

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВЕТЕРИНАРНОЇ
МЕДИЦИНИ ТА БІОТЕХНОЛОГІЙ ІМЕНІ С.З. ГЖИЦЬКОГО
ФАКУЛЬТЕТ МЕХАНІКИ, ЕНЕРГЕТИКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА

першого (бакалаврського) рівня вищої освіти

на тему: “ **Автоматизація системи взаємодії складу харчової
продукції із виробничим цехом** ”

Виконав: ст.гр. Акт-41

Спеціальності 151 – «Автоматизація та
комп'ютерно-інтегровані технології»
(шифр і назва)

Медведев Михайло Олександрович
(Прізвище та ініціали)

Керівник: к.т.н., доц. Луб П.М.
(Прізвище та ініціали)

Рецензенти: к.т.н., доц. Кригуль Р.Є.
(Прізвище та ініціали)

ДУБЛЯНИ-2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ПРИРОДОКОРИСТУВАННЯ

ФАКУЛЬТЕТ МЕХАНІКИ, ЕНЕРГЕТИКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Перший (бакалаврський) рівень вищої освіти
151 – „Автоматизація та комп'ютерно-інтегровані технології”

“ЗАТВЕРДЖУЮ”

Завідувач кафедри _____
д.т.н., проф. А.М. Тригуба
“ _____ ” _____ 202_ р.

ЗАВДАННЯ

на кваліфікаційну роботу студенту

Медведев Михайло Олександрович

1. Тема роботи: «Автоматизація системи взаємодії складу харчової продукції із виробничим цехом»

Керівник роботи Луб Павло Миронович, к.т.н., доцент.

Затверджені наказом по університету від 123/к-с від 25.02.2025.

2. Строк подання студентом роботи 10.06.2025 р.

3. Початкові дані до роботи: 1. Статистичні дані виробничої галузі; 2. Рекламні постери та каталоги обладнання для складських приміщень; 3. Методика проектування систем автоматизації; 4. Типові схеми, технічні характеристики обладнання; 5. ДСТУ, СНіПи.

4. Зміст розрахунково-пояснювальної записки:

1. Аналіз стану питання та об'єкта проектування.

2. Технологічні передумови та проектування системи автоматизації.

3. Проектування програмно-апаратного забезпечення автоматизованої системи складу.

4. Охорона праці та безпека в надзвичайних ситуаціях.

Висновки та пропозиції.

Список використаних джерел.

Додатки.

5. Перелік презентаційного матеріалу _____

6. Консультанти з розділів:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1, 2, 3	<i>Луб П.М., доцент кафедри інформаційних технологій</i>		
4	<i>Городецький І.М., доцент кафедри інженерної механіки</i>		

7. Дата видачі завдання 25.02.2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	<i>Написання першого розділу та означення головних завдань роботи</i>	<i>25.02.2025 – 01.04.2025</i>	
2	<i>Виконання другого розділу та формування головних показників для розрахунків</i>	<i>01.04.2025 – 01.05.2025</i>	
3.	<i>Виконання третього розділу та узагальнення отриманих результатів роботи</i>	<i>01.04.2025 – 01.05.2025</i>	
5.	<i>Написання розділу: «Охорона праці та безпека в надзвичайних ситуаціях»</i>	<i>01.05.2025 – 01.06.2025</i>	
7.	<i>Завершення оформлення кваліфікаційної роботи. Перевірка подібності тексту.</i>	<i>01.06.2025 – 09.06.2025</i>	
8.	<i>Створення презентаційного матеріалу та завершення роботи в цілому</i>	<i>09.06.2025 – 16.06.2025</i>	

Студент _____ Медведєв М.О.
(підпис)

Керівник роботи _____ Луб П.М.
(підпис)

Автоматизація системи взаємодії складу харчової продукції із виробничим цехом. Медведєв М.О. Кваліфікаційна робота. Кафедра ІТ. Дубляни, Львівський НУВМБ, 2025.

56 с. текст. част., 30 рис., 3 табл., 10 слайдів презентації, 20 літ. джерел, 2 додатки.

У кваліфікаційній роботі розглянуто особливості функціонування сучасних складів товарної продукції та проаналізовано їхню класифікацію, конструктивні елементи, а також способи інтеграції зі смарт-виробництвом. Описано основне складське обладнання, особливості його застосування та роль у забезпеченні безперервного виробничого циклу.

На основі принципів Industry 5.0 спроектовано систему автоматизації складу з урахуванням гнучкості, автономності та інтеграції кіберфізичних систем. Особливу увагу приділено розробці моделі автономного транспортного засобу, що виконує навігацію складським простором та забезпечує автоматизоване переміщення матеріальних ресурсів відповідно до заданого маршруту.

Реалізовано програмно-апаратну частину автоматизованої системи: розроблено блок управління, силовий драйвер двигунів, програмне забезпечення для мікроконтролера та реалізовано бездротову передачу команд по Wi-Fi. Результати підтверджують ефективність застосування автоматизованих транспортних засобів у складі для підвищення продуктивності, зниження впливу людського фактора та цифровізації логістичних процесів складу товарної продукції.

Розроблено заходи з охорони праці та безпека в надзвичайних ситуаціях.

Ключові слова: автоматизація складу, роботизований транспортний засіб, мікроконтролер, програмно-апаратний комплекс.

Зміст

ПЕРЕДМОВА	6
1. АНАЛІЗ СТАНУ ПИТАННЯ ТА ОБ'ЄКТА ПРОЕКТУВАННЯ.....	7
1.1. Основні визначення та класифікація складів товарної продукції.....	7
1.2. Обладнання та виробничі системи складських підрозділів.....	10
1.3. Аналіз використання складських роботів у сучасному виробництві	14
2. ТЕХНОЛОГІЧНІ ПЕРЕДУМОВИ ТА ПРОЕКТУВАННЯ СИСТЕМИ АВТОМАТИЗАЦІЇ.....	18
2.1. Industry 5.0 та системи автоматизації складу	18
2.2. Основні компоненти автоматизованої системи	22
2.3 Розробка системи навігації транспортного засобу	24
2.4. Модель системи управління рухом автономного транспортного засобу товарного складу	26
3. ПРОЕКТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ СКЛАДУ	31
3.1. Проектування блоку управління.....	31
3.2. Результати проектування силового драйвера для автоматизованого віка	36
3.3. Результати розробки програмного забезпечення для мікроконтролерної плати	38
3.4. Передача даних по Wi-Fi каналу із управління роботизованим візком	41
4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ	44
4.1. Структурно-функціональний аналіз та модель травмонебезпечних ситуацій	44
4.2. Розрахунок штучного заземлення	46
4.3. Безпека в надзвичайних ситуаціях	47
ВИСНОВКИ ТА ПРОПОЗИЦІЇ	49
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	51
ДОДАТКИ.....	53

ПЕРЕДМОВА

Сучасні тенденції розвитку виробничої діяльності зумовлюють суттєве зростання потреб підприємств у складах, які забезпечують тимчасове зберігання запасів матеріальних ресурсів, незавершеного виробництва та готової продукції.

Автоматизовані складські системи не тільки виключають ручну працю, а й дозволяють економити складські площі, прискорювати складські операції і покращувати контроль за матеріально-технічними запасами, оскільки ПК стежить за місцезнаходженням кожного виробу на складі. Ці системи називають також автоматизованими складами [9].

Водночас актуальність автоматизованих складських систем зростає в умовах цифрової трансформації виробництва та логістики, коли швидкість, точність і ефективність обробки матеріальних потоків стають вирішальними факторами конкурентоспроможності підприємств. Впровадження таких систем дає змогу інтегрувати складську логістику з загальною виробничою інфраструктурою, мінімізувати втрати часу та ресурсів, оперативно реагувати на зміну попиту, а також забезпечувати прозорість і контроль усіх етапів руху продукції. Тому питання модернізації та автоматизації складів є надзвичайно важливим для сучасного бізнесу.

Мета роботи – розробка автоматизованої системи взаємодії складу харчової продукції із виробничим цехом завдяки проектуванню та використанню автоматизованого транспортного засобу складу.

Завдання кваліфікаційної роботи:

- провести аналіз типів складів, їх технічного оснащення та взаємодії з виробничими підрозділами;
- розкрити модель системи управління автоматизованим транспортним засобом.
- спроектувати програмно-апаратну частину системи автоматизації, включаючи блок управління та силовий драйвер.
- описати бездротову передачу даних і програмне забезпечення для керування роботизованим візком для складу товарної продукції.

1. АНАЛІЗ СТАНУ ПИТАННЯ ТА ОБ'ЄКТА ПРОЕКТУВАННЯ

1.1. Основні визначення та класифікація складів товарної продукції

Склад продукції – це спеціально обладнана територія або будівля, призначена для приймання, зберігання, обробки, комплектації та відвантаження товарів. У контексті харчової промисловості склад є критично важливим елементом логістичного ланцюга, який забезпечує [1, 2]:

- збереження сировини (наприклад, борошно, молоко, овочі), напівфабрикатів (тісто, сирна маса) та готової продукції (упаковані хлібобулочні вироби, молочні продукти);
- дотримання умов зберігання, визначених санітарними нормами (температура, вологість, захист від шкідників);
- координацію між постачальниками, складом і виробничим цехом для забезпечення безперервного виробництва.

Згідно з ДСТУ 4303:2004 "Склади. Терміни та визначення", склад — це комплекс споруд, обладнання та процесів, що забезпечує управління матеріальними потоками. У харчовій промисловості додатково застосовуються стандарти НАССР (Hazard Analysis and Critical Control Points), які регулюють безпечність зберігання продуктів.

Функції складу в харчовій промисловості [1, 2]:

- 1) *зберігання*: Забезпечення умов для збереження якості та безпеки продуктів (наприклад, холодильні камери для молочних продуктів).
- 2) *обробка*: Сортування, пакування, маркування сировини чи готової продукції.
- 3) *комплектація*: Формування партій сировини для передачі до виробничого цеху.
- 4) *транспортна логістика*: Координація переміщення матеріалів між складом, цехом і зовнішніми постачальниками.

5) *контроль якості*: Перевірка сировини перед зберіганням і відвантаженням.

6) *облік і моніторинг*: Відстеження запасів, термінів придатності, партій сировини (згідно з принципами трасування).

Класифікація складів дозволяє систематизувати їх за критеріями, які впливають на організацію роботи, вибір обладнання та рівень автоматизації. У контексті вашої теми особливу увагу приділимо тим характеристикам, які визначають ефективність взаємодії складу з цехом.

Таблиця 1.1 – Загальна класифікація складів [7, 19]

Критерій	Типи складів
За призначенням	Сировини, напівфабрикатів, готової продукції, прицехові, центральні
За температурним режимом	Холодильні, морозильні, з контрольованою температурою, універсальні
За рівнем автоматизації	Ручні, напівавтоматизовані, повністю автоматизовані
За типом власності	Власні, орендовані, логістичні центри
За способом зберігання	Палетні, полицеві, гравітаційні
За розташуванням	Внутрішні, зовнішні

Щодо нормативної бази та стандартів для функціонування складів товарної продукції в Україні то основні із них [8, 9]: 1) ДСТУ 4303:2004: Визначає терміни та вимоги до складів; 2) HACCP (ISO 22000): Регулює безпечність харчових продуктів, включаючи вимоги до складів (контроль температури, санітарія, трасування); 3) ДСТУ ISO 22005:2009: Встановлює принципи трасування у харчовій промисловості; 4) СанПіН 2.3.4.545-96: Санітарні норми для складів харчової продукції (Україна); 5) Регламент ЄС №852/2004: Міжнародні стандарти гігієни для складів (якщо підприємство експортує продукцію).

Загалом нормативні документи також формують вимоги до складів та розташування продукції: 1) розділення зон для сировини, напівфабрикатів і готової продукції; 2) наявність систем вентиляції, охолодження, захисту від шкідників; 3) обов'язковий облік партій і термінів придатності; 4) регулярні перевірки відповідності санітарним нормам.

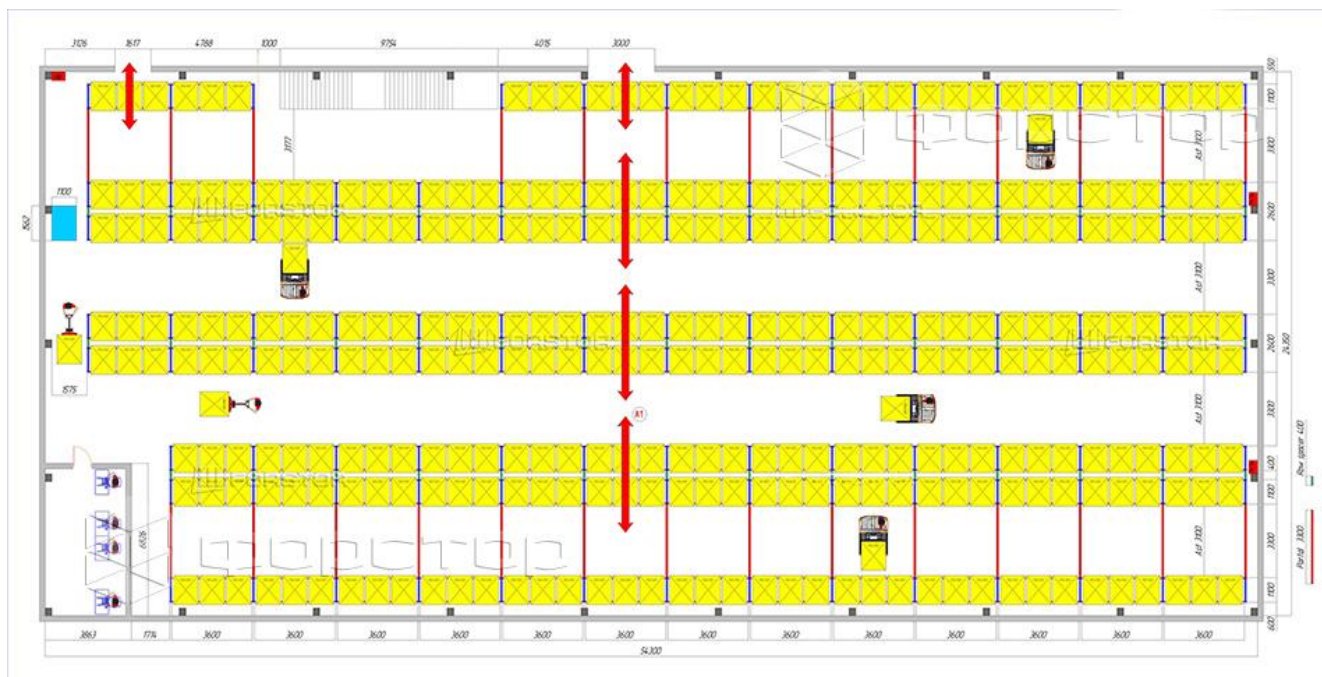


Рисунок 1.1 – Загальна схема розташування товарів на складі

Аналіз сучасних тенденцій у складській логістиці харчової промисловості переконує в таких базових тенденціях: 1) автоматизація та роботизація – впровадження AS/RS, роботів для комплектації, AGV (Amazon використовує роботів Kiva для складів); 2) Інтернет речей (IoT) – датчики для моніторингу умов зберігання (температура, вологість) у реальному часі; 3) Big Data та аналітика – прогнозування попиту на сировину для оптимізації запасів; 4) екологічність – використання енергоефективного обладнання (наприклад, холодильних систем з низьким енергоспоживанням); 5) інтеграція з виробничими системами – WMS об'єднуються з ERP і MES для синхронізації складських і виробничих процесів.

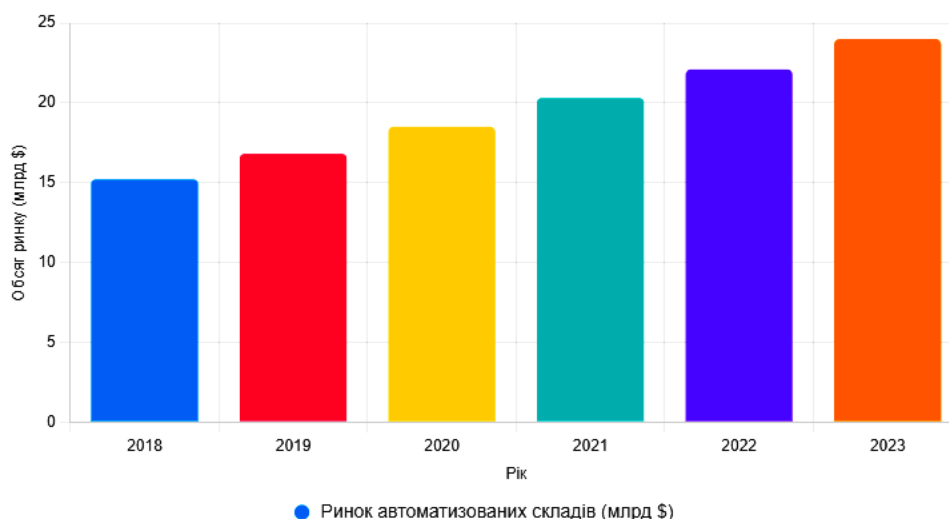


Рисунок 1.2 – Тенденції зростання ринку автоматизованих складів [1]

Отже, склади продукції відіграють ключову роль у виробничому процесі, забезпечуючи безперервність постачання матеріалів та зберігання готової продукції. Вони виступають буферною ланкою між різними етапами виробництва, оптимізують логістику і сприяють ефективному управлінню запасами. Завдяки складам можливо мінімізувати простой обладнання, уникнути перебоїв у виробничому циклі, забезпечити своєчасне виконання замовлень та підвищити загальну гнучкість і конкурентоспроможність підприємства.

1.2. Обладнання та виробничі системи складських підрозділів

Обладнання складів поділяється на кілька категорій залежно від функцій: зберігання, транспортування, обробка та контроль. У харчовій промисловості обладнання має відповідати стандартам ДСТУ, уникати псуванню продуктів і виконувати санітарні норми.

Для ефективного зберігання продукції у харчовому виробництві використовується спеціалізоване обладнання, яке забезпечує належні умови та впорядкованість матеріалів і готової продукції.

Основу складають різні типи стелажів: палетні — для розміщення великих об'ємів сировини, таких як борошно чи цукор, або готової продукції на піддонах;

полицеві — для зберігання дрібних партій, як-от спеції чи харчові добавки; а також гравітаційні стелажі, що забезпечують автоматичне переміщення товарів за принципом FIFO.

Таблиця 1.2 – Класифікація складів виробничої сфери [19, 20]

Категорія	Обладнання	Приклади
Зберігання	Стелажі, холодильні камери	AR Racking, Viessmann
Транспортування	Навантажувачі, конвеєри, AGV	Jungheinrich, Interroll, Kuka
Обробка	Сканери, сортувальники, пакувальники	Zebra, Dematic, Bosch
Контроль	Датчики, вентиляція	Testo, Camfil

Прикладом надійного виробника таких систем є AR Racking. Крім того, критично важливими є холодильні та морозильні камери, які підтримують температурні режими 0-5°C для охолоджених продуктів і -18°C для заморожених, як-от риба чи морозиво.

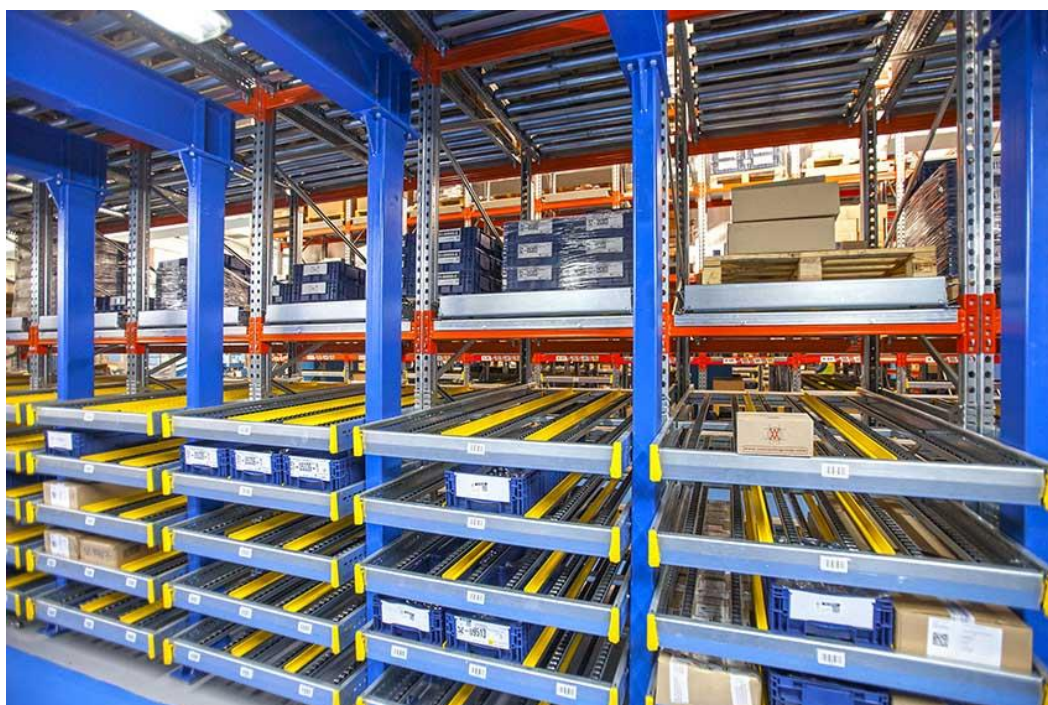


Рисунок 1.3 – Стелажні конструкції складів продукції AR Racking [16, 20]

У харчовій промисловості обладнання для зберігання відіграє вирішальну роль у забезпеченні належних умов для сировини, напівфабрикатів та готової продукції. Для цього широко використовуються різноманітні види стелажів: палетні — зручні для зберігання великогабаритних матеріалів або готової продукції на піддонах (наприклад, мішки з борошном або цукром), полицеві — призначені для дрібної сировини на зразок спецій і харчових добавок, а також гравітаційні стелажі, які забезпечують автоматичне просування товарів вперед, підтримуючи принцип «першим прийшов — першим пішов» (FIFO).

Надійні рішення в цій сфері пропонує компанія AR Racking. Не менш важливими є холодильні та морозильні камери, які дозволяють зберігати продукти при оптимальних температурах: від 0 до +5°C для охолоджених товарів (молочна продукція, м'ясо, овочі) та до -18°C для заморожених виробів, таких як риба чи морозиво.

Сучасні моделі, наприклад, від Viessmann, обладнані системами контролю температури та вологості. Окрім цього, для внутрішнього переміщення та зберігання використовуються пластикові ящики, які відповідають санітарним нормам, а також вакуумні контейнери для збереження якості напівфабрикатів, таких як тісто.



Рисунок 1.4 – Охолоджувальні системи Viessmann для складу [16, 20]

Транспортне обладнання на складах харчової продукції забезпечує ефективне і безпечне переміщення вантажів між зонами зберігання та виробництвом. Одними з основних засобів є навантажувачі та штабелери:

електричні навантажувачі, як-от Jungheinrich EFG 316, застосовуються для транспортування палет із сировиною чи готовою продукцією, а штабелери — зручні для роботи у вузьких проходах та в умовах обмеженого простору. Важливу роль відіграють і конвеєрні системи — стрічкові або роликові, які забезпечують безперервну подачу сировини до виробничих ліній.



Такі візки, як Kuka AGV, активно використовуються у міжнародних компаніях, зокрема Danone, для підвищення точності та ефективності логістичних операцій.



У сучасних складських системах важливу роль відіграє технологічне обладнання для обліку, сортування та пакування продукції. Для ефективної ідентифікації та контролю переміщення партій сировини чи готової продукції використовуються

сканери штрих-кодів і RFID-системи. Завдяки RFID-міткам можливо в реальному часі відстежувати переміщення товарів на складі, що значно підвищує точність обліку й зменшує ризики помилок; прикладом такого обладнання є рішення від Zebra Technologies.

Автоматичні сортувальні системи, як-от Dematic, дозволяють швидко й точно розподіляти продукцію — наприклад, сортувати борошно за типами перед подачею до цеху.



Обладнання для контролю умов зберігання є ключовим елементом у забезпеченні якості та безпеки харчової продукції на складах. Одним із найважливіших компонентів є датчики температури та вологості, які інтегруються з системами управління складом (WMS) і дозволяють в режимі реального часу відслідковувати стан мікроклімату. Це особливо актуально для продукції, чутливої

до коливань температури, наприклад, молочних виробів або заморожених напівфабрикатів. Серед надійних рішень у цій сфері варто відзначити обладнання від компанії Testo [16, 20].

1.3. Аналіз використання складських роботів у сучасному виробництві

Сучасних роботів, що використовуються для автоматизації складських процесів, поділяють на дві групи: промислові та колаборативні [11, 14].

Промислові роботи – це програмовані машини, що замінюють ручну працю при складних повторюваних діях. Така техніка обладнана датчиками обліку даних у реальному часі. На складах цей тип представлений підіймальними механізмами та автоматичними транспортерами.

Колаборативні роботи (коботи) являють собою кооперацію людини і машини. Ці пристрої виконують певні дії разом з людиною. Одна з переваг коботов – можливість програмування, щоб техніка могла працювати автономно або під управлінням людини. У складській сфері коботи представлені маніпуляторами для переміщення вантажів і пакувальними машинами.

Найбільший у світі інтернет-магазин має у своєму розпорядженні понад 200 тисяч роботів, вони залучені на складах 180 країн. Розробкою автоматизованих систем для гіганта займається власна компанія AmazonRobotics (раніше KivaSystems), що перейшла у власність корпорації у 2012 році. Роботизований склад Amazon використовує системи для автоматичного транспортування товарів. По складу товари переміщуються за допомогою конвеєрів і навантажувачів, що керуються людиною. Зберігаються товари на переносних полицях. Після того, як продукт вноситься в базу даних, система направляє до нього безпілотного робота. Робот підіймає полку з товаром і транспортує її до оператора, який обирає потрібний продукт. Складські роботи, зображені на рис. 1.5, переміщаються по комплексу, орієнтуючись за QR-кодами на підлозі. Датчики запобігають зіткненню роботів між собою. Роботи здатні самостійно заряджатися [11, 14].

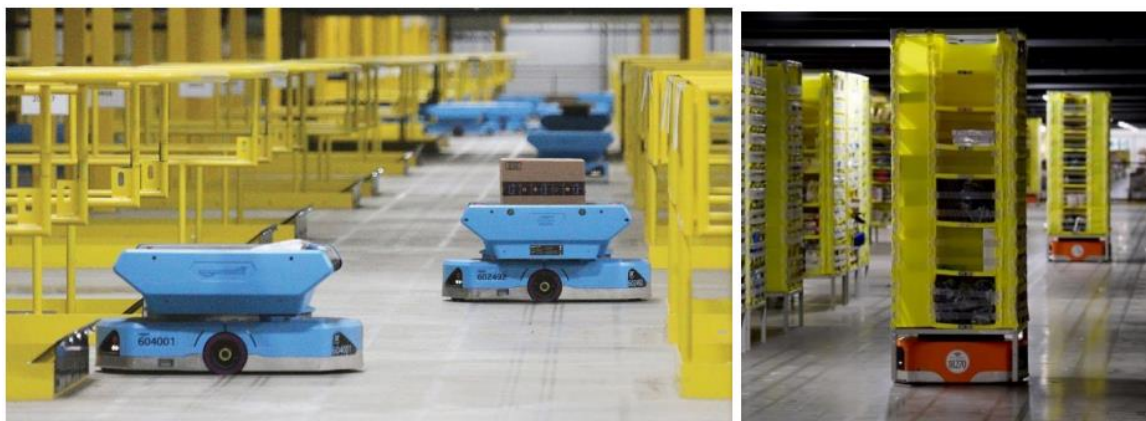


Рисунок 1.5 – Складські роботи Amazon

Cainiao, або China SmartLogistics Network, – китайська логістична компанія групи Alibaba, що доставляє товари по території Китаю за 24 години, а по всьому світу – за 72. Компанія володіє 30 складами загальною площею 1,7 млн . Склад у Хойяні провінції Гуандун використовує для переміщення вантажів безпілотних роботів, розроблених Quicktron Intelligent Technology. Маневрові роботи, зображені на рис. 1.6, здатні обертатися на 360° і переміщати вантажі вагою до 500 кг. Уникнути зіткнення дозволяє вбудована система лазерів. Комунікація з оператором і отримання завдань здійснюється через Wi-Fi. Якщо акумулятор розряджається, робот може поповнити запаси енергії автоматично. Як повідомляє компанія, безпілотні складські роботи дозволили скоротити працю персоналу на 70% [11, 14].



Рисунок 1.6 – Роботи на складах від Cainiao

Osado – найбільший онлайн-ритейлер продовольчих продуктів з Великобританії. Центр дистрибуції в Андовері щотижня обробляє 65 000 замовлень. Особливість автоматичної системи (рис. 1.7), в тому, що роботи пересуваються рейками над осередками з товаром уздовж спеціальної розмітки у вигляді сітки. Їх пересування регулює система контролю трафіку на основі 4G, що дозволяє уникнути зіткнення. Роботи розвивають швидкість 4 м/с. Зарядка акумуляторів відбувається автоматично у спеціальних відсіках. Роботи привозять ящики з продуктами до станцій сортування, де згодом робот або людина формує замовлення. Порожні осередки роботи заповнюють новими продуктами [11, 14].



Рисунок 1.7 – Роботи від Ocado

Sagawa Global – одна з найбільших логістичних компаній Японії. Новий склад в Токіо Sagawa X-Frontier являє собою сучасний автоматизований комплекс, зображений на рис. 1.8. За допомогою армії роботів і систем штучного інтелекту компанія управляє складськими запасами, організовує простір в складських приміщеннях, транспортує вантажі між будівлями і навіть приносить замовлення безпосередньо співробітникам, які працюють на складі [11, 14].

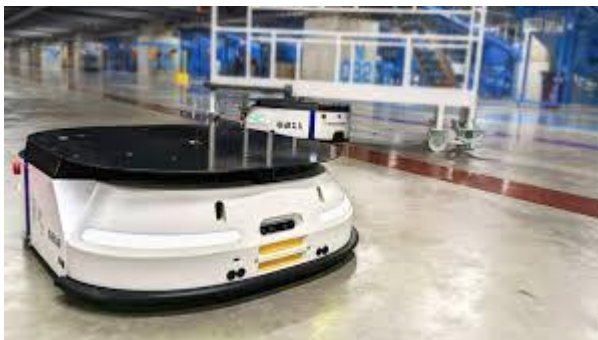


Рисунок 1.8 – Sagawa X-Frontier

Логістична компанія DHL з 2020 року використовує роботи свого партнера, американського розробника автономних мобільних роботів LocusRobotics. Так звані Locus-боти, зображені на рис. 1.9, вже застосовуються на десятках складів у Північній Америці.



Рисунок 1.9 – Складські роботи DHL

Роботи здатні самостійно пересуватися, знаходити необхідні товари й доставляти їх співробітникам. Таким чином компанія скорочує час виконання повторюваних або фізично складних завдань [11, 14].

2. ТЕХНОЛОГІЧНІ ПЕРЕДУМОВИ ТА ПРОЕКТУВАННЯ СИСТЕМИ АВТОМАТИЗАЦІЇ

2.1. Industry 5.0 та системи автоматизації складу

Концепція Industry 5.0 становить новий етап розвитку промислових технологій, що спирається на досягнення Industry 4.0, але з акцентом на гармонізацію технологічних рішень із людськими потребами, екологічною сталістю та гнучкістю виробничих процесів. У контексті складської логістики харчової промисловості Industry 5.0 відкриває можливості для створення високоефективних автоматизованих систем, які забезпечують швидке реагування на зміни у виробничих графіках, дотримання суворих санітарних норм, оптимізацію енергоспоживання та інтеграцію з системами управління підприємством. У цьому підпункті розглядаються основні принципи Industry 5.0, ключові технології для автоматизації складів, переваги їхнього впровадження, а також економічні аспекти та практичні приклади.

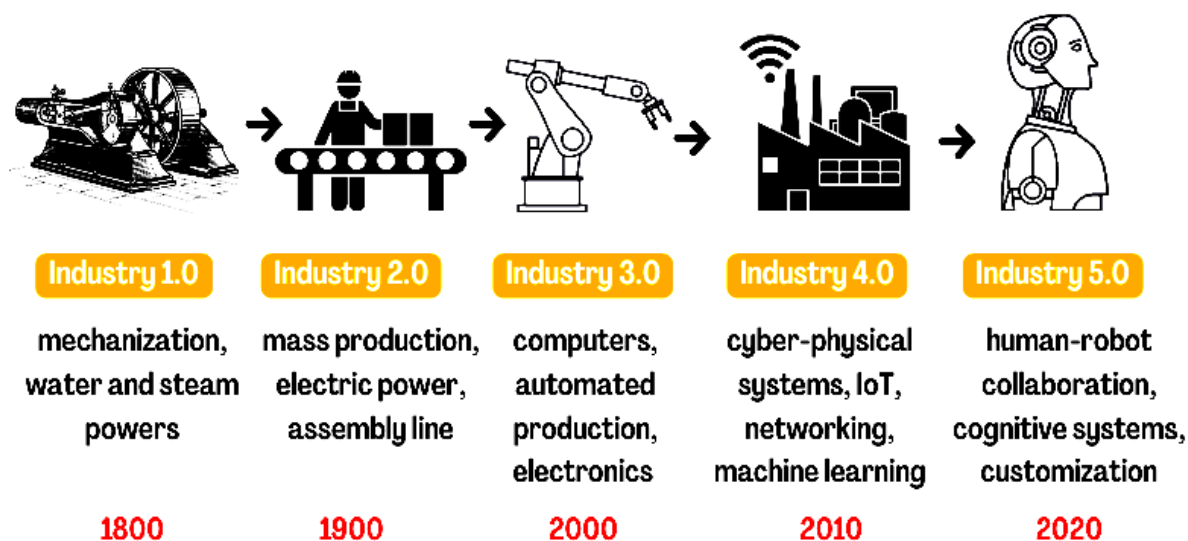


Рисунок 2.1 – Еволюція розвитку промислових технологій [11]

Industry 5.0 базується на фундаменті Industry 4.0, яка фокусувалася на цифровізації та автоматизації через впровадження таких технологій, як Інтернет речей (IoT), Big Data, штучний інтелект (ШІ), хмарні обчислення та робототехніка.

Однак Industry 5.0 виходить за межі суто технологічного прогресу, додаючи три основні принципи: людиноцентричність, екологічність і гнучкість. Людиноцентричність передбачає, що технології мають полегшувати роботу персоналу, а не повністю замінювати його. Наприклад, колаборативні роботи (коботи) працюють пліч-о-пліч із операторами, беручи на себе монотонні завдання, такі як пакування чи сортування. Екологічність зосереджується на використанні енергоефективного обладнання та скороченні відходів, що сприяє сталому розвитку. Гнучкість забезпечує адаптацію систем до змін у виробничих планах, сезонності сировини чи попиту на продукцію.

Таблиця 2.1 – Відмінності технологій Industry [11]

№ п/п	Industry 4.0,	Industry 5.0
1	Центрована навколо виробничої ефективності шляхом кращої цифрової інтеграції та використання Штучного Інтелекту	Забезпечує галузеві фреймворки (рамки розвитку), які комбінують завдання конкурентоспроможності та сталості
2	Технологічні зміни є головним чинником змін в переході на кібер-фізичну взаємодію	Фокусує на впливі альтернативних (крім технологій) режимів керування моделями, що ведуть до сталості та стійкості
3	Присутнє вирівнювання з оптимізацією бізнес моделей в рамках існуючих ринків капіталу та економічних моделей, які, наприклад, автоматично фокусуються на скороченні витрат та максимізації прибутку для стейкхолдерів	Робить працівників більш значимими та сильнішими через використання цифрових пристроїв, й уособлює, таким чином, людиноцентричний підхід до використання технологій
4	Немає фокусу на характеристиках дизайну та ефективності в напрямках повторного використання ресурсів та матеріалів, щоб мінімізувати кліматичні та соціальні ризики	Будує шлях по переходу до сталої економіки
5	-	Розширює корпоративну відповідальність до повних ЛДВ
6	-	Вводить індикатори, що показують для кожної галузевої екосистеми прогрес в досягненні стійкості, процвітання та сталості

Для складів харчової промисловості, де обробляються швидкопсувні продукти, такі як молоко, м'ясо чи овочі, принципи Industry 5.0 мають особливе значення. Склади мають забезпечувати швидке реагування на зміни у виробничому

графіку, наприклад, перехід від виробництва йогуртів до сирів, а також дотримуватися стандартів HACCP, ISO 22000 та ISO 22005.

Автоматизовані системи дозволяють оптимізувати енергоспоживання, зокрема за рахунок енергоефективних холодильних камер, і синхронізувати складські процеси з виробничими через інтеграцію з системами ERP і MES. Такі рішення підвищують продуктивність, знижують втрати сировини та забезпечують відповідність санітарним нормам.

Ключові технології Industry 5.0 для автоматизації складів [11]:

- *Інтернет речей (IoT).* Технологія IoT забезпечує моніторинг умов зберігання в реальному часі за допомогою датчиків, таких як Testo 174H, які відстежують температуру (0–5°C для молочних продуктів) і вологість із точністю $\pm 0,5^\circ\text{C}$ та $\pm 3\%$. Датчики передають дані до системи управління складом (WMS) через Wi-Fi або LoRaWAN, генеруючи сповіщення у разі відхилень, наприклад, якщо температура перевищує 5°C. Для складу площею 1000 м² із десятима холодильними зонами необхідно 50 датчиків вартістю \$100 кожен, що становить \$5000. IoT також підтримує трасування партій через RFID-мітки, які відповідають стандарту ISO 22005, дозволяючи відстежувати походження сировини з точністю 99,99%. Впровадження RFID-системи Zebra коштує \$20,000, що знижує ризик штрафів за порушення стандартів (\$5000/рік).



- *Автоматизовані керовані транспортні засоби (AGV).* AGV, наприклад, Kuka KMP 600, оснащені LIDAR і камерами, транспортують палети з сировиною, як-от борошно чи цукор, у проходах шириною 1,5 м, скорочуючи час доставки до цеху з десяти до трьох хвилин. Для складу 1000 м², який обробляє 10



тонн сировини щодня, потрібно три AGV вартістю \$50,000 кожен, тобто \$150,000. Це дозволяє замінити двох операторів, заощаджуючи \$19,200 щороку (\$800/міс $\times$ 12 $\times$ 2). З урахуванням скорочення помилок і простоїв окупність

становить 7-8 років. AGV підвищують гнучкість, адаптуючись до змін у маршрутах доставки.

- *Штучний інтелект і Big Data.* ШІ аналізує історичні дані та сезонність, прогножуючи попит на сировину, наприклад, пікові постачання овочів у липні–серпні. Алгоритми машинного навчання в WMS, таких як SAP (Systems, Applications & Products in Data Processing) EWM (Extended Warehouse Management), реалізують принцип FIFO, автоматично обираючи партії з найближчим терміном придатності, що знижує втрати сировини на 10-15%.

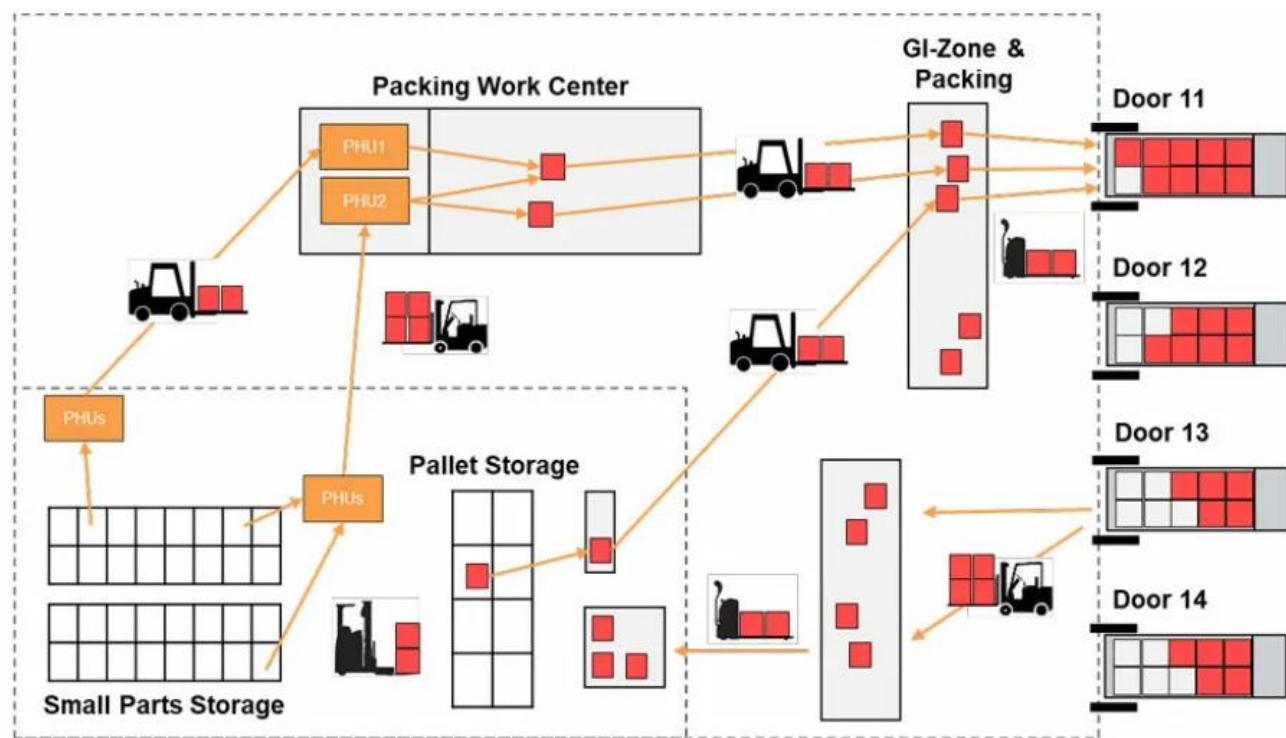


Рисунок 2.2 – Автоматизована система управління розгорнутим складом – SAP EWM [14]

- *Колаборативні роботи (коботи).* Коботи, такі як Universal Robots UR5, працюють поруч із операторами, виконуючи пакування чи маркування готової продукції, наприклад, хлібобулочних виробів, зі швидкістю 100 одиниць за годину порівняно з 50 одиницями для людини. Один кобот вартістю \$35,000 замінює 1,5 оператора, заощаджуючи \$14,400 щороку ($\$800/\text{міс} \times 1,5 \times 12$). Окупність становить 2,5 роки. Коботи підвищують людиноцентричність, дозволяючи персоналу зосередитися на складніших завданнях.

- *Цифрові двійники (Digital Twins).* Віртуальні моделі складу, створені за допомогою платформ, як-от Siemens MindSphere, моделюють маршрути AGV, розташування сировини та взаємодію з цехом, скорочуючи простой на 20%. Впровадження цифрового двійника коштує \$150,000, але заощаджує \$20,000 щороку від \$100,000 втрат через простой. Окупність становить 7,5 років. Цифрові двійники підвищують гнучкість, дозволяючи тестувати нові конфігурації без фізичних змін.

- *Хмарні обчислення.* Хмарні WMS, наприклад, Oracle Cloud WMS, обробляють дані в реальному часі, інтегруючись із ERP і MES для синхронізації замовлень сировини з виробничим планом.

Річна підписка коштує \$30,000, але скорочення помилок в обліку (2% від \$500,000 обороту) заощаджує \$10,000 щороку. Хмарні рішення забезпечують масштабованість і доступність даних, що критично для динамічних складських операцій.



Отже, технології Industry 5.0, такі як IoT, AGV, ШІ, коботи, цифрові двійники та хмарні WMS, дозволяють створювати гнучкі, енергоефективні та людиноцентричні складські системи. Забезпечують швидку подачу сировини, відповідність стандартам і зниження відходів, але потребують значних інвестицій і вирішення питань інтеграції та кібербезпеки. Впровадження таких систем підвищує конкурентоспроможність підприємств харчової промисловості.

2.2. Основні компоненти автоматизованої системи

Побудову систем здійснюють за допомогою структур, що описують стійкі зв'язки між їх елементами. При описі АС використовують такі види структур, що відрізняються типами елементів і зв'язків між ними:

- функціональні (елементи – функції, завдання, процедури; зв'язки – інформаційні);
- технічні (елементи – пристрої, компоненти та комплекси; зв'язки – лінії і канали зв'язку);
- організаційні (елементи – колективи людей і окремі виконавці; зв'язки – інформаційні, підпорядкування та взаємодії);
- документальні (елементи – неподільні складові частини та документи АС; зв'язки – взаємодії, вхідність та підпорядкування);
- алгоритмічні (елементи – алгоритми; зв'язки – інформаційні);
- програмні (елементи – програмні модулі та вироби; зв'язки – керуючі);
- інформаційні (елементи – форми існування і подання інформації в системі; зв'язки – операції перетворення інформації в системі).

Беручи до уваги класичну систему автоматизованої системи, було складено схему автоматизованої системи процесів логістики складського приміщення, яка матиме наступний вигляд, зображений на рисунку 2.3.

У ній підібрані саме ті комплекси й структури, які в сукупності реалізують основну задачу системи [1].

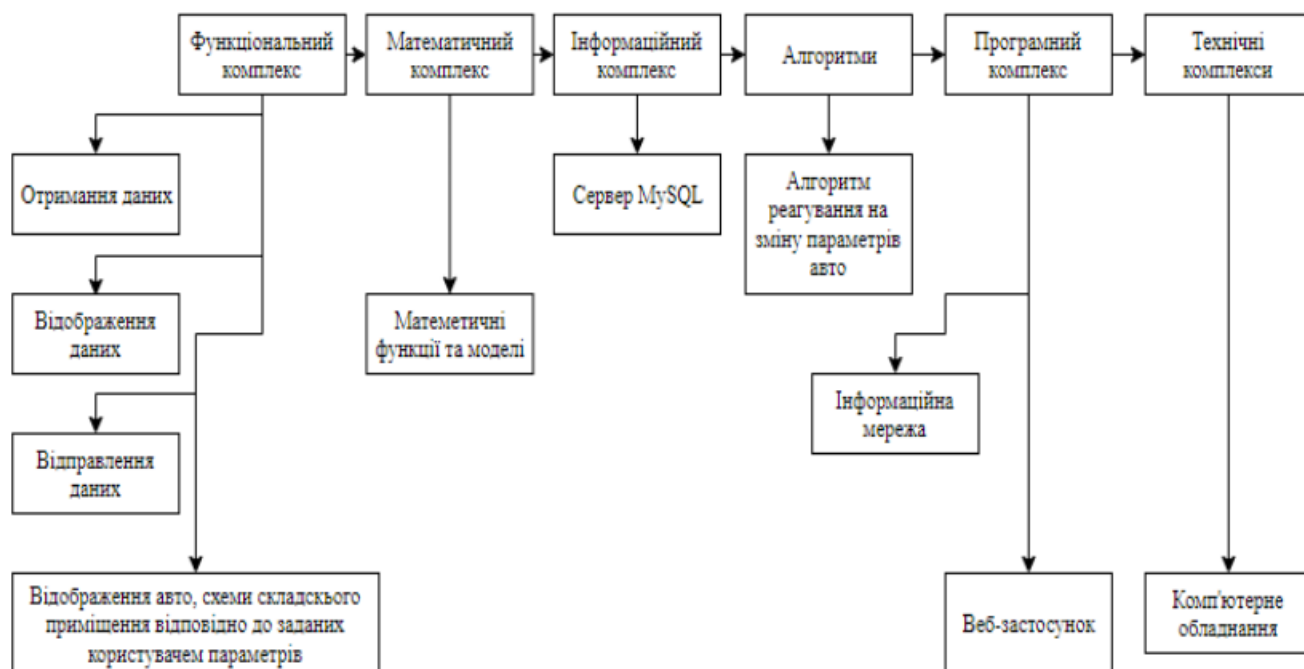


Рисунок 2.3 – Загальна схема автоматизованої системи [1]

Функціональний комплекс відповідає за основні задачі веб-застосунку, а саме: отримання, відображення та відправлення даних, авто, схеми складського приміщення за відповідними параметрами користувача. Інформаційний комплекс представляє собою сховище даних, з якими працює система (наприклад, база даних – MySQL) [19].

За функціонування та злагоджену роботу програми відповідає алгоритмічний блок, основна функція якого – реагування на зміну вхідних параметрів. Основною структурою програмного комплексу є розроблений програмний засіб (додаток) та інформаційна мережа [17].

2.3. Розробка системи навігації транспортного засобу

Система навігації запропоновано нами автоматичного транспортного засобу базується на принципі радіочастотної ідентифікації. Дана система складається зі зчитуючого пристрою та пасивного трансподера (відомого як RFID-мітка або тег).

На рис. 2.4 схематично зображено поверхню для руху транспортного засобу, вона представляє собою двовимірний масив, що складається елементів розміщених на певній відстані один від одного.

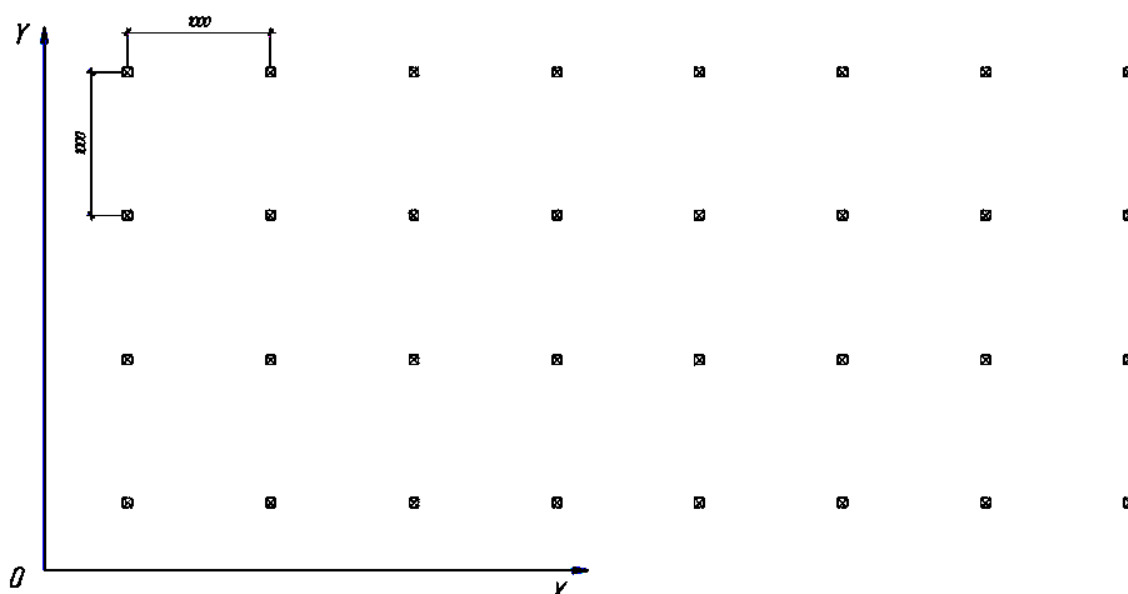


Рисунок 2.4 – Поверхня для руху автоматичного транспортного засобу

Кожен елемент, в свою чергу, представляє собою сукупність пасивних трансподерів (або RFID тегів), що монтуються в отвір у поверхні для руху (рис. 2.5).

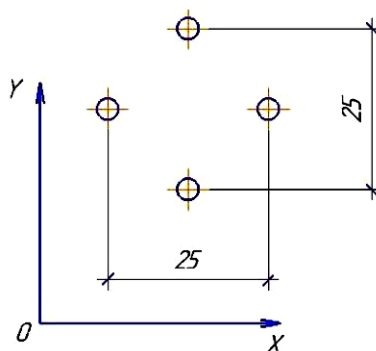


Рисунок 2.5 – Принцип розташування пасивних трансподерів

Така схема руху транспортного засобу дозволяє підвищити точність, збільшити швидкість і тим самим збільшити продуктивність системи без значних фінансових вкладень порівняно з традиційними рішеннями.

Принцип роботи автоматичного транспортного засобу з системою навігації (рис. 2.6, а): зчитуючий пристрій (5) випромінює сигнал з частотою 13,56 МГц, пасивний трансподер (4) індукуює сигнал зчитуючого пристрою, тим самим отримує струм для функціонування кремнієвого чіпу та відправляє сигнал у відповідь з унікальним номером ідентифікації.

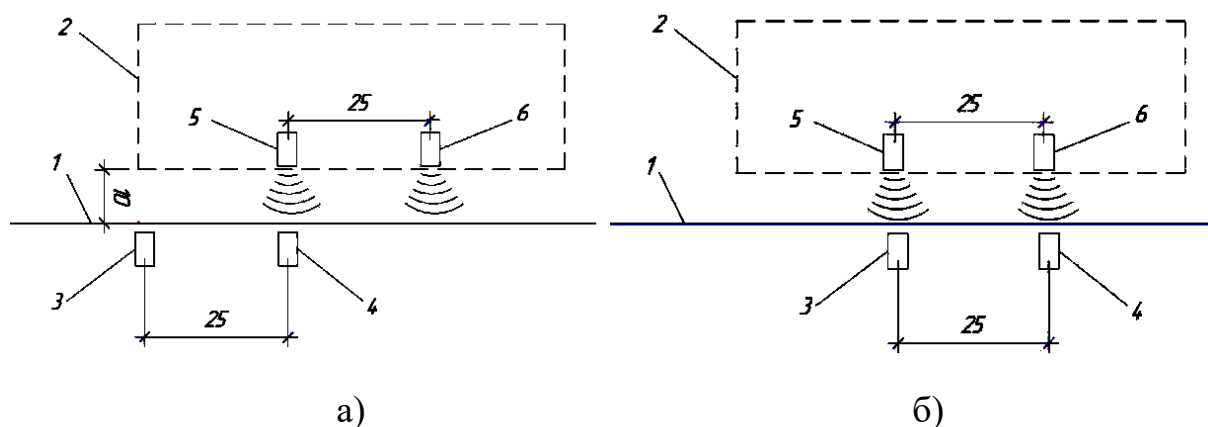


Рисунок 2.6 – Принцип роботи системи навігації: 1 – поверхня для руху; 2 – автоматичний транспортний засіб; 3 і 4 – пасивні трансподери; 5 і 6 – зчитуючі пристрої

Якщо номер RFID-тега відповідає кінцевій точці руху, транспортний засіб знижує свою швидкість на 75%. Коли обидва зчитуючі пристрої (5 та 6) (рис. 2.7, б), отримують сигнал мітки (3 та 4), що рівний заданому, автоматичний транспортний засіб зупиняється та виконує запрограмовані функції маневрування, такі як поворот або підйом вантажу.

2.4. Модель системи управління рухом автономного транспортного засобу товарного складу

На першому етапі побудови комп'ютерної моделі руху транспортного засобу щодо заданої траси вирішуються два завдання: перше – відображення траси, заданої у вигляді відхилень, у відповідну криву координатної площини XY ; друга – розрахунок положення транспортного засобу в площині XY щодо точки на траєкторії з урахуванням бічного відхилення від неї.

На підставі отриманого раніше математичного опису руху транспортного засобу розроблена комп'ютерна модель в програмі Simulink системи MATLAB, представлена на рис. 2.7, яка дозволяє будувати траєкторію руху транспортного засобу в площині XY при заданих значеннях інтервальних параметрів.

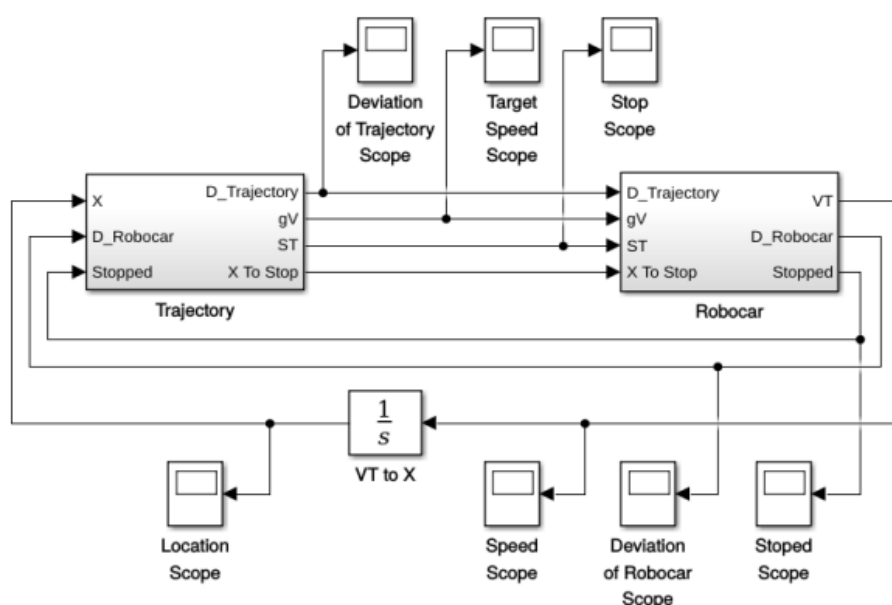


Рисунок 2.7 – Модель руху транспортного засобу в програмі Simulink

Блок "Robocar" в представленій моделі реалізує структурну схему узагальненої моделі системи управління рухом транспортного засобу (рис. 2.8).

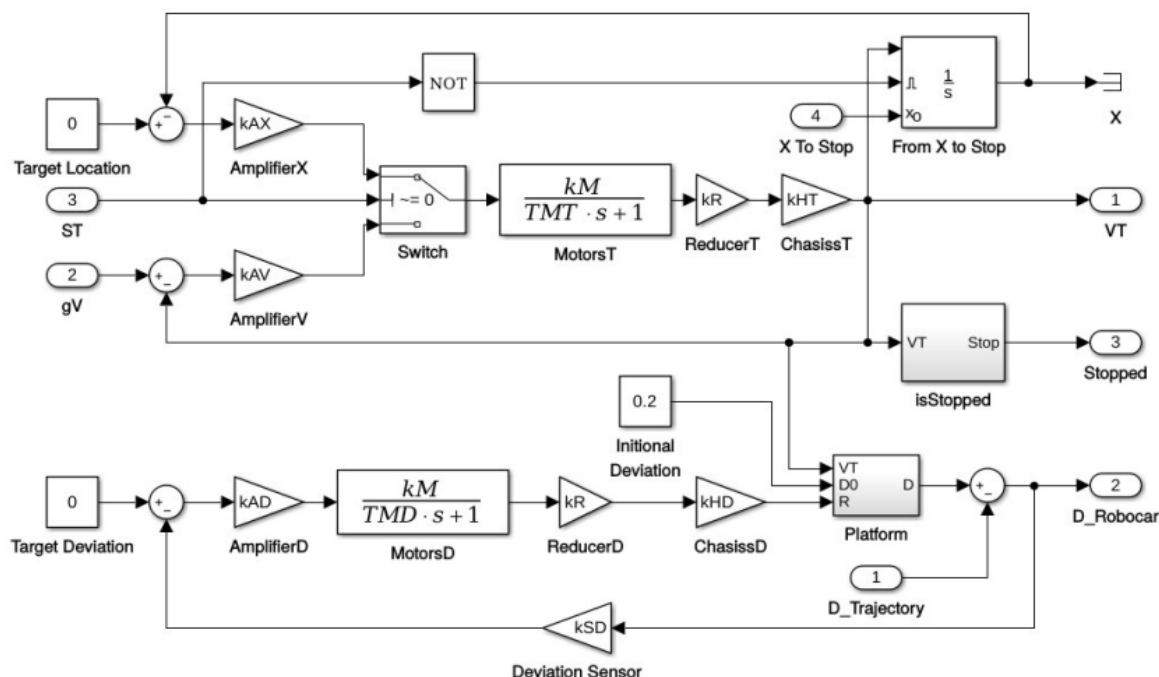


Рисунок 2.8 – Програмна реалізація блоку " Robocar»

Початковими сигналами блоку є відхилення траєкторії від прямолінійного напрямку, задана швидкість поздовжнього руху, значення сигналу ST і відстань до точки зупинки в місці навантаження або розвантаження транспортного засобу. Вихідні сигнали блоку: поточна швидкість поздовжнього руху, бічне відхилення транспортного засобу від траси і сигнал, що приймає значення, рівне одиниці, при повній зупинці транспортного засобу і нульове значення в іншому випадку.

Вихідні сигнали блоку "Robocar" формують вхідні сигнали блоку "Trajectory" (рис. 2.9), який, в першу чергу, призначений для обчислення точок заданої траси в горизонтальній площині XY і визначення положення транспортного засобу в кожен момент модельного часу відносно заданої траси.

Управління двигунами здійснюється за рахунок подачі ШІМ-напруги на якірну обмотку при постійній номінальній напрузі на обмотці збудження. Передача крутного моменту від валів двигунів до коліс здійснюється через черв'ячні універсальні одноступеневі редуктори. Зазначені редуктори є понижувальними і призначені для збільшення крутного моменту за рахунок зменшення частоти

обертання вихідного валу. Пристрій спряження, призначений для узгодження контролера з силовою частиною управління двигунами постійного струму, що являє собою розроблену друковану плату з розташованими на ній польовими транзисторами VT1, VT2 з ізольованим затвором, відкриття/закриття яких забезпечує подачу керуючих ШІМ імпульсів на електродвигуни. Принципова схема пристрою спряження представлена на рис. 2.9.

Напівпровідникові діоди VD1, VD2 застосовані в схемі, виконують роль шунта, організовуючи електричний ланцюг для протікання струмів, породжених ЕРС самоіндукції. Контролер управляє подачею імпульсів на двигуни коліс шляхом зміни сигналів на виходах Q0.0 і Q0.1. Датчики кутової швидкості коліс підключені до дискретних входів і 0.1 і І 0.6, які налаштовані на апаратні програмнонезалежні швидкісні лічильники.

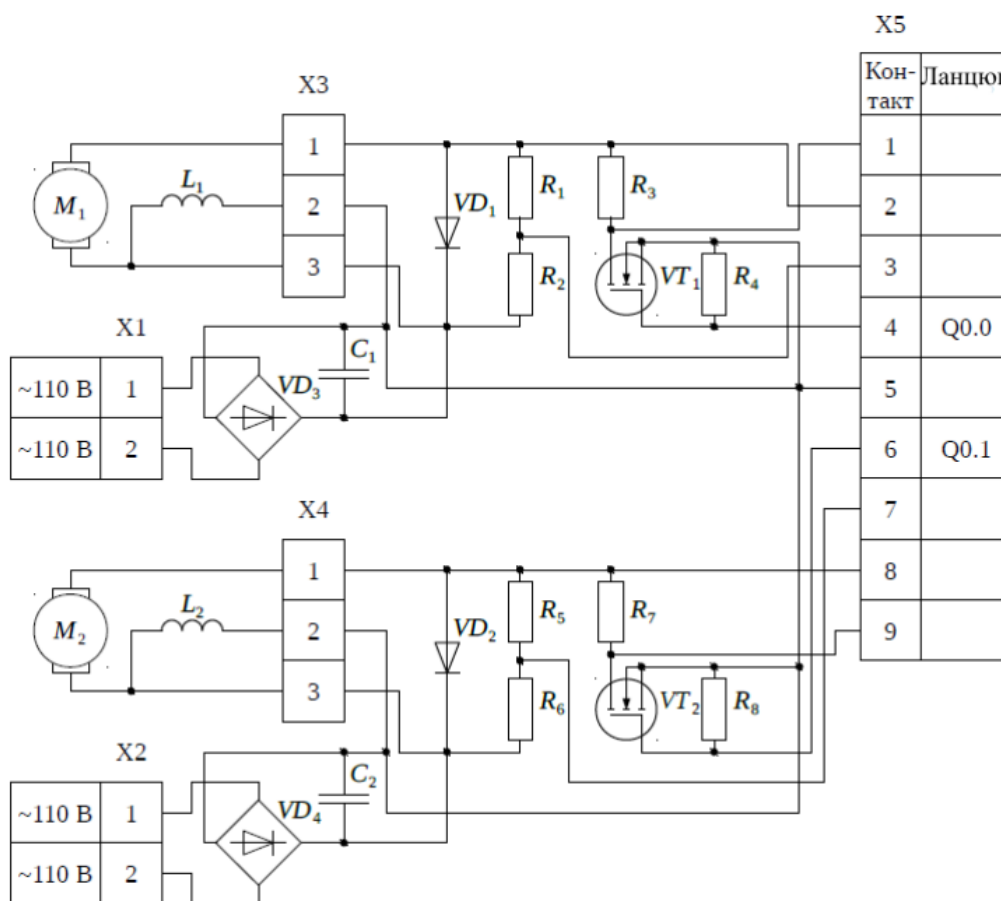


Рисунок 2.9 – Принципова схема пристрою сполучення контролера і двигунів

Виходячи з розглянутого математичного опису мобільного робота і конструкції експериментального стенду, розглянемо можливість змішаного

моделювання руху транспортного засобу з використанням програмного забезпечення, що реалізує математичний опис його динаміки [13]. При цьому елементи автоматизованої системи управління, такі як підсилювальний пристрій, двигуни, редуктори фізично реалізуються на експериментальному стенді.

З іншого боку, формування величини бічного відхилення від траси і помилку неузгодженості необхідно виробляти програмно на основі математичних залежностей, що враховують кутові швидкості обертання кругових платформ, на які спираються колеса транспортного засобу. Такий поділ моделі на фізичну і математичну складові породжує наступні потоки даних між автоматизованим візком, ПК-сервером верхнього рівня і ПК-клієнтами верхнього рівня (рис. 2.10).

Формування потоків даних відбувається відповідно до функціональної схеми, з урахуванням того, що бічне відхилення формується на підставі початкового положення транспортного робота і швидкостей обертання коліс, які отримуються від датчиків швидкостей обертання кругових платформ, причому бортовий контролер виконує, в тому числі, функції по вводу-виводу сигналів.



Рисунок 2.10 – Основні потоки даних між підсистемами гібридної моделі

Технічний опис може бути реалізовано у вигляді скрипту, модуля 80 або підпрограми, що запускається на сервері верхнього рівня (рис. 2.11). На основі отриманих швидкостей обертання кругових платформ визначається відстань $x(t)$,

пройдена мобільним роботом, і бічне відхилення $\Delta(t)$ від точки на траєкторії, що відповідає розрахованому відстані.

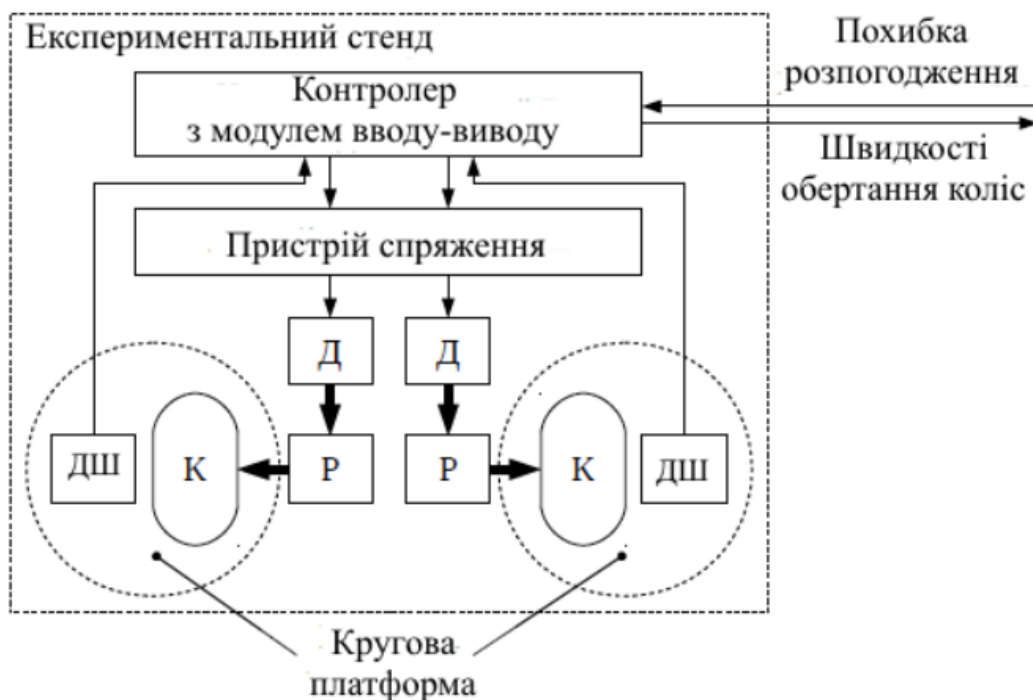


Рисунок 2.11 – Формування потоків даних експериментальним стендом

Знайдені величини повністю характеризують точку поточного положення мобільного робота щодо кінематичної траєкторії. Сукупність послідовно отриманих точок дає віртуальну траєкторію, по якій робот би рухався, перебуваючи на підлозі приміщення, тобто коли колеса не спиралися б на обертові кругові платформи. При цьому бічне відхилення має розраховуватися з урахуванням зміщення $\Delta_0(t)$ заданої траєкторії від прямолінійного руху мобільного робота.

3. ПРОЕКТУВАННЯ ПРОГРАМНО-АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ СКЛАДУ

3.1. Проектування блоку управління

Проаналізувавши вимоги до системи, вибравши контури регулювання, розроблено функціональну схему автоматизованого візка, який працюватиме між виробничим цехом та складським приміщенням (рис. 3.1). Цей автоматизований транспортний засіб складається з рами та набору виконавчих і керуючих елементів, а також панелі оператора.

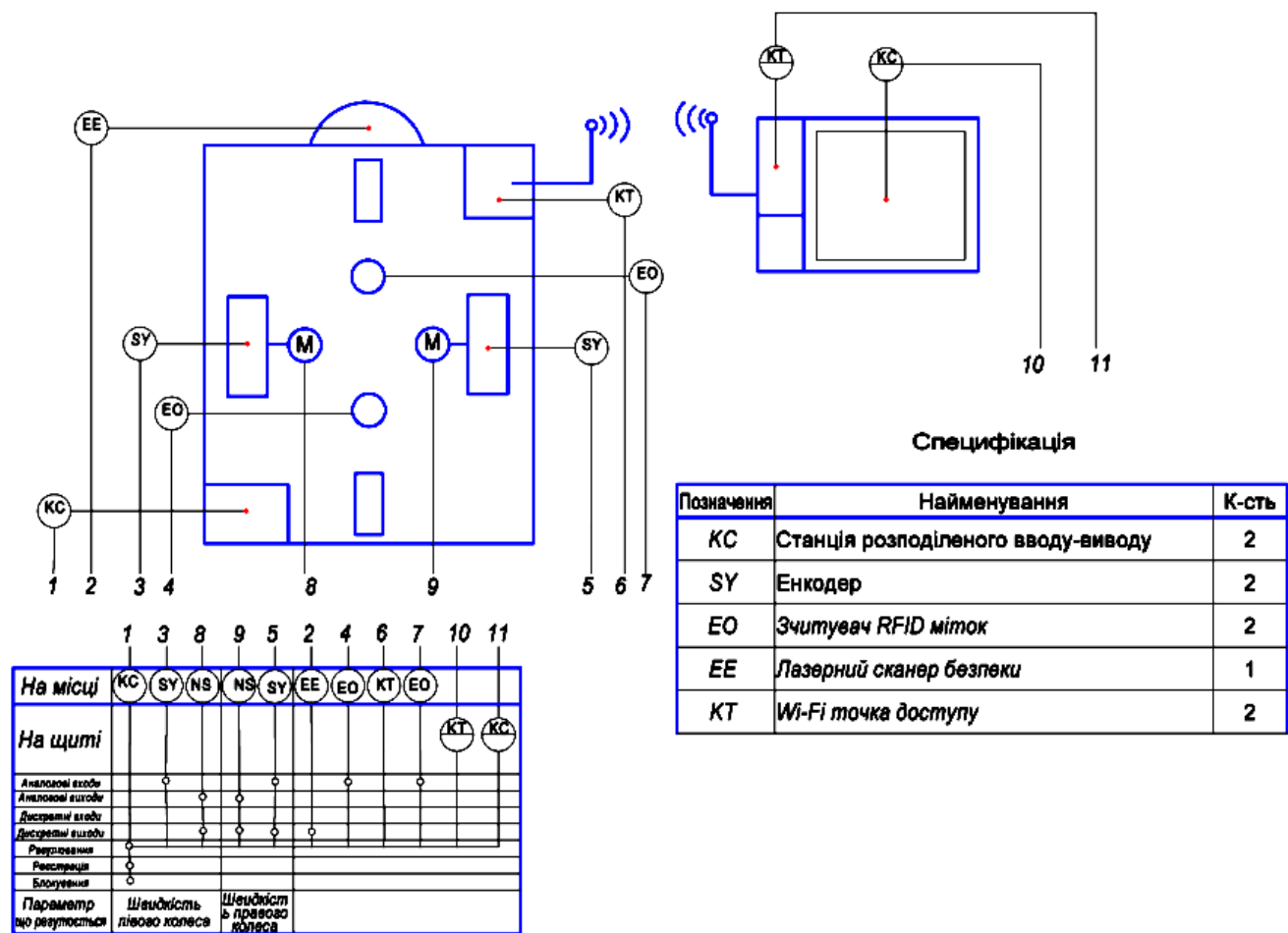


Рисунок 3.1 – Функціональна схема автоматизованого візка

Для автоматизації даного процесу передбачено: 1) контур регулювання швидкості лівого колеса, що складається з енкодера (3), силового драйвера та

мотору (8); 2) контур регулювання швидкості правого колеса, що складається з енкодера (5), силового драйвера та мотору(9).

Система навігації транспортного засобу базується на принципі радіочастотної ідентифікації. Використовуються два зчитувачі (4 та 7). Для зв'язку станції вводу-виводу (10) та ведучого програмованого логічного контролера (1) використовуються безпроводні маршрутизатори (6 та 11).

Для реалізації цього завдання спроектовано мікропроцесорний блок управління транспортним засобом, структурна схема якого наведена на рис. 3.2.

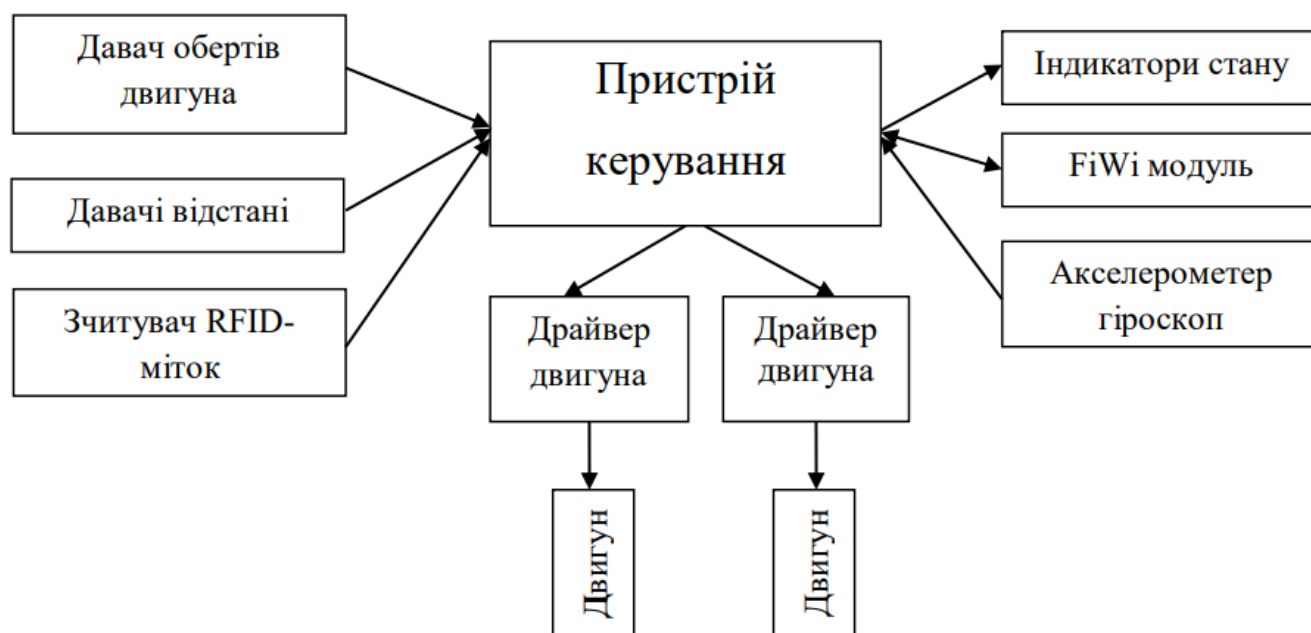


Рисунок 3.2 – Структурна схема блока управління

Система управління є дворівневою:

I рівень – представляє мікропроцесорний блок управління (МБУ);

II рівень – автоматизоване робоче місце (АРМ), тобто ПК оператора.

Використана плата управління наведена на рис. 3.3.

Вибір мікроконтролера. Технічна реалізація автоматизованої системи керування виконана на базі мікроконтролерної плати Arduino Mega 2560, основним елементом якої є мікроконтролер ATmega2560. Вибір цієї плати обґрунтований її низькою вартістю, наявністю необхідної кількості каналів введення/виведення та комунікаційних портів, низьким енергоспоживанням, простотою розробки програмного забезпечення.

На основі плати Arduino Mega 2560 розроблено мікропроцесорний блок управління, який має рідкокристалічний дисплей та клавіатуру [15].

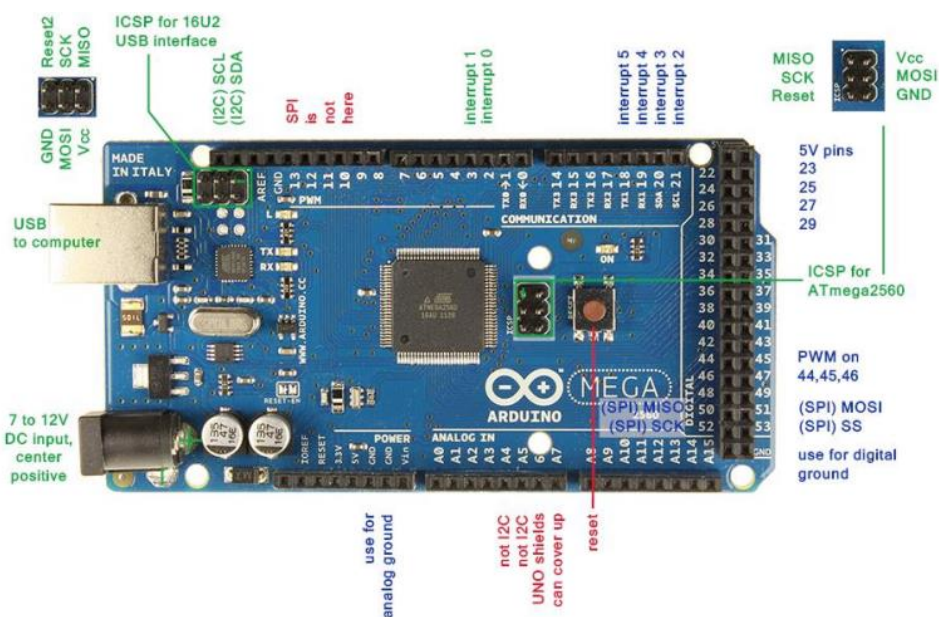


Рисунок 3.3 – Зовнішній вигляд Arduino Mega 2560

Вибір датчика та стабілізатора напруги. Інфрачервоний далекомір дозволяє визначати відстань до об'єктів [18]. Це модель GP2Y0A021 компанії Sharp. Сенсор визначає відстань по відбитому променю світла в інфрачервоному спектрі. Далекомір може використовуватися для об'їзду перешкод і орієнтування на місцевості.

Напруга живлення: 4,5-5,5 В; Струм: 30-40 мА; Діапазон відстаней: 10-80 см.



а)



б)

Рисунок 3.4 – Інфрачервоний далекомір (а) та стабілізатор напруги (б) [14]

Основні характеристики стабілізатора напруги LM7805CT: 1) вихідний струм – до 1 А ;2) вихідна напруга – 5В; 3) тепловий захист від перевантаження; 4) захист від короткого замикання.

Модуль WiFi та Модуль MPU-6050. WiFi-модуль для Arduino — це електронний компонент, який забезпечує бездротове з'єднання мікроконтролера з інтернетом або локальною мережею через Wi-Fi [20]. Він використовується для реалізації IoT-проектів (Інтернет речей), де необхідна передача даних без дротів — наприклад, для розумного дому, бездротових сенсорів, віддаленого моніторингу або керування пристроями.

Одним із найпопулярніших модулів такого типу є ESP8266 (наприклад, модулі ESP-01, ESP-12) або ESP32, які сумісні з Arduino IDE і можуть працювати як самостійні контролери або як розширення для плати Arduino. З їхньою допомогою можна підключати Arduino до Wi-Fi мережі, отримувати дані з інтернету, передавати показники з датчиків на сервер або створювати веб-інтерфейси.

Основні характеристики модуля ESP8266 ESP-01 включають робочу частоту 2.4 ГГц, підтримку стандартів бездротової передачі даних IEEE 802.11 b/g/n, інтерфейс підключення UART (послідовний), напругу живлення 3.3 В, а також кілька режимів роботи: WiFi-клієнт, точка доступу або обидва одночасно (STA+AP). Модуль підтримує основні мережеві протоколи, зокрема TCP/IP, HTTP, MQTT та інші.

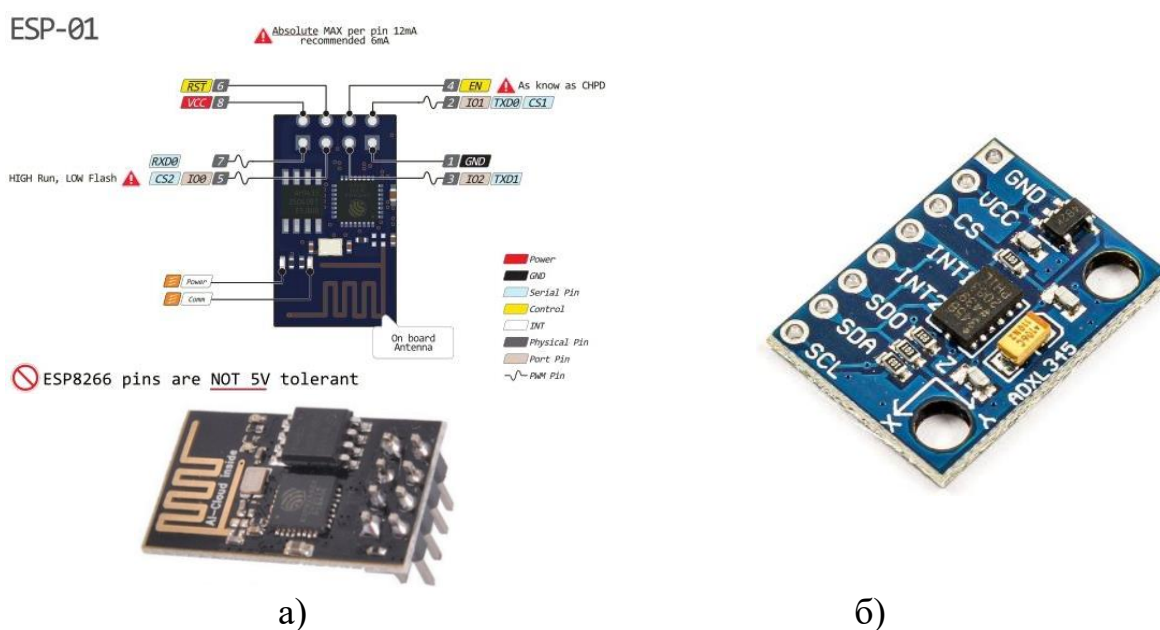


Рисунок 3.5 – WiFi-модуль ESP8266 ESP-01 (а) та Модуль MPU-6050 (б)

Модуль MPU-6050 компактний та легкий модуль 3-х осьового акселерометра і 3-х осьового гіроскопа, керований по протоколу I2C (TWI). Даний датчик відмінно підходить для визначення положення в просторі.

Модуль MPU-6050 має 6 ступенів вільності, 3 для акселерометра та 3 для гіроскопа. Можна відслідковувати прискорення та повороти. Характеристики наступні: 1) живлення: 3...5 В; 2) інтерфейс – I2C; 3) 16-бітний АЦП; 4) точність гіроскопа: 250, 500, 1000, 2000 °/с.

RFID модуль 125 кГц та Мотор-редуктор RS-90ZY-24-3600 [19]. Модуль RDM6300 призначений для зчитування кодів із RFID карток та брелків. Працюють на частоті 125 кГц, призначені для зчитування безконтактних карток та ідентифікаторів і широко використовуються в системах контролю доступу. Вони мають частоту роботи 125 кГц, живляться від 5 В, використовують інтерфейс UART TTL зі швидкістю передачі 9600 бод, забезпечують ефективну відстань зчитування до 5 мм та мають компактні розміри $38.5 \times 19 \times 9$ мм.

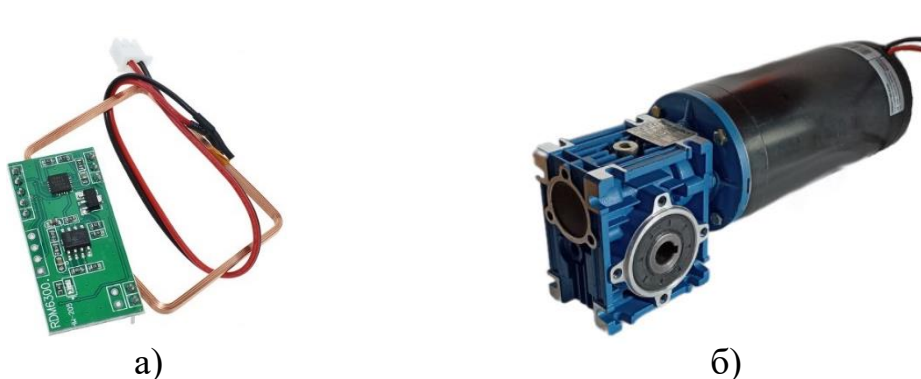


Рисунок 3.6 – Модуль зчитування кодів RFID (а) та Мотор-редуктор RS-90ZY-24-3600 (б)

Необхідний електродвигун повинен мати зусилля на валу 6.3 кг м. Пропонується у проекті використати мотор-редуктор з вбудованим енкодером китайської фірми Dake motor. Із заводу даний мотор-редуктор обладнується енкодером з магнітним принципом дії, розширення енкодера 64 імп./оберт, загальне розширення 192 000 імп./оберт.

Енкодер – це пристрій, який визначає кут повороту обертового об'єкта, видаючи імпульсний цифровий код. Використовується для визначення швидкості

обертання валу (осі), коли немає потреби зберігати абсолютний кутове положення при виключенні живлення. Тобто, якщо вал нерухомий, передача імпульсів припиняється. Іншими словами, якщо включити енкодер цього типу, то відлік повороту кута почнеться з нуля, а не з кута на який він був виставлений до моменту виключення. Магнітний енкодер включає в себе вал з магнітом і двома датчиками Холла, які реєструють послідовність проходження магнітних полюсів (північні і південні), вимірює швидкість і напрямок обертання.

3.2. Результати проектування силового драйвера для автоматизованого віка

Н-міст – це проста схема, що складається з чотирьох ключів з навантаженням між ними (рис. 3.7) [2].

Q1 ... Ключі Q4 зазвичай біполярні або польові транзистори. Якщо схема працює в високовольтних мережах, то можуть використовуватися IGBT транзистори.

Крім можливості зібрати міст з дискретних елементів, існують ще й інтегральні рішення (весь міст в одній мікросхемі, та до того ж ще і керуюча логіка там же).

Діоди D1 ... D4 називаються обмежувальними діодами і найчастіше є діодами Шотткі. У загальному випадку всі чотири ключа в схемі

можуть бути незалежно переведені в стан «включено» або «вимкнено» (відповідно, транзистори відкриті або закриті). Це накладає ряд цілком очевидних обмежень. Принципову схему силового драйвера для автоматичного транспортного засобу було розроблено в середовищі програми KiCad (рис. 3.8).

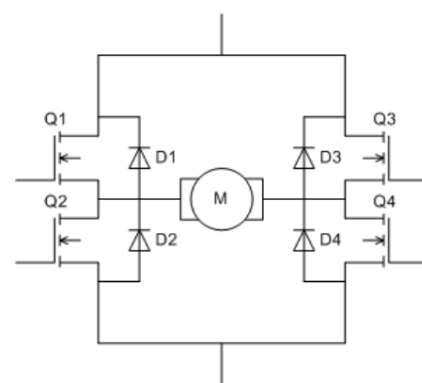


Рисунок 3.7 – Н-міст'

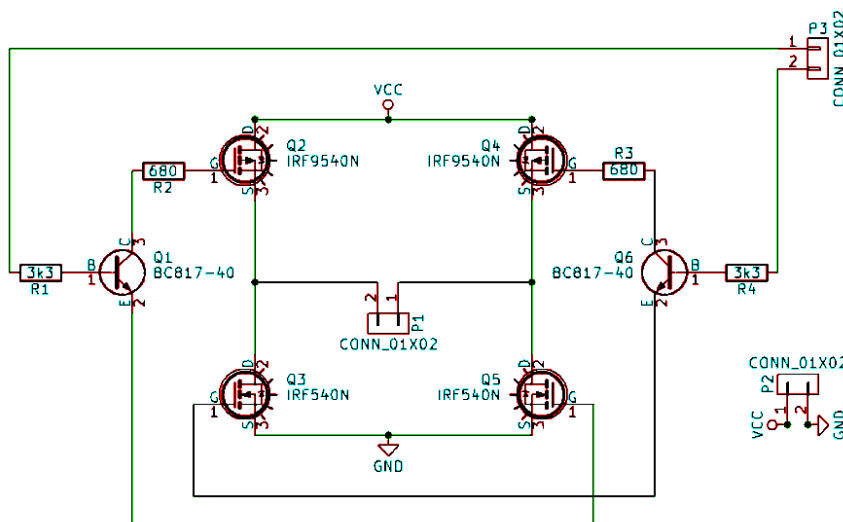


Рисунок 3.8 – Принципова схема силового драйвера

Н-міст виконаний на транзисторах Q2, Q3, Q4, Q5, а за допомогою додаткових транзисторів Q1 і Q6 реалізовано об'єднане управління синфазними ключами. Сигнал А управляє транзисторами Q2 і Q5, а сигнал В — транзисторами Q4 і Q3.

Принцип роботи схеми наступний. Коли рівень сигналу А стає високим — починає текти струм через резистор R1 і р-п переходи база-емітер (БЕ) транзисторів Q1 і Q5. Ці транзистори відкриваються, в результаті чого з'являється струм через перехід БЕ транзистора Q2, резистор R2 і відкритий транзистор Q1. Це призводить до відкриття транзистора Q2.

Коли рівень сигналу А стає низьким — р-п переходи БЕ транзисторів Q1 і Q5 замикаються, транзистори закриваються, і струм через БЕ перехід транзистора Q2 припиняється. Як наслідок, транзистор Q2 також закривається.

Розрахунок елементів схеми. Напруга живлення становить 24 В, максимальний струм двигуна — 25 А, сигнал управління — 5 В (логічний рівень "1" відповідає 5 В, "0" — близько 0 В).

Спочатку обираються транзистори Q2, Q3, Q4, Q5. Вони повинні витримувати напругу 24 В і струм 25 А. Підходять IRF540N (N-канальні) і їх комплементарна пара IRF9540N (P-канальні). Ці MOSFET-транзистори розраховані на струм до 33 А, напругу до 100 В і мають коефіцієнт посилення 40.

Визначення струму керування транзисторів Q2 і Q5: $33 / 100 = 33 \text{ мА}$.

Розрахунок резистора R2. Вважається, що на кожному з трьох р-п переходів (БЕ транзисторів Q2, Q4 і відкритого Q1) падає по 0,5 В:

$$(24 - 3 \times 0,5) / 33 = 681 \text{ Ом.}$$

Вибирається найближче менше стандартне значення: 680 Ом. Тоді струм керування буде $(24 - 1,5) / 680 = 33 \text{ мА}$, а розсіювана потужність на резисторі: $U^2/R = 847 \text{ мВт}$.

Отже, резистор має бути розрахований на потужність не менше 1 Вт.

Транзистор Q1 повинен витримувати напругу 24 В і струм 33 мА. Обрано BCB17-40 (45 В, 0,8 А, коефіцієнт підсилення $h_{FE} \geq 30$).

Розрахунок струму керування Q1: $33 \text{ мА} / 30 = 1,1 \text{ мА}$.

Розрахунок резистора R1. З урахуванням падіння 0,5 В на кожному БЕ переході (Q1 і Q5):

$$(5 - 2 \times 0,5) / 1,1 = 3,6 \text{ кОм.}$$

Вибрано найближче менше стандартне значення: 3,3 кОм.

У такому випадку струм буде 1,2 мА, а потужність, що розсіюється: 12,1 мВт — можна використовувати стандартний резистор на 0,125 Вт.

Силові ключі верхнього плеча Н-моста реалізовані на Р-канальних MOSFET транзисторах IRF9540N виробництва Infineon Technologies, що керуються логічними рівнями TTL.

Силові ключі нижнього плеча побудовані на N-канальних MOSFET транзисторах IRF540N, які також підтримують TTL-рівень керування.

3.3. Результати розробки програмного забезпечення для мікроконтролерної плати

Програмне забезпечення розробленої системи автоматизованого керування візком складається з програми для плати Arduino Mega 2560 та програми управління на ПК. Програмне забезпечення для мікроконтролера розроблене в

середовищі Arduino IDE мова програмування C з елементами C++. В програмі використовується лише одна бібліотека для роботи з пам'яттю:

```
#include "EEPROM.h"

// Функція setup() — ініціювання входів/виходів та
// параметрів мікроконтролера
void setup() {
    Serial.begin(9600);           // Ініціалізація серійного порту
    pinMode(HORN, OUTPUT);        // Вихід на сигнал клаксона
    pinMode(IN1, OUTPUT);         // Вихід IN1 для керування мотором
    pinMode(IN2, OUTPUT);         // Вихід IN2 для керування мотором
    pinMode(IN3, OUTPUT);         // Вихід IN3 для другого каналу
    pinMode(IN4, OUTPUT);         // Вихід IN4 для другого каналу
    pinMode(EN1, OUTPUT);         // Вихід керування швидкістю (енабл)
    pinMode(INM1, OUTPUT);        // Додатковий канал керування
    pinMode(INM2, OUTPUT);        // Додатковий канал керування
    pinMode(trigPin, OUTPUT);     // Вихід для ультразвукового сенсора (тригер)
    pinMode(echoPin, INPUT);      // Вхід для ультразвукового сенсора (відлуння)
    timer_init();                 // Ініціалізація таймера
}
```

Функції `void loop()` це основна функція програми яка виконується циклічно і в ній описується основні операції мікроконтролера.

Блок схема основної функції `void loop()` наведена на рис. 3.9.

Код програми для мікропроцесорної плати Arduino Mega 2560 написаний мовою C наведений в Додатку А.

Цей код реалізує керування роботизованим транспортним засобом (візком) із чотирма приводними колесами (4WD), оснащеним звуковим сигналом (сиреною) та ультразвуковим датчиком відстані, на основі Arduino.

Основне призначення програми — приймати команди з комп'ютера через послідовний порт (Serial), обробляти їх, керувати рухом лівого та правого моторів, активувати звуковий сигнал і здійснювати вимірювання відстані до перешкоди. Також реалізовано механізм автоматичного вимкнення приводу після встановленого інтервалу бездіяльності (autoOFF), який зчитується або зберігається у пам'яті EEPROM.

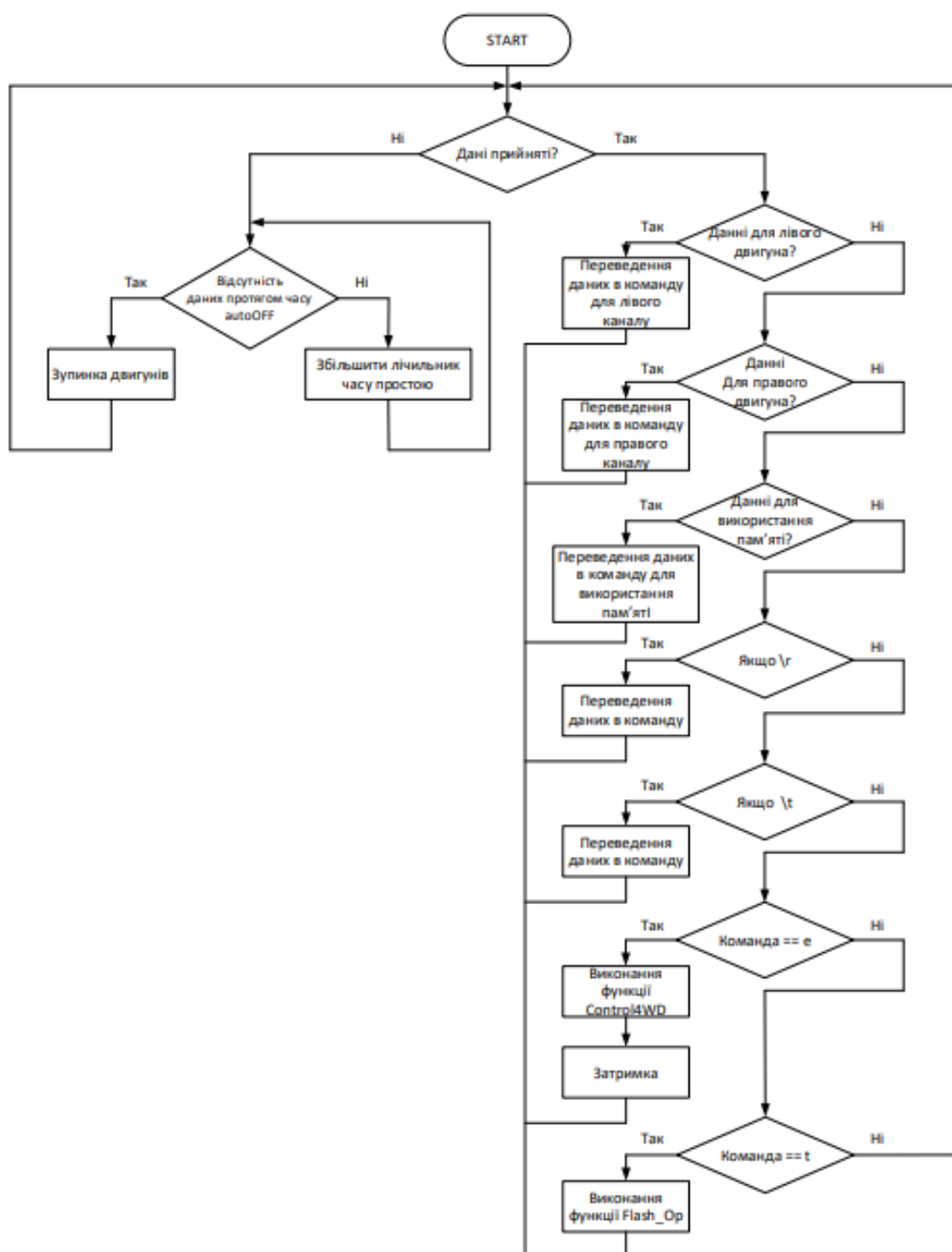


Рисунок 3.9 – Блок схема основної функції void loop() програми для мікропроцесорної плати Arduino Mega 2560

Завдання головних функцій у розробленій програмі:

setup() — ініціалізує пінові виходи, серійне з'єднання та зчитує параметри з EEPROM.

`loop()` — постійно перевіряє вхідні дані з порту та виконує відповідні дії: керування моторами, збереження/зчитування параметрів, відлік часу без активності.

`Control4WD()` — реалізує керування швидкістю та напрямком обертання лівого і правого моторів на основі отриманих значень.

`Flash_Op()` — дозволяє зчитувати або записувати параметри конфігурації (наприклад, час автоматичного вимкнення) в EEPROM.

`echo()` — вимірює відстань до об'єкта перед датчиком за допомогою ультразвукового сенсора HC-SR04.

`timer_init()` — ініціалізує таймер автоматичного вимкнення згідно з конфігурацією EEPROM.

У програмі реалізовано збереження параметрів у пам'яті EEPROM, що дозволяє зберігати налаштування навіть після перезапуску пристрою. Керування роботою здійснюється через просту систему команд, які надсилаються у вигляді символів (наприклад, 'L', 'R', 'F', 'H'), завдяки чому можна розділяти логіку дій і чітко інтерпретувати введені дані. Вбудовано захист від зіткнень: якщо ультразвуковий датчик фіксує перешкоду на відстані 20 см або менше, рух відповідного мотору блокується, що підвищує безпечність керування. Обмін даними здійснюється через послідовний порт (UART), що забезпечує простоту взаємодії з комп'ютером або іншим контролером. Такий підхід дозволяє ефективно керувати рухом робота та налаштуваннями в реальному часі.

3.4. Передача даних по Wi-Fi каналу із управління роботизованим візком

Управління роботизованим транспортним засобом (візком) здійснюється наступним чином — з виробничого цеху приходить замовлення щодо переміщення продукту. Це замовлення обробляється, визначається ідентифікатор кожного товару в замовленні, розраховується маршрут автоматизованого візка.

Далі дані передаються по Wi-Fi каналу на контролер автоматизованого візка, який в свою чергу через драйвер двигунів управляє приводами. На ПК формуються команди переміщення засобу залежно від даних з датчиків та положення інших роботизованих візків (рис. 3.10).

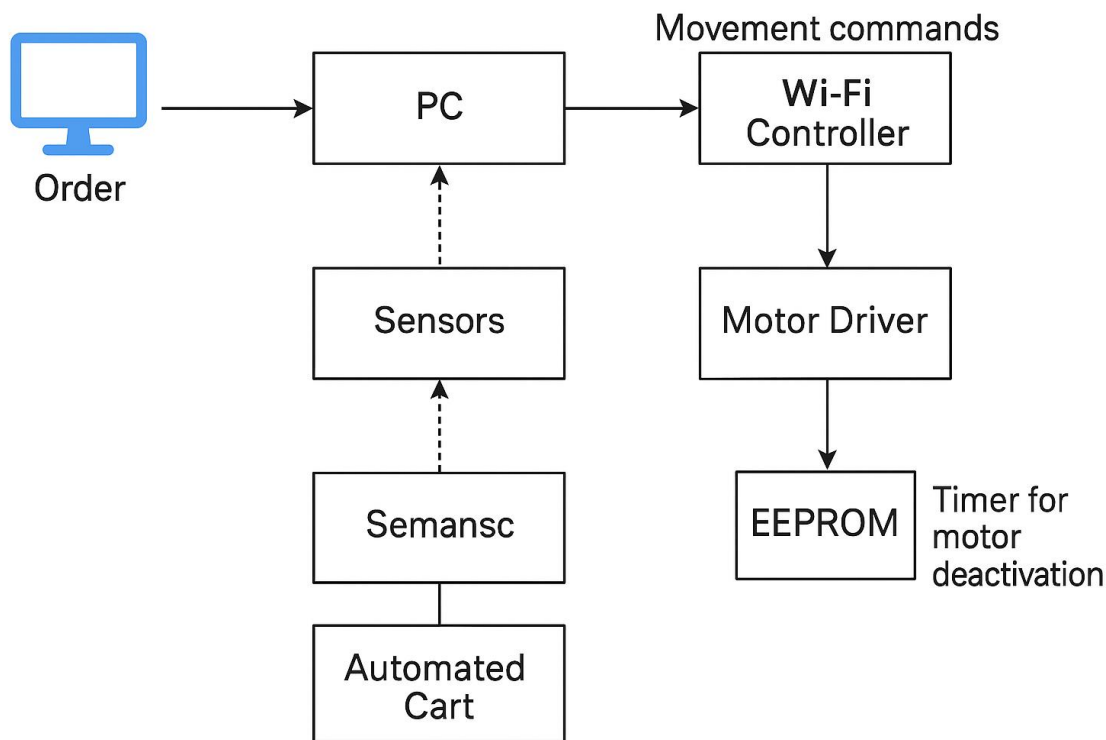


Рисунок 3.10 – Загальна схема логіки керування роботизованим візком через Wi-Fi

Всі розрахунки проводяться на ПК, і відразу ж обчислюються значення ШІМ для лівого і правого двигунів. Через Wi Fi передаються команди виду:

L-255 \ rR-120 \ r

L – команда для лівого двигуна,

R – для правого;

“-“ позначає обертання двигунів для руху назад;

255 - число ШІМ, для Arduino це максимальна швидкість обертання

\ R - кінець команди.

За цією командою платформа буде рухатися назад і трохи повертати в праву сторону, тому правий двигун буде обертатися повільніше лівого.

L255 \ rR-255 \ r

За цією командою лівий двигун буде обертатися вперед, а правий – назад, що змусить платформу робота обертатися навколо своєї осі проти годинникової стрілки.

У програмі контролера передбачений таймер, який відключає двигуни, якщо остання команда була отримана більш, ніж n-секунд тому. Встановлення кількості секунд зберігається в EEPROM пам'яті контролера. Для роботи з пам'яттю мікроконтролера передбачені команди Fr – читання значень і Fw – запис значень.

Код програми для мікропроцесорної плати Arduino Mega 2560 написаний мовою C наведений в Додатку Б.

Отже, команди з ПК: передаються у вигляді $L \pm \text{value} \backslash r$, $R \pm \text{value} \backslash r$, Fr, FwNNNN.

Fr: зчитує поточний таймаут з EEPROM.

FwNNNN: записує новий таймаут (у мс) в EEPROM.

stopMotors(): зупиняє обидва двигуни, якщо команда не надходила протягом заданого часу.

PWM (ШИМ): використовується для регулювання швидкості обертання.

EEPROM: зберігає таймаут автоматичного зупинення після втрати зв'язку.

Цей код підходить для базових Wi-Fi роботів, керованих з ПК або з центрального сервера через ESP8266 або ESP32, підключених до Arduino як Wi-Fi-шлюз.

4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Структурно-функціональний аналіз та модель травмонебезпечних ситуацій

У зображеннях процесів формування, виникнення аварій та виробничих травм усі випадкові події, що утворюють конкретну аварійну ситуацію, пов'язані між собою причинно-наслідковими зв'язками.

Метод логічного моделювання потенційних аварій, травм та катастроф відкриває можливість розробити досконалу систему управління ОП виробництва, яка базується на оперативному пошуку виробничих небезпек, їх глибокому аналізу й терміновому прийнятті заходів для усунення потенційних небезпек ще до виникнення травмонебезпечних та катастрофічних ситуацій. Деякі небезпечні ситуації в табл. 4.1.

Працівники, що обслуговують електрообладнання вентильної системи, зобов'язані знати Правила безпечної експлуатації електроустановок споживачів відповідно до займаної посади або роботи, як вони виконують, і мати відповідну групу з електробезпеки [4].

Працівники, що порушили вимоги Правил безпечної експлуатації електроустановок, усуваються від роботи і несуть відповідальність (дисциплінарну, адміністративну, кримінальну) згідно з чинним законодавством. Такі працівники не допускаються до робіт в електроустановках без позачергової перевірки знань вимог правил безпечної експлуатації електроустановок.

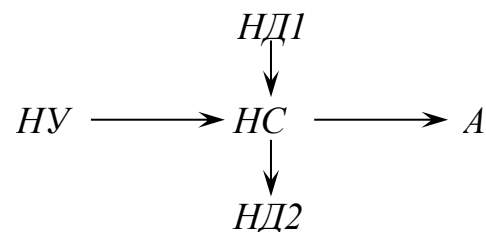
Забороняється допускати до роботи в електроустановках осіб, які не пройшли навчання і перевірку знань Правил безпечної експлуатації електроустановок.

Працівнику, який пройшов перевірку знань Правил безпечної експлуатації електроустановок, видається посвідчення встановленої форми.

Таблиця 4.1. Моделювання процесів виникнення травмонебезпечних і аварійних ситуацій

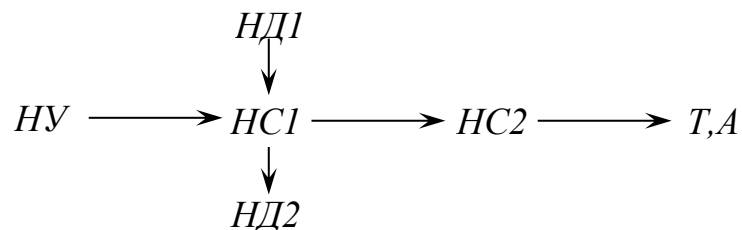
Вид робіт	Виробнича небезпека			Можливі наслідки	Заходи запобігання небезпечним ситуаціям
	Небезпечна умова (НУ)	Небезпечна дія (НД)	Небезпечна ситуація (НС)		
Використання механічної вентиляції	Оператор не перевіряв обладнання НУ	Пошкоджений трубопровід мережі НД1 Закупорений трубопровід шланга НД2	Відмова вентиляційної системи (двигуна) НС	Аварія	Розвісити плакати, провести інструктажі із експлуатації обладнання системи

Модель процесу:



Використання електронних пристроїв регулювання	Пошкоджена ізоляція провідників з'єднання НУ	Пробій на корпус НД1 Коротке замикання НД2	Ураження людини електричним струмом НС1 Виведення обладнання із ладу НС2	Травма Аварія	Заміна провідників, установлення захисного обладнання (запобіжників, захист від ураження людини струмом) тощо
--	--	---	---	---------------	---

Модель процесу:



Посвідчення про перевірку знань працівника є документом, який засвідчує право на самостійну роботу в електроустановках на зазначеній посаді за фахом.

4.2. Розрахунок штучного заземлення

Вибір штучного заземлення проводиться в залежності від характеру ґрунту і способу забивання стержнів [4]. Розраховуємо заземлюючий контур підстанції напругою 10/0,4 кВ з глухозаземленою нейтраллю. Характер ґрунту – чорнозем з $\rho = 2 \cdot 10^4$ Ом·см. Кліматична зона – IV ($K_c = 1,2$, $K_n = 1,5$). Струм замикання на землю в мережі становить 50 А.

В відповідності з діючими правилами, опір заземлюючого пристрою повинен становити

$$R = \frac{125}{I_z} = \frac{125}{50} = 2,5 \text{ Ом}, \quad (4.1)$$

де I_z – струм замикання на землю, А.

Приймаємо 3 Ом. Контур заземлення розміщуємо в ряд з $a = 5$ м, $l = 2,5$ м. В якості стержневого заземлювача приймаємо кутникові сталь 50х50х5 мм, а протяжного – пластинчасту сталь 40х4 мм.

Опір одиночного стержня становить:

$$R_o = 0.00318 \rho \cdot K_c, \text{ Ом} \quad (4.2)$$

де K_c – коефіцієнт сезонності для стержневого заземлювача ($K_c = 1,2$).

$$R_o = 0.00318 \cdot 2 \cdot 10^4 \cdot 1.2 = 76.32 \text{ Ом}.$$

Число стержнів приймаємо 15. При цьому коефіцієнт використання стержневих заземлювачів становить $\eta_c = 0,7$. Опір всіх стержнів розтікання струму становить:

$$R_c = \frac{R_o}{n \cdot \eta_c}, \text{ Ом}, \quad (4.3)$$

де n – число стержнів, шт.

$$R_c = \frac{76.32}{15 \cdot 0.7} = 7.3 \text{ Ом}.$$

Довжина протяжного заземлювача становить $l = 35$ м (3500 см); приймаємо $t = 50$ см, $b = 0,4$ см. Опір протяжного заземлювача становить:

$$R_{np} = \frac{0,366}{l} \cdot \rho \cdot 2 \cdot \lg \frac{2 \cdot l^2}{t \cdot b}, \text{ Ом} \quad (4.4)$$

$$R_{np} = \frac{0,366}{3500} \cdot 1,2 \cdot 10^4 \cdot 2 \cdot \lg \frac{2 \cdot 3500^2}{0,4 \cdot 50} = 3,2 \text{ Ом}$$

Коефіцієнт використання протяжного заземлювача $\eta_n = 0,71$. Дійсний опір протяжного заземлення становить:

$$R_n = \frac{R_{np}}{\eta_n} = \frac{3,2}{0,71} = 4,5 \text{ Ом} \quad (4.5)$$

Опір всього заземлюючого пристрою становить:

$$R_u = \frac{R_c \cdot R_n}{R_c + R_n} = \frac{4.5 \cdot 7.3}{4.5 + 7.3} = 2.78 < 3 \text{ Ом} \quad (4.6)$$

Отже, число стержнів вибрано вірно.

4.3. Безпека в надзвичайних ситуаціях

Забезпечення захисту населення і території у разі загрози та виникнення надзвичайних ситуацій є одним з найважливіших завдань не лише підприємства, але й цілої держави. Актуальність проблеми забезпечення природо-техногенної безпеки населення і території зумовлена тенденціями зростання втрат людей і шкоди територіям, що спричиняються небезпечними природними явищами, промисловими аваріями і катастрофами.

Інженерний захист проводиться з метою виконання вимог ІТЗ із питань забудови міст, розміщення ПНО, будівлі будинків, інженерних споруд та інше.

Медичний захист проводиться для зменшення ступеня ураження людей, своєчасного надання допомоги постраждалим та їх лікування, забезпечення епідеміологічного благополуччя в районах надзвичайних ситуацій.

Біологічний захист включає своєчасне виявлення чинників біологічного зараження, їх характеру і масштабів, проведення комплексу адміністративно-господарських, режимно-обмежувальних і спеціальних протиепідемічних та медичних заходів.

Радіаційний і хімічний захист включає заходи щодо виявлення і оцінки радіаційної та хімічної обстановки, організацію і здійснення дозиметричного та хімічного контролю, розроблення типових режимів радіаційного захисту, забезпечення засобами індивідуального захисту, організацію і проведення спеціальної обробки.

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

Склади продукції відіграють ключову роль у виробничому процесі, забезпечуючи безперервність постачання матеріалів та зберігання готової продукції. Вони виступають буферною ланкою між різними етапами виробництва, оптимізують логістику і сприяють ефективному управлінню запасами. Завдяки складам можливо мінімізувати простой обладнання, уникнути перебоїв у виробничому циклі, забезпечити своєчасне виконання замовлень та підвищити загальну гнучкість і конкурентоспроможність підприємства.

На підставі проведеного аналізу підкреслено актуальність підвищення ефективності виробництва за рахунок автоматизації транспортно-складської системи, що є найбільш вузьким місцем у виробництві з точки зору автоматизації, шляхом розробки і застосування автоматизованих систем переміщення виробів мобільними транспортними засобами (візком).

Проаналізовано роботу складу та зокрема системи управління з комп'ютерно-інтегрованою автоматизованою системою, а саме, розглянуто алгоритм визначення показників функціонування багатоканальної системи, при цьому автоматизована система роботи складу не повинна виходити за межі стійкості та забезпечувати безперебійну роботу і мінімізацію витрат.

Проаналізувавши вимоги до системи, вибравши контури регулювання, розроблено функціональну схему автоматизованого візка, який працюватиме між виробничим цехом та складським приміщенням. Цей автоматизований транспортний засіб складається з рами та набору виконавчих і керуючих елементів, а також панелі оператора. Підібрано керуючі та виконавчі елементи для цього автоматизованого транспортного засобу.

Програмне забезпечення розробленої системи автоматизованого керування візком складається з програми для плати Arduino Mega 2560 та програми управління на ПК. Програмне забезпечення для мікроконтролера розроблене в середовищі Arduino IDE мова програмування C з елементами C++. В програмі використовується лише одна бібліотека для роботи з пам'яттю.

Основне призначення програми — приймати команди з комп'ютера через послідовний порт (Serial), обробляти їх, керувати рухом лівого та правого моторів, активувати звуковий сигнал і здійснювати вимірювання відстані до перешкоди. Також реалізовано механізм автоматичного вимкнення приводу після встановленого інтервалу бездіяльності (autoOFF), який зчитується або зберігається у пам'яті EEPROM.

У програмі системи автоматизованого керування візком реалізовано збереження параметрів у пам'яті EEPROM, що дозволяє зберігати налаштування навіть після перезапуску пристрою. Керування роботою здійснюється через просту систему команд, які надсилаються у вигляді символів (наприклад, 'L', 'R', 'F', 'H'), завдяки чому можна розділяти логіку дій і чітко інтерпретувати введені дані.

Вбудовано захист від зіткнень: якщо ультразвуковий датчик фіксує перешкоду на відстані 20 см або менше, рух відповідного мотору блокується, що підвищує безпечність керування. Обмін даними здійснюється через послідовний порт (UART), що забезпечує простоту взаємодії з комп'ютером або іншим контролером. Такий підхід дозволяє ефективно керувати рухом робота та налаштуваннями в реальному часі.

Таким чином, розроблено програмне забезпечення автоматизованої системи керування, що складається з програми для плати Arduino Mega 2560 та програми управління на ПК. Також розроблено програму на мові C щодо передачі даних по Wi-Fi каналу із управління роботизованим візком.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Автоматизація складів: огляд рішень та тенденцій розвитку [URL: https://uit.kiev.ua/](https://uit.kiev.ua/) (дата звернення 07.06.2025)
2. Автоматизація складу: як це працює? [URL: https://www.youtube.com/watch?v=xmJTiRoIsY0](https://www.youtube.com/watch?v=xmJTiRoIsY0) (дата звернення 07.06.2025)
3. Веб-сайти виробників та постачальників складського обладнання: [URL: https://www.jungheinrich.com/en](https://www.jungheinrich.com/en) (дата звернення 07.06.2025)
4. Довідник нормативних документів у сфері охорони праці, пожежної безпеки, гігієни праці та соціального страхування від нещасних випадків / К.: Фонд соціального страхування від нещасних випадків на виробництві та професійних захворювань України, 2009. – 244 с.
5. Документація на мікроконтролер ATmega328P: <https://www.microchip.com/wwwproducts/en/ATmega328p> (дата звернення 07.06.2025)
6. Мехатроніка: навчальний посібник. Ловейкін В.С., Ромасевич Ю.О., Човнюк Ю.В. Мехатроніка. Навчальний посібник. 2012. С. 270-276.
7. Основи автоматики та автоматизації: Навчальний посібник. Пістун Є., Стасюк І. 2018 р. С. 87- 106.
8. Переваги впровадження автоматизованих транспортних систем на складах [URL: https://quantum-software.com/es/](https://quantum-software.com/es/) (дата звернення 07.06.2025)
9. Підбір обладнання для автоматизації складів [URL: https://forstor.ua/ua/customer-reference/skladskaya-logistika/avtomatizaciya-skladakak-sdelat-pervyj-shag/](https://forstor.ua/ua/customer-reference/skladskaya-logistika/avtomatizaciya-skladakak-sdelat-pervyj-shag/) (дата звернення 07.06.2025)
10. Принципи проектування програмного забезпечення для складських систем. [URL: https://eurodnpr.com.ua/avtomatizaciya-skladskogo-ucheta/](https://eurodnpr.com.ua/avtomatizaciya-skladskogo-ucheta/) (дата звернення 07.06.2025)
11. Про Індустрію 5.0 – чому це стає актуальним для України. [URL: https://www.industry4ukraine.net/publications/pro-industriyu-5-0-chomu-cze-staye-aktualnym-dlya-ukrayiny/](https://www.industry4ukraine.net/publications/pro-industriyu-5-0-chomu-cze-staye-aktualnym-dlya-ukrayiny/) (дата звернення 07.06.2025)

12. Робототехніка на складі: нове слово в оптимізації логістики URL: <https://forstor.ua/ca-conveyors/> (дата звернення 07.06.2025)
13. Шило Н. Ю. Засоби захисту систем промислової автоматизації та управління / Назар Юрійович Шило // ХНУРЕ. – 2020. – 66 URL: https://openarchive.nure.ua/bitstream/document/14073/1/Shylo_140-144.pdf.
14. Add-ons for SAP Extended Warehouse Management (EWM). 2025. URL: <https://www.valantic.com/en/sap-services/add-ons-for-sap-extended-warehouse-management-ewm/> (дата звернення 15.06.2025)
15. Arduino Mega 2560 Rev3/A000067. Технічні характеристики. URL : <https://evo.net.ua/arduino-mega-2560-rev3/?srsltid=AfmBOooA9Tx4J42vzlvNFgrWxLAU274qQMqJGwtzWJvtm05BdVozbJ5p> (дата звернення 08.05.2025)
16. Amazon Robotics (Kiva Systems) URL: <https://www.roboticsbusinessreview.com/robotic-company> (дата звернення 27.05.2025)
17. Automated guided vehicle URL: Automated guided vehicle - Wikipedia (дата звернення 10.06.2025)
18. Boston Dynamics Handle URL: <https://www.wired.com/story/what-boston-dynamics-rolling-handle> (дата звернення 20.05.2025)
19. Handbook of Industrial Automation: Richard L. Shell, Ernest L. Hall. 2020 p. С. 157-177.
20. Robotics URL: <https://en.wikipedia.org/wiki/Robotics> (дата звернення 07.06.2025)
21. Tesla Semi URL: <https://www.tesla.com/semi> (дата звернення 10.05.2025)

ДОДАТКИ

Додаток А.

Код програми для мікропроцесорної плати Arduino Mega 2560 написаний мовою C

```

#include "EEPROM.h"

// Визначення пінів
#define IN1 3
#define IN2 4
#define EN1 2
#define IN3 5
#define IN4 6
#define EN2 7
#define HORN 13
#define echoPin 9
#define trigPin 8
// Команди
#define CMD_L 'L'
#define CMD_R 'R'
#define CMD_H 'H'
#define CMD_F 'F'
#define CMD_r 'r'
#define CMD_w 'w'
char incomingByte; // Прийнятий символ з
серійного порту

// Буфери для даних команд
char L_Data[4];
byte L_index = 0;
char R_Data[4];
byte R_index = 0;
char H_Data[1];
byte H_index = 0;
char F_Data[8];
byte F_index = 0;
char command;
unsigned long currentTime, lastTimeCommand,
autoOFF, loopTime;

void setup() {
    Serial.begin(9600);
    // Налаштування пінів
    pinMode(HORN, OUTPUT);
    pinMode(IN1, OUTPUT);
    pinMode(IN2, OUTPUT);
    pinMode(IN3, OUTPUT);
    pinMode(IN4, OUTPUT);
    pinMode(EN1, OUTPUT);
    pinMode(EN2, OUTPUT);
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);

    currentTime = millis();
    loopTime = currentTime;
    timer_init();
}
// Ініціалізація таймера та параметра авто
вимкнення (autoOFF)
void timer_init() {
    uint8_t sw_autoOFF = EEPROM.read(0);

    if (sw_autoOFF == '1') {
        char var_Data[3];
        var_Data[0] = EEPROM.read(1);
        var_Data[1] = EEPROM.read(2);
        var_Data[2] = EEPROM.read(3);
        var_Data[3] = '\0'; // Завжди додавайте
        нульовий термінатор для atoi
        autoOFF = atoi(var_Data) * 100;
    }
    else if (sw_autoOFF == '0') {
        autoOFF = 999999;
    }
    else if (sw_autoOFF == 255) {
        autoOFF = 2500;
    }
    else {
        autoOFF = 2500; // Значення за замовчуванням
    }

    currentTime = millis();
}

void loop() {
    // Обробка вхідних даних з серійного порту
    if (Serial.available() > 0) {
        incomingByte = Serial.read();

        // Визначення типу команди
        if (incomingByte == CMD_L) {
            command = CMD_L;
            memset(L_Data, 0, sizeof(L_Data));
            L_index = 0;
        }
        else if (incomingByte == CMD_R) {
            command = CMD_R;
            memset(R_Data, 0, sizeof(R_Data));
            R_index = 0;
        }
        else if (incomingByte == CMD_H) {
            command = CMD_H;
            memset(H_Data, 0, sizeof(H_Data));
            H_index = 0;
        }
        else if (incomingByte == CMD_F) {
            command = CMD_F;
            memset(F_Data, 0, sizeof(F_Data));
            F_index = 0;
        }
        else if (incomingByte == '\r') {
            command = 'e'; // Виконати команду
            Control4WD
        }
        else if (incomingByte == '\t') {
            command = 't'; // Виконати команду Flash_Or
        }

        // Запис даних у відповідні буфери
        if (command == CMD_L && incomingByte !=
        CMD_L) {

```

```

        if (L_index < sizeof(L_Data)) {
            L_Data[L_index++] = incomingByte;
        }
    }
    else if (command == CMD_R && incomingByte !=
CMD_R) {
        if (R_index < sizeof(R_Data)) {
            R_Data[R_index++] = incomingByte;
        }
    }
    else if (command == CMD_H && incomingByte !=
CMD_H) {
        if (H_index < sizeof(H_Data)) {
            H_Data[H_index++] = incomingByte;
        }
    }
    else if (command == CMD_F && incomingByte !=
CMD_F) {
        if (F_index < sizeof(F_Data)) {
            F_Data[F_index++] = incomingByte;
        }
    }
    else if (command == 'e') {
        Control4WD(atoi(L_Data),      atoi(R_Data),
atoi(H_Data));
        delay(10);
    }
    else if (command == 't') {
        Flash_Op(F_Data[0],  F_Data[1],  F_Data[2],
F_Data[3], F_Data[4]);
    }

    lastTimeCommand = millis();
}

// Автоматичне вимкнення двигунів при
відсутності команд
if (millis() >= (lastTimeCommand + autoOFF)) {
    Control4WD(0, 0, 0);
}

// Функція керування 4WD (чотириколісним
приводом)
void Control4WD(int mLeft, int mRight, uint8_t Horn)
{
    // Лівий двигун
    if (mLeft > 0) {
        if (echo() <= 20) {
            digitalWrite(EN1, LOW);
            digitalWrite(IN1, LOW);
            digitalWrite(IN2, LOW);
        }
        else {
            analogWrite(EN1, mLeft);
            digitalWrite(IN1, LOW);
            digitalWrite(IN2, HIGH);
        }
    }
    else if (mLeft < 0) {
        analogWrite(EN1, abs(mLeft));
        digitalWrite(IN1, HIGH);
        digitalWrite(IN2, LOW);
    }
    else {

```

```

        digitalWrite(EN1, LOW);
        digitalWrite(IN1, LOW);
        digitalWrite(IN2, LOW);
    }
    // Правий двигун
    if (mRight > 0) {
        if (echo() <= 20) {
            digitalWrite(EN2, LOW);
            digitalWrite(IN3, LOW);
            digitalWrite(IN4, LOW);
        }
        else {
            analogWrite(EN2, mRight);
            digitalWrite(IN3, LOW);
            digitalWrite(IN4, HIGH);
        }
    }
    else if (mRight < 0) {
        analogWrite(EN2, abs(mRight));
        digitalWrite(IN3, HIGH);
        digitalWrite(IN4, LOW);
    }
    else {
        digitalWrite(EN2, LOW);
        digitalWrite(IN3, LOW);
        digitalWrite(IN4, LOW);
    }
    // Управління звуковим сигналом
    digitalWrite(HORN, Horn);
}
// Функція роботи з EEPROM (читання/запис)
void Flash_Op(char FCMD, uint8_t z1, uint8_t z2,
uint8_t z3, uint8_t z4) {
    if (FCMD == CMD_r) {
        Serial.print("FData:");
        Serial.write(EEPROM.read(0));
        Serial.write(EEPROM.read(1));
        Serial.write(EEPROM.read(2));
        Serial.write(EEPROM.read(3));
        Serial.print("\r\n");
    }
    else if (FCMD == CMD_w) {
        EEPROM.write(0, z1);
        EEPROM.write(1, z2);
        EEPROM.write(2, z3);
        EEPROM.write(3, z4);
        timer_init(); // Оновлення параметрів після
запису
        Serial.print("FWOK\r\n");
    }
}
// Функція ультразвукового датчика відстані
float echo() {
    long duration, distance;
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    distance = duration / 58.2;
    return (distance);
}

```

Додаток Б.

Код програми передача даних по Wi-Fi каналу із управління роботизованим візком

```

#include <EEPROM.h>

#define IN1 3 // Лівий двигун вперед
#define IN2 4 // Лівий двигун назад
#define EN1 5 // ШІМ лівого двигуна

#define IN3 6 // Правий двигун вперед
#define IN4 7 // Правий двигун назад
#define EN2 9 // ШІМ правого двигуна

#define DEFAULT_TIMEOUT 5000 // мс, за
замовчуванням

unsigned long lastCommandTime = 0;
unsigned long timeoutMs = DEFAULT_TIMEOUT;

String input = "";
int leftPWM = 0;
int rightPWM = 0;

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

  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(EN1, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  pinMode(EN2, OUTPUT);

  loadTimeoutFromEEPROM();
  stopMotors();
}

void loop() {
  while (Serial.available()) {
    char c = Serial.read();
    if (c == '\r') {
      processCommand(input);
      input = "";
      lastCommandTime = millis();
    } else {
      input += c;
    }
  }

  if (millis() - lastCommandTime > timeoutMs) {
    stopMotors();
  }
}

void processCommand(String cmd) {
  if (cmd.startsWith("L")) {
    leftPWM = cmd.substring(1).toInt();
  } else if (cmd.startsWith("R")) {
    rightPWM = cmd.substring(1).toInt();
  } else if (cmd.startsWith("Fr")) {
    Serial.print("Timeout: ");
    Serial.println(timeoutMs);
  } else if (cmd.startsWith("Fw")) {
    int t = cmd.substring(2).toInt();
    if (t > 0 && t < 60000) {
      timeoutMs = (unsigned long)t;
      saveTimeoutToEEPROM(timeoutMs);
      Serial.println("Timeout saved.");
    }
  }

  applyMotorPower(leftPWM, rightPWM);
}

void applyMotorPower(int left, int right) {
  // Лівий двигун
  if (left >= 0) {
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, LOW);
  } else {
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
    left = -left;
  }
  analogWrite(EN1, constrain(left, 0, 255));

  // Правий двигун
  if (right >= 0) {
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, LOW);
  } else {
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);
    right = -right;
  }
  analogWrite(EN2, constrain(right, 0, 255));
}

void stopMotors() {
  analogWrite(EN1, 0);
  analogWrite(EN2, 0);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);
}

void loadTimeoutFromEEPROM() {
  unsigned int t = EEPROM.read(0) << 8 |
EEPROM.read(1);
  if (t > 0 && t < 60000) {
    timeoutMs = t;
  } else {
    timeoutMs = DEFAULT_TIMEOUT;
  }
}

void saveTimeoutToEEPROM(unsigned int t) {
  EEPROM.write(0, (t >> 8) & 0xFF);
  EEPROM.write(1, t & 0xFF);
}

```