

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр

Освітній рівень

Роботизований стенд лінійного переміщення

Назва теми

КвРАКІТ.021048.01.20.ПЗ

Рівень вищої освіти перший

Галузь знань 15 «Автоматизація та приладобудування»

Шифр, назва

Спеціальність 151 «Автоматизація та комп'ютерно-інтегровані технології»

Шифр, назва

Освітня програма «Автоматизація та комп'ютерно-інтегровані технології»

Назва

Виконав:

студент IV курсу, групи АКІТ-21-1

  
Підпис

Віталій ЧАПАЙЛО  
Ім'я, ПРІЗВИЩЕ

Керівник

  
Підпис

Микола ФЕДУЛА  
Ім'я, ПРІЗВИЩЕ

Нормоконтролер

  
Підпис

Людмила КОРЕЦЬКА  
Ім'я, ПРІЗВИЩЕ

До захисту допускаю:  
зав. кафедри АКІТтаР

  
Підпис

Валерій МАРТИНЮК  
Ім'я, ПРІЗВИЩЕ

« 22 » червень 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних Технологій

Кафедра АВТОМАТИЗАЦІЇ, КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ ТЕХНОЛОГІЙ ТА РОБОТОТЕХНІКИ

Рівень вищої освіти БАКАЛАВР

Галузь знань 15 «Автоматизація та приладобудування»

Шифр і назва

Спеціальність 151 «Автоматизація та комп'ютерно-інтегровані технології»

Шифр і назва

Освітня програма Автоматизація та комп'ютерно-інтегровані технології

ЗАТВЕРДЖУЮ

Завідувач кафедри

07 . 02 . 2025

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Чапайло Віталій Олегович

Прізвище, ім'я, по батькові студента

1 Тема роботи Роботизований стенд лінійного переміщення

Керівник роботи Федула Микола Васильович, к.т.н., доцент

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

Затверджено наказом ректора університету від 07.02.2025 №23

2 Строк подання студентом роботи на кафедру 02.06.2025

3 Вихідні дані до роботи Завдання на кваліфікаційну роботу

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

Теоретичні основи автматизацій та аналіз існуючих рішень

Розробка алгоритму та програмне моделювання автоматизованої системи

Реалізація системи та тестування її роботи





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

Структурна схема автоматизованої лінії переміщення

Реалізація логіки керування маніпулятором

Людино-машинний інтерфейс

6 Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
Антиплагіат	Микола ФЕДУЛА, доцент кафедри АКІТтаР		
Нормоконтроль	Людмила КОРЕЦЬКА, доцент кафедри АКІТтаР		

7 Дата видачі завдання 07 лютого 2025 р.

## КАЛЕНДАРНИЙ ПЛАН


Назва етапів (розділів) кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1 Вибір та затвердження теми кваліфікаційної роботи; розробка завдання на кваліфікаційну роботу; складання календарного графіка виконання кваліфікаційної роботи	01.03.2025	виконано
2 Вивчення предметної області, в якій планується використання системи автоматизації; аналіз вимог до системи автоматизації	15.03.2025	виконано
3 Проектування та розробка загальної архітектури і структури системи автоматизації, інтерфейсу користувача; вибір засобів реалізації системи автоматизації	29.03.2025	виконано
4 Програмна реалізація та тестування системи автоматизації	12.04.2025	виконано
5 Написання тексту пояснювальної записки та розробка графічних матеріалів	19.04.2025	виконано
6 Остаточне коригування кваліфікаційної роботи з урахуванням зауважень керівника; оформлення кваліфікаційної роботи як документа відповідно до вимог	11.05.2025	виконано
7 Отримання супровідних документів (відгуку керівника, рецензії, довідки про перевірку плагіат): нормоконтроль	30.05.2025	виконано
8 Підготовка до захисту кваліфікаційної роботи	02.06.2025	виконано

Студент

  
\_\_\_\_\_  
Підпис

Віталій ЧАПАЙЛО  
Ім'я, ПРІЗВИЩЕ

Керівник

  
\_\_\_\_\_  
Підпис

Микола ФЕДУЛА  
Ім'я, ПРІЗВИЩЕ

## АНОТАЦІЯ

Тема кваліфікаційної роботи: «Роботизований стенд лінійного переміщення»

Автор роботи: Віталій ЧАПАЙЛО


Керівник роботи: Микола ФЕДУЛА

Пояснювальна записка: 65с., 9 рис., 42 джерела, 3 додатки

Графічна частина: 10 презентаційних слайдів

ПЛК, TIA PORTAL, АВТОМАТИЗАЦІЯ, РОБОТИЗОВАНА ЛІНІЯ, КОНВЕЄР, MANIPULATOR, HMI.

**Мета роботи:** розроблення програмно-апаратної моделі роботизованої лінії транспортування, яка охоплює конвеєрну систему та два маніпулятори. Об'єктом дослідження виступає автоматизована система управління роботизованою лінією переміщення. Предметом дослідження виступають підходи до розробки програмної логіки та візуалізації для ПЛК Siemens у середовищі TIA Portal, а також розробка структурної, функціональної та монтажної документації лінії. В процесі виконання роботи було використано метод системного аналізу для оцінки типів маніпуляторів і контролерів, проведено експериментальне моделювання логіки в TIA Portal та розроблено програмні блоки FC1 (LAD) і FC2 (SCL). Створено структурну схему, таблицю теґів, розроблено HMI-інтерфейс та проведено логічну перевірку системи. Отримані результати можуть бути застосовані в навчальному процесі, для лабораторних стендів з автоматизації, а також як приклад промислового рішення для виробничих ліній.

  
Підпис студента

19.06.2025  
Дата

## ЗМІСТ

ВСТУП.....	4
1. ТЕОРЕТИЧНІ ОСНОВИ ПОБУДОВИ РОБОТИЗОВАНИХ ЛІНІЙ ПЕРЕМІЩЕННЯ .....	6
1.1 Поняття та роль автоматизованих систем у сучасному виробництві.....	6
1.2 Типи систем транспортування та їх призначення.....	7
1.3 Роботизовані маніпулятори, їх класифікація, принцип роботи, сфери використання .....	9
1.4 Концепція створення автоматизованих ліній транспортування .....	11
1.5 Програмовані логічні контролери: будова, функції, стандарти.....	13
1.6 ПЛК Siemens серії S7-1200: огляд, технічні характеристики.....	15
1.7 Середовище TIA Portal як фундамент для програмування автоматизованих систем .....	17
1.8 Мови програмування LAD, SCL та FBD: порівняльний аналіз .....	19
1.9 Людино-машинні інтерфейси: їх значення в управлінні та візуалізації .....	22
1.10 Висновки до першого розділу.....	23
2. РОЗРОБКА АЛГОРИТМУ ТА ПРОГРАМНЕ МОДЕЛЮВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ .....	25
2.1 Обґрунтування вибору обладнання та архітектури системи.....	25
2.2 Структурна схема автоматизованої системи керування .....	29
2.3 Розробка алгоритму роботи лінії (опис логіки процесу) .....	30
2.4 Втілення логіки керування у середовищі TIA Portal .....	32
2.5 Програмування в LAD та SCL .....	35

КвРАКІТ.021048.01.20.ПЗ				
Зм.	Арк.	Недокум.	Підпис	Дата
Виконав.		Чайніло В.О.		19.06.23
Перевір.		Федула М.В.		20.06.23
Н.контр.		Корещька Л.О.		20.06.23
Затвер.		Мартинюк В.В.		20.06.23
Розробка система автоматичного керування				
		Літера	Аркуш	Аркушів
		и		
ХНУ АКІТ-21-1				

2.6	Налаштування вхідних/вихідних сигналів та PLC Tags .....	40
2.7	Симуляція роботи системи (планування) .....	41
2.8	Розробка НМІ -інтерфейсу для взаємодії з оператором.....	42
2.9	Розробка принципової та монтажноі електричних схем.....	44
2.10	Висновки до другого розділу .....	49
3.	РЕАЛІЗАЦІЯ СИСТЕМИ ТА ТЕСТУВАННЯ ЇЇ РОБОТИ. ....	51
3.1	Середовище реалізації автоматизованої системи .....	51
3.2	Підготовка до запуску програми та моделювання .....	52
3.3	Перевірка роботи логіки керування .....	53
3.4	Робота НМІ -інтерфейсу в процесі керування .....	55
3.5	Аналіз роботи системи на прикладах.....	58
3.6	Типові помилки, труднощі реалізації та їх усунення .....	59
3.7	Висновки до третього розділу.....	60
	ВИСНОВКИ.....	672
	ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	673
	ДОДАТОК А.....	67
	ДОДАТОК Б .....	68
	ДОДАТОК В.....	69

## ВСТУП

В умовах сьогодення, коли промисловість стрімко розвивається, виникає нагальна потреба впроваджувати гнучкі й результативні автоматизовані системи.

Ці системи мають гарантувати високу продуктивність та незмінну якість випущеної продукції. Одним із ключових методів вдосконалення виробничого процесу є використання роботизованих комплексів, що об'єднують в собі транспортування, маніпулювання та керування технологічними процесами.

Дана робота присвячена розробці автоматизованої роботизованої лінії, компонентами якої є конвеєр та два маніпулятори. Така конфігурація широко застосовується в сучасних виробництвах, де необхідно забезпечити точне та синхронізоване переміщення деталей чи комплектуючих між різними етапами обробки або збирання.

У рамках дипломного проєкту передбачено: створення структурної схеми, розробка принципової та монтажно-електричних схем, програмування логіки керування за допомогою TIA Portal, а також моделювання функціонування системи з використанням PLCSIM. Особлива увага буде зосереджена на забезпеченні послідовної взаємодії між виконавчими механізмами та датчиками для забезпечення безперебійного циклу роботи лінії.

Актуальність цієї теми зумовлена необхідністю скоротити залучення ручної праці, підвищити безпеку на виробництві та забезпечити стабільний технологічний процес з мінімальним впливом оператора.

Втілення таких рішень відкриває перспективи для масштабування та адаптації виробництва під мінливі потреби ринку.

Мета роботи: розроблення програмно-апаратної моделі роботизованої лінії транспортування, яка охоплює конвеєрну систему та два маніпулятори.

Здійснення моделювання функціонування лінії в PLCSIM. Це необхідно для автоматизації процесу переміщення предметів.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						4
Зм.	Арк.	№докум.	Підпис	Дата		

Об'єкт – процес автоматизованого пересування об'єктів на виробництві за допомогою роботизованих пристроїв. Предмет – структура та засади створення автоматизованої системи управління лінією транспортування на базі програмованого логічного контролера Siemens S7-1200, зокрема алгоритми взаємодії конвеєра, сенсорів та маніпуляторів, а також програмне втілення управління в середовищі TIA Portal.

Для досягнення мети роботи необхідно виконати ряд завдань:

1. Провести теоретичний аналіз сучасних систем автоматизації та засобів керування.
2. Розробити структурну, принципову та монтажну схеми системи.
3. Реалізувати програмне забезпечення у середовищі TIA Portal із використанням мов LAD та SCL.
4. Створити HMI-інтерфейс для візуалізації та керування системою.
5. Виконати моделювання роботи системи та перевірити коректність логіки.

Розроблена система може бути використана в освітньому процесі для підготовки фахівців з автоматизації, а також як базова модель для впровадження у виробничих лініях транспортування. Вона дозволяє демонструвати принципи побудови ПЛК-систем, тестування алгоритмів і взаємодії з HMI.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						5
Зм.	Арк.	№докум.	Підпис	Дата		

# ТЕОРЕТИЧНІ ОСНОВИ ПОБУДОВИ РОБОТИЗОВАНИХ ЛІНІЙ ПЕРЕМІЩЕННЯ

## 1.1 Поняття та роль автоматизованих систем у сучасному виробництві

У нинішньому промисловому середовищі автоматизовані системи відіграють вирішальну роль у забезпеченні ефективності, стабільності та безпеки технологічних процесів.

Стрімкий розвиток електроніки, інформаційних технологій та систем керування сприяв появі нового покоління автоматизованих комплексів, здатних працювати безперервно, з високою точністю та мінімальним втручанням оператора. Саме автоматизація стала основою для побудови розумного виробництва – так званої індустрії 4.0, де ключовими факторами є адаптивність, інтеграція та цифровізація [1-5].

Автоматизована система – це сукупність технічних засобів, програмного забезпечення, виконавчих і вимірювальних пристроїв, які разом забезпечують виконання технологічного процесу з мінімальною участю людини. Такі системи призначені для спрощення, прискорення та стандартизації операцій, що особливо важливо в умовах серійного та масового виробництва.

Однією з головних переваг автоматизованих систем є зменшення впливу людського фактора. За допомогою надійного алгоритмічного забезпечення, датчиків і виконавчих механізмів можна досягти високого ступеня повторюваності операцій, що безпосередньо впливає на якість продукції. Окрім цього, автоматизація дозволяє зменшити витрати часу на виконання типових дій, підвищити загальну продуктивність і знизити собівартість виробу.

Залежно від рівня складності, автоматизовані системи можуть включати прості блоки керування для однієї машини або інтегровані комплекси, що охоплюють кілька взаємозалежних процесів.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		6

Особливу роль у таких системах відіграють програмовані логічні контролери [1] (ПЛК), що дозволяють задавати гнучку логіку взаємодії між елементами системи, адаптуватися до змін технологічних умов і зберігати надійність у роботі навіть у складних умовах експлуатації.

У розрізі автоматизованих ліній переміщення важливою складовою є організація чіткої послідовності подій, таких як виявлення об'єкта, переміщення його в потрібне положення, виконання технологічної операції (захват, подача, сортування тощо), перевірка результату. Усі ці дії реалізуються за допомогою взаємодії сенсорів, електроприводів, виконавчих механізмів і систем логічного керування [2-9].

На сьогоднішній день автоматизовані лінії активно використовуються не лише у великих виробничих комплексах, а й у навчальних закладах – для формування практичних навичок у майбутніх фахівців.

Таким чином, вивчення принципів побудови автоматизованих систем є важливою частиною підготовки інженерів з автоматизації, мехатроніки та робототехніки.

## 1.2 Типи систем транспортування та їх призначення

Системи транспортування є невід'ємною частиною автоматизованих виробничих ліній та логістичних центрів. Вони забезпечують переміщення сировини, заготовок, готової продукції чи комплектуючих між різними робочими зонами, машинами або технологічними етапами.

Ефективність транспортувальної системи безпосередньо впливає на ритмічність і продуктивність всього виробничого процесу [8].

В основу класифікації систем транспортування покладено принцип дії, характер об'єкта, що переміщується, спосіб подачі та ступінь автоматизації.

Серед найбільш поширених типів транспортувальних систем виділяють стрічкові конвеєри, які призначені для безперервного переміщення продукції за

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						7
Зм.	Арк.	№докум.	Підпис	Дата		

рахунок руху гнучкої стрічки. Вони є найпростішими у конструкції та широко використовуються у легкій, харчовій, машинобудівній та логістичній галузях. Роликові конвеєри застосовуються для переміщення об'єктів із жорсткою основою, наприклад, коробок, ящиків або піддонів. Їхнє переміщення здійснюється за рахунок обертання валиків або під дією сили тяжіння. Ланцюгові та пластинчасті конвеєри використовують у випадках, коли необхідне переміщення важких або нестандартних об'єктів. Вони мають підвищену міцність і довговічність. Пневмотранспортери забезпечують переміщення сипучих або дрібних матеріалів за допомогою стисненого повітря чи вакууму [9-15]. Такі системи часто використовуються в хімічній, фармацевтичній та харчовій промисловості. Підвісні системи транспортування дозволяють ефективно організувати переміщення в умовах обмеженого простору, зокрема у складських приміщеннях та малогабаритних цехах.

Залежно від рівня автоматизації транспортувальні системи можуть бути ручними, напівавтоматичними або повністю автоматизованими. В останньому випадку їх інтегрують до загальної системи керування, що базується на ПЛК. Це дозволяє забезпечити синхронізацію із технологічним обладнанням, маніпуляторами, сенсорами та виконавчими пристроями. Автоматизовані системи транспортування відіграють важливу роль в організації потокової структури виробництва. Їхнє грамотне проектування дозволяє уникнути простоїв, перевантаження, аварійних ситуацій, а також забезпечити гнучке переналагодження під зміни обсягів або типів продукції. Крім того, сучасні конвеєрні системи нерідко мають можливість зворотного зв'язку, тобто обмінюються інформацією з ПЛК або SCADA-системою, що дозволяє формувати адаптивне, саморегульоване середовище виробництва [10-11].

У рамках цієї дипломної роботи передбачається створення моделі стрічкового конвеєра з двома маніпуляторами, де кожен елемент взаємодіє через логіку, реалізовану у середовищі TIA Portal [6].

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						8
Зм.	Арк.	№докум.	Підпис	Дата		

Така система дозволяє моделювати типовий цикл переміщення та сортування об'єктів, подібно до реальних умов сучасного виробництва.

### 1.3 Роботизовані маніпулятори, їх класифікація, принцип роботи, сфери використання

Роботизовані маніпулятори є невід'ємною складовою сучасних автоматизованих виробництв. Їх призначення - виконання різноманітних операцій, таких як захоплення, транспортування, орієнтація, збирання або сортування предметів. Завдяки високій точності та відтворюваності рухів, маніпулятори суттєво збільшують ефективність технологічного процесу, особливо у серійному чи масовому виробництві [5, 16].

Різноманіття конструкцій та принципів роботи маніпуляторів обумовлює кілька класифікаційних характеристик. За типом приводу розрізняють пневматичні, електричні та гідравлічні маніпулятори. Пневматичні приводи переважно використовуються для простих завдань з легкими об'єктами, оскільки відзначаються простотою, швидкістю та відносною дешевизною. Електричні приводи забезпечують більшу точність і керованість, характеризуються компактністю та наявністю зворотного зв'язку. Гідравлічні ж приводи застосовуються у системах з великими навантаженнями, де потрібна значна сила.

Залежно від кількості ступенів свободи маніпулятори можуть мати 2–3 ступені, що достатньо для виконання базових операцій, таких як захоплення, поворот або переміщення по прямій, або 4–6 і більше ступенів, що дозволяє здійснювати складні маніпуляції та точне позиціонування в тривимірному просторі [14].

За типом керування розрізняють програмовані та адаптивні маніпулятори. Програмовані виконують операції за заздалегідь заданим алгоритмом, що робить їх найбільш поширеними у промисловості. Адаптивні ж системи здатні

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						9
Зм.	Арк.	№докум.	Підпис	Дата		

реагувати на зміни навколишнього середовища завдяки використанню сенсорів, таких як датчики зору чи зусилля [15-19].

Принцип дії маніпулятора, узагальнено, включає кілька послідовних етапів. Спочатку система отримує сигнал про наявність об'єкта від відповідного датчика або контролера. Далі відбувається активація виконавчого механізму, наприклад, подача повітря до пневмоприводу. Потім здійснюється захоплення об'єкта за допомогою затискача, вакууму, магнітного поля або іншого механізму. Після цього об'єкт переміщується в задану точку за допомогою кінематичної структури, і на завершення – вивільняється [20].

У додатку А представлено структурну схему лінії переміщення, яка відображає основні вузли системи та взаємозв'язки між ними.

Маніпулятори знаходять широке застосування в різних галузях промисловості та науки. У машинобудуванні вони використовуються для завантаження та розвантаження верстатів. В електроніці – для точного компонування плат і мікроелементів. У харчовій промисловості маніпулятори виконують операції з упаковки, сортування та розподілу продукції. В автомобілебудуванні їх застосовують під час зварювання, фарбування та збирання деталей [21]. У медицині маніпулятори використовуються для точного дозування лікарських засобів і виконання мікрomanipуляцій. У сільському господарстві автономні роботизовані маніпулятори збирають урожай, акуратно зрізуючи плоди й мінімізуючи пошкодження рослин. На великих логістичних складах вони автоматизують сортування та переміщення пакунків, істотно скорочуючи час обробки замовлень.

У космічних системах маніпуляторні руки орбітальних апаратів виконують захоплення та сервісне обслуговування супутників, продовжуючи їхній ресурс без повернення на Землю. Навіть у кіновиробництві роботизовані крани-маніпулятори забезпечують ультраточне керування рухом камер, відкриваючи режисерам можливість для реалізації складних динамічних сцен [22-25].

					КВРАКІТ.021048.01.20.ПЗ	Арк. 10
Зм.	Арк.	№докум.	Підпис	Дата		

У навчальних і демонстраційних системах маніпулятори моделюють спрощені версії реальних процесів. Вони дозволяють вивчити принципи логіки, синхронізації з сенсорами, керування у реальному часі. В рамках даної роботи буде реалізовано модель взаємодії двох маніпуляторів із конвеєром, яка демонструє типову логіку сортування або подачі об'єктів.

#### 1.4 Концепція створення автоматизованих ліній транспортування

Автоматизовані лінії транспортування є ключовим елементом сучасних виробничих та логістичних систем. Їх основне завдання – забезпечити безперервне, точне та контрольоване переміщення об'єктів між стадіями технологічного процесу. Побудова таких ліній базується на принципах модульності, гнучкості, масштабованості та інтеграції засобів управління.

До базових елементів автоматизованих систем належать різні складові, що забезпечують їхню ефективну роботу. Конвеєрна система відповідає за фізичне переміщення об'єктів і може бути стрічковою, роликовою, ланцюговою, штовхальною або іншого типу – залежно від конкретного завдання. Сенсори та датчики використовуються для визначення положення об'єкта, його наявності, швидкості чи інших параметрів, найчастіше застосовуються індуктивні, оптичні датчики або кінцеві вимикачі. Виконавчі механізми, до яких належать маніпулятори, пневматичні поршні, серводвигуни тощо, забезпечують виконання додаткових операцій, таких як захоплення, підйом або сортування. Контролер (програмований логічний контролер, або ПЛК) виконує обробку сигналів від сенсорів, реалізує логіку управління та формує відповідні вихідні дії. Інтерфейс користувача, наприклад HMI або SCADA, дозволяє візуалізувати стан системи, здійснювати управління, а також отримувати повідомлення про помилки або збої [26].

Конструктивно лінії проектуються за принципом послідовного або циклічного переміщення об'єктів. У першому випадку – об'єкти рухаються безперервно або зупиняються на кожному етапі; у другому – рух відбувається

тільки після виконання певної умови (наприклад, спрацювання датчика). Такий підхід дає змогу організувати логічно завершену технологічну операцію на кожній позиції [27].

На рівні управління автоматизована система поділяється на кілька функціональних рівнів. Фізичний рівень включає датчики, двигуни та інші пристрої, що безпосередньо приймають і передають сигнали. Логічний рівень, представлений програмованим логічним контролером (ПЛК), відповідає за обробку подій, реалізацію логіки та взаємодію між окремими блоками системи. Візуалізаційний рівень, наприклад НМІ, забезпечує відображення інформації оператору та надає можливість взаємодії з технологічним процесом [28-30].

Принципи побудови таких систем ґрунтуються на модульності, що передбачає можливість легкого розширення без необхідності істотних змін в архітектурі. Важливим є й принцип гнучкості, який дозволяє змінювати алгоритми роботи без перепідключення обладнання. Інтегрованість забезпечує безперервний обмін даними між ПЛК, НМІ, сенсорами та SCADA-системами. Особливу увагу приділяють безпеці, яка включає наявність аварійних режимів, захист від помилок і обмеження доступу до системи.

У межах цієї дипломної роботи реалізується концептуальна модель автоматизованої лінії, що складається з одного стрічкового конвеєра та двох маніпуляторів. Рух деталей контролюється за допомогою трьох датчиків: на вході, в зоні першого маніпулятора та на виході. ПЛК Siemens [2] S7-1200 обробляє сигнали, формує команди виконавчим пристроям та забезпечує повну синхронізацію всіх елементів [5, 29].

Проект моделює реальну ситуацію в промисловості, де об'єкт транспортується до певної точки, зупиняється, захоплюється або обробляється, після чого передається далі. Це дозволяє наочно продемонструвати ефективність логіки управління, заснованої на сигнальних подіях, та відтворити типову структуру виробничої лінії.

					КВРАКІТ.021048.01.20.ПЗ	Арк. 12
Зм.	Арк.	№докум.	Підпис	Дата		

## 1.5 Програмовані логічні контролери: будова, функції, стандарти

Програмовані логічні контролери [1] (ПЛК) становлять фундамент сучасних систем автоматизації. Вони забезпечують надійне, гнучке та адаптивне керування процесами у реальному часі, даючи змогу інтегрувати різноманітні пристрої, сенсори, приводи та системи в єдиний керований технологічний комплекс.

ПЛК (англ. PLC [3] – Programmable Logic Controller) – це мікропроцесорний пристрій, призначений для отримання, опрацювання сигналів та формування керуючих команд відповідно до алгоритму, визначеного користувачем. Він поєднує в собі апаратну частину (вхідні та вихідні модулі, блок живлення, процесор) та програмне забезпечення, яке реалізує логіку управління [1, 5, 30-32].

ПЛК отримує сигнали з входів (кнопки, датчики), виконує програму, яка містить умови та операції, та видає керуючі сигнали на виходи (мотор, реле, лампи та ін.). Програма виконується циклічно з високою частотою (мілісекунди), що забезпечує швидку реакцію системи в режимі реального часу [33-35].

Типова структура програмованого логічного контролера (ПЛК) включає кілька основних компонентів. Центральний процесорний модуль (CPU) відповідає за обробку даних, виконання користувацької програми, обмін інформацією з пам'яттю та периферійними пристроями. Вхідні модулі приймають дискретні або аналогові сигнали від датчиків і вимикачів, тоді як вихідні модулі передають керувальні сигнали на реле, пускачі або інші виконавчі механізми [36]. Комунікаційні інтерфейси, зокрема Ethernet, PROFINET, RS-485 та інші, забезпечують обмін даними з іншими пристроями системи. Пам'ять у складі ПЛК зберігає програму користувача, змінні та конфігураційні дані. Блок живлення, зазвичай на 24 В постійного струму, подає енергію як на логічні схеми, так і на зовнішні пристрої [37-39].

Контролери виконують низку основних функцій. Зокрема, логічне управління дозволяє реалізовувати умовні переходи, перевірку станів і керування технологічними об'єктами. Послідовне виконання команд забезпечує правильний порядок дій у межах циклу процесу. Також використовуються таймери, лічильники й тригери, які дозволяють реалізувати затримки у роботі або підрахунок подій. Для роботи з фізичними величинами застосовується обробка аналогових сигналів, що дає змогу контролювати температуру, тиск, рівень тощо [40]. Важливою є можливість зв'язку з візуалізаційними системами, зокрема з панелями оператора НМІ, що дає змогу відображати інформацію та взаємодіяти з процесом. Крім того, ПЛК виконує функції діагностики та обробки помилок – виявляє аварійні ситуації й формує відповідні сигнали для реагування.

Програмування програмованих логічних контролерів регламентується міжнародним стандартом ІЕС 61131-3, який визначає основні мови програмування, що використовуються для створення керувальних програм. Усього стандарт передбачає п'ять мов. Серед них графічна мова LAD (Ladder Diagram), яка нагадує схеми релейної логіки та є зручною для електротехніків. Інша графічна мова – FBD (Function Block Diagram) – дозволяє будувати логіку у вигляді взаємопов'язаних функціональних блоків. Низькорівнева текстова мова STL (Statement List) використовується для більш точного програмування та подібна до асемблера. Високорівнева текстова мова SCL (Structured Control Language) має синтаксис, подібний до мов Pascal або C, і застосовується для складних алгоритмів [38-41]. Також використовується мова CFC (Continuous Function Chart), яка, на відміну від FBD, не має жорсткої структури сітки, що надає більше гнучкості у розміщенні елементів логіки [40-42].

Це дозволяє інженерам вибирати найзручніший спосіб реалізації логіки, враховуючи завдання та підготовку користувача.

Використання програмованих логічних контролерів має низку важливих переваг. Насамперед, ПЛК вирізняються високою надійністю та стійкістю до промислових електромагнітних завад, що забезпечує стабільну роботу в

складних умовах. Однією з ключових переваг є гнучкість – зміну логіки керування можна здійснити шляхом перепрограмування, без необхідності фізичного втручання або перепаявання схем [42]. Крім того, контролери характеризуються швидкою реакцією на зміну вхідних сигналів, що є критично важливим для точного управління процесами. Тривалий термін служби ПЛК та можливість подальшої модернізації дозволяють використовувати їх як у нових, так і в оновлених автоматизованих системах.

Отже, ПЛК є універсальним та ефективним рішенням для створення як простих, так і складних систем управління – від одиничного механізму до розгалужених ліній автоматизації. У цьому дипломному проєкті використано ПЛК Siemens [2] S7-1200, який підтримує різні мови програмування та має достатню кількість входів та виходів для реалізації лінії переміщення з двома маніпуляторами.

#### 1.6 ПЛК Siemens серії S7-1200: огляд, технічні характеристики

Контролери Siemens [2] S7-1200 – це представники сучасного покоління програмованих логічних контролерів (ПЛК). Їх розроблено для вирішення різноманітних задач автоматизації на об'єктах малого та середнього масштабу. Ці контролери поєднують надійність, гнучкість, здатність до розширення та зручність програмування. Завдяки цим характеристикам, вони користуються популярністю в промисловості, сфері навчання, наукових дослідженнях та лабораторних роботах.

Серія S7-1200 була розроблена компанією Siemens [2] як наступник лінійки S7-200. Основним акцентом при розробці було забезпечення підтримки сучасних протоколів зв'язку, інтеграції з середовищем TIA Portal, розширених можливостей діагностики та інтегрованих функцій безпеки. Серія підтримує різні мови програмування, такі як LAD, FBD, STL, SCL [4], CFC, відповідно до стандарту IEC 61131-3. Це дає змогу реалізовувати як прості логічні умови, так і складні алгоритми управління.

					КВРАКІТ.021048.01.20.ПЗ	Арк. 15
Зм.	Арк.	№докум.	Підпис	Дата		

Найбільш поширеними моделями ПЛК є CPU 1211C, яка є базовою і призначена для виконання простих завдань; CPU 1212C, що має більше входів і виходів і підходить для проєктів середньої складності; CPU 1214C – розширена модель з додатковими можливостями підключення; а також CPU 1215C і 1217C, які застосовуються у більш складних і продуктивних системах.

В даному проєкті використовується CPU 1214C DC/DC/DC (6ES7 214-1AG40-0XB0), яка має достатню кількість вбудованих входів/виходів та підтримує розширення за допомогою додаткових модулів. Для обґрунтування вибору контролера доцільно проаналізувати його технічні характеристики, наведені в таблиці 1.1.

Таблиця 1.1 – Технічні характеристики CPU 1214C

Параметр	Значення
Дискретні входи (вбудовані)	14
Дискретні виходи (вбудовані)	10 (типу DC)
Аналогові входи	2 (10-бітні, стандарт)
Комунікаційні інтерфейси	PROFINET (вбудований), можливість розширення
Обсяг програми	до 100 кБ
Пам'ять для даних	до 50 кБ
Годинник реального часу	вбудований
Швидкість циклу	~0,08 мс/кБ програми
Живлення	24 В постійного струму (DC)

Можливості та переваги ПЛК включають високу продуктивність, що робить їх оптимальними для задач реального часу. Вони підтримують інтеграцію комунікацій через такі протоколи, як Ethernet, Modbus TCP, USS і G120. Легке розширення системи досягається за рахунок використання додаткових модулів введення/виведення та комунікаційних модулів. Крім того, ПЛК без проблем інтегруються з панелями оператора Siemens і системами SCADA, що забезпечує зручність у візуалізації та управлінні. Вбудовані функції безпеки включають управління доступом, резервне копіювання даних та шифрування проєктів, що

підвищує захист інформації. Для розробки та налагодження застосовується середовище PLC SIM, яке підтримує симуляцію роботи без необхідності фізичного устаткування.

У рамках даного дипломного дослідження контролер S7-1200 CPU 1214C реалізує логіку автоматизованої лінії транспортування. Його входи з'єднані з сенсорами, що фіксують присутність об'єкта на лінії, а виходи керують двигуном конвеєра та двома маніпуляторами. Програма написана в середовищі TIA Portal [6] V17 з використанням мов LAD та SCL, а її функціонування перевіряється за допомогою PLCSIM. У додатку Б наведено реальне зображення контролера Siemens S7-1200, що застосовувався для реалізації керування.

### 1.7 Середовище TIA Portal як фундамент для програмування автоматизованих систем

### 1.7 Середовище TIA Portal як фундамент для програмування автоматизованих систем

TIA Portal (Totally Integrated Automation Portal) – це багатогранна платформа, розроблена Siemens [2] для проектування, налаштування, програмування, моделювання та підтримки систем автоматизації. Вона гарантує всебічну інтеграцію програмованих логічних контролерів (ПЛК), панелей оператора (НМІ), перетворювачів частоти, систем вводу/виводу та інших елементів у єдиному програмному середовищі.

Основна ідея TIA Portal [6] полягає в створенні уніфікованого інтерфейсу для кожного етапу розробки автоматизованої системи. Це дозволяє зменшити витрати часу, мінімізувати вірогідність помилок, покращити уніфікацію та стандартизацію рішень. Розробник може працювати над одним проектом, водночас використовуючи конфігуратор обладнання, програмну логіку, візуалізацію, зв'язок та діагностику.

TIA Portal містить низку функціональних компонентів, серед яких STEP 7 (Basic/Professional) – інструмент для налаштування та програмування ПЛК Siemens, включно з моделями LOGO!, S7-1200 та S7-1500. Для створення НМІ

					КВРАКІТ.021048.01.20.ПЗ	Арк. 17
Зм.	Арк.	№докум.	Підпис	Дата		

та SCADA-інтерфейсів призначений WinCC у різних версіях: Basic, Comfort і Advanced. Компонент Startdrive використовується для конфігурування частотних перетворювачів SINAMICS. Віртуальний симулятор PLCSIM дозволяє тестувати програми контролерів без необхідності використання реального обладнання. Крім того, до складу TIA Portal входять інструменти Security та Diagnostics Tools, які забезпечують безпеку, контроль версій і моніторинг стану системи.

TIA Portal [6] володіє інтуїтивно зрозумілим інтерфейсом з вкладками, панелями компонентів та структурованим деревом проекту. Розробка логіки можлива різними мовами – LAD, FBD, STL, SCL – які можна поєднувати в рамках одного проекту.

Графічний редактор дає змогу швидко розробляти функціональні блоки, підключати входи/виходи, визначати умови, використовувати бібліотеки готових елементів, таймери, лічильники тощо. Всі зміни відбуваються в реальному часі, з можливістю компіляції та попереднього перегляду.

Переваги використання TIA Portal полягають в інтеграції всіх етапів проекту – від вибору обладнання до запуску системи. Висока швидкість розробки досягається завдяки використанню шаблонів, бібліотек і функції автозаповнення. Потужні інструменти налагодження забезпечують онлайн-моніторинг, роботу з таблицями змінних та встановлення точок зупину для зручного відлагодження. Гнучка структура проекту підтримує багатократні блоки, структури, масиви та користувацькі типи даних (UDT). Безкоштовна симуляція за допомогою PLC SIM дає змогу повністю протестувати логіку керування без фізичного ПЛК. Крім того, система автоматично уніфікує документацію, формуючи схеми, описи та списки змінних.

В цій роботі TIA Portal [6] V17 використовується як головна платформа для створення програми управління автоматизованою лінією.

У межах реалізації проекту було виконано апаратне налаштування ПЛК Siemens S7-1200. Для всіх датчиків і виконавчих механізмів створено таблицю змінних (PLC Tags), що забезпечує зручне оперування даними. Основна логіка

					КВРАКІТ.021048.01.20.ПЗ	Арк. 18
Зм.	Арк.	№докум.	Підпис	Дата		

керування реалізована у вигляді окремих функціональних блоків: FC1 написаний мовою LAD, а FC2 – мовою SCL. Організацію виконання програмних блоків забезпечено через головний організаційний блок OB1. Проведено повноцінне тестування роботи програми у симуляторі PLC SIM, що дозволило перевірити її працездатність без використання фізичного обладнання. За потреби було створено макет інтерфейсу HMI для запуску, зупинки та моніторингу системи в реальному часі.

Отже, використання TIA Portal [6] дозволяє не лише реалізувати функціональну модель, а й надати їй вигляду справжнього промислового рішення, підготовленого до впровадження або масштабування.

### 1.8 Мови програмування LAD , SCL та FBD: порівняльний аналіз

Мови програмування для програмованих логічних контролерів [1] (ПЛК) регламентуються міжнародним стандартом IEC 61131-3, що визначає п'ять базових мов. В межах цієї дипломної роботи застосовуються найбільш поширені серед них – LAD (Ladder Diagram), SCL (Structured Control Language) та FBD (Function Block Diagram). Кожна з цих мов має свої характеристики, переваги та традиційні сфери використання.

LAD [8], або релейно-контактна схема, є графічною мовою, що імітує вигляд електричних схем релейної логіки. Головними елементами є умовні контакти (NO/NC), котушки, таймери, лічильники та логічні операції.

Серед основних переваг мови LAD слід відзначити її візуальну наочність, завдяки чому вона є зрозумілою навіть технічним спеціалістам, які не мають досвіду програмування. Ця мова добре підходить для реалізації простих логічних послідовностей, таких як запуск, зупинка або фіксація стану обладнання. Крім того, вона дозволяє швидко здійснювати налагодження та вносити зміни до програми.

Водночас до недоліків LAD можна віднести складність реалізації на ній складних математичних операцій або детальної обробки даних. Також її важко

					КВРАКІТ.021048.01.20.ПЗ	Арк. 19
Зм.	Арк.	№докум.	Підпис	Дата		

масштабувати у випадках, коли проєкт передбачає велику кількість умов або комбінацій.

LAD [8] є однією з основних мов для початкового навчання роботі з ПЛК, тому широко застосовується у лабораторних роботах, зокрема в цьому проєкті (FC1).

SCL [4] – це мова високого рівня, текстова, подібно до Pascal чи C. Вона створена для втілення в життя комплексних алгоритмів, як-от обчислення, цикли, логічні перевірки, а також для роботи зі змінними.

Мова програмування SCL має низку переваг, зокрема вона спрощує реалізацію складної логіки, підтримує роботу зі структурами даних, масивами та циклами, а також забезпечує кращу масштабованість у проєктах із великою кількістю умов. Водночас SCL вимагає базових знань у сфері програмування, що може стати бар'єром для початківців. Крім того, вона є менш інтуїтивно зрозумілою порівняно з графічною мовою LAD.

В рамках дипломної роботи, мову SCL [4] використано для розробки блоку FC2, що дублює логіку керування у текстовому вигляді..

В дипломному проєкті SCL використовується в FC2 для демонстрації альтернативного підходу до реалізації логіки.

FBD – графічна мова, що представляє логіку у вигляді блоків-функцій, з'єднаних лініями передачі сигналів. Кожен блок виконує певну дію: логічну операцію, функцію таймера, лічильника тощо.

Мова програмування FBD зручна для реалізації логіки, яка має чітку структуру, й ідеально підходить для об'єктно-орієнтованого підходу. Вона візуально простіша за SCL, але водночас гнучкіша, ніж LAD, що робить її придатною для широкого кола завдань. Найпростішими елементами залишаються логічні вентиля AND OR XOR та елементи пам'яті SR і RS які забезпечують формування сигналів утримання. На вищому рівні абстракції застосовують таймери TON TOF TP і лічильники STU STD що мають внутрішній стан і власний таймбейс синхронізований із циклом виконання процесора.

					КВРАКІТ.021048.01.20.ПЗ	Арк. 20
Зм.	Арк.	№докум.	Підпис	Дата		

Завдяки відокремленню передачі даних від часової послідовності викликів FBD стає інтуїтивним інструментом для побудови комбінованої та секційної логіки коли чіткі залежності між сигналами важливіші за послідовність інструкцій які обробляє процесор. Візуальна природа нотації полегшує ревізію коду командою технічного аудиту оскільки ланцюги причинно наслідкових зв'язків видно без додаткових коментарів. Об'єктно орієнтований стиль підтримується через можливість інкапсулювати групу блоків у власноруч створений функціональний блок з параметрами та локальною пам'яттю тобто інженер проектує ієрархічні структури що легко повторно використовувати в інших проектах. Проте FBD має і свої недоліки: вона малоефективна при реалізації програм з великою кількістю умов або послідовних дій, а при створенні складних схем займає багато екранного простору, що ускладнює навігацію та редагування. Для вибору мови програмування ПЛК розглянуто порівняльні характеристики LAD, SCL та FBD у таблиці 1.2.

Таблиця 1.2 – Порівняння LAD, SCL, FBD

Характеристика	LAD	SCL	FBD
Тип мови	Графічна	Текстова	Графічна
Зручність читання	Висока	Висока	Середня
Поріг входу	Низький	Середній	Середній
Можливості	Стандартні логічні	Підтримка складної логіки, обчислень, циклів	Обмежено складні
Застосування	Початкові проекти	Розширені алгоритми, математичні задачі	Модульні схеми

Вибір мови програмування для ПЛК залежить від кількох ключових чинників: складності поставленого завдання, кваліфікації інженера, важливості читабельності коду, потреб у масштабованості та наявності технічної підтримки.

У представленій дипломній роботі використання комбінації LAD [8] та STL забезпечує можливість реалізації логіки як у візуальному форматі, так і на

низькому рівні, що в результаті збільшує гнучкість та функціональні можливості розробленої системи.

### 1.9 Людино-машинні інтерфейси: їх значення в управлінні та візуалізації

В сучасних автоматизованих системах велике значення відіграють людино-машинні інтерфейси (Human-Machine Interface) – інструменти взаємодії оператора з технологічним процесом.

НМІ [5] забезпечують візуалізацію параметрів, керування станом системи та діагностику подій в реальному часі.

Основні функції НМІ охоплюють відображення поточного стану системи, включаючи інформацію про стан датчиків, приводів та аварійні сигнали, а також передачу команд від оператора до ПЛК, наприклад, для запуску, зупинки чи вибору режиму роботи. Крім того, інтерфейс дозволяє зчитувати та вводити змінні параметри, такі як температура, швидкість чи значення лічильників, а також записувати та переглядати журнал подій, що включає історію аварій, збоїв та змін у режимах. Графічне представлення інформації у вигляді анімацій обладнання, кольорової індикації та діаграм додатково підвищує зручність користування. Завдяки цим можливостям НМІ забезпечує зручну експлуатацію, підвищує безпеку та зменшує час реагування на зміни в роботі системи.

Існує кілька типів НМІ залежно від складності та призначення. Найпростішими є текстові панелі, що мають сегментні екрани та кнопки керування. Більш функціональні графічні панелі оснащені екранами з підтримкою векторної графіки, кольорового зображення та сенсорного управління. Сенсорні екрани або Touch Panels є повноцінними мультимедійними пристроями з інтерактивним інтерфейсом. Веб-інтерфейси дозволяють здійснювати віддалений доступ до системи через звичайний веббраузер. Найпотужніші можливості мають SCADA-системи, які представляють собою розширений варіант НМІ з додатковими функціями збору, архівації та аналітики даних.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						22
Зм.	Арк.	№докум.	Підпис	Дата		

У середовищі TIA Portal інтегровано модуль WinCC, який дає змогу створювати інтерфейси користувача для панелей SIMATIC HMI. Серед основних можливостей – використання редактора екранів із вбудованими об'єктами, такими як кнопки, індикатори та графіки, прив'язка елементів інтерфейсу до змінних ПЛК, створення багаторівневої структури меню, реалізація анімації руху, миготіння та зміни кольору, підтримка кількох мов повідомлень, а також налаштування звітів, рецептів, системи паролів та журналів подій.

Використання HMI має ряд переваг. Насамперед це підвищення інформативності: оператор отримує повну картину процесу в реальному часі. Завдяки підказкам, логіці підтверджень та блокуванням вводяться механізми зменшення ймовірності помилок. Інтерфейс відрізняється ергономікою – замість традиційного набору кнопок і ламп застосовується інтуїтивно зрозуміле графічне середовище. Крім того, HMI легко адаптується до змін у вимогах, а також забезпечує можливість дистанційного керування і моніторингу процесів.

У межах даного дипломного проєкту реалізовано базовий HMI-інтерфейс для керування та спостереження за роботизованою лінією. Зокрема, він містить кнопки для запуску і зупинки процесу, графічне відображення станів датчиків (на вході, в позиції, на виході), індикацію роботи двигуна конвеєра та обох маніпуляторів, повідомлення про зупинку чи помилки, а також можливість тестування інтерфейсу через симуляцію у WinCC без потреби у фізичній панелі.

Використання HMI суттєво спрощує роботу оператора, дозволяє швидко діагностувати стан системи та забезпечує зручність тестування в симуляційному середовищі PLC [3]SIM + WinCC Runtime Advanced.

#### 1.10 Висновки до першого розділу

У першому розділі було проведено огляд теоретичних основ побудови автоматизованих систем керування, зокрема роботизованих ліній переміщення. Розглянуто принципи роботи ПЛК, основні мови програмування відповідно до

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						23
Зм.	Арк.	№докум.	Підпис	Дата		

стандарту ІЕС 61131-3, особливості роботи середовища ТІА Portal, а також класифікацію конвеєрів та промислових маніпуляторів.

На основі аналізу визначено доцільність використання контролера Siemens S7-1200 та програмного середовища ТІА Portal для реалізації поставленого завдання (у додатку Б наведено реальне зображення контролера Siemens S7-1200, що застосовувався для реалізації керування). Обґрунтовано вибір мов LAD і SCL як найбільш зручних для реалізації логіки керування в умовах побудови навчального або виробничого стенду.

Зібрані теоретичні відомості стали підґрунтям для розробки власної системи керування, опис якої наведено в наступних розділах роботи.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		24

## 2. РОЗРОБКА АЛГОРИТМУ ТА ПРОГРАМНЕ МОДЕЛЮВАННЯ АВТОМАТИЗОВАНОЇ СИСТЕМИ

### 2.1 Обґрунтування вибору обладнання та архітектури системи

Проектування автоматизованої системи потребує виваженого підбору апаратної складової, що буде відповідати функціональним, технічним та економічним критеріям. У рамках даної дипломної роботи планується створення навчального макету лінії переміщення предметів з використанням стрічкового конвеєра, двох роботизованих маніпуляторів, сенсорів позиціонування та ПЛК Siemens [2] S7-1200.

У якості основного елемента керування автоматизованої системи було обрано контролер Siemens S7-1200. Це рішення зумовлене низкою технічних і функціональних переваг. Насамперед варто відзначити повну сумісність цього контролера з інтегрованим середовищем розробки TIA Portal, що дозволяє охопити всі етапи створення системи – від апаратного конфігурування до моделювання роботи.

Контролер оснащено вбудованими цифровими входами та виходами, кількість яких повністю відповідає вимогам проєкту. При необхідності функціональність S7-1200 може бути розширена за рахунок додаткових модулів – як аналогових, так і комунікаційних. Однією з переваг є також підтримка різних мов програмування згідно зі стандартом IEC 61131-3, зокрема LAD, SCL і FBD, що надає розробнику гнучкість у виборі стилю програмування.

Окремо слід згадати високу надійність цього ПЛК у промислових умовах та надзвичайно швидке виконання логіки програми – зазвичай цикл займає близько однієї мілісекунди, що особливо важливо для задач реального часу.

В межах проєкту використовується модель CPU 1214C DC/DC/DC, яка забезпечує оптимальне співвідношення між функціональністю та вартістю.

Під час проектування було досліджено декілька видів роботизованих маніпуляторів, які відрізняються за принципом дії, типом приводу,

					КВРАКІТ.021048.01.20.ПЗ	Арк. 25
Зм.	Арк.	№докум.	Підпис	Дата		

конструкцією, кількістю ступенів свободи та вартістю. Порівняльна таблиця наведена у таблиці 2.1.

У результаті проведеного аналізу для реалізації задачі переміщення об'єктів було прийнято рішення використовувати пневматичні маніпулятори з двома ступенями свободи. Такий вибір обумовлений низкою переваг, що відповідають вимогам автоматизованої системи. Зокрема, ці маніпулятори характеризуються простою інтеграцією до існуючого середовища керування, що спрощує процес програмування та налагодження.

Крім того, пневматичні приводи забезпечують високу швидкість спрацьовування, що особливо важливо для виконання циклічних операцій із переміщення. Також слід відзначити низький рівень енергоспоживання та надійну роботу навіть в умовах обмеженого простору, де не завжди можливо застосувати електромеханічні або гідравлічні рішення.

Також важливою була сумісність з цифровими виходами ПЛК, що дозволяє керувати пневмоклапанами без потреби в додаткових інтерфейсах.

Для реалізації транспортувальної частини системи було обрано стрічковий конвеєр з електроприводом постійного струму. Такий тип приводу забезпечує безперервність та плавність руху стрічки, що є критично важливим для точного позиціонування об'єктів у процесі автоматизації. Окрім цього, конструкція стрічкового конвеєра є зручною для створення макету, а також дозволяє легко реалізувати як постійне, так і позиційне транспортування в залежності від логіки керування.

Додатковою перевагою є сумісність з різними типами датчиків, які використовуються для виявлення об'єктів на стрічці, що полегшує інтеграцію в загальну систему контролю та управління. Мотор конвеєра управляється через дискретний вихід ПЛК з можливістю запуску/зупинки згідно з логікою.

У системі передбачено використання трьох сенсорів, кожен з яких виконує конкретну функцію. Sensor\_Start відповідає за виявлення наявності об'єкта на вхідній ділянці транспортера, сигналізуючи про початок обробки.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						26
Зм.	Арк.	№докум.	Підпис	Дата		

Sensor\_ConvPos інформує систему про те, що об'єкт досяг зони дії першого маніпулятора, що дозволяє ініціювати відповідні дії захоплення або переміщення. Нарешті, Sensor\_End визначає проходження об'єкта до кінцевої точки конвеєра, що є сигналом про завершення одного з етапів технологічного процесу.

Всі датчики – індуктивного або оптичного типу, з дискретним виходом 24 В DC, що дозволяє безпосередньо підключати їх до входів ПЛК без додаткової обробки сигналу.

Архітектура системи побудована відповідно до типової моделі автоматизації, де всі компоненти взаємодіють у визначеній послідовності. На вхідному рівні використовуються кнопки керування Start/Stop та сенсори, які передають сигнали до ПЛК. У свою чергу, програмна логіка обробляється в контролері за допомогою двох функціональних блоків: FC1, реалізованого мовою LAD, та FC2, написаного на SCL. Як результат обробки, формуються вихідні сигнали, що забезпечують запуск конвеєра й активацію маніпуляторів. Додатково передбачено зв'язок із віртуальною HMI-панеллю для керування та моніторингу процесу.

Для обґрунтування вибору типу маніпулятора проведено порівняльний аналіз основних технічних параметрів різних моделей (таблиця 2.1). У класичному конвеєрі гнучкість системи залишається низькою, оскільки всі транспортні маршрути заздалегідь визначені й не змінюються; навпаки, конвеєр, доповнений маніпулятором, отримує середній рівень гнучкості, бо сам маніпулятор здатний виконувати різноманітні дії та передавати об'єкти на різні ділянки, тоді як автономний робот-маніпулятор забезпечує високу гнучкість завдяки своїй мобільності й можливості перепрограмування під нові завдання.

Таблиця 2.1 – Порівняльна характеристика роботизованих маніпуляторів

Критерій	Класичний конвеєр	Конвеєр з маніпулятором	Автономний робот
Гнучкість системи	Низька – фіксовані маршрути	Середня – маніпулятор	Висока – мобільність і гнучке програм.
Простота впровадження	Висока	Середня	Низька – потребує складної навігації
Вартість обладнання	Низька	Середня/висока (залежить від типу маніпулятора)	Висока
Навчальна цінність	Базовий рівень	Висока – демонструє інтеграцію ПЛК, датчиків, приводів	Дуже висока – містить багато суміжних технологій
Масштабованість	Складна	Добра – можна додавати станції обробки	Обмежена через індивідуальність системи
Реалістичність моделі виробництва	Часткова	Висока – відповідає сучасним виробничим тенденціям	Висока, але складно відтворити реальні умови

## 2.2 Структурна схема автоматизованої системи керування

Структурна схема автоматизованої системи є важливим етапом проектування, який дає змогу візуально відобразити взаємозв'язки між усіма функціональними компонентами майбутньої системи. Вона виступає

фундаментом для подальшого формування електротехнічної документації, розробки логіки керування та забезпечення взаємодії між пристроями.

У рамках дипломного проєкту розроблено систему автоматизованого переміщення об'єктів, до складу якої входять кілька ключових компонентів. Центральним елементом є програмований логічний контролер Siemens S7-1200 (CPU 1214C), що виконує функції обробки сигналів, реалізації логіки керування та обміну даними з виконавчими механізмами й операторським інтерфейсом. Транспортування об'єктів між зонами обробки здійснюється за допомогою стрічкового конвеєра з електроприводом постійного струму.

Перший маніпулятор відповідає за первинне втручання в потік, наприклад, сортування або перенаправлення об'єктів, тоді як другий виконує завершальні дії, зокрема розміщення або скидання. У системі також застосовуються три сенсори: Sensor\_Start виявляє об'єкти на вході, Sensor\_ConvPos визначає їх положення для першого маніпулятора, а Sensor\_End сигналізує про завершення переміщення по лінії. Для ручного керування передбачено кнопки Start і Stop, а панель оператора HMI забезпечує зручну візуалізацію процесу та можливість дистанційного управління. Опис взаємодії компонентів (у додатку В розміщено фотографії основних елементів автоматизації, таких як датчики, маніпулятори та виконавчі механізми, які використовуються для реалізації логіки переміщення об'єктів лінією).

Компоненти взаємодіють між собою, використовуючи дискретні сигнали напругою 24 В DC. Вхідні сигнали від сенсорів та кнопок надходять на цифрові входи ПЛК. Програмна логіка аналізує ці сигнали, ухвалює рішення про

					КВРАКІТ.021048.01.20.ПЗ	Арк. 29
Зм.	Арк.	№докум.	Підпис	Дата		

виконання дій та формує вихідні сигнали для виконавчих елементів – двигуна конвеєра та маніпуляторів.

HMI [5] через з'єднання PROFINET взаємодіє з ПЛК, передаючи команди та отримуючи інформацію про стан процесу. Це дозволяє оператору контролювати хід виконання задач і втручатися в разі необхідності.

Умовна структурна схема системи подана нижче.

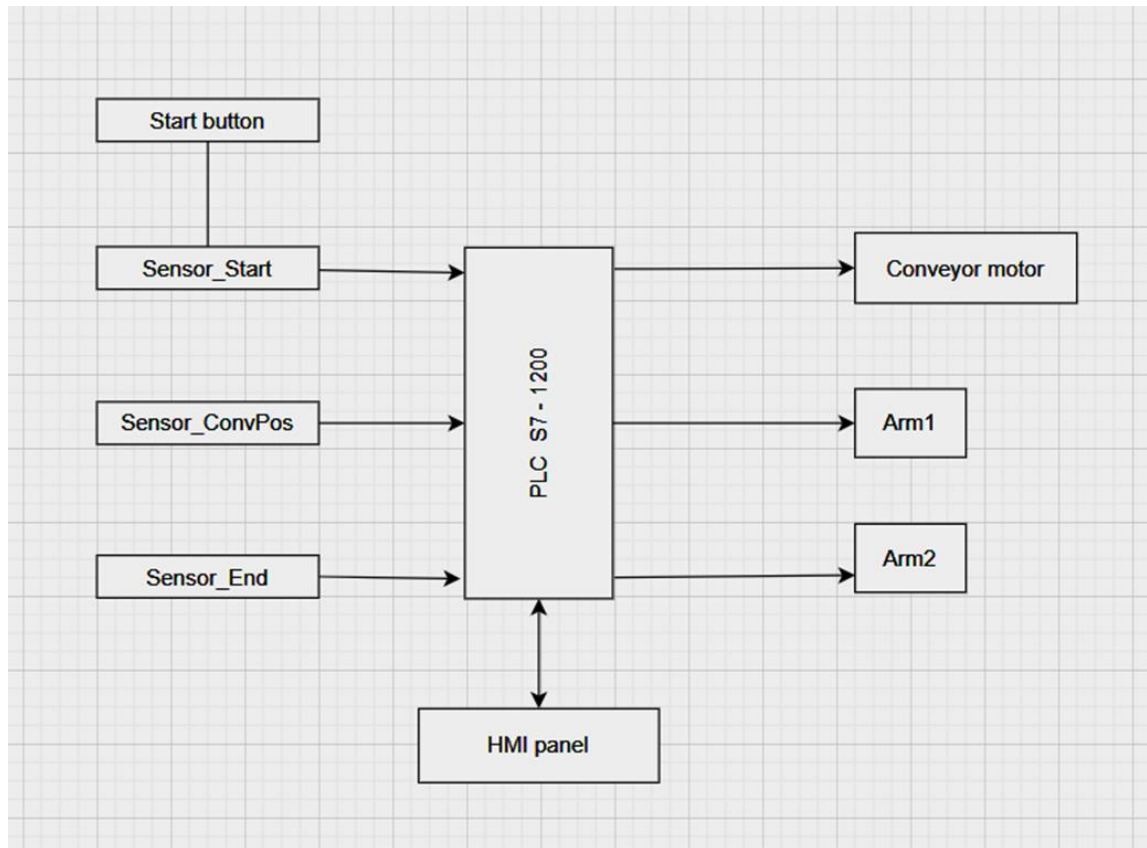


Рисунок 2.1 - Структурна схема автоматизованої лінії переміщення з використанням ПЛК S7-1200

### 2.3 Розробка алгоритму роботи лінії (опис логіки процесу)

Ефективність функціонування автоматизованої лінії переміщення прямо залежить від коректно розробленого алгоритму взаємодії між датчиками, виконавчими механізмами, контролером та оператором.

Логіка роботи мусить бути чіткою, послідовною та достатньо адаптивною для змін умов або розширення функціоналу.

Алгоритм роботи автоматизованої лінії розроблено на основі подійного керування (event-driven control), де кожна дія ініціюється у відповідь на зміну стану конкретного елемента (наприклад, активація датчика).

Логіка роботи системи організована у вигляді послідовного алгоритму, який починається з початкового стану, де всі виконавчі механізми знеструмлені, а на панелі НМІ відображається повідомлення про готовність до запуску. Після натискання користувачем кнопки Start активується змінна Cycle\_Enable, що дозволяє виконання циклу. Увімкнення змінної викликає запуск конвеєра, після чого система переходить у режим очікування надходження об'єкта. Його поява фіксується сенсором Sensor\_Start, і система готується до взаємодії з першим маніпулятором. Коли об'єкт досягає зони дії Sensor\_ConvPos, конвеєр автоматично зупиняється, і активується перший маніпулятор (Arm1), який виконує передбачену дію – наприклад, сортування або переміщення. Після завершення операції Arm1 вимикається, а конвеєр знову запускається для транспортування об'єкта до фінальної точки, де розташований сенсор Sensor\_End.

Коли об'єкт досягає кінцевої позиції, спрацьовує Sensor\_End, що призводить до зупинки конвеєра й активації другого маніпулятора (Arm2). Цей маніпулятор виконує завершальну дію, таку як укладання чи скидання об'єкта. Після цього Arm2 повертається у вихідне положення, а система переходить у початковий стан, готовий до обробки наступного об'єкта.

Передбачено декілька сценаріїв для зупинки системи. Вона може бути зупинена вручну – натисканням фізичної або віртуальної кнопки Stop, автоматично – у разі виявлення помилки або відсутності об'єкта, або ж за командою з НМІ. У будь-якому з випадків зупинки всі виконавчі механізми вимикаються, змінна Cycle\_Enable скидається, а на НМІ з'являється відповідне повідомлення.

					КВРАКІТ.021048.01.20.ПЗ	Арк. 31
Зм.	Арк.	№докум.	Підпис	Дата		

Для підвищення функціональності системи реалізовано додаткові можливості. Зокрема, реалізовано захист від повторного запуску циклу до завершення попереднього, аварійну зупинку через змінну Emergency\_Stop, індикацію поточного стану системи за допомогою LED-сигналів (віртуальних або фізичних), а також підрахунок виконаних циклів за допомогою змінної Cycle\_Count, що дає змогу вести статистику роботи системи.

#### 2.4 Втілення логіки керування у середовищі TIA Portal

Середовище TIA Portal (Totally Integrated Automation Portal) від Siemens [2] було обрано як головний інструмент для розробки, налаштування та імітації автоматизованої лінії транспортування. Його модульний дизайн дозволяє ефективно впроваджувати логіку управління програмованим логічним контролером (ПЛК), здійснювати взаємодію з вхідними/вихідними пристроями та НМІ.

На початковому етапі реалізації було створено новий проєкт у середовищі TIA Portal V17. Назва проєкту була обрана відповідно до тематики дипломної роботи, що дозволило забезпечити логічну відповідність усіх компонентів системи. В межах цього проєкту було здійснено ряд ключових дій, спрямованих на побудову апаратної та програмної структури системи керування.

Перш за все, до конфігурації було додано апаратний контролер Siemens S7-1200, зокрема модель CPU 1214C DC/DC/DC (каталожний номер 6ES7 214-1AG40-0XB0). Після додавання контролера виконано налаштування базової апаратної конфігурації, що включає призначення адрес цифровим входам та виходам, визначення типів сигналів (дискретні), а також параметрів живлення. Це дозволило адаптувати контролер до особливостей майбутньої системи та підготувати його до взаємодії з зовнішніми пристроями.

Паралельно було сформовано структуру програмної частини проєкту, яка охоплює ключові програмні блоки. Серед них: блок FC1 (Main\_LAD), в якому реалізовано основну логіку у вигляді релейно-контактної схеми за допомогою

					КВРАКІТ.021048.01.20.ПЗ	Арк. 32
Зм.	Арк.	№докум.	Підпис	Дата		

мови LAD; блок FC2 (Main\_SCL), який дублює частину логіки у вигляді програмного коду на мові SCL, що дозволяє гнучкіше обробляти складні умови. Обидва функціональні блоки викликаються у рамках організаційного блоку OB1, який задає послідовність виконання основної логіки керування.

З метою забезпечення ефективної взаємодії програмованого логічного контролера з периферійними пристроями було сформовано таблицю тегів (PLC Tags), що містить перелік усіх змінних, які використовуються в системі автоматизованого керування. Кожна змінна відповідає певному фізичному або логічному елементу, зокрема датчикам, виконавчим механізмам, кнопкам керування, індикації та внутрішнім станам системи.

У таблиці були визначені і структуровані відповідні імена змінних, типи даних, а також їх призначення. Такий підхід дозволив створити централізовану структуру для зручної прив'язки змінних до програмного коду та елементів інтерфейсу НМІ. Створення таблиці тегів значно спростило процес розробки, підвищило читабельність програми та уможливило швидке внесення змін у разі модернізації або розширення системи. У таблиці 2.2 наведено перелік змінних, які використовуються в програмній логіці керування роботизованою лінією.

Таблиця 2.2 – Перелік змінних PLC Tags

Ім'я тега	Тип	Напрямок	Призначення
Start	Bool	Вхід	Кнопка запуску
Stop	Bool	Вхід	Кнопка зупинки
Sensor_Start Bool	Bool	Вхід	Датчик на вході лінії
Sensor_ConvPos	Bool	Вхід	Датчик позиції об'єкта для Arm1
Sensor_End Bool	Bool	Вхід	Датчик завершення проходу
Conv_Motor	Bool	Вихід	Привід стрічкового конвеєра
Arm1_Move	Bool	Вихід	Активація першого маніпулятора
Arm2_Move	Bool	Вихід	Активація другого маніпулятора
Cycle_Enable	Bool	Внутрішня	Активний цикл

Візуальне представлення створених змінних у середовищі TIA Portal [6] подано у таблиці 2.3 нижче.

Таблиця 2.3 – Змінні (PLC Tags) у середовищі TIA Portal

PLC tags									
	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	Start	Default tag table	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	кнопка запуску
2	Stop	Default tag table	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	кнопка стоп
3	Cycle_Enable	Default tag table	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Дозвіл на виконан...
4	Sensor_Start	Default tag table	Bool	%I0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Датчик на початку...
5	Sensor_ConvPos	Default tag table	Bool	%I0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Датчик біля маніп...
6	Sensor_End	Default tag table	Bool	%I0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Датчик на виході ...
7	Arm1_Move	Default tag table	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Керування перши...
8	Arm2_Move	Default tag table	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Керування други...
9	Conv_Motor	Default tag table	Bool	%Q0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Привід конвеєра
10	<Add new>				<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Усі змінні мають зрозумілі назви та використовуються як у LAD, так і SCL [4] логіці.

Програмна логіка системи була реалізована у функціональному блоці FC1 (Main\_LAD) з використанням релейно-контактної мови LAD, що забезпечує візуальне відображення логічних послідовностей. Основна послідовність дій передбачає запуск автоматичного циклу після натискання кнопки Start, за умови активної змінної Cycle\_Enable. Після цього відбувається активація конвеєра шляхом встановлення значення змінної Conv\_Motor в TRUE. У разі спрацювання датчика Sensor\_ConvPos конвеєр автоматично зупиняється, що дозволяє виконати операцію першого маніпулятора. Після завершення дії Arm1 конвеєр знову активується і транспортує об'єкт до кінцевої точки. Коли об'єкт досягає зони датчика Sensor\_End, конвеєр повторно зупиняється, активується другий маніпулятор Arm2, після чого система повертається у початковий стан очікування наступного циклу.

Паралельно з цим, у функціональному блоці FC2 (Main\_SCL) реалізовано дублювання частини вищезазначеної логіки з використанням текстової мови програмування SCL. Це було зроблено з метою демонстрації альтернативного підходу до реалізації автоматизованих алгоритмів, а також для порівняння зручності та гнучкості різних мов програмування, підтримуваних TIA Portal.

Організаційний блок OB1 забезпечує виклик обох функціональних блоків – FC1 та FC2 – у визначеній послідовності. Завдяки цьому реалізується циклічне виконання логіки керування під час роботи програмованого логічного контролера.

Перед компіляцією було виконано перевірку на синтаксичні помилки та правильність адресації сигналів. Після успішної компіляції проект був готовий до імітації в PLC SIM [3] та прив'язки до HMI .

## 2.5 Програмування в LAD та SCL

Для реалізації логіки автоматизованої лінії переміщення у середовищі TIA Portal було створено два основних програмних блоки. Перший блок, FC1, реалізований за допомогою мови LAD (Ladder Diagram) і відповідає за виконання основної логіки управління системою. У цьому блоці сформовано покрокову реакцію системи на сигнали з датчиків, керування виконавчими механізмами, а також забезпечено контроль за ходом виконання циклу. Другий блок, FC2, створено з використанням текстової мови програмування SCL (Structured Control Language). Він дублює частину логіки, реалізованої в FC1, і слугує демонстрацією альтернативного способу розробки програмного забезпечення для ПЛК, з акцентом на гнучкість, масштабованість та ефективність при написанні складних алгоритмів.

Мова LAD – це релейно-контактна схема, що візуально подібна до електричних схем з контакторами та котушками. Такий підхід дає можливість наочно описати логіку запуску/зупинки, активації механізмів та умов переходу між фазами.

Основна логіка, реалізована у функціональному блоці FC1, охоплює кілька взаємопов'язаних підсистем керування. Перш за все, реалізовано механізм активації автоматичного циклу за допомогою змінної Cycle\_Enable, яка активується натисканням кнопки Start і деактивується при натисканні Stop. Для

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						35
Зм.	Арк.	№докум.	Підпис	Дата		

цього застосовується або RS-тригер, або комбінація паралельно з'єднаних логічних контактів.

Керування конвеєром базується на стані змінної `Cycle_Enable` та показаннях датчика `Sensor_ConvPos`. Якщо система перебуває в активному стані (`Cycle_Enable` дорівнює `TRUE`), а об'єкт ще не досягнув зони першого маніпулятора (`Sensor_ConvPos` дорівнює `FALSE`), відбувається увімкнення конвеєра (`Conv_Motor = TRUE`). Коли ж об'єкт фіксується датчиком `Sensor_ConvPos`, конвеєр зупиняється шляхом присвоєння змінній `Conv_Motor` значення `FALSE`.

Управління першим маніпулятором (`Arm1`) реалізується за допомогою логічної умови: якщо датчик позиції `Sensor_ConvPos` активний та змінна `Cycle_Enable` залишається істинною, то відбувається активація змінної `Arm1_Move`, що забезпечує виконання дії маніпулятора. Після завершення дії `Arm1` повертається у вихідне положення.

Аналогічно, другий маніпулятор (`Arm2`) активується в момент, коли об'єкт досягає кінцевої точки лінії, що фіксується спрацюванням датчика `Sensor_End`. У цьому випадку змінна `Arm2_Move` встановлюється в `TRUE`, і маніпулятор виконує завершальну операцію.

Крім основної логіки керування виконавчими механізмами, у програмі реалізовані додаткові умови безпеки та забезпечення безперервності циклу. Зокрема, передбачено контроль стану кнопок «Старт» та «Стоп», що дозволяють оператору оперативно запускати або зупиняти роботу системи у будь-який момент часу. Змінна `Cycle_Enable` активується лише за наявності сигналу з кнопки «Старт» та неактивного сигналу з кнопки «Стоп», що гарантує безпечний запуск та зупинку технологічного процесу.

Усі програмні дії реалізовані у вигляді окремих функціональних блоків, що підвищує модульність та зручність обслуговування системи. Блок `FC1`, написаний мовою `LAD`, відповідає за основну логіку перемикання вихідних сигналів відповідно до станів датчиків та команд оператора. Блок `FC2`,

реалізований мовою SCL, дублює аналогічну логіку у вигляді текстового програмного коду, що дозволяє продемонструвати альтернативний спосіб програмування ПЛК.

Таке розмежування дозволяє гнучко модифікувати логіку в майбутньому, зберігаючи зрозумілу та структуровану архітектуру програми.

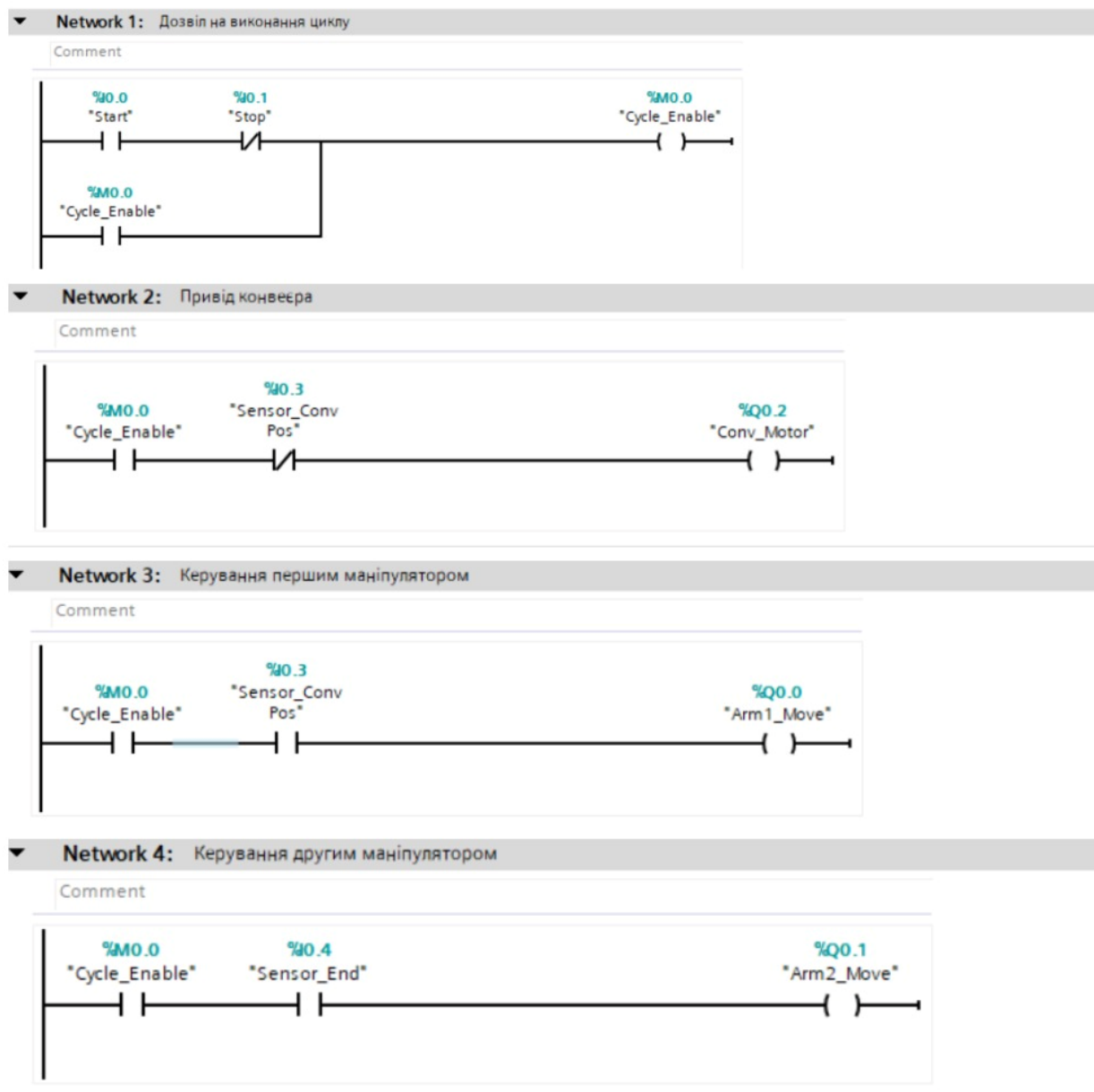


Рисунок 2.4 – Реалізація логіки керування в FC1 (LAD)

Функціональний блок FC2 розроблений задля втілення тієї ж логіки за допомогою структурованого тексту SCL [4]. Цей спосіб забезпечує можливість

більш лаконічно та гнучко викласти послідовність операцій, вдаючись до умовних операторів та логічних виразів.

```
1 // Увімкнення Cycle_Enable при Start
2 IF "Start" THEN
3     "Cycle_Enable" := TRUE;
4 END_IF;
5
6 // Вимкнення Cycle_Enable при Stop
7 IF "Stop" THEN
8     "Cycle_Enable" := FALSE;
9 END_IF;
10
11 // Керування конвеєром: активний цикл і Sensor_ConvPos = FALSE
12 IF "Cycle_Enable" AND (NOT "Sensor_ConvPos") THEN
13     "Conv_Motor" := TRUE;
14 ELSE
15     "Conv Motor" := FALSE;
16 END_IF;
17
18 // Активація Arm1 при Sensor_ConvPos
19 IF "Sensor_ConvPos" THEN
20     "Arm1_Move" := TRUE;
21 ELSE
22     "Arm1_Move" := FALSE;
23 END_IF;
24
25 // Активація Arm2 при Sensor_End
26 IF "Sensor_End" THEN
27     "Arm2_Move" := TRUE;
28 ELSE
29     "Arm2_Move" := FALSE;
30 END_IF;
31
```

Рисунок 2.5 – Реалізація логіки в FC2 (SCL)

Мова SCL [4] дає змогу втілювати не тільки елементарну логіку, а й складні обчислення, цикли, а також працювати з масивами даних. У порівнянні з LAD, вона вирізняється покращеною масштабованістю та компактністю. Її використання в дипломному проєкті надає можливість продемонструвати альтернативний метод побудови тієї ж логіки, що збільшує гнучкість та спрощує внесення змін до програмного коду.



## 2.6 Налаштування вхідних/вихідних сигналів та PLC Tags

Для повноцінної реалізації логіки управління в середовищі TIA Portal [6] необхідно правильно налаштувати сигнали, які надходять до ПЛК і виходять із нього. Цей процес передбачає створення таблиці тегів (PLC Tags), у якій кожному фізичному чи логічному елементу системи (кнопці, датчику, виконавчому механізму) призначається іменованій тег.

Процес створення таблиці тегів (PLC Tags) у середовищі TIA Portal [6] V17 таблиця тегів формується у відповідному розділі проекту. До неї додаються змінні типу Bool для цифрових входів і виходів. Назви змінних обираються максимально зрозумілими та відповідають їх функціональному призначенню в системі.

Усі вхідні та вихідні сигнали системи були реалізовані у формі тегів типу Bool, які були створені в таблиці PLC Tags у середовищі TIA Portal [6]. Їхнє призначення, напрямок і назви наведені на рисунку 2.3, що міститься в підпункті 2.4.

Для забезпечення чіткого функціонування автоматизованої системи в програмі передбачено використання набору основних змінних. До входів системи відносяться кнопки Start та Stop, що забезпечують запуск та зупинку циклу роботи. Інформацію про стан об'єктів на лінії забезпечують три сенсори: Sensor\_Start, який фіксує появу об'єкта на початку конвеєра, Sensor\_ConvPos – визначає положення об'єкта в зоні першого маніпулятора, та Sensor\_End – фіксує проходження об'єкта у фінальній зоні. Серед вихідних сигналів, що подаються на виконавчі механізми, передбачено сигнали Conv\_Motor для увімкнення транспортувального конвеєра, а також Arm1\_Move та Arm2\_Move – для керування відповідними маніпуляторами. Крім того, у програмі використовується внутрішня логічна змінна Cycle\_Enable, яка дозволяє контролювати активний стан системи та запобігає передчасному запуску нового циклу до завершення попереднього.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						40
Зм.	Арк.	№докум.	Підпис	Дата		

## 2.7 Симуляція роботи системи (планування)

Після завершення розробки логіки керування та налаштування змінних у середовищі TIA Portal [6] V17, наступним кроком зазвичай є тестування програми за допомогою вбудованого симулятора PLCSIM. Цей інструмент забезпечує можливість моделювати роботу ПЛК без необхідності використання фізичного обладнання, спостерігати за змінними та перевіряти алгоритми в режимі реального часу.

В межах цього проєкту реальне тестування з використанням PLC [3]SIM не проводилося, оскільки програмне забезпечення PLCSIM V17 наразі не встановлене на робочому комп'ютері. Водночас підготовчі етапи налаштування симуляції опрацьовані теоретично та інтегровані в структуру проєкту.

У разі наявності доступу до симулятора PLC SIM процес тестування розробленої системи передбачає декілька послідовних етапів. Насамперед здійснюється компіляція проєкту в середовищі TIA Portal з метою перевірки правильності побудованої логіки керування. Далі запускається програмне забезпечення PLC SIM із вибором відповідної моделі контролера S7-1200. Після цього створена програма завантажується до віртуального ПЛК, який переводиться в робочий режим RUN. Для перевірки роботи системи виконується імітація вхідних сигналів (зокрема Start, Stop та сигналів від сенсорів Sensor\_Start, Sensor\_ConvPos, Sensor\_End) шляхом зміни станів відповідних змінних у таблиці тегів.

У ході симуляції контролюється правильність формування вихідних сигналів на виконавчі механізми, зокрема активація Conv\_Motor, Arm1\_Move та Arm2\_Move, що дозволяє перевірити повний цикл роботи системи без підключення реального обладнання.

Усі змінні коректно визначені у таблиці PLC Tags, а логіка, реалізована у функціональних блоках FC1 (LAD) та FC2 (SCL), підключається через OB1.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						41
Зм.	Арк.	№докум.	Підпис	Дата		

Таким чином, система готова для подальшого тестування з PLCSIM або фізичним контролером [1].

Хоч і тестова симуляція не була виконана, структура проєкту дозволяє за необхідності швидко завантажити програму до PLC [3]SIM або реального пристрою. Програма повністю скопільована, без помилок і відповідає стандартним вимогам до тестування автоматизованих систем переміщення.

## 2.8 Розробка НМІ -інтерфейсу для взаємодії з оператором

НМІ (Human-Machine Interface) є одним із ключових компонентів у системах автоматизованого управління, що забезпечує взаємодію оператора з контролером [1]. Завдяки НМІ користувач отримує можливість запускати та зупиняти робочий процес, здійснювати моніторинг його виконання, зчитувати повідомлення, переглядати діагностику та виконувати інші важливі операції.

Розробка НМІ-інтерфейсу для даного дипломного проєкту передбачає досягнення кількох основних цілей. Насамперед, необхідно забезпечити оператору зручні засоби для керування роботою лінії переміщення об'єктів. Важливою складовою є візуалізація стану датчиків та виконавчих механізмів, зокрема конвеєра та маніпуляторів, що дозволяє оперативно відслідковувати поточний процес. Особлива увага приділяється створенню зрозумілого та інтуїтивно зручного графічного інтерфейсу, який мінімізує ймовірність помилок під час експлуатації системи. Окрім цього, розробка НМІ виконується для демонстрації ефективної інтеграції між програмованим логічним контролером та панеллю оператора за допомогою середовища TIA Portal.

На головному екрані панелі управління передбачено розміщення основних елементів інтерфейсу, що забезпечують керування та моніторинг процесу. Серед них передбачено кнопки для запуску та зупинки системи, які дають можливість оператору розпочинати або зупинити роботу лінії за потреби. Окрім того, розміщені індикатори стану основних вузлів системи, зокрема стану конвеєра, першого та другого маніпуляторів, а також показники спрацювання сенсорів.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						42
Зм.	Арк.	№докум.	Підпис	Дата		

Кожен індикатор змінює свій колір залежно від активності відповідного пристрою, що дозволяє оператору оперативно оцінювати стан системи. Така організація головного екрану сприяє зручності експлуатації та підвищенню безпеки роботи автоматизованої лінії. Для зручності управління процесом переміщення об'єкта створено НМІ-інтерфейс, структура якого представлена в таблиці 2.4.

Таблиця 2.4 – Структура елементів інтерфейсу НМІ

Елемент	Призначення
Кнопка Start	Запуск циклу роботи
Кнопка Stop	Аварійна зупинка
Індикатор Cycle_Enable	Сигнал про активний стан системи
Індикатор Sensor_ConvPos	Об'єкт у зоні Arm1
Індикатор Sensor_End	Об'єкт у кінцевій точці
Індикатор Conv_Motor	Стан мотора конвеєра (ON/OFF)
Індикатори Arm1_Move, Arm2_Move	Стан активності маніпуляторів

Усі елементи прив'язані до змінних із таблиці PLC [3] Tags, що дає змогу здійснювати моніторинг у режимі реального часу.

Процес розробки НМІ [5]-інтерфейсу складався з кількох послідовних етапів. Спочатку до конфігурації пристроїв проєкту було додано панель оператора НМІ [5], наприклад, модель КТР600 Basic або її аналог. Далі здійснювалося створення екрану, на якому розташовувалися основні елементи інтерфейсу – кнопки керування та індикатори стану. Кожен створений об'єкт на екрані прив'язувався до відповідної змінної з таблиці тегів, що дозволяло забезпечити обмін даними між панеллю та контролером. Для кнопок Start та Stop

були налаштовані події, які змінюють значення відповідних тегів при натисканні. Індикатори стану отримали кольорову індикацію: наприклад, зелений колір відповідав активному стану (TRUE), тоді як сірий – неактивному (FALSE). Завершальним етапом передбачалася симуляція або попередній перегляд створеного інтерфейсу у середовищі TIA Portal [6]; однак через відсутність доступу до PLCSIM Advanced цей крок в межах даного проєкту не виконувався. На рисунку 2.8 подано умовне зображення вигляду головного екрана HMI [5], що відповідає структурі логіки ПЛК.

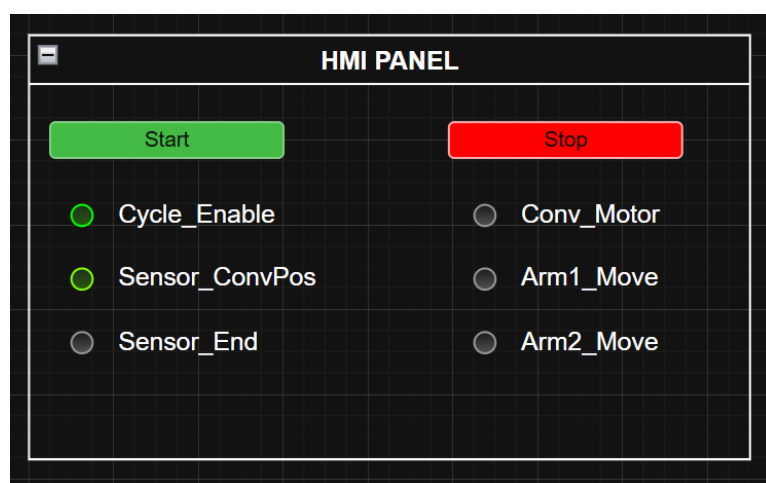


Рисунок 2.8 – Головне вікно HMI - панелі з кнопками керування та індикаторами станів системи

## 2.9 Розробка принципової та монтажно-ї електричних схем

Одним із ключових етапів створення автоматизованої системи є розробка електричних схем, які забезпечують коректне підключення обладнання, його захист, подачу живлення та взаємодію між компонентами системи.

У процесі реалізації автоматизованої системи переміщення об'єктів особливу роль відіграють електричні схеми, які забезпечують візуалізацію структури апаратної частини проєкту, встановлюють взаємозв'язки між усіма функціональними елементами системи, а також полегшують її практичну імплементацію. Електричні схеми є невід'ємною частиною документації, що

супроводжує будь-який проект автоматизації, оскільки дозволяють наочно відобразити логіку взаємодії програмованого логічного контролера з підключеними до нього датчиками, виконавчими механізмами, пристроями індикації та елементами керування.

Однією з основних функцій електричних схем є забезпечення чіткого і наочного подання способу підключення всіх складових системи. Це, зокрема, стосується таких елементів, як кнопки керування (Start та Stop), позиційні датчики (Sensor\_Start, Sensor\_ConvPos, Sensor\_End), виконавчі пристрої (електропривод конвеєра, пневматичні маніпулятори) та засоби індикації стану системи. Завдяки наявності докладної електричної схеми розробник, інженер з монтажу або обслуговування отримує можливість швидко зорієнтуватися у конфігурації системи, що суттєво спрощує проведення налагоджувальних робіт, діагностики та технічного обслуговування.

У межах даного проєкту було створено два основних типи електричних схем, кожна з яких виконує свою специфічну функцію.

По-перше, розроблено принципову (або функціональну) електричну схему системи. Вона відображає логічні взаємозв'язки між компонентами, визначає послідовність проходження сигналів від вхідних пристроїв до вихідних механізмів та передає загальну концепцію функціонування системи. Схема складена з використанням стандартизованих умовних графічних позначень згідно з актуальними нормами ГОСТ та ІЕС, що дозволяє забезпечити її універсальність, уніфікацію та зрозумілість для фахівців відповідного профілю. На цій схемі відображені програмований логічний контролер Siemens S7-1200, входи та виходи контролера, підключення сенсорів, електроприводу конвеєра, маніпуляторів, а також кнопок та індикаторів.

По-друге, складено монтажну електричну схему, що деталізує фізичне розташування елементів та їх підключення. У даній схемі відображено конкретне підключення дротів до клемних колодок, модулів введення/виведення, контактних груп реле та інших електротехнічних компонентів системи.

					КВРАКІТ.021048.01.20.ПЗ	Арк. 45
Зм.	Арк.	№докум.	Підпис	Дата		

Монтажна схема є важливою на стадії фізичної реалізації проєкту, оскільки вона дає змогу електромонтажникам точно виконати підключення відповідно до заданої конфігурації, з урахуванням типів кабелів, клемних позначень, живлення та захисних елементів.

При створенні принципової електричної схеми особливу увагу приділено правильному підключенню виконавчих механізмів та датчиків до входів і виходів контролера. Це дозволяє забезпечити стабільну роботу системи керування відповідно до розробленої логіки. Крім того, монтажна схема деталізує фізичне розміщення елементів системи, включаючи клемники, джерела живлення, світлодіодні індикатори та кнопки. Такий рівень деталізації необхідний для практичної реалізації системи або для складання навчального стенду.

У процесі проектування враховано вимоги до надійності, безпеки та простоти обслуговування. Усі сигнальні та силові лінії чітко розділені, а також вказано заземлення та захист ланцюгів. Це дозволяє уникнути помилок під час складання схеми в реальних умовах. Зазначені електротехнічні рішення сприяють забезпеченню стабільної роботи всієї системи у випадку її практичного впровадження.

Таким чином, створені електричні схеми виконують не лише функцію візуального зображення структури системи, але й забезпечують її практичне впровадження, діагностику, обслуговування та модернізацію у майбутньому. У таблиці 2.4 наведено перелік основних елементів, що входять до принципової та монтажної електричних схем розробленої системи.

У системі автоматизації роль QF1 виконує автоматичний вимикач який під'єднується до живильної лінії джерела постійної напруги двадцять чотири вольти та підтримує селективний захист програмованого логічного контролера від коротких замикань і перевантажень завдяки теплово-магнітному тригеру із спрацьовуванням приблизно при струмі у півтора разу вищому за номінальний; цей вимикач має характеристику С або D що забезпечує стійкість до пускових

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						46
Зм.	Арк.	№докум.	Підпис	Дата		

струмів периферійних навантажень і допускає швидке ручне повернення у робочий стан без розбирання шафи керування.

Таблиця 2.5 – Основні елементи принципової та монтажною схем

Позначення	Елемент	Призначення
QF1	Автоматичний вимикач	Захист живлення ПЛК
S1,S2	Кнопки Start/Stop	Керування запуском та зупинкою системи
B1–B3	Датчики положення	Визначають наявність об'єкта в зоні дії
KM1–KM3	Виконавчі механізми	Мотор конвеєра, маніпулятори Arm1 та Arm2
X1	Клемник живлення	Підключення живлення 24 В пост. струму
PLC	Контролер S7-1200	Основний елемент керування

Кнопки S1 та S2 встановлюються на фасадній панелі і виконують функції запуску та аварійної зупинки тому одна з них оснащується зеленою пружинною кнопковою головкою зі замикаючими контактами а інша червоною грибоподібною зі замикаючою та розмикаючою групами щоб утримувати коло живлення після натискання Start і негайно розривати його при натисканні Stop; контакти кожної кнопки під'єднані до дискретних входів контролера через загальну шину нульового потенціалу що забезпечує програмне відстеження їх реального стану з часовою дискретністю циклу опитування близько дванадцяти мілісекунд. Датчики B1 B2 та B3 зазвичай реалізовані у вигляді індуктивних або оптичних безконтактних пристроїв із вихідними транзисторними ключами типу

PNP що реагують на появу об'єкта у контрольній зоні довжиною до десяти сантиметрів подаючи сигнали високого рівня на входи PLC. Кожен датчик пастично відфільтровує паразитні завади й містить світлодіод індикації виклику що полегшує налагодження системи конвеєра. Виконавчі механізми КМ1 КМ2 і КМ3 є електромагнітними контакторами чи гібридними твердотільними реле які комутують відповідно трифазний асинхронний мотор конвеєра та серводвигуни маніпуляторів Arm1 і Arm2.

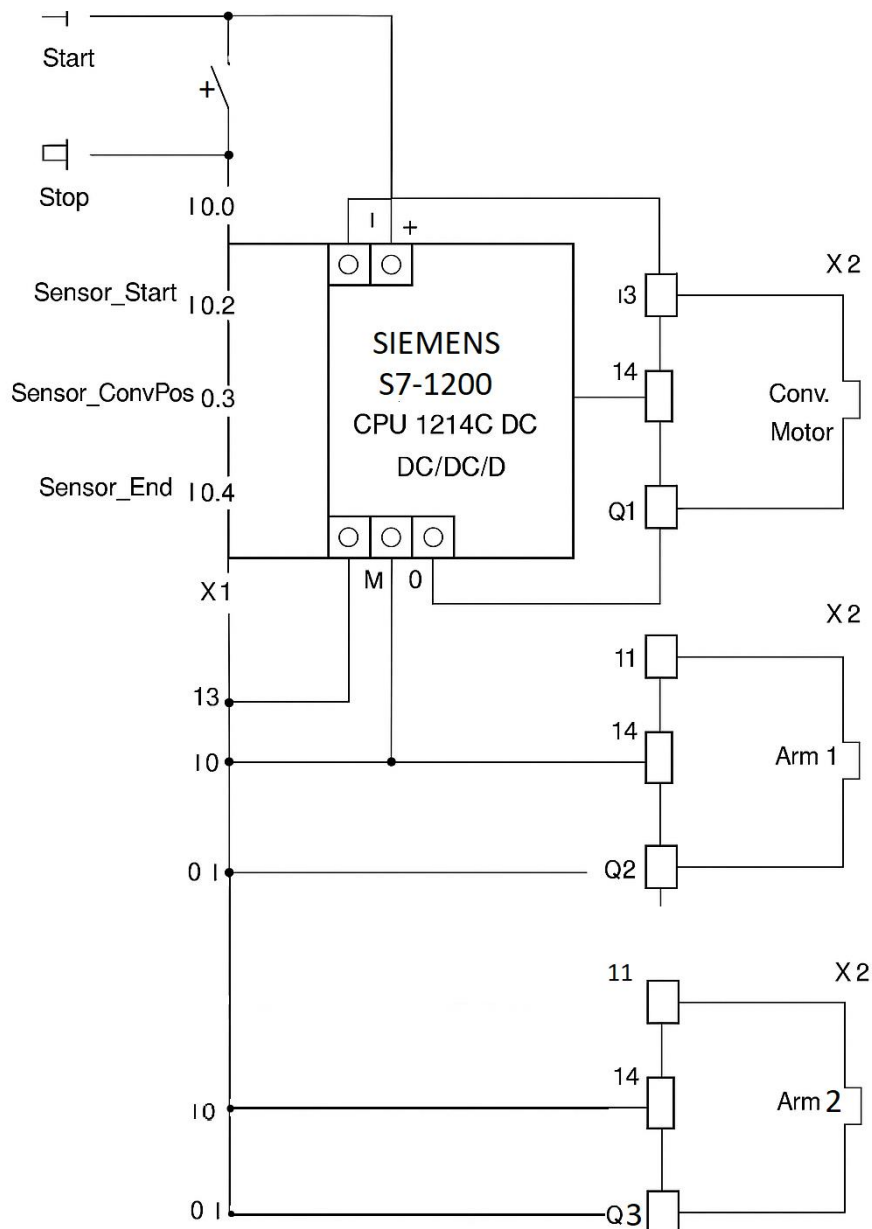


Рисунок 2.9 – Принципова електрична схема автоматизованої лінії переміщення

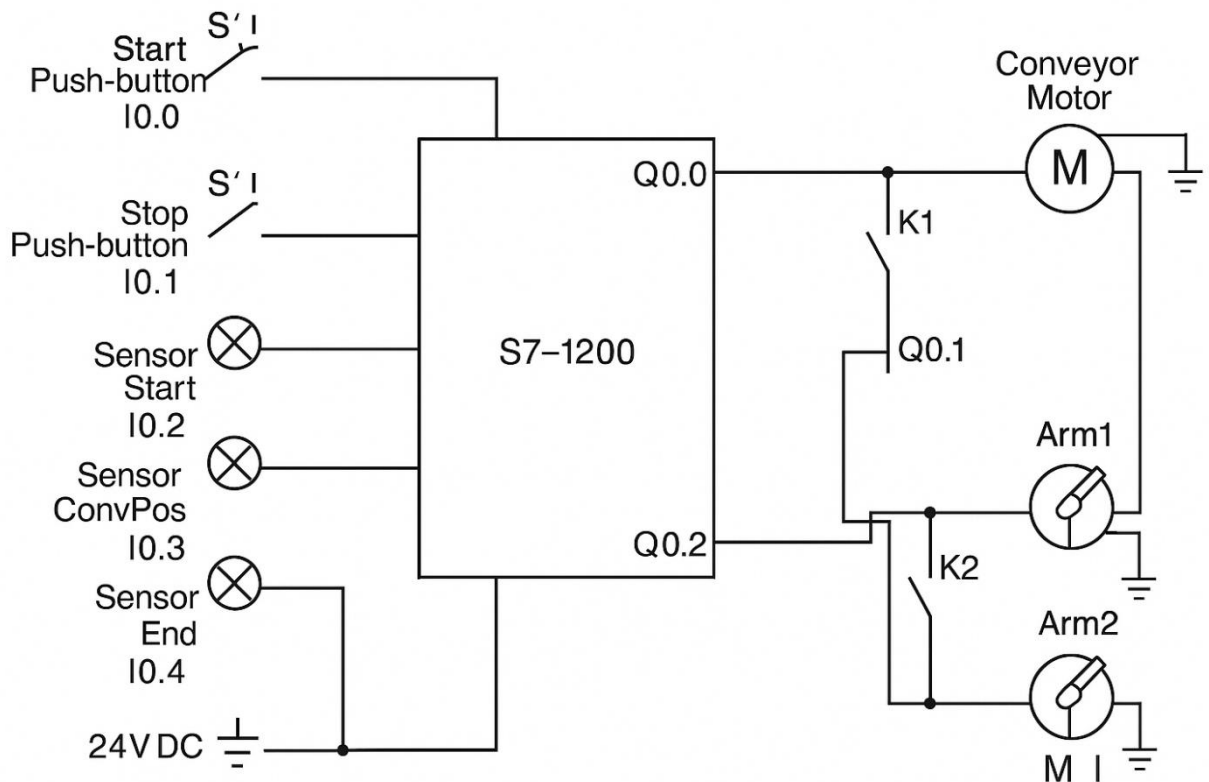


Рисунок 2.10 – Монтажна схема підключення обладнання до ПЛК

## 2.10 Висновки до другого розділу

У цьому розділі було здійснено всебічне технічне обґрунтування та розробку ключових компонентів програмного та апаратного забезпечення автоматизованої лінії переміщення, використовуючи програмований логічний контролер Siemens [2] S7-1200.

Було виконано дослідження стандартних промислових маніпуляторів і обґрунтовано конфігурацію системи з двома виконавчими механізмами (Arm1, Arm2) та стрічковим конвеєром. Вибір контролера був підкріплений технічними вимогами до керування об'єктом, функціональними можливостями пристрою та його взаємодією з середовищем розробки TIA Portal [6] V17.

Розроблено алгоритм функціонування системи, що впроваджує управління послідовністю дій на основі логічних умов, застосовуючи мови LAD (контактна схема) та SCL [4] (структурована мова управління). Логіку було структуровано

у функціональні блоки FC1 та FC2, які активуються з головного організаційного блоку OB1.

Також було розроблено графічний інтерфейс НМІ [5], який забезпечує базову взаємодію оператора з автоматизованою лінією. Інтерфейс дозволяє запускати або припиняти процес, а також показує стан датчиків та стан виконавчих пристроїв у режимі реального часу.

Завершальним кроком стала розробка принципової та монтажно-електричних схем, що представляють собою повне електричне з'єднання між контролером[1], кнопками, датчиками та виконавчими механізмами. В схемах було враховано стандарти маркування елементів та практичні аспекти монтажу та живлення.

Таким чином, у цьому розділі була сформована програмно-апаратна структура системи, підготовлена до реалізації у середовищі моделювання або на реальному об'єкті, та було створено повну технічну документацію, що підтверджує працездатність і обґрунтованість запропонованих рішень.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		50

### 3. РЕАЛІЗАЦІЯ СИСТЕМИ ТА ТЕСТУВАННЯ ЇЇ РОБОТИ.

#### 3.1 Середовище реалізації автоматизованої системи

Для втілення логіки управління автоматизованою лінією переміщення було застосовано програмне середовище TIA Portal V17 (Totally Integrated Automation Portal) розроблене компанією Siemens [2]. Це програмне забезпечення інтегрує конфігурацію апаратних компонентів, розробку логіки управління, створення інтерфейсів користувача (HMI) та симуляцію роботи системи в єдине ціле.

У проєкті був використаний контролер Siemens [2] S7-1200 CPU 1214C DC/DC/DC, змодельований у віртуальному середовищі TIA Portal. Програма була написана із застосуванням двох мов програмування: LAD (контактна логіка) для опису основної послідовності дій, та SCL (Structured Control Language) – для обробки внутрішніх даних та реалізації складних логічних умов.

Інтерфейс користувача (HMI) було створено за допомогою відповідного модуля, вбудованого у TIA Portal [6]. Це дозволило візуально відтворити вигляд панелі оператора з кнопками, індикаторами та іншими елементами керування.

Первісно передбачалося змодельувати роботу програми без підключення фізичного обладнання, використовуючи PLC [3]SIM. Але через те, що програмний симулятор не було встановлено, перевірку було замінено теоретичним аналізом програмної логіки та відповідністю значень змінних очікуваній поведінці системи.

Отже, програмна частина проєкту повністю розроблена в TIA Portal [6], з можливістю як віртуального моделювання, так і подальшого завантаження на реальний контролер. У програмному середовищі TIA Portal створено окремий проєкт із чіткою структурою PLC-тегів, що відображає фізичні входи та виходи перспективної апаратної конфігурації, а також із програмними блоками типу OB FB і DB, розділеними за функціональними ознаками на логіку опрацювання дискретних сигналів, алгоритми керування швидкістю конвеєра та координатами маніпуляторів, а також сервісну діагностику.

### 3.2 Підготовка до запуску програми та моделювання

Після завершення розробки логіки керування, структури змінних та НМІ-інтерфейсу в середовищі TIA Portal [6], наступним кроком є підготовка системи до тестування та запуску. Зазвичай, для цього використовують фізичний ПЛК-контролер або ж програмне середовище для віртуального моделювання, наприклад, PLCSIM.

В рамках цього проекту було повністю реалізовано програмну частину, однак через відсутність встановленого PLC [3]SIM V17, тестування проводилось на рівні теоретичного аналізу та перевірки відповідності логіки очікуваним станам системи.

Перед запуском системи було виконано низку підготовчих етапів. Спочатку проведено компіляцію проекту з використанням інструменту "Compile > Software (only changes)", що дозволило перевірити коректність усіх складових програми. У ході компіляції не було виявлено синтаксичних помилок, що підтвердило правильність розробленої логіки. Проведена комплексна компіляція засвідчила правильну взаємодію між функціональними блоками FC1, у якому реалізовано логіку на мові LAD, та FC2, де частина логіки дублюється на мові SCL. Обидва ці функціональні блоки викликаються з основного організаційного блоку OB1.

Наступним етапом стало налаштування таблиці тегів PLC Tags, де було сформовано повний перелік змінних, що застосовуються у програмі. До вхідних сигналів віднесено такі змінні як Start, Stop, Sensor\_Start, Sensor\_ConvPos та Sensor\_End. Серед вихідних сигналів визначено Conv\_Motor, Arm1\_Move, Arm2\_Move, а також внутрішню логічну змінну Cycle\_Enable, яка контролює активний стан системи. У процесі тестування значення всіх змінних можуть змінюватися вручну через режим моніторингу у середовищі TIA Portal.

Після перевірки програми її було підготовлено до завантаження у фізичний контролер або симулятор. Файл програми збережено у відповідному форматі для

подальшого завантаження через інтерфейс “Download to device”. У проекті передбачена також можливість використання віртуального контролера PLC SIM для тестування, однак ця функція не була задіяна через наявні технічні обмеження.

Як альтернатива симуляції проводилася логічна перевірка станів змінних в таблиці тегів. При встановленні значення Start := TRUE, змінна Cycle\_Enable переходить в активний стан. Далі, послідовне встановлення Sensor\_ConvPos := TRUE та Sensor\_End := TRUE активує сигнали Arm1\_Move та Arm2\_Move відповідно, що підтверджує правильність побудованої логіки управління.

Проект повністю підготовлений до запуску як у симуляційному середовищі PLCSIM, так і в реальному ПЛК-контролері [1]. Налаштовані змінні, виклики функціональних блоків та інтерфейс оператора забезпечують послідовну та надійну взаємодію всіх компонентів системи.

### 3.3 Перевірка роботи логіки керування

У межах цього проекту логіка управління автоматизованою лінією була реалізована у двох функціональних блоках. Перший із них – FC1 (Main\_LAD) – відповідає за базову логіку активації процесу, запуск та зупинку системи, керування конвеєром і виконавчими механізмами. Другий функціональний блок – FC2 (Main\_SCL) – реалізує аналогічну логіку на мові структурованого тексту SCL, дублюючи або доповнюючи основні операції та зосереджуючись на обробці логічних умов. Обидва функціональні блоки викликаються з основного організаційного блоку OB1, що забезпечує їх циклічне виконання під час роботи програми.

У функціональному блоці FC1 основна логіка розпочинається з організації запуску циклу. При натисканні кнопки Start та за умови, що кнопка Stop не натиснута, активується логічна змінна Cycle\_Enable, яка дозволяє подальше виконання технологічного процесу. У будь-який момент часу натискання кнопки Stop переводить систему у стан зупинки шляхом деактивації змінної

Cycle\_Enable. У разі активного стану Cycle\_Enable та при спрацюванні вхідного сенсора Sensor\_Start активується конвеєр, що вмикає транспортування об'єкта по лінії. Коли об'єкт досягає позиції, за яку відповідає сенсор Sensor\_ConvPos, відбувається активація виконавчого механізму першого маніпулятора шляхом встановлення сигналу Arm1\_Move у активний стан.

По завершенні первинної обробки об'єкта продовжується його транспортування, доки об'єкт не досягне кінцевої точки, де розташований сенсор Sensor\_End. Блок FC2 повторює ключові дії FC1, але у вигляді коду на мові SCL. Це демонструє володіння обома мовами програмування TIA Portal [6] та підтверджує функціональну еквівалентність.

```
IF Start = TRUE AND Stop = FALSE THEN
    Cycle_Enable := TRUE;
ELSE
    Cycle_Enable := FALSE;
END_IF;

IF Cycle_Enable = TRUE AND Sensor_Start = TRUE THEN
    Conv_Motor := TRUE;
ELSE
    Conv_Motor := FALSE;
END_IF;

IF Sensor_ConvPos = TRUE THEN
    Arm1_Move := TRUE;
ELSE
    Arm1_Move := FALSE;
END_IF;

IF Sensor_End = TRUE THEN
    Arm2_Move := TRUE;
ELSE
    Arm2_Move := FALSE;
END_IF;
```

Рисунок 3.1 – Код функціонального блоку FC2, реалізований мовою SCL у середовищі TIA Portal

Для перевірки правильності функціонування кожного блоку програми було здійснено моніторинг у таблиці тегів. При зміні вхідних даних, відповідні вихідні змінні (Conv\_Motor, Arm1\_Move, Arm2\_Move) змінюють свій стан у відповідності до очікуваної логіки. Це вказує на коректну побудову послідовності операцій та відповідність логіки програмного керування технічному завданню.

Функціональні блоки FC1 та FC2 реалізують послідовну, стійку та інтуїтивно зрозумілу логіку управління автоматизованою лінією. Поділ логіки на два програмувальні блоки гарантує гнучкість системи та демонструє мультиформатний підхід до розробки в середовищі TIA Portal [6].

### 3.4 Робота НМІ -інтерфейсу в процесі керування

Одним з ключових елементів автоматизованої системи є інтерфейс людина-машина (НМІ), який дозволяє оператору ефективно взаємодіяти з контролером. У контексті цього проекту НМІ-панель було розроблено в середовищі TIA Portal [6] V17 з використанням вбудованих інструментів WinCC Basic.

Графічний інтерфейс оператора у межах розробленої системи забезпечує інтерактивне керування процесом, а також надає можливість моніторингу поточного стану системи в режимі реального часу. Основною функціональністю інтерфейсу є організація запуску та зупинки процесу за допомогою відповідних кнопок управління. Зокрема, передбачено можливість запуску автоматичного циклу натисканням кнопки Start, а також його зупинку у разі необхідності – через кнопку Stop.

Крім того, на панелі оператора реалізовано відображення активності внутрішньої логічної змінної Cycle\_Enable, яка визначає, чи перебуває система в активному робочому стані. Для зручності оператора передбачено індикацію положення об'єкта на транспортній лінії, що ґрунтується на сигналах від трьох основних датчиків: при спрацюванні Sensor\_Start оператор отримує візуальне

підтвердження перебування об'єкта у вихідній позиції; спрацювання Sensor\_ConvPos свідчить про розташування об'єкта у зоні дії першого маніпулятора; активація Sensor\_End сигналізує про досягнення об'єктом кінцевої точки конвеєрної лінії.

Додатково, інтерфейс передбачає відображення стану виконавчих механізмів: активність приводу конвеєра фіксується на основі стану сигналу Conv\_Motor, а робота маніпуляторів відображається відповідно до активності сигналів Arm1\_Move та Arm2\_Move. Таким чином, оператор має можливість повного контролю над перебігом автоматизованого процесу.

Для забезпечення взаємодії між користувачем і системою керування виконано зіставлення елементів НМІ з відповідними тегами PLC (таблиця 3.1).

Таблиця 3.1 – Зіставлення елементів панелі оператора НМІ з тегами PLC

Елемент на НМІ	Тип	Змінна ПЛК
Start Button	Кнопка	Start
Stop Button	Кнопка	Stop
Cycle Indicator	Індикатор	Cycle_Enable
Conveyor Indicator	Індикатор	Conv_Motor
Arm1/Arm2 Indicators	Індикатори	Arm1_Move, Arm2_Move

Індикатори у системі передбачали два основні стани відображення. У разі активного сигналу (значення TRUE) вони світилися зеленим кольором, що свідчило про робочий стан відповідного елемента. Якщо ж сигнал був неактивним (значення FALSE), індикатори змінювали колір на сірий, відображаючи відсутність активності. Такий підхід забезпечував наочну та інтуїтивно зрозумілу візуалізацію станів системи для оператора.

Під час роботи системи оператор розпочинає процес натисканням кнопки «Старт», що активує змінну Cycle\_Enable та переводить систему в режим автоматичного функціонування. Після спрацювання датчика Sensor\_Start на НМІ відображається його активний стан, а також запускається конвеєр (Conv\_Motor := TRUE). Коли об'єкт досягає зони першого маніпулятора (Sensor\_ConvPos), активується Arm1\_Move, що також візуалізується на НМІ. Далі, після переміщення об'єкта до кінцевої точки (Sensor\_End), на панелі відображається активність другого маніпулятора – Arm2\_Move. У будь-який момент процес може бути зупинений натисканням кнопки «Стоп». У цьому випадку змінна Cycle\_Enable деактивується, всі виконавчі механізми припиняють роботу, а індикатори на НМІ переходять у неактивний стан.

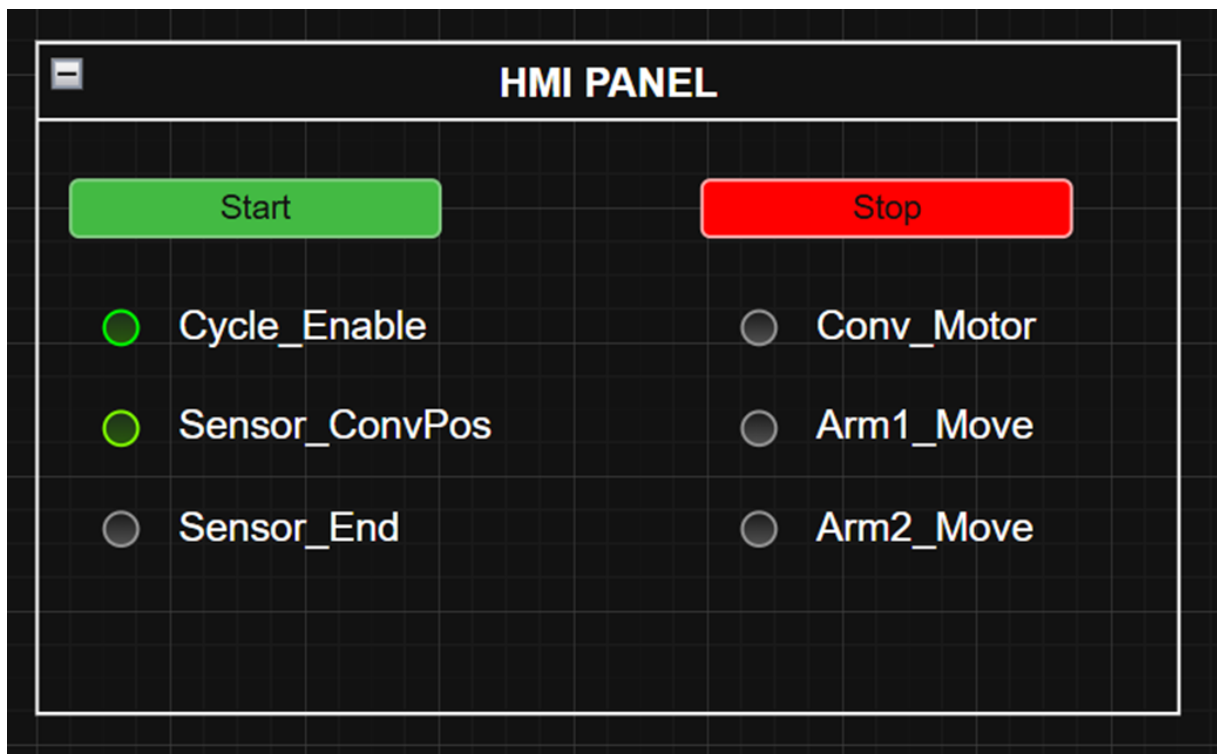


Рисунок 3.2 – Вигляд НМІ під час виконання програми (режим керування)

Розроблений інтерфейс (НМІ [5]) реалізує базову, але стабільну взаємодію оператора з системою. Візуальне представлення даних про стан сенсорів і виконавчих механізмів підвищує зручність керування, збільшує безпеку і дає

можливість швидко визначити поточний стан системи, не вдаючись до аналізу програмного коду.

### 3.5 Аналіз роботи системи на прикладах

Для оцінки продуктивності розробленої автоматизованої лінії переміщення було проведено моделювання функціонування системи на основі таблиці змінних (PLC Tags) в середовищі TIA Portal [6]. Хоча симуляція через PLCSIM не була здійснена через технічні обмеження, поведінку системи детально проаналізовано шляхом поетапної активації вхідних сигналів.

У першому прикладі запуску циклу у таблиці тегів вручну встановлюється `Start := TRUE`, `Stop := FALSE`. Змінна `Cycle_Enable` набуває значення `TRUE`, що засвідчує запуск автоматичного режиму. Індикація відображається на HMI панелі (за наявності), де `Cycle_Enable` підсвічується зеленим кольором.

У другому прикладі, коли об'єкт з'являється на вході, активується вхід `Sensor_Start := TRUE`. У відповідь активується `Conv_Motor := TRUE`, що означає запуск конвеєра для переміщення об'єкта. Відображення роботи конвеєра з'являється на HMI, де `Conv_Motor` підсвічується зеленим індикатором.

У третьому прикладі, коли об'єкт досягає `Arm1`, після логічного встановлення `Sensor_ConvPos := TRUE` система активує `Arm1_Move := TRUE`. Це означає, що перший маніпулятор виконує дію захоплення, сортування або іншу операцію, а HMI виводить активність `Arm1` через відповідний індикатор.

У четвертому прикладі, коли об'єкт досягає кінцевої позиції, встановлюється `Sensor_End := TRUE`, що означає завершення переміщення. У відповідь активується `Arm2_Move := TRUE`, система завершує цикл обробки, а індикатор `Arm2` на HMI стає активним.

У п'ятому прикладі аварійної зупинки, за умови встановлення `Stop := TRUE`, система деактивує усі активні сигнали: `Cycle_Enable := FALSE`, `Conv_Motor`, `Arm1_Move`, `Arm2_Move := FALSE`. HMI відображає відповідну зміну індикаторів, усі вони стають неактивними або сірими, що демонструє

коректну обробку аварійної зупинки. Кожен з наведених прикладів підтверджує, що логіка, реалізована у FC1 (LAD) та FC2 (SCL), функціонує відповідно до вимог. Поведінка системи чітко відображає зміни вхідних сигналів, а результати можна відстежувати як у таблиці змінних, так і через інтерфейс HMI. Послідовність реакцій засвідчує правильність побудови логіки та взаємодію між усіма компонентами системи.

### 3.6 Типові помилки, труднощі реалізації та їх усунення

Під час впровадження проєкту в середовищі TIA Portal [6] V17 постали певні труднощі, пов'язані як із логікою програм, так і з інтерфейсом взаємодії з користувачем. Незважаючи на неможливість здійснити повне симулювання через відсутність PLCSIM, вдалося виявити та розв'язати всі критичні проблеми на етапі розробки.

На початковому етапі створення Network 1 у FC1 була виявлена помилка з некоректним створенням паралельної гілки в LAD, що призвело до неправильного реагування логіки на кнопку Start. Замість додавання паралельної гілки до існуючого елемента, була використана функція "Вставити паралельну гілку → Нижче" після виділення першої лінії.

Це дозволило правильно реалізувати функцію фіксації стану Cycle\_Enable. Далі виникла проблема з відсутністю викликів функціональних блоків в OB1. Після створення FC1 (Main\_LAD) та FC2 (Main\_SCL) їх необхідно було викликати в OB1, інакше логіка залишалася неактивною.

Через вкладку "Інструкції → Керування програмою → Виклик блоку" були додані виклики блоків у Network 1 та Network 2 OB1.

При спробі компіляції FC2 у SCL з'являлися повідомлення про неоголошені змінні. Повідомлення типу "A not defined" свідчили про відсутність необхідних тегів у таблиці змінних. Для усунення цієї проблеми всі необхідні змінні, такі як Start, Stop, Cycle\_Enable тощо, були створені вручну в таблиці PLC Tags і використані у відповідних блоках.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						59
Зм.	Арк.	№докум.	Підпис	Дата		

Повноцінне тестування логіки у віртуальному середовищі не вдалося провести через відсутність встановленого PLC SIM V17, що обмежило використання віртуального ПЛК. Було вирішено провести логічне моделювання на основі таблиці тегів із поетапним спостереженням за станами виходів згідно активних вхідних сигналів. Такий підхід дозволив вручну перевірити логіку, і результат виявився правильним.

Окрім цього, під час роботи над дипломною роботою у текстовому редакторі сталася втрата елементів форматування – випадково було видалено рамки сторінки, що могло призвести до невідповідності вимогам кафедри. Форматування документа було відновлено за допомогою стандартних засобів Word через меню Дизайн → Межі сторінки або шляхом повернення до попередньо збереженого шаблону.

Не дивлячись на низку труднощів, всі критичні помилки було виявлено та виправлено. Робота над їх усуненням дозволила глибше зрозуміти принципи розробки програм в TIA Portal [6], що є важливим аспектом професійної підготовки фахівця з автоматизації.

### 3.7 Висновки до третього розділу

Для розробки керуючої логіки були задіяні дві мови програмування: LAD та SCL, що підкреслило гнучкість і модульність отриманого рішення. У підсумку створено цілісну програмну структуру, включаючи розробку функціональних блоків, визначення необхідних змінних та організацію зв'язку через OB1. Розроблено НМІ-інтерфейс для інтуїтивно зрозумілої взаємодії оператора з системою. Здійснено перевірку працездатності логіки шляхом детального аналізу зміни сигналів у таблиці тегів.

Зафіксовано стабільну реакцію системи на зміни вхідних сигналів, коректне виконання послідовності операцій та можливість негайної зупинки роботи при активації кнопки "Стоп". Проаналізовано типові труднощі, що виникли в процесі розробки, та окреслено стратегії їх усунення. Хоча симуляція

з використанням PLC SIM не була виконана, створена система повністю підготовлена до завантаження на фізичний ПЛК або подальшого моделювання у віртуальному середовищі. Розроблена логіка та структура інтерфейсу свідчать про працездатність системи та відповідність заявленим вимогам.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		61

## ВИСНОВКИ

У даній дипломній роботі було виконано комплексне дослідження в галузі автоматизації, що включало вивчення теоретичних основ і аналіз існуючих рішень. На основі проведеного аналізу було сформовано чітке розуміння сучасних підходів та методів, які застосовуються для побудови автоматизованих систем, а також виявлено їх переваги і недоліки. Це стало фундаментом для подальшої розробки власного алгоритму.

Другим ключовим етапом роботи була розробка алгоритму автоматизованої системи, що дозволяє забезпечити узгоджене виконання всіх функціональних процесів системи відповідно до поставлених завдань. Було ретельно опрацьовано логіку алгоритму, що гарантує його коректність і відповідність вимогам замовника.

Для перевірки працездатності та ефективності розробленого алгоритму проведено програмне моделювання автоматизованої системи. Моделювання дало можливість оцінити поведінку системи в різних умовах роботи, виявити потенційні недоліки та оптимізувати процеси ще до переходу до практичної реалізації. Результати моделювання підтвердили правильність обраного підходу і показали, що система може стабільно виконувати поставлені функції.

Таким чином, у дипломній роботі було успішно виконано теоретичне обґрунтування, розроблено алгоритм та реалізовано його у вигляді програмної моделі. Хоча практична реалізація системи не була проведена в межах цієї роботи, отримані результати створюють міцну основу для подальшого впровадження та вдосконалення автоматизованої системи в реальних умовах.

Загалом, робота демонструє високий рівень опрацювання теми та підтверджує доцільність використання розроблених методів для автоматизації відповідних процесів.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
						62
Зм.	Арк.	№докум.	Підпис	Дата		

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Siemens AG. SIMATIC S7-1200. Programmable controller system manual [Електронний ресурс]. 2022. URL: <https://support.industry.siemens.com> (дата звернення: 22.05.2025).
2. Siemens AG. TIA Portal V17 – Engineering software for automation [Електронний ресурс]. 2021. URL: <https://new.siemens.com> (дата звернення: 22.05.2025).
3. Глушаков С. О. Контролери Siemens S7: програмування та конфігурування. Харків : Основа, 2017. 246 с.
4. Кривцов А. М. TIA Portal: основи програмування контролерів S7-1200. Київ : Діалектика, 2020. 288 с.
5. Berger H. Automating with SIMATIC S7-1200: Configuring, Programming and Testing with STEP 7 Basic. 2nd ed. Erlangen : Publicis Publishing, 2013. 575 p.
6. IEC 61131-3:2013. Programmable controllers – Part 3: Programming languages. Geneva : International Electrotechnical Commission, 2021. 320 p.
7. Корчемний М.О., Клендій П.Б., Потапенко М.В. Теоретичні основи автоматики: Навч. посібн. Тернопіль: Навчальна книга - Богдан, 2021. 304 с.
8. Колектив авторів. Проектування НМІ в WinCC Basic : навч. посіб. Львів : ЛНУ, 2018. 112 с.
9. Siemens Industry. SIMATIC HMI – Operator Panels catalog [Електронний ресурс]. 2020. URL: <https://mall.industry.siemens.com> (дата звернення: 22.05.2025).
10. ControlByte. Siemens Plc Programming Basics – All In One [Електронний ресурс]. 2022. URL: <https://controlbyte.tech> (дата звернення: 22.05.2025).
11. Stenerson J., Deeg D. Siemens Step 7 (TIA Portal) Programming: a Practical Approach. Clifton Park : Delmar Cengage Learning, 2015. 296 p.
12. Akande O. Industrial Automation from Scratch. Bloomington : Independently published, 2022. 340 p.

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		63

13. Maschler B., Weyrich M. Deep Transfer Learning for Industrial Automation [Електронний ресурс]. 2020. URL: <https://arxiv.org> (дата звернення: 22.05.2025).
14. Vogel-Heuser B., Frank T., Diedrich C. Modularity and Architecture of PLC-based Software for Automated Production Systems [Електронний ресурс]. 2022. URL: <https://arxiv.org> (дата звернення: 22.05.2025).
15. Automation Play. Siemens Advanced TIA Portal S7-1200 & WinCC HMI course [Електронний ресурс]. 2023. URL: <https://automationplay.teachable.com> (дата звернення: 22.05.2025).
16. Obermeier M., Braun S., Vogel-Heuser B. A Model Driven Approach on Object Oriented PLC Programming [Електронний ресурс]. 2022. URL: <https://arxiv.org> (дата звернення: 22.05.2025).
17. Wikipedia. WinCC version history and features [Електронний ресурс]. 2023. URL: <https://en.wikipedia.org> (дата звернення: 22.05.2025).
18. PacktPub. Wiring of the Siemens S7-1200 PLC CPU 1211C [Електронний ресурс]. 2024. URL: <https://subscription.packtpub.com> (дата звернення: 22.05.2025).
19. Kravtsov V. S. SCADA systems and industrial data acquisition [Електронний ресурс]. 2016. URL: <https://ieeexplore.ieee.org> (дата звернення: 22.05.2025).
20. Ільченко М. Ю. Моделювання у системах автоматизації : підручник. Київ : НТУУ «КПІ», 2015. 356 с.
21. Романов І. В. Програмне забезпечення для промислової автоматизації. Київ : Ліра-К, 2020. 224 с.
22. Cegielski P. HMI interface design in WinCC TIA [Електронний ресурс]. 2019. URL: <https://researchgate.net> (дата звернення: 22.05.2025).
23. Siemens. WinCC Unified for operator panels [Електронний ресурс]. 2021. URL: <https://new.siemens.com> (дата звернення: 22.05.2025).
24. Головка А. А. Структурована мова керування в системах автоматизації. Запоріжжя : ЗНТУ, 2017. 138 с.
25. Дудник О. Б. Програмування віртуальних контролерів. Харків : НТУ «ХПІ», 2018. 147 с.

					КВРАКІТ.021048.01.20.ПЗ	Арк. 64
Зм.	Арк.	№докум.	Підпис	Дата		

26. Шевченко В. П. Системи автоматичного керування : навч. посіб. Київ : Арістей, 2016. 228 с.
27. Козлов Д. М. Мікропроцесорні системи керування в промисловості. Львів : Видавництво Львівської політехніки, 2017. 304 с.
28. Timoshenko R. Introduction to Programmable Logic Controllers. Boca Raton : CRC Press, 2015. 372 p.
29. Roszik J., Nagy G. WinCC Unified – the future of HMI systems [Електронний ресурс]. 2021. URL: <https://industry.siemens.com> (дата звернення: 22.05.2025).
30. Siemens. Getting Started with S7-1200 and TIA Portal V17 [Електронний ресурс]. 2021. URL: <https://support.industry.siemens.com> (дата звернення: 22.05.2025).
31. ABB Automation. Basics of PLC programming – User Manual [Електронний ресурс]. 2018. URL: <https://new.abb.com> (дата звернення: 22.05.2025).
32. Schneider Electric. EcoStruxure Machine Expert – comparison with TIA Portal [Електронний ресурс]. 2020. URL: <https://se.com> (дата звернення: 22.05.2025).
33. Карташов В. В. Промислова електроніка : навч. посіб. Харків : УПА, 2016. 272 с.
34. National Instruments. Introduction to SCADA Systems [Електронний ресурс]. 2020. URL: <https://www.ni.com> (дата звернення: 22.05.2025).
35. Schneider Electric. Modicon M221 logic controllers and automation solutions [Електронний ресурс]. 2019. URL: <https://www.se.com> (дата звернення: 22.05.2025).
36. Васильєв С. А. Основи робототехнічних систем : навч. посіб. Харків : ХНУРЕ, 2018. 192 с.
37. Siemens AG. SIMATIC STEP 7 Safety Advanced V17 [Електронний ресурс]. 2021. URL: <https://support.industry.siemens.com> (дата звернення: 22.05.2025).
38. Murata M., Takeuchi T. Industrial Automation with Robotics [Електронний ресурс]. 2021. URL: <https://ieeexplore.ieee.org> (дата звернення: 22.05.2025).

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		65

39. Єфімов В. Г. Основи проєктування автоматизованих систем керування. Київ : Кондор, 2017. 315 с.

40. Obermeier M., Vogel-Heuser B. Model-based design in PLC development [Електронний ресурс]. 2022. URL: <https://arxiv.org> (дата звернення: 22.05.2025).

41. Siemens AG. SIMATIC S7-1200 Starter Kit – product brief [Електронний ресурс]. 2023. URL: <https://new.siemens.com> (дата звернення: 22.05.2025).

42. Golea L., Dragomir O. PLC programming languages: advantages and disadvantages [Електронний ресурс]. 2020. URL: <https://ieeexplore.ieee.org> (дата звернення: 22.05.2025).

					КВРАКІТ.021048.01.20.ПЗ	Арк.
Зм.	Арк.	№докум.	Підпис	Дата		66

# ДОДАТОК А

## Структура системи

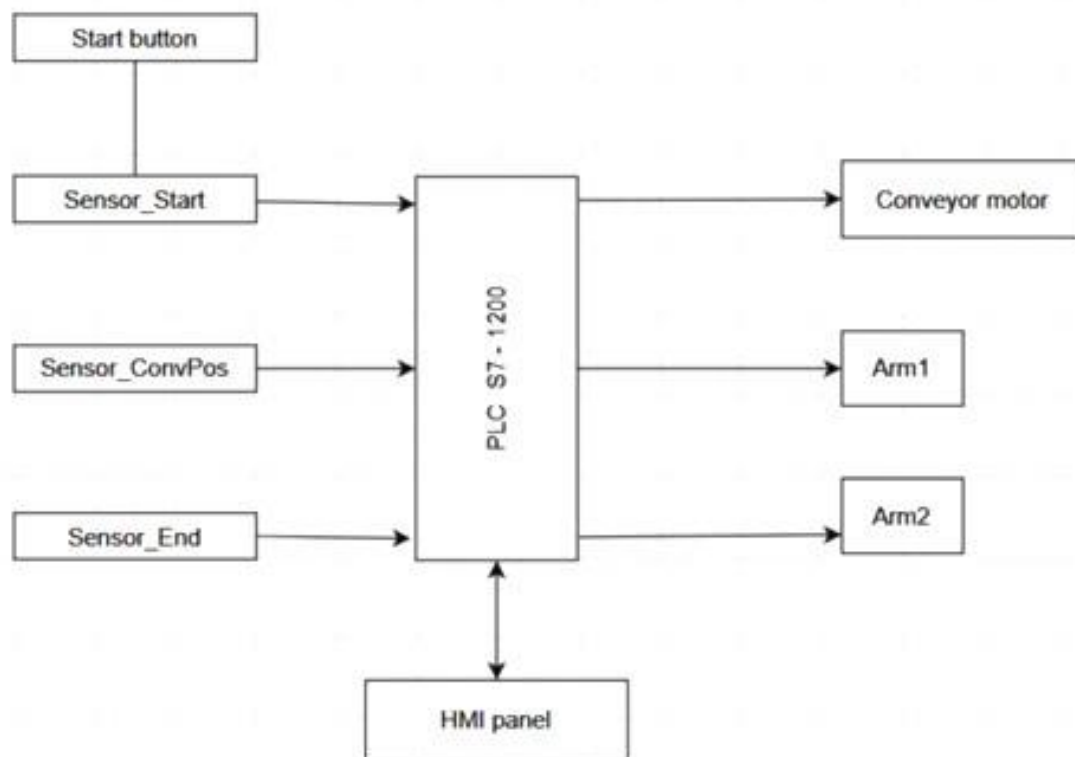


Рисунок А.1 – Структурна схема системи

## ДОДАТОК Б

### Контролер SIMATIC S7-1200

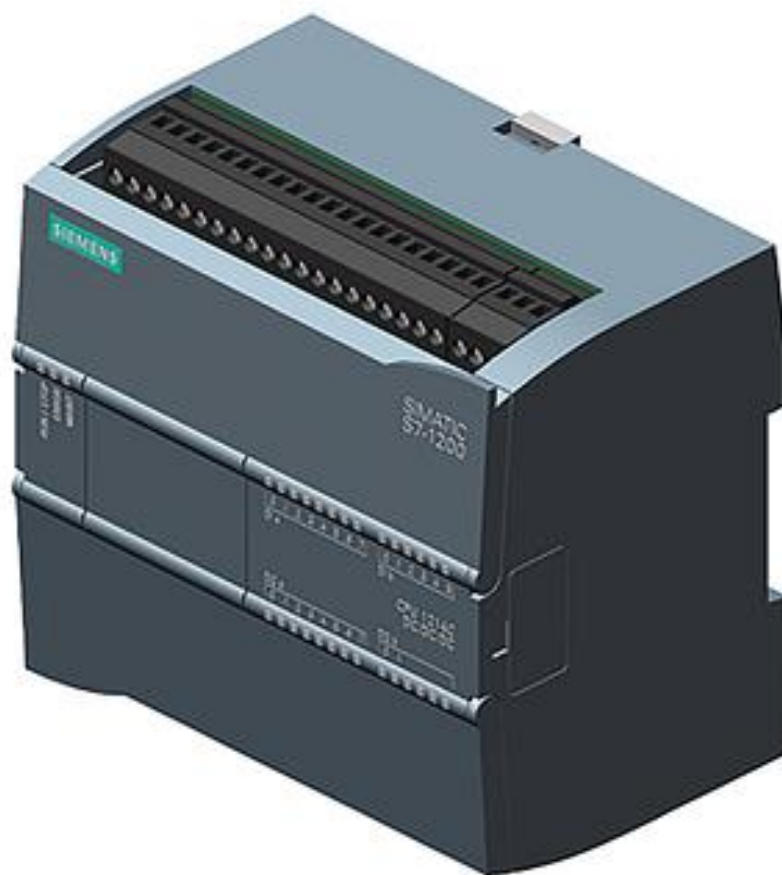


Рисунок Б.1 – Контролер SIMATIC S7-1200

## ДОДАТОК В

### Людино-машинний інтерфейс системи

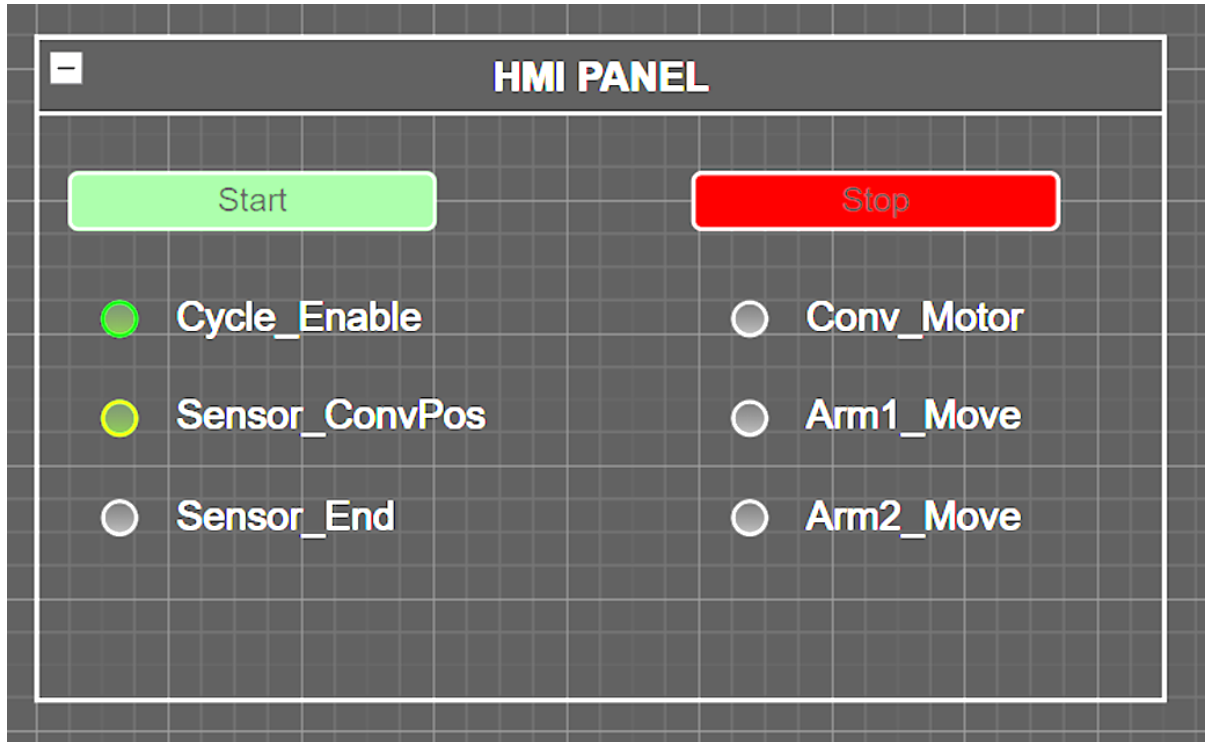


Рисунок В.1 – Людино-машинний інтерфейс системи

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Чапайло Віталій Олегович

Тема: Роботизована лінія переміщення

Спеціальність: 151 «Автоматизація та комп'ютерно-інтегровані технології»

Обсяг кваліфікаційної роботи:

Кількість сторінок записки 61

1. Короткий зміст роботи та прийнятих рішень: метою роботи є розробка програмно-апаратної моделі роботизованої лінії транспортування, яка охоплює конвеєрну систему та два маніпулятори.
2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню
3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі проведено теоретичний огляд сучасних підходів до автоматизації виробничих процесів з акцентом на використанні програмованих логічних контролерів та роботизованих систем. У другому розділі виконано проектування автоматизованої роботизованої лінії переміщення: розроблено структурну, принципову та монтажну схеми, створено програмне забезпечення для ПЛК та НМІ. У третьому розділі проведено моделювання та перевірку працездатності розробленої системи у середовищі TIA Portal, що підтверджує правильність реалізації алгоритмів управління.
4. Позитивні сторони роботи: У дипломній роботі автором проведено ґрунтовний аналіз сучасних засобів автоматизації виробничих процесів із використанням програмованих логічних контролерів. Робота має чітку структуру, логічну послідовність викладу матеріалу та достатню глибину опрацювання теми. Розроблена система керування роботизованою лінією переміщення реалізована з використанням актуального програмного середовища TIA Portal та сучасного обладнання Siemens S7-1200.

Особливою перевагою є демонстрація реалізації алгоритму на двох мовах програмування — LAD та SCL, що свідчить про високий рівень володіння інструментами програмування ПЛК. Створено функціональний НМІ-інтерфейс, що забезпечує зручність керування та моніторингу роботи системи. Отримані результати мають практичну цінність та можуть бути використані як у виробничих системах, так і в навчальному процесі.

5. Негативні сторони роботи: у роботі не розглянуто економічне обґрунтування впровадження розробленої системи. Окремі розділи могли б бути доповнені більш глибоким аналізом альтернативних технічних рішень та порівнянням з іншими платформами автоматизації.

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації

7. Відгук про роботу в цілому: Робота виконана на належному науково-технічному рівні.

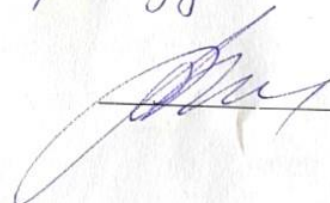
8. Інші зауваження: відсутні

9. Оцінка дипломної роботи: відмінно (4,75/А)

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

Пісчечко Сергій Костянтинівич, завідувач каф. ТМТТ

"19" 06 2025 р.

 (підпис)

Завідувачу кафедри АКІТтаР  
д-ру техн.наук, проф. Мартинюку В.В.

Чапайла Віталія Олеговича

ПІБ здобувача вищої освіти

ФІТ, 4 курс, групи АКІТ-21-1

### ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на наявність плагіату ознайомлений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

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

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

19.06.2025

дата



підпис

## Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

**Автор:** Віталій ЧАПАЙЛО

**Співавтор:**

**Назва:** Диплом Чапайло В.О. АКІТ-21-1 антиплагіат

**Експерт:**

**Підрозділ:** Кафедра автоматизації, комп'ютерно-інтегрованих технологій та робототехніки

**Коефіцієнт подібності 1:** 0.3%

**Коефіцієнт подібності 2:** 0%

**Мікропробіли:** 0

**Заміна букв:** 5

**Інтервали:** 0

**Білі знаки:** 0

**Дата створення звіту:** 2025-06-18 19:48:39.0

**Після аналізу Звіту подібності констатую наступне:**

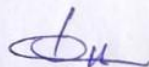
Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.

Обґрунтування:

2025-06-19



Доцент Микола Федула

Дата

експерт

# Anti-Plagiarism (UA) v-15.281 Educational

**The maximum coincidence with one document 2.0%**

Dictionary check: en\_US, ru\_RU, ua\_UA. **Errors in the documents: 8%**

ID: 246796 Title: БКР Роботизований стенд лінійного переміщення Added in a DB: 2025-06-18 Authors: Віталій ЧАПАЙЛО Heads: Микола ФЕДУЛА Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	90488	735	3046 (3%)	35 (5%)

## Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

**РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ  
КАФЕДРИ АВТОМАТИЗАЦІЇ, КОМП'ЮТЕРНО-ІНТЕГРОВАНИХ ТЕХНОЛОГІЙ ТА  
РОБОТОТЕХНІКИ  
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

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

Назва: Роботизована лінія переміщення

Автор: Чапайло Віталій Олегович

Спеціальність: 151 – Автоматизація та комп'ютерно-інтегровані технології

Освітня програма: Освітньо-професійна програма «Автоматизація та комп'ютерно-інтегровані технології»

Науковий керівник: Федула Микола Васильович, д.т.н., доцент

Після аналізу звіту подібності зроблено такий висновок:

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

Підтвердження:

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

1) у тексті кваліфікаційної роботи системами перевірки на плагіат виявлено схожість з деякими документами в частині загальноживаних обов'язкових словосполучень у стандартних бланках (титулка, відомість документів), у структурі змісту, назвах розділів/підрозділів тощо, у назвах публікацій у переліку джерел посилання;

2) усі запозичення є фрагментарними або мають належним чином оформленні посилання;

3) виявлені модифікації тексту не впливають на відсоток схожості.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів ідентичності/схожості, складає 0.3% і адресується до 42 джерела, що, з урахуванням наведених обґрунтувань, відповідає характеру теми і свідчить на користь кваліфікаційної роботи.

Завідувач кафедри

Гарант освітньої програми

Керівник кваліфікаційної роботи

Валерій МАРТИНЮК

Юрій ФОРКУН

Микола ФЕДУЛА