на базі Raspberry Pi

Монтажна схема – це документ, що містить інформацію про те, як збирати та підключати компоненти пристрою або системи. Дана схема надає докладну інструкцію з монтажу, підключення та розміщення компонентів, а також містить інформацію про підключення джерел живлення, сигналів, заземлення та інших важливих параметрів.

Основу проєктованого програмно-технічного пристрою розумного поливу на базі Raspberry Pi складають такі апаратні компоненти: давачі вимірювання вологості ґрунту (гігрометр) на базі компаратора LM393, модулі реле, водяні насоси, аналогово-цифровий перетворювач ADS1115, а також одноплатна комп'ютерна система Raspberry Pi.

Створення монтажної плати проєктованого програмного-технічного пристрою здійснювалось у системі автоматизованого проєктування (САПР)

					КВРКІ. 2001128.20.01.07 ПЗ	Арк. 37
Зм..	Арк.	№докум.	Підпис	Дата		

Fritzing, що забезпечує можливість створювати монтажні плати та електричні схеми за допомогою графічного інтерфейсу. У бібліотеці Fritzing міститься значна кількість компонентів, включаючи популярні електронні компоненти, такі як резистори, конденсатори, мікроконтролери, датчики тощо.

Створення монтажної плати передбачає перенесення всіх необхідних компонентів на робочу область за допомогою Drag & Drop механіки. Варто відзначити, що за замовчуванням не всі компоненти присутні у Fritzing. Тому для проектування даного програмно-технічного пристрою всі необхідні компоненти, які відсутні у стандартній бібліотеці, було додатково завантажено та встановлено. Для створення загальної шини живлення та шини заземлення, до яких під'єднуюватимуться всі компоненти пристрою, було додано макетну плату.

Всі датчики вимірювання вологості ґрунту, в загальній кількості 3 одиниці, було пі'єднано до входів A0-A2 АЦП ADS1115, що дозволило перетворити аналоговий сигнал у цифровий. Заземлення мікросхеми пі'єднано до загальної шини заземлення. Контакт мікросхеми Addr з'єднано із входом GND, що дозволило встановити адресу на шині I2C мікросхеми 0x48. Контакти SDA SCL мікросхеми ADS1115 пі'єднано до пінів Raspberry Pi GPIO2 та GPIO3 відповідно. Для підключення цих ліній додано по одному підтягуючому резисторі номіналом 1кОм.

Проектований програмно-технічний пристрій розумного поливу має у своєму складі також і 3 модуля реле, до яких підключенні 3 водяні насоси, які використовуються для поливу ділянок ґрунту. Кожен насос пі'єднаний до контактів Comm та NC (normally closed) реле. В свою чергу модулі реле пі'єднанні до контактів GPIO2 та GPIO3 одноплатної комп'ютерної системи Raspberry Pi. Контакти живлення та заземлення виведені в окремі шини, що з'єднані з зовнішнім джерелом живлення за допомогою DC2.1 роз'єму.

Монтажну схему програмно-технічного пристрою розумного поливу на базі Raspberry Pi наведено на рис. 3.6.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		38

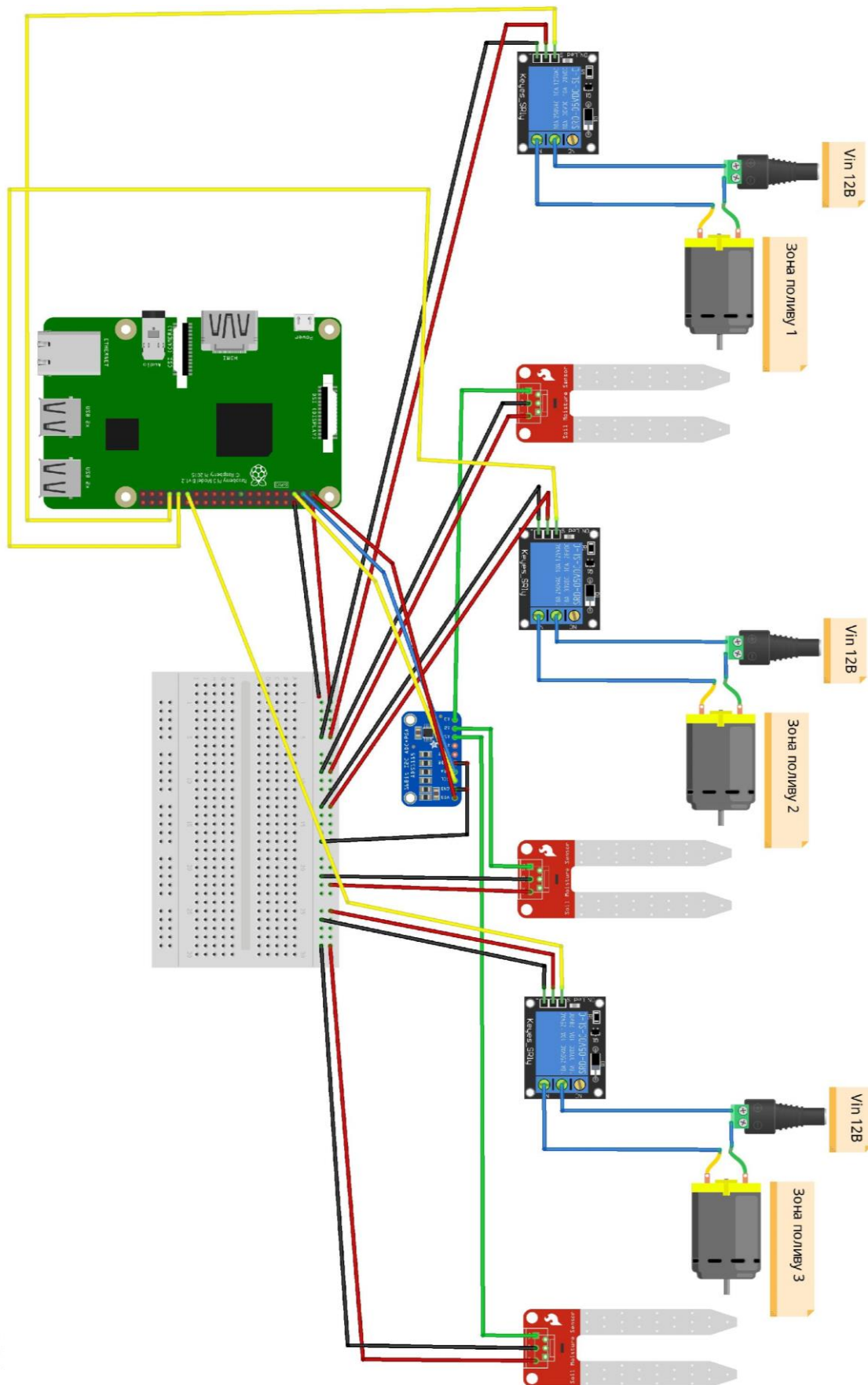


Рисунок 3.6 – Монтажна схема програмно-технічного пристрою розумного поливу на базі Raspberry Pi

Зм.	Арк.	№докум.	Підпис	Дата

### 3.3 Схема електрична принципова програмно-технічного пристрою розумного поливу на базі Raspberry Pi

Відповідно до визначених вимог, проєктований програмно-технічний пристрій розумного поливу на базі Raspberry Pi повинен взаємодіяти із фізичним середовищем, реалізуючи функції моніторингу рівня вологості ґрунту на декілької ділянках (отримання даних від зовнішнього середовища) та автоматичний полив заданих ділянок ґрунту за допомогою водяних насосів (зміна зовнішнього середовища). Таким чином для реалізації програмно-технічного пристрою розумного поливу на базі Raspberry Pi було задіяно наступні дві основні групи компонентів: датчі вимірювання вологості ґрунту та водяні насоси. Схема електрична принципова проєктованого програмно-технічного пристрою розумного поливу на базі Raspberry Pi наведено на рис. 3.7. Загалом проєктований програмно-технічний пристрій здатний керувати трьома насосами та трьома датчами вимірювання вологості ґрунту.

Для під'єднання водяних насосів (позначення на схемі M1-M3) було використано реле (на схемі позначені як REL1-REL3). Реле з'єднує два електричних кола: первинний та вторинний контури. Первинний контур або керуючий контур забезпечує сигнал для ввімкнення або вимкнення реле. В середині реле існує електромагнітна котушка, яка генерує магнітне поле під час проходження струму. Це магнітне поле притягує рухомий якор, який знаходиться в сердечнику реле. Коли знімається живлення із котушки, рухомий якор повертається у вихідне положення. На кінці рухомого якоря розташований рухомий контакт.

До вторинного контуру або керованого контуру здійснюється під'єднання навантаження, яким у даному випадку є водяний насос. Коли рухомий якор притягується до котушки індуктивності, він замикає вторинний ланцюг, тим самим забезпечуючи прямий зв'язок між контактом реле Comm і контактом NO.

Керування реле здійснюється за допомогою біполярного NPN транзистора BC547 (на схемі позначено як Q1-Q3, що відповідає кожному окремому реле).

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						40
Зм.	Арк.	№докум.	Підпис	Дата		

Вихід бази транзистора через резистор 1 кОм під'єднується до GPIO контакту одноплатної комп'ютерної системи Raspberry Pi, колектор підключається до реле, а емітер до заземлення.

Для нівелювання явища стрибка напруги або зворотній електрорушійній силі (ЕРС) у ситуації коли живлення від'єднується від реле та захисту, у схему було додано діод (на схемі позначено як D1-D3, що відповідає кожному окремому реле), що підключається паралельно з котушкою реле. Коли через котушку реле протікає струм, створюється магнітне поле. Таким чином, коли струм раптово припиняється, магнітне поле в котушці має розсіюватися. Це призводить до накопичення величезної ЕРС на відкритих спаях котушки реле. Діод блокує потік зворотної ЕРС, тим самим захищаючи схему.

Водяні насоси підключаються до вторинного кола реле. Дане коло, окрім безпосередньо самого насосу, включає живлення 12 В. Наявність окремих джерел живлення для кожного насосу продиктовано можливістю використання програмно-технічного пристрою на декількох ділянках ґрунту, що розташовуються один від одного на певній відстані (наприклад, окремі грядки).

Другу основну групу складають давачі вимірювання рівня вологості ґрунту. Для даного пристрою було обрано FC-28. Даний давач вимірює об'ємний вміст води в ґрунті та представляє значення вологості в напрузі. Він має як аналоговий так і цифровий вихід, проте для більш точного вимірювання показнику вологості слід використати аналоговий вхід. Проте, в одноплатній комп'ютерній системі Raspberry Pi не має вбудованих аналогових входів, як до прикладу у мікроконтролерній платі Arduino. Тому з метою, отримання аналогових даних від давача вологості ґрунту також було використано мікросхему АЦП ADS1115. Дана мікросхема (на схемі має позначення U2) має 4 аналогових виходи, із яких було задіяно три, і до яких під'єднанні давачі визначення вологості ґрунту. Живлення мікросхеми ADS1115 забезпечується 3.3 В, що отримуються від Raspberry Pi. Контакти ADS1115 SDA та SCL через pull-up реєстри під'єднуються до пінів Raspberry Pi GPIO2 та GPIO3 відповідно. Контакт ADDR з'єднано із контактом заземлення на мікросхемі. Таке з'єднання дозволяє надати АЦП ADS1115 адресу на I2C шині 0x48.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк. 41
Зм..	Арк.	№докум.	Підпис	Дата		

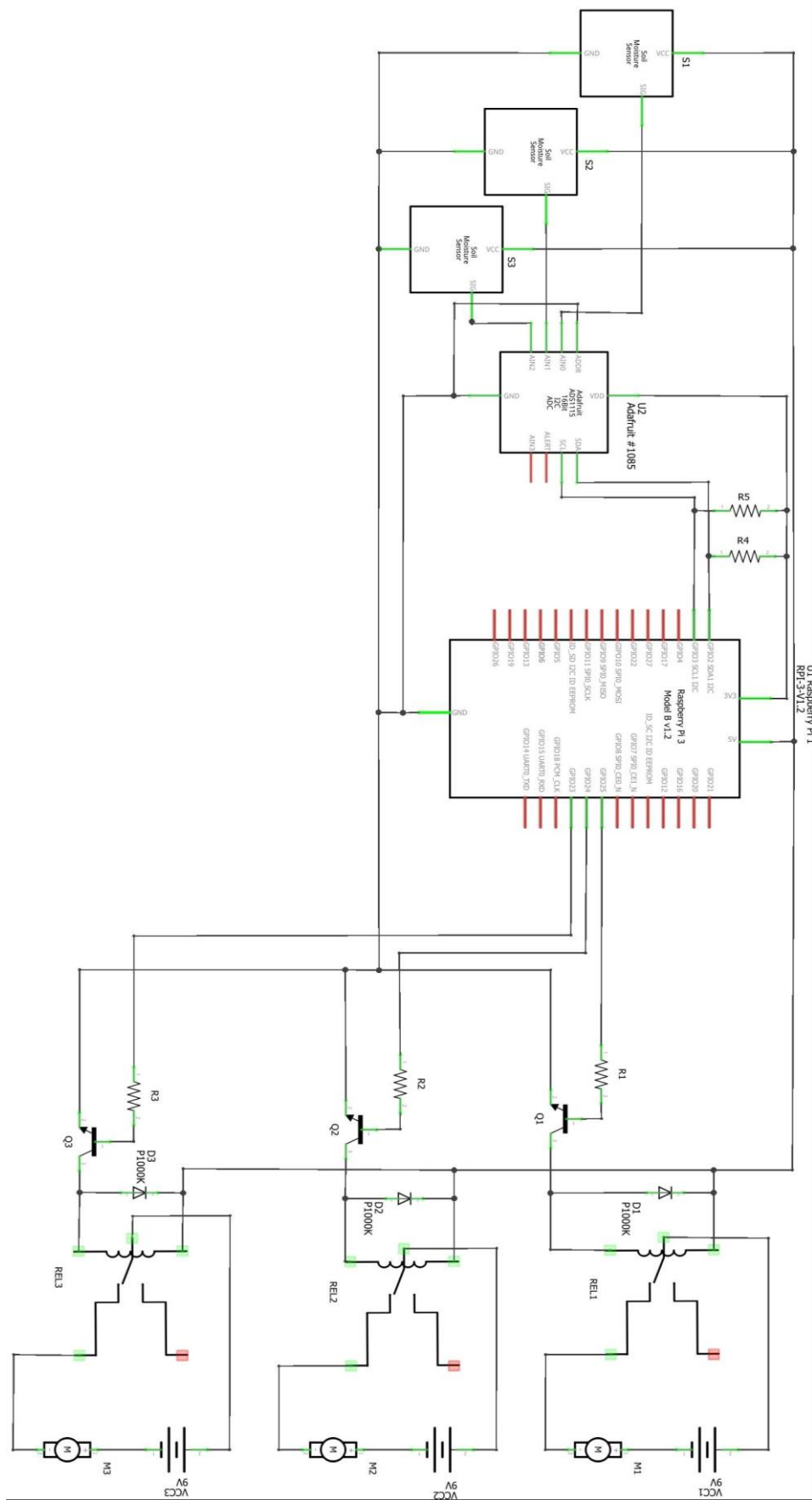


Рисунок 3.7 – Схема електрична принципова проєктованого програмно-технічного пристрою розумного поливу на базі Raspberry Pi

Зм.	Арк.	№докум.	Підпис	Дата
-----	------	---------	--------	------

### 3.4 Реалізація функцій програмно-технічного пристрою розумного поливу на базі Raspberry Pi

З огляду на поставлені вимоги, в програмно-технічного пристрою повинні бути реалізовані наступні функційні вимоги:

1. Нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту. Дана функція повинна виконуватись на регулярній основі, постійно нагадуючи власнику програмно-технічного засобу про доцільність перевірки рівня води. Слід відзначити, що безпосередньо значення рівня води не вимірюється, а контроль за наявністю води покладений на користувача.

2. Визначення вологості ґрунту та виконання автоматичного поливу. Реалізація даного функціоналу передбачає постійний моніторинг рівня вологості ґрунту на декількох ділянках (під ділянкою мається на увазі окремий горщик для рослини або грядка). У випадку зниження рівня вологості ґрунту на ділянці програмно-технічний пристрій повинен виконати полив відповідної ділянки ґрунту.

Розглянемо детальніше реалізацію кожної функції.

Розпочнемо створення програмної складової програмно-технічного пристрою із реалізації функціоналу, що оперує водяними насосами. Для цього створимо файл `Hardware.py`, а в ньому клас `Relay`. В середині класу у масиві `__relayPins` зберігається інформація про піни до яких під'єднані всі три водяні насоси. Масив `__relayStates` зберігає інформацію про поточний стан всіх водяних насосів, причому, оскільки насоси під'єднуються до реле до контакту NC (normally close), то увімкнення насосу здійснюватиметься по низькому рівню, тому в початковому стані на всі три контакти, що керують насосами подано високий рівень напруги. Основними функціями класу `Relay` є реалізація увімкнення та вимкнення реле, що представлені методами `on` та `off` відповідно:

```
def on(self, relayNumber):  
    self.setRelayState(relayNumber, False) #on
```

Зм.	Арк.	№докум.	Підпис	Дата

```

GPIO.output(self.__relayPins[relayNumber],self.getRelayState(relayNu
mber))

        print("relay {0} on
{1}".format(relayNumber,self.getRelayState(relayNumber)))

    def off(self,relayNumber):
        self.setRelayState(relayNumber,True)#off

GPIO.output(self.__relayPins[relayNumber],self.getRelayState(relayNu
mber))

        print("off
{0}".format(self.getRelayState(relayNumber)))

```

Зміна значень у масиві `__relayStates` реалізується за допомогою методу `setRelayState(self,relayNumber,relayState)`, тоді як отримання стану насосу – `getRelayState(self,relayNumber)`.

Таким чином остаточний вигляд класу `Relay` представлено наступним чином:

```

class Relay():
    __relayStates = [True,True,True]#off
    __relayPins = [34,35,36]

    GPIO.setmode(GPIO.BOARD)
    for i in range(0,3):
        GPIO.setup(__relayPins[i],GPIO.OUT)

    def __init(self):
        for i in range(0,3):
            #relays port to out

GPIO.output(self.__relayPins[i],self.__relayState[i])
            #turn off relay

```

```

        print("Relay          construct          with          state
{0}").format(self.__relayStates[i]))

    def getRelayState(self, relayNumber):
        return self.__relayStates[relayNumber]

    def printRelaysSatate(self):
        for i in range(0,3):
            print(self.__relayStates[i])

    def setRelayState(self, relayNumber, relayState):
        self.__relayStates[relayNumber]=relayState

    def on(self, relayNumber):
        self.setRelayState(relayNumber, False) #on

#GPIO.output(self.__relayPins[relayNumber], self.getRelayState(relayN
umber))

        print("relay          {0}          on
{1}").format(relayNumber, self.getRelayState(relayNumber))

    def off(self, relayNumber):
        self.setRelayState(relayNumber, True) #of

#GPIO.output(self.__relayPins[relayNumber], self.getRelayState(relayN
umber))

        print("off
{0}").format(self.getRelayState(relayNumber))

```

Для реалізації функцій роботи з часом реалізуємо клас TimeKeeper, що розмістимо у файлі TimeKeeper.py. Даний клас зберігає стан двох полів: поточний час current\_time та час, коли було здійснено останній полив time\_last\_watered. Основними функціями даного класу є реалізація методів встановлення поточного часу set\_current\_time(self, updated\_time), встановлення часу останнього поливу

					КВРКІ. 2001128.20.01.07 ПЗ	Арк. 45
Зм..	Арк.	№докум.	Підпис	Дата		

set\_time\_last\_watered та отримання поточного часу get\_current\_time()). Таким чином остаточний вигляд класу TimeKeeper представлено наступним чином:

```
import datetime
class TimeKeeper:
    def __init__(self, current_time):
        self.current_time = current_time
        self.time_last_watered = None

    def set_current_time(self, updated_time):
        self.current_time = updated_time

    def set_time_last_watered(self, updated_time):
        self.time_last_watered = updated_time

    @staticmethod
    def get_current_time():
        now = datetime.datetime.now()
        return now.strftime("%I:%M:%S %p")
```

Останні кроком реалізації проектованого програмно-технічного пристрою є створення головного файлу, у якому буде здійснено створено функції керування розумним поливом. Особливістю роботи програмно-технічного пристрою є відправка сповіщення на електронну пошту користувача про здійснений полив ділянки ґрунту, а також повідомлення-нагадування про необхідність доливу води у резервуар, з якого насоси забирають воду. Для реалізації такої функціональності використано модуль smtplib. Модуль smtplib в Python надає можливість надсилати електронні листи за допомогою протоколу SMTP (Simple Mail Transfer Protocol). Він дозволяє з'єднатися з SMTP-сервером, аутентифікуватися зі своїми обліковими даними (якщо потрібно) та відправляти електронні повідомлення.

Базовий шаблон для відправки електронного повідомлення за допомогою smtplib може бути представлений наступним чином:

```
import smtplib
```

Зм.	Арк.	№докум.	Підпис	Дата

```

HOST = "mySMTP.server.com"
SUBJECT = "Test email"
TO = "testUser1@adress.com"
FROM = " testUser2@adress.com "
text = "Hello"

BODY = "\r\n".join((
    "From: %s" % FROM,
    "To: %s" % TO,
    "Subject: %s" % SUBJECT ,
    "",
    text
))

server = smtplib.SMTP(HOST)
server.sendmail(FROM, [TO], BODY)
server.quit()

```

Реалізація алгоритму визначення вологості ґрунту та виконання автоматичного поливу із надсилання повідомлення на пошту передбачає визначення поточного часу за допомогою методу класу TimeKeeper.get\_current\_time() та визначення необхідності перевірки вологості ґрунту, шляхом зіставлення поточного часу із заданим часом перевірки. Якщо настав час перевірки, здійснюється отримання у циклі даних із кожного давача вимірювання вологості ґрунту. Якщо значення рівня вологості для і-го давача більше за задане порогове значення, здійснюється полив відповідної і-ї ділянки ґрунту, оновлення часу останнього поливу та надсилання на електронну пошту листа із зазначенням часу поливу та номеру ділянки, на якій було здійснено полив.

На рисунку 3.8 наведено блок-схему алгоритму визначення вологості ґрунту та виконання автоматичного поливу із надсилання повідомлення на пошту



Рисунок 3.8 – Блок-схема алгоритму визначення вологості ґрунту та виконання автоматичного поливу із надсилання повідомлення на пошту

Ще однією функцією програмно-технічного пристрою є надсилання листа-нагадування про необхідність поповнення води в резервуарі. Даний сценарій передбачає створення функції `check_if_need_send_remind_leter()`. Алгоритм роботи даної функції передбачає перевірку поточного часу із заданим часом-константою. Якщо час співпадає – здійснюється надсилання листа-нагадування.

На рисунку 3.9 наведено блок-схему алгоритму нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту.



Рисунок 3.9 – Блок-схема алгоритму нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту

Для виконання реалізованих раніше функцій із заданою періодичністю використано модуль `Schedule`. Модуль `Schedule` – це планувальник у процесі для періодичних завдань, які використовують для конфігурації шаблону конструктора. Розклад дозволяє запускати функції Python (або будь-які інші виклики) періодично через заздалегідь визначені проміжки часу за допомогою простого, зручного для людини синтаксису. Бібліотека розкладів

використовується для планування завдання на певний час кожного дня або певного дня тижня. Є можливість також встановити час у 24-годинному форматі. Фактично бібліотека Schedule узгоджує системний час із запланованим часом, що був встановлений. Коли запланований час і системний час збігаються, викликається функція завдання (командна функція, яка запланована).

Для демонстрації роботи реалізованого програмно-технічного пристрою розумного поливу було встановлено періодичність визначення вологості ґрунту та виконання автоматичного поливу із надсилання повідомлення на пошту кожного дня о 10.35.10, а нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту здійснювалось кожного четверга о 10.35.30. Весь код керування був огорнутий у блок try-except-finally. Робота пристрою продовжуватиметься до тих пір поки не буде натиснуто комбінацію клавіш Ctrl+C. Після цього буде здійснено скидання GPIO портів одноплатної комп'ютерної системи Raspberry Pi до початкового стану:

```
schedule.every().days.at("10:35:10").do(main)
schedule.next_run()
schedule.every().thursday.at("10:35:30").do(check_if_need_send_
remind_leter)

try:
    while 1:
        schedule.run_pending()
        time.sleep(1)
except KeyboardInterrupt:
    print("Ctrl+C")
finally:
    GPIO.cleanup()
    print("finish")
```

На рис. 3.10 наведено блок-схему алгоритму планування.

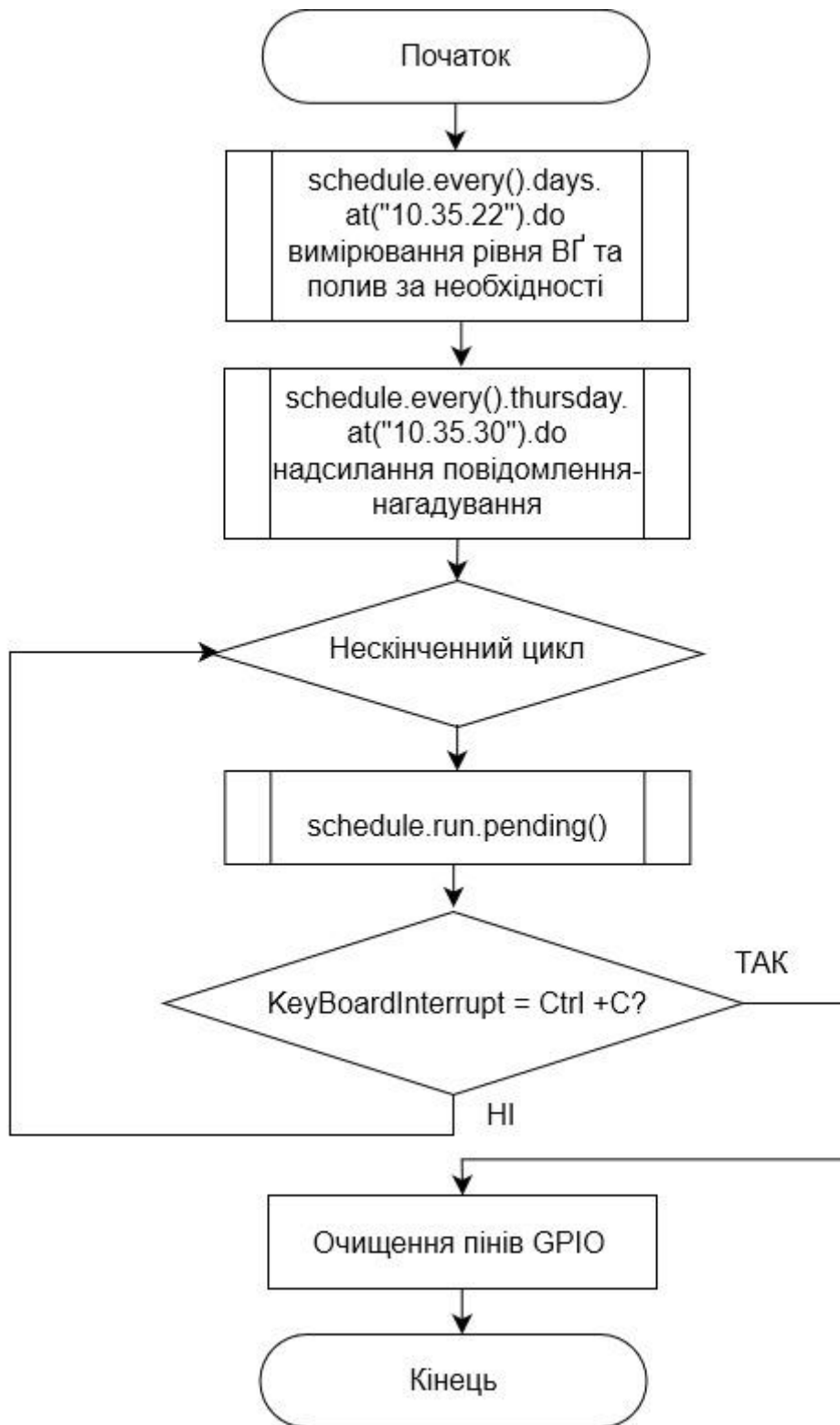


Рисунок 3.10 – Блок-схема алгоритму планування

Таким чином весь код файлу watering.py подамо у наступному лістингу:

```

import schedule
from datetime import datetime, date
import time
  
```

```

import RPi.GPIO as GPIO
import Hardware
import TimeKeeper as TK
import smtplib
import Adafruit_ADS1x15

# WATERING_TIME must be in "00:00:00 PM" format
WATERING_TIME = '10:35:10 AM'
REMIND_TIME = '10:35:30 AM'
SECONDS_TO_WATER = 3
RELAY = Hardware.Relay()
EMAIL_MESSAGES = {
    'last_watered': {
        'subject': 'Raspberry Pi: Plant Watering Time',
        'message': 'Your plant was last watered at'
    },
    'check_water_level': {
        'subject': 'Raspberry Pi: Check Water Level',
        'message': 'Check your water level!',
    }
}

adc = Adafruit_ADS1x15.ADS1115()
GAIN = 1
thresholdSoilMoisture = [300,400,300]
currentSoilMoisture = []

def send_email(message):
    port = 587
    smtp_server = "smtp.gmail.com"
    FROM = TO = "user1@gmail.com"
    password = "hnjktppghdwqtryu"

    mailserver=smtplib.SMTP(smtp_server, port)
    mailserver.starttls()
    mailserver.login(FROM, password)
    mailserver.sendmail(FROM, TO, message)

```

```

print("sent!")
mailserver.quit()

def send_last_watered_email(time_last_watered,pumpNumber):
    message = EMAIL_MESSAGES['last_watered']['message']
    subject = EMAIL_MESSAGES['last_watered']['subject']
    complete_message = '' + time_last_watered + '' + 'with
pump' + pumpNumber
    complete_message = "Subject: {}\n\n{} {}".format(subject,
message, time_last_watered)
    print(time_last_watered)
    send_email(complete_message)

def send_check_water_level_email():
    message = EMAIL_MESSAGES['check_water_level']['message']
    subject = EMAIL_MESSAGES['check_water_level']['subject']
    complete_message = ''
    complete_message = "Subject:  {}\n\n{}".format(subject,
message, REMIND_TIME)
    send_email(complete_message)

def water_plant(relay, relayNumber, seconds):
    relay.on(relayNumber)
    print("Plant is being watered with pump
{}!".format(relayNumber))
    time.sleep(seconds)
    print("Watering is finished!")
    relay.off(relayNumber)

def main():
    GAIN = 1
    time_keeper =
TK.TimeKeeper(TK.TimeKeeper.get_current_time())
    if(time_keeper.current_time == WATERING_TIME):

```

```

        for i in range(0,3):
            currentSoilMoisture[i] = adc.read_adc(i, gain=GAIN)
            if currentSoilMoisture[i] >=
thresholdSoilMoisture[i]:
                water_plant(RELAY, i, SECONDS_TO_WATER)

time_keeper.set_time_last_watered(TK.TimeKeeper.get_current_time())
            print("\nPlant was last watered at {0} with
pump {1}".format(time_keeper.time_last_watered,i))

send_last_watered_email(time_keeper.time_last_watered,i)

def check_if_need_send_remind_leter():
    time_keeper =
TK.TimeKeeper(TK.TimeKeeper.get_current_time())
    if(time_keeper.current_time == REMIND_TIME):
        send_check_water_level_email()
        print("send remind")

schedule.every().days.at("10:35:10").do(main)
schedule.next_run()
schedule.every().thursday.at("10:35:30").do(check_if_need_send_
remind_leter)

try:
    while 1:
        schedule.run_pending()
        time.sleep(1)
except KeyboardInterrupt:
    print("Ctrl+C")
finally:
    GPIO.cleanup()
    print("finish")

```

Результати роботи програмно-технічного пристрою у вигляді надісланих повідомлень на електронну пошту наведено на рис. 3.11 та рис. 3.12.

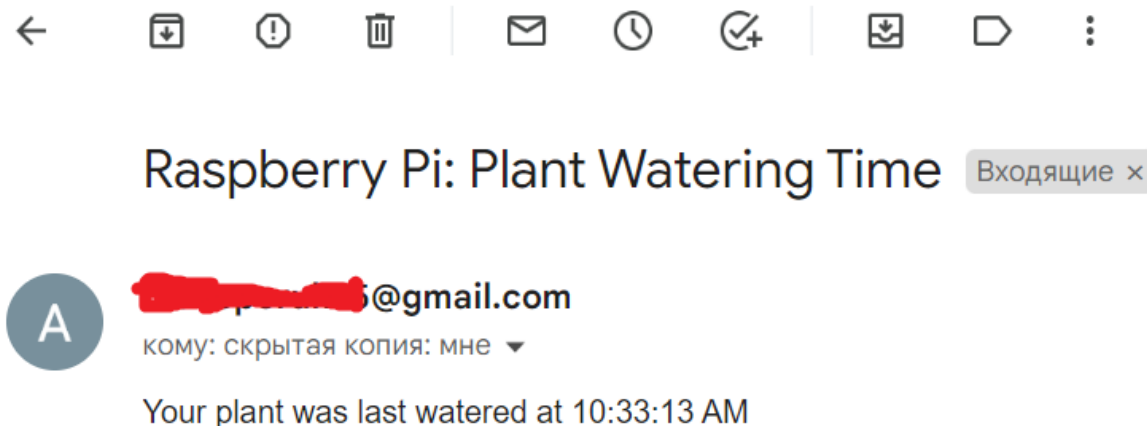


Рисунок 3.11 – Отримане повідомлення від програмно-технічного пристрою розумного поливу про здійснений полив ділянки ґрунту

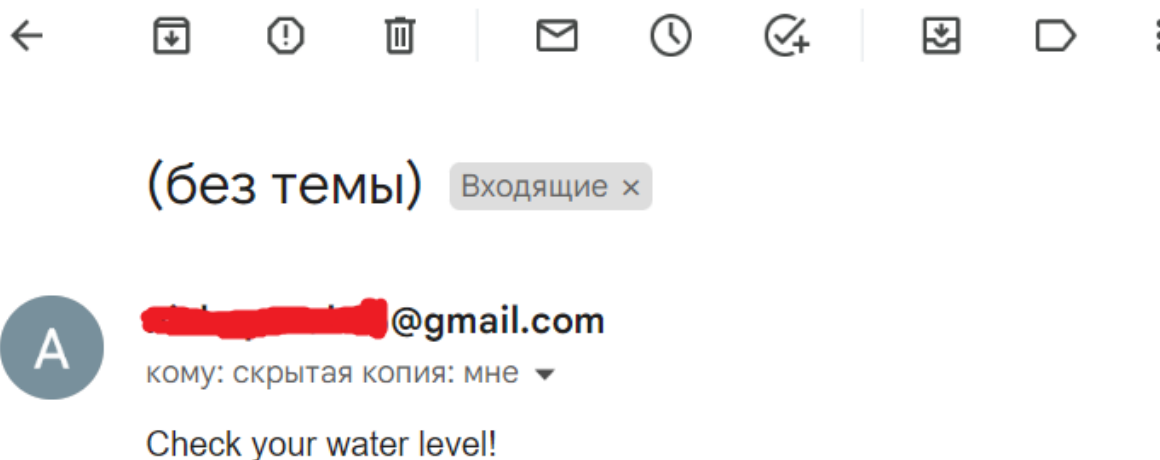


Рисунок 3.12 – Отримане повідомлення від програмно-технічного пристрою розумного поливу про необхідність перевірки рівня води в резервуарі

Слід відзначити, що запропонована програмна реалізація програмно-технічного пристрою дозволяє працювати тільки з акаунтами Google, оскільки в якості поштового сервера використано smtp.gmail.com (проте, є можливість роботи і з іншими поштовими серверами). Також слід відзначити, що для


можливості отримання листів на електронну скриньку Google потрібно дозволити доступ ненадійним додаткам, у яких є доступ до акаунту. Для цього слід встворити пароль для застосунку, виконавши наступні кроки:

- Перейти до Керування акантом Google.
- Вибрати вкладку Безпека і у секції Вхід у акаунт Google увімкнути двохфакторну аутентифікацію.
- Виконати вказівки мастера та запам'ятати згенерований пароль (даний пароль використовується для входу до акаунту – у файлі watering.py password = "hnjktppghdwqtryu").
- Далі у тій самій секції Вхід у акаунт вибрати паролі застосунків. У полі застосунок вибрати пошта, а у полі пристрій вибрати Інше та ввести Raspberry Pi. В результаті має бути додано дозвіл на надсилання листів від Raspberry Pi (рис. 3.13).

## ← Пароли приложений

Пароли приложений позволяют входить в аккаунт Google на устройствах, которые не поддерживают двухэтапную аутентификацию. Такой пароль достаточно ввести один раз. [Подробнее...](#)

Ваши пароли приложений

Название	Дата создания	Дата последнего использования	
Raspberry Pi	10:12	10:15	

Выберите приложение и устройство, для которых нужно создать пароль приложения.

Приложение ▼ Устройство ▼

Рисунок 3.13 – Створення паролю для застосунку

### 3.5 Висновки за розділом 3

Таким чином приведено монтажну схему та схему електричну принципову програмно-технічного пристрою розумного поливу. Для програмної реалізації функцій запропонованого програмно-технічного пристрою виконано встановлення та налаштування необхідного програмного забезпечення. Реалізовано функції нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту, а також визначення вологості ґрунту та виконання автоматичного поливу. Особливістю даного програмно-технічного пристрою є можливість автоматизованого поливу на декількох ділянках ґрунту (по декілька датчиків вимірювання рівня вологості ґрунту та водяних насосів), що може значно розширити сферу застосування даного пристрою. Ще однією важливою перевагою розробленого пристрою є досить низька вартість його компонентів.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						57
Зм.	Арк.	№докум.	Підпис	Дата		

## ВИСНОВКИ

Однією з головних переваг автоматичних систем поливу ґрунту є економія води. Це досягається завдяки тому, що системи поливу можуть бути налаштовані на автоматичне регулювання часу та кількості води, яка використовується для поливу. Це дає змогу оптимізувати процес поливу, зменшити кількість води, яка витрачається на полив та, як результат, зменшити витрати на воду.

Також автоматичні системи поливу дозволяють зберегти час та зусилля фермерів та господарів приватних ділянок, оскільки системи можуть працювати без необхідності постійного контролю та управління. Це дає змогу використовувати час та ресурси більш ефективно, зосереджуючись на інших аспектах діяльності.

В результаті виконання кваліфікаційної роботи було спроектовано та реалізовано систему розумного поливу на базі Raspberry Pi. Спроектована система забезпечує періодичний моніторинг вологості ґрунту на декількох ділянках, здійснює полив та виконує функції нагадування та сповіщення про виконанні дії шляхом надсилання повідомлень на електронну пошту користувача.

У першому розділі розглянуто прилади для вимірювання вологості ґрунту та проведено огляд відомих підходів до проектування систем автоматизованого поливу. Визначено основні складові компоненти із яких складаються пристрою автоматизованого та розумного поливу.

У другому розділі було визначено функційні та нефункційні вимоги до проектованого програмно-технічного пристрою, що передбачає виконання функцій нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту, а також визначення рівня вологості ґрунту та здійснення автоматичного поливу. При проектуванні пристрою основна увага була приділена можливості виконання функцій вимірювання і поливу на декількох (трьох) ділянка, а не на одній. Також запропоновано структуру програмно-

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		58

технічного пристрою та проведено аналіз апаратних та програмних компонентів.

У третьому розділі приведено монтажну схему та схему електричну принципову програмно-технічного пристрою розумного поливу. Для програмної реалізації функцій запропонованого програмно-технічного пристрою виконано встановлення та налаштування необхідного програмного забезпечення. Реалізовано функції нагадування про рівень води в резервуарі у формі надсилання листа на електронну пошту, а також визначення вологості ґрунту та виконання автоматичного поливу. Особливістю даного програмно-технічного пристрою є можливість автоматизованого поливу на декількох ділянках ґрунту (по декілька датчиків вимірювання рівня вологості ґрунту та водяних насосів), що може значно розширити сферу застосування даного пристрою. Ще однією важливою перевагою розробленого пристрою є досить низька вартість його компонентів.

					КВРКІ. 2001128.20.01.07 ПЗ	Арк.
						59
Зм.	Арк.	№докум.	Підпис	Дата		

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. 3 Ways to Automatic Plan Irrigation System using Microcontroller, URL: <https://www.elprocus.com/microcontroller-based-automatic-irrigation-system/>
2. Introduction of Automatic Irrigation Systems for Tree Fruit Orchards, URL <https://extension.psu.edu/introduction-of-automatic-irrigation-systems-for-tree-fruit>
3. Karpagam G.R., Kumar B.V., Maheswari J.U., Gao X.-Z. Smart Cyber Physical Systems Chapman and Hall, CRC, 2020 – 294 p.
4. Sanil P. Raspberry Pi based energy management system. Artificial Intelligence Evolution, *Institute of Computer Engineering*. 2020. Vol 1(2). 63 – 144.
5. Joan M., Shahriar Alamgir Md., Mahabubur Rahman Md., Bhuiyan M. Raspberry Pi projects. International Journal of Engineering & Technology, *International Journal of Engineering And Science*, Vol. 10(2), 2019, 1-12.
6. Katravath R. et al Fire Alarm Robot and Authentication System Using Raspberry Pi and Cloud, *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol. 8, Iss. 4S2, 2019. 256-259
7. Глибовець А.М., Моголівський В.О. Аналіз систем підтримки розумного будинку. *Control systems and computers*. 2019. No 5(283). 30–37.
8. Arduino ua, Модуль давача якості повітря MQ135, URL: <https://arduino.ua/ru/prod1201-modul-datchika-kachestva-vozdyha-mq135>
9. Buniyamin N. Development of Fire Alarm System using Raspberry Pi and Arduino Uno, *Conference: International Conference On Electrical, Electronics and Systems Engineering (ICEESE)*, December 2013. 37-42.
10. Brandt P., Brandt D. Science and Engineering Projects Using the Arduino and Raspberry Pi. Journal of Engineering, *Automatic Control And Information Sciences*. Vol. 31(2), 2021. 1-97.
11. Bhuvanewari S. Fire Detection Using Raspberry Pi, *International Journal of Electrical Engineering and Technology*, Vol. 12, № 1, 2022, 73-80.
12. Конспект лекцій з дисципліни «Комп'ютерні системи» для студентів напряму підготовки «Комп'ютерна інженерія» / І. М. Лазарович. – Івано-





*International Conference of Smart Systems and Emerging Technologies (SMARTTECH)*, 2020, 155-160, doi: 10.1109/SMARTTECH49988.2020.00046.

30. Khan M. N. A., Tanveer T., Khurshid K., Zaki H. and Zaidi S. S. I., Fire Detection System using Raspberry Pi, *2019 International Conference on Information Science and Communication Technology (ICISCT)*, 2019, 1-6, doi: 10.1109/CISCT.2019.8777414.

31. Проєктування комп'ютеризованих систем управління: Опорний конспект лекцій. – Тернопіль, THEU. URL: [http://dspace.tneu.edu.ua/retrieve/52377/Лекції\\_ПКСУ.pdf](http://dspace.tneu.edu.ua/retrieve/52377/Лекції_ПКСУ.pdf).

32. Тарарака В.Д. Архітектура комп'ютерних систем: навч. посіб, Житомир: ЖДТУ, 2018. 383 с.

33. Raspberry Pi, URL: <https://www.raspberrypi.org/>

34. Момот Т.В., Мураєв Є.В. Компаративний аналіз зарубіжних практик розвитку розумних міст та можливості їх імплементації в Україні. *Електронний науково-практичний журнал «Інфраструктура ринку»*. 2020. Вип. 42. 232–237.

35. Nisan N., Schocken S. The Elements of Computing Systems, second edition: Building a Modern Computer from First Principles 2nd Edition, *The MIT Press*, 2021.

36. Yadin A. Computer Systems Architecture. Chapman and Hall, *CRC*, 2016. 467 p.

37. Null L., Lobur Y. Essentials of Computer Organization and Architecture, *Jones & Bartlett Learning*; 5th edition, 2018. 744 p.

38. Kravets A.G., Bolshakov A.A., M.V. Shcherbakov Cyber-Physical Systems: Industry 4.0 Challenges (Studies in Systems, Decision and Control, 260) , *Springer*; 1st ed., 2020. 349 p.

39. Rea P., Ottaviano E., Machado J. and Antosz K. Design, Applications, and Maintenance of Cyber-Physical Systems, *Engineering Science Reference*, 2021. 314 p. DOI: 10.4018/978-1-7998-6721-0

40. Степаненко О.І. Пасивний будинок – шлях до ефективного використання енергії, *Енергетика: економіка, технології, екологія*. 2014. №3.

41. Гайдукевич С.В., Семенова Н.П., Леськів Я.А. Особливості SMART-технологій на прикладі автоматизації житлового будинку, *Таврійський науковий вісник*. №1. 2022. 12-21.

42. Rhee J.H., Ma J.H., Seo J., Cha S.H., Review of applications and user perceptions of smart home technology for health and environmental monitoring, *J. Comput. Des. Eng.* No 9. 2022, 857–889

43. Бабенко О. В. Актуальність технологій розумних будинків для підвищення енергоефективності економіки держави / О. В. Бабенко, М. С. Омелянчук, *Матеріали XLVIII наук.-техн. конф. підрозділів Вінниць. нац. техн. ун-ту (НТКП ВНТУ-2019)* : зб. доп. Вінниця, 2019. 2920-2921

44. Li B. S. X., Wan B., Wang C., Zhou X., Chen X. Definitions of predictability for cyber physical systems, *J. of Systems Architecture*. 2016. DOI: 10.1016/j.sysarc.2016.01.007.

45. Poliakov, M., Larionova, T. Control Systems with programmable logic controllers, Remote and virtual tools in engineering: textbook, *general editorship Dr.Ing.Karsten Henke*. Zaporizhzhya: Dike Pole, 2016. 250 p.

46. С. Г. Натрошвілі, Г. Р. Натрошвілі, Т. Г. Бабина, Б. М. Злотенко, Т. І. Кулік Комп'ютерно-інтегрована система керування природним і штучним освітленням розумного будинку, *Вісник Хмельницького національного університету. Серія : Технічні науки*. 2020. № 5 (289). 65-71.

47. Monk S. Programming Arduino Next Steps: Going Further with Sketches . *McGraw-Hill Education TAB*, 2018. 320 p.

48. Barrett S.F. Microchip AVR® Microcontroller Primer: Programming and Interfacing , *Morgan & Claypool Publishers*, 2019. 374 p.

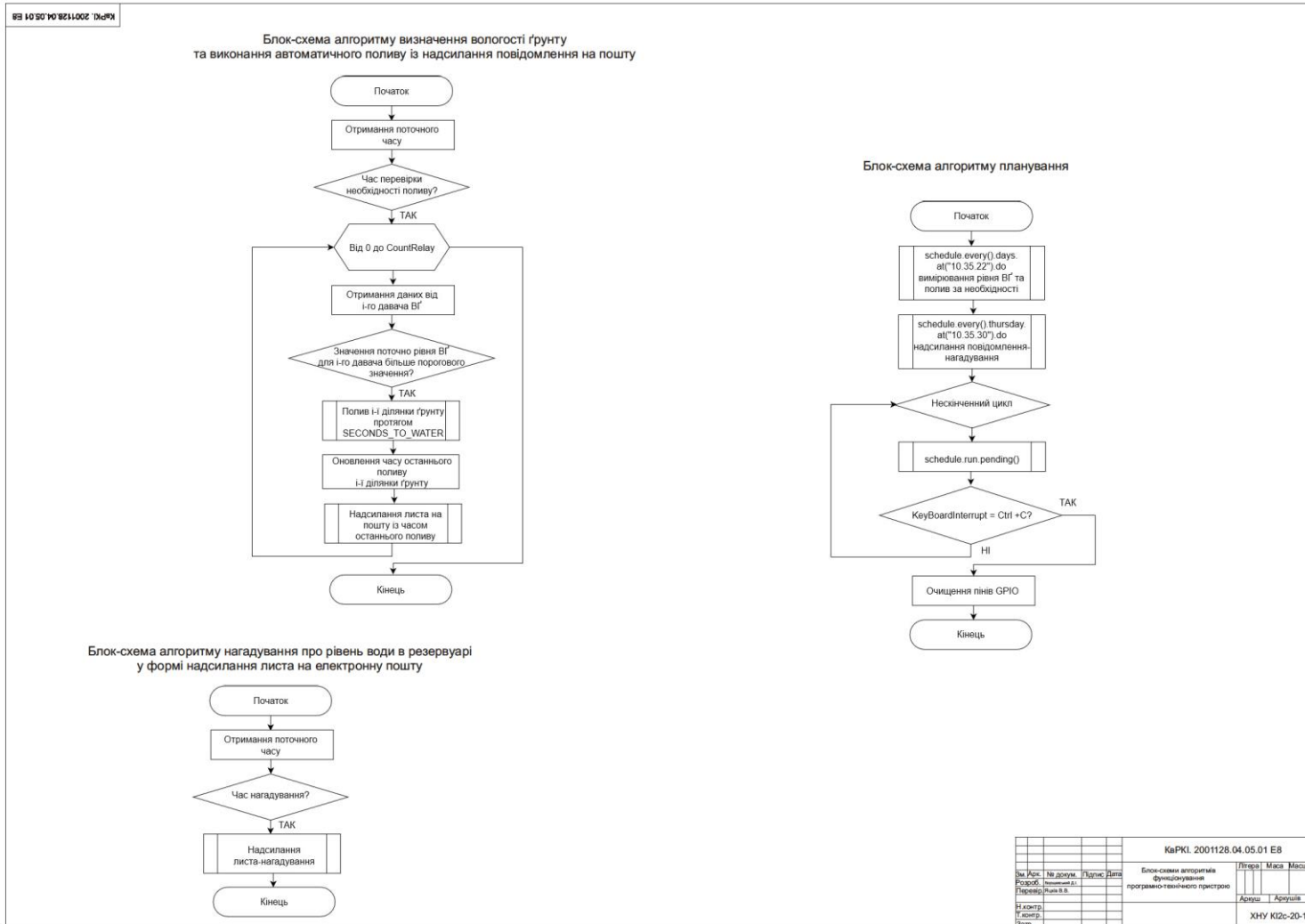
49. Hu Y., Tilke D., Adams T. et al. Smart home in a box: usability study for a large scale self-installation of smart home technologies. *J Reliable Intell Environ* 2. 2016, 93-106

50. Степаненко О.І. Пасивний будинок – шлях до ефективного використання енергії. *Енергетика: економіка, технології, екологія*. 2014. №3.56-58.

51. Barrett S.F. Microchip AVR® Microcontroller Primer: Programming and Interfacing, *Morgan & Claypool Publishers*, 2019. 374 p.
52. Kravets A.G., Bolshakov A.A., M.V. Shcherbakov Cyber-Physical Systems: Industry 4.0 Challenges (Studies in Systems, Decision and Control, 260), *Springer*; 1st ed., 2020. 349 p.
53. Papazoglou P. M. An Educational Guide to the AVR Microcontroller Programming: AVR Programming::Demystified (Assembly Language) (Volume 1) , *CreateSpace Independent Publishing Platform*, 2018. 274 p.
54. Василенко В. І., Ремізов І.А. Особливості побудови інтелектуальних енергетичних систем будівель та споруд. *Енергетичний менеджмент: стан та перспективи розвитку – PEMS : зб.матеріалів IV міжнар. наук.-техн. конф.*, 4–7 черв. 2019 р. К. : НТУУ «КПІ», 2019. 21-22.
55. Kishita Y., Mizuno Y., Fukushige S., Umeda Y. Scenario structuring methodology for computer-aided scenario design: An application to envisioning sustainable futures, *Technol. Forecast. Soc. Chang.* 2020. 160. 120207
56. Teslyuk, V., Kazarian, A., Kryvinska, N., Tsmots, I., Teslyuk, T. Automated synthesis method of smart home systems based on the architectural pattern redux. *CEUR Workshop Proceedings*, 2019. 2533. 58-69
57. Yadin A. Computer Systems Architecture, Chapman and Hall, *CRC*, 2016. 467 p.
58. Nisan N., Schocken S. The Elements of Computing Systems, second edition: Building a Modern Computer from First Principles 2nd Edition, *The MIT Press*, 2021. 344 p.
59. Balta-Ozkan N., Davidson R., Bicket M., Whitmarsh L., *Social barriers to the adoption of smart homes, Energy Policy*, Volume 63. 2013, 363-374
60. Котунова, Д. Г. Огляд DIY елементів для систем «Smart Home» / Д. Г. Котунова, О. М. Павловський, *XIII Науково-практична конференція студентів, аспірантів та молодих вчених «Погляд у майбутнє приладобудування»*, 13-14 травня 2020 р., м. Київ, Україна : збірник праць конференції. – Київ : КПІ ім. Ігоря Сікорського, 2020. – С. 35–38.

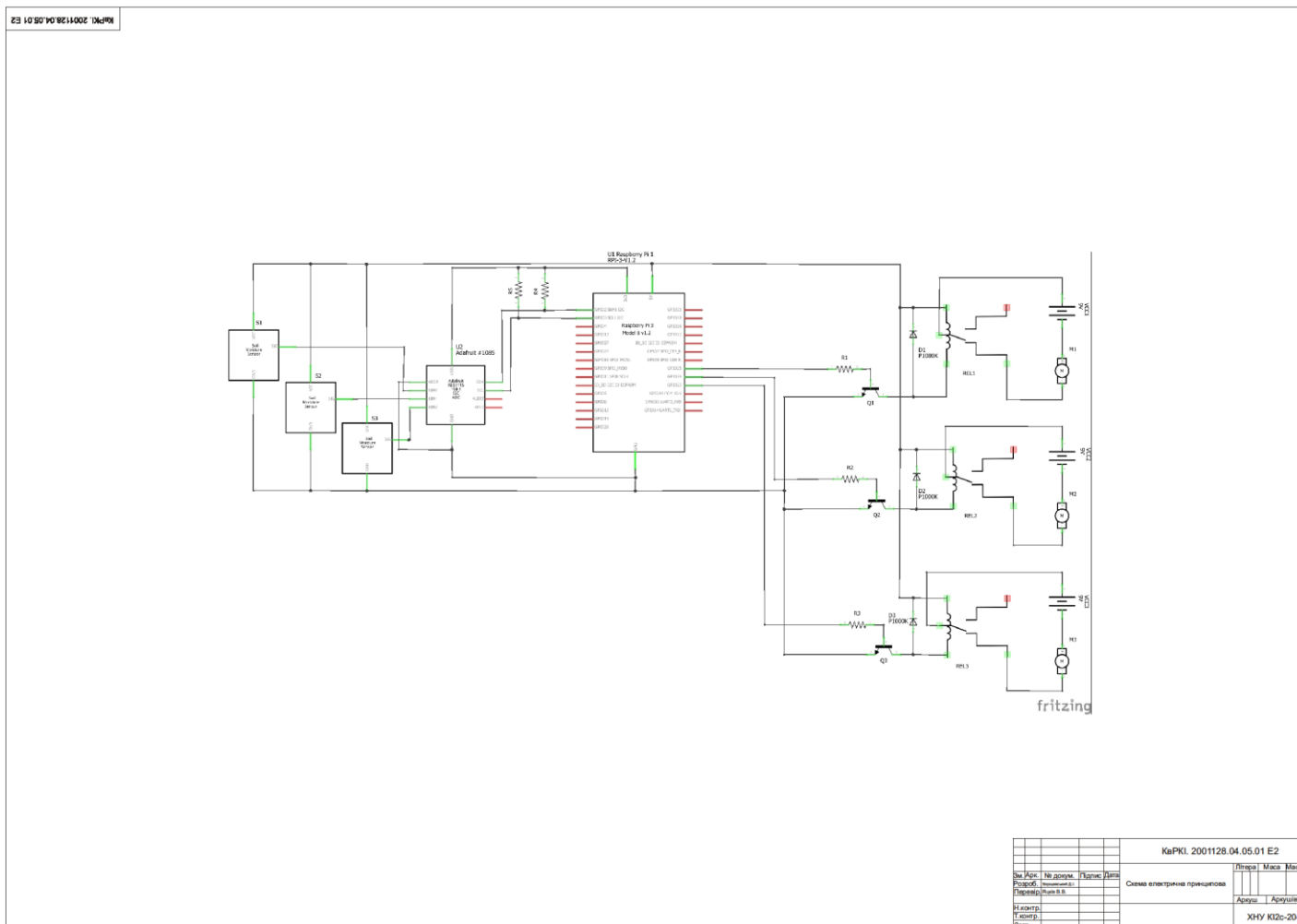
# ДОДАТОК А

## Копія креслення «Блок-схеми алгоритмів функціонування програмно-технічного пристрою»



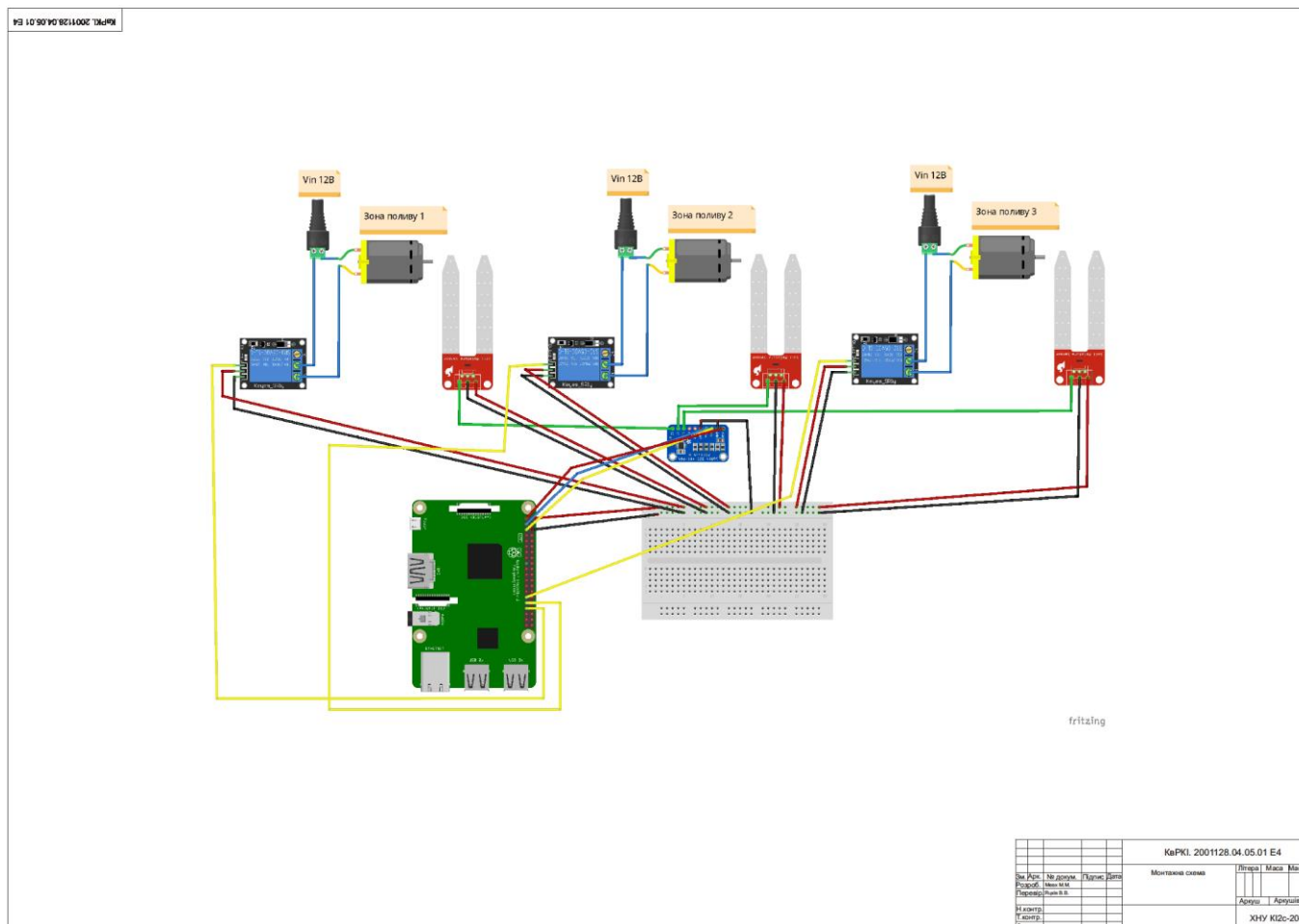
# ДОДАТОК Б

## Копія креслення «Схема електрична принципова»



# ДОДАТОК В

## Копія креслення «Монтажна схема»



Ім'я користувача:  
Кафедра КІ

Дата перевірки:  
04.05.2023 19:03:43 EEST

Дата звіту:  
04.05.2023 19:13:16 EEST

ID перевірки:  
1014924856

Тип перевірки:  
Doc vs Internet + Library

ID користувача:  
100005591

Назва документа: Борщевський Програмно-технічний пристрій розумного поливу на базі Raspberry Pi

Кількість сторінок: 68 Кількість слів: 10735 Кількість символів: 79278 Розмір файлу: 4.96 MB ID файлу: 1014619896

1691 слово позначене як "вилучене" та не враховується у підрахунку слів

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

## 12.1% Схожість

Найбільша схожість: 2.3% з Інтернет-джерелом (<https://uhbdp.org/article/priladi-dlya-vimiryuvannya-vologosti-gruntu...>)

11% Джерела з Інтернету 166 ..... Сторінка 70

4.28% Джерела з Бібліотеки 35 ..... Сторінка 71

## 0% Цитат

Цитати 1 ..... Сторінка 72

Не знайдено жодних посилань

## 0% Вилучень

Немає вилучених джерел

## Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи 3

Підозріле форматування 12 сторінок

## Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 10.0%

Словнича перевірка: en\_US, ru\_RU, ua\_UA. Помилки в документах: 13%

ID: 113023 Назва: БКР Програмно-технічний пристрій розумного пошуку на базі Raspberry Pi Додано в БД: 2023-05-04 Автор: Д.І. Борщевський Керівник: В.В. Яцків Консультанти: Оповісти:	Документ		Суцільний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	70261	580	11622 (17%)	112 (19%)

ID	Джерело плагіату	Наявність плагіату в документі	
		Символи	Лексеми
		112609	6992 (10.0%)

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Борщевський Денис Ігорович

Тема: Програмно-технічний пристрій розумного поливу на базі Raspberry Pi

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість листів креслень 3 ; кількість сторінок записки 55

1. Короткий зміст роботи та прийнятих рішень У роботі запропоновано програмно-технічний пристрій розумного поливу на базі Raspberry Pi

2. Висновок про відповідність роботи дипломному завданню \_\_\_\_\_  
Дипломний проект відповідає виданому завданню

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі проведено аналіз предметної області. У другому розділі спроектовано програмно-технічний пристрій розумного поливу на базі Raspberry Pi. У третьому розділі проведено реалізацію програмно-технічного пристрою розумного поливу на базі Raspberry Pi

4. Позитивні сторони роботи: Запропоновано структуру програмно-технічного пристрою розумного поливу на базі Raspberry Pi та здійснено його реалізацію.

5. Негативні сторони роботи: В роботі відсутні оцінки кількісних та якісних характеристик розробленого пристрою та не проведено оцінку вартості розробленого пристрою.

6. Оцінка графічного оформлення та пояснювальної записки роботи:  
пояснювальна записка та листи креслення виконані згідно діючих вимог

7. Відгук про роботу в цілому: В загальному робота виконана на достатньому рівні.

8. Інші зауваження: —

9. Оцінка дипломної роботи:

Розглянувши позитивні та негативні сторони представленої дипломної роботи вважаю, що робота заслуговує оцінки «добре» 3,75 (С)

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи)

к. т. н. доцент. кафедри ІТЗ Турман Жан  
Васильович

“ 5 ” червня 2023р.

JB TPP

Завідувачу кафедри КПС  
д-р.техн.наук, проф. Говорущенко Т. О.

Борщевський Денис Ігорович  
ПІБ здобувача вищої освіти

ФІТ, 3 курсу, групи К12с-20-1

### ЗАЯВА

З правилами чинного Положення «Про дотримання академічної доброчесності в Хмельницькому національному університеті» від 26.09.2020 (зі змінами від 26.11.2020), згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіатоповішений (а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

05.06.2023

дата



підпис

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ  
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ  
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Програмно-технічний пристрій розумного поливу на базі Raspberry Pi

Автор: Борщевський Денис Ігорович

Спеціальність: 123 – Компютерна інженерія

Освітня програма: освітньо-професійна

Науковий керівник: Яцків Василь Васильович, к.т.н, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укріплення запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

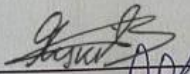
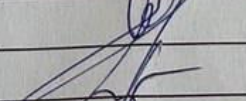
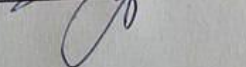
- 1) запозичення розміщені в розділі аналізу існуючих аналогів та відомих рішень, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) частина списку використаних джерел було визначено як плагіат.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 12,1% і адресується до 201 першоджерела, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІС

  
\_\_\_\_\_  
  
\_\_\_\_\_  
  
\_\_\_\_\_

В. В. Яцків

С. М. Лисенко

Т. О. Говорущенко