

Хмельницький національний університет  
Факультет програмування  
та комп'ютерних і телекомунікаційних систем  
Кафедра комп'ютерної інженерії та системного програмування

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр  
Освітній рівень

Програмована схема керування світлофорами на базі мікро-ЕОМ для  
навчального макета, що моделює ситуації на перехресті вулиці  
Назва теми

КвРКІ.170376.17.03.32 ПЗ  
Шифр

Галузь знань 12 «Інформаційні технології»

Шифр, назва

Спеціальність 123 «Комп'ютерна інженерія»

Шифр, назва

Освітня програма «Комп'ютерна інженерія»

Назва

Виконав: студент IV курсу, група КІ-17-Б

Підпис

В.Р. Кучерук

Ініціали, прізвище

Керівник

Підпис, дата

В.М. Стецюк

Ініціали, прізвище

Нормоконтролер

Підпис, дата

С.М. Лисенко

Ініціали, прізвище

До захисту допускаю:  
Зав. кафедри комп'ютерної  
Інженерії та системного  
Програмування

Підпис

Т.О. Говоруценко

Ініціали, прізвище

«14» червня 2021 р.

Хмельницький 2021

# ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет ПРОГРАМУВАННЯ ТА КОМП'ЮТЕРНИХ І ТЕЛЕКОМУНІКАЦІЙНИХ СИСТЕМ

Кафедра КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ

Освітній рівень БАКАЛАВР

Галузь знань 12 ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

Спеціальність 123 КОМП'ЮТЕРНА ІНЖЕНЕРІЯ

Освітня програма ОСВІТНЯ ПРОГРАМА «КОМП'ЮТЕРНА ІНЖЕНЕРІЯ»

ЗАТВЕРДЖУЮ

Зав. кафедри Т.О.Говорушченко

“И” 01 2021 р.

## ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Кучеруку Владиславу Руслановичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмована схема керування світлофорами на базі мікро-ЕОМ для навчального макета, що моделює ситуації на перехресті вулиці

Керівник проекту (роботи) Стецюк В.М.

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

Затверджена наказом ректора університету від 05.02.2021 р. № 11

2. Строк подання студентом проекту (роботи) на кафедру 07.06.2021 р.

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

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

Дослідження предметної області та постановка задачі

Аналіз аналогічних рішень Реалізація системи управління Алгоритм функціонування і

програмне забезпечення Матеріальні затрати

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

Структурна \_\_\_\_\_ схема \_\_\_\_\_ і \_\_\_\_\_ схема

функціонування(Е1) \_\_\_\_\_

Функціональна схема(Е2) \_\_\_\_\_

Принципова схема(Е3) \_\_\_\_\_

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання при
Нормоконтроль	Лисенко С.М., професор кафедри КІСП		
Антиплагіат	Нічепорук А.О., доцент кафедри КІСП		

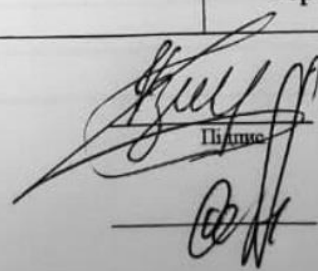
7. Дата видачі завдання « 11 » 01 2021 р.

**КАЛЕНДАРНИЙ ПЛАН**

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Прим
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	11.01.2021	викон
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2021	викон
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2021	викон
4	Робота над розділом 2 – проектування апаратної частини системи керування світлофорів	01.04.2021	викон
5	Робота над розділом 3 – проектування алгоритмів і програмної частини системи керування світлофорів	30.04.2021	викон
6	Оформлення пояснювальної записки згідно вимог	31.05.2021	викон
7	Попередній захист ВКР	02.06.2021	викон
8	Захист ВКР на засіданні ЕК	Червень 2021 року	

Студент

Керівник проекту (роботи)

  
Підпис

Кучерук В.Р.

Ініціали, прізвище

Стецюк В.М.



## АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмована схема керування світлофорами на базі мікро-ЕОМ для навчального макета, що моделює ситуації на перехресті вулиці».

Автор роботи: Кучерук Владислав Русланович.

Керівник роботи: Стецюк Василь Миколайович.

Пояснювальна записка: 57 с., 31 рис., 4 табл., 4 дод., 41 джерело.

Графічна частина: 10 презентаційних слайдів.

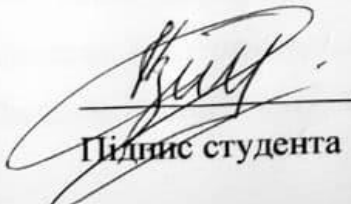
Адаптивна система керування світлофором, RASPBERRY PI, FSM, OpenCV, Python, Cvlib, вирівнювання гістограми, YOLO.

Мета роботи - розробка системи адаптивного керування світлофорами на перехресті. Для виконання роботи був проведений аналіз вже існуючих рішень адаптивних систем, їх переваг і недоліків.

Об'єктом дослідження є програмно-технічний засіб керування світлофорами на перехресті.

Предметом дослідження є процес керування світлофорами на перехресті.

Практичне значення має спроектована система керування світлофорами, яка за рахунок ідентифікації кількості транспорту на перехресті дозволить підвищити ефективність функціонування світлофорів. Адаптивна система дозволяє гнучко обчислювати час зеленого світла, який залежить від виявленої щільності руху на смугах руху.

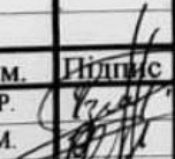
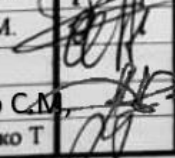

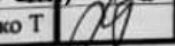
  
Підпис студента

17.06.21

Дата

## ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ.....	4
ВСТУП.....	5
1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ.....	7
1.1 Основні компоненти адаптивної системи управління світлофорами.....	7
1.2 Огляд систем керування світлофорами.....	9
1.3 Висновки.....	23
1.4 Постановка задачі.....	23
2 ПРОЄКТУВАННЯ ТЕХНІЧНОГО ЗАСОБУ.....	24
2.1 Вибір основних компонентів та програмного забезпечення.....	24
2.2 Структура та загальний схема функціонування системи.....	28
2.3 Апаратна реалізація системи управління світлофорами.....	30
2.4 Висновки.....	33
3 АЛГОРИТМ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ СВІТЛОФОРАМИ.....	35
3.1 Блок-схема функціонування системи.....	35
3.2 Операції з обробки зображень.....	36
3.3 Структура макетної транспортної розв'язки і розробка діаграми переходів станів контролера світлофору.....	41
3.4 Алгоритм розрахунку часу зеленого сигналу світлофора.....	47
3.5 Аналіз витрат проекту.....	49
3.6 Висновки.....	50
ВИСНОВКИ.....	52
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	54
Додаток А Фрагмент вихідного коду.....	54
Додаток Б Копія креслення «Схема електрична структурна».....	62

<b>КвРКІ.170376.17.03.32 ПЗ</b>								
Зм.	Арк.	№докум.	Підпис	Дата	Програмована схема керування світлофорами на базі мікро-ЕОМ для навчального макета, що моделює ситуації на перехресті вулиці. Пояснювальна записка	Літера	Аркуш	Аркушів
Виконав	Кучерук В.Р.						2	64
Перевір.	Стецюк В.М.							
Н.контр.	Лисенко С.М.			14.06				
Затвер.	Говорущенко Т					ХНУ, КІ-17-3		

Додаток В Копія креслення «Схема електрична функціональна» .....	63
Додаток Г Копія креслення «Схема електрична принципова» .....	64

					КВРКІ.170376.17.03.32 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		3

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ІЧ-давач – інфрачервоний давач.

ПЛІС – програмована логічна інтегральна схема.

RGB – колірна схема (червоний, зелений, синій).

ПК – персональний комп'ютер.

FSM – переходи станів кінцевого автомата.

					КВРКІ.170376.17.03.32 ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		4

## ВСТУП

Затори на дорогах є однією з головних проблем сучасної України і ця проблема стає все більш серйозною з кожним днем. Затори на дорогах трапляються, коли наявна пропускна спроможність не відповідає кількості транспорту. Ще деякі фактори викликають затори, такі як високі темпи зростання міського населення, швидке збільшення кількості автомобілів, неефективні стратегії будівництва доріг та управління, і особливо функціонування неадаптивних світлофорних систем, які застаріли і не здатні впоратись із високою щільністю руху.

Затори дорожнього руху призводять до численних негативних впливів на наше суспільство та навколишнє середовище, наприклад, втрата часу та затримки учасників дорожнього руху, що знижує продуктивність праці працівників, витрачаючи паливо та збільшуючи викиди парникових газів, що погіршує якість повітря та призводить до глобального потепління, заважає або блокує проїзд аварійних автомобілів. Для подолання наслідків заторів використання адаптивної системи керування світлофорами життєво важливе для забезпечення плавного руху транспортних засобів на транспортних маршрутах.

Незважаючи на те, що в традиційній системах керування світлофорами впроваджено багато функцій для покращення ефективності руху потоку та неефективність існуючих методів призводить до поширення заторів та збільшення дорожньо-транспортних пригод.

Однією з причин цієї проблеми дорожнього руху є неефективність методів та алгоритмів, що використовуються в існуючій системі світлофорів, які не здатні пристосуватися до постійно мінливої дорожньої ситуації, що в кінцевому підсумку призводить до поширення заторів та збільшення дорожньо-транспортних пригод. Тому, щоб запобігти подальшому погіршенню ситуації, існує нагальна потреба у впровадженні системи та технології, здатної вдосконалити поточну систему управління світлофорами.

Тобто, існує необхідність розробити функціонуючу в режимі реального часу розумну систему управління світлофором, яка здатна подавати належний

					КВРКІ.170376.17.03.32 ПЗ	Арк. 5
Зм.	Арк.	№докум.	Підпис	Дата		

синхронізуючий сигнал, який залежить від зміни щільності руху на транспортному розв'язку. Тобто система має виявляти та рахувати транспортні засоби та рахувати час сигналу відповідно до щільності руху.

					КВРКІ.170376.17.03.32 ПЗ	Арк.
						6
Зм.	Арк.	№докум.	Підпис	Дата		

# ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

## 1.1 Основні компоненти адаптивної системи управління світлофорами

Традиційні контролери світлофора використовували фіксований заздалегідь визначений графік руху для кожного напрямку на перехресті[1]. Управління виконував електромеханічний контролер, який складається з трьох основних частин - таймера циферблата, соленоїда та блоку кулачка. Двигун і редукторний вузол управляють таймером циферблата, який, у свою чергу, відповідає за включення або знеструмлення соленоїда, який, у свою чергу, управляє кулачковим вузлом, який відповідає за подачу струму до кожного сигналу. Таймер набору номера використовується для повторення фіксованих інтервалів тривалості [2,3].

Однак ідея регулятора світлофора з фіксованим часом не зручна для міст, де потік руху є змінним. Як наслідок, учасники дорожнього руху повинні зіткнутися з такими недоліками:

1. Сильні затори. Великі затори значно збільшились у нашому місті на певних смугах до і після робочих годин. Однак звичайна система керування світлофорами забезпечує фіксовану тривалість зеленого сигналу для смуг, що мають затори, і смуг вільних від транспортних засобів.

2. Очікування порожніх доріг. Бувають випадки, коли на певних дорогах на перехрестях немає автомобілів, але світлофор все одно видає зелений сигнал для цієї дороги. Це пов'язано з тим, що існуюча система управління світлофорами заздалегідь запрограмована і учасникам дорожнього руху потрібно чекати, поки загориться зелений колір і це даремно витрачає час учасників дорожнього руху.

3. Під час проведення масових заходів дорожній рух є більш інтенсивним. Завдяки постійному таймеру в існуючій системі світлофорів це призводить до серйозних заторів.

З цієї причини необхідна динамічна адаптивна система управління дорожнім рухом, яка керує сигналами світлофорів відповідно до щільності руху.

					КВРКІ.170376.17.03.32 ПЗ	Арк. 7
Зм.	Арк.	№докум.	Підпис	Дата		

І щоб мінімізувати згадані вище недоліки, існує необхідність розробити функціонуючу в режимі реального часу розумну систему управління світлофором, яка здатна подавати належний синхронізуючий сигнал, який залежить від зміни щільності руху на транспортному розв'язку. Тобто система має виявляти та рахувати транспортні засоби та рахувати час сигналу відповідно до щільності руху [4,5].

Адаптивна система управління дорожнім рухом складається з таких компонентів[6-10]:

1. Світлофор: це основний елемент керування дорожнім рухом, який може бачити водій автомобіля або пасажир. Це можуть бути звичайні розрядні лампи розжарювання або набір світлодіодів.

2. Блок детектора: Це блок, який виявляє присутність транспортних засобів і надсилає цю інформацію контролеру для обробки.

3. Блок контролера: Це блок, який приймає вихід детектора, який вказує на присутність транспортних засобів і тим самим робить розрахунок щільності руху і відповідно контролює світлофор. Це може бути комп'ютер на основі мікропроцесора або простий мікроконтролер.

Практично існує три основних типи детекторів:

1. Детектор на основі індуктивної петлі: Він складається з котушки дроту, вбудованої в пази на дорожньому покритті. Котушка індуктивності з'єднана з детектором, який виявляє зміну резонансної частоти контуру котушки і відповідно контролює спрацьовування реле, яке використовується для спрацьовування дорожніх сигналів. В основному це працює за принципом, що коли автомобіль рухається по котушці індуктивності, індуктивність котушки зменшується. Ця зменшена індуктивність призводить до збільшення резонансної або коливальної частоти і електронний блок відповідно посиляє електричні імпульси на блок управління для управління перемиканням світлофора. Недоліком такої системи є схильність до електромагнітних перешкод, тобто електромагнітне випромінювання від інших пристроїв також може впливати на магнітне поле, а отже, і на індуктивність котушки. Вони також більш схильні до

					КВРКІ.170376.17.03.32 ПЗ	Арк. 8
Зм.	Арк.	№докум.	Підпис	Дата		

поломок і вимагають високих витрат на встановлення, а також можуть спричиняти порушення руху.

2. Давачі встановлені на стовпах: Це може бути проста композиція ІЧ-фотодіодів, які можуть виявити присутність транспортних засобів. Це працює за принципом, що коли автомобіль проходить між ІЧ-передавачем та ІЧ-приймачем, ІЧ-світло блокується, і в результаті опір фотодіода збільшується. Цю зміну опору можна перетворити на електричні імпульси, що використовуються для управління світлофорами.

3. Камери, які використовують технології комп'ютерного зору. У цьому випадку використовується декілька послідовних операцій обробки зображень, що використовуються для виділення інформації з відеокадрів для виявлення транспортних засобів.

## 1.2 Огляд систем керування світлофорами

В роботі [11] розробники впровадили адаптивну світлофорну систему, використовуючи мікроконтролер 89V51RD. У запропонованій системі мова асемблера використовувалася для програмування мікроконтролерної світлофорної системи та синхронізації її із зовнішнім апаратним забезпеченням - ІЧ-давачами. ІЧ-давачі були встановлені по обидва боки доріг відповідно і це дозволяє мікроконтролеру приймати рішення щодо часу, виходячи з кількості транспортних засобів, виявлених давачами на кожній дорозі, що перетинається. Під час роботи ІЧ-система спрацьовуватиме, коли будь-яка перешкода, подібна транспортним засобам, проїжджає по дорозі між ІЧ-передавачем і приймачем. Виходячи з різної кількості транспортних засобів, мікроконтролер визначає різні діапазони для кожної затримки світлофора і відповідно оновлює затримки.

У цьому дослідженні система реєструє кількість транспортних засобів за заздалегідь визначений інтервал запису в режимі реального часу. Інтелектуальний контролер світлофора має систему баз даних для зберігання кількості транспортних засобів за певний інтервал часу. Ці записані дані можна завантажити з мікроконтролера на ПК за допомогою послідовного методу

					КВРКІ.170376.17.03.32 ПЗ	Арк. 9
Зм.	Арк.	№докум.	Підпис	Дата		

передачі, щоб користувач міг проаналізувати стан дорожнього руху на відповідних світлофорах, підключених до системи. Адміністратор може виконувати деякі функції світлофора, такі як оновлення затримок світла або параметрів алгоритмів, стирання пам'яті та завантаження даних за допомогою командної системи в мікроконтролері. Рисунок 1.1 нижче ілюструє архітектуру інтелектуального контролера світлофора з використанням ІЧ-давачів.

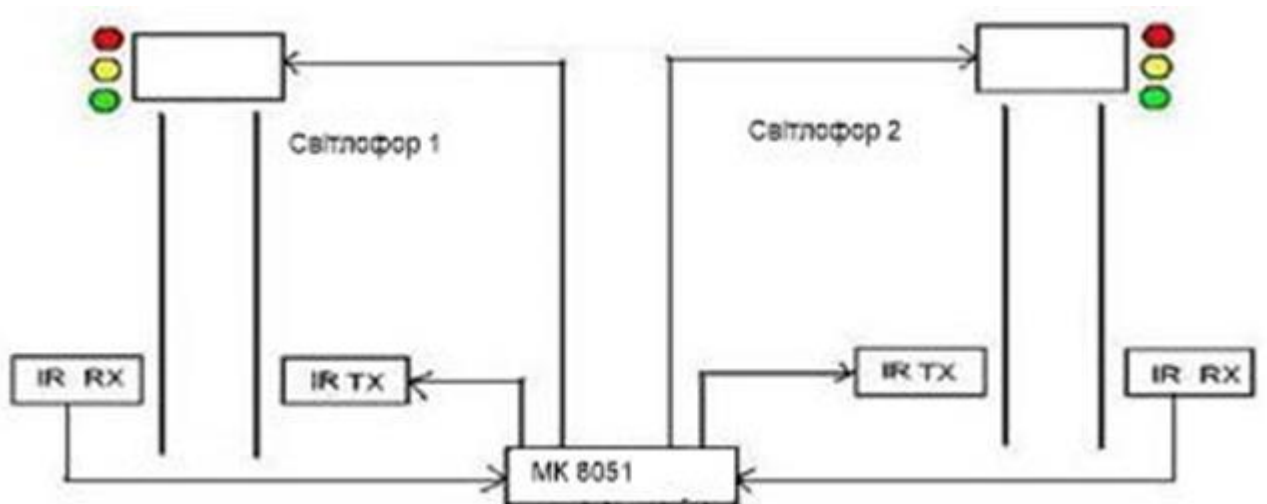


Рисунок 1.1 – Архітектура адаптивної світлофорної системи

В роботі [12] в апаратному забезпеченні контролера замість мікроконтролерів було використано недорога програмована матриця із програмуванням мовою опису апаратури Verilog. Блок-схема управління є досить важливою в процесі реалізації контролера, оскільки вона реалізує заздалегідь визначений порядок світлових сигналів, а також забезпечує загальне відображення для програміста в проекті щодо потоку програми. В даній системі використовується діаграма переходів станів реалізації поведінки системи, як показано на рисунку 1.2. Запропонована система керування складається з кінцевої кількості станів, які представляють різні дії в системі. Наприклад, у стані  $cnt = 00$ ,  $dir = 00$ , буде увімкнено зелене світло на північному світлофорі, тоді як в інших світлофорах буде увімкнено червоне світло. У стані  $cnt=00$ ,  $dir=01$ , зелене світло загориться на східному світлофорі, на інших світлофорах будуть червоні сигнали.

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

Перехід між цими станами відбуватиметься, коли будуть змінюватися умови введення або пов'язана змінна, що ініціюється, як тайм-аут. Входи/ виходи ПЛІС підключені до зовнішніх світлодіодів.

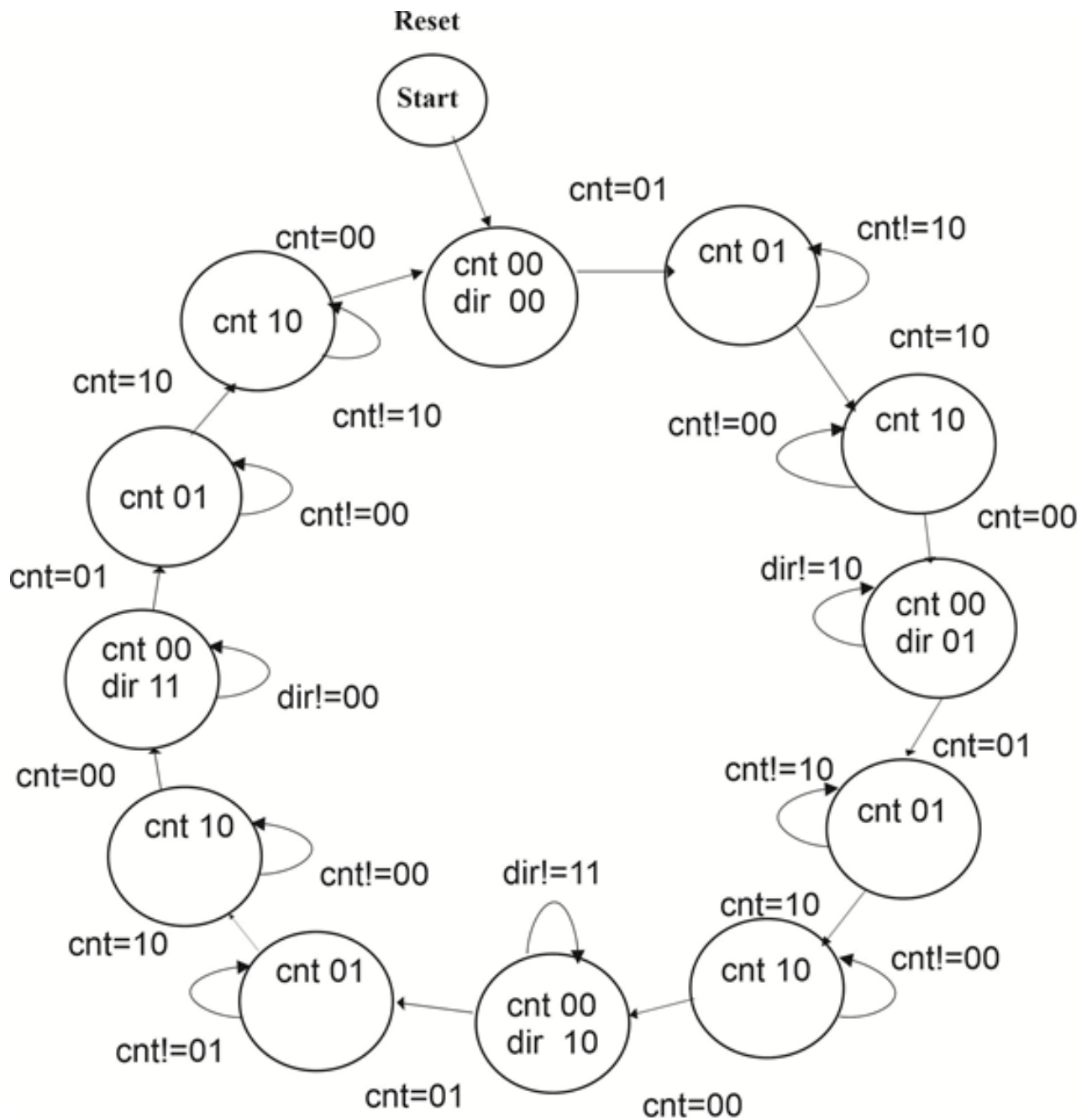


Рисунок 1.2 – Діаграма станів

Розробники у роботі [13] спроектували та впровадили адаптивну систему керування світлофором на основі програмованих матриць і за допомогою інфрачервоних (ІЧ) давачів. У запропонованій системі для виявлення

транспортних засобів використовуються ІЧ-давачі. Кожна смуга зустрічного руху на транспортній розв'язці має інфрачервоні давачі передавача та приймача.

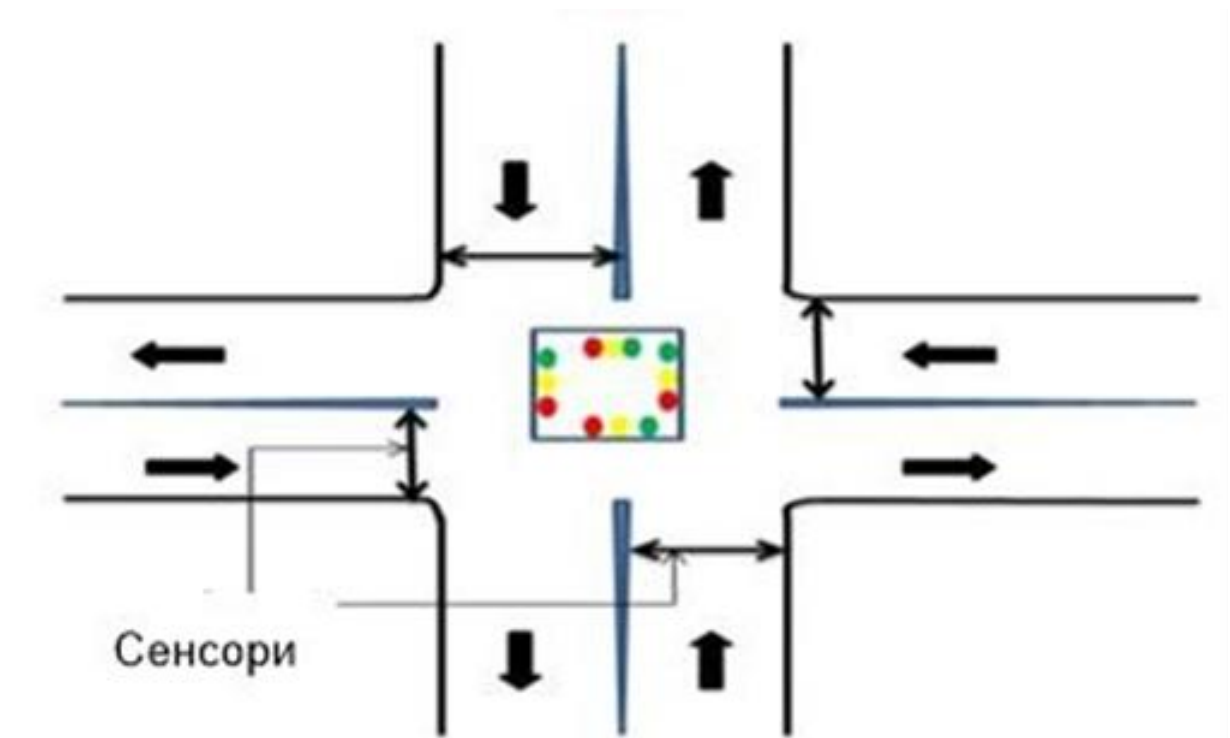


Рисунок 1.3– Модель транспортної розв'язки

Дана система керування світлофорами працює залежно від транспортних засобів, виявлених ІЧ-давачами. Якщо на кожній смузі не виявлено жодного транспортного засобу, будуть увімкнені усі червоні сигнали дорожнього руху. Якщо на одній смузі виявлено транспортний засіб, світлофор увімкне зелене світло лише для цієї смуги, і в той же час інші світлофори залишаться червоними. Якщо транспортні засоби будуть виявлені на кожній смузі, рух буде дозволено послідовно: північ (найвищий пріоритет), схід, південь, а потім захід. Контролер світлофора складається з двох основних компонентів: автомата для відстеження поточного стану та наступного стану руху та лічильника для управління переходом з одного стану в інший. В наведеній нижче таблиці наведено стани системи керування світлофорами на основі вищезазначених специфікацій (таблиця 1.1).

Таблиця 1.1 – Таблиця станів контролера світлофора

						КВРКІ.170376.17.03.32 ПЗ	Арк.
							12
Зм.	Арк.	№докум.	Підпис	Дата			

Стан	Опис стану	ПІВНІЧ	СХІД	ПІВДЕНЬ	ЗАХІД
1	2	3	4	5	6
S0	Увімкнено всі червоні сигнали	Червоний	Червоний	Червоний	Червоний
S1	ПІВНІЧ	Зелений	Червоний	Червоний	Червоний
S2	СХІД	Червоний	Зелений	Червоний	Червоний
S3	ПІВДЕНЬ	Червоний	Червоний	Зелений	Червоний
S4	ЗАХІД	Червоний	Червоний	Червоний	Зелений
S5	ПІВНІЧ	Жовтий	Червоний	Червоний	Червоний
S6	СХІД	Червоний	Жовтий	Червоний	Червоний
S7	ПІВДЕНЬ	Червоний	Червоний	Жовтий	Червоний
S8	ЗАХІД	Червоний	Червоний	Червоний	Жовтий

Після побудови таблиці станів кожний стан був описаний за допомогою блок-схеми та діаграми станів, як показано на рисунку 1.4.

Зм.	Арк.	№докум.	Підпис	Дата

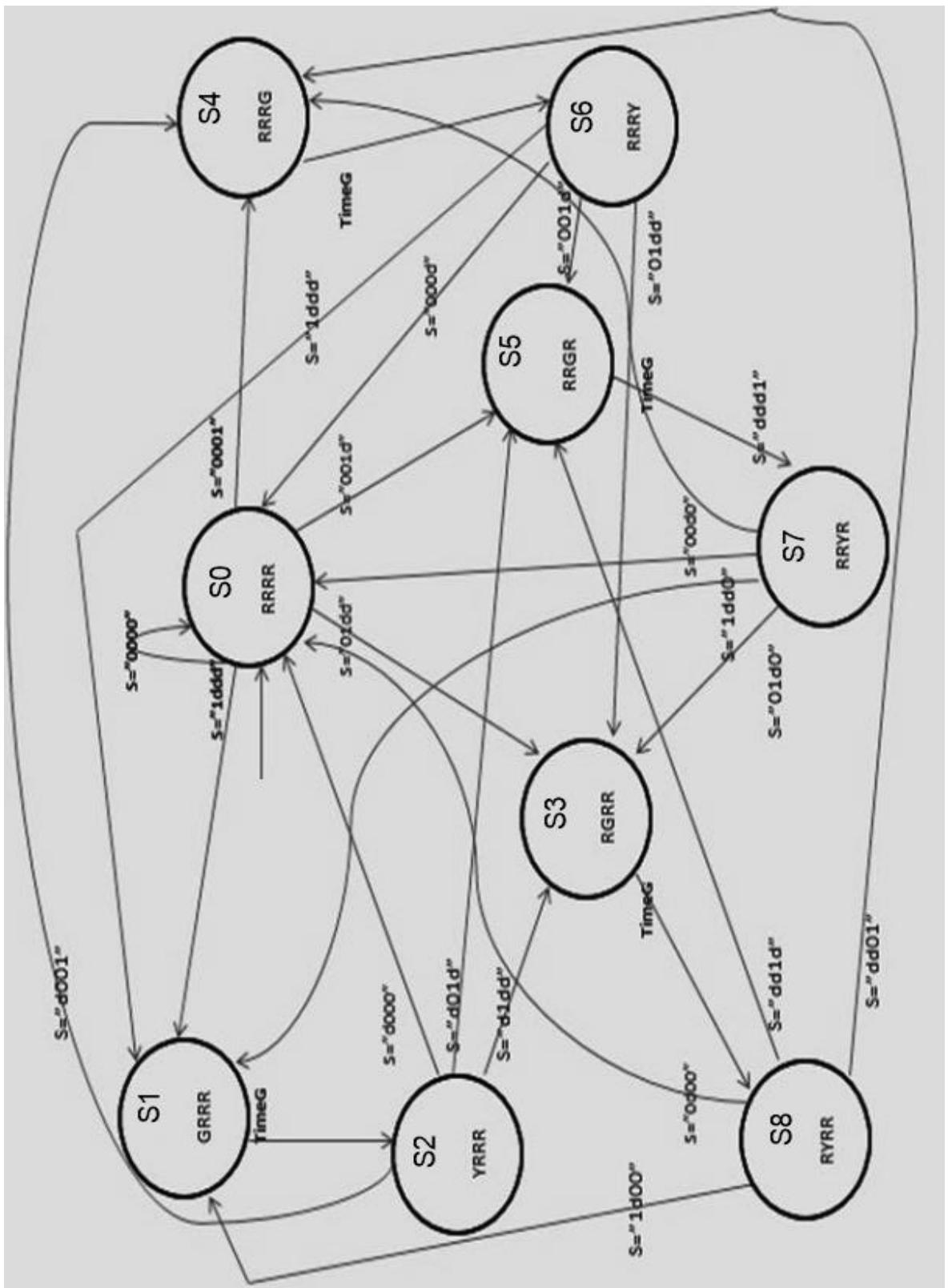


Рисунок 1.4 - Діаграма станів

В роботі [14] розробники впровадили систему керування із використанням індуктивного детектора петлі, який розміщений під поверхнею проїжджої частини, як показано на рисунку 1.5.

Зм.	Арк.	№докум.	Підпис	Дата

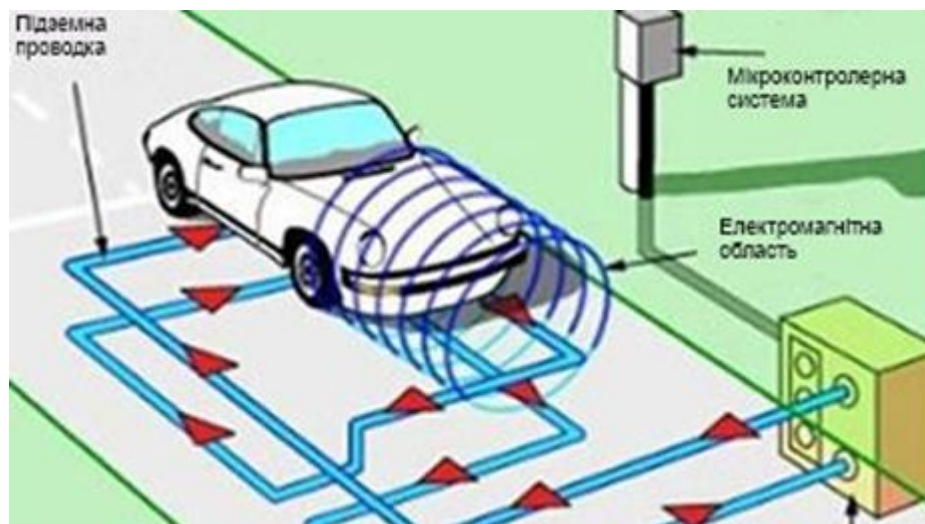


Рисунок 1.5 - Система керування світлофором на основі індуктивної петлі

У запропонованій системі на індуктивний контур подавався змінний струм і це змушує систему індуктивного контуру діяти як електричний ланцюг, а вхідний кабель є індуктивним елементом.

Коли транспортний засіб потрапляє в індуктивну петлю, електромагнітне поле, яке створюється змінним струмом у дачу контуру, буде індукувати невеликий струм у транспортних засобах. Відповідно до закону Ленца, індукований вихровий струм у транспортному засобі буде генерувати власне електромагнітне поле, яке протилежне напрямку електромагнітного поля від котушки дача, така наявність вихрового струму зменшить індуктивність індуктивної петлі. В результаті, зменшення індуктивності призводить до зменшення імпедансу і спрацьовує схема генератора для передачі імпульсного сигналу в систему збору даних, що вказує на присутність транспортного засобу, а потім використовується як вхід для блоку управління світлофором.

На рисунку 1.6 показана блок-схема системи управління на основі індуктивного циклу.

Зм.	Арк.	№докум.	Підпис	Дата

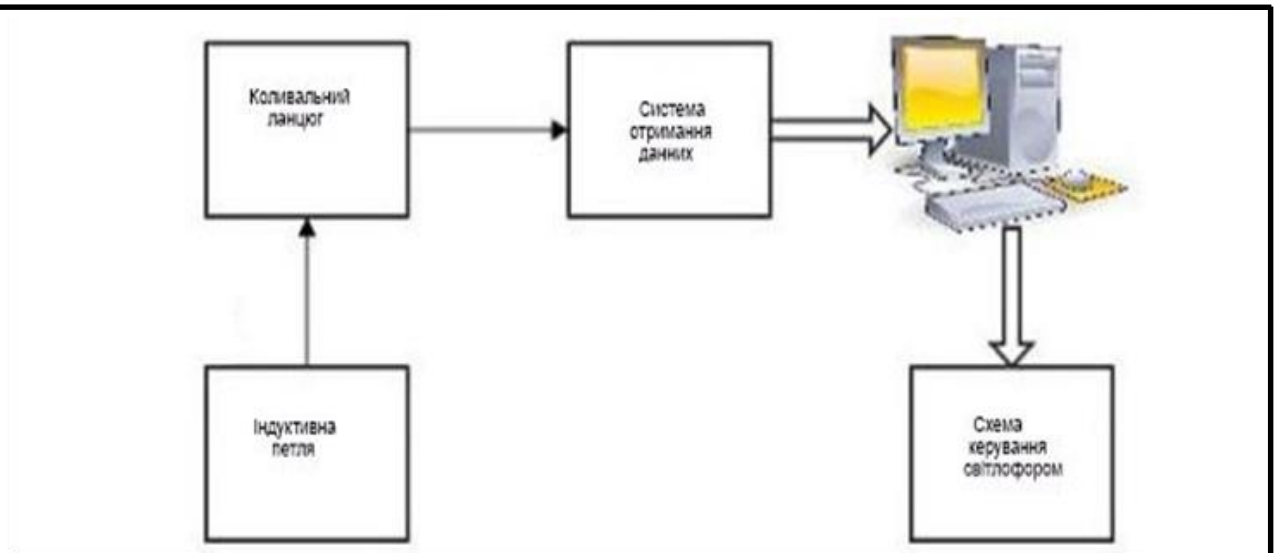


Рисунок 1.6 – Схема світлофорної системи на основі індуктивної петлі

Зміна амплітуди індуктивності петлі в індуктивній петлі дає відповідну інформацію про тип транспортного засобу (автомобіль або велосипед), швидкість, кількість та довжину транспортного засобу до блоку управління дорожнім рухом. Автори провели дослідження для аналізу трьох типів індуктивних структур петлі. На рисунку 1.7, мала петля, позначена як Loop1, була використана для виявлення малих транспортних засобів, таких як велосипед, мотоцикл, тоді як Loop2 була використана для виявлення великогабаритних транспортних засобів, таких як автомобілі, автобуси, вантажівки тощо.

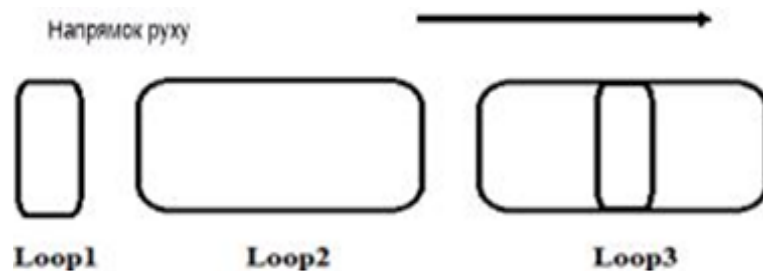


Рисунок 1.7 – Типи індуктивних детекторів петлі

Запропонована система здатна виявляти транспортний засіб з точністю близько 97%.

В проекті [15] розробники застосували методи обробки зображень в системі управління світлофорами, використовуючи цифрові камери для здійснення

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

моніторингу в режимі реального часу стану дорожнього руху на перехрестях. У цьому дослідженні існує кілька операцій обробки зображень, що використовуються для виділення інформації з відеокадрів для виявлення транспортних засобів.

1. Збір зображень: використовує веб-камеру для моніторингу стану дорожнього руху. Коли на дорозі немає транспортних засобів, зображення дороги фіксується як еталонне зображення для інших алгоритмів обробки зображень, наприклад, віднімання фону. Далі отримане зображення перетворюється у відтінки сірого.

2. Покращення зображення: передбачає такі операції, як освітлення, різкість, щоб виділити певні деталі та особливості з несприятливого зображення.

3. Відновлення зображення: Видаляє або зменшує шум, щоб покращити результат для подальших операцій із зображенням.

4. Сегментація зображення: Розбиття зображення на складові об'єкти. Сегментовані вихідні дані використовуються для аналізу або служать для представлення.

5. Морфологічна обробка.

В роботі [16] за допомогою програмного забезпечення MATLAB та мікроконтролеру Arduino було реалізовано систему з використанням технологій обробки зображень для управління дорожнім сигналом.

Використання програмного забезпечення MATLAB у цій системі зосереджено на частині обробки зображень, тоді як мікроконтролер Arduino використовувався для керування світлофором. Запропонована система складається з чотирьох веб-камер, які підключені до комп'ютера з встановленим програмним забезпеченням MATLAB. Після цього алгоритми серійної обробки зображень у програмному забезпеченні MATLAB використовувались для обробки відеокадрів з кожної камери для виявлення та підрахунку автомобіля.

Кількість виявлених транспортних засобів використовується як вхідний параметр алгоритму хронометражу для обчислення та розподілу належного зеленого часу для певної смуги. Якщо певна дорога має велике транспортне навантаження, система розрахує і призначить довший час для цієї переповненої

дороги порівняно з іншими менш заторів. На рисунку 1.8 показана блок-схема розробленої системи.

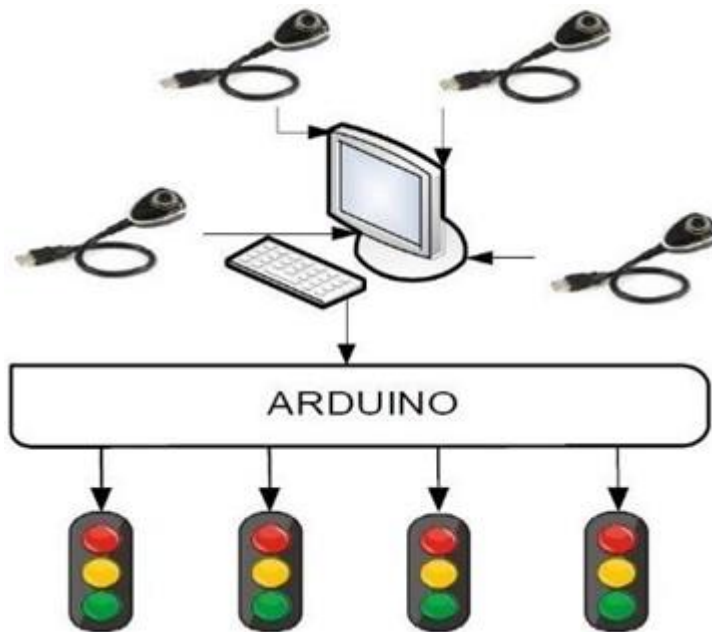


Рисунок 1.8 – Структурна схема системи

Особливістю запропонованої системи є можливість коригування алгоритмів синхронізації із зміною часу; денний та нічний. У денному режимі система застосовуватиме алгоритми віднімання фону для отримання маски переднього плану між фоновим опорним зображенням та безперервними рухомими кадрами. Після цього використовується перетворення RGB у градації сірого для перетворення 3-канального зображення переднього плану (червоного, зеленого та синього) в 1-канальне зображення (чорно-білого), щоб зменшити потужність обчислень для подальшої операції обробки зображення.

Крім того, були застосовані методи фільтрації для усунення шумів і функція `bwboundaries` використовується для пошуку меж виявленого об'єкта та підрахунку кількості об'єкта. У нічний час дослідники запропонували рішення для вирішення проблеми недостатнього освітлення - нічний режим. У нічному режимі система працює повністю аналогічно денному режиму, але виявляє передні фари автомобілів, а не виконує безпосереднє виявлення та підрахунок транспортних

засобів. Оскільки кожна машина має дві фари, кількість виявлених фар, поділена на два, буде номером автомобіля.

Запропонована система використовуватиме розрахункову кількість транспортних засобів, що знаходяться всередині смуг, для обчислення загального часу для завершення одного циклу. Після обчислення періоду циклу застосовували метод коефіцієнта ваги для розподілу періоду циклу на кожен смугу відповідно до кількості виявлених транспортних засобів.

В статі [17] дослідники впровадили інтелектуальну систему сигналів дорожнього руху, використовуючи методи обробки зображень. В системі веб-камера була встановлена на певній висоті і забезпечує моніторинг у реальному часі на кожній смузі, що перетинається. Після цього реальні часові рамки трафіку, отримані з веб-камери, передаються на ПК із встановленим для обробки відео MATLAB. При обробці відео оцінка щільності руху здійснювалась шляхом підрахунку кількості транспортних засобів на кожній смузі. Крім того, система здатна розпізнавати аварійні машини, такі як швидка допомога. У цьому дослідженні пріоритетна умова для світлофорної системи від високого пріоритету до низького пріоритету, як показано нижче:

1. Якщо на іншій смузі виявлено кілька аварійних транспортних засобів, смузі з аварійною машиною, найближчою до сигналу дорожнього руху, буде надано найвищий пріоритет.

2. Якщо буде виявлено аварійний автомобіль, пріоритет сигналу буде надаватися цій смузі.

3. Якщо аварійного автомобіля немає, пріоритет сигналу надається смузі руху з найбільшою щільністю руху.

4. Якщо щільність руху однакова на різних смугах, жодній смузі не надається пріоритет сигналу, і сигнал дорожнього руху буде слідувати за попередньо встановленим таймером.

Методи обробки зображень, задіяні у цій системі, включають наступні етапи. В якості еталонного кадру приймається зображення, у яких кадри дороги в прямому ефірі були отримані з веб-камери, а фонове зображення без

					КВРКІ.170376.17.03.32 ПЗ	Арк. 19
Зм.	Арк.	№докум.	Підпис	Дата		

транспортного навантаження. На рисунку 1.9 показано фоновий опорний кадр і поточний отриманий кадр.

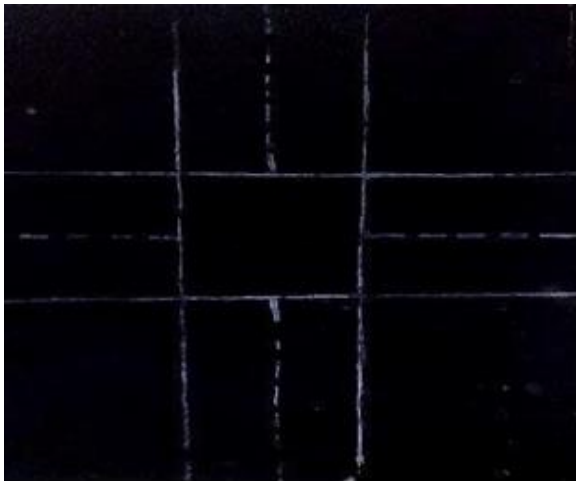


Рисунок 1.9 - Фоновий кадр(ліворуч) та кадр реального часу

Потім обрізані з кадрів області використовуються як регіони інтересів.

Крім того, для екстреного виявлення транспортного засобу проводилась операція сегментації зображень на основі кольору для виявлення синього та червоного кольорів сигналів швидкої допомоги. Якщо виявлено об'єкт червоного та синього кольорів, обчислюється відстань між об'єктами червоного та синього кольорів. Якщо розрахована відстань потрапляє в межах заздалегідь визначеної порогової відстані, система буде вважати це наявністю швидкої допомоги і надаватиме пріоритет сигналу цій смузі.

Потім для оцінки щільності трафіку опорний кадр і кадри реального часу перетворюються в зображення у відтінки сірого. Далі використовується метод віднімання фону для віднімання відтінків сірого в реальному часі для отримання масок переднього плану, які вказують на наявність транспортних засобів на дорозі. Крім того, застосовується кілька морфологічних операцій для усунення небажаних шумів на 8-бітному зображенні у відтінках сірого. Після цього покращене зображення додатково перетворюється у чорно-біле зображення та фільтрується за допомогою операції Гаусового фільтра, щоб отримати контури транспортних засобів на дорозі.

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ.170376.17.03.32 ПЗ

Арк.  
20

В кінці використовується операція із зображенням, яка малює межі білих областей у зображенні. Кількість знайдених меж буде використана для позначення кількості транспортних засобів (див. Рис. 1.10 ).

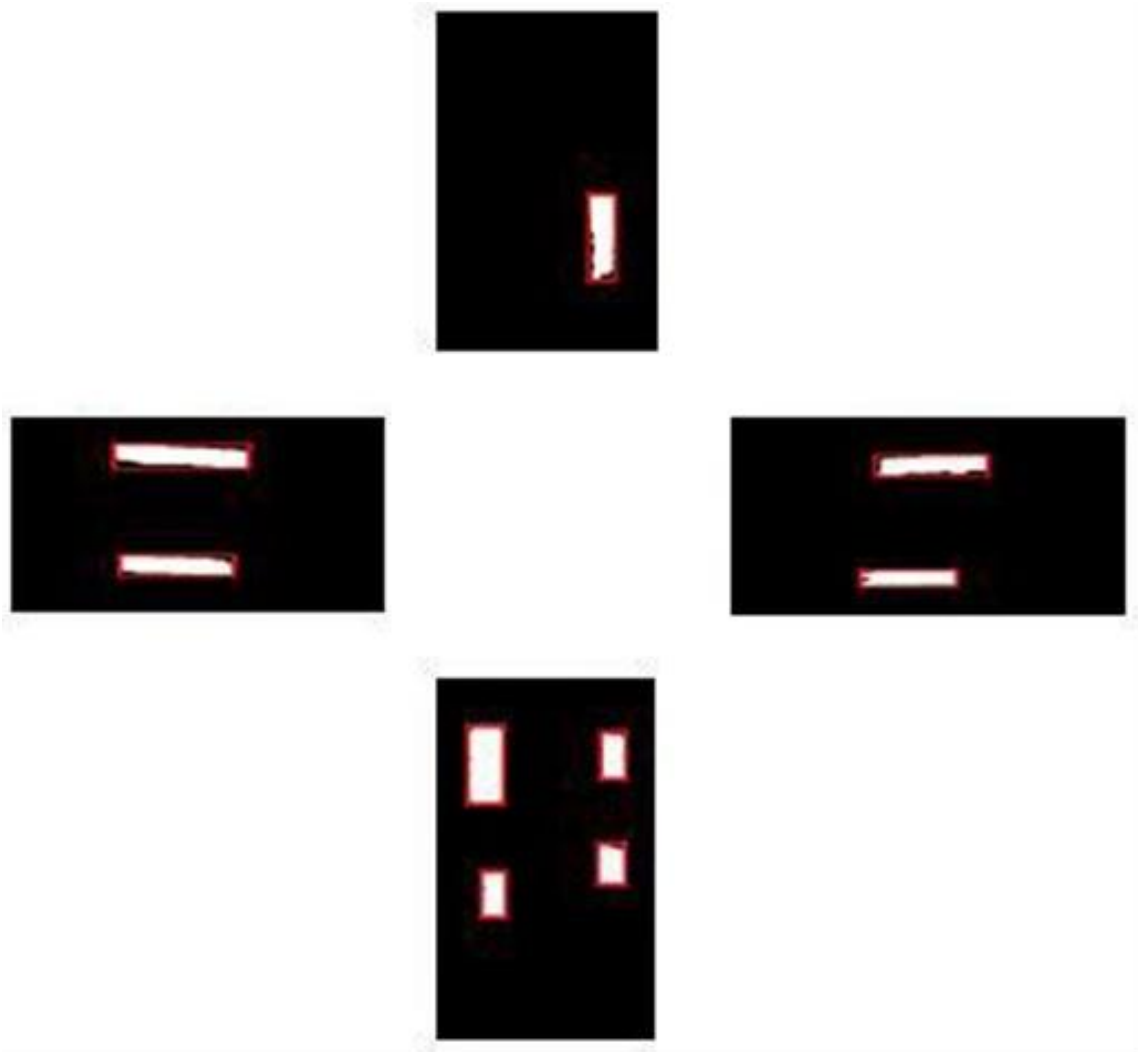


Рисунок 1.10 – Межі об'єктів

Порівняння всіх типів датчиків, що використовуються в системах адаптивного регулювання світлофорів наведено в таблиці 1.2.

Зм.	Арк.	№докум.	Підпис	Дата

Таблиця 1.2 – Порівняння методів, застосованих іншими дослідниками для впровадження адаптивної системи керування світлофором

Тип давача	Переваги	Недоліки
1	2	3
ІЧ-давачі	<ul style="list-style-type: none"> <li>- Легко реалізувати</li> <li>- Система виявлення недорога.</li> <li>- Система динамічна, оскільки запропонована система має можливість збирати дані та модифікувати параметри алгоритму.</li> </ul>	<ul style="list-style-type: none"> <li>- Давачі повинні бути добре захищені або закріплені в безпечному місці.</li> <li>- Підрахунок транспортних засобів може бути неточним через: пішоходів, двох автомобілів паралельно або зупинку транспортного засобу між давачами</li> </ul>
Детектор індуктивної петлі	<ul style="list-style-type: none"> <li>- Висока точність виявлення транспортного засобу</li> <li>- Погода не впливає на виявлення</li> <li>- Вміє підраховувати та визначати типи транспортних засобів.</li> </ul>	<ul style="list-style-type: none"> <li>- Дорогий і складний в установці</li> <li>- Процес технічного обслуговування може перешкоджати руху транспорту</li> </ul>
Обробка зображень	<ul style="list-style-type: none"> <li>- Висока точність.</li> <li>- Порівняно недорогий і простий у реалізації.</li> <li>- Може використовуватися одночасно як система спостереження, так і система контролю дорожнього руху.</li> <li>- Якість зображення та ефективність виявлення автомобіля можна покращити, використовуючи відповідні алгоритми обробки зображень.</li> </ul>	<ul style="list-style-type: none"> <li>- Якість зображення або відео залежить від використовуваної цифрової камери та впливу погоди.</li> </ul>

### 1.3 Висновки

На основі аналізу адаптивних систем керування світлофорами можна зробити висновок, що методи засновані на обробці зображень є найбільш точним і гнучкими в налаштуванні. Однак вони не використовують сучасні методи розпізнавання і класифікації зображень в галузі глибокого машинного навчання і нейронних мереж, які на сьогодні досить потужні з солідним потенціалом до розвитку і вдосконалення. Сучасні мікроконтролерні пристрої і їх програмне забезпечення дозволяють їх використовувати у вбудованих пристроях.

Тому метою роботи є розробка адаптивної системи керування світлофорами для підвищення ефективності систем управління світлофорів за рахунок зменшення часу очікування кожної смуги руху транспортних засобів та максимізації потоку транспортних засобів через перехрестя, використовуючи спеціальний алгоритм для обчислення зеленого часу.

### 1.4 Постановка задачі

Для досягнення поставленої мети мають бути вирішені наступні завдання:

- Аналіз особливостей реалізації адаптивних систем керування світлофорами, їх переваг і недоліків .
- Розробка структурної схеми і схеми функціонування системи управління світлофорами
- Вибір основних компонентів та програмного забезпечення для реалізації системи управління світлофорами.
- Розробка функціональної та принципової схем системи управління світлофорами.
- Розробка алгоритмів функціонування та програмного забезпечення системи.
- Провести аналіз ефективності розробленої системи та виявити недоліки та потенціал до удосконалення.

## ПРОЄКТУВАННЯ ТЕХНІЧНОГО ЗАСОБУ

### 2.1 Вибір основних компонентів та програмного забезпечення

З огляду робіт видно, що багато систем керування розроблені з використанням кінцевих автоматів (FSM). Це пояснюється тим, що їх простота дозволяє будь-яку систему управління світлофором легко спроектувати, впровадити та швидко реалізувати.

Виберемо для підсистеми управління світлофорами реалізацію на основі кінцевих автоматів.

Для підсистеми детектування потоку транспорту систему виберемо комп'ютерного зору на основі використання цифрових відеокамер. В останні роки системи комп'ютерного зору зробили помітний стрибок у розвитку і майже позбавились недоліків. Також сучасні мікроконтролери і програмне забезпечення дозволяють їх легко реалізувати.

Обробка зображень включає наступні етапи:

- Імпортування зображення або відео за допомогою інструментів, які можуть бути як програмними, так і апаратними.
- Маніпулювання та аналіз зображення за допомогою алгоритмів обробки зображень.
- Отримання вихідних даних з оброблених зображень.

Як основну частину системи керування і розпізнавання виберемо Raspberry Pi (RPi) [18-22], одноплатний мікроконтролер розміром із кредитну картку, який можна підключити до монітора комп'ютера та використовує периферійні пристрої, такі як клавіатура та миша. RPi використовує чотириядерний процесор ARM і є одним з найпопулярніших і найпотужніших мікроконтролерів, який широко використовується студентами університетів для реалізації своїх проєктів, пов'язаних з обробкою зображень або відео, додатків на основі IoT, робототехнічних контролерів, серверних додатків тощо.

Raspberry Pi Foundation офіційно пропонує оптимізовану ОС, розроблену на основі Debian, Raspbian для розробників для взаємодії з апаратним

					КВРКІ.170376.17.03.32 ПЗ	Арк. 24
Зм.	Арк.	№докум.	Підпис	Дата		

забезпеченням, але користувачі можуть встановити кілька сторонніх ОС, таких як Ubuntu, Windows 10 IoT Core та ін. Raspberry Pi з встановленою ОС Raspian має графічний інтерфейс робочого столу, який має багато можливостей, від редагування документів, перегляду веб-сторінок та ігор до простого графічного дизайну. Таким чином, використовуючи Raspberry Pi можна використовувати високорівневі програмні засоби для управління.

Технічні характеристики Raspberry Pi 3B +:

- Broadcom BCM2837 64-розрядний чотириядерний 1,4 ГГц.
- 1 Гб оперативної пам'яті.
- Бездротова локальна мережа та Bluetooth 2,4 ГГц та 5 ГГц.
- 4 порти USB (USB 2.0).
- Композитний відеопорт та 4-полюсний стерео вихід.
- Повнорозмірний порт HDMI.
- Порт камери CSI на 15 контактів.
- Слот Micro SD для зберігання операційної системи та даних.
- 40 контактів GPIO.



Рисунок 2.1 – Raspberry Pi 3 Модель B +

Отримання зображень - це дія для отримання зображення із зовнішнього джерела. У запропонованій системі отримання зображень здійснюється за

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ.170376.17.03.32 ПЗ

Арк.  
25

допомогою апаратного джерела модуля камери Pi, як показано на рисунку 2.2, який застосовується для зйомки зображення перехрестя у реальному часі.

Модуль камери повинен бути розміщений у такому положенні, щоб поле зору камери охоплювало перехрестя. Через обмежену фокусну відстань, відсутність можливостей автофокусування та відносно низьку роздільну здатність Pi-камери, відстань між модулем камери та моделлю прототипу руху стає критичною, що може вплинути на чіткість зображення та спричинити певний шум зображення, який неможливо усунути за допомогою техніки обробки зображень. Тому розташування камери є життєво важливим і вимагає поступових спроб і помилок під час встановлення для кращого процесу розпізнання.



Рисунок 2.2 – Модуль камери Pi

У запропонованій системі використаємо встановлену безкоштовну бібліотеку OpenCV на мікроконтролерній платі Raspberry Pi 3B+ з модулем цифрової камери Pi, яка використовується для отримання зображення, обробки та оцінки кількості транспортних засобів на кожній смузі [23-27].

OpenCV - це бібліотека комп'ютерного зору та машинного навчання з відкритим кодом, представлена Intel більше десяти років тому. З тих пір багато програмістів внесли свій внесок в цю бібліотеку і на тепер містить понад 2500 оптимізованих алгоритмів комп'ютерного зору. На сьогодні остання версія OpenCV - 4.1.0 (OpenCV 4), ця версія є найбільш функціональною, найшвидшою і стабільною OpenCV, оскільки вона підтримується різними мовами програмування, такими як C++, Python тощо і багатьма операційними платформами, включаючи Linux, Android та macOS.

Обробка зображень - це метод цифрової обробки сигналу, який використовує алгоритми обробки зображень у бібліотеці OpenCV для обробки кожного пікселя із зображення або відео для отримання необхідної інформації.. Обробка зображення, яка розглядає вхідне зображення або відеокадр як 2D-сигнали (x, y), а після цього застосовує до них операції обробки сигналів і отримує вихідне зображення, що містить виділені характеристики, пов'язані із вхідним зображенням. Зі стрімким зростанням технологій сьогодні обробка зображень стає популярною завдяки надзвичайно застосовним в різних аспектах бізнесу, безкоштовних та відкритих джерел. Таким чином, він формує основний напрямок досліджень в галузі інформатики та штучного інтелекту

Також для класифікації ситуації на перехресті використовуємо Cvlіb - високоякісну, просту у використанні бібліотека Computer Vision з відкритим кодом для Python на основі згорткових нейронних мереж [28-38].

Вона був розроблена з акцентом на можливість швидких та швидких експериментів. Можливість переходу від ідеї до прототипу із найменшою затримкою є ключовим фактором для проведення хороших досліджень.

Керівні принципи cvlib значною мірою натхнені бібліотекою Keras:

- простота;
- зручність користування;
- модульність;
- розширюваність.

За умови встановлення пакетів OpenCV і Tensorflow cvlib можна повністю встановити через pip:

```
pip install opencv-python tensorflow
```

```
pip install cvlib
```

Вона містить функції розпізнавання обличчя, виявлення статті, виявлення 80 різних типів об'єктів.

## 2.2 Структура та загальний схема функціонування системи

На основі вищесказаного представимо структурну схему управління світлофорами навчального макету (рисунок 2.3)

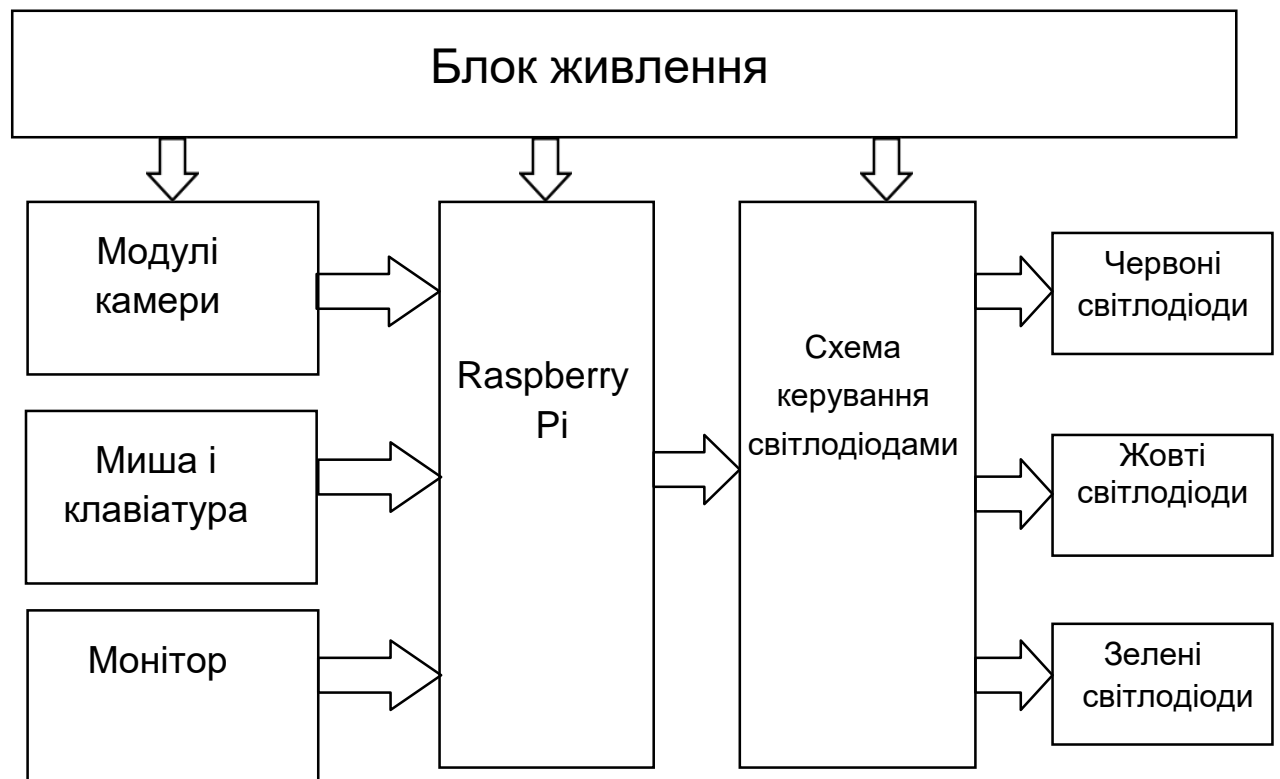


Рисунок 2.3 – Структурна схема системи

Архітектура системи складається з наступних апаратних компонентів:

- Raspberry Pi 3B +.
- Світлофорна схема.
- Схема керування.
- Монітор.
- Миша і клавіатура.
- Модуль камери.
- І наступних програмних компонентів.
- IDE Spyder з мовою Python.
- Бібліотеки OpenCV і CVLIB.
- ОС Raspbian.

На рисунку 2.4 показано загальна схема функціонування адаптивної системи керування світлофором з обробкою зображень. На першому кроці використовується модуль камери для моніторингу та захоплення смуг руху на перехресті. Далі використовуються такі методи обробки зображень, як покращення зображення, вибор певної смуги руху, виявлення транспортних засобів в черзі та підрахунок. Кількість транспортних засобів, підрахованих на певній смузі, використовується у алгоритмі для обчислення потрібного зеленого часу для максимізації потоку руху через розв'язку. Після розрахунку часу, обчислений час надсилається через послідовну чергу мови Python процесу, який реалізує логіку керування світлофором, тобто відображає зелене світло відповідно до отриманого значення часу. Після зеленого світла з увімкненням жовтого світла надсилається сигнал запиту процесу обробки зображень для виконання розрахунку часу для наступного увімкнення зеленого.

Ці процеси повторюються для забезпечення оптимального потоку руху на розв'язку.

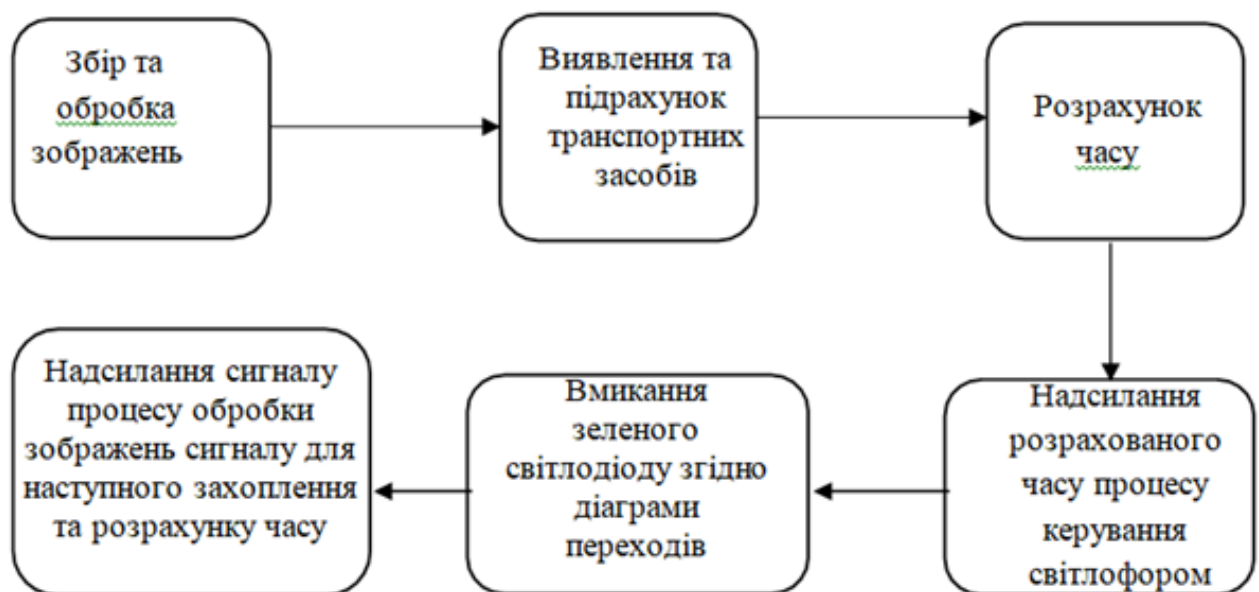


Рисунок 2.4 – Схема функціонування системи

## 2.3 Апаратна реалізація системи управління світлофорами

На рисунку 2.5 нижче показано розроблену функціональну схему керування світлофорами, яку реалізовано за допомогою Raspberry Pi. Світлодіоди для керування макетом підключаються до виводів загального призначення (GPIO) на платі Raspberry Pi. Одна схема світлофора визначеного напрямку складається з 4 резисторів, 2 зелених світлодіодів, 1 жовтого світлодіода та 1 червоного світлодіода. Таким чином загалом отримуємо 16 світлодіодів (4 червоних, 4 жовтих та 8 зелених світлодіодів) та резисторів, які необхідні для всіх напрямків світлофорної схеми для обмеження струму.

Основним керуючим елементом зазначеної схем є мікроконтролер BCM2837BO. Він працює під управління операційної системи Raspbian, яка встановлюється на карті мікроSD і дозволяє встановлювати мову Python і відповідні бібліотеки для реалізації роботи. Для керування і отримання інформації з Pi камери використовується інтерфейс CSI CAM. Для налаштування і керування модулем Raspberry Pi зі сторони розробників використовуються:

- клавіатура, яка використовує інтерфейс USB через USB HUB 2.0;
- монітор через HDMI-інтерфейс;
- контролер локальної мережі LAN9514;
- контролер мережі Wi-Fi BCM43438.

При реалізації принципової схеми використовуємо відкриту документацію платі, а Raspberry Pi 3 Model B+ (Reduced Schematic) [39].

Дана схема надає для користування і розробки в основному схеми інтерфейсів для під'єднання зовнішніх пристроїв і пристроїв, які забезпечують їх функціонування.

					КВРКІ.170376.17.03.32 ПЗ	Арк. 30
Зм.	Арк.	№докум.	Підпис	Дата		



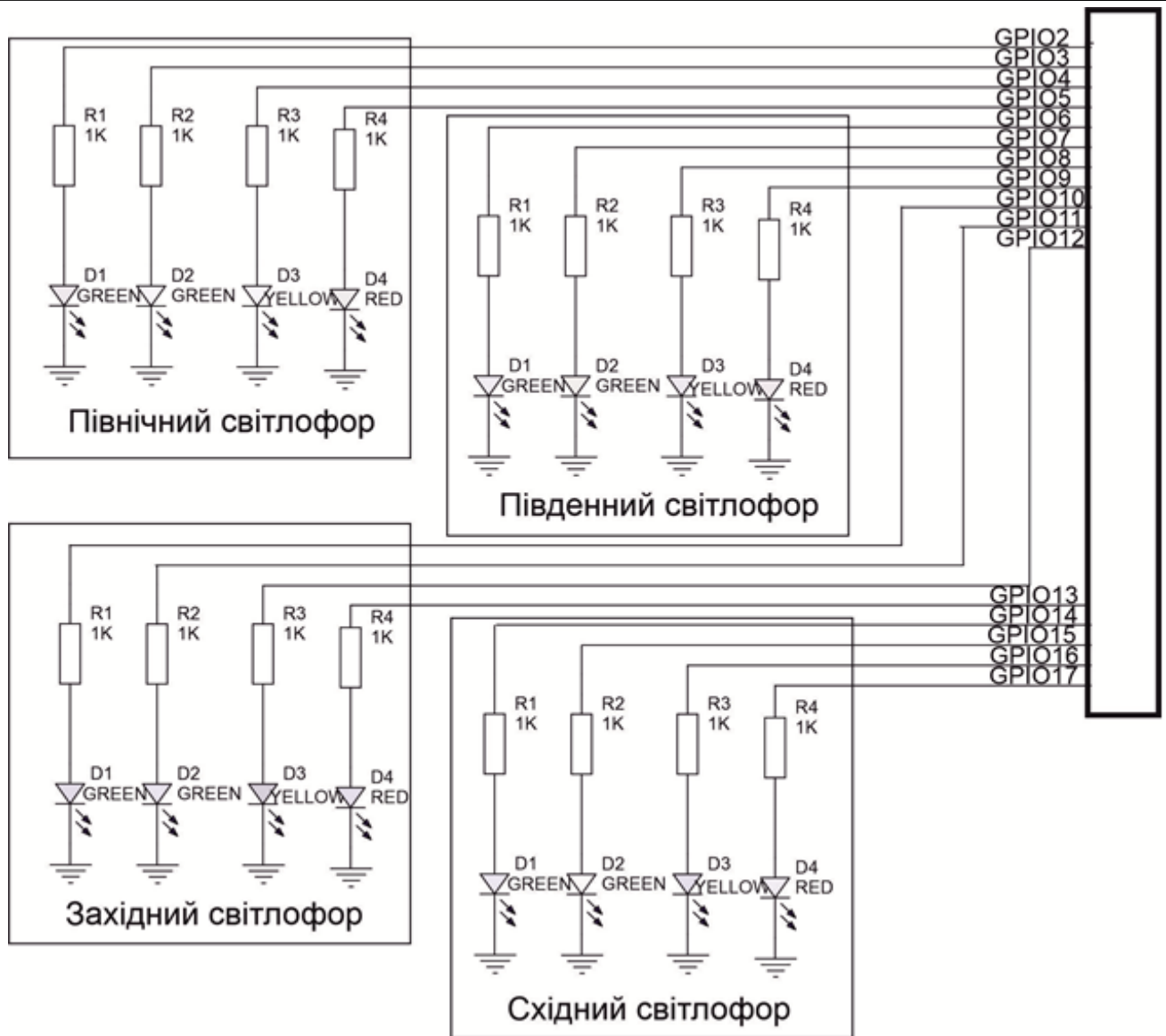


Рисунок 2.6 – Принципова схема керування світлодіодами світлофорів

Під час реалізації схеми слід враховувати максимальний струм для світлодіодів. У схемі світлофора кожен світлодіод повинен мати резистор, який підключається послідовно, щоб запобігти перевищенню струму через світлодіод, інакше світлодіод згорить. У цій роботі використовують 5-міліметрові світлодіоди з максимальним струмом близько 20 мА. Отже, для забезпечення надійності схеми необхідний струм менше 10 мА. Для отримання цього поточного значення для обчислення мінімально необхідного значення резистора використовується рівняння закону Ома, як показано в рівнянні нижче:

$$R = (V_s - V_f) / I$$

$$R = (3.3 \text{ В} - 1.8 \text{ В}) / 0.01 \text{ А}$$

$$R = 150 \text{ Ом},$$

де,  $V_s$  - це напруга, що подається вихідним контактом GPIO, максимальна напруга - 3,3 В,  $V_f$  - це напруга світлодіода, типове значення становить 1,8 В,  $I$  - поточне значення, припустимо, що воно становить 10 мА,  $R$  - значення опору в омах, Ом.

Виходячи з розрахунку вище, проста схема керування світлофором складається з чотирьох резисторів на 150 Ом та 4 світлодіодів, як показано в конструкції рисунку 2.7. З наведеної нижче схеми світлофорної плати, яка отримана за допомогою програми Eagle 7.1.0 схема має 6 контактів, для світлодіодів використовуються 4 роз'єми, що здійснювали підключення до плати Raspberry Pi через резистор 150 Ом. Ще один контакт - це контакт земля (GND).

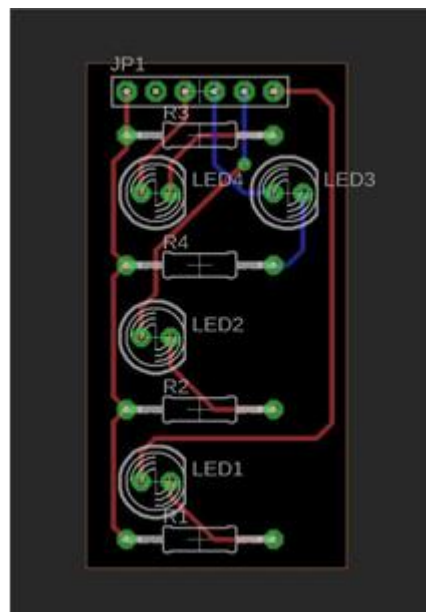


Рисунок 2.7 – Друкована плати схеми управління світлофором

## 2.4 Висновки

Для підсистеми управління світлофорами реалізацію на основі кінцевих автоматів. Для системи детектування потоку транспорту систему виберемо комп'ютерного зору на основі використання цифрових відеокамер.

Зм.	Арк.	№докум.	Підпис	Дата

Як основну частину системи керування і розпізнавання вибрано Raspberry Pi (RPi), одноплатний мікроконтролер. RPi використовує чотири ядерний процесор ARM і є одним з найпопулярніших і найпотужніших мікроконтролерів.

Розроблено структурну схему управління світлофорами навчального макету та загальний алгоритм функціонування адаптивної системи керування світлофором з обробкою зображень.

Розроблену функціональну і принципові схеми керування світлофорами, яку реалізовано за допомогою Raspberry Pi. Світлодіоди для керування макетом підключаються до виводів загального призначення (GPIO1) на платі Raspberry Pi.

Основним керуючим елементом зазначеної схем є мікроконтролер BCM2837BO. Він працює під управління операційної системи Raspbian, яка встановлюється на карті microSD і дозволяє встановлювати мову Python і відповідні бібліотеки для реалізації роботи.

					КВРКІ.170376.17.03.32 ПЗ	Арк.
						34
Зм.	Арк.	№докум.	Підпис	Дата		

# АЛГОРИТМ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ УПРАВЛІННЯ СВІТЛОФОРАМИ

## 3.1 Блок-схема функціонування системи

Тема роботи – розробка системи керування світлофором для макета. Для цього використовується мікрокомп'ютер на основі Raspberry Pi . Його вхідні-вихідні порти дозволяють керувати чотирма світлофорами макету на перехресті. Однак, якщо використовувати дану систему для керування реальними світлофорами, то буде потрібно робити систему з декількох функціональних підсистем.

Одна на основі Raspberry Pi буде займатися розпізнаванням і підрахунком автомобілів, інші, можливо менш потужні в обчислювальному плані, безпосереднім керуванням світлофорами. Відповідно і програмне забезпечення буде розроблятися з двох паралельно функціонуючих частин.

Для роботи, яка розробляється для макета перехрестя керування світлофором і розпізнавання і підрахунок автомобілів буде вестися на одній платі Raspberry Pi. Відповідно програмне забезпечення буде складатися з двох процесів на мові Python, які виконуються на одній платі Raspberry Pi і взаємодіють через механізм черг Queue.

Процеси будуть запускатися паралельно на Raspberry Pi. На рисунку 3.1 показані алгоритми роботи цих двох процесів і механізм їх взаємодії.

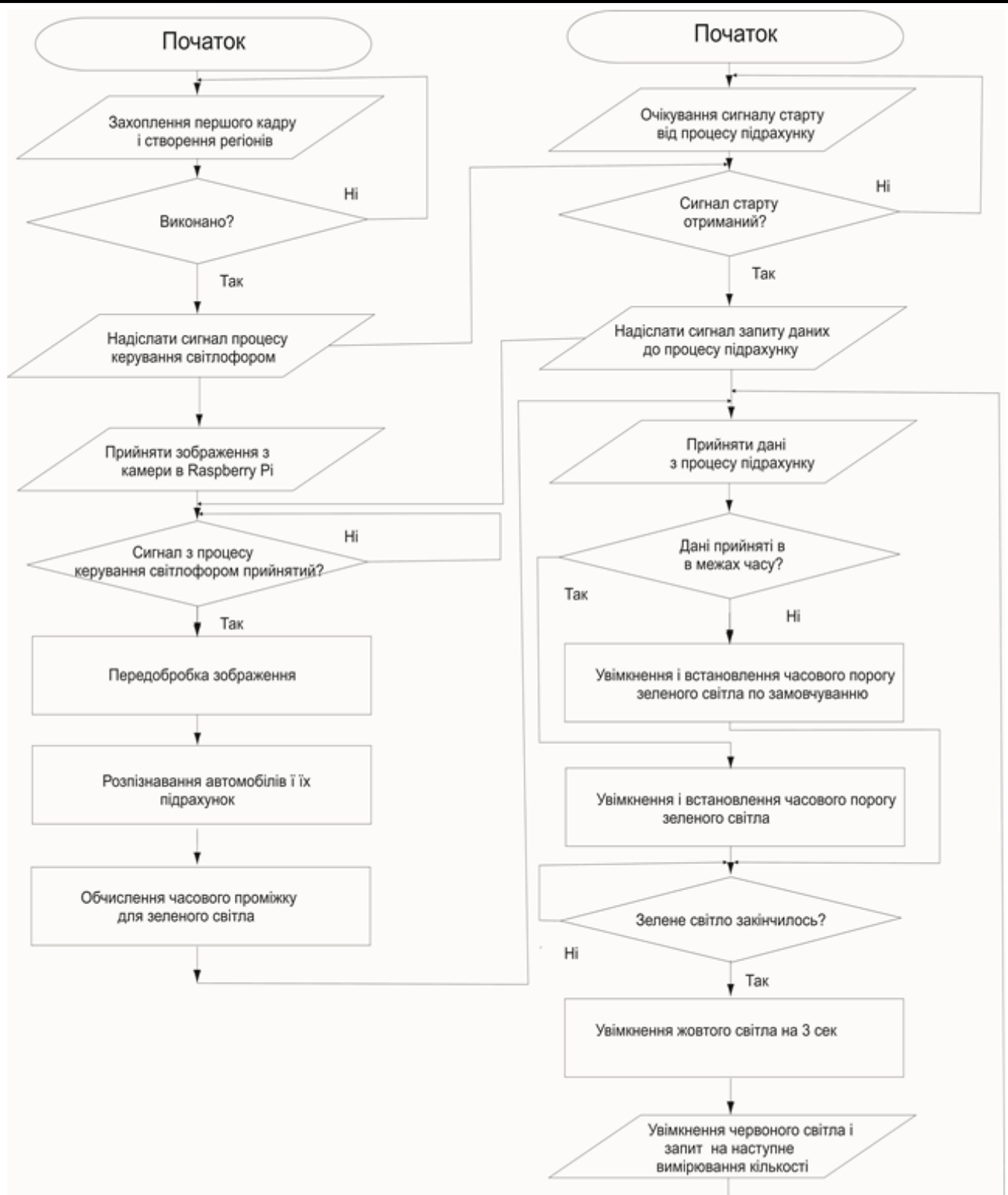


Рисунок 3.1 – Блок-схема запропонованої системи

### 3.2 Операції з обробки зображень

Блок-схема операцій з обробки зображень показана на рис. 3.2.

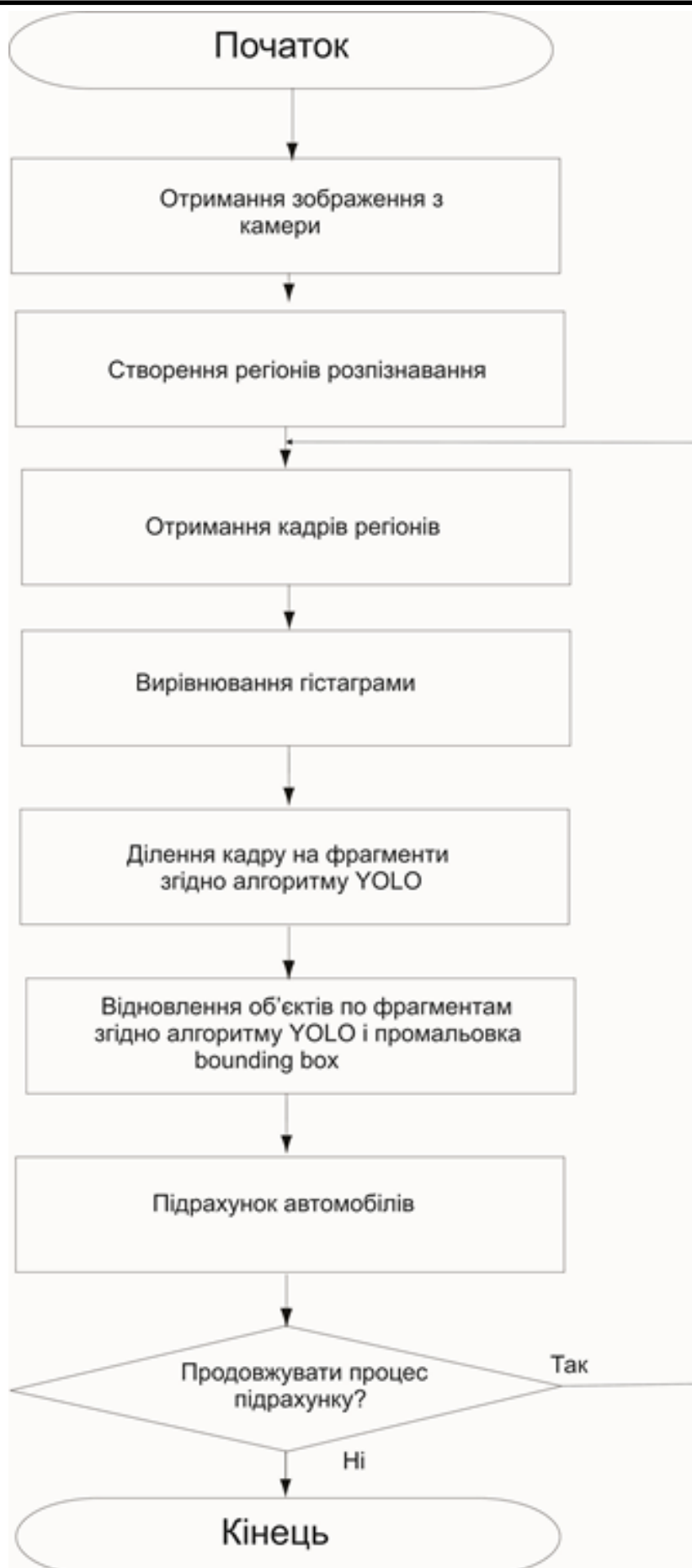


Рисунок 3.2 – Блок-схема обробки зображень

Зм.	Арк.	№докум.	Підпис	Дата

Регіони розпізнавання використовуються для встановлення меж для виявлення транспортних засобів та ігнорування перешкод або шумів, що виникають поза межами. У цій роботі вони створюються на кожній з доріг на перехресті, як показано на рисунку 3.3.



Рисунок 3.3 – Приклад вибору регіонів розпізнавання на прикладі перехрестя вулиць Кам'янецької та Подільської м. Хмельницького

#### Покращення зображення (вирівнювання гістограм)

Вирівнювання гістограм - це техніка вдосконалення зображення, яка модифікує або регулює інтенсивність зображення для отримання кращого контрасту. Цей метод зазвичай збільшує глобальний контраст багатьох зображень, особливо коли вони придатні для використання даних зображення представлено значеннями близької контрастності. Завдяки цій корекції, інтенсивності можна краще розподілити на гістограмі. Це дозволяє зонам з

меншим локальним контрастом отримати вищий контраст. Вирівнювання гистограми досягає цього, розподіляючи найчастіші значення інтенсивності.

Метод може використовуватись для зображень з яскравими або темними фоном.

З рисунка 3.4 видно, що перед застосуванням вирівнювання гистограми, інтенсивність кольору кольорового зображення є низькою в динамічному діапазоні, що означає, що певний діапазон рівнів інтенсивності утримується на багато пікселів. Як результат, зображення показує менше деталей. Однак, коли застосовується вирівнювання гистограми, це змінює зображення, регулюючи значення функція щільності ймовірності (PDF) оригінальної гистограми, так що весь діапазон рівнів інтенсивності розподіляється на кожен піксель майже однаково, в результаті зображення матиме більш рівномірний розподіл інтенсивності, покращену якість зображення, підвищену яскравість і одночасно більше деталей на зображенні з низьким освітленням.

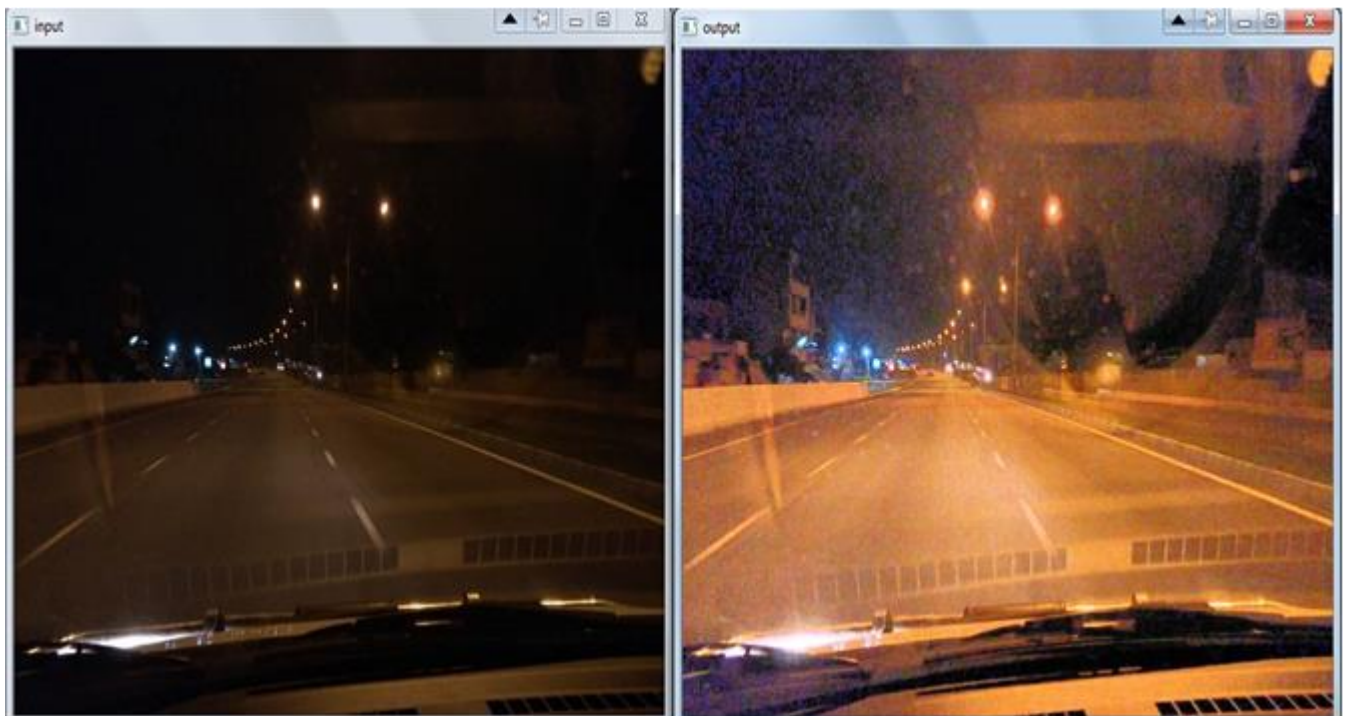


Рисунок 3.4 – Зображення до та після вирівнювання гистограми

Виявлення і підрахунок автомобілів на перехресті

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ.170376.17.03.32 ПЗ

Арк.  
39

Виявлення кількості автомобілів на перехресті реалізується за допомогою одного виклику функції - `detect_common_objects()`. Вона повертає обмежувальні координати, кореспондуючі мітки та оцінки достовірності для виявлених об'єктів на зображенні.

```
import cvlib as cv
from cvlib.object_detection import draw_bbox
bbox, label, conf = cv.detect_common_objects(img)
output_image = draw_bbox(img, bbox, label, conf)
```

Для реалізації автомобілів використовується згортова нейронна модель глибокого навчання YOLOv4, навчена за допомогою датасета COCO і здатна виявляти до 80 типів об'єктів, у тому числі автомобілі. Darknet YOLOv4 швидше і точніше, ніж нейронні мережі Google TensorFlow EfficientDet і FaceBook Pytorch / Detectron RetinaNet / MaskRCNN.

YOLOv4 досить складна модель для роботи на центральному процесорі. При роботі веб-камерою у реальному часі використовуємо `tiny yolo` зменшену версію оригінальної моделі YOLO. Вона працює значно швидше, але менш точно.

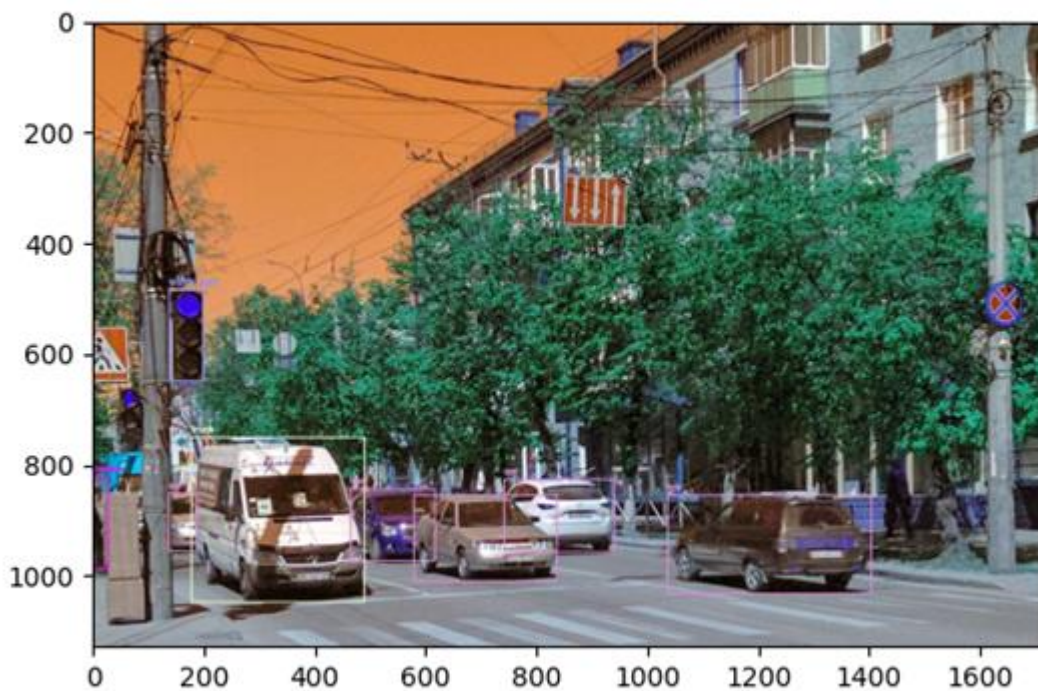
```
bbox, label, conf = cv.detect_common_objects(img, confidence=0.25,
model='yolov4-tiny')
```

Для демонстрації роботи функції на рисунку 3.5 а) показаний фрагмент програми для підрахунку транспортних засобів, а рисунку 3.5 б) приклад роботи програми.

```
from cv2 import cv2
import matplotlib.pyplot as plt
import cvlib as cv
from cvlib.object_detection import draw_bbox
draw_bboxim = cv2.imread('3.png')
bbox, label, conf =
cv.detect_common_objects(draw_bboxim)
output_image = draw_bbox(draw_bboxim, bbox, label,
conf)
plt.imshow(output_image)
plt.show()
print('Number of cars in the image is '+
str(label.count('car')))
```

a)

					КВРКІ.170376.17.03.32 ПЗ	Арк. 40
Зм.	Арк.	№докум.	Підпис	Дата		



б)

Рисунок 3.5 – Розпізнавання і підрахунок транспортних засобів

а) фрагмент програми для підрахунку транспортних засобів;

б) приклад роботи програми для перехресті вулиць Кам'янецької і Вайсера.

3.3 Структура макетної транспортної розв'язки і розробка діаграми переходів станів контролера світлофору

На рисунку 3.6 показано загальну структуру макетного транспортного вузла, який складається з північного, східного та західного та південного напрямків. Чотири світлофори з набором з чотирьох сигналів: два зелені, жовті та червоні ліхтарі використовуються для управління потоком руху транспортних засобів з кожного напрямку смуги руху.

Зм.	Арк.	№докум.	Підпис	Дата

КВРКІ.170376.17.03.32 ПЗ

Арк.  
41

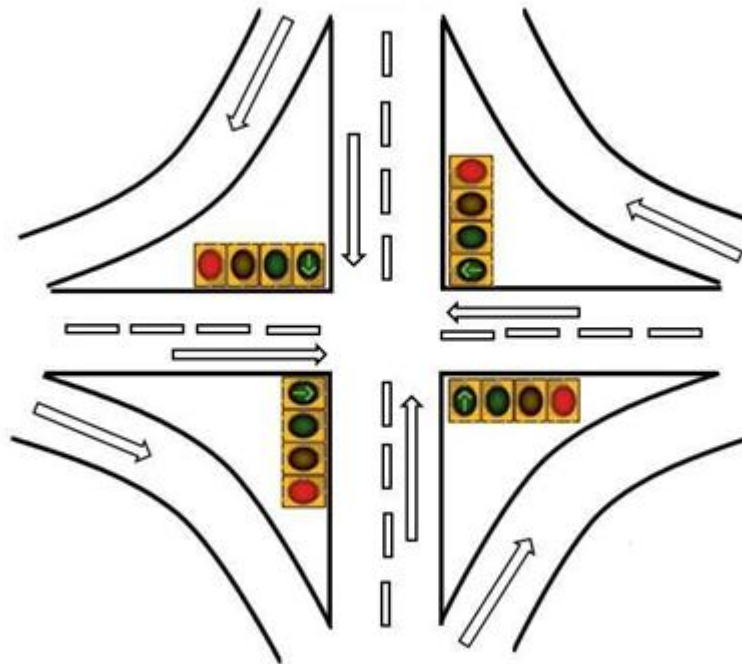


Рисунок 3.6 – Модель транспортної розв'язки

З наведеної вище моделі транспортної розв'язки транспортні засоби можуть вільно рухатися із західного на південний, південний на східний, східний на північний та північно-західний напрямки. Однак для транспортних засобів, яким потрібно рухатись прямо чи ліворуч, водії зобов'язані дотримуватись дорожніх знаків із задалегідь визначеними послідовностями або етапами.

Для проектування контролера світлофора проектування фази руху є одним із важливих кроків, щоб розділити конфлікт руху транспортних засобів на перехресті на різні фази. У запропонованій системі чотирифазні сигнали використовуються при реалізації схеми керування, а рух транспортних засобів через перехрестя руху в різні фази показано на рисунках нижче.

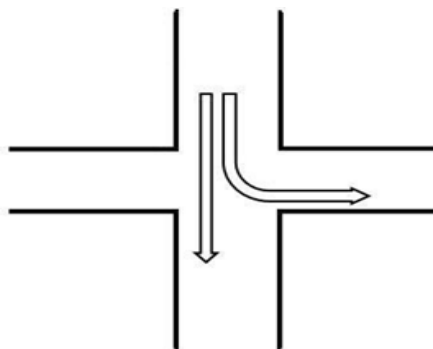


Рисунок 3.7 – Сигнал дорожнього руху фази 1

Зм..	Арк.	№докум.	Підпис	Дата

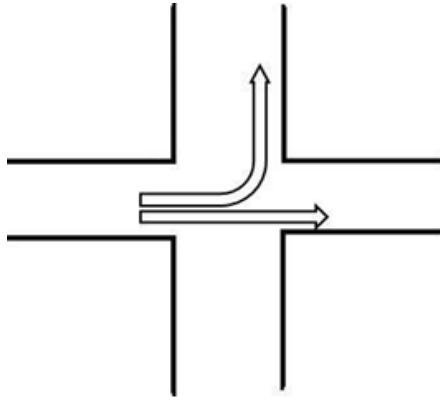


Рисунок 3.8 – Сигнал дорожнього руху фази 2

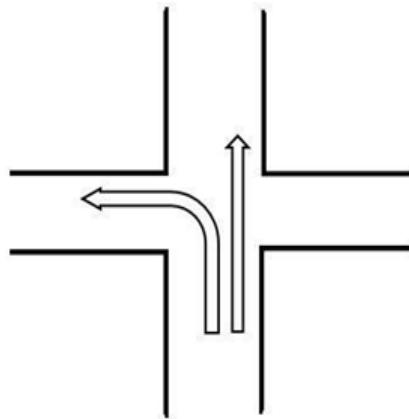


Рисунок 3.9 – Сигнал дорожнього руху фази 3

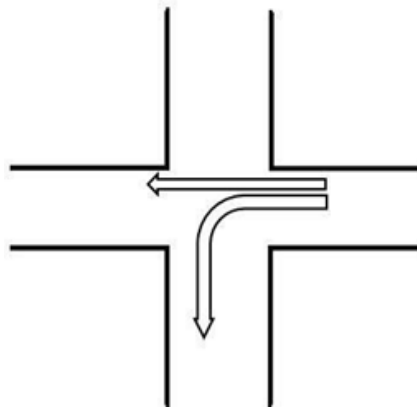


Рисунок 3.10 – Сигнал дорожнього руху фази 4

Відповідно сигналів у фазах руху можна побудувати таблицю стану контролера світлофора (Табл. 3.1).

Зм.	Арк.	№докум.	Підпис	Дата

Таблиця 3.1 – Таблиця станів контролера світлофора

Стани				Напрямок смуг руху				Вихідні дані
Поточний стан	Збереження стану	Наступний стан		Північ	Схід	Південь	Захід	S_light, W_light, N_light, E_light
		clr = 0	clr = 1					
Бездіяльність (Idle) (D)	A	I	Idle	Червоний	Жовтий	Червоний	Червоний	0001 0001 0001 0010
A	C	B	Idle	Зелений	Червоний	Червоний	Червоний	1100 0001 0001 0001
B		I	Idle	Жовтий	Червоний	Червоний	Червоний	0010 0001 0001 0001
C	E	D	Idle	Червоний	Зелений	Червоний	Червоний	0001 1100 0001 0001
D		I	Idle	Червоний	Жовтий	Червоний	Червоний	0001 0010 0001 0001
E	G	F	Idle	Червоний	Червоний	Зелений	Червоний	0001 0001 1100 0001
F		I	Idle	Червоний	Червоний	Жовтий	Червоний	0001 0001 0010 0001
G	A	H	Idle	Червоний	Червоний	Червоний	Зелений	0001 0001 0001 1100
H		I	Idle	Червоний	Червоний	Червоний	Жовтий	0001 0001 0001 0010
I		Збереження стану	Idle	Червоний	Червоний	Червоний	Червоний	0001 0001 0001 0001

Блок-схема функціонування контролера світлофора показана на рисунку 3.11.

Рисунок 3.11 вище ілюструє блок-схему реалізованої системи керування. Спочатку контролер світлофора надішле сигнал запиту та отримає значення часу зеленого світлофора для відповідного напрямку, розраховане Raspberry Pi.

Якщо дані отримані неправильно через проблему з підключенням або отримане значення не знаходиться в межах діапазону, система контролера світлофора використовуватиме зелений час за замовчуванням для контролю дорожнього руху.

В іншому випадку на світлофорі відобразатиметься зелене світло відповідно до отриманого зеленого значення часу, система завжди порівнює отримані дані зі значенням лічильника, щоб обмежити зелений період часу. Коли період зеленого світла вичерпався зелене світло вимикається, вмикається жовте світло буде і контролер світлофору надсилає сигнал для отримання часу зеленого сигналу для наступної фази дорожнього сигналу.

Цей процес постійно повторюється.

					КВРКІ.170376.17.03.32 ПЗ	Арк.
						45
Зм.	Арк.	№докум.	Підпис	Дата		



Рисунок 3.11 - Блок-схема контролера світлофора

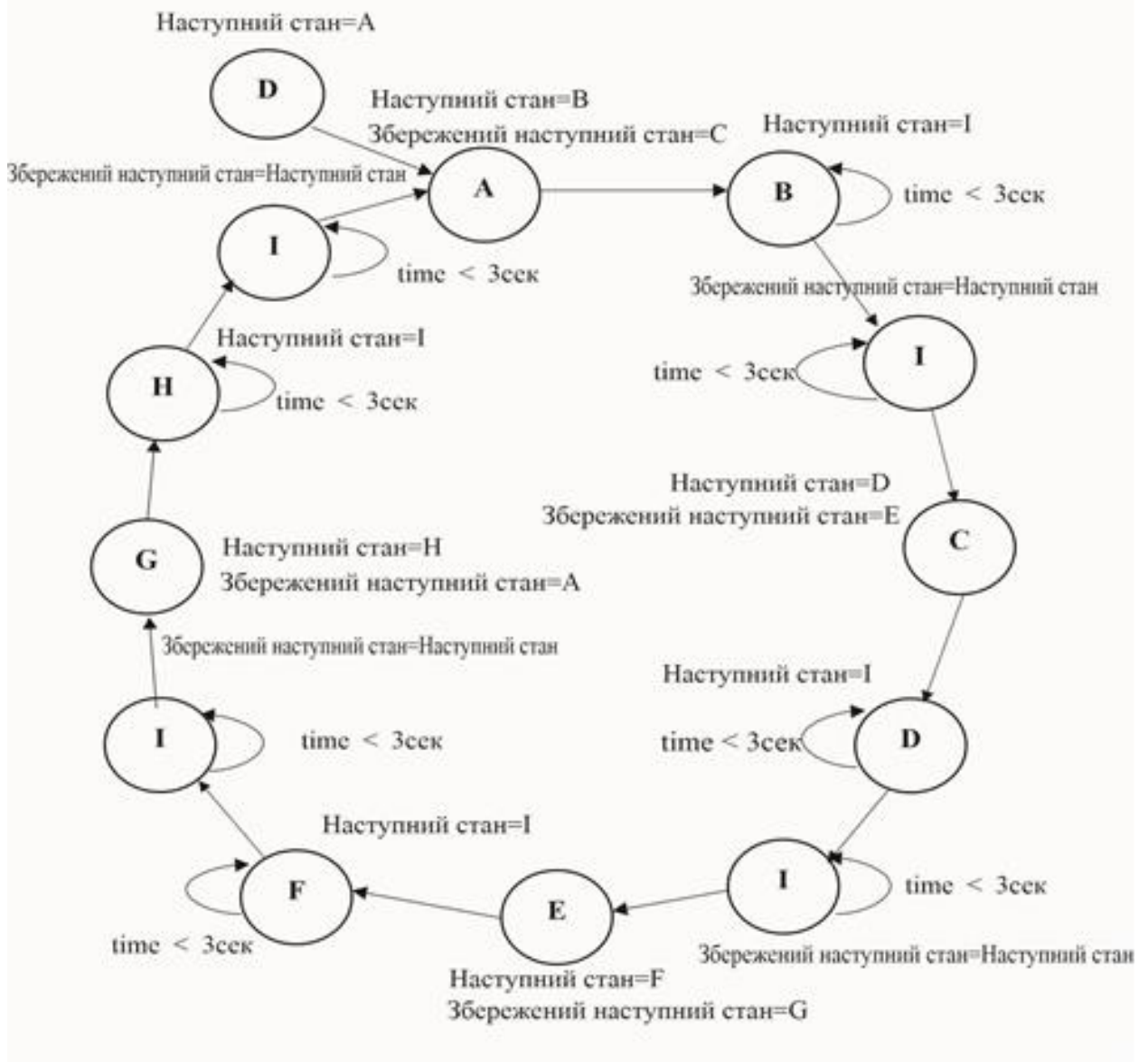


Рисунок 3.12 – Діаграма переходів станів контролера світлофору

### 3.4 Алгоритм розрахунку часу зеленого сигналу світлофора

Розрахуємо час зеленого сигналу світлофора, щоб мінімізувати час очікування на кожній дорозі перехрестя та зробити систему більш ефективною.

Алгоритм синхронізації реалізований на основі рисунка 3.13 [ 17 ]. Час необхідний для проїзду транспортного засобу через перехрестя, називається пробігом. Теоретично пробіг у випадку одного авто буде відносно більшим, ніж середній пробіг з кількох авто, оскільки він включає більший час втрат при запуску -  $e_n$  (час реакції водія + час прискорення автомобіля). Середній пробіг з двох автомобілів порівняно коротший, оскільки час втрат при запуску може



$$T = Sv + h * , \quad (3.2)$$

де, T - час зеленого сигналу світлофора;

Sv- загальний час втрат при запуску;

N - кількість виявлених транспортних засобів;

h - насичення пробігу (1,2 с /транспортний засіб).

Система буде обчислювати та регулювати зелений час залежно від навантаження на транспорт , але вона все одно передбачає мінімальний та максимальний часовий ліміт, які встановлені відповідно до 5 секунд та 25 секунд, і це розглядається як запобіжний захід для помилок при виявленнях автомобілів.

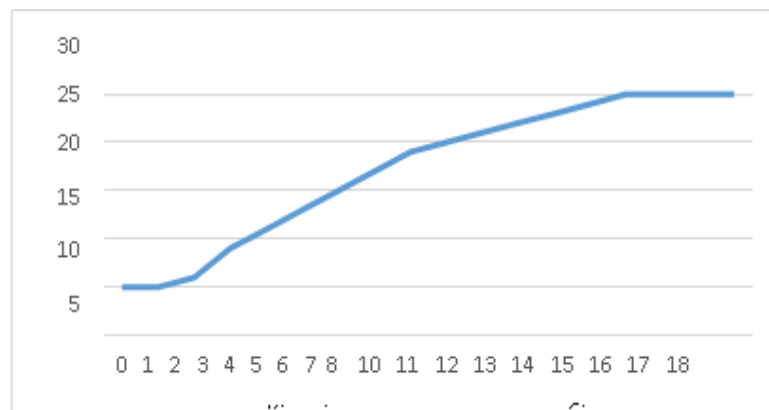


Рисунок 3.14 – Максимальний та мінімальний період зеленого світла

### 3.5 Аналіз витрат проекту

Загальний обсяг витрат, здійснених в проекті викладений у таблиці 4.1.

Таблиця 4.1 –Перелік обладнання та компонентів із ціною

№	Компоненти	Кількість	Зауваження	Ціна
1	2	3	4	5
1.	Raspberry Pi 3B +	1	<a href="https://xcraft.com.ua/raspberry-pi-3-b-plus">https://xcraft.com.ua/raspberry-pi-3-b-plus</a>	1380 грн

Кінець таблиці 4.1

1	2	3	4	5
2.	Модуль камери OV5647	1	<a href="https://diylab.com.ua/ua/p546135357-camera-kamera-dlya.html">https://diylab.com.ua/ua/p546135357-camera-kamera-dlya.html</a>	513 грн
3.	Перемички	20		30 грн
4.	Світлодіоди (червоний)	4	<a href="https://arduino.ua/product/888-svetodiod-3mm-beliy-ultrayarkii">https://arduino.ua/product/888-svetodiod-3mm-beliy-ultrayarkii</a>	4 грн
5.	Світлодіоди (жовтий)	4	<a href="https://arduino.ua/product/888-svetodiod-3mm-beliy-ultrayarkii">https://arduino.ua/product/888-svetodiod-3mm-beliy-ultrayarkii</a>	4 грн
6.	Світлодіоди (зелений)	8	<a href="https://arduino.ua/product/888-svetodiod-3mm-beliy-ultrayarkii">https://arduino.ua/product/888-svetodiod-3mm-beliy-ultrayarkii</a>	8 грн
7.	Текстоліт -5x3 см	4	<a href="https://arduino-geek.pp.ua/">https://arduino-geek.pp.ua/</a>	28 грн
Загальна вартість :				1967 грн

### 3.6 Висновки

Програмне забезпечення розробляється з двох паралельно функціонуючих двох процесів на мові Python, які виконуються на одній платі Raspberry Pi і взаємодіють через механізм черг Queue. Розроблений алгоритм функціонування і механізм їх взаємодії.

Розроблена блок-схема операцій з обробки зображень, яка включає вибір регіонів розпізнання, вирівнювання гістограми, виявлення і підрахунок автомобілів на перехресті.

Регіони розпізнавання використовуються для встановлення меж для виявлення транспортних засобів та ігнорування перешкод або шумів, що виникають поза межами. У цій роботі вони створюються на кожній доріг на перехресті.

В результаті вирівнювання гістограми зображення матиме більш рівномірний розподіл інтенсивності, покращену якість зображення, підвищену яскравість і одночасно більше деталей на зображенні з низьким освітленням.

Виявлення кількості автомобілів на перехресті реалізується за допомогою одного виклику функції `detect_common_objects` бібліотеки `Cvlib`. де використовується згортова нейронна модель глибокого навчання YOLOv4, навчена за допомогою датасета COCO.

Для проектування контролера світлофора проектування фази руху є одним із важливих кроків, щоб розділити конфлікт руху транспортних засобів на перехресті на різні фази. У запропонованій системі чотирифазні сигнали використовуються при реалізації схеми керування, а рух транспортних засобів через перехрестя руху в різні фази показано на рисунках нижче.

Була розроблена структура макетної транспортної розв'язки і діаграми переходів станів контролера світлофору.

На основі алгоритму синхронізації  $p$  розрахований час зеленого сигналу світлофора, щоб мінімізувати час очікування на кожній дорозі перехресті та зробити систему більш ефективною.

## ВИСНОВКИ

1. Результати роботи дозволяють підвищити ефективність системи функціонування світлофора за рахунок впровадження адаптивної системи, яка дозволяє ідентифікувати кількість транспорту на перехресті. Адаптивна система дозволяє гнучко обчислювати час зеленого світла, який залежить від виявленої щільності руху на смугах руху.

2. Для апаратної реалізації системи керування і розпізнавання вибраний Raspberry Pi, один найпопулярніших і найпотужніших одноплатних мікроконтролерів. У запропонованій системі отримання зображень здійснюється за допомогою модуля камери Pi. Розроблені структурна, функціональна і принципова схеми пристрою.

3. Програмна частина роботи поділена на дві основні підсистеми: автоматичну систему управління світлофором та систему виявлення і підрахунку транспортних засобів. Впровадження системи було здійснено за допомогою Raspberry Pi 3B+ і написано мовою програмування Python з використання бібліотек Open CV і Cvlib. Підхід виявлення транспортного засобу робить акцент на використанні систем згорткових нейронних мереж глибокого навчання, а саме системи YOLO. Автоматична система управління світлофором була написана мовою програмування Python на платі Raspberry Pi 3B+ з використанням автомата кінцевого стану. Дві частини виконуються, як паралельні процеси у мові Python, які взаємодіють через черги мови Python.

4. Було оцінено ефективність роботи системи розпізнавання і підрахунку кількості автомобілів, яка становить 95 відсотків, що є задовільним результатом.

У системи є деякі недоліки і ці недоліки потребують вирішення і вдосконалень, щоб система могла застосовуватися у реальній галузі.

Рекомендації щодо вдосконалення системи:

1. Коли транспортні засоби знаходяться занадто близько один до одного, система має тенденцію до неправильного обліку транспортних засобів, це впливає на точність результату та обчислить небажаний зелений час для транспортних засобів. Тому рекомендується використати декілька камер для моніторингу

					КВРКІ.170376.17.03.32 ПЗ	Арк.
						52
Зм.	Арк.	№докум.	Підпис	Дата		

кожної смуги в розв'язку замість однієї камери на всіх смугах руху. Цей підхід дозволить вирішити помилки виявлення внаслідок блокування заднього автомобіля передньою машиною та покращить ефективність виявлення та підрахунку автомобіля.

2. У системі застосовується техніка покращення зображення для підвищення яскравості та відновлення деяких деталей у зображенні з низьким освітленням, але такий підхід здатний лише покращити зображення певною мірою, це може спричинило погану реакцію системи за несприятливих умов освітлення наприклад, у нічний час, дощові дні або в погано освітленому середовищі. Тому в майбутньому рекомендується використовувати більш потужний мікроконтролер, модель для ідентифікації автомобілів в нічний час або камеру з підсвіткою для нічного бачення.

					КВРКІ.170376.17.03.32 ПЗ	Арк.
						53
Зм.	Арк.	№докум.	Підпис	Дата		

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Попович М.Г., Лозинський О.Ю., Клепиков В.Б. Електромеханічні системи автоматичного керування та електроприводи: Навч. посібник / За ред. П.Г. Поповича, О.Ю. Лозинського. К.: Либідь, 2005. 680 с.
2. Голуб А.П., Кузнецов Б.І., Опришко І.О., Соляник В.П. Системи керування електроприводами: Навчальний посібник. К.: НМК ВО, 1992, 352с.
3. Гаврилов Е.В., Дмитриченко М.Ф., Доля В.К., Лановий О.Т., І.Е. Лінник, Поліщук В.П. Організація дорожнього руху. Підручник. Книга IV / К., 2007. 452 с.
4. Єресов В.І., Кулай П.Л. Автоматизовані системи управління дорожнім рухом - Посібник для студентів спеціальності 5.100.401 “Організація дорожнього руху”. Полтава, Друкарня УМВС, 2004,152 с.
5. Кременец Ю.А. Технические средства организации дорожного движения: Учебник для вузов. М.: Транспорт, 1990. 255 с.
6. Капитанов В.Г., Хилажев Е.Б. Управление транспортными потоками в городах – М.: Транспорт, 1985. – 94 с.
7. Самойлов Д.С, Юдин В.А., Рушевський П.В. Организация и безопасность дорожного движения М.: Высш. школа. 1981. 256 с.
8. Кременец Ю.А. Технические средства организации дорожного движения– М.: Транспорт, 1995. 255 с.
9. Засоби автоматизації. URL: [http://www.wikiznanie.ru/ruwz/index.php/Алгоритмы\\_адаптивного\\_регулирования\\_светофорной\\_сигнализации](http://www.wikiznanie.ru/ruwz/index.php/Алгоритмы_адаптивного_регулирования_светофорной_сигнализации) (дата звернення: 15.05.2021).
10. Дзюба О.П., Павловська О.М. Розробка концепції динамічного адаптивного світлофорного регулювання дорожнього руху на перехрестях безпека руху автомобільний транспорт Науково-виробничий журнал № 2 (232) березень – квітень 2013 р.
11. Sinhmar, P. Intelligent traffic light and density control using IR sensors and microcontroller. International journal of advanced technology & engineering research (IJATER), 2(2), pp. 30-35, 2012.

					КВРКІ.170376.17.03.32 ПЗ	Арк. 54
Зм.	Арк.	№докум.	Підпис	Дата		



23. About OpenCV. – Електрон. дан. –URL: <https://opencv.org/about/>, (дата звернення: 15.05.2021).
24. Распознавание образов с использованием OpenCV Материалы блога – URL: <http://recog.ru> (дата звернення: 15.05.2021).
25. OpenCV. URL: <http://opencv.org/>(дата звернення: 15.05.2021).
26. Комп'ютерний зір. —URL: [https://en.wikipedia.org/wiki/Computer\\_vision](https://en.wikipedia.org/wiki/Computer_vision), (дата звернення: 15.05.2021).
27. A Gentle Introduction to Computer Vision. – Електрон. дан. – URL: <https://machinelearningmastery.com/what-is-computer-vision/>, (дата звернення: 15.05.2021).
28. Рідкокаша А.А., Голдер К.К. Основи систем штучного інтелекту. Навчальний посібник Черкаси: Відлуння-Плюс, 2002. - 240с.
29. Руденко О.Г., Бодяньський Є.В. Штучні нейронні мережі. Навч. посіб. / – Харків: ТОВ "Компанія СМІТ", 2006. – 404 с.
30. Гудфелов І.А. Глибоке навчання МІТ Press. – 2016. – №2. – с. 50.
31. Neural Network fundamentals. –URL: <https://medium.com/datadriveninvestor/neural-network-fundamentals-1956a3000c24>, (дата звернення: 15.05.2021).
32. A Comprehensive Guide to Types of Neural Networks. –URL: <https://www.digitalvidya.com/blog/types-of-neural-networks/>, (дата звернення: 15.05.2021).
33. Getting Started with R-CNN, Fast R-CNN, and Faster R-CNN. –URL: <https://www.mathworks.com/help/vision/ug/getting-started-with-r-cnn-fast-r-cnn-and-faster-r-cnn.html>, (дата звернення: 15.05.2021).
34. YOLO: Real-Time Object Detection. –URL <https://pjreddie.com/darknet/yolov2/>, (дата звернення: 15.05.2021).
35. YOLOv3: An Incremental Improvement. – URL: <https://pjreddie.com/media/files/papers/YOLOv3.pdf> (дата звернення: 15.05.2021).
36. Практичний Python 3 для початківців. –URL: <https://pythonworld.ru/samouchitel-python/>, (дата звернення: 15.05.2021).

37. Опис Python. – URL : <https://uk.wikipedia.org/wiki/Python>, (дата звернення: 15.05.2021).

38. Декоратори Python. – URL: <https://tproger.ru/translations/demystifying-decorators-in-python>, (дата звернення: 15.05.2021).

39. Raspberry Pi Schematics URL : <https://www.raspberrypi.org/documentation/hardware/raspberrypi/schematics/README.md>(дата звернення: 15.05.2021).

40. ГОСТ 19.701-90 ЕСКД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.

41. ГОСТ 19.003-80 ЕСКД. Схемы алгоритмов и программ. Обозначения условные графические.

					КВРКІ.170376.17.03.32 ПЗ	Арк.
						57
Зм.	Арк.	№докум.	Підпис	Дата		

## ДОДАТОК А

### (обов'язковий)

#### Фрагмент вихідного коду програми підрахунку кількості автомобілів

```
import cv2
import time
import numpy as np
import os
import cvlib as cv
from cvlib.object_detection import draw_bbox
winname = 'FYP2-IMG_PROC'
cv2.namedWindow(winname)
cap = cv2.VideoCapture(0) ##0 is use main cam
cap.set (cv2.CAP_PROP_FRAME_WIDTH,1280)
cap.set (cv2.CAP_PROP_FRAME_HEIGHT, 800)
_, frame1 = cap.read()
cv2.imshow(winname, frame1)
Mat_ctrl = []
count_point = 0
count_lane = 0
roi_corners1 = np.array([[x1_1, y1_1], (x1_2, y1_2), (x1_3, y1_3), (x1_4,
y1_4)]], dtype=np.int32)
roi_corners2 = np.array([[x2_1, y2_1), (x2_2, y2_2), (x2_3, y2_3), (x2_4,
y2_4)]], dtype=np.int32)
roi_corners3 = np.array([[x3_1, y3_1), (x3_2, y3_2), (x3_3, y3_3), (x3_4,
y3_4)]], dtype=np.int32)
roi_corners4 = np.array([[x4_1, y4_1), (x4_2, y4_2), (x4_3, y4_3), (x4_4,
y4_4)]], dtype=np.int32)
def Cut_Roi( image, roi_corners):
    mask = np.ones(image.shape, dtype=np.uint8)
    mask.fill(255)
    # points to be cropped
    # fill the ROI into the mask
    cv2.fillPoly(mask, roi_corners, 0)
    # The mask image
    cv2.imwrite('image_masked.png', mask)
    # applying th mask to original image
    masked_image = cv2.bitwise_or(image, mask)
    return masked_image

def formula (car_num):
```

```

reaction_time =2.5
saturation_flow = 1.2
Start_up_loss = 0
x = 0
if car_num != 0 :
    while x != car_num:
        x +=1
        Start_up_loss = (reaction_time)*(pow(0.75, (x-
1)))+Start_up_loss
    g_time = Start up loss + saturation flow*car num
else :
    g_time = 0
return int (g_time)
def enhance (hsv_in):
    copyImage = hsv_in.copy()
    h, w = hsv_in.shape[:2]
    mask = np.zeros([h+2, w+2], np.uint8)
    cv2.floodFill(hsv_in, mask:, (0,0),255)
    hsv_in_inv = cv2.bitwise_not(hsv_in)
    hsv_in = copyImage | hsv_in_inv
    return hsv_in

def equalize_hist (image) :
    blue = image[:, :,0]
    blue = cv2.equalizeHist(blue)
    green = image[:, :,1]
    green = cv2.equalizeHist(green)
    red = image[:, :,2]
    red = cv2.equalizeHist(red)
    merge= cv2.merge((blue, green, red))
    return merge

def Traf_1():
    global T_A,T_B,T_C,T_D,que_r,G_T
    ROI_trA = Cut_Roi(main_frame,roi_corners1)
    ROI_trA = equalize_hist(ROI_trA)
    in_hsv_frame = cv2.cvtColor (ROI_trA, cv2.COLOR_BGR2HSV)
    T_A = Count_Car (in_hsv_frame)
    G_T = formula(T_A )
    que_r = 0
    que.transmit(G_T)
    return que_r

def Traf_2 () :

```

```

global T_A, T_B, T_C, T_D , que_r,G_T
ROI_trB = Cut_Roi(main_frame,roi_corners2)
ROI_trB = equalize_hist(ROI_trB)
in_hsv_frame = cv2.cvtColor(ROI_trB, cv2.COLOR_BGR2HSV)
T_B = Count_Car (in_hsv_frame)
G_T = formula(T_B )
que_r = 0
que.transmit(G_T)
return que_r

def traf_3 () :
    global T_A, T_B, T_C, T_D , que_r,G_T
    ROI_trC = Cut_Roi(main_frame,roi_corners3)
    ROI_trC = equalize_hist(ROI_trC)
    in_nsv_frame = cv2.cvtColor(ROI_trC , cv2.COLOR_3GR2HSV)
    T_C = Count_Car (in_hsv_frame)
    G_T = formula(T_C )
    que_r = 0
    que.transmit(G_T)
    return que_r

def Traf_4 () :
    global T_A, T_B, X_C, X_D , uarc_r,G_T
    ROI_trD = Cut_Roi(main_frame,roi_corners3)
    ROI_trD = equalize_hist(ROI_trD)
    in_hsv_frame = cv2.cvcColor(ROI_trD, cv2.COLOR_BGR2HSV)
    T_D = Count_Car (in_hsv_frame)
    G_T = formula(T_D )
    que_r = 0
    que.transmit(G_T)
    return que_r

def Reset () :
    global T_A , X_B , T_C , X_D,que_r
    k=27
    que.transmit(0)
    que_r = 0
    return k

def Count_Car (draw_bboxim) :
    bbox, label, conf = cv.detect_common_objects(draw_bboxim)
    output_image = draw_bbox(draw_bboxim, bbox, label, conf)
    return label.count('car'))

def Main_Control(que_r):

```

```

switch={1:Traf_1, #send time A
        2:Traf_2, #send time B
        3:Traf_3, #send time C
        4:Traf_4, #send time D
        0:Reset,}

return switch.get(que_r,default)()

def main() :
    global T_A, T_B, T_C, T_D
    G_T=0
    CaptureMod = True
    ref_rate = 2
    global frame1,main_frame
    global View1 , View2 , View3 , View4, View5
    global que_r
    que_r = 0
    while cap.isOpened:
        if CaptureMod = True:
            ret,frame1 = cap.read()
            cv2.imshow(winname,frame1)
            cv2.moveWindow('FYP2-IMG_PROCL, 0, 0)
            fyp.lanes.ROI_show(frame1,count_lane)
        elif CaptureMod = False:
            ret,frame1 = cap.read()
            _, main_frame = cap.read()

            k = cv2.waitKey(ref_rate)
            if k == 27 or n==27:
                cap.release()
                cv2.destroyAllWindows ()
                que.main(False)
                break

            elif count_lane = 4 and CaptureMod = True:
                CaptureMod = False
                cv2.destroyWindow('FYP2-IMG_PROC')
                que.main(True)

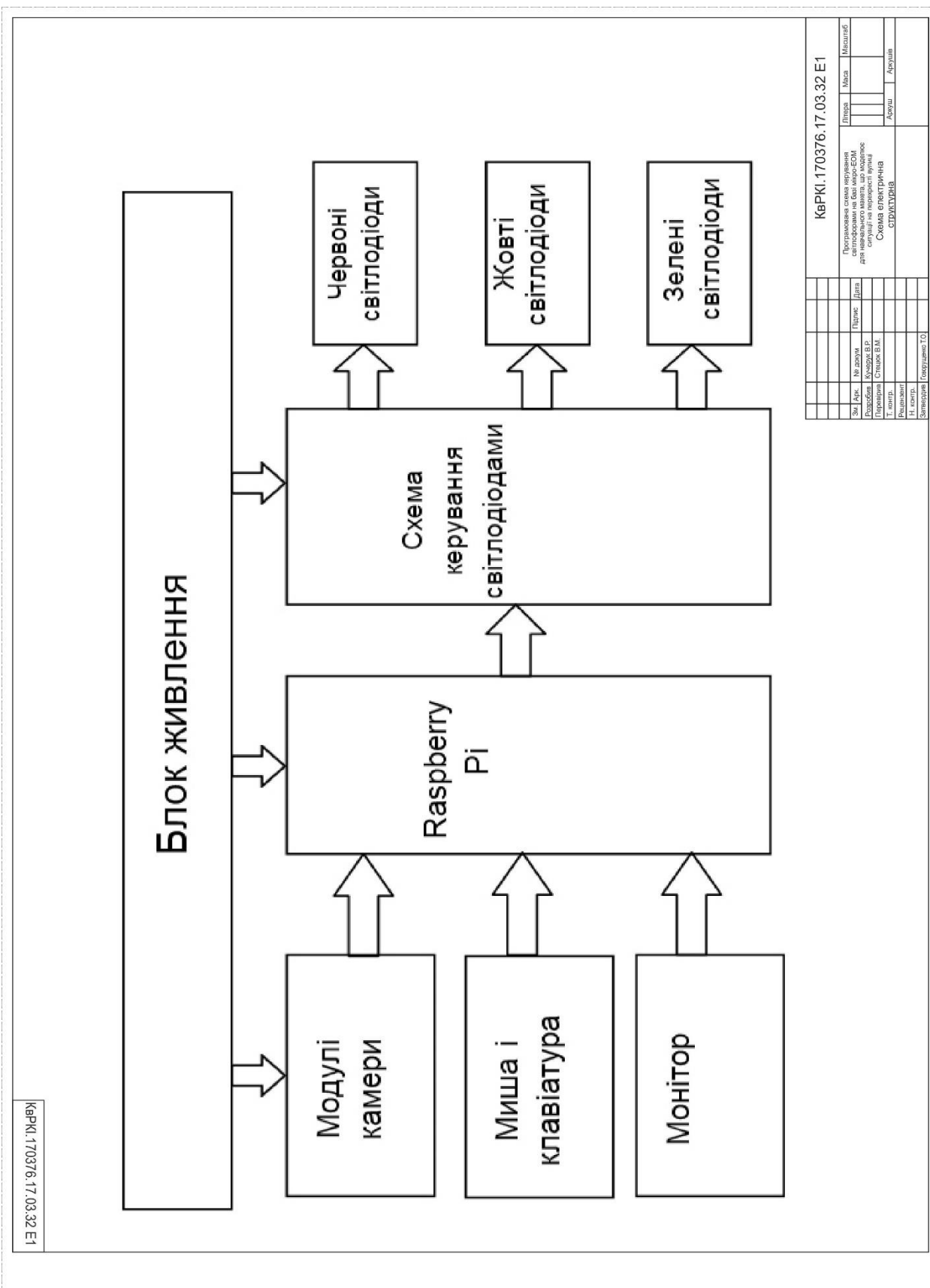
                que.transmit(1)
                ref_rate=60

if __name__=="__main__":
    main()

```

**ДОДАТОК Б**  
(обов'яз'язковий)

Копія креслення «Схема електрична структурна»



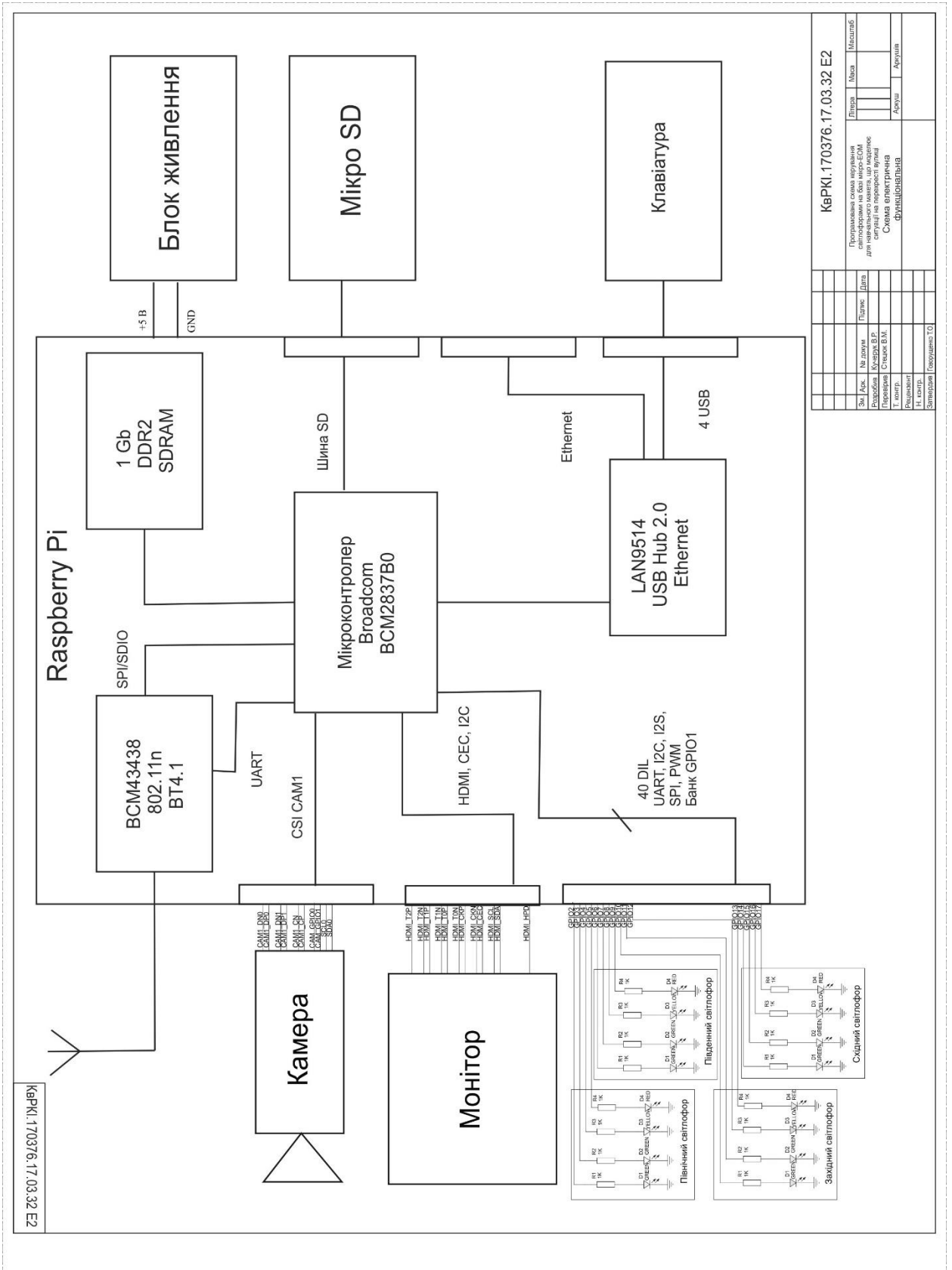
КВРКІ.170376.17.03.32.Е1

КВРКІ.170376.17.03.32.Е1		Підпис	Місяць
Продумана схема керування світлодіодами на базі мікро-ЕОМ для навчального заняття, що моделює роботу системи керування світлодіодами		Дата	Рік
Схема електрична СТРУКТУРНА		Архив	Архив
Зм. №	№ пром.	Підпис	Дата
Розробив	Коректував		
Перевірив	Стяжок В.М.		
Т. конст.			
Редакційн.			
Н. конст.			
Семантика	Сторінки	ТО	

# ДОДАТОК В

## (обов'яз'язковий)

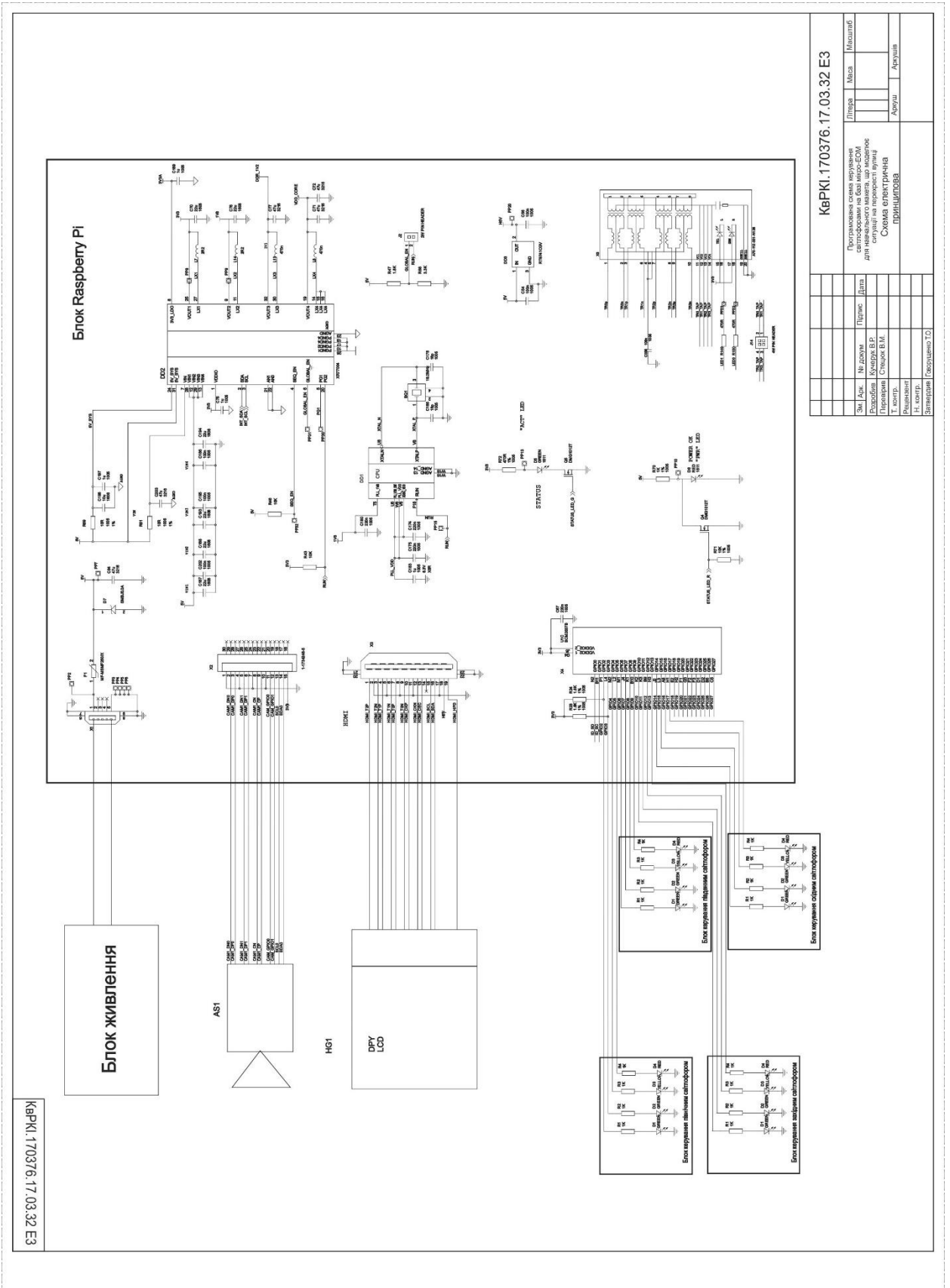
Копія креслення «Схема електрична функціональна»



# ДОДАТОК Г

## (обов'яз'язковий)

Копія креслення «Схема електрична принципова»



Ім'я користувача:  
Кафедра КІ

ID перевірки:  
1008235466

Дата перевірки:  
08.06.2021 20:38:55 EEST

Тип перевірки:  
Doc vs Internet + Library

Дата звіту:  
08.06.2021 20:40:04 EEST

ID користувача:  
100005591

Назва документа: Кучерук\_Програмована схема керування світлофорами на базі мікро-ЕОМ для

Кількість сторінок: 68 Кількість слів: 9362 Кількість символів: 72315 Розмір файлу: 5.45 MB ID файлу: 1008308437

## 8.3% Схожість

Найбільша схожість: 3.09% з джерелом з Бібліотеки (ID файлу: 1008214668)

5.79% Джерела з Інтернету 458

Сторінка 70

3.67% Джерела з Бібліотеки 94

Сторінка 72

## 0% Цитат

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

## 0% Вилучень

Немає вилучених джерел

## Модифікації

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

Замінені символи 34

## Anti-Plagiarism v-15.257

Максимальное совпадение с одним документом 3.0%

Словари проверки: en\_US, ru\_RU, ua\_UA, Ошибок в документах: 9%

ID: 92767 Название: Програмова схема керування світлофорами на базі мікро-ЕОМ, для навчального макета, що моделює ситуації на перехресті вулиці Добавлено в БД: 2021-06-08 Авторы: Кучерук В.Р. Руководитель: Стецюк В.М. Консультанты: Олоненты:	Документ		Суммарное совпадение по Базе Данных	
	Символы	Лексемы	Символы	Лексемы
	57229	497	2557 (4%)	38 (8%)

Источник плагиата

ID	Описание	Наличие плагиата в документе	
		Символы	Лексемы

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник Кучерук Владислав Русланович

Тема Програмована схема керування світлофорами на базі мікро-ЕОМ для навчального макета, що моделює ситуації на перехресті вулиці

Спеціальність 123 Комп'ютерна інженерія

**Обсяг кваліфікаційної роботи:**

кількість листів креслень 3; кількість сторінок записки 58

1. Короткий зміст КР та прийнятих рішень В рамках кваліфікаційної роботи метою є розробка адаптивної системи керування світлофорами для підвищення ефективності системи за рахунок зменшення часу очікування кожної смуги руху транспортних засобів та максимізації потоку транспортних засобів через перехрестя.
2. Висновок про відповідність КР дипломному завданню Кваліфікаційна робота у повній мірі відповідає поставленому завданню як в теоретичній, так і в практичній частині даної роботи
3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому, теоретичному, розділі кваліфікаційної роботи досить широко розглянуті методи вирішення задачі побудови адаптивних світлофорів, були проаналізовані варіанти їх реалізації, недоліки і переваги. Зроблено висновок про доцільність і перспективність використання комп'ютерного зору в даній темі. У наступному розділі було здійснено обґрунтування структури системи керування світлофорами та схеми функціонування, а також розроблені функціональна та принципова системи на основі мікро-ЕОМ Raspberry Pi. В третьому розділі розроблені алгоритми і реалізоване програмне забезпечення системи керування світлофорами. Програмне забезпечення розроблено з двох паралельно функціонуючих підсистем на мові Python, автоматичної системи контролера світлофора та систему виявлення і підрахунку транспортних засобів. Проведена оцінка вартості проекту. В загальному усі розділи відповідають завданню та містять сучасні методи вирішення поставлених завдань.
4. Позитивні сторони проекту Дипломний проект відповідає сучасним вимогам до проектування систем керування та містить ряд інноваційних рішень, зокрема, з точки зору реалізації адаптивних систем керування світлофорами. Зокрема можна виділити використання при розпізнаванні і підрахунку автомобілів сучасної бібліотеки комп'ютерного зору cvlib на основі системи згорткових нейронних мереж глибокого навчання YOLO4, оскільки на сьогоднішній день таке рішення є досить затребуваним і має значні перспективи. Також важливим позитивним аспектом кваліфікаційної роботи є реалізація системи керування на основі двох підсистем, що працюють паралельно. Що дає змогу масштабувати дане рішення при використанні його на реальних перехрестях

5. Негативні сторони роботи Характеристики використаної для керування мікро-ЕОМ Raspberry 3В+, зокрема обсяг оперативної пам'яті та продуктивність процесора, можуть бути недостатніми при реальному, немакетному, керування світлофорами. Також можливі проблеми у нічний час з ідентифікацією машин.

6. Оцінка графічного оформлення та пояснювальної записки кваліфікаційної роботи Графічне оформлення виконане відповідно до суті кваліфікаційної роботи. На першому листі креслення відображено структурна схема системи керування світлофорами. На другому листі креслення реалізована функціональна схема системи керування світлофорами. На третьому листі креслення на основі схеми Raspberry PI розроблена принципова схема системи керування світлофорами. В загальному графічне оформлення виконане на належному рівні. Пояснювальна записка відповідає задекларованим нормам для її оформлення.

7. Відгук про роботу в цілому В загальному кваліфікаційна робота виконана на високому рівні. Весь матеріал роботи структурований, чіткий та послідовний. Усі розділи роботи йдуть у вірній послідовності, що дозволяє чітко розуміти викладений матеріал в рамках даної кваліфікаційної роботи. Графічний матеріал дозволяє наочно побачити доцільність та ефективність рішень, які були прийняті за основу при проектуванні системи керування світлофорами.

8. Інші зауваження

9. Оцінка кваліфікаційної роботи Розглянувши позитивні та негативні сторони представленої кваліфікаційної роботи, можна зробити висновок, що він заслуговує оцінку «відмінно».

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

Бедрашюк Л.А. зав. каф. ІТЗ

« 18 » 06 2021 р.

(підпис)

Завідувачу кафедри КІСП  
д-ру техн. наук, проф. Говорущенко Т. О.

Кучерук В.Р.

ІІБ здобувача вищої освіти

ФПКТС, 4 курсу, групи КІ-17-3

### 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 плагіатоповіщений (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.

17.06.21

дата

  
підпис

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ  
КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА СИСТЕМНОГО ПРОГРАМУВАННЯ  
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

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

Назва: Програмована схема керування світлофорами на базі мікро-ЕОМ для навчального макета, що моделює ситуації на перехресті вулиці

Автор: Кучерук Владислав Русланович

Спеціальність: 123 – Комп'ютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Стецюк Василь Миколайович

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 укриття 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 розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
- 2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
- 3) окремі виявлені збіги є загальноживаними фразами або виразами, про що свідчить посилання системи на збіг з джерелами на один фрагмент речення;
- 4) в якості запозичень в окремих місцях системою зафіксовано послідовності та стандарти які йдуть згідно оформлення роботи;
- 5) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 8,3% і адресується до 552 першоджерел, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КІСП

В. М. Стецюк

С. М. Лисенко

Т. О. Говорушенко