

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії

(повна назва факультету)

Кафедра комп'ютерних систем та мереж

(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Комп'ютеризована паркувальна система для транспортних засобів.

Виконав(ла): студент(ка) IV курсу, групи СІс-43

спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

| | |
|----------------------|------------------------|
| | <u>Ярмусь О.С.</u> |
| (підпис) | (прізвище та ініціали) |
| Керівник | <u>Осухівська Г.М.</u> |
| (підпис) | (прізвище та ініціали) |
| Нормоконтроль | <u>Луцик Н.С.</u> |
| (підпис) | (прізвище та ініціали) |
| Завідувач кафедри | <u>Осухівська Г.М.</u> |
| (підпис) | (прізвище та ініціали) |
| Рецензент | <u>Млинко Б.Б.</u> |
| (підпис) | (прізвище та ініціали) |

Тернопіль
2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Осухівська Г.М.

(підпис)

(прізвище та ініціали)

« » 2022

р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього
ступеня

бакалавр

(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

студенту Ярмуся О.С.

(прізвище, ім'я, по батькові)

1. Тема роботи Комп'ютеризована паркувальна система для транспортних засобів.

Керівник роботи Осухівська Г.М.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 23 » 03 2022 року № 4/7-180

2. Термін подання студентом завершеної роботи _____

3. Вихідні дані до роботи Технічне завдання

4. Зміст роботи (перелік питань, які потрібно розробити)

Аналіз вимог до комп'ютеризованої паркувальної системи, Аналіз можливих рішень поставленого завдання, Розробка узагальненої структури комп'ютеризованої паркувальної системи, Обґрунтування вибору апаратного забезпечення для проектування комп'ютеризованої паркувальної системи, Обґрунтування вибору програмного забезпечення для проектування комп'ютеризованої паркувальної системи, Проєктування комп'ютеризованої паркувальної системи, Реалізація проєктних рішень, Тестування комп'ютеризованої паркувальної системи, Безпека життєдіяльності, основи охорони праці

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)

Структурна схема

Блок-схема роботи комп'ютеризованої паркувальної системи

Електрична принципова схема комп'ютеризованої паркувальної системи

Алгоритм роботи програми

6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|---|---|----------------|------------------|
| | | завдання видав | завдання прийняв |
| <i>Розділ 4 Безпека життєдіяльності, основи охорони праці</i> | <i>Лазарюк В.В. доцент, к.т.н.</i> | 25.03.2022 | |
| | | | |
| | | | |

7. Дата видачі завдання 25.03.2022р.

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів роботи | Термін виконання етапів роботи | Примітка |
|-------|---|--------------------------------|-----------------|
| 1 | <i>Аналіз вимог до комп'ютеризованої паркувальної системи</i> | <i>25.03.2022–27.03.2022</i> | <i>Виконано</i> |
| 2 | <i>Аналіз можливих рішень поставленого завдання</i> | <i>28.03.2022–02.04.2022</i> | <i>Виконано</i> |
| 3 | <i>Розробка узагальненої структури комп'ютеризованої паркувальної системи</i> | <i>03.04.2022–15.04.2022</i> | <i>Виконано</i> |
| 4 | <i>Обґрунтування вибору апаратного забезпечення для проєктування комп'ютеризованої паркувальної системи.</i> | <i>16.04.2022–20.04.2022</i> | <i>Виконано</i> |
| 5 | <i>Обґрунтування вибору програмного забезпечення для проєктування комп'ютеризованої паркувальної системи.</i> | <i>21.04.2022–30.04.2022</i> | <i>Виконано</i> |
| 6 | <i>Проєктування комп'ютеризованої паркувальної системи</i> | <i>01.04.2022–20.05.2022–</i> | <i>Виконано</i> |
| 7 | <i>Реалізація проєктних рішень</i> | <i>21.05.2022–03.06.2022</i> | <i>Виконано</i> |
| 8 | <i>Тестування комп'ютеризованої паркувальної системи</i> | <i>04.06.2022–07.06.2022</i> | <i>Виконано</i> |
| 9 | <i>Безпека життєдіяльності, основи охорони праці</i> | <i>08.06.2022–15.06.2022</i> | <i>Виконано</i> |
| 10 | <i>Підготовка до захисту</i> | <i>16.06.2022–23.06.2022</i> | <i>Виконано</i> |
| | | | |
| | | | |
| | | | |

Студент

_____ (підпис)

Ярмусь О.С.

_____ (прізвище та ініціали)

Керівник роботи

_____ (підпис)

Осухівська Г.М.

_____ (прізвище та ініціали)

АНОТАЦІЯ

Комп'ютеризована паркувальна система для транспортних засобів. // Кваліфікаційна робота бакалавра // Ярмусь Олександр Сергійович// Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних систем та мереж, група СІс-43 // Тернопіль, 2022 // с. – 66, рис. – 22, табл. – 4, кресл. – 4, додат. – 3, бібліогр. – 13.

Ключові слова: комп'ютеризована паркувальна система, Arduino UNO, ультразвуковий датчик відстані, HC-SR04 текстовий екран, відтворювач звуку, HPA17A, LCD дисплей.

В кваліфікаційній роботі бакалавра розроблено комп'ютеризовану паркувальну систему для транспортних засобів на базі мікроконтролера Arduino UNO.

Кваліфікаційна робота складається з чотирьох розділів.

У першому розділі проводиться аналіз технічного завдання, а саме вимог до комп'ютеризованої системи та аналіз можливих рішень.

В другому розділі описується процес проектування та реалізації проекту, як вбудованої системи.

В третьому розділі приводиться розробка програмного забезпечення для функціонування пристрою. Також проводиться тестування комп'ютеризованої системи.

Четвертий розділ описує безпеку життєдіяльності та основи охорони праці.

ABSTRACT

Computerized parking system for vehicles. // Bachelor's thesis // Yarmus Oleksandr Serhiiovych // Ternopil National Technical University named after Ivan Pulyuy, Faculty of Computer Information Systems and Software Engineering, Department of Computer Systems and Networks, SIS-43 Group // Ternopil, 2022 // with. - 66, fig. - 22, table. - 4, chair. - 4, appendix. - 3, bibliogr. - 13.

Keywords: computerized parking system, Arduino UNO, ultrasonic distance sensor, HC-SR04 text screen, audio player, HPA17A, LCD display.

In the qualification work of the bachelor, a computerized parking system for vehicles based on the Arduino UNO microcontroller was developed.

Qualification work consists of four sections.

The first section analyzes the technical task, namely the requirements for a computerized system and the analysis of possible solutions.

The second section describes the process of designing and implementing a project as an embedded system.

The third section describes the development of software for the operation of the device. The computer system is also being tested.

The fourth section describes safety and basics of labor protection.

ЗМІСТ

| | |
|---|----|
| СПИСОК СКОРОЧЕНЬ | 7 |
| ВСТУП | 8 |
| РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ | 10 |
| 1.1 Аналіз вимог до комп'ютеризованої паркувальної системи | 10 |
| 1.2 Аналіз можливих рішень поставленого завдання | 11 |
| РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА | 16 |
| 2.1 Розробка узагальненої структури комп'ютеризованої паркувальної системи | 16 |
| 2.2 Обґрунтування вибору апаратного забезпечення для проектування комп'ютеризованої паркувальної системи | 17 |
| 2.3 Обґрунтування вибору програмного забезпечення для проектування комп'ютеризованої паркувальної системи | 25 |
| 2.4 Проектування комп'ютеризованої паркувальної системи | 31 |
| РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА | 38 |
| 3.1 Реалізація проєктних рішень | 38 |
| 3.2 Тестування комп'ютеризованої паркувальної системи | 45 |
| РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ | 46 |
| 4.1. Значення адаптації в трудовому процесі. | 46 |
| 4.2. Розрахунок та розробка схеми захисного занулення | 47 |
| 4.3. Перша медична допомога при ураженні електричним струмом | 50 |
| ВИСНОВКИ | 53 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ | 54 |
| ДОДАТКИ | |

| | | | | | | | | | |
|------------------|-----------------|--|--|--|---|----------------------------------|----|----------------|--|
| | | | | | | КС КРБ 123.237.00.00 ПЗ | | | |
| <i>Змн.</i> | | | | | | | | | |
| <i>Розроб.</i> | Ярмусь О.С. | | | | Комп'ютеризована паркувальна система для транспортних засобів. | <i>Літ.</i> | | <i>Акрушів</i> | |
| <i>Перевір.</i> | Осухівська Г.М. | | | | | 6 | 55 | | |
| <i>Реценз.</i> | Млинко Б.Б. | | | | | <i>ТНТУ, каф. КС, гр. СІс-43</i> | | | |
| <i>Н. контр.</i> | Луцик Н.С. | | | | | | | | |
| <i>Затверд.</i> | Осухівська Г.М. | | | | | | | | |

СПИСОК СКОРОЧЕНЬ

КПС – комп'ютеризована паркувальна система.

МК – мікроконтролер.

АПС – автоматизована (автомобільна) система паркування.

ЕБУ – електронний блок управління.

ПУЕ – правила улаштування електроустановок.

ОС – операційна система.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 7 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

ВСТУП

Люди постійно працюють над створенням нових винаходів, які можуть покращити їхнє життя. Зараз ми маємо усі можливі ресурси для того, щоб створити щось нове. На даний момент технології використовуються у будь-якій сфері життя для покращення роботи. Також необхідно зважати не лише на покращення життя, а і на його безпеку. Разом із новими технологіями створюються нові небезпеки. Тому на це варто звернути увагу. У цій роботі описано створення і програмування незамінного приладу кожного водія – паркувальної комп'ютеризованої системи.

В основі комп'ютеризованої паркувальної системи (КПС) є радар, який використовується для вимірювання відстані між власним транспортним засобом та іншими об'єктами. Принцип роботи такої КС заснований на зборі інформації про наближені об'єкти.

Завдяки такій системі користувач буде попереджений про ймовірність зіткнення з об'єктом і допоможе уникнути аварії. Така комп'ютеризована система має стати обов'язковим атрибутом того, хто за кермом і, найцікавіше, що його можна створити самостійно, маючи спеціальні датчики. У даному випадку паркувальна комп'ютеризована система створена на платі Arduino UNO – флагманській платформі для розробки на базі мікроконтролера (МК) ATmega 328P.

Основні завдання кваліфікаційної роботи:

- розробити комп'ютеризовану паркувальну систему для транспортних засобів, використовуючи ультразвуковий датчик відстані;
- узагальнити і вдосконалити знання з програмування у ході роботи з середовищем Arduino IDE;
- у тестовому режимі визначити роботоздатність системи.

Вибрана тема безперечно є актуальною. Звичайно, можна навчитися паркуватися, але людина завжди може не доглянути та в'їхати в щось чи в когось, на відмінну від запрограмованої КПС, яка би сповіщала у таких ситуаціях. Моментів, яких людина може не врахувати під час паркування багато, тому

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 8 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

інколи людині потрібно мати більше інформації, щоб правильно скоординувати свої рухи та коректно припаркуватися. Завдяки комп'ютеризованій паркувальній системі для транспортних засобів цієї проблеми можна уникнути.

| | | | | | | |
|-------------|-------------|-----------------|---------------|-------------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 9 |
| <i>Змн.</i> | <i>Арк.</i> | <i>№ докум.</i> | <i>Підпис</i> | <i>Дата</i> | | |

РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ

1.1 Аналіз вимог до комп'ютеризованої паркувальної системи

Кожна система, яка розробляється, повинна відповідати конкретним вимогам, відповідно завданням, які поставлені. Комп'ютеризована паркувальна система для транспортних засобів повинна відповідати таким вимогам:

- робота в реальному часі;
- використовувати ультразвукові датчики для вимірювання дистанції до найближчих об'єктів;
- КПС повинна відповідати на запити відповідно до розробленого внутрішнього набору правил роботи та команд;
- висока частота опитування датчиків для забезпечення швидкості показників та сигналів;
- точно вимірювати відстань до об'єкту (100-500мм);
- видавати застережливі звуки з різним інтервалом (200-300мс) відносно відстані;
- вчасно повідомляти про дальність об'єкта;
- виводити повідомлення на екран про відстань до перешкоди;
- миготіти різними за кольором світлодіодами, відносно відстані;
- бути енергоефективною.

До основних вимог до конструювання комп'ютеризованих паркувальних систем можна віднести надійність та якість. Що стосується погодних умов – комп'ютеризована паркувальна система повинна працювати, як в жару (+60°C), так і в холод (-30°C).

Також, однією із головних вимог є те, щоб КПС відповідала вимогам технічного завдання (Додаток А). Для дотримання вище поставлених вимог, необхідно визначитися з особливостями розроблювального пристрою, провести

| | | | | | | | |
|-----------|-----------------|--|--|--|---|---------------------------|---------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | | |
| Змн. | | | | | | | |
| Розроб. | Ярмусь О.С. | | | | РОЗДІЛ 1 АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ | Літ. | Акрушів |
| Перевір. | Осухівська Г.М. | | | | | 10 | 6 |
| Реценз. | Млинко Б.Б. | | | | | ТНТУ, каф. КС, гр. СІс-43 | |
| Н. контр. | Луцик Н.С. | | | | | | |
| Затверд. | Осухівська Г.М. | | | | | | |

огляд апаратних пристроїв та принцип побудови структурних схем, виявити та усунути недоліки.

1.2 Аналіз можливих рішень поставленого завдання

Комп'ютеризована паркувальна система (КПС) – це пристрій, який використовує ультразвукові датчики для виявлення перешкод біля переднього і заднього бамперів, які важко побачити безпосередньо з водійського сидіння, а також сповіщає про ситуацію за допомогою сигналізації та РК-індикаторів для підтримки керування на паркувальних майданчиках з можливими різноманітними перешкодами та з обмеженими розмірами. Поки відстань між бампером і перешкодою не досягне приблизно 50см, індикатор буде вказувати на ситуацію, а коли відстань від перешкоди буде менше 50см, пролунає звуковий сигнал, щоб привернути додаткову увагу [1].

Такі системи можуть використовуватися як складові автоматизованих систем паркування [1]. У багатьох проєктах впровадження автоматизованої системи паркування дозволило заощадити 50% необхідного простору в порівнянні зі звичайним паркуванням.

Автоматизовані (автомобільні) системи паркування (APS) – це система, призначена для мінімізації площі або обсягу, необхідного для паркування автомобілів, які забезпечують паркування автомобілів на кількох рівнях, розміщених вертикально, наприклад, в багатоповерхових гаражах щоб максимізувати кількість паркувальних місць при мінімізації використання землі. APS, однак, використовує механічну систему для транспортування автомобілів до та з паркувальних місць (а не водія), щоб усунути більшу частину місця, витраченого даремно в багатоповерховому гаражі [2].

Такі системи паркування працюють з комбінацією транспортних пристроїв, програмної технології та мобільного додатка.

APS також широко відомі під низкою інших назв, зокрема: автоматизована парковка (APF), автоматизована система зберігання та пошуку транспортних

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 11 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

засобів (AVSRS), система паркування автомобіля, механічна парковка та роботизована парковка.

Усі APS використовують загальну концепцію зменшення площі паркувальних місць – виведення водія та пасажирів із автомобіля перед його паркуванням. З повністю автоматизованим або напівавтоматизованим APS автомобіль під'їжджає до точки входу в APS, а водій і пасажир виходять з автомобіля. Потім автомобіль автоматично або напівавтоматично переміщується (з необхідністю певних супутніх дій) на місце для паркування [2].

Економія простору, яку забезпечує APS, у порівнянні з багатоповерховим гаражем, обумовлена в першу чергу значним зменшенням простору, не пов'язаним безпосередньо з паркуванням автомобіля: ширина та глибина паркувального місця (і відстань між паркувальними місцями) різко скорочуються, оскільки не потрібно робити жодного дозволу на заїзд автомобіля на місце для паркування або на відкриття дверей автомобіля (для водіїв та пасажирів); не потрібні смуги або пандуси, щоб під'їхати автомобілем до/від в'їзду/виїзду на місце для паркування; висота стелі зведена до мінімуму, оскільки в зоні паркування немає пішоходів (водіїв і пасажирів), а також для розміщення пішоходів на парковці не потрібні пішохідні доріжки, сходи чи ліфти [2].

При цьому постійно виникали проблеми із роботизованими системами паркування. На додаток до технічних проблем, хоча системи добре працюють у ситуаціях із відносно збалансованою пропускною здатністю, таких як торгові центри та залізничні вокзали, вони не підходять для високого пікового обсягу, наприклад, дуже високе використання годин пік або застосування, наприклад, стадіони. Крім того, паркувальники, не знайомі з системою, можуть викликати проблеми, наприклад, якщо не натиснути кнопку, щоб попередити повністю автоматизовану систему про наявність автомобіля, який потрібно припаркувати [2].

Існує багато різних видів КПС, це залежить від вбудованих в систему модулів. Найпростіші паркувальні системи (ПС) містять в собі лише два ультразвукових датчики, що вимірюють відстань. В загальному датчики встановлюються на задній бампер автомобіля. КПС вмикається, коли водій

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 12 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

вмикає передачу для заднього ходу. З часом ПС почали робити надійнішими та точнішими. В системах уже вбудовували по 4-8 датчиків для вимірювання відстані. Найчастіше ультразвукові датчики розміщуються в межах відстані тридцяти чи сорока сантиметрів, в залежності від розмірів автомобіля. Датчики розставляють на такій малій відстані, щоб уникнути так званих «сліпих зон». Датчики можуть бути встановлені не лише ззаду чи спереду, а і по боках автомобіля.

Зазвичай модулі індикації та модуль управління з'єднуються за допомогою дроту, який проходить вздовж корпусу автомобіля. З винайденням нових технологій удосконалюються і КПС. У нових моделях вбудовані модулі, що передають дані по бездротовому зв'язку.

Бувають випадки, коли КПС виявляє перешкоди навіть тоді, коли не потрібно. Наприклад:

- коли на дорозі автомобіль пересікається з автомобілем, що їде в зворотньому напрямку;
- якщо не рівна поверхня дороги;
- якщо датчик забруднений.

У випадку, якщо КПС не посилає сигнали – слід перевірити підключення усіх контактів.

Загальний вигляд паркувальних систем зображено на рисунку 1.1 [3].



Рисунок 1.1 – Загальний вигляд паркувальних систем

Технічні характеристики:

Напруга - 12В;

| | | | | | | |
|------|------|---|--------|------|-------------------------|--------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. 6 |
| | | | | | | 13 |
| Змн. | Арк. | № | Підпис | Дата | | |

Працює в межах від -30°C до $+80^{\circ}\text{C}$;

Містить: Блок керування, LED екран, датчики для паркування, кабель живлення, коробку [3].

Вартість таких систем близько 1000грн.

В роботі розробляється КПС, яка міститиме: блок керування, датчик для вимірювання відстані, LED екран, кабель живлення, 9 світлодіодів. Технічні вимоги вказані в Додатку А.

Після аналізу поставлених завдань розроблено алгоритм роботи комп'ютеризованої паркувальної системи для транспортних засобів.

На рис. 1.2 показано покроковий алгоритм роботи КПС.

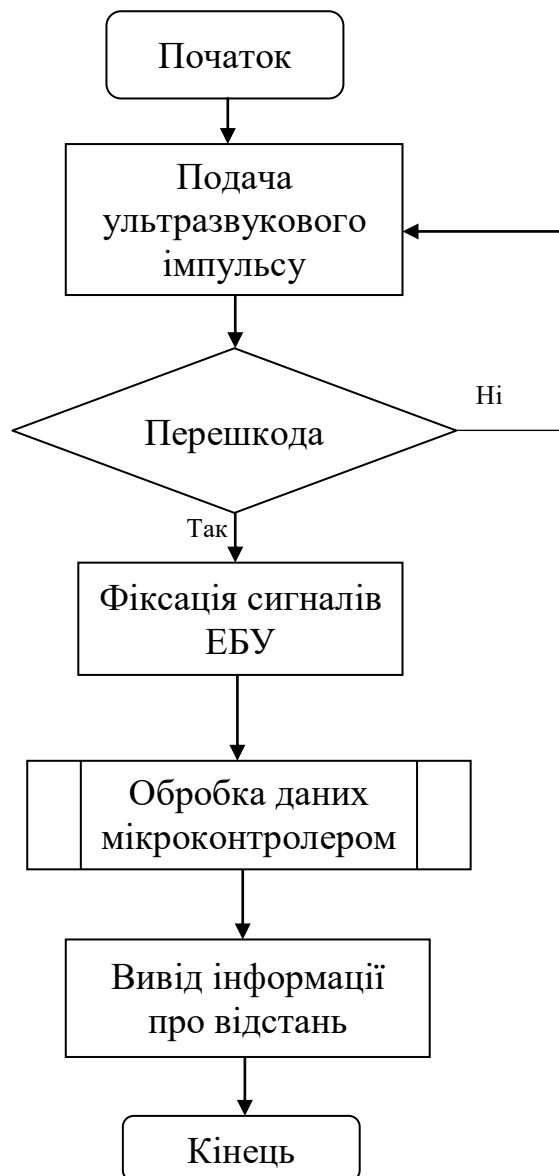


Рисунок 1.2 – Алгоритм роботи комп'ютеризованої паркувальної системи

Комп'ютеризована паркувальна система, відповідно до розробленого алгоритму працює таким чином: після подачі живлення до системи, вбудований ультразвуковий датчик починає випромінювати ультразвукові імпульси, які відбиваються від об'єкта (перешкоди) і ультразвуковий імпульс повертається назад до датчика відстані. Ультразвуковий датчик отримує відбиті імпульси та передає його до електронного блоку управління (ЕБУ), який фіксує інформацію, тоді вбудований мікроконтролер розприділяє її на датчики вихідної інформації (світлодіоди, датчики звуку, монітор).

Ультразвуковий датчик не ловить ультразвукових імпульсів, якщо відсутні перешкоди.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 15 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА

2.1 Розробка узагальненої структури комп'ютеризованої паркувальної системи

Для проектування КПС відповідно до поставлених завдань було обрано необхідне апаратне та програмне забезпечення. Перед початком створення системи була створена узагальнена структурна схема комп'ютеризованої паркувальної системи для транспортних засобів (рис. 2.1.)

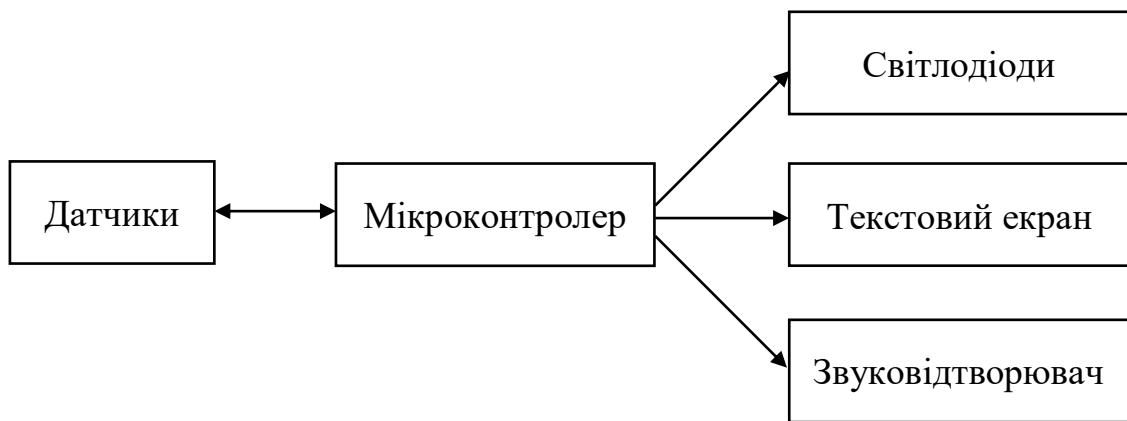


Рисунок 2.1 – Узагальнена схема підключення апаратних компонентів КПС

В основі системи використовується мікроконтролер, який обмінюється даними з датчиком (ультразвуковий датчик відстані HC-SR04), та відправляє інформацію на компоненти що видають сигнали (світлодіоди, текстовий екран, відтворювач звуку НРА17А).

Спираючись на узагальнену схему (рис. 2.1) та вимоги технічного завдання (Додаток А) було обрано ряд модулів для створення комп'ютеризованої паркувальної системи:

- плата ArduinoUNO – платформа для розробки на базі мікроконтролера ATmega 328P;

| | | | | | | | | | | | | |
|-----------|-----------------|--|--|--|------------------------------|--|--|---------------------------|----|---------|--|--|
| | | | | | КС КРБ 123.237.00.00 ПЗ | | | | | | | |
| Змн. | | | | | | | | | | | | |
| Розроб. | Ярмусь О.С. | | | | РОЗДІЛ 2 ПРОЄКТНА ЧАСТИНА | | | Літ. | | Акрушіє | | |
| Перевір. | Осухівська Г.М. | | | | | | | | 16 | 22 | | |
| Реценз. | Млинко Б.Б. | | | | | | | ТНТУ, каф. КС, гр. СІс-43 | | | | |
| Н. контр. | Луцик Н.С. | | | | | | | | | | | |
| Затверд. | Осухівська Г.М. | | | | | | | | | | | |

- текстовий екран 16x2. LCD дисплей МТ–16S2Н для виведення тексту з підсвічуванням. Дисплей виконаний на чипі, що відповідає стандартам HD44780, який є де-факто стандартом для LCD-екранів [5];
- ультразвуковий датчик відстані HC-SR04 – точний датчик для вимірювання рухів, може визначати відстань до півтора метра з точністю до 3мм;
- відтворювач звуку НРА17А, що не має вбудованого генератора частоти.

Оскільки комп'ютеризована паркувальна система розробляється на платі Arduino UNO то для її програмування було обрано середовище програмування Arduino IDE.

2.2 Обґрунтування вибору апаратного забезпечення для проектування комп'ютеризованої паркувальної системи

Комп'ютеризована паркувальна система для транспортних засобів створена на платі Arduino UNO – платформа для розробки на базі мікроконтролера ATmega328P.

На платі Arduino Uno передбачено все необхідне для зручної роботи з мікроконтролером, а саме: роз'єм USB, 14 цифрових входів/виходів, 6 аналогових входів, роз'єм живлення, роз'єм для програмування компонентів схеми (ICSP), кнопка скидання, кварцовий резонатор на 16МГц.

Самим головним елементом платформи Arduino Uno є 8-бітний мікроконтролер AVR – ATmega328P – мозок Arduino, який програмується через USB-роз'єм. Він містить три типи пам'яті. Він має 32КБ енергонезалежної флеш-пам'яті. Він використовується для зберігання програм і зберігається на платі, навіть після того, як його вилучили з джерела живлення. 2КБ енергозалежної пам'яті SRAM, яка використовується для зберігання змінних, які використовуються програмами під час роботи. 1КБ енергонезалежної пам'яті EEPROM. Це використовується для зберігання даних, які залишаються доступними навіть після вимкнення та повторного ввімкнення плати. [4]

Контакти плати Arduino UNO:

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 17 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

- VIN – напруга зовнішнього джерела живлення;
- виводи живлення (3,3V, 5V, GND) – використовуються ці контакти для підключення до схем на 3,3V, 5V або GND;
- IOREF – з порта платам розширення надається інформація про робочу напругу мікроконтролера;
- цифрові контакти вводу/виводу (2-13) – підтримують вхід або вихід від 0 до 5V;
- зовнішні переривання – це контакти два і три, можуть бути налаштовані так, щоб визначити переривання з низькими або змінюючими значеннями [1];
- ШІМ-контакти – будь-які контакти з ними можна використовувати для генерування імпульсно-модульованих прямокутних хвиль;
- контакт 13 – управляє вбудованим світлодіодом, який використовується Arduino для отримання живлення і корисний для налагодження;
- аналогові вхідні контакти – контакти від A0 до A5 забезпечують роздільну здатність 10 біт. Використовується для вимірювання безперервної напруги від 0V до 5V;
- аналоговий опорний контакт (AREF) – вхідний контакт використовується додатково, якщо потрібно, щоб зовнішні опорні напруги використовувалися для АЦП, а не внутрішніх Vref;
- Reset Pin – потрібно перевести цю лінію на низький рівень, щоб скинути мікроконтролер;
- логічний рівень для: одиниці – 5V, нуля – 0V;
- максимальний струм виходу – 40mA;
- TWI/I²C: порти SDA і SCL взаємодіють з периферією синхронного протоколу, через два дроти. Бібліотека Wire використовується для їх роботи [4];
- SPI – за допомогою цих пінів здійснюється зв'язок по інтерфейсу SPI. Також потрібно використовувати SPI бібліотеку;
- UART (асинхронний приймач/передавач) – це послідовний інтерфейс. ATmega328 має лише один модуль UART. Виводи (RX, TX) UART підключені до

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 18 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

схеми конвертера USB-в-UART, а також підключені до контактів нуль (0) та один (1) в цифровому заголовку [4];

– роз'єм USB – використовується задля прошивки Arduino UNO через комп'ютер. Роз'єм для підключення зовнішнього живлення становить від 7В до 12В [4];

– зовнішнє джерело живлення – дозволяє Arduino працювати, коли він не підключений до USB-порту для живлення. Він приймає напругу 7-12В [1];

– USB-штекер – це живлення Arduino без необхідності використання зовнішнього джерела живлення, і це те, що ви використовуєте для завантаження ескізів (програм) на мікроконтролер, а також для зв'язку з іншим ескізом Arduino (через послідовний порт, `println()` тощо).

На рис. 2.2. зображена структурна електрична схема плати Arduino UNO.

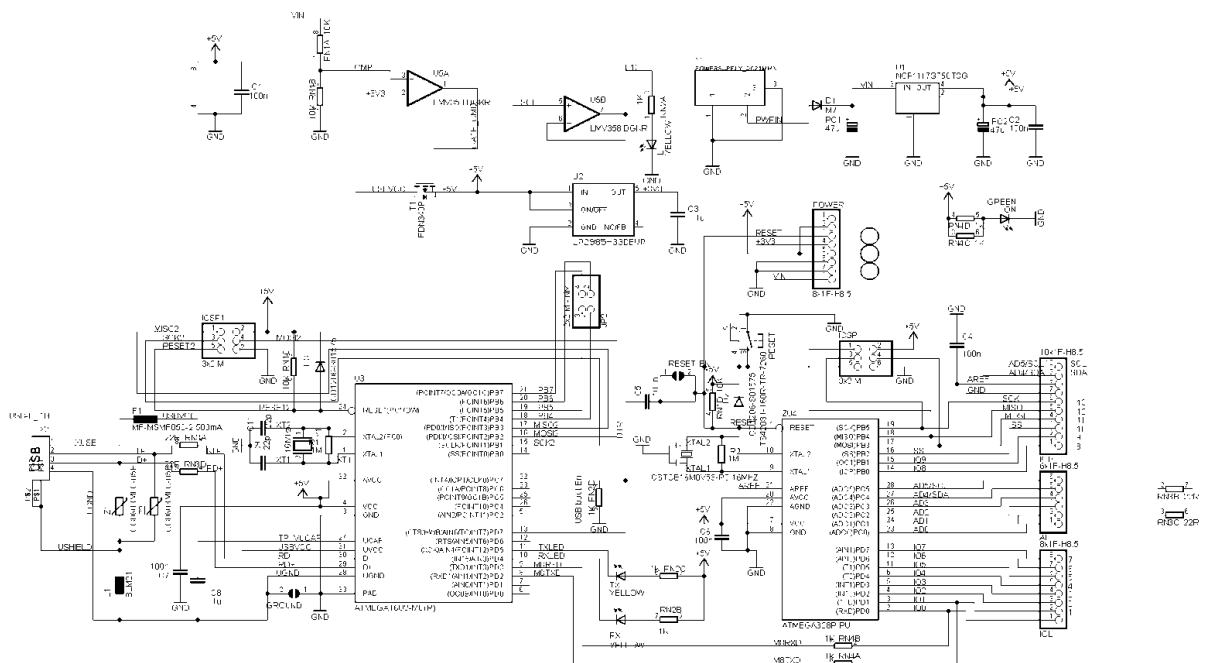


Рисунок 2.2 –Принципова електрична схема плати Arduino UNO

В таблиці 2.1 описані характеристики Arduino UNO

Таблиця 2.1 – Характеристики плати Arduino UNO

| | |
|--|---------------------------|
| Мікроконтролер | АТmega328 |
| Тактова частота | 16 МГц |
| Напруга логічних рівнів | 5 В |
| Напруга живлення на виходах | 7–12 В |
| Портів загального призначення введення/виведення | 20 |
| Максимальний струм з піна введення/виведення | 40 мА |
| Максимальний вихідний струм піна | 3.3V: 50 мА 5V: 800 мА |
| Портів з підтримкою ШІМ | 6 |
| Портів, підключених до АЦП | 6 |
| Розрядність АЦП | 10 біт |
| Flash-пам'ять | 32 КБ |
| EEPROM-пам'ять | 1 КБ |
| Оперативна пам'ять | 2 КБ |
| Габарити | 69 × 53 мм |

Платформа Arduino складається з двох частин – програмної та апаратної. Для програмування мікроконтролера використовується мова програмування C++, відома так само як Wiring. Розроблювати програми для мікроконтролера можна в середовищі Arduino IDE.

Для зручного перегляду відстані було обрано текстовий екран 16x2. LCD дисплей МТ–16S2Н для виведення тексту з підсвічуванням. На екрані одночасно можуть поміститися 32 символи (по 16 символів у 2х рядах) [5].

Основна особливість дисплея – незвичайна нумерація виходів. Усього їх 16, але йдуть вони не по порядку, а дуже нетривіальним способом. І лише три з них підписані: 14, 1 та 16. Але, виявляється, це цілком достатньо.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 20 |

Далі необхідно підключити ланцюг, який відповідає за відображення символів. Для цього призначені контакти 1, 2 та 3 на дисплеї. Перед підключенням необхідно вимкнути Arduino від живлення.

Перший це земля.

Наступний контакт – живлення +5В.

Третій – контрастність.

В таблиці 2.2 наведено характеристики LCD дисплея МТ–16S2Н

Таблиця 2.2 – Характеристики LCD дисплея МТ–16S2Н

| | |
|-------------------------------|---|
| Напруга | 3,3–5 В |
| Максимальний струм споживання | 1мА |
| Струм підсвічування | 100 мА |
| Індикація | 2 рядки по 16 символів. Символи відображаються в матриці 5 × 8 точок; |
| Габарити | 84 × 44 × 13 мм |

Індикатор дозволяє:

- працювати, як по восьми, так і по чотирьохбітній шині даних;
- приймати команди, які передаються по шинах даних;
- зберігати з шин даних дані в ОЗУ і навпаки;
- запам'ятовувати до восьми зображень або символів, що задаються користувачем;
- управляти контрастністю і підсвічуванням [5].

Наступним, невід'ємним елементом КПС є ультразвуковий датчик відстані HC-SR04 – датчик, що дозволяє вимірювати відстані від двох сантиметрів до півтора метра, з точністю до трьох міліметрів. Цей модуль містить в собі вузол контролю, ультразвуковий передавач та приймач. Датчик практично не реагує на сонячне проміння та електромагнітні шуми. Спереду датчика HC-SR04 розташовані два ультразвукові датчики: Transmitter (ТСТ40-16Т) – передає

ультразвукові хвилі; Receive (TCT40-16R) – приймає уже відбиті ультразвукові хвилі. В центрі модуля розташовано вивідний кварцовий генератор на 27МГц [6].

Принцип дії:

1) Для початку посилаємо високий рівень на вихід trig (тригер) протягом 10мкс;

2) Модуль посилає ультразвукові імпульси частотою 40 кГц і якщо є перешкоди то приймає імпульси назад;

3) Якщо повернувся сигнал, то на модулі, на виході echo автоматично встановлюється низький рівень на 150мс.

Також за допомогою формули (2.1) можна розрахувати відстань до перешкоди за часом, який минув, S:

$$S = \frac{tc}{2} \quad (2.1)$$

де t – час імпульсу, який виміряли; c- sound velocity, швидкість звуку (340м/с) [7].

Перевагами датчика є те, що він не реагує на світло сонця чи колір об'єкта, так, як це роблять інфрачервоні датчики вимірювання відстані.

Недоліками є те, що датчик HC-SR04 не зреагує, як на перешкоду, на пухнасті або тонкі предмети.

На рис. 2.5 зображено плату ультразвукового датчика вимірювання для вимірювання відстані.

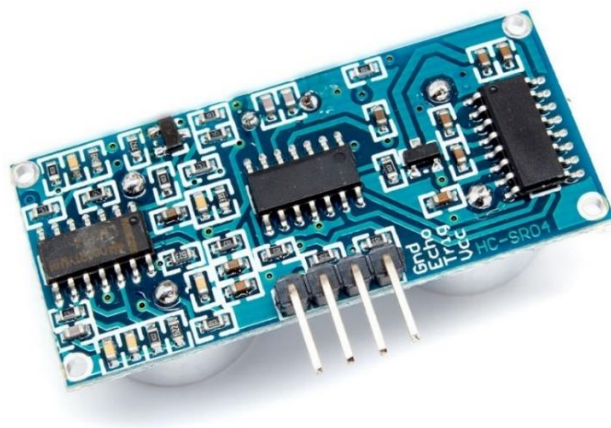


Рисунок 2.3 – Плата ультразвукового далекоміра

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 22 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Ще одним недоліком є те, що датчик HC-SR04 не містить в собі великого набору інтерфейсів. Натомість можна скористуватися вбудованими бібліотеками:

- Ultrasonic – найпопулярніша бібліотека для HC-SR04;
- NewPing – відрізняється більшою точністю і швидкістю роботи;

Ультразвуковий датчик відстані HC-SR04 містить чотири піни:

- Vcc – контакт для живлення;
- Echo – цифровий вихід;
- Trig – цифровий вхід. Для того, щоб запустити вимірювання, потрібно подати на нього логічну одиницю на 10мкс. Рекомендовано виконувати вимірювання не частіше, а ніж 50мс;
- GND – (земля) негативний контакт живлення [6].

На рис. 2.6 зображена принципова електрична схема ультразвукового датчика відстані HC-SR04. На схемі зображено три контролери: STC11, MAX232, TL074. Схему такого пристрою можна розділити на 2 блоки: Transmit – блок, що відповідає за передачу сигналу; Reseive – блок, що отримує звукові імпульси.

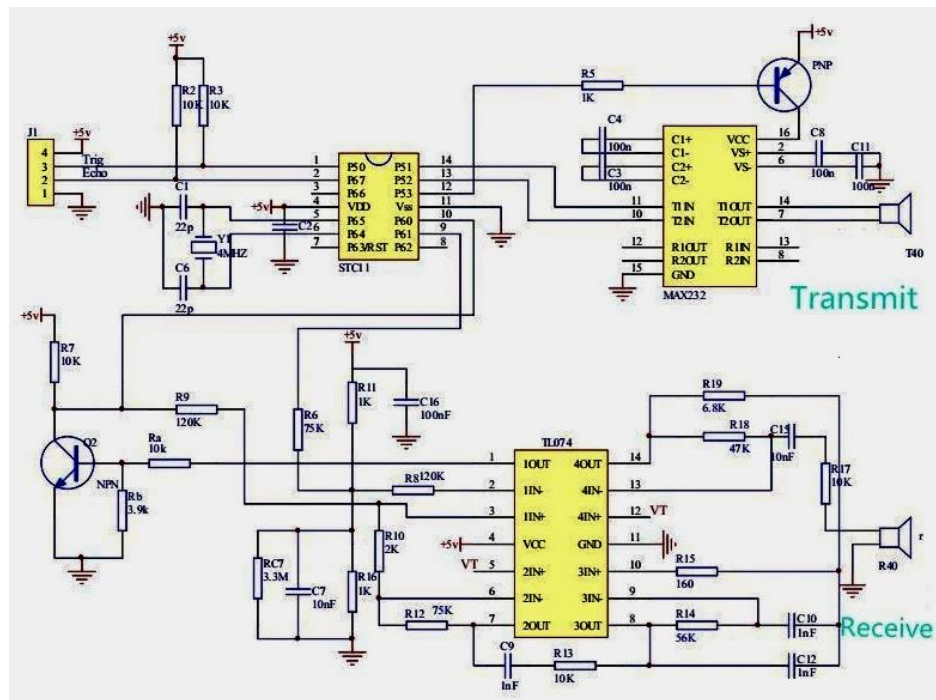


Рисунок 2.4 – Принципова електрична схема ультразвукового датчика відстані HC-SR04

В таблиці 2.3 наведені характеристики ультразвукового датчика відстані HC-SR04

Таблиця 2.3 – Характеристики ультразвукового датчика відстані HC-SR04

| | |
|-----------------------|-------------|
| Робоча напруга | 3.8 - 5.5В |
| Тип | HC-SR04 |
| Струм | 8 мА |
| Частота | 40 кГц |
| Максимальна дистанція | 1500 мм |
| Мінімальна дистанція | 0 см |
| Роздільна здатність | 3 мм |
| Ширина імпульсів | 10 мкс |
| Кут | 15 градусів |
| Зовнішні габарити | 37x20x15 мм |

Задля підключення модуля до Arduino, можна використати існуючу функцію `tone`, або ж бібліотеку `Tone`, яка має багато можливостей.

Система розміщена на макетній платі (так звана breadboard або) – чудова річ для експериментальних дослідів з електронікою. Нею можна користуватися не використовуючи паяльника. Можна швидко зібрати електричні схеми.

З боків макетної плати розташовані пази, за допомогою яких можна підключити ще плати. Дошку можна закріпити до пристрою, відклеївши захисну плівку.

Характеристики макетної плати наведено в таблиці 2.4

Таблиця 2.4 – Характеристики макетної плати

| | |
|------------------------------|--------------|
| Загальна кількість контактів | 830 точок |
| Контактів живлення | 200 точок |
| Контактів для монтажу | 630 точок |
| Діаметр контакту | 0,8 мм |
| Крок пікселів | 2,54 мм |
| Габарити | 165×55×10 мм |

Наступним елементом КС є змінний резистор (потенціометр) на основі вуглецевого провідника, який змінює опір від нуля до номінального опору в 10кОм.

Характеристики потенціометра:

- Потужність: 0.25Вт;
- Точність: $\pm 20\%$;
- Максимальна робоча напруга: 250В;
- Характеристика зміни опору: лінійна [8].

2.3 Обґрунтування вибору програмного забезпечення для проектування комп'ютеризованої паркувальної системи

Для написання коду для мікроконтролера було обрано програмне середовище Arduino IDE.

Середовище для програмної розробки Arduino включає в себе: вбудований текстовий редактор для написання програмного коду, область повідомлень, консоль панель інструментів та ще кілька пунктів меню.

Для того, щоб завантажити код на мікроконтролер Arduino використовується власна програмна оболонка IDE. Оболонка розроблена за допомогою мови програмування Java на базі проекту Processing. Вона працює на ОС Windows, MacOSX і Linux. Має власні бібліотеки Arduino. Arduino IDE

містить текстовий редактор, менеджер проєктів, препроцесор, компілятор і інструменти для завантаження програми в мікроконтролер [10].

Програми Arduino IDE пишуться на мові програмування C++ (використовується компілятор AVR-GCC), написані програми називаються Sketch. Також є можливість підключати стандартні файли C++.

При написанні коду в редакторі потрібно прописати декілька обов'язкових функцій.

Функція `main ()` уже є створена процесором Arduino.

Функція `setup ()` потрібна для стартування програми.

Функція `loop ()` використовується для нескінченного циклу.

Відносно проєкту бібліотеки підбираються самі, програмісту не потрібно їх прописувати та бібліотеки призначені для користувача – потрібно вказувати.

Менеджер проєкту Arduino IDE нестандартно додає бібліотеки до проєкту. Бібліотеки додаються в окрему папку в проєктному каталозі IDE у текстовому вигляді на стандартній C++ мові. Перед початком створення проєкту програміст вибирає потрібні йому бібліотеки і вони автоматично додаються до списку компіляції [10].


Для написання найпростішої програми на Arduino потрібно лише дві функції `setup ()` та функція `loop ()`, яка викликається після функції `setup ()` і працює в нескінченному циклі постійно, поки працює мікроконтролер [10]


Записування програми на мікроконтролер Arduino проходить через завантажувач – спеціальний запрограмований пристрій (продається разом з платою Arduino). Завантажувач розроблений на базі Atmel AVR Application Note AN109, він може працювати з використанням інтерфейсів RS-232, USB або Ethernet, в залежності від вбудованого інтерфейсу на платі Arduino. В деяких випадках, для того, щоб запрограмувати мікроконтролер, необхідно придбати окремий перехідник [10].


Користувач самостійно може запрограмувати мікроконтролер. Для цього IDE має інтегровану підтримку програматора на базі AVRDUDE. Може підтримувати кілька типів програматорів.


| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 26 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |


У платформі IDE є панель інструментів, на якій розташований ряд кнопок, якими дуже зручно і швидко користуватися:


 Verify/Compile – перевіряє код на помилки та компілює.


 Stop – зупиняє процес роботи запущеної програми (Serialmonitor).

 New – дозволяє створити новий файл для написання коду (скетчу).

 Open – показує меню для доступу до скетчів в блокноті.

 Save – служить для збереження скетчу.

 Upload to I/O Board – компіляція коду дозволяє завантажити його в Arduino.

 SerialMonitor – відкриття моніторингу послідовної шини (Serial monitor).

Більше команд можна знайти в меню: File, Edit, Help, Sketch, Tools.

На рис. 2.5 показано вкладку Edit. За допомогою якої можна: копіювати, вставляти, вирізати, змінювати розмір шрифту, знаходити потрібний елемент коду. Особливими функціями цієї вкладки є:

- Copy for Discourse – служить для того, щоб копіювати в буфер обміну виділений код програми.
- Copy as HTML – використовується для копіювання коду програми, як HTML код.

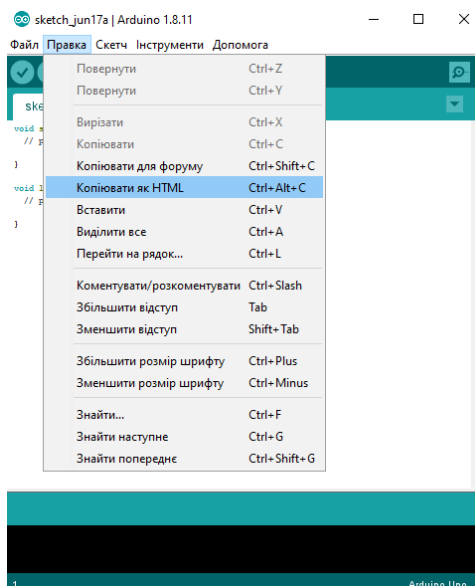


Рисунок 2.5 – Вкладка «Скетч»

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 27 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

На рис. 2.6 зображена вкладка Скетч, яка має такі характеристики:

- Verify / Compile – перевіряє чи присутні помилки в коді.
- ImportLibrary – служить для додавання бібліотек в потрібний скетч, при цьому прописує директиву #include в код.
- Show Sketch Folder – показує папку, в якій містить файли проекту.
- Add File – добавляє файл до проекту. Новий файл буде відображатися в іншій закладці вікна проекту.

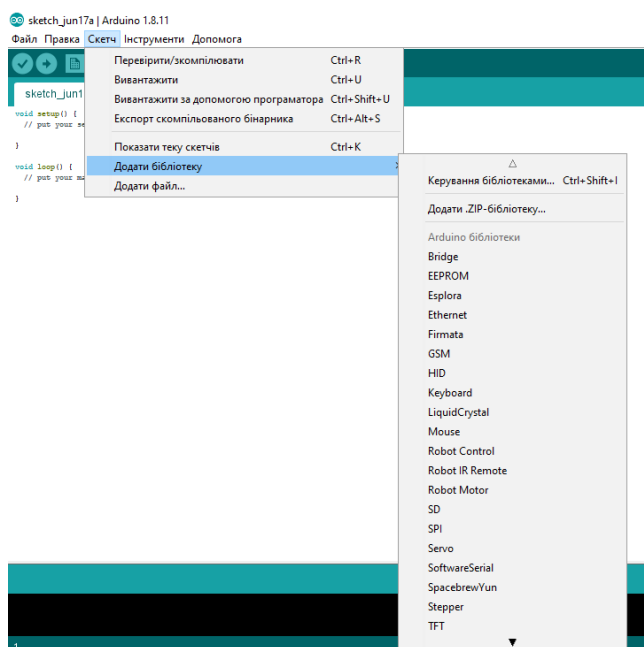


Рисунок 2.6 – Вкладка «Скетч»

Вкладка Tools показана на рис. 2.7 має наступні функції:

- Auto Format – ця функція допомагає редагувати код, наприклад, відкриває та закриває дужки поміщаючи в них твердження.
- Board – дозволяє вибрати платформу для користування.
- Serial Port – доступний список пристроїв для передачі даних на комп'ютері. При відкритті меню Tools список автоматично оновлюється.
- Burn Bootloader – за допомогою цього пункту меню можна записувати завантажувач в мікроконтролер.

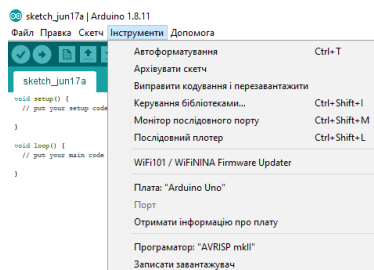


Рисунок 2.7 – Вкладка «Інструменти»

В програмному середовищі Arduino використовується принцип блокнота – файл в якому зберігаються програми (скетчі). Коди програм з блокнота можна відкрити за допомогою меню File> Sketchbook а також кнопкою Open, що знаходиться на панелі інструментів. Після того, як програма вперше запуситься, Arduino автоматично створює нову директорію для блокнота. Можна також змінити місцезнаходження блокнота за допомогою діалогового вікна Preferences. [9].

Особливістю Arduino IDE є те, що можна одночасно працювати з багатьма файлами, оскільки, кожен файл може бути відкритим в окремій вкладці. Файли можна зберігати без розширення або ж із стандартними розширеннями платформи Arduino IDE (* .c) – файли C, (* .cpp) – файли C++, чи головними файлами, які мають розширення (.h).

Якщо проект потрібно завантажити, спочатку необхідно вказати серійний порт, для цього переходимо в меню Tools> Board і Tools> Serial Port. Для кожної операційної системи порти можуть позначатися по різному. Для ОС Mac порти можуть позначатися: dev / tty.usbserial-1B1, для ОС Windows прикладом послідовного порту може бути COM 1 і вище, для ОС Linux – dev / ttyUSB0.

Коли платформа та порти були вибрані, на панелі інструментів потрібно натиснути кнопку завантаження. Більшість платформ Arduino автоматично перезавантажуються перед завантаженням. В іншому випадку потрібно натиснути

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 29 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

кнопку перезавантаження. Зазвичай під час перезавантаження мигають два світлодіода RX і TX. Після завантаження на екран буде виведене повідомлення про успішне завершення процедури або повідомлення про помилку, якщо завантаження буде не вдалим.

Для завантаження програми в мікроконтролер використовується Завантажувач Arduino (Bootloader). Його роботу можна побачити по миготінні світлодіода (13 пін).

В Arduino IDE бібліотеки на початку створення проєкту вибираються програмістом. З їх допомогою скетчі стають більш функціональними як в роботі з обробкою даних так і з апаратною частиною. Для підключення додаткових бібліотек необхідно в меню вибрати Sketch> Import Library. Разом з вибором бібліотек з'являться відповідна кількість директив #include, при потребі їх можна видалити.

Окрім стандартних бібліотек Arduino IDE дозволяє завантажити нові з різних ресурсів. Для того, щоб додати нові бібліотеки, потрібно спочатку створити нову директорію «libraries» в папці блокнота.

Платформа Arduino IDE підтримує апаратні засоби різних виробників. Для цього вони додаються в конкретний підкаталог каталогу блокнота. Нові платформи можуть використовувати власні бібліотеки, характеристики програматора, завантажувач. Важливо звернути увагу на розміщення платформ.

Використовувати папку "arduino", не рекомендується, оскільки можуть бути перезаписані вбудовані файли платформи Arduino.

Моніторинг послідовної шини (Serial Monitor): відображає дані, що передаються в платформу Arduino. Для цього, щоб дані були відправлені, потрібно набрати текст та натиснути Send або Enter. Після чого потрібно вибрати швидкість передачі даних зі списку. При підключенні моніторингу до послідовної шини платформа Arduino перезавантажеться. Також можливо обмінювати інформацію з платформою через програми Processing, Flash, MaxMSP і т.д.

Налаштування платформи стандартне при її установці. Для зміни налаштувань можна перейти у вікні Preferences. Всі інші налаштування знаходяться в файлі, його розташування показано у вікні Preferences.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 30 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Особливу увагу потрібно звернути на вибір платформи. При виборі слід звернути увагу на її характеристики: швидкість ЦП, швидкість передачі даних, налаштування завантаження скетчів до мікроконтролера, компіляція проєктів.

2.4 Розробка комп'ютеризованої паркувальної системи

Основним елементом схеми є плата Arduino UNO з мікроконтролером ATmega 328P (рис.2.8), до якого підключені всі інші модулі системи.

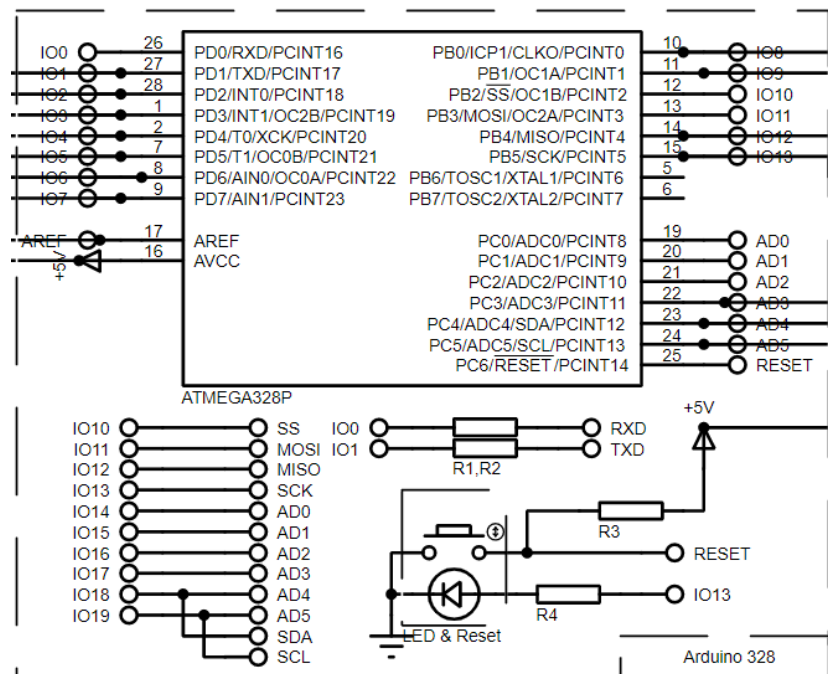


Рисунок 2.8 – Електрична принципова схема Arduino

На рис. 2.9 показано електричну принципову схему підключення ультразвукового датчика відстані HPA17A до мікроконтролера ATmega 328P. Контакт Echo з'єднаний з 13-м контактом до мікроконтролера, Trig з 12-м. Контакт GND(Ground) під'єднаний до землі, а контакт Vcc під'єднаний до контакту на мікроконтролері, що виводить 5В.

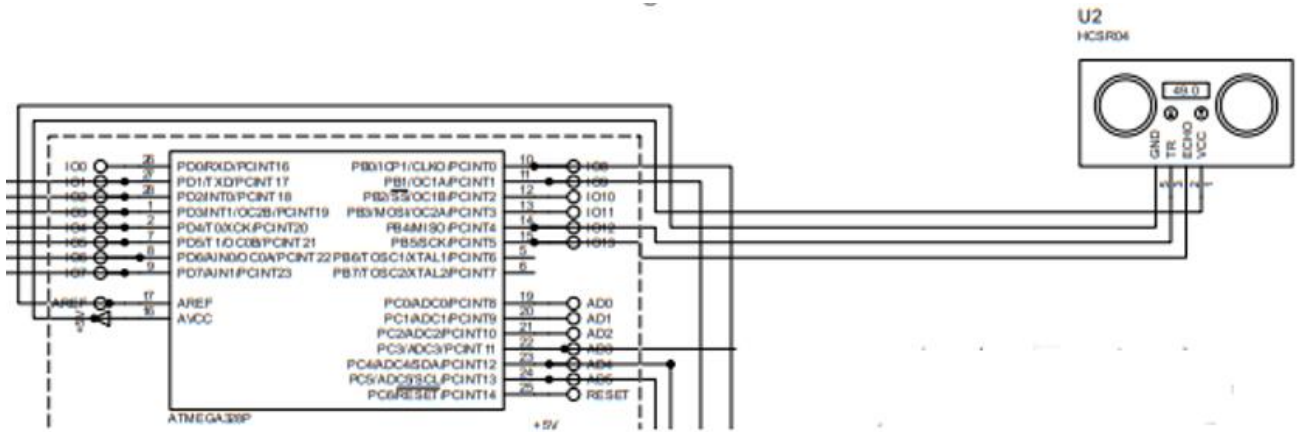


Рисунок 2.9 – Електрична принципова схема підключення HPA17A до мікроконтролера АТмега 328Р

На рис. 2.10 показано електричну принципову схему підключення LCD екрану та звукодавача до мікроконтролера АТмега 328Р. Звукодавач підключений контакт (+) до мікроконтролера на потрі AD3 та контакт (-) до землі. LCD екран зображений на рис. Має 4 контакти GNG – підключений до землі; VCC- підключений до 5В на платі Arduino; SCL та SDA підключені до мікроконтролера портів AD4 та AD5.

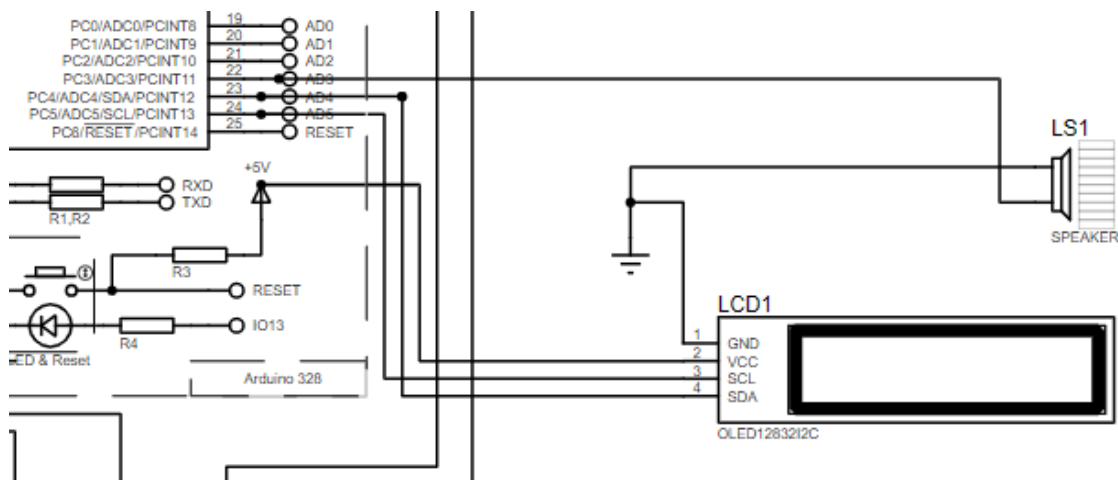


Рисунок 2.10 – Електрична принципова схема підключення LCD екрану та звукодавача до мікроконтролера АТмега 328Р

Наступна схема (рис. 2.11) – схема підключення світлодіодів до мікроконтролера. Світлодіоди підключені послідовно до потрів мікроконтролера

від 1 до 9. Між світлодіодом та контролером стоїть резистор 220Ом для стабільної роботи світлодіода. Інша ніжка кожного світлодіода під'єднана до землі.

Загальна принципова схема комп'ютерної паркувальної системи для транспортних засобів зображена на рис. 2.11.

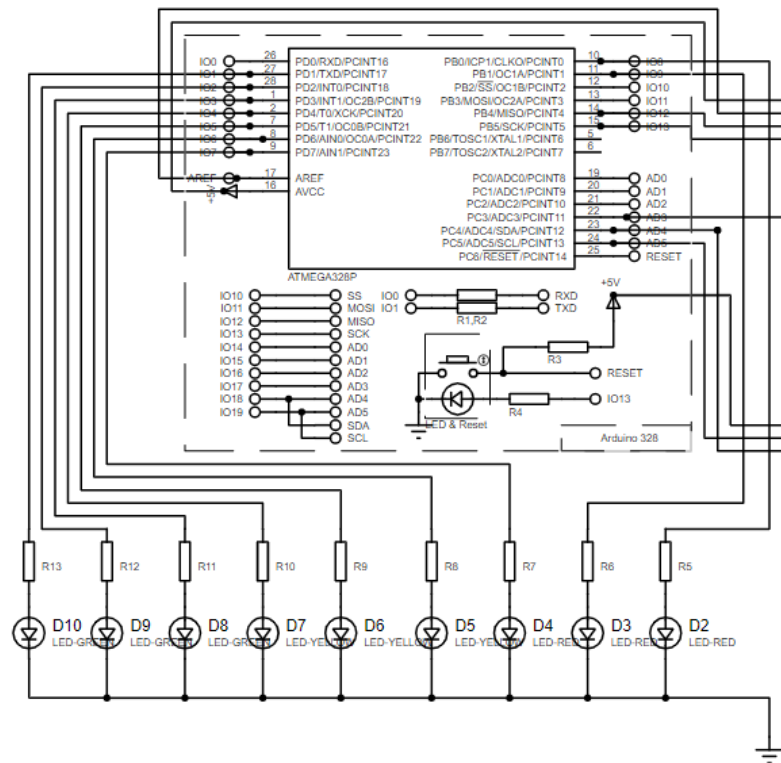


Рисунок 2.11 – Принципова електрична схема підключення світлодіодів до мікроконтролера АТмега 328Р

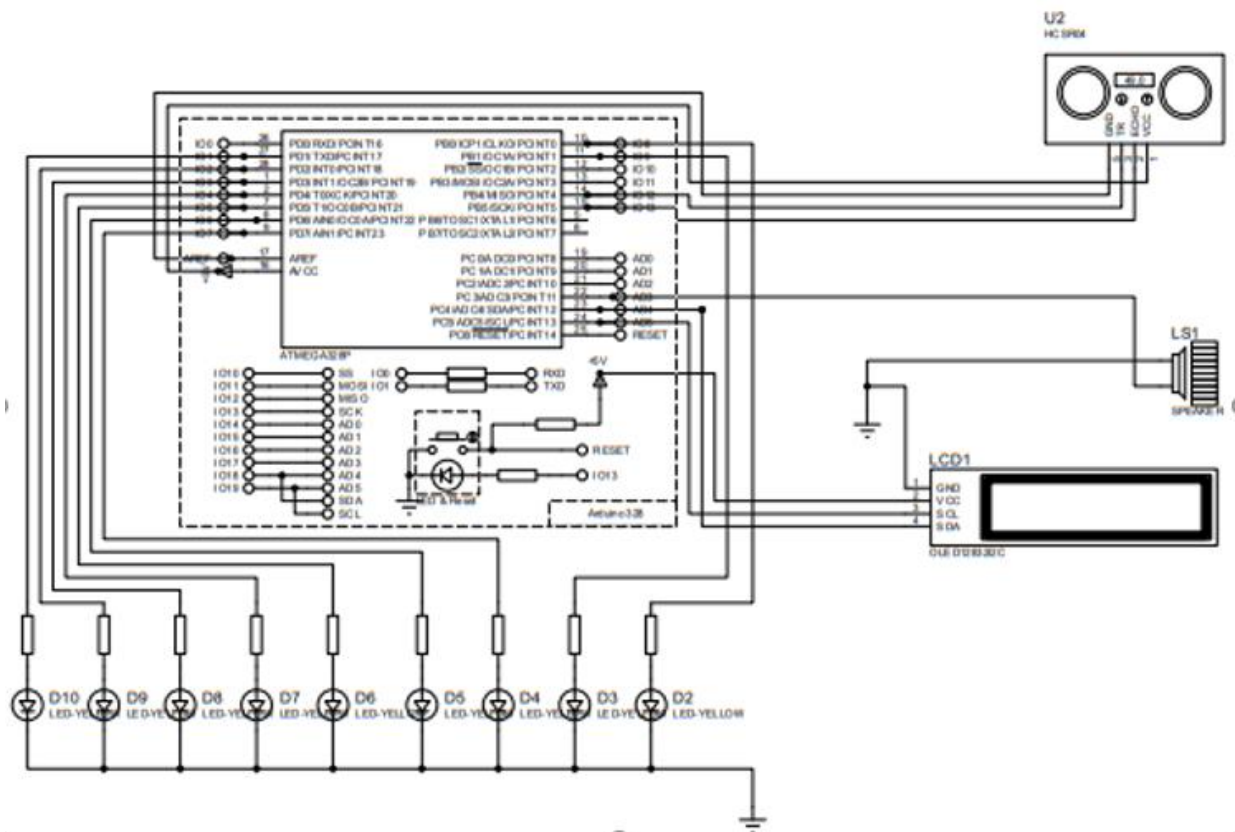


Рисунок 2.12 – Електрична принципова схема комп'ютеризованої паркувальної системи для транспортних засобів

Відповідно спроектованої принципової електричної схеми реалізуємо проєкт. Спроектована комп'ютеризована паркувальна системи в загальному міститиме 25 радіотехнічних компонентів не рахуючи з'єднувальних провідників:

- 10 світлодіодів;
- 10 резисторів номіналом 220 Ом;
- 1 потенціометр;
- 1 ультразвуковий далекомір;
- 1 дисплей;
- 1 плата;
- 1 динамік.

Першим кроком є розміщення світлодіодів на платі. Робимо загальний мінус для світлодіодів. Приєднуємо катод до шини яка відповідає за мінус на платі. До контактів анод на світлодіодах необхідно підключити резистори 220Ом, без підключення резисторів, світлодіод згорить.

На макетній платі необхідно під'єднати резистори послідовно з пінами Arduino від 2 до 10. Результат роботи показаний на рис. 2.13.

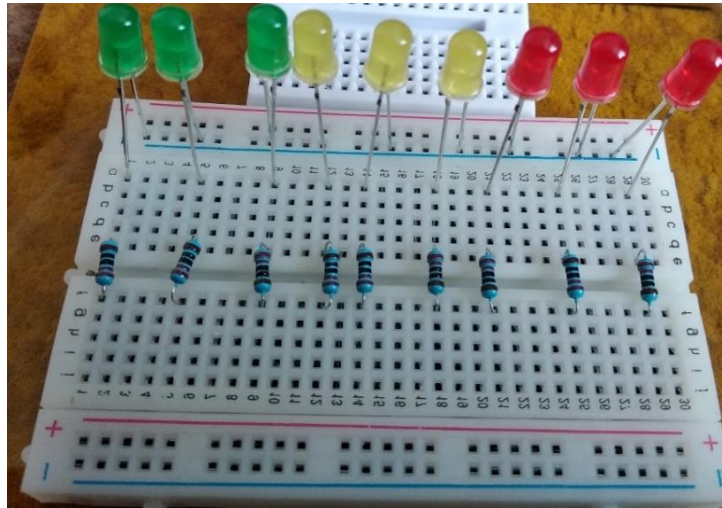


Рисунок 2.13 – Під'єднання світлодіодів

В наступному кроці (рис. 2.14) встановлюємо ультразвуковий датчик для вимірювання відстані, цей датчик буде розміщено на окремій макетній платі.

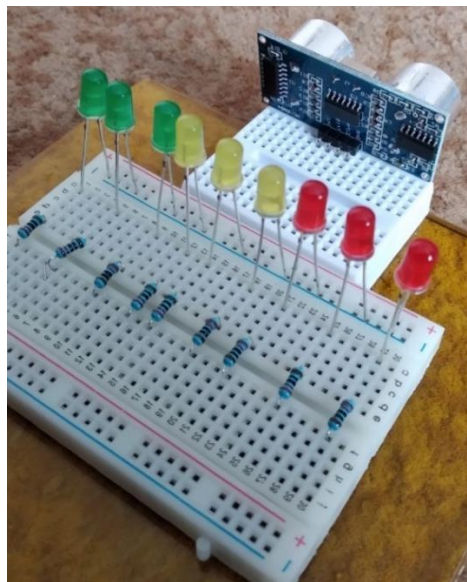


Рисунок 2.14 – Під'єднання ультразвукового датчика відстані

До портів датчика відстані КС підключаємо дроти (Тато–Тато). Як і на структурній схемі контакт Echo з'єднуємо з 13–м контактом на платі Arduino, а

Trig з 12-м., контакт GND(Ground) до контакту земля, а контакт Vcc під'єднуємо до контакту, що виводить 5V.

Відтворювач звуку HPA17A має два контакти – «+» та «-». Негативний контакт «-» потрібно з'єднати з мінусом від ультразвукового датчика для вимірювання відстані. Позитивний контакт «+» з'єднаний з одинадцятим піном на платі Arduino UNO.

Наступним кроком є під'єднання дисплею до комп'ютеризованої системи (рис. 2.15). На текстовий екран будуть виводитися повідомлення про відстань до об'єкту (перешкоди). Дисплей підключатиметься дротами з контактами Тато-Мама.

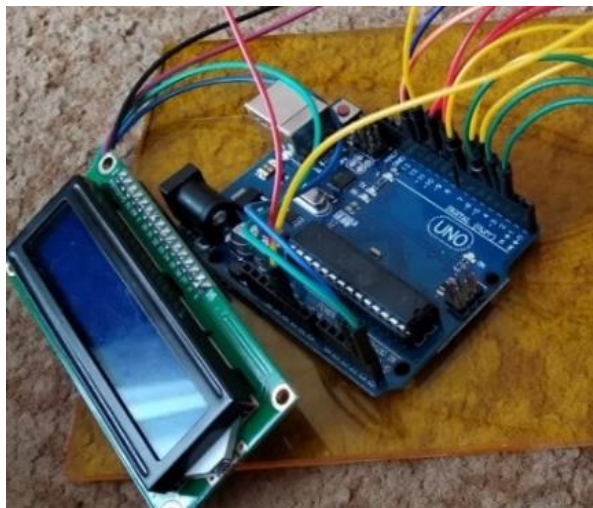


Рисунок 2.15 – Підключення дисплею

Після того як визначили до яких пінів буде під'єднаний текстовий редактор, переходимо до проектної реалізації – підключаємо модуль до плати Arduino UNO.

Розроблений макет комп'ютерної паркувальної системи для транспортних засобів розроблена на базі Arduino UNO зображена на рис. 2.16.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 36 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

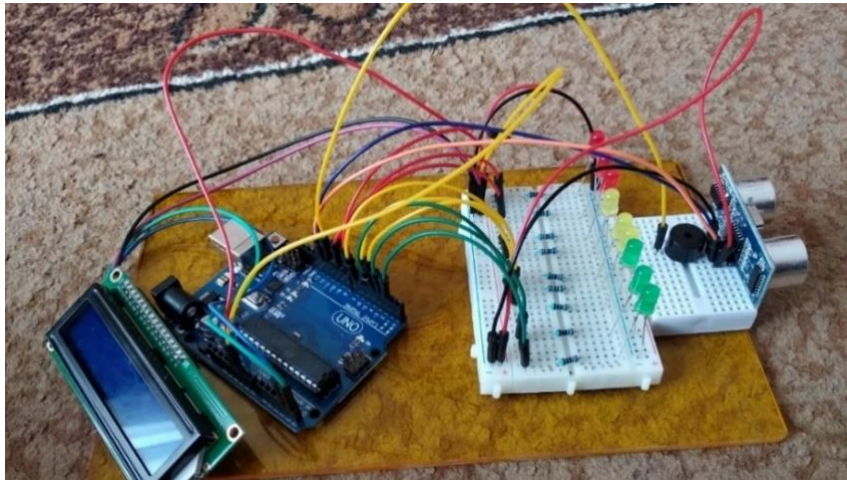


Рисунок 2.16 – Завершений вигляд комп'ютеризованої паркувальної системи для транспортних засобів

РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА

3.1 Реалізація проєктних рішень

Для функціонування сконструйованої комп'ютерної системи необхідно написати код. Код програми для мікроконтролера буде написаний у середовищі Arduino IDE. Написання коду в Arduino IDE базується на мові програмування C++.

Будь-яка програма на C++ повинна містити:

- Директиви для підключення бібліотек;
- функції програми;
- головна функція програми `main()`;
- опис змінних та функцій.

В цілому, програма містить декілька функцій. Між функціями можуть бути представлення оголошення, що відповідають за задання об'єктів та оператори процесорної обробки. Функції що використовує користувач, в більшості випадків оголошуються головній функції `main()`, функції завжди повинні бути описані перед початком їх використання [10].

Для створення програми, потрібно керуватися такими вимогами:

- усі функції повинні бути оголошеними;
- закінчення операторів здійснюється за допомогою символу «;»;
- фігурні дужки (« { » та « } »). Всі функції, що знаходяться в фігурних дужках, розуміється, як один оператор;
- повинен бути відступ між блоками (3-4 символи), при цьому, блоки що знаходяться на одному і тому ж рівні повинні бути вирівняні по вертикалі. У даному випадку написання програмного коду здійснювалося в середовищі програмування Arduino IDE [10].

| | | | | | | | |
|-----------|-----------------|--|--|--|-------------------------------|---------------------------|---------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | | |
| Змн. | | | | | | | |
| Розроб. | Ярмусь О.С. | | | | РОЗДІЛ 3 ПРАКТИЧНА ЧАСТИНА | Літ. | Акрушів |
| Перевір. | Осухівська Г.М. | | | | | 38 | 9 |
| Реценз. | Млинко Б.Б. | | | | | ТНТУ, каф. КС, гр. СІс-43 | |
| Н. контр. | Луцик Н.С. | | | | | | |
| Затверд. | Осухівська Г.М. | | | | | | |

Директива `#include` використовується, щоб включити зовнішні бібліотеки у програму. Це надає доступ програмісту до багатьох груп стандартних бібліотек C++ (груп готових функцій), а також бібліотек, що були написані виключно для Arduino [10].

Функція `setup` запускається, як тільки починає запускатися скетч. Цю функцію використовують для введення змінних, запуску необхідних бібліотек, визначення режимів роботи виводів, і т. д. `Setup` запускається тільки один раз після кожного початку подачі живлення або кожного перезапуску плати Arduino.

Коли функція `setup()` виконається, свою роботу розпочне функція `loop()`, яка створює цикл для можливості вносити якісь зміни чи реагувати на них [10].

Оператор `for` – лічильник, дозволяє виконувати операції блоку певну кількість раз. Зазвичай лічильники потрібні для збільшення та завершення роботи циклу. Оператор `for` актуальний в тому випадку, якщо необхідно повторити якусь операцію, також дуже зручно поєднувати його з масивами для зручної роботи з даними [10].

Лічильник збільшення зазвичай використовується для збільшення і завершення циклу. Оператор `for` корисний для будь-якої повторюваної операції, він часто застосовується в поєднанні з масивами для роботи з сукупністю даних/пінів [10].

Оператор `while` – цикл, який виконується безперервно, доки вираз у круглих дужках, не буде дорівнювати хибному значенні (`false`). Для того, щоб цикл не продовжувався безкінечно, мусять відбутися якісь зміни. Зміни можуть відбуватися в будь-яких умовах, як в програмному коді так і під час роботи (тестування) апаратної частини.

Перед початком написання коду необхідно створити алгоритм роботи програми за яким вона має працювати.

На рис. 3.1 зображено алгоритм роботи програми. Алгоритм складається з послідовних циклів. Спочатку підключаємо всі піни, до яких підключені елементи системи. Для всіх пінів ставимо стан очікування. Після чого чекаємо на подачу сигналу з пінів, по яких під'єднаний ультразвуковий датчик відстані. Він ловить отриманий імпульс відбитий від перешкоди та інформацію про відстань передає

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 39 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

мікроконтролеру. Якщо сигнал надійшов, умова виконалася то програма продовжується, якщо умова не виконалася – повертаємося в стан очікування.

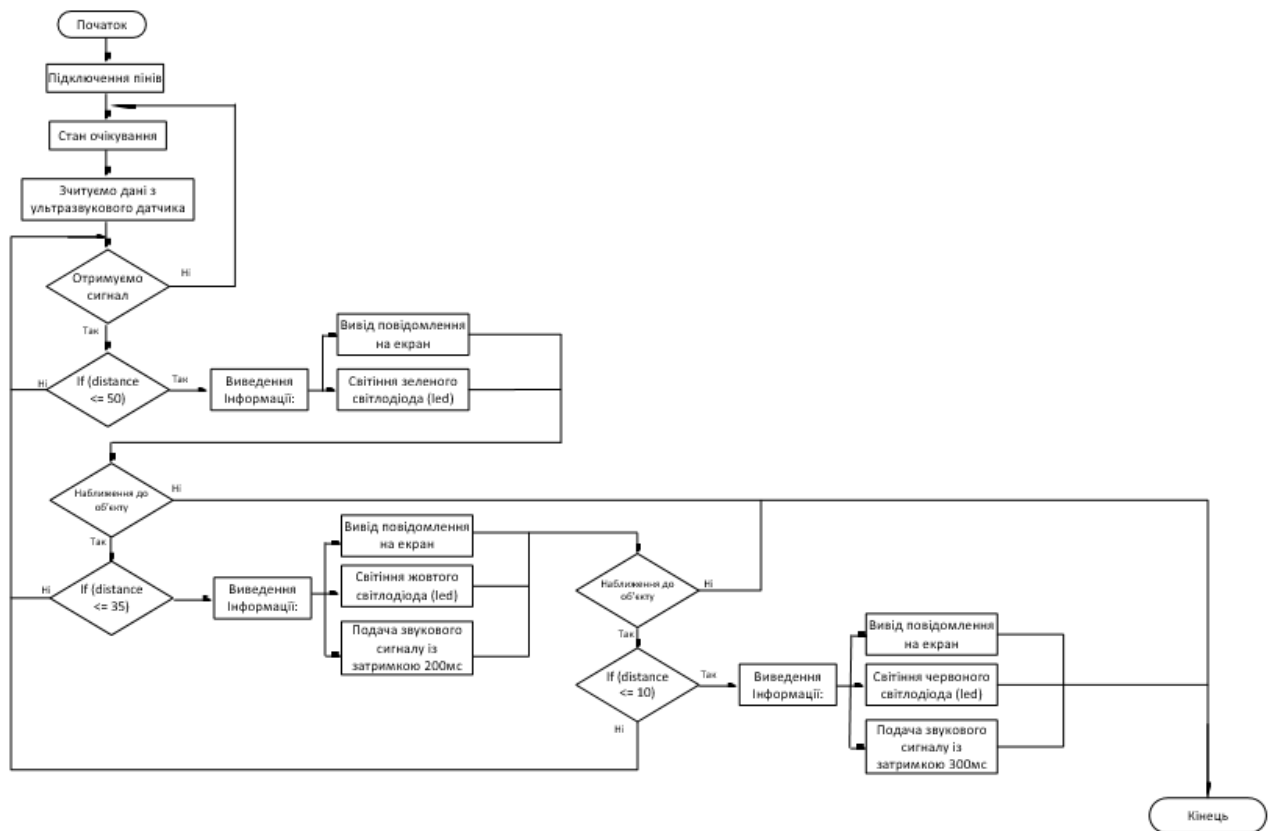


Рисунок 3.1. Алгоритм роботи програми

Відповідно алгоритму (рис. 3.1), мікроконтролер отримує інформацію про відстань та за конкретної умови, наприклад, якщо відстань становить 50см, умова виконана то сигнал повинен надійти до пристроїв що виводять інформацію – тоді програма закінчує свою роботу. Якщо умова що відповідає за відстань 50см не виконана тоді переходимо до наступної умови, в якій вказано дистанцію 35см, відповідно, якщо ця умова виконана, згідно програми мікроконтролер повинен передати інформацію на датчики що виводять інформацію і програма закінчена. Якщо ця умова не виконана переходимо до наступної де вказана відстань дорівнює 10см. Якщо ж жодна з умов не виконана, всі підключені піни лишаються в стані очікування.

Оскільки при паркуванні датчик відстані працює постійно, то і алгоритм роботи програми складений лінійно, так, що при наближенні до перешкоди

сигнали будуть змінюватися. При умові що дистанція 50см буде мигати зелений світлодіод та показувати повідомлення на екрані, при умові 35см буде мигати жовтий світлодіод, показувати повідомлення та звучатиме застережливий звук із затримкою 200мс. При умові 10см буде мигати червоний світлодіод, показувати повідомлення та звучатиме застережливий звук із затримкою 300мс.

На рис. 3.2 – 3.6. Зображені частини коду, який був написаний для спільного функціонування комп'ютерної системи. Кожен фрагмент коду описаний та проаналізований, містить структурне пояснення.

На рис. 3.2. показаний фрагмент коду, в якому прописані бібліотеки для подальшої роботи. Бібліотека Wire.h служить для підключення пристроїв через шину I2C. Бібліотека LiquidCrystal_i2c.h використовувалася для взаємодії з дисплеєм. А також підключені бібліотеки для роботи з пінами.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#define trigPin 12
#define echoPin 13
#define led 2
#define led2 3
#define led3 4
#define led4 5
#define led5 6
#define led6 7
#define led7 8
#define led8 9
#define led9 10
#define buzzer 11
```

Рисунок 3.2 – Лістинг програми, що відповідає за підключення бібліотек

На рис. 3.3. прописаний код для режиму роботи певного входу чи виходу для світлодіодів. За світлодіоди зеленого кольору відповідають команди pinMode – встановлюємо контакт і його тип, тобто led – контакт(пін), OUTPUT – тип. Так для зелених світлодіодів служать піни led, led2 та led3. В даному випадку вони вимкнені. За це відповідає тип OUTPUT. Для жовтих – led4, led5 та led6, для червоних – led7, led8, led9.

Пін buzzer служить для роботи звуковідтворювача.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 41 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Пін trigPin – це відправлення звукових хвиль.

Пін echoPin – ловить звукові хвилі, які уже відбилися від об'єкта.

Команда lcd.init () – служить для ініціалізації LCD екрану

```
void setup() {  
  pinMode(trigPin, OUTPUT);  
  pinMode(echoPin, INPUT);  
  pinMode(led, OUTPUT);  
  pinMode(led2, OUTPUT);  
  pinMode(led3, OUTPUT);  
  pinMode(led4, OUTPUT);  
  pinMode(led5, OUTPUT);  
  pinMode(led6, OUTPUT);  
  pinMode(led7, OUTPUT);  
  pinMode(led8, OUTPUT);  
  pinMode(led9, OUTPUT);  
  pinMode(buzzer, OUTPUT);  
}
```

Рисунок 3.3 – Лістинг програми для встановлення режимів роботи входів/виходів

На рис. 3.4. показані рядки коду, в яких зазначені функціональні параметри комп'ютеризованої паркувальної системи: відстань, мигання світлодіодів, пицання звуковідтворювача, вивід інформації на екран і в залежності взаємодія цих всіх модулів. Команда if(distance < 50) служить для визначення перешкоди, що знаходиться даліше півметра.

Рядки коду:

```
lcd.print("Distance: ")
```

```
lcd.print(distance);
```

```
lcd.print(" cm");
```

Дозволяють вивести вказані повідомлення на екран.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 42 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

void loop()
{
  lcd.begin(16, 2);
  long duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  lcd.print("Distance = 50 cm");
  lcd.setCursor(2,1);
  if (distance <= 50) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led, HIGH);
  }
  else {
    digitalWrite(led, LOW);
  }
}

```

Рисунок 3.4 – Лістинг програми, що відповідає за параметри відстані

В частині коду на рис. 3.5. описані параметри відстані між перешкодою та автомобілем. Спочатку при відстані до об'єкта 30-25 см. Буде чути сигнал із затримкою 200мс. При відстані 20 см звуковідтворювач буде видавати сигнали з затримкою 300мс.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 43 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

        if (distance < 30) {
            lcd.backlight();
            lcd.print("Distance ");
            lcd.print(distance);
            lcd.print(" cm");
            lcd.setCursor(2,2);
            digitalWrite(led5, HIGH);
        }
        else {
            digitalWrite(led5, LOW);
        }
        if (distance < 25) {
            lcd.backlight();
            lcd.print("Distance ");
            lcd.print(distance);
            lcd.print(" cm");
            lcd.setCursor(2,2);
            digitalWrite(led6, HIGH);
        }
        else {
            digitalWrite(led6, LOW);
        }
        if (distance < 20) {
            lcd.backlight();
            lcd.print("Distance ");
            lcd.print(distance);
            lcd.print(" cm");
            lcd.setCursor(2,2);
            digitalWrite(led7, HIGH);
            tone(11, 7500);
            delay(300);
            noTone(11);
        }
    }
}

```

Рисунок 3.6 – Лістинг програми для визначення відстані

В загальному, код програми є послідовним та зрозумілим, його можна розділити на два етапи: підключення бібліотек та пінів апаратної частини та керування апаратними пристроями. Повний лістинг програм знаходиться в Додатку В.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 44 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

3.2 Тестування

Розроблена комп'ютеризована паркувальна система працює на відстані 50см від автомобіля для перешкоди. Може працювати при кліматичних умовах -20° до $+60^{\circ}$. КПС не реагує на гострі чи пухнасті предмети.

Створена КС працює від живлення у 5 В, з допомогою платформи Arduino. Звукові сповіщення можна регулювати за допомогою резистора змінного струму, іншими словами потенціометра. Індикація про відстань додатково надається світлодіодами встановленими в монтажній платі.

Результат тестування комп'ютеризованої паркувальної системи показаний на рис. 3.7.



Рисунок 3.7 – Результат роботи пристрою

На екран вивело інформацію про діапазон відстані 50см. та ра конкретну відстань до об'єкту. При наближені чи віддалені об'єкта від КПС відстань на екрані буде змінюватися.

В результаті тестування КПС працює коректно: після того, як ультразвуковий датчик відстані зафіксує перешкоду, інформація про відстань виводиться на вбудований LCD екран, відповідно від відстані мигають світлодіоди різних кольорів та звучить застережливий звук.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 45 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ХОРОНИ ПРАЦІ

4.1 Значення адаптації в трудовому процесі

Будь-яка праця людини безпосередньо пов'язана із тим виробничим середовищем в якому вона працює. Працівник може нормально здійснювати трудову діяльність лише тоді, коли умови зовнішнього середовища відповідають зручним умовам для праці людини. Якщо умови стають несприятливими, то на протидію їм організм людини включає спеціальний механізм, який зберігає постійність внутрішнього середовища, або змінює його в межах допустимого. Такий механізм називається адаптацією. Під час створення комп'ютеризованої системи людині, щоб адаптуватися та виконати свою роботу потрібні наступні фактори: необхідне технічне обладнання, обстановка в якій би не було лишніх звуків, адже для створення будь-якого проекту необхідний спокій та сконцентрованість.

Адаптація – це динамічний процес пристосування організму та його органів до мінливих умов зовнішнього середовища.

У трудовій діяльності процес адаптація поділяється на:

- Фізіологічна адаптація – це відчуття людини, що сприймаються її організмом. Людина реагує та сприймає як негативно так і позитивно зміни зовнішнього середовища, які впливають на неї [11].

- Психічна адаптація – це процес на підсвідомому рівні людини. Вона встановлює деякі оптимальні відповідності особистості до навколишнього середовища в процесі праці. Зрозуміло, що такі властивості, як гальмування мислення та низька швидкість переробки інформації, обмежений діапазон сприйняття, порушення функції пам'яті гальмують адаптацію; висока рухливість нервових процесів, навпаки, її підвищує [11].

| | | | | | | | | |
|-------------|------|-----------------|--------|------|---|---------------------------|------|---------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | | | |
| Змн. | Арк. | № докум. | Підпис | Дата | | | | |
| Розроб. | | Ярмусь О.С. | | | РОЗДІЛ 4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ХОРОНИ ПРАЦІ | Літ. | Арк. | Акрушів |
| Перевір. | | Осухівська Г.М. | | | | | 46 | 7 |
| Н. Контроль | | Луцик Н.С. | | | | ТНТУ, каф. КС, гр. СІс-43 | | |
| Консультант | | Лазарюк В.В. | | | | | | |
| Затверд. | | Осухівська Г.М. | | | | | | |

- Соціальна адаптація – це пристосування працівника до відносин у робочому колективі з його поведінкою, нормами, характером, правилами [11].
- Професійна адаптація – це адаптація до трудової діяльності з усіма її складовими: адаптація до робочого місця, знарядь та засобів праці, об'єктів та предметів праці, особливостей технологічного процесу, часових параметрів роботи тощо [11].

Кожен із видів адаптації впливає на працездатність та здоров'я працівника, формує у нього певний рівень чутливості та стійкості до психоемоційних перевантажень, внаслідок розвитку яких може істотно змінитися надійність професійної діяльності [11].

В результаті, щоб вдало працювати над проектом людині потрібно: спокій робоча атмосфера та необхідне обладнання під рукою.

4.2 Розрахунок та розробка схеми захисного занулення

Захисне занулення в електроустановках полягає в приєднанні до багаторазового заземленого нульового проводу електричної мережі корпусів та інших конструктивних металевих частин електрообладнання, які не знаходяться під напругою, але внаслідок пошкодження ізоляції можуть опинитися під нею. На рис. 4.1 зображена принципова схема занулення.

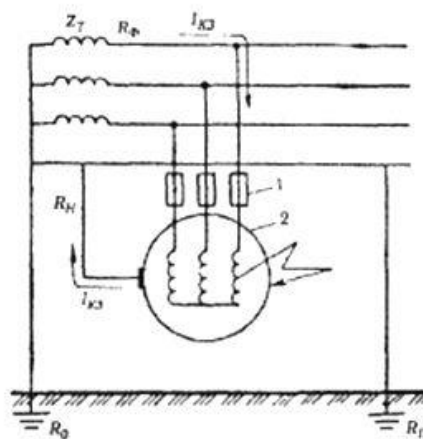


Рисунок 4.1 – Принципова схема занулення

Призначення занулення – усунення небезпеки ураження електричним струмом у випадку порушення ізоляції і появи на корпусах обладнання небезпечної напруги.

Принцип дії занулення – перетворення пробивання на корпус в одноразове коротке замикання, це замикання між фазовим та нульовим проводами з метою створення струму, здатного забезпечити спрацювання захисту і завдяки цьому автоматично від'єднати пошкоджену установку від живлячої електромережі [12].

У разі пробиванні на корпус у колі короткого замикання виникає великий струм короткого замикання, який забезпечує швидке перегорання плавких вставок протягом 5...7 с або відключення пошкоджених фаз автоматичними пристроями, які реагують на струм короткого замикання за 1...2 с. Протягом часу, котрий залежить від швидкості спрацювання захисту, людина, що торкається пошкодженого обладнання, опиняється під фазовою напругою. Якщо захисне занулення не спрацює протягом визначеного часу, то людина може бути уражена електричним струмом [12].

Загальні вимоги електробезпеки повинні відповідати ГОСТу-12.1.030-81 ССБТ. Для захисту від уражень електричним струмом використовують захисне заземлення. Воно повинно захищати людей від уражень електричним струмом у випадку дотику до металевих неструмопровідних частин, які можуть опинитись під напругою внаслідок пошкодження ізоляції, це досягається з'єднанням металевих частин електроустановок з землею, або її еквівалентом [12].

Згідно з класифікацією приміщень за ступенем небезпеки ураження електричним струмом (ПУЕ 1.1.6.), приміщення в якому проводяться всі роботи відноситься до першого класу (без підвищеної небезпеки). Під час роботи використовуються електроустановки з напругою живлення 36В, 220В, та 360В. Опір контура заземлення повинен мати не більше 4Ом.

Розрахунок проводять за допомогою методу коефіцієнта використання (екранування) електродів. Коефіцієнт використання групового заземлювача η – це відношення діючої провідності цього заземлювача до найбільш можливої його провідності за нескінченно великих відстаней між його електродами.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 48 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Коефіцієнт використання вертикальних заземлювачів η_v в залежності від розміщення заземлювачів та їх кількості знаходиться в межах 0,4...0,99. Взаємну екрануючу дію горизонтального заземлювача (з'єднувальної смуги) враховують за допомогою коефіцієнта використання горизонтального заземлювача η_g .

При замиканні фазного проводу на помилково заземлений корпус електроустановки струм корпусного замикання визначається вираженням:

$$I_{K3} = \frac{U_{\phi}}{R_0 + \frac{R_3 * R_{\text{ч}}}{R_3 + R_{\text{ч}}}} \quad (4.1)$$

або з урахуванням того, що $R_3 \ll R_{\text{ч}}$, можна записати,

$$I_{K3} = \frac{U_{\phi}}{R_0 + R_3} \quad (4.2)$$

де U_{ϕ} – фазна напруга мережі;

R_0 – опір розтіканню струму робочого заземлення нейтралі, значення якого нормується «ПУЕ».

Прийнявши $U_{\phi} = 220\text{В}$ і $R_0 = R_3 = 4\text{Ом}$, визначимо струм корпусного замикання:

$$I_{K3} = \frac{220}{4+4} = 27,5\text{А} \quad (4.3)$$

З наведених виразів видно, що в мережі із глухозаземленою нейтраллю струм корпусного замикання I_{K3} обмежується не опором ізоляції, як у мережі з ізолюваною нейтраллю, а незначними по величині опорами R_0 і R_3 , внаслідок чого цей струм завеликий та порівнянний зі струмом корпусного замикання.

При цьому напруга на корпусі електроустановки відносно землі буде дорівнювати:

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 49 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

$$U_K = I_{K3} * R_3 = 27,5 * 4 = 110V = 0,5U_{\phi} \quad (4.4)$$

Струм, що протікає через тіло людини, яка доторкнулася до цього корпусу, визначається як:

$$I_{ч} = \frac{U_K}{R_{ч}} = \frac{110}{1000} = 110mA \quad (4.5)$$

З аналізу наведених вище значень можна зробити наступні висновки:

- захисне заземлення, хоча й шунтує тіло людини, однак через велике значення загального струму замикання I_{K3} , навіть менша частина цього струму, що проходить через тіло людини $I_{ч}$ виявляється вище гранично припустимого рівня та є небезпечною для його життя;
- напруга на корпусі електроустановки U_K хоча і знижується захисним заземленням, але залишається вище гранично припустимого рівня [12];
- струм замикання I_{K3} , обмежений опорами R_0 і R_3 , виявляється недостатнім для надійного спрацювання захисних пристроїв (автоматичних вимикачів, плавких запобіжників і ін.) та відключення пошкодженої електроустановки.

Так у наведеному числовому прикладі струм силою $I_{K3} = 27,5A$ може привести до перегорання плавкої вставки запобіжника номінальним струмом не більш 9A. Відключення ж електроустановок, захищених запобіжниками з більшими номінальними струмами плавких вставок, забезпечене не буде.

Таким чином, у мережах напругою до 1000В з глухозаземленою нейтраллю дія захисного заземлення не ефективна, чим і пояснюється неприпустимість його застосування в цих мережах [12].

4.3 Перша домедична допомога людині, уражену електричним струмом

Працюючи з будь-якою комп'ютерною системою, людина наражає себе на небезпеку бути ураженою електричним струмом. Працюючи з мікросхемами

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 50 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

сильних електричних ударів не отримаєш, можуть бути електричні знаки – це сірі або блідо-жовті плями в місці ураження струмом. Вони безболісні та через деякий час зникають. Також можна отримати електричні опіки, які виникають внаслідок нагрівання тканин тіла людини струмом понад 1А під час різних коротких замикань, якщо сильно нагрітий провідник. До таких наслідків може призвести неправильне поводження з технікою, замикання контактів і т.п.

Відповідно «Порядку надання домедичної допомоги постраждалим при ураженні електричним струмом та блискавкою» [13] є ряд встановлених правил надання першої домедичної допомоги потерпілим раженим електричним струмом.

Якщо ж під час роботи з КС людина отримала сильне ураження електричним струмом в першу чергу потрібно:

- не зволікати;
- звільнити потерпілого від струмопровідних частин обладнання;
- швидко відключити від мережі ту частину електрообладнання, до якої доторкається людина.

Після звільнення потерпілого від дії струму, потрібно відразу надати йому першу домедичну допомогу. Так як при ураженні струмом смерть часто буває клінічною, і не слід зважати на відсутність дихання, пульсу, серцебиття.

Можна виділити три стани людського організму внаслідок дії електроструму і відповідні заходи необхідної допомоги:

- потерпілий при свідомості, йому слід забезпечити повний спокій, 2-3 годинне спостереження і обов'язковий огляд лікаря.
- якщо потерпілий непритомний, але дихає, його потрібно покласти горизонтально, розстебнути комір і пасок, привести до свідомості (дати нюхати нашатирний спирт) [13].
- якщо відсутнє дихання або ускладнене потрібно зробити штучне дихання та непрямий масаж серця. Перед тим як робити штучне дихання необхідно закинути потерпілому голову, підкласти під шию якийсь предмет так, щоб шия була припіднята, але голова була притиснута до землі. Так кисень потраплятиме в легені потерпілого. При диханні рот в рот здійснюють по два вдихи через кожних 30 натискань на проекцію серця до прощупування пульсу.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 51 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

При неможливості даного способу допустимо використовувати тільки непрямий масаж серця [13].

Непрямий масаж серця (рис. 4.2) – потрібно робити протягом перших 3 хвилин після того, як зупинилося серце після чого він стає менш ефективним. Потерпілого кладуть на спину на рівній поверхні, той хто надає першу медичну допомогу повинен випрямленими руками в ліктях виконуючи близько ста ритмічних натискань на протязі однієї хвилини на грудну клітку з частотою натискань 5-6см і натискати потрібно доти доки не розправиться грудна клітка [13].

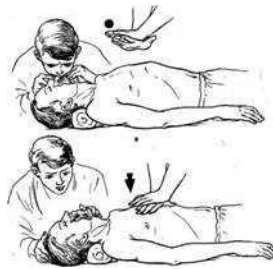


Рисунок 4.2 – Непрямий масаж серця

Якщо швидка ще не приїхала, можна самотужки обробити опіки, при цьому накласти сухі марлеві пов'язки. Якщо потерпілий знаходиться у свідомості, йому можна дати знеболюючі та заспокійливі засоби.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 52 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

ВИСНОВОК

У результаті виконання кваліфікаційної роботи бакалавра було розроблено комп'ютеризовану паркувальну систему для транспортних засобів з виводом інформації на екран, з світловою індикацією та звуковим сповіщенням.

КПС розроблена на базі мікроконтролера ATmega 328P платформи Arduino UNO у середовищі програмування Arduino IDE.

У першому розділі описано аналіз вимог до комп'ютеризованої паркувальної системи та аналіз можливих рішень поставлених завдань. Розглянуто роботу можливих аналогів та розроблена блок-схема для виконання системи.

Другий розділ включає в себе розробку узагальненої структури комп'ютеризованої паркувальної системи, обґрунтування апаратного забезпечення (плата ArduinoUNO, текстовий екран 16x2 – LCD дисплей МТ–16S2Н, ультразвуковий датчик відстані HC-SR04, відтворювач звуку НРА17А) та опис роботи середовища програмування Arduino IDE. А також, розроблено електричну принципову схему комп'ютеризованої паркувальної системи для транспортних засобів.

В розділі три представлено реалізацію проектних рішень та результати тестування комп'ютеризованої паркувальної системи. Описано алгоритм роботи програми та розроблене програмне забезпечення.

У четвертому розділі розглянуті питання з основ безпеки життєдіяльності та охорони праці.

В цілому, розроблена комп'ютерна паркувальна система реагує на перешкоди на відстані 0,5м. Повідомляє про виявлені перешкоди, видаючи звуковий сигнал та виводить повідомлення про точну відстань до об'єкту.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 53 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Комп'ютерна паркувальна система: Веб-сайт.. URL: <https://www.webl.io.jp/content/%E3%83%91%E3%83%BC%E3%82%AF%E3%83%88%E3%83%AD%E3%83%8B%E3%83%83%E3%82%AF%E3%82%B7%E3%82%B9%E3%83%86%E3%83%A0>
2. Автоматизована система паркування: Веб-сайт.. URL: https://en.wikipedia.org/wiki/Automated_parking_system
3. Комп'ютерна паркувальна система (парктронік): Веб-сайт.. URL: <https://rozetka.com.ua/ua/153237050/p153237050/>
4. ARDUINO UNO R3: Веб-сайт.. URL: <https://chip.by/product/arduino-uno-kupit-v-minske>
5. LCD дисплей МТ–16S2Н: Веб-сайт.. URL: <https://mugwa.elama.pp.ua/prem/ekran-16h2.html>
6. Ультразвуковий датчик відстані HC-SR04: Веб-сайт.. URL: <https://arduino.ua/prod182-yltrazvykovo-i-datchik-rasstoyaniya-hc-sr04>
7. Виявлення руху за допомогою ультразвукового датчика присутності: Веб-сайт.. URL: <http://mikrotik.kpi.ua/index.php/courses-list/category-arduino/27-how-to-detect-the-presence-of-strangers-using-sensors-and-arduino>
8. Потенціометр: Веб-сайт.. URL: <https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D1%82%D0%B5%D0%BD%D1%86%D1%96%D0%BE%D0%BC%D0%B5%D1%82%D1%80>
9. Робота із середовищем Arduino IDE: Веб-сайт.. URL: https://robohobby.by/lessons/arduino_ide_tutorial/
10. Програмування Ардуїно: Веб-сайт.. URL: <https://doc.arduino.ua/ru/prog/>
11. Значення адаптації в трудовому процесі: Веб-сайт.. URL: <https://library.if.ua/book/9/921.html>
12. Розрахунок занулення: Веб-сайт.. URL: https://cpo.stu.cn.ua/Oksana/rozrah_rozd_OP_DP_bak_spec_mag/120.html

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 54 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

13. Порядок надання домедичної допомоги постраждалим при ураженні електричним струмом та блискавкою: Веб-сайт.. URL: <https://zakon.rada.gov.ua/laws/show/z0775-14#Text>

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | Арк. |
| | | | | | | 55 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

Додаток А
Технічне завдання

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Тернопільський національний технічний університет імені Іван Пулюя
Факультет комп'ютерно-інформаційних систем і програмної інженерії

Кафедра комп'ютерних систем та мереж

“Затверджую”

Завідувач кафедри КС _____ Осухівська Г.М.

“ ___ ” _____ 2021 р

КОМП'ЮТЕРИЗОВАНА ПАРКУВАЛЬНА СИСТЕМА ДЛЯ
ТРАНСПОРТНИХ ЗАСОБІВ

ТЕХНІЧНЕ ЗАВДАННЯ на ___ листках

На здобуття освітньо-кваліфікаційного рівня бакалавр

Напрямок 123 Комп'ютерна інженерія

Спеціальність 123 Комп'ютерна інженерія

«УЗГОДЖЕНО»

Керівник кваліфікаційної роботи

_____ к.т.н., зав. каф. КС

Осухівська Г.М.

« ___ » _____ 2021 р

«ВИКОНАВЕЦЬ»

Студент групи СІс-43

_____ Ярмусь О.С.

« ___ » _____ 2021 р

Тернопіль 2021

1. Назва та підстава для виконання роботи

1.1 Комп'ютеризована паркувальна система для транспортних засобів

1.2. Підставою для виконання кваліфікаційної роботи бакалавра (КРБ) є Наказ по Університету (№ 4/7-180 від 23.03.2022 р.).

2. Виконавець

2.1. Студент групи СІс-44 кафедри КС

Тернопільського національного технічного університету ім. І. Пулюя
Ярмусь Олександр Сергійович

3. Мета роботи

3.1. Метою роботи є розробити структуру та програмне забезпечення для комп'ютеризованої паркувальна система для транспортних засобів

4. Склад виробу

4.1. До складу КС повинні входити:

- 1) Плата ArduinoUNO.
- 2) Текстовий екран 16x2. LCD дисплей МТ-16S2Н.
- 3) Ультразвуковий датчик відстані HC-SR04.
- 4) Відтворювач звуку НРА17А.
- 5) Змінний резистор (потенціометр).
- 6) Світлодіоди.
- 7) Макетна плата.

5. Технічні вимоги

5.1. Вимоги по призначенню.

5.1.1. Комп'ютеризована система повинна мати наступні параметри:

- Охоплення відстані: 0,5м;
- Точність вимірювання 3мм;
- Висока частота сигналу;

5.2. Вимоги до умов експлуатації:

5.2.1. По умовам експлуатації виріб повинен відповідати вимогам ГОСТ 15150 для УХЛ4.1

5.2.2. Працювати при кліматичних умовах -20°C до $+60^{\circ}\text{C}$.

5.3. Конструктивні вимоги.

5.3.1. Конструювання корпусу приладу в КРБ не передбачено.

5.3.2. Для побудови системи мають бути використані сучасні компоненти

5.3.3. При побудові системи необхідно передбачити розміщення роз'ємів живлення і обміну даними.

5.3.4. Габаритні розміри при макетуванні, мм, не більше:

– довжина 300мм

– ширина 200мм

– висота 80мм

5.3.5. Маса макету, кг, не більше 2

5.3.6. Конструкція макету повинна забезпечувати доступ до всіх комплектуючих виробів при тестуванні.

5.4. Вимоги до надійності.

5.4.1. Система повинна відповідати вимогам ДСТУ 2862-94.

5.4.2. Наробка на відмову, не менше 5000 год.

5.5. Вимоги метрології.

5.5.1. Вимірювання параметрів системи при моделюванні повинно виконуватись на універсальних вимірювальних приладах.

6. Економічні показники

6.1. Собівартість КС не повинна перевищувати 1000 грн.

7. Вимоги до документації

7.1. Конструкторська документація повинна відповідати вимогам ЄСКД, ДСТУ та ГОСТ.

8. Стадії та етапи розробки КРБ

8.1 Стадії та етапи виконання КРБ наведенні в табл. 1.

Таблиця 1 – Стадії та етапи виконання КРБ

| № п/п | Назва етапу | Строки виконання | |
|----------|----------------------|------------------|------------|
| | | початок | кінець |
| 1 | Технічне завдання | – | 26.03.2022 |
| 2 | Розділ 1 | 23.03.2022 | 15.04.2022 |
| 3 | Розділ 2 | 03.04.2022 | 30.04.2022 |
| 4 | Розділ 3 | 01.05.2022 | 07.06.2022 |
| 5 | Розділ охорона праці | 08.06.2022 | 14.06.2022 |
| 6 | Нормаконтроль | 15.06.2022 | 21.06.2022 |
| 7 | Захист | з 22.06.2022 | – |

9. В дане ТЗ можуть вноситись зміни по узгодженню сторін

Додаток Б
Перелік документів

| Позначення | Найменування | Кількість | Примітка |
|------------|---------------|-----------|----------|
| D1 | LED1 | | |
| D2-D4 | LED-GREEN | | |
| D5-D7 | LED-YELLOW | | |
| D8-D10 | LED-RED | | |
| R1-R4 | MF 47 1W 470m | | |
| R5-R13 | 2200m | | |
| K1 | Кнопка | | |
| Мікросхеми | | | |
| U1 | Arduino 328 | 1 | |
| U2 | HCSR04 | 1 | |
| U3 | SPEACER | 1 | |
| U4 | OLED12832I2C | 1 | |

| | | | | | | | | |
|-----------|-----------------|--|--|--|--|---------------------------|----|---------|
| | | | | | КС КРБ 123.237.00.00 ПЗ | | | |
| Змн. | | | | | | | | |
| Розроб. | Ярмусь О.С. | | | | Комп'ютеризована паркувальна система для транспортних засобів. | Літ. | | Акрушіє |
| Перевір. | Осухівська Г.М. | | | | | | 61 | 1 |
| Реценз. | Млинко Б.Б. | | | | | ТНТУ, каф. КС, гр. СІс-43 | | |
| Н. контр. | Луцик Н.С. | | | | | | | |
| Затверд. | Осухівська Г.М. | | | | | | | |

Додаток В
Лістинг програми

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#define trigPin 12
#define echoPin 13
#define led 2
#define led2 3
#define led3 4
#define led4 5
#define led5 6
#define led6 7
#define led7 8
#define led8 9
#define led9 10
#define buzzer 11

LiquidCrystal_I2C lcd(0x27, 16, 2);

int sound = 250;

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  pinMode(led, OUTPUT);
  pinMode(led2, OUTPUT);
```

```
pinMode(led3, OUTPUT);
pinMode(led4, OUTPUT);
pinMode(led5, OUTPUT);
pinMode(led6, OUTPUT);
pinMode(led7, OUTPUT);
pinMode(led8, OUTPUT);
pinMode(led9, OUTPUT);
pinMode(buzzer, OUTPUT);

lcd.init();

}

void loop()
{
lcd.begin(16, 2);
  long duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1;
  lcd.print("Distance = 50 cm");
  lcd.setCursor(2,1);
  if (distance <= 50) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
```

```
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led, HIGH);
}
else {
    digitalWrite(led,LOW);
}
if (distance < 45) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led2, HIGH);
}
else {
    digitalWrite(led2, LOW);
}
if (distance < 40) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led3, HIGH);
}
else {
    digitalWrite(led3, LOW);
}
if (distance < 35) {
    lcd.backlight();
```



```
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led4, HIGH);
    tone(11, 5000);
    delay(200);
noTone(11);
}
else {
    digitalWrite(led4,LOW);
}
if (distance < 30) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led5, HIGH);
}
else {
    digitalWrite(led5,LOW);
}
if (distance < 25) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led6, HIGH);
}
```

```
else {
    digitalWrite(led6,LOW);
}
if (distance < 20) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led7, HIGH);
    tone(11, 7500);
    delay(300);
noTone(11);

}
else {
    digitalWrite(led7,LOW);
}
if (distance < 15) {
    lcd.backlight();
    lcd.print("Distance ");
    lcd.print(distance);
    lcd.print(" cm");
    lcd.setCursor(2,2);
    digitalWrite(led8, HIGH);
}
else {
    digitalWrite(led8,LOW);
}

if (distance < 10) {
```

```
        lcd.backlight();
        lcd.print("Distance ");
        lcd.print(distance);
        lcd.print(" cm");
        lcd.setCursor(2,2);
        digitalWrite(led9, HIGH);

    }
    else {
        digitalWrite(led9,LOW);
    }
}
```