

Міністерство освіти і науки України  
Чорноморський національний університет імені Петра Могили  
Факультет комп'ютерних наук  
Кафедра комп'ютерної інженерії

ДОПУЩЕНО ДО ЗАХИСТУ  
Завідувач кафедри, канд. техн.  
наук, доцент  
\_\_\_\_\_ Я. М. Крайник  
« \_\_ » \_\_\_\_\_ 2022 р.

## КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Галузь знань: 12 Інформаційні технології  
Спеціальність: 123 Комп'ютерна інженерія  
Тема: **Система збору інформації для водія з виведенням критичних параметрів на смартфон**  
Шифр: 123 – БР.ПЗ.00 – 405.21810521

Виконав:  
студент 4 курсу, групи 405,  
спеціальності  
123 Комп'ютерна інженерія  
\_\_\_\_\_ Д. В. Полівода

Керівник:  
канд. техн. наук, доцент  
\_\_\_\_\_ Л. В. Солобуто

Миколаїв 2022

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Чорноморський національний університет імені Петра Могили**  
**Факультет комп'ютерних наук**  
**Кафедра комп'ютерної інженерії**

ЗАТВЕРДЖУЮ  
Завідувач кафедри,  
канд. техн. наук, доцент  
\_\_\_\_\_ Я. М. Крайник  
« \_\_ » \_\_\_\_\_ 2022 р.

**ЗАВДАННЯ**

**на виконання кваліфікаційної роботи**

Видано студенту групи 405 факультету комп'ютерних наук

Полівода Данило Володимирович

(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи:

Система збору інформації для водія з виведенням критичних параметрів на смартфон

Затверджена наказом по ЧНУ від «\_\_» \_\_\_\_\_ 2022 р. No \_\_\_\_\_

2. Строк представлення кваліфікаційної роботи «\_\_» \_\_\_\_\_ 2022 р.

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

Розроблена система має допомагати водієві під час парковки та відтворювати критичні параметри на смартфоні

---

---

4. Перелік питань, що підлягають розробці

Аналіз інформації стосовно методів та технологій систем допомоги водієві на базі електронних датчиків. Обрати алгоритм збирання інформації на базі Arduino Uno. Розробити та протестувати систему в умовах лабораторного приміщення. Зробити аналіз результатів тестування з метою подальшого вдосконалення системи збору інформації.

5. Перелік графічних матеріалів

Схема підключення апаратної частини системи,  
демонстрація інтерфейсів,

демонстрація програмного коду

6. Завдання до спеціальної частини

Розглянути основні державні будівельні норми України, щодо вентиляції та кондиціонування, забирання зовнішнього та викид витяжного повітря, витрати припливного повітря та організація повітрообміну. Оволодіти навиками розрахунку систем повітрообміну при загально обмінній вентиляції виробничих приміщень.

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
ст. викладач А. О. Алексєєва	кафедра екології Медичного інституту ЧНУ ім. Петра Могили	спеціальна частина з охорони праці

Керівник роботи \_\_\_\_ канд. тех. наук, доцент, Л. В. Солобуто \_\_\_\_

(посада, прізвище, ім'я, по батькові) (підпис)

Завдання прийнято до виконання \_\_ Полівода Данило Володимирович \_\_

(прізвище, ім'я, по батькові студента) (підпис)

Дата видачі завдання « \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р.

**КАЛЕНДАРНИЙ ПЛАН**  
**виконання кваліфікаційної роботи**

Тема: Система збору інформації для водія з виведенням критичних параметрів на смартфон.

№	Найменування роботи	Початок	Закінчення	Примітки
1.	Розробка та затвердження завдання на виконання ДР	21.10.2021	02.11.2021	Виконано
2	Огляд літератури за темою роботи	03.11.2021	30.11.2021	Виконано
3	Складання календарного плану КР	01.12.2022	15.12.2021	Виконано
4	Аналіз інформації стосовно технологій та методів збору інформації	10.01.2022	23.01.1021	Виконано
5	Підбір необхідних для реалізації системи компонентів	24.01.2022	10.03.2022	Виконано
6	Проектування схеми підключень апаратної частини системи	10.03.2022	15.04.2022	Виконано
7	Реалізація системи	15.04.2022	21.04.2022	Виконано
8	Перевірка працездатності, тестування розробленої системи, аналіз результатів тестування	22.04.2022	22.05.2022	Виконано
9	Відгук керівника КР	23.05.2022	01.06.2022	Виконано
10	Оформлення КР та презентації	02.06.2022	09.06.2022	Виконано
11	Попередній захист	10.06.2022	11.06.2022	Виконано
12	Рецензування	14.06.2022	18.06.2022	Виконано
13	Завершення оформлення КР та презентації	18.06.2022	24.06.2022	Виконано
14	Захист кваліфікаційної роботи	28.06.2022	28.06.2022	Виконано

Розробив студент \_\_\_\_\_ Полівода Данило Володимирович \_\_\_\_\_  
(прізвище, ім'я, по батькові) (підпис)  
«\_\_» \_\_\_\_\_ 2022 р.

Керівник роботи \_\_\_ канд. тех. наук, доцент, Л. В. Солобуто \_\_\_\_\_  
(посада, прізвище, ім'я, по батькові) (підпис)  
«\_\_» \_\_\_\_\_ 2022 р.

## АНОТАЦІЯ

до кваліфікаційної роботи

«Система збору інформації для водія з виведенням критичних параметрів на смартфон»

Студент: Полівода Данило Володимирович

Керівник: канд. тех. наук, доцент Солобуто Л. В.

Кваліфікаційна робота присвячена розробці системі збору інформації для водія з виведенням критичних параметрів на смартфон. Практичне значення результатів дослідження та розроблення полягає у можливості використання такої системи в автомобілях будь якого цінового діапазону.

Пояснювальна записка кваліфікаційної роботи складається зі вступу, трьох розділів, висновків та додатків.

У вступі визначається актуальність теми, сформульовані мета, об'єкт, предмет та завдання роботи.

У першому розділі аналізуються сучасні системи збору інформації, проводиться аналіз проблем цих систем та їх вирішення.

В другому розділі відбувається огляд методів, схеми та оцінка збору критичних параметрів.

У третьому розділі описано розробку програмної частини системи: обґрунтовано вибір елементів апаратної частини, описано алгоритм роботи, використані функції та інтерфейс програмних середовищ.

У висновках наведено аналіз виконаної роботи та отриманих результатів дослідження та розроблення.

В додатку А наводиться програмний код для систем збору критичних параметрів.

Кваліфікаційна робота складається з 73 сторінок, 31 рисунки, 3 додатків та 22 використаних джерел посилання.

Ключові слова: система збору інформації, автомобільні помічники, дальномір, спідометр, ультразвуковий датчик.

## **ABSTRACT**

of the Bachelor's Thesis

«Information collection system for the driver with output of critical parameters to the smartphone »

Student: Polivoda Danylo Volodymyrovych

Scientific Advisor: Ph.D. in Computer Systems and Components, Associate  
Professor L. V. Solobuto

Qualification work is devoted to the development of a system of information failure for the driver with the output of critical parameters on a smartphone. The practical significance of the results of research and development lies in the possibility of using such a system in cars of any price range.

The explanatory note of the qualification work consists of an introduction, three sections, conclusions and appendices.

The introduction determines the relevance of the topic, formulates the purpose, object, subject and objectives of the work.

The first section analyzes modern information collection systems, analyzes the problems of these systems and their solutions.

The second section reviews methods, schemes, and evaluates the collection of critical parameters.

The third section describes the development of the software part of the system: the choice of hardware elements is substantiated, the algorithm of work is described, the functions and the interface of software environments are used.

The conclusions provide an analysis of the work performed and the results of research and development.

Annex A provides code for critical parameter collection systems.

The qualification work consists of 73 pages, 31 figures, 3 appendices and 22 used reference sources.

Keywords: information collection system, car assistants, rangefinder, speedometer, ultrasonic sensor.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	9
ВСТУП.....	10
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД ТА ПОСТАНОВКА ЗАДАЧІ.....	12
1.1 Системи збору інформації для водіїв .....	12
1.2 Особливості систем збору інформації.....	18
1.2.1 Адаптивний круїз-контроль .....	18
1.2.2 Система екстреного гальмування .....	21
1.2.3 Паркувальні асистенти .....	25
1.4 Огляд сучасних систем збору інформації .....	28
1.5 Постановка задачі.....	32
Висновки до розділу 1.....	32
РОЗДІЛ 2 АНАЛІЗ ТА МЕТОДИ РОБОТИ АПАРАТНО-ПРОГРАМНОГО КОМПЛЕКСУ .....	33
2.1 Аналіз критичних параметрів .....	33
2.2 Фактори що впливають на збір критичних параметрів.....	36
2.3 Методи збору критичних параметрів .....	40
2.3.1 Далекомір оптичного діапазону.....	40
2.3.2 Спідометр.....	42
2.3.3 Ультразвукової датчик .....	44
Висновки до розділу 2.....	45
РОЗДІЛ 3 РЕАЛІЗАЦІЯ СИСТЕМИ ПАРКУВАННЯ .....	46
3.1 Фактори що впливають на збір критичних параметрів.....	46
3.1.1 Мікроконтролер (Arduino Uno).....	46
3.1.2 Lidar TF Luna.....	48
3.1.3 Ультразвуковий датчик HC-SR04 .....	50
3.1.4 Цифровий датчик Холла А3144 .....	52
3.2 Вибір технології та мови програмування.....	53
3.2.1 Вимоги до синтаксису і структури кода.....	55

---

3.2.2	Опис використаних функцій .....	57
3.3	Вибір компонентів АПЗ .....	59
	Бібліотека Ultrasonic.h .....	59
3.3.1	Бібліотека TFLidar.h .....	60
3.3.2	Бібліотека Stepper.h .....	60
3.3.3	Бібліотека SoftwareSerial.h .....	60
3.4	Опис інтерфейсів АПЗ.....	61
3.5	Реалізація системи .....	64
	Висновки до розділу 3.....	66
	В процесі розробки програмної частини системи збору критичних параметрів на базі мікроконтролера та датчиків були .....	66
	<b>ВИСНОВКИ</b> .....	67
	<b>ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ</b> .....	68
	<b>ДОДАТОК А КОД ПРОГРАМИ В ARDUINO IDE</b> .....	70



---

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ADAS	–	Advanced driver-assistance systems
ABS	–	Antilock Braking System
AEB	–	Autonomous Emergency Braking systems
APS	–	Acoustic Parking System
ENCAP	–	European New Car Assessment Programme
I2C	–	Inter-Integrated Circuit
MRR	–	Middle Range Radar
UART	–	Universal Asynchronous Receiver/Transmitter
ВООЗ	–	Всесвітня організація охорони здоров'я
МОН	–	Міністерство освіти і науки
ПЛІС	–	Програмована логічна інтегральна схема
ШИМ	–	Широтно-імпульсна модуляція

---

## ВСТУП

В сучасному світі більшість повсякденних завдань спрощується або автоматизується і кожного дня ця тенденція зростає. У побут сучасної людини щільно увійшли електроніка та техніка, що покращує їх життя, а саме: смартфони та автомобілі. Популярність цих систем, обумовлена прагненням людини до комфорту і зручності.

В наш час кількість автомобілів зростає з кожним роком. Все складніше стає переміщатися на автомобілі у великому місті. Автомобіль ХХІ століття буквально насичений розумними системами, які беруть під контроль практично всі параметри, в тому числі і процес парковки в умовах стислого простору.

Звичайно, така увага до якості переміщення, безаварійності руху, як і сама можливість подібного контролю з боку інтелектуальних технологій, приводить к подальшому розвитку інформаційних технологій. Далека перспектива автомобілебудування – це позбавити машину від рук водія, інакше кажучи, автономне керування фантастичного 5-го рівня, коли транспортний засіб рухається без участі людини.

**Мета:** розробка системи збору інформації для водія з виведенням критичних параметрів на смартфон.

**Об'єкт:** методи та технології визначення критичних параметрів при паркуванні автомобіля.

**Предмет:** апаратне та програмне забезпечення для організації безпечного руху.

Для досягнення поставленої мети необхідно вирішити такі **завдання:**

- провести аналітичний огляд літератури та патентної інформації, визначити переваги та недоліки існуючих систем;
- визначити технічні параметри та можливості системи, що проектується;

- 
- обґрунтувати вибір складових, комплектуючих та технологій для розроблення;
  - розробити схеми алгоритму роботи системи для парковки;
  - розробити програмне забезпечення для приладу;
  - створити систему паркування та протестувати її роботу;
  - оцінити перспективи подальших розроблень та удосконалень;
  - розробити питання з охорони праці та безпеки життєдіяльності.

**Практичне значення** отриманих результатів: готовий результат виконання цієї роботи можна використовувати не тільки в системах парктроніку, а й при переміщенні по дорогах, особливо в умовах недостатнього освітлення, що забезпечує безпеку руху.

**Апробація:** матеріали по темі дипломної роботи були представлені у форматі тез на Ольвійському форумі, який проводився з 23.06.2022 по 26.06.2022

---

## РОЗДІЛ 1

### АНАЛІТИЧНИЙ ОГЛЯД ТА ПОСТАНОВКА ЗАДАЧІ

#### 1.1 Системи збору інформації для водіїв

При покупці автомобіля дедалі частіше визначальним фактором стає наявність систем допомоги водієві. За статистикою, при придбанні нових автомобілів кожен п'ятий легковий автомобіль оснащений такими системами. При цьому у 2013 році системи допомоги були встановлені лише у кожному десятому новому автомобілі. Якби всі автомобілі були оснащені системою такими системами, можна було б запобігти 72% ДТП, у яких постраждали люди.

За даними ВООЗ, ситуація з ДТП погіршується. Смертність унаслідок аварій на дорогах зросла до 1,35 млн людей на рік, або в середньому близько 3700 летальних випадків щодня. Крім того, десятки мільйонів дістають поранення чи інвалідність щороку або їхнє життя змінюється через отримані травми. Ці втрати істотно впливають на життя сім'ї та суспільства загалом.

Підвищення безпеки, що забезпечується системами допомоги водієві, є однією з причин їхньої популярності. Системи збору інформації для водіїв – це обов'язковий крок на шляху до глобальної мети знизити рівень смертності на дорогах до нуля. Ця ж мета знаходить своє відображення і в рейтингу безпеки нових авто Euro NCAP, що сприяє розвитку систем попередження аварійного гальмування та систем контролю руху. Усе це впливає обсяги виробництва датчиків, що необхідні для роботи таких систем як: адаптивний круїз контроль з системою екстреного гальмування та паркувальні асистенти.

Впровадження таких систем гальмується через наступні фактори:

- висока вартість більшості систем (причому, чим якісніша система, тим вона дорожча), тому комерційні перевізники економлять гроші, а муніципальні часто не мають достатніх бюджетних коштів;

- невідповідність дорожньої інфраструктури: як і у випадку з безпілотними технологіями великою перешкодою для використання в Україні подібних систем є систематична відсутність дорожньої розмітки (або те, що вона не видно під снігом/брудом), що відразу унеможлиблює реалізацію контролю руху в смузі;
- слабка адаптація рішень під замовників в Україні, для перевізників було б важливо, щоб системи не лише попереджала їх водіїв, а й мала певний інтерфейс диспетчера/керівника, де відображаються тривожні події та формуються звіти з безпечного водіння;
- відсутність нормативної бази, яка б забезпечила впровадження систем як мінімум у критично важливих транспортних областях – перевезення шкільними автобусами, «далекобійні» вантажоперевезення, перевезення небезпечних вантажів тощо.

Ці системи є комплектом з камери переднього вигляду, камери «на водія» та контролера обробки даних. Вони забезпечують безперервний контроль дорожньої ситуації попереду автомобіля, а також контроль стану водія.

Спільнота автомобільних інженерів (Society of Automotive Engineers, SAE) підготувала стандарт SAE J3016, стандарт класифікації автоматизованих систем управління рухом ТС SAE. Оскільки класифікація з цього стандарту стала найпоширенішою, вона наведена у таблиці. 1.1.

Перевага цих систем у тому, що вони не вимагають тісної інтеграції з самим автомобілем – їх елементи встановлюються на зовнішні елементи панелі приладів та/або лобового скла. Максимум, що потрібне для їх функціонування – це підключення комплекту до бортової мережі автомобіля для електроживлення.

Таблиця 1.1 – Стандарт класифікації автоматизованих систем

№	Назва	Опис	
0	Без автоматизації	Весь процес водіння контролює водій, завдання керування динамікою автомобіля лежить повністю на ньому, навіть якщо ТЗ оснащено системою попередження та впливу.	Водій проводить моніторинг навколишнього обстановки самостійно
1	Що включає допомогу водієві	Сприяння водію в керуванні кермової системою або гальмами та педаллю подачі палива, використовуючи інформацію про зовнішнє середовище. Система чекає, коли водій зробить всі дії, що залишилися, крім тих, які видає система.	
2	Частково автоматизований	Режим управління включає дві системи, які здатні впливати і на рульове управління і на прискорення/гальмування, спільно функціонуючи.	
3	Умовно автоматизований	Умовно автоматизований динамікою автомобіля, але тільки в тому випадку, якщо водій відповідним чином відреагує на її сигнал та втрутиться.	Система автоматизації водіння буде слідкувати за зовнішнім середовищем
4	Високо автоматизований	Система автоматизації водіння бере на себе управління ТЗ, навіть якщо водій не відреагував на вимогу втрутитися у процес управління, тобто дозвіл водія не є необхідним. Рішення приймає сама система управління динамікою автомобіля.	
5	Повністю автоматизований	Система автоматизації водіння повністю бере на себе ті ж функції, що і водій, при цьому вона може працювати в будь-яких зовнішніх умовах середовища.	

Безпілотні автомобілі здаються фантастикою, але більшість необхідних технологій вже представлена на ринку. Ці технології отримали назву інтелектуальних систем допомоги водієві (Advanced Driver Assistance Systems – ADAS) і є вбудованими функціями, які допомагають водієві уникнути аварії (активна безпека) або зменшити тяжкість її наслідку (пасивна безпека). Автовиробники вбудовують все більше таких систем у нові моделі, щоб зробити керування більш зручним та безпечним.

Системи допомоги водієві поділяються на три основні категорії:

– підказки водієві;

- попередження водію;
- безпосередня допомога водієві.

Функція підказок пов'язані з виявленням перешкод: наприклад, нічне бачення і адаптивна система освітлення, які регулюють фари на згинах дороги. До них входить система розпізнавання дорожніх знаків, що обмежують швидкість, та система кругового огляду, що допомагає безпечно паркуватися.

Системи попередження водія виконують функцію сигналізації про можливу небезпеку. Наприклад, оповіщення системи допомоги при паркуванні, системи попередження про лобове зіткнення та виїзд із займаної смуги.

Третя група включає функції допомоги водієві, що здійснюють активну стабілізацію автомобіля або безпосереднє маневрування. Адаптивний круїз-контроль регулює швидкість автомобіля для того, щоб підтримувати постійну відстань до транспортного засобу попереду вашого транспортного засобу; система екстреного гальмування дозволяє уникнути або знизити наслідки удару в транспортний засіб перед вами, а система підтримки смуги руху допомагає запобігти зміщенню за межі смуги.

На сьогоднішній день можна виділити 43 інтелектуальні системи допомоги водієві:

- 1) Система контролю тиску в шинах (TPMS).
- 2) Помічник гальмування (Brakeassist, BAS, HBB, EBA, AFU).
- 3) Система визначення перешкод (Forward Collision Warning).
- 4) Система розпізнавання об'єктів, що рухаються (Moving Object Detection).
- 5) Система допомоги при повороті (Turningassistant).
- 6) Помічник проїзду перехресть (Intersectionassistant).
- 7) Система допомоги при виїзді з паркування заднім ходом (Rear Cross Traffic Alert).
- 8) Система розпізнавання пішоходів (Pedestrian Detection System).

- 
- 9) Система передаварійної безпеки (Collisionavoidancesystem, Precrashsystem).
  - 10) Круїз контроль / Адаптивний круїз-контроль (Cruisecontrol / Adaptivecruisecontrol).
  - 11) Алкозамок (AlcoholInterlockDevice).
  - 12) Помічник спуску зі схилу (Hill descent control).
  - 13) Система при торканні автомобіля на підйомі (HAS).
  - 14) Помічник руху в пробці (Traffic Jam Assistant).
  - 15) Магістральний автопілот (Highway pilot).
  - 16) Адаптивна система освітлення (Adaptivelightcontrol: swivellingcurvelights).
  - 17) Система автоматичного перемикання далекого/ближнього світла (Glare-free high beam and pixel light).
  - 18) Система контролю сліпих зон (Blind spot monitor).
  - 19) Система допомоги при перебудові (Lane change assistance).
  - 20) Система контролю сходу зі смуги руху (Lanedepturewarningsystem).
  - 21) Система утримання у смузі (Lane Keep Assist).
  - 22) Система контролю втоми водія (Driverdrowsinessdetection).
  - 23) Екстрений помічник водія (Emergencydriverassistant).
  - 24) Система допомоги при паркуванні (Park Assist System).
  - 25) Інтелектуальна система допомоги при паркуванні з дистанційним керуванням (RemoteParkAssistSystem).
  - 26) Помічник паркування з причепом (Trailerparkingassist).
  - 27) Антиблокувальна система гальм (ABS), включаючи протибуксувальну систему (TCS).
  - 28) Система кругового огляду (Surround View system).
  - 29) Система розпізнавання сигналів світлофора (Traffic Light Recognition).
  - 30) Система розпізнавання дорожніх знаків (Traffic sign recognition).



- 
- 31) Сигнал про неправильний напрямок руху (Wrong-way driver warning, Wrong Way Alert)
  - 32) Помічник перевищення/адаптації швидкості (Intelligent Speed Adaptation)
  - 33) Система нічного бачення (Automotive night vision).
  - 34) Відеореєстратор (Video Recorder).
  - 35) Автомобільна навігаційна система (Automotive navigation system).
  - 36) Сигнальні звуки електромобілів (Electric vehicle warning sounds).
  - 37) Система захисту пішоходів (Pedestrian protection system).
  - 38) Датчик дощу світла (Rain & light sensor).
  - 39) Система зв'язку між автомобілями (Vehicular communication systems, V2V).
  - 40) Система курсової стійкості (ESP).
  - 41) Помічник руху з причепом (Trailer assist).
  - 42) Система розпізнавання багажника на даху (Roof rack detection).
  - 43) Помічник бокового вітру (Crosswind stabilization).

Подібні системи дозволяють визначати та попереджати водія звуковим сигналом про наступні події:

- ризик зіткнення з автомобілем, що йде попереду;
- ризик зіткнення з пішоходом/велосипедистом.

Від того, яка система використовується в автомобілі, залежать всі подальші характеристики і ефективність безпеки. Системи збору інформації з подальшим виведенням критичних параметрів можна класифікувати за:

- принципом роботи;
- габаритами;
- вартістю;
- виробником;
- стандартом автоматизації.

## 1.2 Особливості систем збору інформації

### 1.2.1 Адаптивний круїз-контроль

Тривалі подорожі на автомобілі для багатьох водіїв стають справжнім випробуванням. Втома і напруга за кермом перетворюють довгоочікуваний сімейний відпочинок на болісне очікування спокою. Справжнім порятунком є наявність у автомобілі круїз-контролю. Пристрій, придуманий у 50-х роках минулого століття у США, поступово завоював популярність у всьому світі.

Круїз-контролем називають спеціальний пристрій в автомобілі, здатний без участі водія підтримувати задану швидкість при автоматичному натисканні педалі газу у разі зниження швидкості, а також її збільшення на спусках. Найзручніше користуватися круїз-контролем у поїздках на далекі відстані, оскільки в міських умовах водіям рідко вдається пересуватися з постійною швидкістю.

Як і базова система круїз-контролю, яка протягом багатьох років поставлялася у вигляді компонента стандартної комплектації, система «адаптивного круїз-контролю» (ACC) може вважатися системою підвищення безпеки. Контролер швидкості автомобіля регулює швидкість руху до необхідної величини, заданої водієм, за допомогою органу управління круїз-контролю. На додаток до цієї функції, адаптивний контролер швидкості автомобіля аналізує інформацію про відносну швидкість і відстань до автомобіля, що йде попереду, а також про додаткові параметри руху керованого вами автомобіля (наприклад, кут повороту, швидкість повороту) і використовує дані для регулювання відстані між автомобілями. Таким чином, регулюється швидкість і підтримується безпечна відстань до автомобіля, що йде попереду. Адаптивний контролер швидкості автомобіля обладнаний радаром далекої дії, щоб виявляти автомобілі, що рухаються попереду по тій же смугі, а також перешкоди, що рухаються в межах дальності дії датчиків і вимагають застосування гальм автомобіля (рис. 1.1).

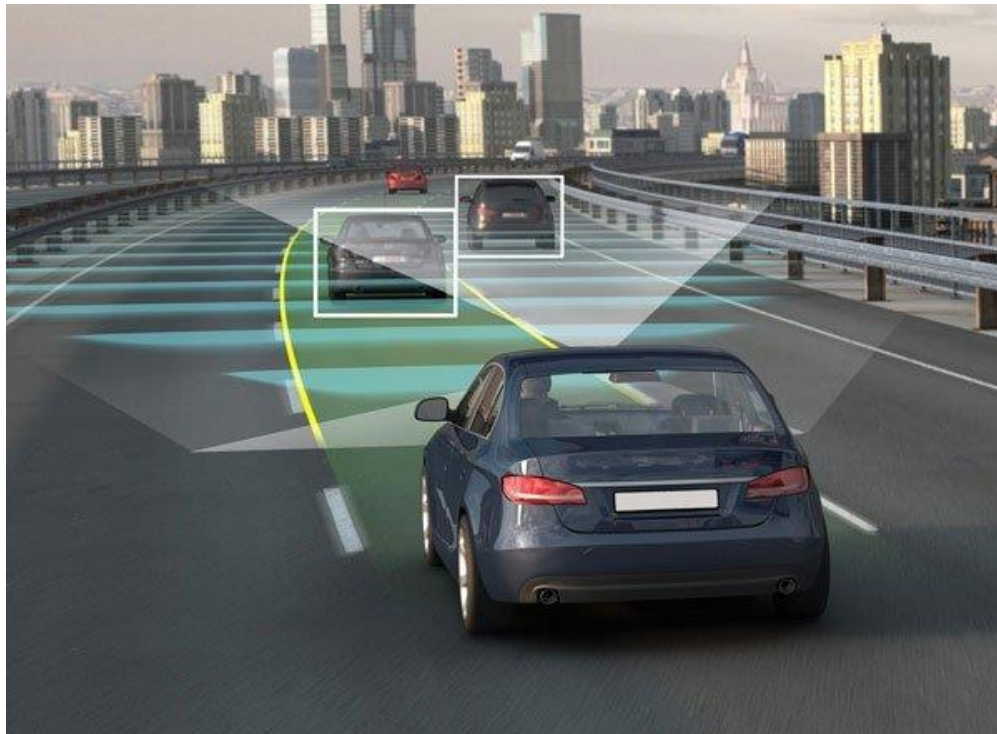


Рисунок 1.1 Контроль по смузі руху

Сучасні системи адаптивного круїз-контролю (АСС) зазвичай обладнані радарним датчиком, що працює в діапазоні частот 76-77 ГГц (Промені радара, випромінювані радарним датчиком, відбиваються від автомобілів, що йдуть попереду; аналізується їх час повернення та співвідношення амплітуд. Ці величини використовуються для розрахунку відстані, швидкості і кутового положення автомобілів, що йдуть попереду. Електронні компоненти для аналізу та управління (блок перевірки радара) змонтовані у корпусі датчика. Вони обмінюються даними через шину даних з іншими електронними блоками, що відстежують швидкість двигуна та дію гальм.

Є також системи адаптивного круїз-контролю, що використовують лазерні промені інфрачервоного діапазону (Lidar). Принцип роботи той самий, але при цьому слід враховувати обмеження функціональності оптичного зондування в погану погоду (туман, дощ, сніг) порівняно з радарним зондуванням (рис. 1.2).

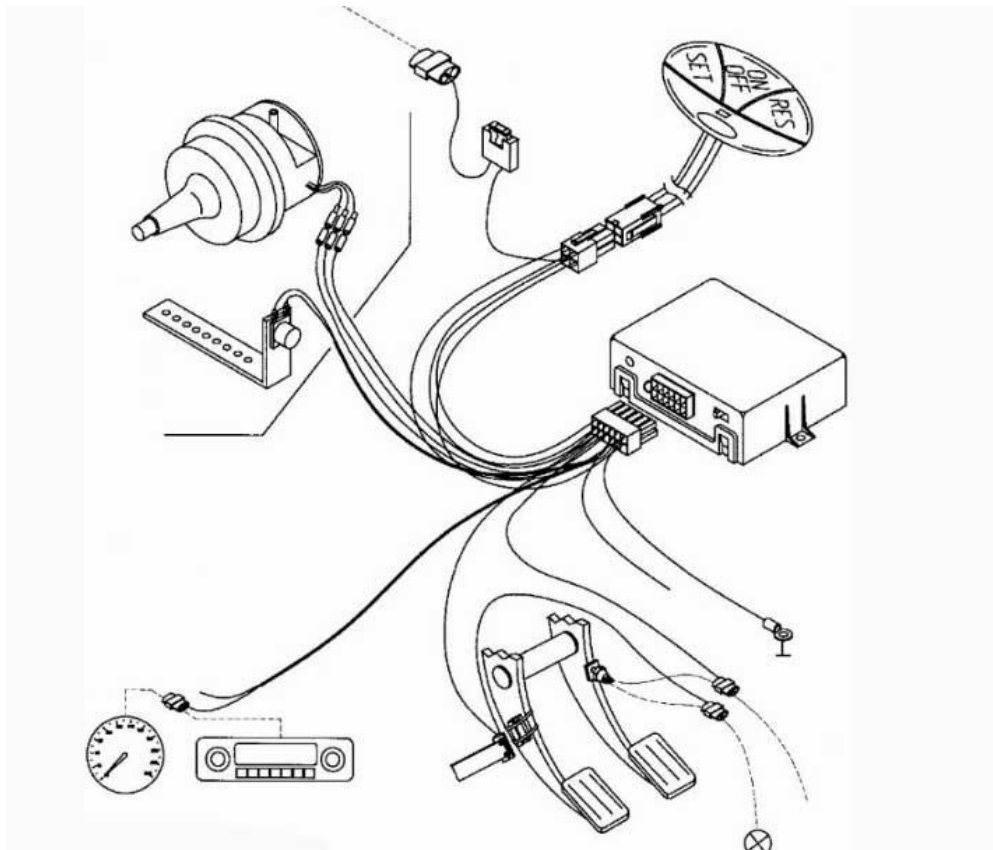


Рисунок 1.2 – Система круїз-контролю

Логічно витікає, що круїз-контроль вже може не просто підтримувати встановлену швидкість, а й адаптуватися до умов руху. Система тісно взаємодіє з радаром та датчиками, що оцінюють дорожню ситуацію. Радари можуть бути лазерного типу, електромагнітного, оптичного. Важливо не це. Важливо те, що система може стати вже найближчими роками одним із видів безпілота.

Схема роботи старих та сучасних простих систем круїз контролю дуже проста:

1. Сервопривід дросельної заслінки або педалі газу підключений до датчика швидкості та обертів колінвалу.
2. Встановлюємо потрібну швидкість, а система узгоджує обороти двигуна з кутом нахилу дроселя (або педалі).
3. Далі швидкість перебуває під контролем системи та утримується на постійному рівні при падінні оборотів двигуна (скажімо, на підйомі) система автоматично збільшить подачу палива у карбюратор або паливну рампу.

4. Як тільки швидкість перевищить заданий ліміт, круїз-контроль зменшить подачу палива та узгодить обороти двигуна зі швидкістю.

5. Варто торкнутися педалі газу, гальма, важеля ручника або спеціальної кнопки, система відключиться і перейде до штатного режиму керування подачею палива.

Прості системи вже навчилися розпізнавати автомобіль і реагувати на нього, аж до того, що наша машина може повністю повторювати швидкісний режим автомобіля, що йде попереду. Більше того, пристрій здатний точно витримувати дистанцію до провідної машини. Це корисно в багатьох випадках, але найефективніше застосування системи з метою безпеки – оцінюючи відстань до передньої машини, її швидкість, система запобігає зіткненням, дає водію розслабитися і відпочити якийсь час, поки спереду їде адекватний водій, використовувати його як шаблон.

Якщо ж попереду нікого немає, тоді водієві доводиться самому реагувати на швидкісні обмеження, раптові перешкоди, дорожню розмітку, але датчики все це запросто можуть відстежувати.

### 1.2.2 Система екстреного гальмування

Одним із ключових пристроїв, що запобігають аваріям або зменшують їх наслідки, є система екстреного гальмування (рис. 1.3). Вона необхідна для ефективної роботи гальмівної системи у критичній ситуації: у середньому гальмівний шлях автомобіля скорочується на двадцять відсотків. Дослівно BAS або Brake assistant можна перекласти як помічник при гальмуванні. Допоміжна система екстреного гальмування (залежно від виду) або допомагає водію при екстреному гальмуванні (через "дотискання" педалі гальма), або автоматично пригальмовує автомобіль без участі водія до повної зупинки.

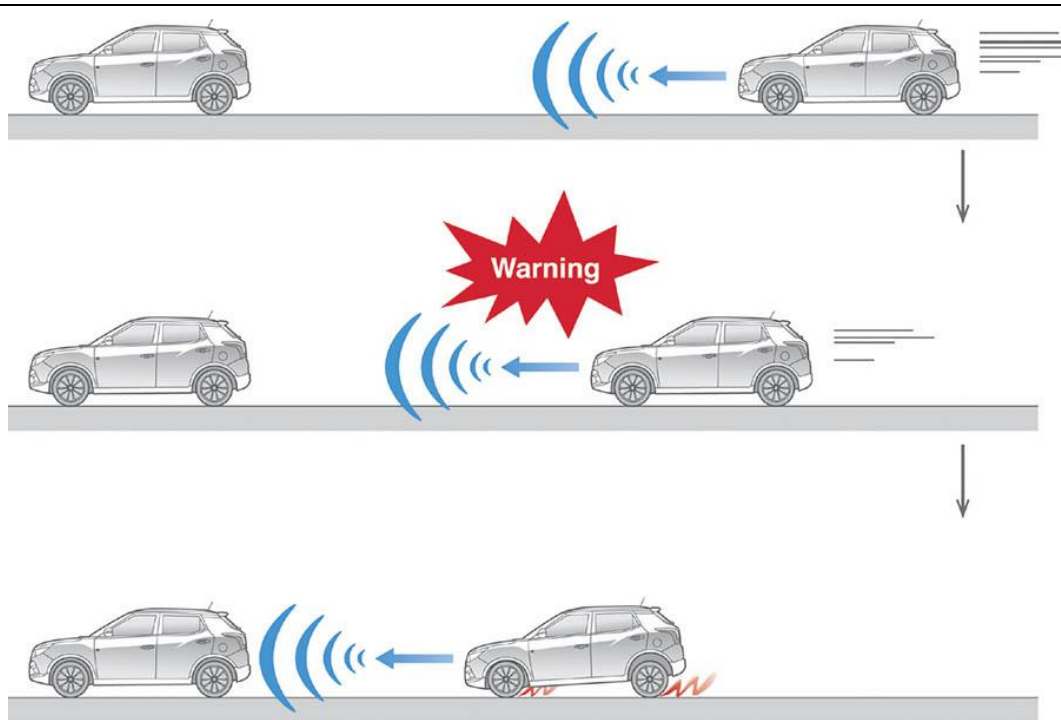


Рисунок 1.3 – Система екстреного гальмування

Існують дві групи допоміжних систем екстреного гальмування:

- допомогу при екстреному гальмуванні;
- автоматичне екстрене гальмування.

Перша створює максимальний гальмівний тиск, що утворюється в результаті натискання водієм на педаль гальма. По суті, вона "гальмує" за водія. Друга – виконує ту ж саму функцію, але вже без участі водія. Цей процес відбувається автоматично. Виходячи з принципу створення максимального гальмівного тиску, система цього типу ділиться на пневматичну та гідравлічну.

#### Пневматична система аварійного гальмування

Наслідуючи назву, пневматична система створює додатковий ресурс для роботи гальмівної системи за допомогою вакуумного підсилювача гальм (рис. 1.4). Найпоширенішими є:

- BA або Break Assist, BAS або Break Assist System, EBA або Emergency Brake Assist – встановлюються на автомобілі Audi, Volkswagen, Skoda,

Toyota, BMW, Volvo, Mercedes-Benz та багато інших європейських автомобілів.

- AFU, встановлюється на французьких автомобілях, тобто на Citroen, Peugeot та Renault.

Вона складається з датчика положення штока вакуумного підсилювача, електронного блоку керування та приводу штока. Така система відноситься до механічних типів систем, оскільки працює за рахунок визначення зусилля, що передається на гальмівну систему.

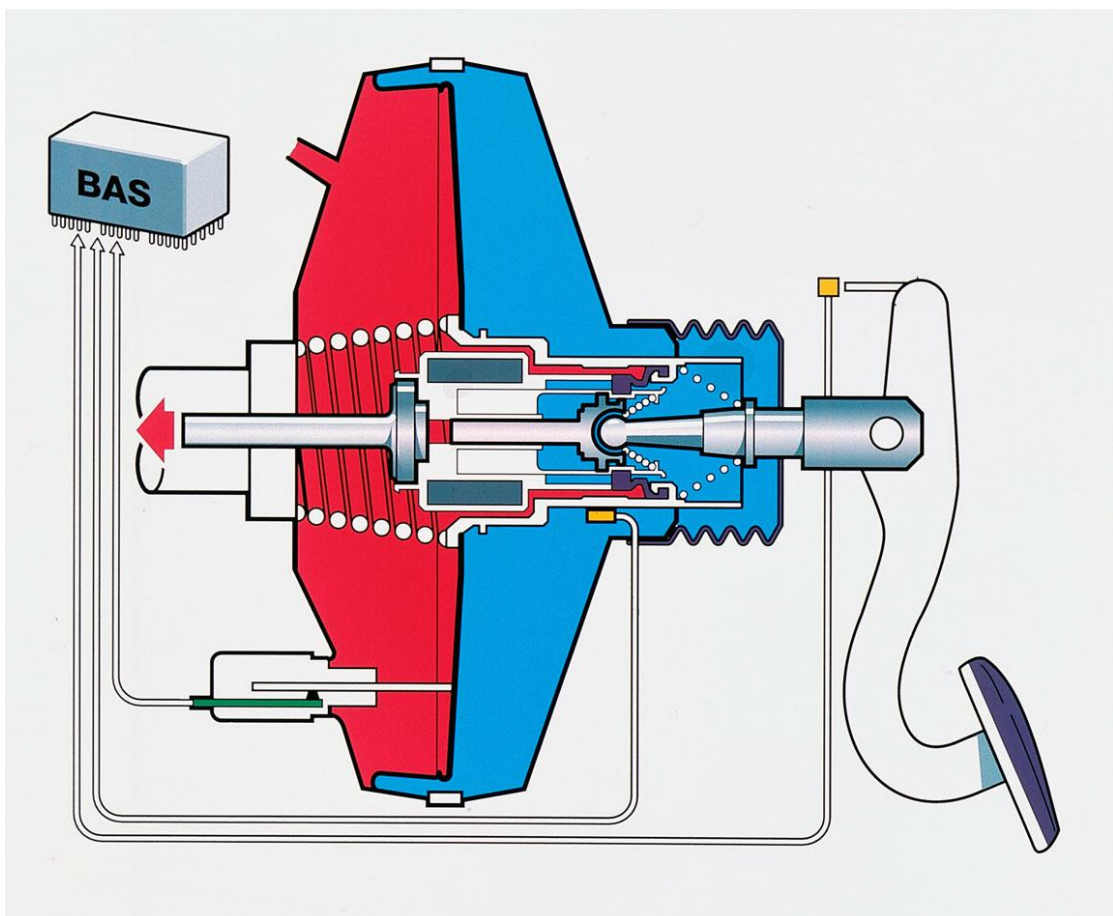


Рисунок 1.4 – Пневматична система аварійного гальмування

#### Гідравлічна система екстреного гальмування.

Щодо гідравлічних систем, то максимальне зусилля на гальмівну систему переноситься за рахунок компресора, який створює більший тиск гальмівної рідини в системі. До такого типу систем належать:

- HBA (Hydraulic BA) та HBB (Hydraulic Brake Booster) – встановлюється на автомобілі марок Volkswagen та Audi.
- SBC (Sensotronic Brake Control) та BA Plus (Brake Assist Plus) – на автомобілі Mercedes-Benz.
- DBC (Dynamic Brake Control) – автомобілі марки BMW.

HBA та HBA працюють схоже – враховується сила натискання педалі гальма та можливість включення додаткового насоса. SBC враховує дуже багато факторів, включаючи швидкість перенесення ноги з педалі на гальмо та час досягнення подачі сигналу до граничного.

#### Електронна система аварійного гальмування

Ця система Brake Assist ставиться на 99% технічно сучасних автомобілів. Представляє собою купу сенсорів, які роблять моніторинг швидкості автомобіля, попереду автомобіля і зусилля натискання педалі гальма, після чого створюється передналаштування початкового і прогресивного зусилля роботи гальмівної системи. Якщо система визначає, що гальмування екстрене – система отримує енергію для роботи від гальмівного поршня, а не підсилювача гальм, як це відбувається у механічній системі.

#### Механічна система аварійного гальмування

Механічна система є найпершою системою аварійного гальмування, тому досить проста, але також дуже ефективна. Вона зчитує два основні параметри – швидкість автомобіля та силу, з якою водій тисне на педаль гальма. Система здійснює обчислення і якщо змінна виходить за певний поріг – вона вмикається та активує гальмівний підсилювач.

#### Система автоматичного екстреного гальмування

У дорогих комплектаціях автомобілі часто комплектуються автоматичними системами екстреного гальмування. Особливість (і відмінність) такої системи від попередніх полягає в тому, що вона спрацьовує незалежно від того, чи хоче водій цього чи ні.



До таких систем належать: Pre-Safe Brake (M-Benz), CMBS (Honda), FCM (Mitsubishi), Active City Stop, вона ж Forward Alert (Ford), City Safety (Volvo), PEBS (Bosch), АЕВ (TRW ).

### 1.2.3 Паркувальні асистенти

Припаркуватися за умов великого міста буває непросто навіть професійним водіям. Запаркувати автомобіль без пошкоджень в умовах обмеженого простору допоможе паркувальний радар (він же парктронік).

Паркувальний радар, також відомий як, Акустична Паркувальна Система (АПС), парктронік, або ультразвуковий датчик паркування – допоміжна паркувальна система, що встановлюється на автомобілях. Одна частина датчика випускає ультразвукової сигнал, друга частина - вловлює відбитий від предмета сигнал і різницю в часі передає на головний прилад. За різницею між випущеними і прийнятим сигналом і визначається відстань до об'єкта (рис. 1.5).

Він знадобиться, як водіям-початківцям, адже найчастіше у них виникають проблеми з паркуванням, так і досвідченим автолюбителям. Радар паркування дозволить "вписатися" навіть у невеликий простір між машинами.



Рис. 1.5 – Паркувальна система

Будь-яка паркувальна система складається з кількох складових:

- ультразвукові датчики;
- блок управління;
- системи виведення даних (це може бути екран або маленькі динаміки, вбудовані в блок управління).

Кількість датчиків може змінюватись в залежності від моделі від двох до восьми та більше. Що більше їх, то краще. Справа в тому, що при роботі всього двох елементів неминуче виникають «мертві зони» між ними та пристрій може не зреагувати на стовп або дерево. Деякі моделі із двома датчиками дозволяють підключити додаткові.

### 1.3 Принцип роботи паркувальних систем

Принцип може бути врізним або накладним, з провідним або бездротовим способом підключення. Бездротове з'єднання дозволяє не тягнути дроти через весь салон, однак і коштують такі пристрої набагато дорожче. Моделі з дисплеєм у комплекті, значно зручніші у використанні. Екран дозволяє контролювати наближення перешкоди (рис. 1.6).

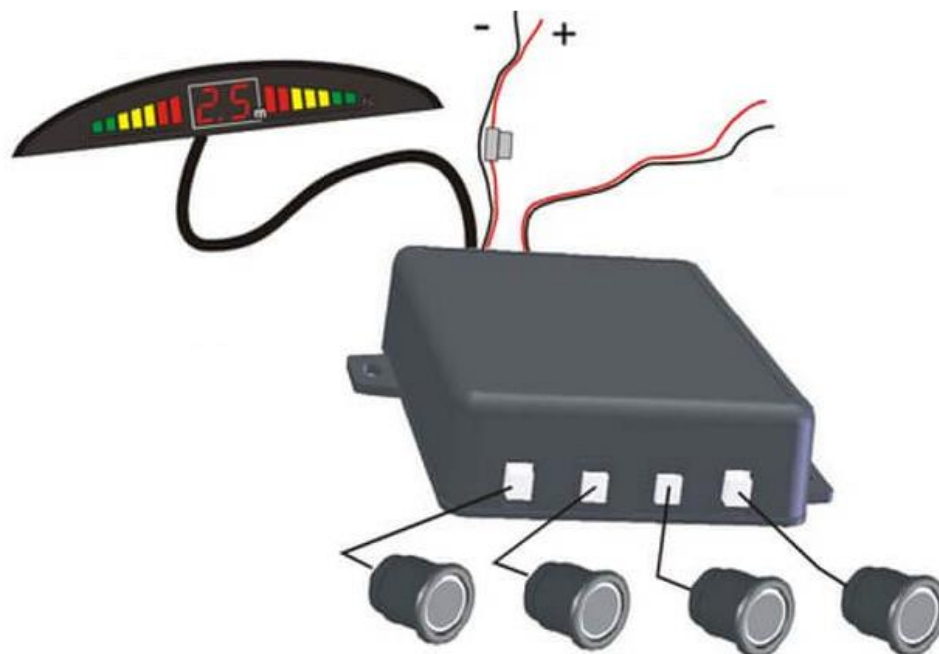


Рис. 1.6 – Паркувальна система

Принцип роботи датчика парктроніка досить простий. Закріплені елементи надсилають ультразвуковий імпульс. Він відбивається від

найближчої перешкоди та повертається назад. Блок управління визначає час за який повернувся відбитий сигнал і, виходячи з цього, розраховує і видає інформацію про відстань до найближчого об'єкта. Дані відображаються на дисплеї гаджета або у вигляді звукового сигналу залежно від конкретної моделі (рис. 1.7).

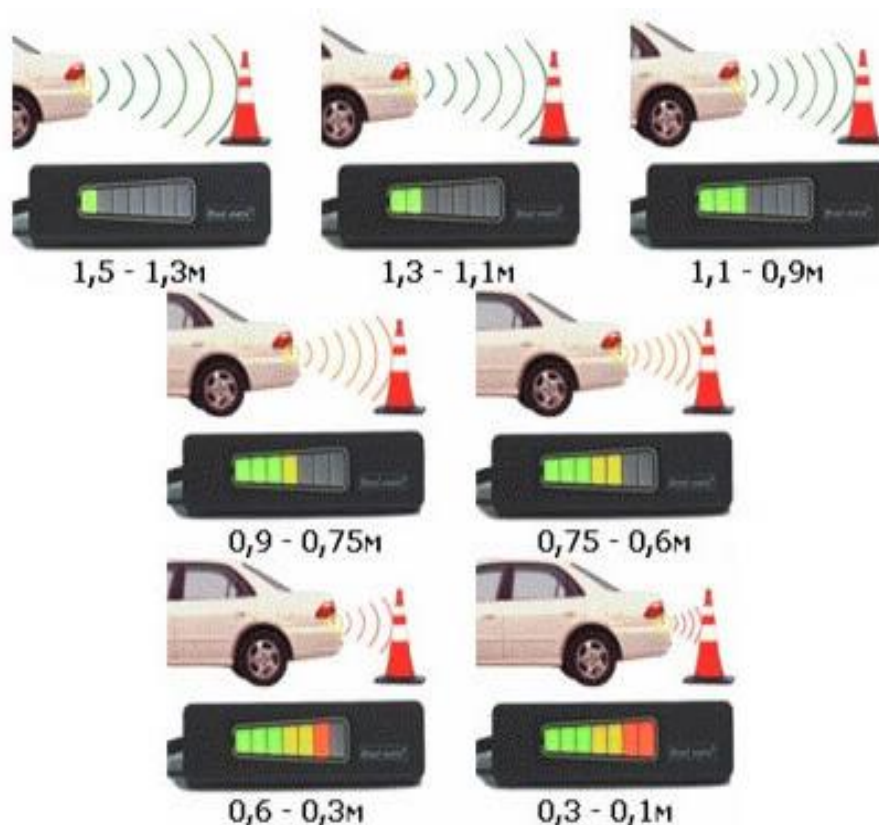


Рис. 1.7 – Принцип роботи паркувальної системи

Ще одна функція блоку управління – тестування та самодіагностика роботи системи. У разі поломки він подає звуковий сигнал, що говорить про те, що пристрій вийшов з ладу.

Залежно від місця розміщення ультразвукових елементів радари паркування можуть бути передніми та задніми. Як працює передній парктронік? Передній радар паркування має на увазі розташування датчиків на передньому і задньому бампері, найчастіше в комплекті йде шість і більш чутливих елементів, чотири з них встановлюються ззаду, два, що залишилися спереду. Принцип його функціонування повністю ідентичний роботі системи

із заднім розташуванням. У деяких пристроях є можливість перемикання між прийомом сигналів лише передніх або задніх чутливих елементів.

Використовувати паркувальний радар досить просто. Важливо, щоб датчики були правильно встановлені, точно у вертикальній площині, в іншому випадку пристрій може сприймати асфальт як перешкоду і сигналізувати про нього. У випадку, якщо радар паркування не оснащений екраном, він сигналізуватиме про наближення перешкоди за допомогою звуку. Чим воно ближче, тим гучнішим буде звук біпера. Якщо відстань досягла критичної позначки (тобто 25-30 см), звук стає безперервним.

Що показує дисплей парктроніки? Це залежить від типу екрана радара. Одними з найпоширеніших і недорогих моделей є пристрої, оснащені світлодіодним дисплеєм, що змінює колір. Якщо екран спалахує зеленим світлом, це сигналізує про наявність перешкоди, у міру наближення до нього, він стає червоним. Додатково можуть відобразитися цифри, що вказують на відстань до об'єкта в метрах. Радари, оснащені екраном, коштують дорожче. Додатково вони можуть бути забезпечені камерою заднього виду.

#### **1.4 Огляд сучасних систем збору інформації**

В Україні паркувальні радари вперше стали відомими під торговою маркою Парктронік (англ. Parktronic), так називається паркувальна система на автомобілях Mercedes-Benz. У зв'язку з цим у розмовній мові словом «парктронік» стали позначати паркувальні радари будь-яких виробників. Інші марки використовують інші назви: BMW і Audi на німецькому називають систему просто «допомогою при парковці» – Parkassistent. Audi також використовує скорочення APS, яке розшифровується як Audi Parkassistentensysteme німецькою або Audi parking system англійською.

#### **Паркувальний радар (парктронік) AMS A8191-0**

Продукція бренду AMS має стабільний попит на ринку автомобільної

електроніки та додаткових аксесуарів. Однією із затребуваних позицій у товарній лінійці виробника є функціональні системи паркування, сумісні з більшістю сучасних авто. Вся продукція AMS проходить всебічний контроль якості на виробництві, за рахунок чого досягається відповідність її високим світовим стандартам (рис. 1.8).



Рис. 1.8 – Паркувальний радар (парктронік) AMS A8191-0

Парктронік AMS A8191-0 Black – це допоміжна система, яка використовується під час паркувального маневрування, руху в умовах обмеженого простору та слабкого освітлення. Паркувальний радар повідомляє водія про зближення з іншими об'єктами за допомогою візуальних та звукових сповіщень.

Комплект поставки AMS A8191-0 Black складається з восьми ультразвукових датчиків чорного кольору (діаметр 18.5 мм), інформативного LED дисплея, біпера для подачі звукових сигналів та блоку керування. Датчики з фаскою встановлюються у передній та задній бампер автомобіля. Для свердління отворів необхідно скористатися мініатюрною фрезою, що входить до комплектації системи. Поверхню сенсорів слід очищати від забруднень, щоб їхня чутливість не знизилася.

На світлодіодний дисплей парктроніка виводиться інформація про відстань від автомобіля до найближчої перешкоди. Водій бачить цифрове

значення в метрах та кількість поділів шкали (жовтого та червоного кольору), що вказує на небезпеку зіткнення з перешкодою. Датчик розпізнають сторонні об'єкти на відстані від 30 см до 2 метрів, маючи показник точності вимірювань 10 см. У системі даної моделі передбачено функцію запам'ятовування виносних елементів автомобіля та можливість регулювання чутливості датчиків. Паркувальний радар не конфліктує з бортовою електронікою і має надійний захист від перешкод.

Також існує більш дорожчий варіант, котрий має додаткові функції.

#### Парктронік Phisung H50P з відеореєстратором

Дзеркало з відеореєстратором з камерою заднього виду і з парктроником на 4 датчики. Основна камера Full 1920x1080p, кут огляду 170 градусів, унікальна нічна зйомка, HD камера заднього виду з розміткою паркування, одночасний запис відео на передню і задню камеру, підтримка карт пам'яті до 32 Гб 10-го класу (рис. 1.9).

Конструкція складається з дзеркала заднього виду з вбудованим відеореєстратором і з камерою заднього виду. Пристрій має IPS дисплей, роздільна здатність 1600×400 із дзеркальним покриттям. Дисплей відображає відео з камер, лінію розмітки та індикацію датчиків парктроніка.



Рис. 1.9 Парктронік Phisung H50P з відеореєстратором

Дзеркало Phisung H50P легко встановлюється та непомітний у салоні автомобіля. З його допомогою можна вчасно фіксувати будь-яку ситуацію на дорозі та переглядати відео в поточному режимі.



Рис. 1.10 – Паркувальна камера

Проаналізувавши товари, було створено таблицю порівняних характеристик (табл. 1.2):

Таблиця 1.2 – Порівняльні характеристики

<b>Характеристики</b>	<b>AMS A8191-0</b>	<b>Phisung H50P</b>
Місце встановлення	Передній і задній бампери	Задній бампер
Тип датчика	Сонар	Сонар + камера
Діаметр датчика	18.5 мм	20.5 мм
Кількість датчиків	8	4
Тип виводу	LED дисплей	Екран
Захист від ел. магн. завад	+	–
Кут огляду	110°	140°
Джерело живлення	Акумулятор	Від прикурювача
Робоча температура	-10 ~ +50С	-30 ~ +70С
Комплектація	Дзеркало-реєстратор – 1 шт. Зарядний пристрій – 1 шт. Блок парктроніка – 1 шт. Датчики парктроніки - 8 шт.	Дзеркало-реєстратор – 1 шт. Камера заднього виду – 1 шт. Блок парктроніка – 1 шт. Датчики парктроніки - 4 шт.

---

## 1.5 Постановка задачі

Багатий набір електроніки – це до чверті ціни автомобіля. Наскільки корисні та необхідні помічники водія? У Європі цим питанням здивувалися кілька років тому, коли вийшла директива ЄС про розширення списку обов'язкового обладнання. Тепер, через шалені темпи подорожчання автомобілів, тема стала гострою і в нас.

Найкориснішими водії вважають контроль парктронік та круїз-контроль із функцією руху в пробці. Хоча до них є претензії. Серед найбільш затребуваних також системи кругового огляду та автоматичного гальмування.

Електроніка "старіє" і з роками втрачає колишню точність. Природний знос, пошкодження при ДТП, заміна вітрового скла призводять до порушення калібрування камер та помилок. На машинах віком 5–10 років система поводить не так, як на нових, що потребує її заміни.

Головною проблемою систем є їх автономність та габарити, тобто такі системи як круїз контроль та екстрене гальмування вбудовані в авто і їх неможливо докупити чи самостійно налаштувати.

Аналіз систем збору критичних параметрів нашоє до того, що необхідно вирішити такі проблеми:

- ціна;
- розміри;
- автономність.

## Висновки до розділу 1

В результаті аналізу існуючих систем парктроніку, що представлені на ринку, було вирішено створити систему, яка би за ціною, розмірами та автономністю, вигідно відрізнялась би від існуючих аналогів.

Система повинна складатись з бюджетних елементів та мати високу якість, мати невеликі розміри (для використання у малолітражних автомобілях) та володіти достатньою автономністю.



---

## РОЗДІЛ 2

### АНАЛІЗ ТА МЕТОДИ РОБОТИ АПАРАТНО-ПРОГРАМНОГО КОМПЛЕКСУ

#### 2.1 Аналіз критичних параметрів

Кожен професійний та компетентний водій повинен вміти застосувати навички гальмування та маневрування у поганих погодних умовах, на слизькому дорожньому покритті та в екстрених дорожніх ситуаціях.

За правилами дорожнього руху водій повинен забезпечити відповідно до дорожньої обстановки, що склалася, необхідне для безпеки зниження швидкості автомобіля, аж до повної його зупинки. Тому знання ефективних прийомів гальмування та вміння їх застосовувати – це запорука впевненості водія у забезпеченні безпеки в екстрених ситуаціях на дорогах.

Гальмівним шляхом вважається відстань, яку машина проїде після активації гальмівної системи до повної зупинки. Це лише технічний параметр, за яким у поєднанні з іншими факторами визначається безпека автомобіля. Цей параметр не включає час реакції водія. Сукупність реакції автомобіліста на екстрену ситуацію та відстані від початку гальмування (водій натиснув на педаль) до повної зупинки транспортного засобу називається зупинний шлях.

Тобто проаналізувавши дану інформацію можна вивести наступну закономірність:

$$L = T + S, \quad (2.1)$$

де:

$L$  – довжина зупинного шляху;

$T$  – час реакції водія;

$S$  – гальмовий шлях.

Зупинний шлях – це відстань, пройдена транспортним засобом з моменту виявлення небезпечним водієм на дорозі до повної зупинки машини. Зупинний шлях включає відстань, пройдену автомобілем за час реакції водія, та час спрацювання гальмівної системи. Враховуючи умови видимості у

напрямку руху, водій повинен вибрати таку швидкість, щоб зупинний шлях автомобіля не перевищував відстань видимості. В іншому випадку швидкість необхідно знизити. На довжину зупинної колії впливає реакція водія, стан транспортного засобу та дорожнього покриття.

Зупинний шлях автомобіля обчислюється за формулою:

$$S = (t1 + t2)V + \frac{kV^2}{2g}, \quad (2.2)$$

де:

$t1$  - час реакції водія;

$t2$  - час спрацьовування гальм;

$k$  - коефіцієнт ефективності гальмування;

$V$  - швидкість руху автомобіля;

$g$  - прискорення вільного падіння.

Час реакції водія – час з моменту виявлення ним небезпеки до здійснення необхідні дії, наприклад, перенесення ноги на педаль гальма, натискання на неї. Залежить від особливостей водія, а також від досвіду водіння, положення його тіла, рук та ніг щодо органів керування автомобілем. На час реакції впливає психоемоційний стан водія. Час реакції збільшується при втомі, захворюваннях і дуже зростає при алкогольному або наркотичне сп'яніння.

Гальмівний шлях автомобіля – це дистанція до повної зупинки, яку встигає подолати машина з того моменту, як водій натиснув на педаль гальма. Важливо розуміти, що зупинний шлях завжди більший, ніж гальмівний. Адже він включає ще й відстань, пройденою з того моменту, як водій виявив небезпеку та натиснув на гальма. Протяжність гальмівного шляху залежить від швидкості, стану проїжджої частини, шин, погодних умов.

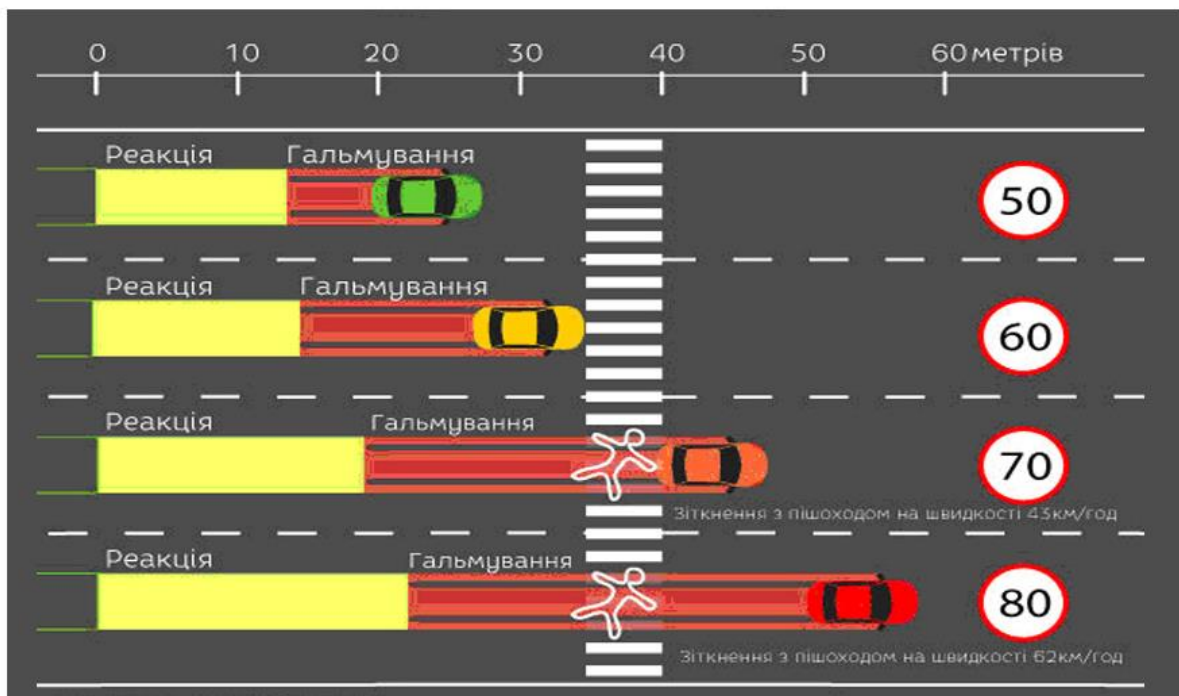


Рисунок 2.1 – Приклад зупинного шляху

Немає такої речі у Всесвіті чи матерії, яка може миттєво зупинитися. Також і будь-який автомобіль, коли водій натискає гальмо педаль, він не відразу може зупинитися. Справа в тому, що для того, щоб автомобіль або будь-який об'єкт у нашому світі зупинився, необхідно, щоб він втратив енергію, яка його рухає. В результаті будь-якого автомобіля є гальмівний шлях, який він проїжджає з моменту натискання педалі гальма до моменту повної зупинки.

Але насправді гальмівний шлях будь-якого авто залежить не лише від його характеристик та гальмівної системи, а й від реакції водія при натисканні педалі гальма. Адже для того, щоб прийняти рішення про необхідність гальмування і натиснути педаль гальма, потрібен час, який хоч і мінімальний, але достатньо, щоб машина встигла проїхати немаленький шлях. Особливо це важливо за великої швидкості руху, де за якісь частки секунди автомобіль проїжджає пристойну відстань. Отже, в результаті, щоб розрахувати реальну довжину гальмівної колії, потрібно враховувати не тільки час і відстань,

пройдений автомобілем з моменту натискання водієм педалі гальма до моменту зупинки машини, а й час, необхідний для ухвалення рішення про гальмування. Справа в тому, що при ухваленні рішення про гальмування ми витрачаємо дорогоцінні секунди. Ось наприклад:

Час відгуку: Перш ніж водій натисне педаль гальма, він повинен оцінити дорожню ситуацію і визначити, чи гальмування необхідно. Також потрібно зрозуміти, яке необхідне гальмування – повна зупинка автомобіля чи просте зниження швидкості. Зазвичай, згідно з численними дослідженнями, для більшості водіїв для цього потрібно близько 0,1 секунди.

Час, необхідний для натискання педалі гальма: Після того, як водій зрозумів, що повинен гальмувати, необхідно ще приблизно 0,8 секунди, щоб перемістити ногу з педалі газу на педаль гальма і натиснути її. В таблиці 2.1 представлено залежності реакції водія та пройденого автомобілем шляху:

Таблиця 2.1 – Залежність довжини зупинного шляху

Швидкість автомобіля	Шлях до початку гальмування	Гальмівний шлях	Зупинний шлях
20 км/год	6м	6м	12м
40 км/год	12м	24м	36м
60 км/год	18м	55м	73м
80 км/год	24м	76м	100м
100 км/год	38м	105м	143м

## 2.2 Фактори що впливають на збір критичних параметрів

Вирішальним значенням для довжини гальмівного шляху, звичайно, є швидкість автомобіля, з якою він рухається дорогою. Також на гальмівну колію впливає якість встановленої на машину гальмівної системи. У тому числі важливу роль, безсумнівно, відіграє стан дороги (сніг, лід, якість асфальту/бетону, тріщини в дорожньому покритті, листя, калюжі тощо). І

звісно ж, не варто забувати про стан шин автомобіля. Адже в певних випадках зношена гума сильно збільшить гальмівну колію автомобіля, тому що не зможе передавати нормальну гальмівну здатність дорожньому покриттю на відміну від нових шин, що мають нормальне зчеплення з дорогою. Також ясно, що на мокрій поверхні гальмівна відстань машини більша, ніж на сухому асфальті. Отже на довжину гальмового шляху впливають такі фактори:

- справність гальмівної системи;
- стан шин;
- виробник, марка та вид шин;
- маси автомобіля;
- наявність ABS та інших сучасних допоміжних систем;
- швидкість руху.

Як вже було вище сказано, гальмівний шлях автомобілів один від одного може відрізнятися в залежності від різних факторів. Проте не слід забувати і про рівень підготовки водія. Особливо важлива для підсумкового гальмівного шляху швидкість реакції водія на дорожню ситуацію, що вимагає зупинки автомобіля. Але швидкість реакції за кермом залежить не тільки від досвіду водіння. Наприклад, чи знаєте ви, що коли ви сідаєте за кермо в сонному стані (не виспалися, втомилися або довго перебували за кермом), швидкість реакції може сповільнитися майже вдвічі в порівнянні зі швидкістю реакції водія, що добре відпочив.

В цілому ж на швидкість прийняття рішення за кермом (час реакції) впливає багато факторів, які наведено нижче:

- погодні умови;
- умови видимості;
- вік та фізичні данні;
- досвід водія;
- стан водія.

Висока швидкість автомобіля призводить до збільшеного гальмівного шляху. Але крім самої швидкості очевидно, важливу роль грає і швидкість вашої реакції перед тим, як ви натиснете на педаль гальма. У середньому у більшості водіїв швидкість реакції на екстрену ситуацію на дорозі при ухваленні рішення про гальмування приблизно однакова. Як бачимо з таблиці, що вища швидкість, то довшим буде пройдений шлях до того самого моменту, доки ми не натиснемо на педаль гальма. І це з огляду на те, що швидкість нашої з вами реакції не змінюється.

Також необхідно знати і про те, що на мокрій або зледенілій дорозі гальмівний шлях звичайно збільшується. Справа в тому, що на слизкій поверхні зчеплення автомобіля з дорогою сильно знижується, що і призводить у разі гальмування до збільшення гальмівного шляху.

Управління автомобілем у стані алкогольного чи наркотичного сп'яніння насправді також впливає на підсумковий гальмівний шлях машини. Справа в тому, що алкоголь чи наркотики знижують та уповільнюють швидкість реакції водія у кілька разів. Відповідно всі рухи за кермом здійснюватимуться повільніше. Хоча це може бути і не помітним збоку, але насправді реакція п'яного водія при гальмуванні (і не лише) справді буде знижена. А на дорозі, як ми з вами знаємо, все вирішують зазвичай якісь мілісекунди. Саме тому водій у п'яному стані за кермом дуже небезпечний. Адже у разі екстреного гальмування водій у стані алкогольного чи наркотичного сп'яніння за потреби натисне на педаль гальма із запізнілою реакцією. У результаті автомобіль, перш ніж почне зупинятися, проїде додатково ще якусь пристойну відстань, що зрештою додасть до гальмівного шляху ще зайвих метрів.

Шини авто також є важливим фактором, що впливає на підсумковий гальмівний шлях автомобіля. По-перше, така довжина гальмівного шляху залежить від ступеня зношування протектора гуми, а також від марки покришок. Звичайно, чим дорожче марка і модель гуми, тим більше і

надійніше вона забезпечує авто зчеплення з дорогою. Крім того, на мокрій дорозі гальмівна колія залежить ще й від стану протектора гуми та від самої її специфікації. Якщо гума достатньо зношена, то глибина протектора шини буде невеликою, що призведе до недостатнього відведення води з покришок. Внаслідок чого зменшиться зчеплення з дорогою, а це в свою чергу вплине на гальмівний шлях при гальмуванні машини.

Неможливо здійснити точні розрахунки на абстрактному автомобілі. Часто гальмівний шлях розраховується з того, якою була ця величина у конкретної машини на певній швидкості. Розрахунки гальмівного шляху в залежності від умов і факторів знадобляться мені в подальшій роботі, тому використавши аналітичні данні Euro NCAP, мною була сформована наступна таблиця с із середньостатистичними даними (табл. 2.2).

Таблиця 2.2 – Середні значення згідно з даними Euro NCAP

Дорожнє покриття	Швидкість руху автомобіля, км/год					
	20	40	60	60	80	100
	Гальмівний шлях автомобіля					
Грунтова дорога	6м	15м	32м	43м	50м	62м
Мокрий асфальт	9м	22м	44м	58м	75м	85м
Засніжена дорога	18м	29м	63м	80м	112м	159м
Зледеніла дорога	32м	53м	110м	169м	234м	278м

## 2.3 Методи збору критичних параметрів

Для роботи системи збору критичних параметрів для водія на основі круїз контролю та екстреного гальмування необхідні такі датчики як далекомір та спідометр. Системні датчики збирають параметри, котрі передається на блок обробки і згодом виводяться для подальших дій водія, що забезпечують рівномірний рух та безпеку (рис. 2.2).

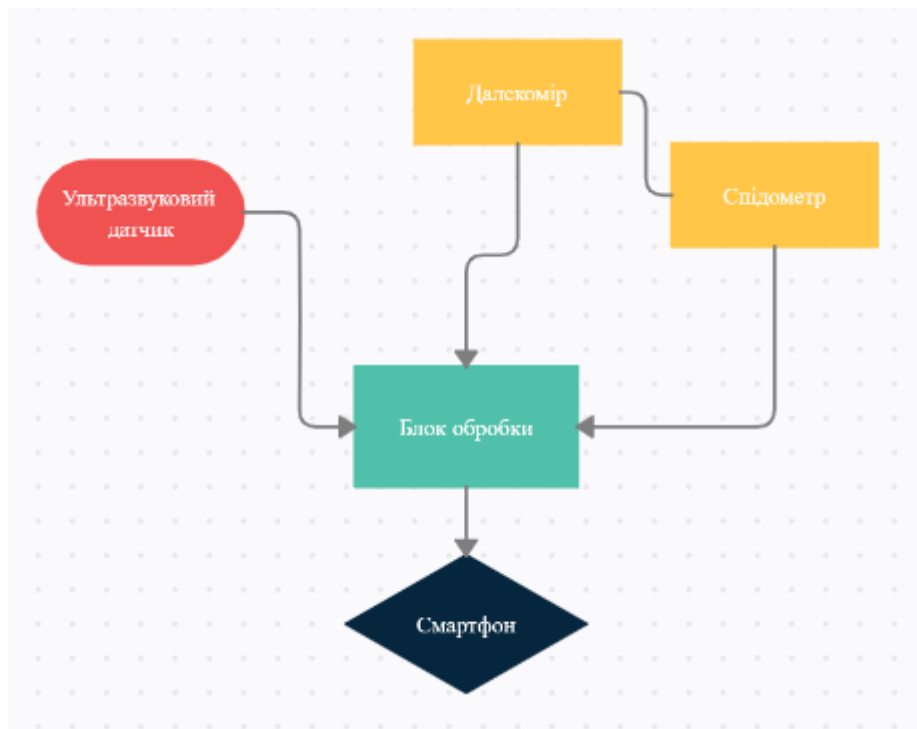


Рисунок 2.2 – Блок-схема системи збору інформації

### 2.3.1 Далекомір оптичного діапазону

Далекомір – це частний випадок лідара, у якого паралельно узький кут спостереження. Пристрій дивиться вперед в узькому сегменті і не отримує сторонніх даних, крім видалення об'єктів. Так працює оптичний Далекомір, заснований на принципі ToF. Робоча дистанція залежить від використовуваного джерела світла: для ІК-светодиодів це десятки метрів, а лазерні лідари здатні стріляти світлом на кілометри уперед (рис. 2.3).



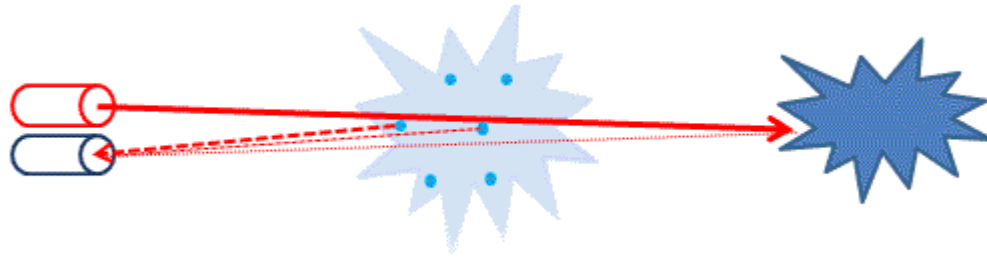


Рисунок 2.3 – Принцип роботи оптичного далекоміру

Одним із способів вимірювання часу польоту світлового променя є використання імпульсного лазера, а потім безпосередньо вимірювання часу, що минув. У таких пристроях потрібна електроніка, здатна розрізняти пікосекунди, і тому вони дуже дорогі. Іншим методом є вимірювання фазового зсуву відбитого світла.

Для вимірювання фазового зсуву використовується колімований інфрачервоний лазер. Для поверхонь, шорсткість яких перевищує довжину хвилі падаючого світла, відбудеться дифузне відображення. Компонент інфрачервоного світла буде повертатися майже паралельно променю, що проходить для об'єктів.

Датчик вимірює зсув фаз між переданим і відбитим сигналами. На малюнку показано, як цю техніку можна використовувати для вимірювання відстані. Довжина хвилі модулюючого сигналу відповідає рівнянню:

$$c = f * \tau, \quad (2.3)$$

де:

$c$  – швидкість світла;

$f$  – частота модуляції;

$\tau$  – відома довжина хвилі модуляції.

Загальна відстань  $D'$ , яку проходить світло, що випромінюється, дорівнює:

$$D = B + 2A = B + (0 * \tau)/2\pi, \quad (2.4)$$

де:

$A$  - виміряна відстань;  $B$  – відстань від одиниці вимірювання фази.

Таким чином, необхідна відстань  $D$  між світлодіодником і ціллю визначається як:

$$D = \tau * \theta / 4\pi, \quad (2.5)$$

де  $\theta$  – електронно виміряна різниця фаз між пропущеним і відбитим світловими променями.

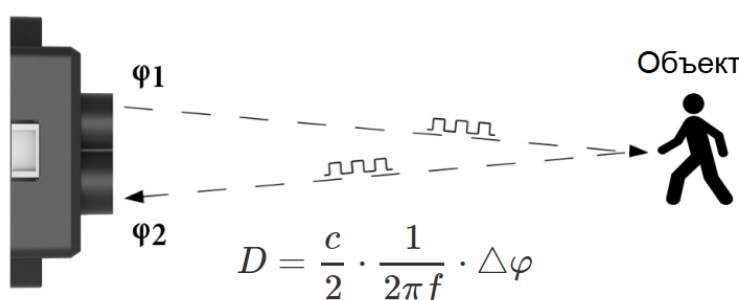


Рисунок 2.4 – Приклад роботи Lidar-датчику

Можна показати, що діапазон обернено пропорційний квадрату амплітуди прийнятого сигналу, що безпосередньо впливає на точність датчика.

### 2.3.2 Спідометр

У датчиках швидкості електронних спідометрів автомобілів використовується ефект Холла, названий на честь американського фізика Е. Холла, який відкрив це ще в 1879 році.

Якщо до провідника або напівпровідника прикладена напруга  $U_p$  і її пронизує під прямим кутом магнітне поле, що володіє індукцією  $B$ , виникає «напруга Холла»  $U_n$ , перпендикулярне напрямку струму від джерела живлення  $I_p$  і напрямку магнітного поля (рис. 2.5).

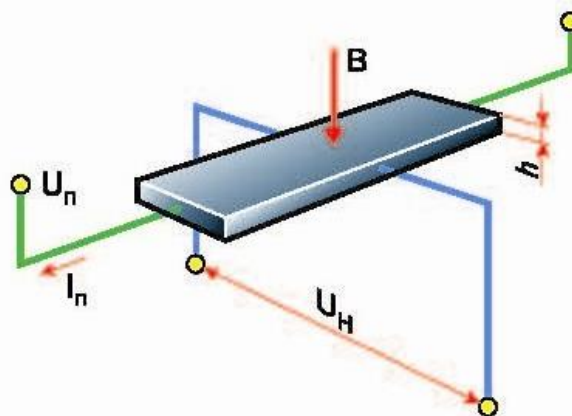


Рисунок 2.5 – Магнітне поле

$U_H = K_H I_n B/h$ , де:  $K_H$  – постійна Холла;  $I_n$  – струм від джерела живлення;  $B$  – магнітна індукція;  $h$  – товщина провідника чи напівпровідника.

З виразу слід, що величина напруги  $U_H$  пропорційна магнітній індукції  $B$ . Якщо магнітне поле  $B$  змінювати з частотою, пропорційної швидкості руху автомобіля, то частота зміни вихідної напруги  $U_H$  теж буде пропорційна швидкості автомобіля. На практиці магнітне поле створюється нерухомим магнітом, а його зміна – спеціальним обертовим екраном з прорізами (рис. 2.6).

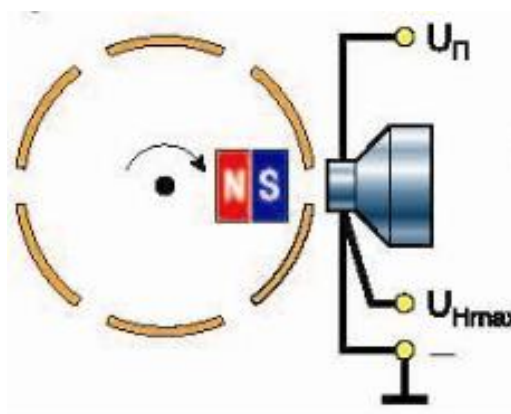


Рисунок 2.6 – Схема датчика Холла

При обертанні екрану його сегменти та прорізи по чергово проходять між магнітом та датчиком Холла. Коли між магнітом і датчиком Холла проходить сегмент екрану, магнітне поле перекривається і на виході датчика мінімальна напруга ( $U_H \min$ ). При проходженні між магнітом і датчиком Холла прорізи екрану на датчик надходить максимальний магнітний потік, і на виході напруга стає максимальною ( $U_H \max$ ).

Таким чином, при обертанні екрану зі швидкістю, пропорційною швидкості руху автомобіля, на виході датчика Холла з'являються імпульси напруги  $U_n$ , частота проходження яких пропорційна швидкості автомобіля.

### 2.3.3 Ультразвукової датчик

Сенсорні пристрої, що перетворюють електричний струм на хвилі ультразвуку, називаються ультразвукові датчики. Їх принцип дії аналогічний роботі радара, вони вловлюють мету за відображеним сигналом. Швидкість звуку – величина стала. На підставі цього таким датчиком обчислюється відстань до деякого об'єкта, що відповідає діапазону часу між виходом сигналу та його поверненням.

Датчики ультразвукового випромінювання використовуються для виявлення об'єктів, контролю їх руху та вимірювання відстаней до них. Принцип дії полягає у наступному. Прилад випромінює звукові коливання із частотою понад 20000 герц. Під час зустрічі з об'єктом вони відбиваються, потрапляють у приймач, і фіксується. Електронна схема відраховує час, який минув з моменту імпульсу до моменту прийому луни. Відстань обчислюється за такою формулою:

$$R = tV/2, \quad (2.6)$$

де:

$t$  – час між імпульсом і прийомом луни,

$V$  – швидкість звуку.

Імпульс ділиться на 2, тому що звукові хвилі проходять шлях, що дорівнює подвійній відстані між об'єктом і датчиком. Швидкість звуку у різних середовищах неоднакова:

- повітря це 331 м/сек;
- дереві –1500 м/сек;
- у питній воді – 1430 м/сек.

Відстань, де виявляються об'єкти – до 20 метрів, за умови, що вони тверда і гладка поверхня. Якщо вони виготовлені з м'якого пористого матеріалу, що поглинає звук – відстань скорочується.

Розглянемо як працює датчик:

- a) для того, щоб ініціалізувати відправлення сигналу далекоміром, необхідно подати високий сигнал тривалістю  $10\ \mu\text{s}$  на пін Trig;
- b) після отримання високого сигналу тривалістю  $10\ \mu\text{s}$  на пін Trig модуль генерує пучок з восьми сигналів частотою 40 кГц і встановлює високий рівень на пині Echo;
- c) після отримання відбитого сигналу модуль встановлює низький рівень на пині Echo.

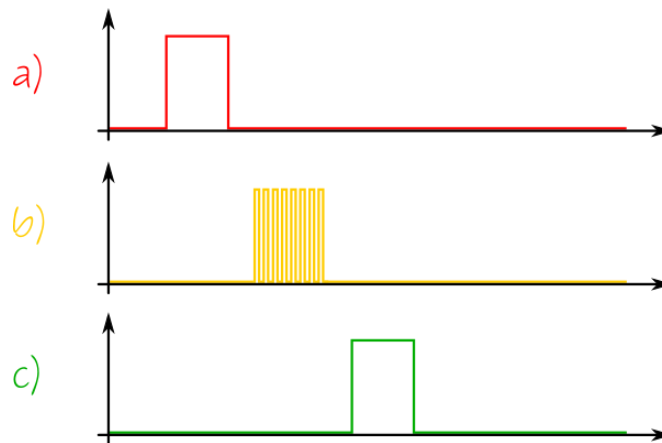


Рисунок 2.6 – Принцип проходження сигналу

Знаючи тривалість високого сигналу на пині Echo, можемо обчислити відстань, помноживши час, який витратив звуковий імпульс, перш ніж повернувся до модуля, на швидкість поширення звуку в повітрі.

## Висновки до розділу 2

У розділі було проаналізовано методи збору критичних параметрів за допомогою таких пристроїв, як: спідометр; далекомір; ультразвуковий датчик.

Також було розглянуто фактори, що впливають на критичні параметри під час керування автомобілем. Для кращих результатів роботи системи для водіїв було усереднено поведінку автомобіля та факторів впливу на дорозі, що було винесено в таблицю.

## РОЗДІЛ 3

### РЕАЛІЗАЦІЯ СИСТЕМИ ПАРКУВАННЯ

#### 3.1 Фактори що впливають на збір критичних параметрів

Перед початком розробки системи необхідно визначитися яким характеристикам мають відповідати комплектуючі. Після порівняння аналогів, обрати найбільш підходящий варіант, відповідний усім необхідним критеріям, для подальшої реалізації системи.

Для даної системи необхідні такі комплектуючі та технології: мікроконтролер, LIDAR (оптичний датчик), спідометр, сонар та смартфон.

##### 3.1.1 Мікроконтролер (Arduino Uno)

В якості програмованої платформи вирішено обрати один з мікроконтролерів родини Arduino. Arduino підходить, як початківцям так і професіоналам та являє собою електронний конструктор з зручною платформою швидкої розробки електронних пристроїв.

На рис. 3.1 зображено апаратну платформу Arduino Uno.



Рисунок 3.1 – Arduino Uno

Arduino Uno – це апаратна платформа, побудована на ATmega328. Цей контролер містить чотирнадцять цифрових входів та виходів (шість з них

можуть бути використані як виходи ШІМ), генератор кварцовий 16 МГц, шість аналогових входів, роз'єм ICSP, силовий роз'єм, USB роз'єм та кнопку рестарту. Для початку роботи з цією платформою потрібно підключити її до комп'ютера за допомогою кабелю USB. Також за допомогою батареї або адаптера AC / DC можна подати живлення.

Arduino Uno використовує мікроконтролер ATmega8U2 що відрізняється від платформ, що використовували мікроконтролер FTDI USB для контакту з USB. У табл. 3.1 наведені технічні характеристики Arduino Uno.

Таблиця 3.1 – Технічні характеристики Arduino Uno

Мікроконтролер	ATmega328
Вхідна напруга (рекомендована, гранична)	5 В, 7-12 В
Цифрові Входи та Виходи	14 (6 з яких можуть використовуватися як виходи ШІМ)
Аналогові входи	6
Постійний струм через вхід та вихід	40 мА
Постійний струм для виведення 3.3 В	50 мА
Флеш пам'ять	32 Кб (ATmega328) з яких 0.5 Кб використовуються для завантажувача
ОЗУ	2 Кб (ATmega328)
Тактова частота	16 МГц
Габарити	68 × 53 × 15 мм
Ціна	600 грн

Для цього проекту було вирішено обрати саме такий контролер, так як його характеристики найбільш повно відповідають даній розробці.

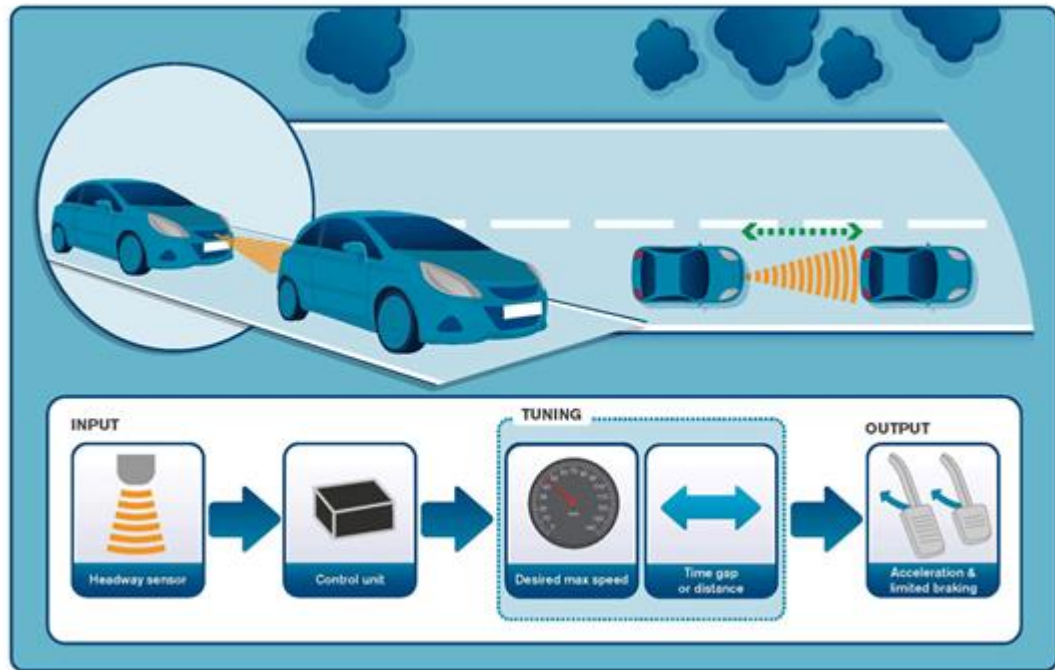


Рисунок 3.2 – Схема роботи системи

### 3.1.2 Lidar TF Luna

Лідар (LiDAR, Light Identification Detection and Ranging – світлове виявлення та визначення дальності) – технологія, яка за допомогою обробки сигналу відбитого світла отримує інформацію про об'єкти, що знаходяться на відстані. Іншими словами, пристрій випускає велику кількість лазерних імпульсів за певний відрізок часу, які виявляють об'єкти навколо себе та визначають відстань до них.

Завдяки лідару автомобіль бачить дорогу, інші авто, знаки, дерева, будинки та перешкоди. Більше того, він може визначати широкий спектр неметалічних матеріалів: скелі, дощ, хімічні сполуки, аерозолі, хмари та навіть окремі молекули.

Для проекту був обраний Lidar TF Luna (рис. 3.2)





Рисунок 3.3 – Lidar TF Luna

Це невеликий однокерований датчик дальності з унікальними оптичними та електричними алгоритмами, який може реалізувати точне та швидке вимірювання відстані, а також зберігати стабільну та високочутливу продуктивність.

TFmini Plus заснований на принципі TOF, а саме на принципі "Час польоту". Виріб періодично випромінює модуляційну хвилю ближнього інфрачервоного променя, яка буде відображена після контакту з об'єктом.

Пристрій отримує час польоту, вимірюючи різницю фаз у зворотному напрямку, а потім обчислює відносний діапазон між продуктом та об'єктом виявлення.

Датчик TFmini Plus значно покращив усі свої показники, збільшивши частоту кадрів з 100 Гц до 1000 Гц, зменшивши сліпу зону до 10 см, підвищивши точність та стабільність. Більше того, TFmini Plus знаходиться в корпусі з типом захисту IP65, що може ефективно запобігати потраплянню води та пилу, а також робить продукт більш пристосованим до зовнішнього світла або навколишнього середовища з різними температурами та відбиттям.

Датчик може забезпечити набагато менше енергоспоживання та більш гнучку частоту виявлення. Датчик сумісний із зв'язком по UART та IIC (надіслати команду для перемикачів) інтерфейсах та живиться від стандартних 5В. Середня потужність 0.55 Вт.

Цей датчик добре працює з усілякими контролерами типу Arduino. Його можна легко інтегрувати в систему при використанні з бібліотекою Arduino,

розробленою від DFRobot. Технічні характеристики TFmini Plus наведені в таблиці 3.2.

Таблиця 3.2 – Технічні характеристики Lidar TF Luna

Інтерфейс	UART, I2C, I/O
Інше	допуск - 5мм
Відстань	0.1-12м
Діапазон температур	-20°C~60°C
Захист	IP65
Кут	3.6°
Логічні рівні	LVTTL (3.3В)
Матеріал	ABS+PC
Напруга	5В±0.5В
Потужність	550мВт (режим низької потужності: 85мВт)
Струм	≤110мА
Частота	кадрів: 1-1000 Гц (регулюється)
Точність	± 5 см @ (0,1-6м), ± 1% @ (6м-12м)
Довжина	850 нм; кабеля - 30см

### 3.1.3 Ультразвуковий датчик HC-SR04

Ультразвукові датчики відстані Ардуїно дуже потрібні в робототехнічних проектах через свою відносну простоту, достатню точність і доступність. Вони можуть бути використані як прилади, що допомагають об'їжджати перешкоди, отримувати розміри предметів, моделювати карту приміщення та сигналізувати про наближення або видалення об'єктів. Одним з найпоширеніших варіантів такого пристрою є датчик відстані, в конструкцію якого входить ультразвуковий далекомір HC SR04.

Робота модуля ґрунтується на принципі ехолокації. Модуль посилає ультразвуковий сигнал та приймає його відображення від об'єкта. Вимірявши час між відправкою та отриманням імпульсу, не складно обчислити відстань до перешкоди.

#### Принцип роботи

1. На вихід trig (тригер) посилаємо високий рівень протягом як мінімум 10мкс.

2. Модуль починає посылати ультразвукові імпульси з частотою 40 кГц і приймає їх назад, якщо в зоні видимості є будь-які перешкоди.
3. Якщо сигнал повертається, модуль встановлює низький рівень на виході echo на 150мс. За часом, який минув з п.1 до низького рівня на виході echo можна розрахувати відстань до перешкоди.

Датчик відстані Ардуїно є приладом безконтактного типу і забезпечує високоточний вимір та стабільність. Діапазон дальності його виміру становить від 2 до 400 см. На його роботу не суттєво впливає електромагнітні випромінювання та сонячна енергія. В комплект модуля з HC SR04 arduino також входять ресівер та трансмітер.

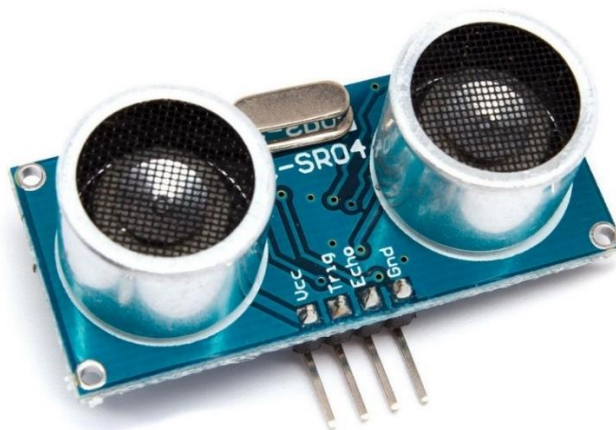


Рисунок 3.4 – HC-SR04

Ультразвуковий далекомір HC SR04 має такі технічні параметри:

- робоча напруга: 3.8 - 5.5В;
- тип: HC-SR04;
- струм: 8 мА;
- частота: 40 кГц;
- максимальна дистанція: 400 см;
- мінімальна дистанція: 0 см;
- роздільна здатність: 3 мм;
- ширина імпульсів: 10 мкс;

- кут: 15°.
- зовнішні габарити: 37x20x15 мм.

### 3.1.4 Цифровий датчик Холла А3144

Вимірювати швидкість транспортного засобу, що рухається, за допомогою плати Arduino можна різними способами, проте використання датчика Холла є найпростішим і економнішим способом зробити це. Датчик Холла - це пристрій, здатний визначати полярність магніту. Якщо один із кінців магніту помістити поруч із датчиком Холла, то датчик змінить свій стан. Існує багато різних виконань цього датчика, але при його покупці пам'ятайте, що для нашого проекту потрібен цифровий датчик Холла.

Для роботи пристрою необхідно прикріпити невеликий шматок магніту на колесо нашого транспортного засобу. При цьому завжди коли магніт перетинатиме (перебуватиме поруч) датчик Холла, датчик буде виявляти це і передаватиме відповідну інформацію на плату Arduino.

Щоразу, коли поруч із датчиком Холла (рис. 3.5) буде виявлятися магніт генеруватиметься переривання в платі Arduino.

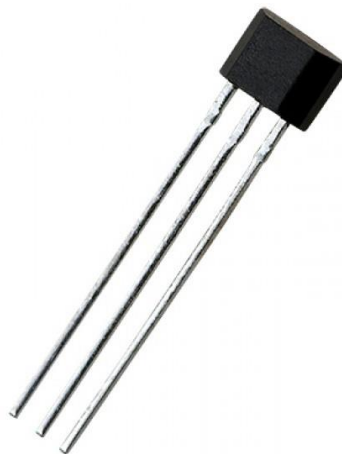


Рисунок 3.5 – Цифровий датчик Холла А3144

#### **Характеристики:**

- висока температурна стабільність, може використовуватися в автомобільній індустрії;

- тип датчика: уніполярний (низький стан виходу відповідає прикладеному південного полюса магніту, високе - видалення магніту; на північний полюс датчики не реагують);
- робоча напруга 4,5 - 24 В;
- цифровий вихід 25 мА, сумісний з цифровою логікою;
- захист від переплюсовки живлення;
- працює з малими магнітами;
- висока надійність;
- малий розмір;
- терморегулятори: перший вивід: (+) живлення, другий: (-) живлення, третій: сигнальний (відкритий колектор).

Центральний контролер збирає інформацію з локальних датчиків і, в залежності від отриманих результатів, виконує запрограмовані дії, не потребуючи втручання користувача.

### **3.2 Вибір технології та мови програмування**

Як програмне середовище розробки для Arduino, буде використовуватися Arduino IDE, оскільки воно відповідає всім основним вимогам:

- кросплатформне;
- безкоштовне;
- відкритий вихідний код;
- легка настройка, установка і простота у використанні;
- різноманітність бібліотек, завдяки яким можна розширювати функціонал програми.

Інтерфейс середовища програмування Arduino IDE показано на рис. 3.4.

В якості мови програмування для Arduino, буде використовуватися мова C/C++. Вона багатofункціональна і не вимагає додаткових налаштувань в середовищі розробки Arduino IDE.

C/C ++ – об'єктно-орієнтована мова програмування, що забезпечує

модульність, роздільну компіляцію, обробку винятків, абстракцію даних. Так само є одним з найбільш широко використовуваних і поширених мов програмування. Використовується не тільки в розробці програмного забезпечення, але і при розробці драйверів різноманітних пристроїв. Добре підходить для програмного середовища Arduino IDE, так як спочатку мова програмування пристроїв Arduino заснована на C/C ++. На даний момент це найпоширеніший і доволі зручний спосіб запрограмувати мікроконтролер.

The image shows a screenshot of the Arduino IDE interface. The window title is "Blink | Arduino 1.8.5". The code editor displays the following C++ code for the Blink example:

```
This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink
*/

// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}

32 Arduino/Genuino Uno on COM1
```

Рисунок 3.6 – Arduino IDE

Є три основні частини, з яких складається мова програмування Arduino. Перш за все, функції, які дозволяють керувати дошкою. За допомогою функцій можливо аналізувати символи, виконувати математичні операції та виконувати різні інші завдання - наприклад, `digitalRead ()` та `digitalWrite ()` дозволяють читати або писати значення.

Кожен скетч, написаний мовою Arduino, містить дві функції. Це `setUp ()` та `loop ()`. Скетч завжди починається з `setUp ()`, який виконується один раз після увімкнення або скидання плати. Після його створення використовується

loop (), щоб повторювати програму повторно, доки не вимкнеться живлення або не відбудеться скидання плати.

Також є значення Arduino, які представляють константи та змінні. Більшість типів даних (array, bool, char, float тощо) подібні до тих, що існують у C ++. Можливо виконати перетворення типу. Остання частина мови ардуїно називається структурою. Вона містить невеликі елементи коду, такі як оператори.

### 3.2.1 Вимоги до синтаксису і структури кода

Спілкування з мікроконтролером відбувається письмово мовою програмування, мова ця дуже чітка, строга і має свій синтаксис і деякі норми оформлення. І якщо синтаксична помилка призводить до помилки компіляції коду або до неправильної роботи пристрою, оформлення коду служить для зручності його читання програмістом.

Що стосується синтаксису, то він не схожий на синтаксис C ++. Проте він має свої вимоги:

- Тіла функцій полягають у фігурні дужки. Код усередині фігурних дужок іноді називають блоком коду.
- Кожна команда закінчується крапкою з комою.
- Метод застосовується до об'єкта через точку. приклад: Serial.begin();
- Виклик функції або методу завжди закінчується дужками, навіть якщо функція не приймає параметри.
- Розділювач десяткових дробів – крапка. Приклад: 0.25 У коми тут інше застосування.
- Комами перераховуються аргументи функцій і способів, члени масиву, і навіть через кому можна здійснити кілька дій в одну строчку.
- Одиночний символ полягає в поодинокі лапки.
- Рядок і масив символів полягає в подвійні лапки.

- Імена змінних можуть містити латинські літери у верхньому та нижньому регістрі (великі та маленькі), цифри та підкреслення.
- Імена змінних не можуть починатися з цифри. Тільки з літери чи підкреслення.
- Реєстр має значення, тобто. велика буква відрізняється від маленької.

До синтаксису можна віднести коментарі, т.к. у різних мовах вони виділяються по-різному. Коментар це типовий текст, який ігнорується на етапі компіляції і потрапляє у підсумкову програму для МК. Коментарі потрібні пояснення коду, як собі самому, і іншим можливим його читачам. У C++ у нас два типи коментарів – однострочні та багатострочні. Додаючи коментарі, необхідно пам'ятати, що компілятор Arduino повністю їх ігноруватиме. Це означає, що він не експортуватиме їх у процесор і не використовуватиме пам'ять мікроконтролера.

Є таке поняття, як форматування (вирівнювання) коду, тобто дотримання пробілів та інтервалів. У всіх серйозних середовищах розробки є автоформатування коду, воно працює як у процесі написання, і “по кнопці”. Arduino IDE – не виняток, у ній код форматується комбінацією клавіш Ctrl+T. Перш ніж переходити до структури та порядку частин коду, потрібно запам'ятати:

- Змінна будь-якого типу повинна викликатись лише після свого оголошення. Інакше буде помилка.
- Оголошення та використання класів або типів даних із бібліотеки/файлу має бути після підключення бібліотеки/файлу.
- Функція може викликатися як до, так і після оголошення, тому що C++ компілювана мова, компіляція проходить у кілька етапів, і функції "виділяються" окремо, тому можуть викликатися в будь-якому місці програми.



При запуску Arduino IDE дає нам заготовівлю у вигляді двох обов'язкових функцій: `setup()` та `loop()`. Код у блоці `setup()` виконується один раз під час кожного запуску МК. Код у блоці `loop()` виконується “по колу” протягом роботи програми, починаючи з завершення виконання `setup()`. Для допитливих: якщо ви вже знайомі з мовою C++, то, можливо, запитаете “а де ж `int main()` і взагалі файл `main.cpp`?”. Все дуже просто: `int main()` за вас уже написали всередині файлу `main.cpp`, який лежить глибоко у файлах “ядра”, а `setup()` та `loop()` вбудовані в нього так:

```
// main.cpp
// где-то в глубинах ядра Arduino
int main() {
    setup();
    for (;;) {
        loop();
    }
    return 0;
}
```

Рисунок 3.7 – Обов'язкові функції

### 3.2.2 Опис використаних функцій

#### 1. Функція розрахунку швидкості

```
void intEnc() {
    unsigned long thisTime = micros();
    if( thisTime-oldTime>65535) {dt = 65535;}else{dt = thisTime - oldTime;}
    oldTime = thisTime;
    if ( digitalRead(INT_DIR_PIN) ) {
        enc++;
        w_interrupt=(360.0f*1000000.0f)/(REDUCTION*IPR*dt);
    } else {
        enc--;
        w_interrupt--(360.0f*1000000.0f)/(REDUCTION*IPR*dt);
    }
}
```

Рисунок 3.8 – Функція розрахунку швидкості

## 2. Функція розрахунку відстані

```
#define PIN_TRIG 12
#define PIN_ECHO 11

#define MAX_DISTANCE 200

NewPing sonar(PIN_TRIG, PIN_ECHO, MAX_DISTANCE);

void setup() {
    Serial.begin(9600);
}

void loop() {
    delay(50);
    unsigned int distance = sonar.ping_cm();

    // Печатаем расстояние в мониторе порта
    Serial.print(distance);
    Serial.println("cm");
}
```

Рисунок 3.9 – Функція розрахунку відстані

## 3. Функція виводу критичних параметрів

```
void adbEventHandler(Connection * connection, adb_eventType event, uint16_t length, uint8_t * data)
{
    if (event == ADB_CONNECTION_RECEIVE)
    {
        Serial.print("data:");
        Serial.println(data[0],DEC);
        if((data[0]) == COMMAND_SEND_TRUE) SendToAndroid = true;
        else if ((data[0]) == COMMAND_SEND_FALSE) SendToAndroid = false;
        else if ((data[0]) == COMMAND_PLAY_BEEP) playBeep();
    }
    else if (event == ADB_CONNECTION_OPEN) Serial.println("ADB connection open");
    else if (event == ADB_CONNECTION_CLOSE) Serial.println("ADB connection close");
    else {
        Serial.println(event);
    }
}

void playBeep() {
    for (int j = 0; j < 10; j++) {
        analogWrite(BeeperPin, 20);
        delay(50);
        analogWrite(BeeperPin, 0);
        delay(150);
    }
}
```

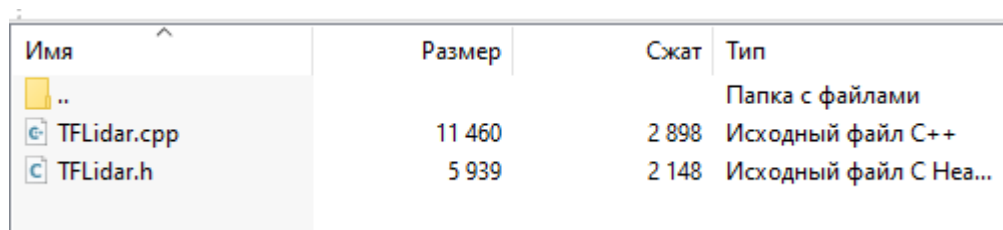
Рисунок 3.10 – Функція виводу критичних параметрів

### 3.3 Вибір компонентів АПЗ

В роботі використовуються бібліотеки та просто зовнішні файли, вони підключаються до головного файлу (файлу прошивки) за допомогою директиви `#include`. Ця команда повідомляє компілятору, що потрібно знайти та додати до програми зазначений файл. Цей файл може містити свої `#include` і тягнути за собою інші файли, таким чином програма може бути розбита на безліч незалежних файлів.

Структура бібліотеки:

- бібліотека - це папка, що складається з файлів з файлами коду C++ (`.cpp`) і файлів заголовків C++ (`.h`);
- файл `.h` описує структуру бібліотеки і оголошує все її змінні і функції;
- файл `.cpp` містить реалізацію функції.



Имя	Размер	Сжат	Тип
..			Папка с файлами
TFLidar.cpp	11 460	2 898	Исходный файл C++
TFLidar.h	5 939	2 148	Исходный файл C Неа...

Рисунок 3.11 – Приклад бібліотеки

Бібліотека `Ultrasonic.h`

Для роботи Arduino з датчиком HC-SR04 є готова бібліотека `Ultrasonic`. Конструктор `Ultrasonic` приймає два параметри: номери пінів, до яких підключені висновки `Trig` та `Echo` відповідно:

```
Ultrasonic ultrasonic(12,13);
```

тут виведення датчика `Trig` підключений до 12-го пину Arduino, а `Echo` - до 13-го.

Бібліотека має один метод `Ranging`, як параметр якого задається, у що перераховувати відстань до об'єкта: сантиметри або дюйми:

---

Таким чином рядок `ultrasonic.Ranging (CM)` поверне відстань до об'єкта (типу `long`) в сантиметрах.

Для використання бібліотеки у своїх проєктах помістимо її в папку `libraries` каталогу установки Arduino.

Скетч, що видає відстань до послідовного порту до об'єкта в сантиметрах, представлений у прикладі.

### 3.3.1 Бібліотека `TFLidar.h`

На апаратному рівні далекомір спілкується з електронікою, що управляє, по шині UART або I<sup>2</sup>C. Для комунікації використовуйте бібліотеку `TFLidar`, яка приховує всі тонкощі спілкування з модулем і полегшить написання коду програм.

### 3.3.2 Бібліотека `Stepper.h`

Бібліотека `Stepper` надає зручний інтерфейс керування біполярними та уніполярними кроковими двигунами. Для керування кроковим двигуном, залежно від його типу (біполярний або уніполярний) та обраного способу підключення, знадобляться деякі додаткові електронні компоненти.

### 3.3.3 Бібліотека `SoftwareSerial.h`

Ардуїно реалізована апаратна підтримка інтерфейсу послідовної передачі даних через висновки 0 і 1 (які також використовуються для зв'язку з комп'ютером за допомогою USB). Апаратна робота з послідовним інтерфейсом здійснюється за допомогою вбудованого в мікроконтролер спеціального пристрою, що називається приймачем UART. Він дозволяє мікроконтролеру Atmega обробляти дані навіть під час роботи над іншими завданнями.

Бібліотека `SoftwareSerial` дозволяє реалізувати послідовний інтерфейс на будь-яких цифрових висновках Ардуїно за допомогою програмних засобів, що дублюють функціональність UART (звідси і назва `SoftwareSerial`). Бібліотека

дозволяє програмно створювати кілька послідовних портів, що працюють на швидкості до 115 200 бод. Для пристроїв, які працюють з інвертованим сигналом, у бібліотеці передбачено відповідний параметр, що включає інвертування.

### 3.4 Опис інтерфейсів АПЗ

Для реалізації проекту було створено електричну схему підключення всіх датчиків до Arduino (рис. 3.12).

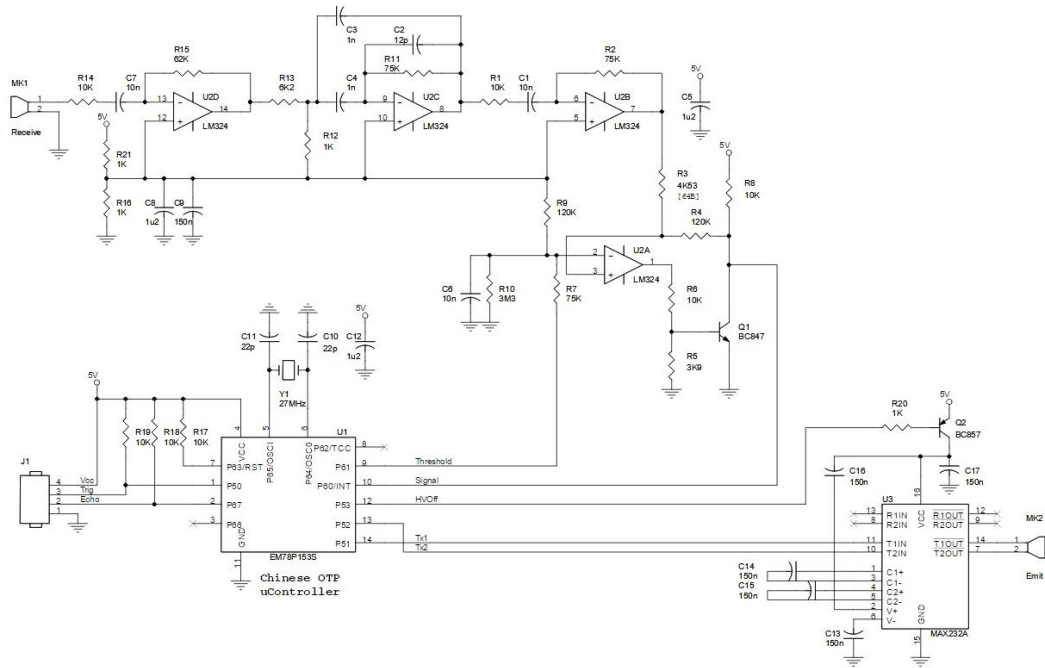


Рисунок 3.12 – Електрична схема підключення датчиків до Arduino

Вимірювати швидкість транспортного засобу, що рухається, за допомогою плати Arduino можна різними способами, проте використання датчика Холла є найпростішим і економічнішим способом зробити це

Для роботи пристрою необхідно прикріпити невеликий шматок магніту на колесо нашого транспортного засобу. При цьому завжди коли магніт перетинатиме (перебуватиме поруч) датчик Холла, датчик буде виявляти це і передаватиме відповідну інформацію на плату Arduino (рис. 3.13).

Щоразу, коли поруч із датчиком Холла буде виявлятися магніт генеруватиметься переривання в платі Arduino.

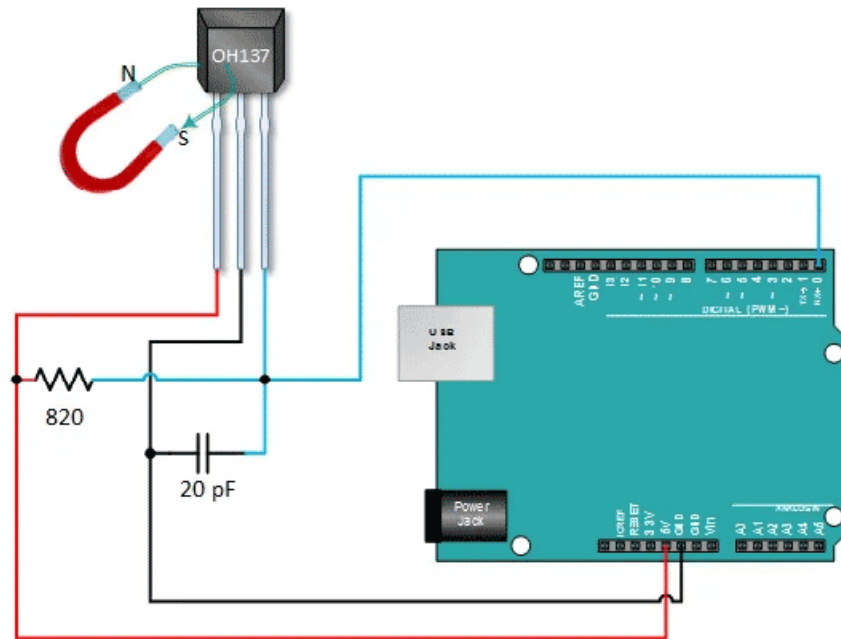


Рисунок 3.13 – Схема підключення датчика Холла до Arduino

Парктронік спроектований таким чином, що подає звуковий сигнал та умовно повідомляє про дистанцію до об'єкта за допомогою світлодіодного індикатора, що складається з восьми світлодіодів. У якості датчика відстані використовується ультразвуковий модуль HC-SR04.

Модуль ультразвукового приймача HC-SR04 використовується для визначення відстані від перешкоди до автомобіля. Модуль видає точне значення відстані від 2 см до 400 см. Той факт, що він забезпечує точні і стабільні показання, робить його більш відповідним елементом для використання в даному випадку. Коли лінія 2 (TRI) модуля отримує "високий" імпульс протягом періоду більше 10 мкс, тільки тоді можна обчислити відстань. Далі відбувається перевірка наявності перешкоди шляхом запуску модуля для відправки восьми циклів ультразвукових імпульсів на частоті 40 кГц, після чого пристрій чекає, поки не буде отримано відбитий ультразвуковий сигнал. Як тільки це станеться, луна-сигнал (контакт 3) модуля встановлюється у високий логічний рівень. Період очікування відбитого імпульсу повністю залежить від відстані, на якій перебуває перешкода. Коли час очікування визначено, можна розрахувати відстань,

використовуючи таку формулу: відстань (у сантиметрах) =  $T/58$ . Тут  $T$  = ширина імпульсу на луні-сигналу в мікросекундах.

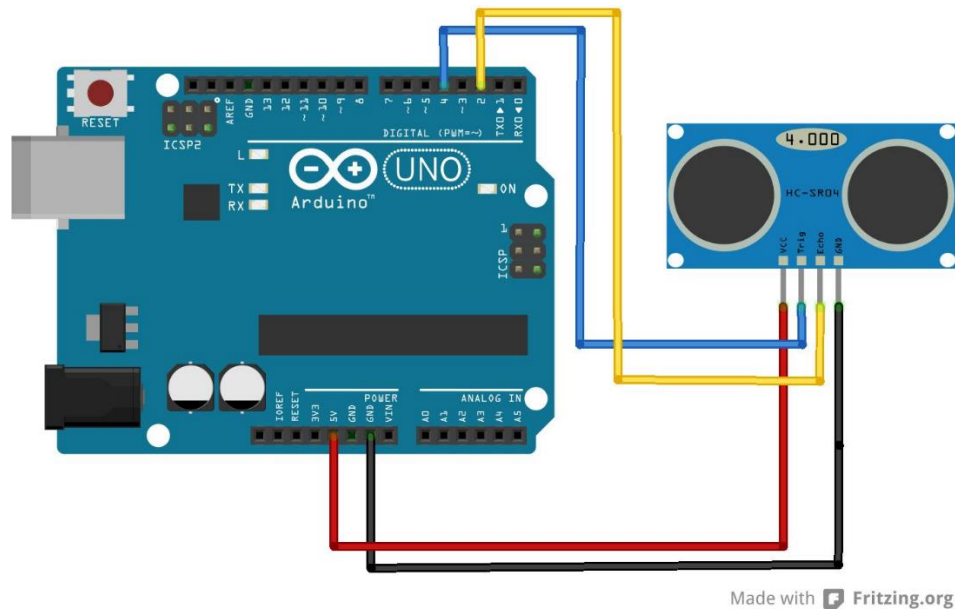


Рисунок 3.15 – Схема підключення HC-SR04 до Arduino

Сканування місцевості – одне з головних завдань для безпілотних роботів, які самостійно прокладають шлях з точки А в Б. Вирішувати її можна по-різному: все залежить від бюджету та поставленої мети, але загальна суть інженерного підходу залишається схожою.

Лідарні системи стали стандартом де-факто для безпілотних автомобілів та роботів. А ще лідер можна приладнати до проекту на Arduino.

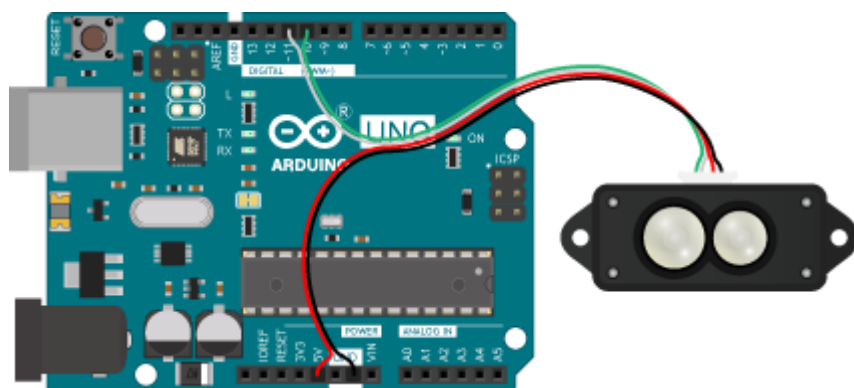


Рисунок 3.16 – Схема підключення TF Luna до Arduino

---

### 3.5 Реалізація системи

Для того щоб вивести параметри з мікроконтролера на смартфон була вибрана платформа Blynk.

Програма Blynk – це добре розроблений конструктор інтерфейсів. Він працює як на платформі iOS, так і на Android. Blynk був розроблений для Інтернету речей. Він може контролювати апаратне забезпечення віддалено, він може відображати дані датчиків, він може зберігати дані, візуалізувати їх і робити багато інших цікавих речей.

У платформі є три основні компоненти:

Blynk App – дозволяє створювати дивовижні інтерфейси для ваших проєктів за допомогою різних віджетів, які ми надаємо.

Blynk Server – відповідальний за всі комунікації між смартфоном і апаратним забезпеченням. Ви можете скористатись нашою Хмарою Blynk або запустити свій приватний Blynk server локально. Це відкрите джерело, він може легко обробляти тисячі пристроїв і навіть може бути запущений на Raspberry Pi.

Blynk Libraries – для всіх популярних апаратних платформ - дозволити спілкування з сервером і обробляти всі вхідні та вихідні команди.

Насамперед для нас ця програма і нам необхідно завантажити програму Blynk для Android. Далі даємо назву нашому проєкту і вибираємо Плату (Пристрій, Device), Arduino Uno .

Вибираємо USB як тип підключення (Connection Type), тому що ми збираємося керувати ним через послідовний зв'язок. У типі підключення USB вам не потрібен інший пристрій. У вікні віджетів багато контролерів. Тому натисніть кнопку і вікно системи буде розміщений на екрані проєкту. Тепер нам потрібно завантажити бібліотеку Blynk для Arduino IDE і помістити бібліотеку в папку Arduino і відкрити IDE Arduino. Тепер підключаємо все згідно зі схемою, що на рис. 3.17.



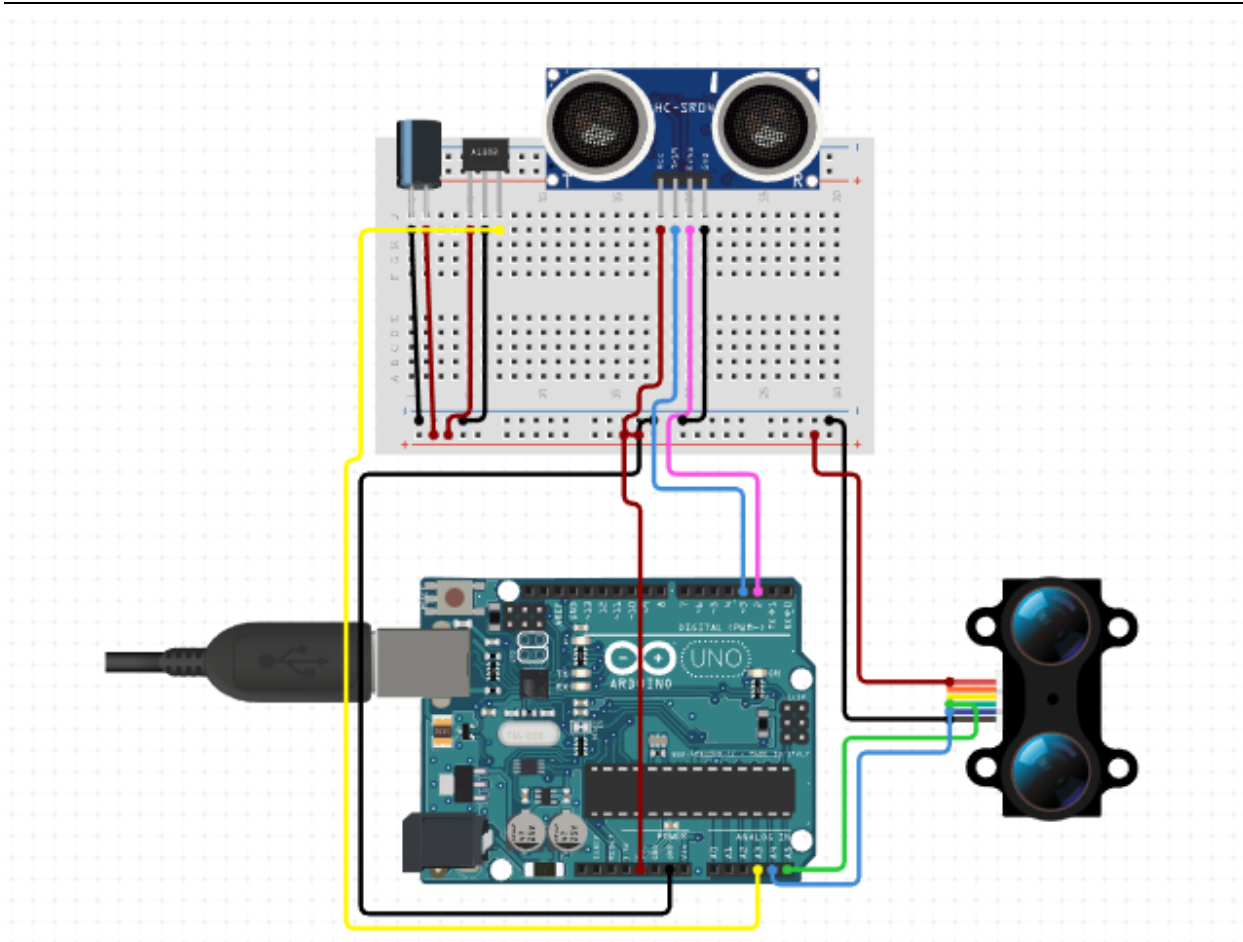


Рисунок 3.17 – Головна схема підключення датчиків

За допомогою програмного інтерфейсу застосунку додаємо спідометр, екран рекомендованої швидкості та паркувальний датчик. І в результаті отримаємо програмний інтерфейс з виводом критичних параметрів на смартфон. Реалізація програмного інтерфейсу наведена на рис. 3.18.

Розроблену систему було протестовано в реальних умовах і отримано хороші результати. Якщо до системи додати ще два датчика, то її можна використовувати в реальних умовах и впевнено паркуватися в складних ситуаціях не дуже досвідченому водієві.

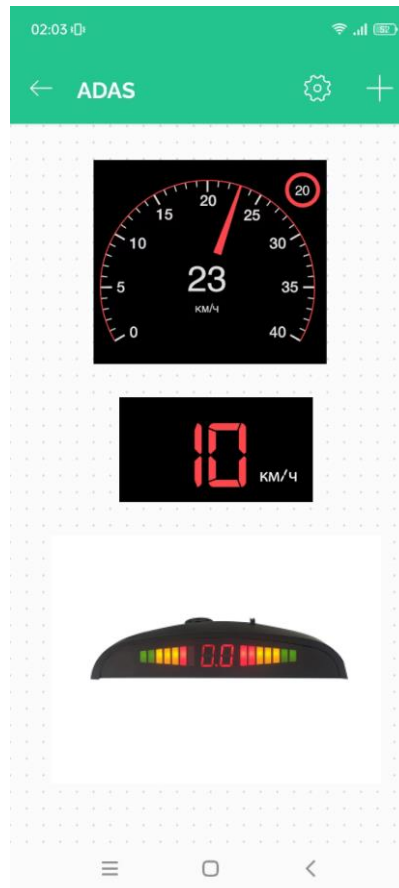


Рисунок 3.18 – Програмний інтерфейс

### 3.6 Висновки до розділу 3

В процесі розробки програмної частини системи збору критичних параметрів на базі мікроконтролера та датчиків були розроблені алгоритми роботи системи, таких як: Lidar TF Luna, датчик Холла А3144 та ультразвукового сонара HC-SR04 та апаратна платформа Arduino Uno. Були підібрані програмні технології та реалізовано систему виводу на смартфон.

В якості мови програмування апаратного модулю використано C++, так як саме на цій мові потрібно програмувати датчики які підключені до системи.

Для виводу критичних параметрів на смартфон для водія була вибрано програмне забезпечення Blynk, що прекрасно з'єднується і працює з Arduino Uno.

Реалізовано і протестовано систему збору інформації для водія.

---

## ВИСНОВКИ

В процесі виконання кваліфікаційної роботи було проведено аналітичний огляд літератури та патентної інформації, визначені переваги та недоліки існуючих систем. Були проаналізовані готові рішення, а саме: розглянуто паркувальні системи, які можуть бути вже вбудованими в авто або встановлені власниками авто.

Визначені технічні параметри та можливості системи, що збирає інформацію для водія.

В результаті огляду комплектуючих, був обґрунтований вибір складових, комплектуючих та технологій для розробки. Для реалізації системи збору критичних параметрів було використано такі датчики: Lidar TF Luna, датчик Холла А3144 та ультразвукового сонара HC-SR04

Розроблен алгоритм роботи системи для парковки.

В якості мови програмування обрана мова C++, а в якості середовища для розробки – Arduino IDE.

Розроблено програмне забезпечення для збору інформації і виводу на смартфон.

В результаті, було створено та протестовано систему збору інформації для водія з виведенням критичних параметрів на смартфон. Ця система може використовуватися на автомобілях будь якого класу та діапазону ціни.

Враховуючи все вищенаведене, всі задачі роботи було виконано, а мету роботи, що полягала у створенні системи збору критичних параметрів з виводом на смартфон досягнуто.

Також було розглянуто питання з охорони праці та безпеки життєдіяльності

Запропоновану систему в майбутньому можна вдосконалити та автоматизувати для вирішення більш глобальних та складніших проблем, з якими можуть зіткнутися водії.

---

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Larry Hardesty (January 21, 2010). "Explained: Gallager codes". MIT News. Retrieved August 7, 2013.
2. R. G. Gallager, "Low density parity check codes," IRE Trans. Inf. Theory, vol. IT-8, no. 1, pp. 21- 28, Jan. 1962.
3. J. Moshieff, N. Resch, N. Ron-Zewi, S. Silas, M. Wootters, "Low-density parity-check codes achieve list-decoding capacity," SIAM Journal on Computing, FOCS20-38-FOCS20-73.
4. Robert G. Gallager (1963). Low Density Parity Check Codes (PDF). Monograph, M.I.T. Press. Retrieved August 7, 2013.
5. David J.C. MacKay and Radford M. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes," Electronics Letters, July 1996.
6. Telemetry Data Decoding, Design Handbook.
7. Thomas J. Richardson and Rüdiger L. Urbanke, "Efficient Encoding of Low-Density Parity-Check Codes," IEEE Transactions on Information Theory, 47(2), February 2001.
8. Y. Kou, S. Lin and M. Fossorier, "Low-Density Parity-Check Codes Based on Finite Geometries: A Rediscovery and New Results," IEEE Transactions
9. Presentation by Hughes Systems Archived 2006-10-08 at the Wayback Machine.
10. HomePNA Blog: G.hn, a PHY For All Seasons.
11. IEEE Communications Magazine paper on G.hn Archived 2009-12-13 at the Wayback Machine.
12. IEEE Standard, section 20.3.11.6 "802.11n-2009", IEEE, October 29, 2009, accessed March 21, 2011.
13. "IEEE SA - IEEE 802.11ax-2021". IEEE Standards Association. Retrieved May 22, 2022.
14. Nick Wells. "DVB-T2 in relation to the DVB-x2 Family of Standards" Archived 2013-05-26 at the Wayback Machine.

- 
15. "5G Channel Coding" (PDF). Archived from the original (PDF) on December 6, 2018. Retrieved January 6, 2019.
  16. Maunder, Robert (September 2016). "A Vision for 5G Channel Coding" (PDF). Archived from the original (PDF) on December 6, 2018. Retrieved January 6, 2019.
  17. Thomas J. Richardson and M. Amin Shokrollahi and Rüdiger L. Urbanke, "Design of Capacity-Approaching Irregular Low-Density Parity-Check Codes," IEEE Transactions on Information Theory, 47(2), February 2001.
  18. Thomas J. Richardson and Rüdiger L. Urbanke, "Efficient Encoding of Low-Density Parity-Check Codes," IEEE Transactions on Information Theory, 47(2), February 2001.
  19. Y. Kou, S. Lin and M. Fossorier, "Low-Density Parity-Check Codes Based on Finite Geometries: A Rediscovery and New Results," IEEE Transactions on Information Theory, vol. 47, no. 7, November 2001, pp. 2711- 2736.
  20. Moon Todd, K. Error correction coding: mathematical methods and algorithms. 2005 by John Wiley & Sons. ISBN 0-471-64800-0. - p.614.
  21. Moon Todd, K. Error correction coding: mathematical methods and algorithms. 2005 by John Wiley & Sons. ISBN 0-471-64800-0. - p.653.
  22. "IEEE Spectrum: Does China Have the Best Digital Television Standard on the Planet?". spectrum.ieee.org. Archived from the original on December 12, 2009.

---

## ДОДАТОК А

### КОД ПРОГРАМИ В ARDUINO IDE

```
#include "TFLidar.h"
#include <Ultrasonic.h>
#include <Stepper.h>
Ultrasonic ultrasonic (11, 12);

int del = 5;
int cm1 = 30;
int cm2 = 20;
int cm3 = 10;
int cm4 = 5;

void setup() {
    Serial.begin(9600);
    pinMode(7, OUTPUT);
    pinMode(2, OUTPUT);
    pinMode(3, OUTPUT);
    pinMode(4, OUTPUT);
}

void loop() {
    int distance, sum, total;
    for (byte i = 0; i <= 10; i++) {
        distance = ultrasonic.Ranging(CM);
        sum = sum + distance;
        delay(del);
    }
    total = sum / 10;
    Serial.println("Distance - " + String(total));

    if (total >= cm1) {
        digitalWrite(4, LOW); digitalWrite(3, LOW); digitalWrite(2, LOW);
    }
    if (total < cm1 && total >= cm2) {
        digitalWrite(4, LOW); digitalWrite(3, LOW); digitalWrite(2, HIGH);
    }
    if (total < cm2 && total > cm3) {
        digitalWrite(4, LOW); digitalWrite(3, HIGH); digitalWrite(2, LOW);
    }
}
```

```
    if (total <= cm3) {
        digitalWrite(4,HIGH); digitalWrite(3,LOW); digitalWrite(2,LOW);
    }

    if (total > cm4) { noTone(7); }
    if (total <= cm4) { tone(7,100); }
}

const int stepsPerRevolution = 200;
Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);
volatile byte REV;
unsigned long int rpm,RPM;
unsigned long st=0;
unsigned long time;
int ledPin = 13;
int led = 0,RPmlen , prevRPM;
int flag = 0;
int flag1=1;
#define bladesInFan 2
float radius=4.7; // в дюймах
int preSteps=0;
float stepAngle= 360.0/(float)stepsPerRevolution;
float minSpeed=0;
float maxSpeed=280.0;
float minSteps=0;
float maxSteps=maxSpeed/stepAngle;
void setup()
{
    myStepper.setSpeed(60);
    Serial.begin(9600);
    pinMode(ledPin, OUTPUT);
    lcd.begin(16,2);
    lcd.print("Speedometer");
    delay(2000);
    attachInterrupt(0, RPMCount, RISING);
}
void loop()
{
    readRPM();
    radius=((radius * 2.54)/100.0); // преобразуем в метры
    int Speed= ((float)RPM * 60.0 * (2.0 * 3.14 * radius)/1000.0);
    int Steps=map(Speed, minSpeed,maxSpeed,minSteps,maxSteps);
```

```
if(flag1)
{
  Serial.print(Speed);
  Serial.println("Kmh");
  lcd.setCursor(0,0);
  lcd.print("RPM: ");
  lcd.print(RPM);
  lcd.print("          ");
  lcd.setCursor(0,1);
  lcd.print("Speed: ");
  lcd.print(Speed);
  lcd.print(" Km/h          ");
  flag1=0;
}
int currSteps=Steps;
int steps= currSteps-preSteps;
preSteps=currSteps;
myStepper.step(steps);
}
int readRPM()
{
  if(REV >= 10 or millis()>=st+1000)
  {
    if(flag==0)
      flag=1;
    rpm = (60/2)*(1000/(millis() - time))*REV/bladesInFan;
    time = millis();
    REV = 0;
    int x= rpm;
    while(x!=0)
    {
      x = x/10;
      RPMlen++;
    }
    Serial.println(rpm,DEC);
    RPM=rpm;
    delay(500);
    st=millis();
    flag1=1;
  }
}
```



```
void RPMCount()
{
    REV++;
    if (led == LOW)
    {
        led = HIGH;
    }
    else
    {
        led = LOW;
    }
    digitalWrite(ledPin, led);
}

void adbEventHandler(Connection * connection, adb_eventType event,
uint16_t length, uint8_t * data)
{
    if (event == ADB_CONNECTION_RECEIVE)
    {
        Serial.print("data:");
        Serial.println(data[0], DEC);
        if((data[0]) == COMMAND_SEND_TRUE) SendToAndroid = true;
        else if ((data[0]) == COMMAND_SEND_FALSE) SendToAndroid = false;
        else if ((data[0]) == COMMAND_PLAY_BEEP) playBeep();
    }
    else if (event == ADB_CONNECTION_OPEN) Serial.println("ADB connection
open");
    else if (event == ADB_CONNECTION_CLOSE) Serial.println("ADB connection
close");
    else {
        Serial.println(event);
    }
}

void playBeep() {
    for (int j = 0; j < 10; j++) {
        analogWrite(BeeperPin, 20);
        delay(50);
        analogWrite(BeeperPin, 0);
        delay(150);
    }
}
```