

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет електроенерготехніки та автоматики

Кафедра автоматизації електромеханічних систем та електроприводу

«На правах рукопису»
УДК _____

«До захисту допущено»

Завідувач кафедри

_____ Сергій КОВБАСА
(підпис)

“ ” _____ 2024 р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою «Електромеханічні системи

автоматизації, електропривод та електромобільність»

**зі спеціальності 141 «Електроенергетика, електротехніка та
електромеханіка»**

на тему:

Лабораторний стенд для дослідження систем автоматизації на основі контролерів
Siemens Simatic S7

Виконав (-ла): студент (-ка) 2 курсу, групи ЕП-31мп

Богданов Валентин Андрійович

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник к.т.н., доцент Волянський Роман Сергійович

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

(підпис)

Консультант _____

(назва розділу)

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

Рецензент к.т.н. доцент Коваленко Михайло Анатолійович

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

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних посилань.
Студент (-ка) _____
(підпис)

Київ – 2024 р.

ВІДОМІСТЬ ДИПЛОМНОГО ПРОЕКТУ

[illegible]

				141.0002.002.МД		
	ПБ	Підп.	Дата			
Розробн.	Богданов В.А.			Відомість дипломного проекту	Лист	Листів
Керівн.	Волянський Р.С.				2	114
Реценз.	Коваленко М.А.				КПІ ім. Ігоря Сікорського Каф. АЕМС-ЕП Гр. ЕП-31мп	
Н/контр.						
Зав.каф.	Ковбаса С.М.					

Пояснювальна записка до дипломного проекту

на тему: Лабораторний стенд для дослідження систем автоматизації на основі
контролерів Siemens Simatic S7

Київ – 2024р.

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Факультет електроенерготехніки та автоматики

Кафедра автоматизації електромеханічних систем та електроприводу

Рівень вищої освіти – другий (магістерський)

Спеціальність – 141 «Електроенергетика, електротехніка та електромеханіка»

Освітньо-професійна програма – «Електромеханічні системи автоматизації, електропривод та електромобільність»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Сергій
КОВБАСА
(підпис)

«___» _____ 20__ р.

ЗАВДАННЯ
на магістерську дисертацію студенту
Богданову Валентину Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема дисертації: Лабораторний стенд для дослідження систем автоматизації
на _____ основі _____ контролерів _____ Siemens _____ Simatic _____ S7

,
науковий керівник дисертації Волянський Роман Сергійович,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «___» _____ 20__ р. № _____

2. Строк подання студентом дисертації 16.12.2024.

3. Об'єкт дослідження: лабораторний стенд для дослідження систем
автоматизації

4. _____ Вихідні _____ дані _____

5. Перелік завдань, які потрібно розробити: Аналітичний огляд літературних джерел; Формулювання вимог до розробляемого лабораторного стенду; Розробка схеми та вибір обладнання; Розробка завдань та методологічних вказівок до лабораторного стенду; Розробка старт-ап проекту.

6. Перелік графічного (ілюстративного) матеріалу

7. Орієнтовний перелік публікацій

8. Консультанти розділів дисертації*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

* Якщо визначені консультанти. Консультантом не може бути зазначено наукового керівника магістерської дисертації.

9. Дата видачі завдання 02.09.2024

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Аналітичний огляд літературних джерел	20.09.2024	
2	Формулювання вимог до розробляемого лабораторного стенду	30.09.2024	
3	Розробка схеми та вибір обладнання	10.10.2024	

4	Розробка завдань та методологічних вказівок до лабораторного стенду	20.10.2024	
5	Розробка алгоритму непрямого векторного керування. Розрахунок параметрів АД. Дослідження динамічних характеристик	25.10.2024	
6	Розробка стартап проєкту	30.10.2024	
7	Написання вступу, висновків по роботі, оформлення пояснювальної записки та графічного матеріалу.	16.12.2024	

Студент

(підпис)

Валентин БОГДАНОВ

Науковий керівник дисертації

(підпис)

Роман ВОЛЯНСЬКИЙ

РЕФЕРАТ

Магістерська дисертація складається з пояснювальної записки 114 сторінок, 25 рисунків, 24 таблиць.

У даній магістерській дисертації розглянуто розробку лабораторного стенду для дослідження систем автоматизації на основі контролерів Siemens Simatic S7. Проведено аналіз особливостей роботи програмованих логічних контролерів (ПЛК) та сформульовано вимоги до стенду.

Описано вибір обладнання, включаючи контролери серій S7-300 та S7-1200, модулі розширення, блоки живлення та допоміжні компоненти. Розроблено електричну принципову схему стенду.

Розроблено програмне забезпечення у середовищі TIA Portal для програмування ПЛК та KEPServerEX для створення інтерфейсу зв'язку. Створено моделюючу програму у MatLab/Simulink для перевірки алгоритмів автоматизації.

Запропоновано методологічні вказівки для навчальних занять, які включають задачі з програмування ПЛК, налагодження мережевих з'єднань та інтеграції SCADA-систем. Проведено тестування стенду в різних режимах роботи.

Запропоновано старт-ап проект для впровадження стенду в навчальні заклади та промислові підприємства. Визначено перспективи вдосконалення стенду та його адаптації до сучасних потреб автоматизації.

АВТОМАТИЗАЦІЯ, ПЛК, SIEMENS, ПРОГРАМУВАННЯ, SCADA,
MATLAB, TIA PORTAL, СИСТЕМА РЕАЛЬНОГО ЧАСУ.

					141.0002.002.МД			
Змн.	Арк.	№ докум.	Підпис	Дата	Лабораторний стенд для дослідження систем автоматизації на основі контролерів Siemens Simatic S7 Реферат	Літ.	Арк.	Акрушів
Розроб.		Богданов В.А.						
Керівн.		Волянський Р. С.					7	114
Реценз.		Коваленко М. А.				КПІ ім. Ігоря Сікорського ФЕА, каф. АЕМС-ЕП гр. ЕП-31мп		
Н. Контр.								
Зав. каф		Ковбаса С. М.						

SUMMARY

The master's thesis consists of an explanatory note of 100 pages, 19 figures, 22 tables.

This master's thesis deals with the development of a laboratory stand for the study of automation systems based on Siemens Simatic S7 controllers. The features of the programmable logic controllers (PLCs) were analyzed and the requirements for the bench were formulated.

The choice of equipment is described, including controllers of the S7-300 and S7-1200 series, expansion modules, power supplies, and auxiliary components. An electrical schematic diagram of the test bench was developed.

Software was developed in the TIA Portal environment for PLC programming and KEPServerEX for creating a communication interface. A modeling program in MatLab/Simulink was created to test automation algorithms.

Methodological guidelines for training sessions, including tasks on PLC programming, network connections and integration of SCADA systems, were proposed. The test bench was tested in various operating modes.

A start-up project for the implementation of the testbed in educational institutions and industrial enterprises is proposed. Prospects for improving the test bench and its adaptation to modern automation needs are determined.

AUTOMATION, PLC, SIEMENS, PROGRAMMING, SCADA, MATLAB, TIA PORTAL, REAL-TIME SYSTEM.

					141.0002.002.MD						
	letter	Nº of document	Sign	Date							
Devel.		V. Bogdanov			Laboratory bench for studying automation systems based on Siemens Simatic S7 controllers Abstract			L.	Page	Pages	
Checked		R. Volianskyu								8	114
Recension		M. Kovalenko						NTUU "I. Sikorskiy KPI", FEA department AEMS-ED gr. EP-31mp			
N. contr.											
Approved		S. Kovbasa									

ЗМІСТ

ВСТУП	11
1 АНАЛІТИЧНИЙ ОГЛЯД	13
1.1 Визначення ПЛК	13
1.2 Опис ПЛК та порівняння з MCU	16
1.3 Мови програмування ПЛК	19
1.4 Входи та виходи ПЛК	20
1.5 Робота в системі реального часу	21
1.6 Принцип роботи	23
1.7 Огляд ПЛК різних розробників	28
Висновок до розділу	41
2 ФОРМУЛЮВАННЯ ВИМОГ ДО РОЗРОБЛЯЄМОГО ЛАБОРАТОРНОГО СТЕНДУ	42
Висновок до розділу	42
3 РОЗРОБКА СХЕМИ ТА ВИБІР ОБЛАДНАННЯ	43
Висновок до розділу	47
4 РОЗРОБКА ЗАВДАНЬ ТА МЕТОДОГІЧНИХ ВКАЗІВОК ДО ЛАБОРАТОРНОГО СТЕНДУ	48
4.1 Розробка загальних вимог до завдання	48
4.2 Розробка варіантів завдання	48
4.3 Створення програми для S7-300	69
4.4 Створення програми для KEPServerEX6	77
4.5 Створення програми для S7-1200	80
4.6 Розробка зв'язку між контролерами	80
4.7 Розробка ПІД регулятора	84
4.8 Створення простої SCADA для перевірки підключення	85
Висновки до розділу	85
5 РОЗРОБКА СИСТЕМИ ВЕКТОРНОГО КЕРУВАННЯ КУТОВОЮ ШВИДКІСТЮ АСИНХРОННОГО ДВИГУНА	87
5.1 Базові принципи проектування алгоритму векторного керування	87
5.2 Алгоритм непрямого векторного керування АД	89

5.3 Розрахунок параметрів номінального режиму роботи та схеми заміщення двигуна	91
5.4 Дослідження динамічних характеристик в номінальному режимі.....	96
Висновки до розділу	100
6 РОЗРОБКА СТАРТАП ПРОЕКТУ	101
6.1 Опис ідеї.....	101
6.2 Технологічний аудит проекту.....	102
6.3 Аналіз ринкових можливостей запуску стартап-проекту.....	103
6.4 Аналіз факторів конкурентоспроможності	105
6.5 Розроблення ринкової стратегії проекту	106
6.6 Розроблення маркетингової програми стартап-проекту.....	108
Висновки до розділу	109
ВИСНОВКИ.....	111
ПЕРЕЛІК ПОСИЛАНЬ.....	113

ВСТУП

Сучасні технології автоматизації є основою для підвищення ефективності та надійності виробничих процесів в різних галузях промисловості, таких як енергетика, машинобудування, хімічна промисловість та транспорт. Вони дозволяють автоматизувати складні технологічні ланцюги, знижуючи людський фактор, підвищуючи швидкість та точність виконання операцій, а також забезпечуючи високий рівень безпеки на підприємствах. Одним із важливих елементів цих технологій є програмовані логічні контролери (ПЛК), які стали основою для автоматизації процесів на багатьох підприємствах у світі.

Програмовані логічні контролери (ПЛК) дозволяють здійснювати автоматичне керування за допомогою програмного забезпечення, що надає можливість інтегрувати різні пристрої, датчики та виконавчі механізми в єдину автоматизовану систему. Сучасні ПЛК володіють широкими функціональними можливостями, включаючи підтримку різноманітних комунікаційних протоколів, здатність працювати в реальному часі, а також забезпечують високу гнучкість та масштабованість в процесі розробки та модернізації автоматизованих систем. Серед провідних виробників ПЛК є компанія Siemens, яка представила серії контролерів Simatic S7, що є одними з найбільш популярних на ринку.

Інтеграція ПЛК в автоматизовані системи не лише покращує ефективність виробництва, але й створює нові можливості для навчання та досліджень у галузі автоматизації. Створення лабораторних стендів для дослідження та тестування таких систем є важливим етапом у підготовці фахівців у цій сфері. Лабораторний стенд на основі контролерів Siemens Simatic S7 дозволяє студентам отримати практичні навички з програмування, налаштування і тестування ПЛК, а також здобути знання про принципи роботи різноманітних датчиків, виконавчих механізмів та автоматизованих систем в цілому.

У зв'язку з цим, метою цієї роботи є розробка універсального лабораторного стенду, який дозволить досліджувати основні принципи автоматизації за допомогою ПЛК Siemens Simatic S7. Стенд повинен забезпечити можливість реалізації різних технологічних процесів, тестування алгоритмів керування та моделювання реальних умов експлуатації ПЛК.

Розробка такого стенду є важливою не лише для навчання студентів, але й для проведення науково-дослідницьких робіт, спрямованих на вдосконалення та оптимізацію систем автоматизації. Система повинна включати необхідні програмні засоби для налаштування, моделювання та тестування алгоритмів керування, а також мати можливість інтеграції з іншими пристроями та системами. Важливою частиною роботи є також розробка методологічних вказівок та завдань для лабораторних робіт, що дозволить ефективно використовувати стенд у навчальному процесі.

Ця робота також дозволяє вирішити важливе завдання – забезпечити інтеграцію лабораторного стенду з сучасними SCADA-системами та іншими програмними засобами для автоматизації, що дозволить студентам ознайомитися з комплексними підходами до автоматизації та розумінням роботи сучасних промислових технологій.

1 АНАЛІТИЧНИЙ ОГЛЯД

Сьогодні важко уявити сучасне виробництво без програмованих логічних контролерів (ПЛК). Вони стали ключовим елементом автоматизованих систем, забезпечуючи надійну та автономну роботу з мінімальним людським втручанням, навіть у складних умовах експлуатації. ПЛК дозволяють автоматизувати процеси, які раніше вимагали постійної уваги та контролю з боку операторів.

Поява ПЛК у 1960-х роках була відповіддю на зростаючу потребу в автоматизації виробничих процесів. Перші ПЛК мали обмежені функції й використовувалися для простих логічних операцій. Проте розвиток технологій зробив їх потужнішими: сучасні контролери підтримують мережеву взаємодію, розширюються за допомогою спеціалізованого програмного забезпечення та мають широкі можливості для інтеграції.

Зараз ПЛК знаходять застосування в багатьох галузях промисловості, включаючи машинобудування, енергетику, нафтову та газову промисловість, транспорт і будівництво. Їх використовують для автоматизації виробничих ліній, роботизованих систем, систем керування енергопостачанням та автоматизованого транспорту.

Головні переваги ПЛК полягають у швидкій наладці, зниженні часу на обслуговування та ремонти, покращенні продуктивності і точності виконання завдань. Крім того, ПЛК підвищують рівень безпеки на виробництві, що допомагає зменшити ризик аварій та забезпечити стабільність роботи систем.

1.1 Визначення ПЛК

Програмований логічний контролер (ПЛК) – це спеціалізований мікропроцесорний пристрій, призначений для роботи у виробничих умовах, який програмується простими мовами, доступними користувачам без глибоких знань у програмуванні. За своєю суттю ПЛК є спрощеною моделлю комп'ютера, в якому цифровий керуючий автомат реалізується програмно.

ПЛК зазвичай має блочно-модульну конструкцію, що дозволяє користувачеві створювати необхідну конфігурацію, додаючи потрібні модулі вводу/виводу до базового модуля.

Основні особливості ПЛК включають простоту програмування за допомогою електричних схем, булевих рівнянь або алгоритмічних мов; здатність працювати в складних умовах завдяки оптоелектронній гальванічній розв'язці входів і виходів та адаптації до розширеного діапазону умов експлуатації. Модульність конструкції забезпечує створення контролерів різного функціонального призначення, що підвищує гнучкість систем управління складними модулями [1].

Використання ПЛК дозволяє суттєво скоротити витрати на проектування завдяки зменшенню вартості програмування та спрощенню адаптації конструкції під конкретні завдання. Процеси розробки і впровадження систем управління прискорюються за рахунок можливості паралельного виконання проектних і виробничих робіт. ПЛК також дозволяє змінювати алгоритми керування безпосередньо під час монтажу, пуску або модернізації, що підвищує адаптивність системи управління.

ПЛК підтримує широкий вибір модулів вводу/виводу для підключення до різних датчиків та виконавчих механізмів, що сприяє ефективному створенню комплексних систем. Вбудована функціональна діагностика полегшує експлуатацію і підвищує ремонтпридатність як контролера, так і обладнання, яким він керує.

Основні характеристики, які визначають вибір ПЛК для конкретного застосування, включають:

- Кількість входів-виходів, що є ключовим параметром, який визначає максимальну кількість датчиків і виконавчих механізмів, підключених до ПЛК.
- Номенклатура доступних модулів вводу/виводу, яка відображає здатність ПЛК різних виробників адаптуватися до умов промислової

експлуатації, враховуючи діапазони напруги живлення і допустимі струми для керуючих пристроїв, датчиків та виконавчих механізмів.

- Обсяг пам'яті для зберігання користувацьких програм, що визначає можливості ПЛК у створенні й виконанні прикладного програмного забезпечення.

- Швидкодія контролера, яка зазвичай визначається тривалістю циклу обслуговування всіх входів та виходів, що впливає на оперативність роботи.

- Типи підтримуваних мов і технологій програмування, які визначають рівень складності у створенні прикладного програмного забезпечення та зручність редагування керуючих програм, збережених у пам'яті.

- Наявність стандартних інтерфейсів, що забезпечує сумісність ПЛК з ієрархічними системами управління, підтримуючи дистанційне введення та коригування програм і даних, а також інтеграцію у контролерні мережі.

Контролери можуть бути реалізовані у різних формах: це можуть бути механічні пристрої, пневматичні чи гідравлічні автомати, релейні або електронні схеми, а також комп'ютерні програми. У ПЛК зазвичай використовуються три типи пам'яті: PROM для зберігання базових керуючих програм, ОЗП з живленням від акумуляторів для налагодження та зберігання оперативної інформації, і REEPROM як основна пам'ять для керуючих програм.

Якщо контролер інтегрується у машину масового виробництва, вартість його проектування розподіляється на велику кількість виробів, що робить її порівняно низькою. Проте для машин, що випускаються в одиничних екземплярах, вартість проектування суттєво перевищує витрати на виробництво. У промисловому виробництві часто необхідно працювати з невеликою кількістю однотипних пристроїв, і важливим є швидке переналаштування обладнання для виготовлення іншої продукції.

Релейні контролери чи мікросхеми з «жорсткою» логікою не можуть бути переналаштовані без суттєвих змін, тоді як програмовані логічні контролери забезпечують таку можливість. Концепція ПЛК з'явилася разом із

розвитком мікропроцесорів. Зазвичай ПЛК — це блок із фіксованим набором входів та виходів задля підключення ініціаторів та актуаторів.

ПЛК класифікуються за кількістю входів-виходів: мікро — до 64, малі — до 128, середні — до 512, великі — до 1024 і надвеликі — понад 1024. Логіка керування визначається програмно на базі мікрокомп'ютерного ядра, тому однакові апаратно ПЛК можуть реалізовувати різні задачі. Входи та виходи ПЛК сумісні з уніфікованими пристроями, але мають обмежені можливості для модифікації [2].

Основним завданням прикладного програмування ПЛК є реалізація алгоритму управління конкретним обладнанням. Контролер автоматично опитує входи та виходи, незалежно від того, як саме вони підключені. Цей процес забезпечується системним програмним забезпеченням. Ідеальний сценарій передбачає, що програмісту не потрібно знати, де знаходяться датчики та виконавчі механізми і як вони під'єднані. Програми розробляються незалежно від моделі або виробника ПЛК завдяки стандартизації мов програмування, що дозволяє використовувати один додаток на різних контролерах, які підтримують відповідний стандарт.

ПЛК є дискретним автоматом із програмним управлінням, який має набір входів, з'єднаних із датчиками, та виходів, що підключаються до виконавчих пристроїв. Контролер зчитує стан входів і виконує програмно визначені дії для керування виходами. ПЛК розроблений для роботи в режимі реального часу в умовах промислового середовища та є зручним для програмування навіть фахівцями без спеціальних знань у галузі інформатики.

1.2 Опис ПЛК та порівняння з MCU

Структура ПЛК, окрім центрального процесора (CPU), містить блок живлення, пристрої пам'яті (оперативна пам'ять – ОЗП для тимчасових даних, постійна пам'ять – ПЗП або Flash для зберігання програм і конфігурацій), модулі введення та виведення (I/O) для прийому сигналів від датчиків і передачі команд виконавчим механізмам, комунікаційні модулі для зв'язку з

іншими пристроями, а також інтерфейси для програмування та обслуговування. Крім цього, до складу ПЛК входять таймери, системний годинник для точного керування процесами, а також периферійні пристрої, що забезпечують інтеграцію та узгоджену роботу всіх компонентів.

Взаємодія між внутрішніми частинами ПЛК і зовнішніми пристроями здійснюється за допомогою мікропрограм, які зберігаються у внутрішній пам'яті та виконуються в циклічному режимі, забезпечуючи стабільне керування у реальному часі.

ПЛК підтримує різноманітні інтерфейси зв'язку, серед яких RS-232, RS-485, Modbus, Profibus, CAN, DeviceNet, AS-interface, CC-Link, а також промисловий Ethernet. Деякі моделі можуть додатково підтримувати сучасні протоколи зв'язку для інтеграції в системи IoT та SCADA. Це дозволяє ПЛК працювати в комплексних автоматизованих середовищах і забезпечувати надійну взаємодію з іншими елементами промислової інфраструктури.

ПЛК забезпечують роботу в режимі реального часу, що дозволяє оперативно реагувати на зміни в процесі. Вони розроблені для надійної експлуатації у важких умовах, стійкі до пилу, вологи, вібрацій і температурних коливань. Завдяки високій швидкості обробки даних, ПЛК підходять для задач зі складним і швидкодіяним управлінням.

Гнучкість і модульність ПЛК дозволяють легко адаптувати та розширювати систему під час розвитку виробничих процесів. Вони інтегруються з іншими системами автоматизації, такими як HMI, SCADA та ERP, що забезпечує комплексний контроль і моніторинг. У сучасній промисловій автоматизації ПЛК є незамінними завдяки своїй надійності, ефективності та здатності виконувати складні завдання, підвищуючи продуктивність і забезпечуючи якість і безпеку виробництвах [3].

Мікроконтролери зазвичай використовуються в невеликих системах та вбудованих застосунках. Вони є компактними та недорогими пристроями, які об'єднують ядро мікропроцесора, пам'ять та різні периферійні елементи на одному чипі. MCU ідеально підходять для застосувань, де потрібне керування

в реальному часі, низьке споживання енергії та компактний дизайн. Їх часто використовують для інтерфейсу з датчиками, збору даних та виконання простих алгоритмів керування. MCU широко застосовуються в побутовій електроніці, автомобільних системах, медичних пристроях та у пристроях Інтернету речей (IoT).

Програмовані логічні контролери (PLC), навпаки, є спеціалізованими промисловими комп'ютерними системами, призначеними для керування та автоматизації складного обладнання і процесів в промислових умовах. Вони розроблені для роботи в суворих середовищах та відомі своєю надійністю. PLC підтримують високу швидкість обробки, контроль у реальному часі та розширені можливості введення/виведення для управління складними виробничими завданнями. Їх активно використовують у таких галузях, як виробництво, нафтогазова промисловість, автомобілебудування та харчова промисловість.

Мікроконтролери зазвичай програмуються на низькорівневих мовах, таких як асемблер чи C, що дозволяє точно контролювати апаратні ресурси. PLC, натомість, програмуються здебільшого за допомогою графічної мови – сходової логіки (Ladder Logic), яка наочно відображає керуючі схеми у вигляді реле та контактів. Деякі моделі PLC підтримують також структурований текст та функціональні блок-схеми.

Мікроконтролери використовуються для малих та середніх проєктів, тоді як PLC підходять для великих промислових систем із підвищеною складністю. PLC забезпечують підтримку великої кількості пристроїв вводу/виводу, що дозволяє одночасно керувати численними датчиками та виконавчими механізмами. Крім того, вони пропонують функції резервування, стійкості до збоїв та підтримку складних протоколів зв'язку.

Мікроконтролери зазвичай є економічнішим рішенням порівняно з PLC, що робить їх привабливими для проєктів з обмеженим бюджетом. PLC, своєю чергою, дорожчі через їхні промислові характеристики, високу надійність та спеціалізоване обладнання і програмне забезпечення.

Хоча мікроконтролери та PLC мають спільну мету – керування та моніторинг процесів – вони орієнтовані на різні масштаби та середовища. MCU краще підходять для невеликих вбудованих систем із низьким енергоспоживанням, тоді як PLC розроблені для великих промислових застосувань із високими вимогами до надійності та складності. Вибір між MCU і PLC залежить від конкретних потреб, масштабу, умов експлуатації та бюджету.

1.3 Мови програмування ПЛК

Існує п'ять основних мов програмування ПЛК:

1. LD (Ladder Diagram) — сходова діаграма (мова релейних схем), що нагадує класичні електричні креслення, використовуючи символи реле, контактів та котушок для опису логіки.

2. FBD (Function Block Diagram) — діаграма функціональних блоків, що графічно відображає логіку за допомогою послідовності блоків, кожен з яких виконує певну функцію.

3. IL (Instruction List) — список інструкцій, подібний до мови асемблера, що дозволяє програмісту працювати безпосередньо з апаратними ресурсами контролера.

4. ST (Structured Text) — структурований текст, мова високого рівня, яка за синтаксисом нагадує Pascal і підходить для написання складних алгоритмів.

5. SFC (Sequential Function Chart) — послідовні функціональні діаграми, які описують процес у вигляді послідовності кроків та переходів, що дозволяє програмувати етапи виробничих операцій із чіткими умовами переходу між ними.

Ці мови дозволяють розробникам створювати логіку керування, що визначає, як ПЛК має взаємодіяти з обладнанням та процесами. Проте не всі моделі ПЛК підтримують усі мови програмування, тому важливо ознайомитися з можливостями конкретного контролера перед його використанням.

Програмування ПЛК ґрунтується на стандарті ІЕС 61131-3, який визначає основні вимоги та правила для мов програмування. Однак, незважаючи на дотримання цього стандарту, синтаксис мов може трохи відрізнятися у різних виробників, що дозволяє їм вносити невеликі зміни в реалізацію. Завдяки цьому перехід між ПЛК різних виробників відбувається швидко та без особливих труднощів.

Приклади мов програмування LD та FBD наведено на рис. 1.1. Приклади мов програмування IL та ST наведено на рис. 1.2.

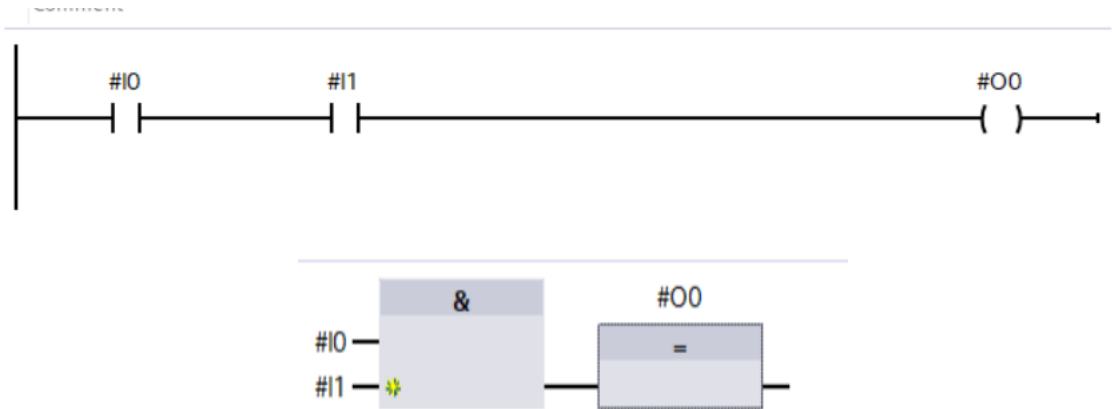


Рисунок 1.1 – Приклади мов програмування LD та FBD

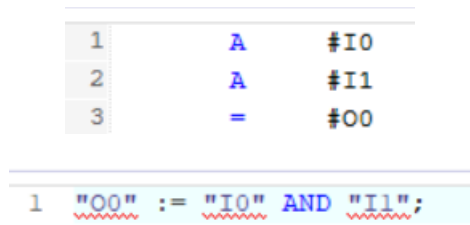


Рисунок 1.2 – Приклади мов програмування IL та ST

1.4 Входи та виходи ПЛК

Спочатку ПЛК були оснащені лише бінарними входами, які могли приймати лише два стани: логічний нуль та логічну одиницю. Логічна одиниця зазвичай означає наявність струму або напруги в ланцюзі, а їхня відсутність – логічний нуль. Пристроями, що генерують такі сигнали, можуть бути кнопки ручного управління, кінцеві вимикачі, датчики руху, контактні термометри та інші. Бінарні виходи також мають два стани – ввімкнений та вимкнений, і

використовуються для керування електромагнітними реле, силовими пускачами, клапанами або світловими індикаторами.

У сучасних ПЛК активно використовуються аналогові входи та виходи. Аналоговий сигнал передає рівень струму або напруги, що відповідає певній фізичній величині в конкретний момент часу. Такими величинами можуть бути температура, тиск, вага, швидкість або частота. Аналогові входи ПЛК мають різноманітні характеристики, серед яких: розрядність АЦП, діапазон вхідного сигналу, метод і час перетворення, рівень шуму, нелінійність, а також можливість автоматичного калібрування та програмного регулювання посилення. Окремі входи призначені для підключення спеціалізованих сенсорів, таких як термопари чи термометри опору, які вимагають особливої підтримки у вигляді лінеаризації, компенсації холодного спаю та інших схем.

Бінарні входи та виходи в контексті ПЛК часто називають дискретними. Крім того, ПЛК можуть мати спеціалізовані порти для підключення специфічних сенсорів і пристроїв, наприклад, квадратурних шифраторів, модулів управління двигунами чи інтерфейсів дисплеїв. Входи й виходи не завжди інтегруються безпосередньо в корпус ПЛК. Останніми роками набули популярності рішення, що дозволяють уникнути прокладання окремих кабелів для аналогових ланцюгів. Вводи та виводи реалізуються у вигляді компактних модулів, розташованих безпосередньо поруч із датчиками або виконавчими механізмами, а підключення до ПЛК здійснюється через єдиний цифровий кабель. Наприклад, в інтерфейсі Actuators/Sensors використовується «жовтий кабель» для передачі даних і живлення всього двома провідниками.

1.5 Робота в системі реального часу

У системах реального часу головною метою є не лише правильність рішення, а й швидкість його отримання. Навіть коректне рішення може бути неприйнятним, якщо воно затримується довше допустимого часу. Системи реального часу поділяють на жорсткі та м'які. Жорсткі системи мають чітко визначений час реакції: перевищення цього ліміту призводить до критичних

або незворотних наслідків. М'які системи допускають певні затримки, при яких погіршується лише ефективність, але не настають катастрофічні наслідки.

Більшість ПЛК використовують модель роботи, що базується на методі сканування, при якому контролер опитує входи, виконує програму й встановлює вихідні сигнали. Це дозволяє виконувати кілька завдань одночасно, хоча насправді вони обробляються послідовно в одному процесорі. Час реакції контролера залежить від кількості завдань у програмі, тому його розширення може збільшити затримку. Така архітектура підходить для м'яких систем реального часу, де час реакції вимірюється в мілісекундах і залишається достатнім для більшості виконавчих пристроїв.

Іноді в системах виникає необхідність фіксувати короткі імпульси від датчиків, що швидше за стандартний цикл ПЛК. У таких випадках використовуються спеціальні лічильні входи, які забезпечують швидку обробку сигналів. Крім того, інтелектуальні модулі, як-от для керування сервоприводами, виконують свої функції автономно, зменшуючи навантаження на основний процесор.

Промислові умови висувають високі вимоги до надійності ПЛК. Контролери повинні бути стійкими до температури, вологості, вібрацій, корозійних газів і електромагнітних перешкод. Це визначає вибір компонентів і конструкторські рішення. У процесі виробництва ПЛК проходять випробування на стійкість до впливу середовища, зокрема вібраційні та кліматичні тести. Конструкції контролерів можуть варіюватися від легких пластикових корпусів для монтажу в шафах до герметичних металевих блоків для роботи в важких умовах.

Окрім того, якісний ПЛК повинен мати вбудовані засоби захисту. До стандартних рішень належать гальванічна розв'язка входів і виходів, дзеркальні вихідні канали, захист від перенапруг та сторожові таймери. Такі заходи дозволяють забезпечити надійність роботи контролера навіть у жорстких виробничих умовах, зменшуючи ймовірність збоїв та простоїв.

1.6 Принцип роботи

Циклічна структура роботи є ключовою особливістю ПЛК. Кожен робочий цикл контролера складається з трьох етапів: зчитування стану входів, обробки інформації згідно з керуючою програмою та формування команд на виходи. Циклічна обробка, або сканування, виконується процесором, і кожне повне проходження всіх входів і виходів називається робочим циклом. Тривалість циклу сканування (Тц) визначає швидкодію ПЛК та повинна бути меншою, ніж час спрацьовування виконавчих механізмів (Тспр.вм), щоб уникнути помилкових сигналів через випадкові перешкоди.

Архітектура ПЛК базується на типовій структурі мікропроцесорного пристрою, який має модулі для зв'язку з об'єктами управління та пульти користувача для програмування, діагностики та налагодження. Пам'ять програм зберігається в ПЗП або ОЗП із підживленням, тоді як оперативні дані використовуються в ОЗП. Завдяки послідовним інтерфейсам ПЛК може інтегруватися з іншими мікропроцесорними системами, а його модульна структура забезпечує гнучкість у виборі модулів для обробки дискретних і аналогових сигналів.

Програма, що керує об'єктом, складається з логічних рівнянь, які виконуються послідовно, у порядку їх розміщення. Якщо використовується режим налаштування, програміст має доступ до значень входів і виходів, а також можливість керувати ними вручну для перевірки роботи датчиків. Рекомендується проводити багаторазове підтвердження стану входів перед подальшою обробкою, що допомагає уникнути помилкових рішень.

На виходах ПЛК формуються команди на запуск чи зупинку виконавчих механізмів, а також сигнали для таймерів і лічильників. Додатково програма може містити умови для безумовних чи умовних переходів між різними етапами роботи. Програмування ПЛК повинно бути доступним для технічного персоналу без залучення спеціалізованих фахівців. Використання складних мов знижує привабливість ПЛК на ринку, тому застосовуються прості та зрозумілі інструменти, наближені до виробничих технологій.

Управління в ПЛК здійснюється шляхом безперервного циклічного контролю, де алгоритми виконуються регулярно з короткими інтервалами часу. Основою роботи є робочий цикл, який складається з кількох етапів: зчитування входів, виконання програми, оновлення виходів і діагностика апаратних ресурсів. Цей процес повторюється постійно, поки ПЛК знаходиться в робочому режимі. Весь цикл, від початку до кінця, називається часом сканування. Після ввімкнення живлення ПЛК проводить самотестування, налаштовує апаратні ресурси й переходить до виконання циклічної програми.

Робочий цикл починається із зчитування вхідних сигналів, які зберігаються в оперативній пам'яті як миттєва копія. Програма працює саме з цією копією, тому вхідні дані не змінюються до наступного циклу, що запобігає неоднозначній обробці в різних гілках програми. Такий підхід дозволяє підвищити швидкість обробки, оскільки читання з пам'яті відбувається швидше, ніж безпосередньо з датчиків або зовнішніх пристроїв. Деякі компоненти, як-от АЦП, можуть працювати з затримкою, тож частина обробки виконується через переривання.

Після виконання призначеної для користувача програми ПЛК оновлює стан виходів, забезпечуючи виконання дій, передбачених програмою. У цій фазі ПЛК приводить фізичні виходи у відповідність із розрахунковими значеннями. Одночасно виконується обслуговування апаратних ресурсів, таких як системні таймери, годинники реального часу та засоби діагностики. Монітор системи виконання дозволяє завантажувати нові програми, переглядати та редагувати змінні, виконувати програми покроково й контролювати час циклу. Це спрощує налаштування та інтеграцію ПЛК у мережі й системи управління.

Час сканування, який визначає швидкодію ПЛК, залежить переважно від тривалості виконання програми користувача. Інші етапи робочого циклу зазвичай мають фіксовану тривалість. Для уникнення варіацій часу циклу в сучасних ПЛК застосовується контроль часу сканування: якщо окремі гілки

виконуються занадто швидко, система додає штучні затримки. У разі відсутності такої функції доводиться використовувати таймери для вирівнювання часу виконання. Це особливо важливо в задачах автоматичного регулювання, де варіації в часі циклу можуть вплинути на точність управління.

Сучасні ПЛК забезпечують високу гнучкість завдяки можливості роботи з різними модулями та режимами. У режимі налаштування користувач може вручну керувати виходами та перевіряти роботу датчиків. Це підвищує ефективність обслуговування та спрощує процес інтеграції.

Процес управління промисловим устаткуванням за допомогою ПЛК ґрунтується на розв'язанні логічних рівнянь, які визначають взаємодію між станами вхідних сигналів, внутрішніми функціями та вихідними командами. Важливо чітко визначати, які функції ПЛК вважаються вихідними, та які сигнали або стани об'єкта й елементів системи можуть виступати аргументами цих функцій. Основними вихідними функціями є ввімкнення та вимкнення виконавчих механізмів, а також запуск та скидання таймерів і лічильників. Вони також забезпечують управління внутрішніми операторами, які зберігають імпульсні сигнали, проміжні стани програми та результати, отримані під час виконання алгоритмів.

Особливість ПЛК полягає в тому, що один і той самий вхідний сигнал може використовуватися в різних частинах програми без обмежень. Це значно відрізняє ПЛК від традиційних релейних схем, де для повторного використання сигналу необхідні додаткові апаратні компоненти. Аналогічно, вихідні функції можуть бути розмножені без використання фізичних блок-контактів або аналогів. Таймери та лічильники можуть багаторазово застосовуватися в межах одного циклу, забезпечуючи гнучке управління. Крім того, програмно можна налаштувати збереження стану виходів, щоб вони залишалися активними навіть після зникнення умов для їх активації, або ж працювати без фіксації стану.

Таймери в ПЛК можуть бути реалізовані як апаратно, так і програмно. Програмні таймери забезпечують більшу гнучкість, оскільки не потребують

жорсткої прив'язки до конкретного виходу та дозволяють задавати різні часові функції, такі як затримка на ввімкнення чи вимкнення. Програмно можна також налаштувати умови для скидання таймера, що дає змогу обнулити його в разі, якщо умови активації зникають до завершення встановленого часу. У різних моделях ПЛК таймери можуть працювати в режимі зворотного або прямого відліку часу, що визначає необхідність додаткових команд для примусового скидання.

Лічильники в ПЛК також мають широкий спектр функцій. Вони можуть виконувати як підсумовування, так і віднімання, а також працювати у реверсивному режимі. Лічильники використовуються не лише для обліку зовнішніх подій, наприклад, кількості виготовленої продукції, але й для підрахунку внутрішніх процесів, таких як кількість виконаних циклів програми. Для точної роботи лічильників ПЛК повинні забезпечувати коректну фіксацію імпульсів, що надходять із зовнішніх датчиків. Використання спеціальних схем фільтрації та виділення одиничних імпульсів дозволяє уникнути "брязкоту" контактів і зменшити вплив перешкод, що підвищує точність підрахунку.

Ще одна важлива функція ПЛК – зони ігнорування в програмі. Цей механізм дозволяє пропускати певні ділянки коду під час виконання робочого циклу, що зменшує загальний час обробки й підвищує швидкодію системи. Завдяки цьому процесор зосереджується лише на активних частинах програми, що сприяє оптимізації роботи й підвищенню надійності. Використання зон ігнорування також дозволяє зменшити кількість умовних переходів і спростити структуру програми, що є важливим для стабільної роботи ПЛК у промислових умовах.

Автоматична діагностика несправностей є ще однією важливою перевагою ПЛК. Контролер моніторить стан каналів вводу-виводу, датчиків, виконавчих механізмів і ліній зв'язку, виявляючи можливі збої. У більшості випадків діагностика та аварійне відключення реалізуються програмно, без потреби у додатковій апаратурі. Важливо зазначити, що пам'ять для

діагностичних програм та алгоритмів аварійної зупинки часто займає більший обсяг, ніж основна керуюча програма. Це підкреслює важливість надійного моніторингу та захисту системи в умовах промислової експлуатації.

ПЛК підтримують використання енергозалежної та енергонезалежної пам'яті. Енергозалежна пам'ять може виконувати роль "нульового захисту", запобігаючи відновленню роботи механізмів після відключення живлення без перевірки всіх блокувань і умов. Енергонезалежна пам'ять забезпечує збереження даних і станів після відновлення живлення, що важливо для безперервної роботи. Однак саме енергозалежна пам'ять запобігає повторному увімкненню механізмів без дотримання потрібної послідовності підключення та без виконання всіх необхідних умов безпеки.

Програмні ПЛК (soft PLC) також набувають популярності. Вони дозволяють виконувати функції ПЛК на комп'ютерах із встановленими платами вводу-виводу, що дає змогу об'єднати в одному середовищі програмування, управління та моніторинг. Проте такі системи мають свої недоліки. Одним із головних є тривалий час завантаження після ввімкнення живлення або перезапуску. Це може створити ризики, якщо система залишається некерованою впродовж декількох хвилин, особливо у випадках, коли перезапуск ініційований автоматично за допомогою сторожового таймера, а стан виконавчих механізмів не відповідає вихідним позиціям. Тому класичні апаратні ПЛК залишаються кращими для критичних систем, де відновлення роботи має відбуватися за лічені мілісекунди.

Таким чином, ПЛК є надзвичайно гнучким та ефективним рішенням для управління промисловими процесами. Їхні можливості багаторазового використання вхідних сигналів, таймерів та лічильників, а також здатність виконувати діагностику без додаткової апаратури роблять їх надійним інструментом у виробничих умовах. Водночас використання зон ігнорування та програмних інструментів оптимізує роботу та підвищує швидкодію системи, забезпечуючи стабільне й безпечне управління технологічними процесами [4].

1.7 Огляд ПЛК різних розробників

Одним із лідерів на ринку промислових ПЛК є Siemens, що пропонує контролери серії SIMATIC, які охоплюють широкий спектр застосувань – від невеликих установок до великих виробничих систем. Контролери цієї серії підтримують такі комунікаційні протоколи, як PROFINET, PROFIBUS та OPC UA, що забезпечує швидкий і надійний обмін даними між різними компонентами системи автоматизації. ПЛК Siemens відомі високою продуктивністю та гнучкістю, що дозволяє користувачам адаптувати рішення під специфічні потреби виробництва. Контролери SIMATIC активно інтегруються з хмарними платформами, такими як MindSphere, що дає можливість здійснювати віддалений моніторинг та управління обладнанням у реальному часі. Крім того, Siemens приділяє значну увагу питанням кібербезпеки – контролери оснащені засобами захисту від несанкціонованого доступу, що є важливим для критичних інфраструктур.

Siemens пропонує широкий спектр програмованих логічних контролерів (ПЛК) під брендом SIMATIC, які охоплюють різні серії для вирішення завдань автоматизації в промисловості, енергетиці, транспорті та будівництві. Кожна серія має свої особливості, функціональність та масштаби застосування, що дозволяє користувачам обирати оптимальне рішення для конкретних потреб.

Серії ПЛК Siemens

SIMATIC S7-200 SMART. Це компактні контролери, які розроблені для малих та середніх об'єктів автоматизації. Вони забезпечують високу продуктивність у компактному форматі й підтримують базові функції управління. Контролери серії S7-200 SMART підходять для автоматизації невеликих виробничих ліній, систем HVAC, насосів та конвеєрних систем. Вони також підтримують Modbus RTU для інтеграції в локальні мережі.

SIMATIC S7-300. Ця серія є однією з найпопулярніших на ринку. Вона забезпечує гнучкість і масштабованість завдяки модульній конструкції. Контролери S7-300 використовуються для середніх і великих промислових установок, таких як виробництво, енергетичні об'єкти та хімічні підприємства.

S7-300 підтримує стандартні протоколи PROFIBUS і PROFINET, що дозволяє легко інтегрувати його в комплексні системи управління.

SIMATIC S7-400. Це високопродуктивна серія для великих та складних систем автоматизації. Контролери S7-400 зазвичай використовуються у процесах, де потрібна висока надійність та безперервна робота, як-от у нафтовій, хімічній промисловості та на великих енергетичних об'єктах. Завдяки підтримці гарячого резервування (hot standby), вони можуть забезпечувати безперебійне функціонування навіть при виході з ладу одного з компонентів.

SIMATIC S7-1200. Серія S7-1200 розроблена для вирішення завдань автоматизації середньої складності. Вона поєднує високу продуктивність з компактністю та є ідеальною для керування виробничими лініями, системами освітлення, вентиляції та HVAC. ПЛК S7-1200 підтримують інтеграцію з хмарними платформами, такими як MindSphere, що дозволяє здійснювати віддалений моніторинг і управління. Контролери цієї серії також підтримують Ethernet/IP та інші сучасні протоколи для підключення до мереж.

SIMATIC S7-1500. Це флагманська серія Siemens, що відрізняється високою продуктивністю, гнучкістю та надійністю. Контролери S7-1500 розроблені для використання в критичних виробничих процесах, які потребують високої швидкодії та точності. Вони підтримують широкий спектр інтерфейсів, включаючи PROFINET, а також функції кібербезпеки для захисту від несанкціонованого доступу. S7-1500 часто застосовуються у поєднанні з технологіями ІоТ для збору та аналізу даних у реальному часі.

LOGO! Це найпростіший контролер у лінійці Siemens, який розроблений для невеликих проектів автоматизації, таких як домашні або офісні системи керування освітленням, вентиляцією чи водопостачанням. LOGO! вирізняється легкістю налаштування та можливістю програмування без глибоких знань у цій галузі, що робить його популярним вибором для малого бізнесу та приватних користувачів.

Для розробки, тестування та впровадження програм для контролерів Siemens використовуються спеціалізовані середовища програмування. Основними інструментами є:

TIA Portal (Totally Integrated Automation Portal)
 TIA Portal – це інтегроване середовище програмування для всієї лінійки контролерів SIMATIC. Воно дозволяє розробляти, тестувати й обслуговувати програми для ПЛК, панелей оператора та інших автоматизованих пристроїв в єдиному інтерфейсі. Середовище підтримує стандартні мови програмування IEC 61131-3 (Ladder Diagram, Function Block Diagram, Structured Text тощо), що спрощує розробку навіть для складних проектів. TIA Portal також дозволяє інтегрувати пристрої в мережі PROFINET та OPC UA, забезпечуючи високу гнучкість при створенні промислових мереж.

STEP 7 – це програмний пакет, призначений для програмування та налагодження ПЛК серій S7-300, S7-400, S7-1200 та S7-1500. Він дозволяє розробляти програми на мовах IEC 61131-3, здійснювати їх налагодження та завантаження на контролери. Крім того, STEP 7 забезпечує моніторинг і діагностику роботи контролера в реальному часі, що підвищує ефективність обслуговування.

LOGO! Soft Comfort. Це програмне забезпечення для контролерів серії LOGO!, що забезпечує просте та зручне програмування завдяки інтуїтивному графічному інтерфейсу. З його допомогою користувачі можуть швидко створювати й налагоджувати програми для автоматизації невеликих систем, таких як керування освітленням або кліматичними установками.

WinCC – це програмний пакет для створення систем SCADA та HMI. Він дозволяє організувати моніторинг та управління виробничими процесами в реальному часі, відображаючи стан обладнання та дозволяючи операторам здійснювати контроль над системою. WinCC інтегрується з TIA Portal та може працювати як із ПЛК Siemens, так і з обладнанням інших виробників.

MindSphere. Це хмарна платформа IoT від Siemens, яка дозволяє збирати та аналізувати дані з різних пристроїв, включаючи ПЛК серій S7-1200 та S7-

1500. MindSphere забезпечує віддалений доступ до даних і підтримку прийняття рішень на основі аналітики в реальному часі. Використання цієї платформи дає змогу підприємствам підвищувати ефективність та виявляти потенційні проблеми до їх виникнення [5].

Schneider Electric зосереджується на розвитку лінійки Modicon, яка вже кілька десятиліть залишається однією з найбільш універсальних на ринку. ПЛК цієї серії широко застосовуються в енергетиці, водопостачанні, харчовій промисловості та інших галузях. Вони підтримують Modbus, Ethernet/IP та CANopen, що дозволяє інтегрувати їх у багаторівневі системи управління. Контролери Modicon вирізняються високою стійкістю до впливу навколишнього середовища, що робить їх надійним рішенням для роботи в агресивних умовах, таких як висока температура, вологість або наявність пилу. Особлива увага приділяється функціям енергоменеджменту, що допомагає підприємствам знизити витрати на електроенергію. Schneider також робить акцент на підтримці стандартних мов програмування за IEC 61131-3, що спрощує налаштування та впровадження систем навіть для технічного персоналу без глибоких знань у програмуванні.

Modicon M221. Це компактні ПЛК, розроблені для невеликих систем автоматизації, таких як HVAC, конвеєри або водяні насоси. Вони підтримують Modbus RTU та Ethernet/IP, що полегшує інтеграцію у локальні мережі. M221 забезпечує високу швидкодію для своїх розмірів і має зручний інтерфейс для налаштування та введення в експлуатацію.

Modicon M241. Контролери цієї серії забезпечують більше обчислювальних можливостей і розширений набір функцій у порівнянні з M221. M241 підтримує інтерфейси Modbus TCP/IP та CANopen, що робить його ідеальним для інтеграції з іншими пристроями та системами. ПЛК цієї серії використовуються в середніх промислових системах для управління лініями виробництва або складськими системами.

Modicon M251. Цей ПЛК орієнтований на управління розподіленими системами з великою кількістю підключених пристроїв. M251 підтримує

Ethernet/IP та Modbus TCP, що робить його ідеальним вибором для інтеграції в багаторівневі мережі автоматизації. Він також забезпечує можливість дистанційного моніторингу та управління через хмарні платформи, що полегшує керування географічно розподіленими об'єктами.

Modicon M262. Це сучасний контролер, розроблений із урахуванням концепцій IoT (Індустріального Інтернету Речей). M262 підтримує комунікаційні протоколи OPC UA та MQTT, що дозволяє інтегрувати його в хмарні сервіси та системи віддаленого моніторингу. Контролери цієї серії підходять для високотехнологічних виробничих ліній, які вимагають інтелектуального керування в реальному часі.

Modicon M580. Це флагманська серія Schneider Electric, що забезпечує високу продуктивність та надійність для критичних систем автоматизації. ПЛК M580 підтримують функції гарячого резервування, що дозволяє їм працювати без зупинки навіть у разі відмови одного з компонентів. Ця серія активно використовується у великих інфраструктурних проектах, таких як нафтопереробні заводи та енергетичні комплекси.

Для програмування та управління ПЛК Schneider Electric використовуються кілька спеціалізованих програмних рішень, що полегшують розробку та обслуговування автоматизованих систем.

EcoStruxure Machine Expert. Це універсальне середовище для програмування та конфігурування контролерів Modicon серій M221, M241, M251 та M262. Воно підтримує мови програмування, визначені стандартом IEC 61131-3, включаючи Ladder Diagram (LD), Structured Text (ST) та Function Block Diagram (FBD). Програма також дозволяє проводити симуляцію та налагодження програм безпосередньо з робочого місця, що значно прискорює процес запуску в експлуатацію.

EcoStruxure Control Expert. Це програмне забезпечення для більш складних систем, таких як контролери серії Modicon M580. Воно забезпечує можливість створення масштабованих та розподілених систем автоматизації. Control Expert також підтримує гаряче резервування та дистанційне керування,

що робить його незамінним для критичних інфраструктурних проєктів. Пакет містить засоби для моніторингу, діагностики та візуалізації процесів у реальному часі.

SoMachine Basic. Це простіша версія EcoStruxure Machine Expert, призначена для швидкого налаштування та програмування невеликих ПЛК, таких як Modicon M221. SoMachine Basic забезпечує інтуїтивний інтерфейс для програмування й підтримує мови IEC 61131-3, що полегшує роботу навіть для операторів без глибоких знань у програмуванні.

EcoStruxure Operator Terminal Expert. Це програмне середовище для створення HMI (інтерфейсів людина-машина) та панелей оператора, які інтегруються з ПЛК Schneider Electric. Воно дозволяє візуалізувати виробничі процеси та надає оператору можливість моніторити й керувати системою в реальному часі.

EcoStruxure Augmented Operator Advisor. Це сучасне рішення, яке використовує технології доповненої реальності (AR) для покращення роботи з обладнанням. Оператори можуть бачити реальний стан обладнання через мобільні пристрої або планшети, що полегшує діагностику та усунення несправностей [6].

Allen-Bradley, бренд компанії Rockwell Automation, орієнтований на розробку високопродуктивних ПЛК для складних виробничих процесів. Серії ControlLogix та CompactLogix забезпечують можливість побудови ієрархічних систем автоматизації, де кілька рівнів контролю взаємодіють через стандартизовані протоколи. Ці контролери підтримують не тільки Ethernet/IP, але й DeviceNet та ControlNet, що дає змогу інтегрувати їх з різноманітними пристроями та системами. ПЛК Allen-Bradley активно впроваджуються в концепцію IIoT (Industrial Internet of Things), дозволяючи підприємствам використовувати аналітику даних для покращення ефективності виробничих процесів. Однією з особливостей цих контролерів є гнучка модульна структура, яка дозволяє розширювати систему відповідно до змін у виробництві без суттєвих витрат.

ControlLogix. Це високопродуктивна серія ПЛК, розроблена для складних промислових процесів і великих виробничих об'єктів. ControlLogix підтримує модульну архітектуру, що дозволяє легко додавати нові функціональні блоки в систему. Контролери цієї серії забезпечують інтеграцію з протоколами Ethernet/IP та DeviceNet, що сприяє створенню ієрархічних систем управління. ControlLogix також підтримує гаряче резервування та інші механізми забезпечення безперервної роботи, що робить їх ідеальним рішенням для критичних інфраструктур.

CompactLogix. Ця серія розроблена для середніх систем автоматизації, де потрібна висока швидкодія та можливість розширення. CompactLogix поєднує в собі гнучкість і потужність ControlLogix із компактною конструкцією, що робить його оптимальним вибором для ліній автоматизації у виробництві, логістичних центрах та енергетиці. Він підтримує Ethernet/IP, що дозволяє легко інтегрувати його з іншими системами автоматизації та мережами IoT.

MicroLogix. Це компактні контролери, розроблені для невеликих автономних систем автоматизації. Вони підтримують базові функції управління та мають інтерфейси для роботи з Modbus та Ethernet/IP. MicroLogix зазвичай використовуються для керування невеликими машинами, насосами або вентиляційними системами, де не потрібна висока продуктивність, але важлива надійність і простота використання.

SLC 500. Хоча ця серія контролерів поступово виходить з ужитку, вона все ще популярна на багатьох виробництвах, які використовують спадкові системи автоматизації. Контролери SLC 500 підтримують RS-232 та DH-485 для зв'язку з іншими пристроями. Вони забезпечують базові функції автоматизації та можуть працювати в системах, що не вимагають складного програмування або високої швидкодії.

Micro800. Це нові компактні контролери для малих і середніх проектів автоматизації, які підтримують мову програмування по стандарту IEC 61131-3. Micro800 відомі своєю низькою вартістю та можливістю легкої інтеграції в

локальні системи автоматизації. Ці контролери часто використовуються у невеликих виробництвах, будівлях та системах керування кліматом.

Для налаштування та управління ПЛК Allen-Bradley компанія Rockwell Automation пропонує кілька програмних рішень, які охоплюють всі етапи роботи – від розробки програмного коду до моніторингу та діагностики систем.

Studio 5000 Logix Designer. Це головне середовище для розробки програм для серій ControlLogix та CompactLogix. Studio 5000 підтримує мови програмування за стандартом IEC 61131-3 (Ladder Diagram, Structured Text, Function Block Diagram тощо). Воно дозволяє здійснювати симуляцію та налагодження програм у реальному часі, що скорочує час введення системи в експлуатацію. Studio 5000 також інтегрується з мережами Ethernet/IP, забезпечуючи зручну роботу з пристроями в мережі.

RSLogix 5000 / RSLogix 500. Це попередні версії програмного забезпечення для програмування ПЛК ControlLogix, CompactLogix та SLC 500. Вони забезпечують повну сумісність зі старими системами автоматизації, що дозволяє підприємствам підтримувати спадкові системи без необхідності заміни обладнання.

Connected Components Workbench (CCW). Це програмне середовище для роботи з контролерами серій Micro800. Воно забезпечує простий інтерфейс для програмування, моніторингу та діагностики невеликих систем автоматизації. CCW підтримує стандарт IEC 61131-3 та дозволяє легко створювати проекти навіть для початківців у сфері програмування.

FactoryTalk View. Це програмне забезпечення для створення НМІ (інтерфейсів людина-машина) та систем моніторингу. FactoryTalk View дозволяє операторам керувати виробничими процесами в реальному часі та відстежувати всі важливі параметри системи. Програма підтримує інтеграцію з контролерами Allen-Bradley через Ethernet/IP та може працювати в комплексі з іншими рішеннями FactoryTalk.

FactoryTalk Analytics. Це програмне рішення для збору та аналізу даних із промислових об'єктів. Воно дозволяє підприємствам використовувати можливості ІоТ для оптимізації виробничих процесів і підвищення ефективності роботи. FactoryTalk Analytics інтегрується з контролерами Allen-Bradley, забезпечуючи аналітику в реальному часі та інструменти для прийняття оперативних рішень.

FactoryTalk AssetCentre. Це рішення для управління та моніторингу активів. AssetCentre забезпечує відстеження змін у системах автоматизації та спрощує керування версіями програмного забезпечення. Завдяки цьому підприємства можуть зменшити час простою та підвищити ефективність обслуговування обладнання [7].

Mitsubishi Electric відома своєю серією MELSEC, яка застосовується в різних галузях, включаючи автомобільну, електронну та хімічну промисловості. Контролери MELSEC вирізняються компактністю та високою швидкістю, що робить їх ідеальними для автоматизації технологічних процесів, де потрібна швидка реакція системи. Ці ПЛК підтримують комунікаційні протоколи CC-Link, Ethernet/IP та PROFINET, що дозволяє легко інтегрувати їх у складні виробничі мережі. Mitsubishi також приділяє увагу розвитку інтелектуальних функцій, таких як управління сервоприводами та роботизованими механізмами, що робить ці контролери важливими компонентами для створення високотехнологічних виробничих ліній.

MELSEC FX. Це компактні контролери, які використовуються для невеликих та середніх систем автоматизації. Серія FX підтримує базові протоколи, такі як Modbus та Ethernet/IP, що дозволяє інтегрувати їх у локальні мережі. Завдяки своїй простоті та універсальності контролери MELSEC FX часто застосовуються для керування конвеєрними системами, насосами, вентиляційним обладнанням та іншими технологічними процесами.

MELSEC L. Ці контролери призначені для середніх систем управління, де потрібна висока швидкість обробки та гнучкість конфігурації. Контролери серії L підтримують такі інтерфейси, як Ethernet/IP, CC-Link та RS-232, що

дозволяє легко інтегрувати їх з іншими пристроями в промисловій мережі. Вони також підтримують функції управління рухом, що робить їх ефективними для автоматизації пакувальних ліній та складальних процесів.

MELSEC Q. Це високопродуктивні контролери, розроблені для великих промислових установок з високими вимогами до надійності та швидкодії. Серія Q підтримує модульну архітектуру, що дозволяє додавати нові модулі вводу/виводу та розширювати функціональні можливості системи. Вони також підтримують такі протоколи, як PROFINET, CC-Link IE та EtherCAT, що сприяє створенню складних мереж автоматизації. MELSEC Q активно використовується на виробництвах з високими вимогами до точності й безперебійної роботи, таких як автомобільна та електронна промисловість.

MELSEC iQ-F. Це сучасна серія контролерів, яка поєднує високу продуктивність з компактністю. Контролери iQ-F підтримують хмарні рішення та інтеграцію з системами IoT, що дозволяє здійснювати віддалений моніторинг та управління. Вони також забезпечують швидку обробку даних та підтримку протоколів Ethernet, що робить їх ідеальними для автоматизації невеликих, але критичних процесів.

MELSEC iQ-R. Це флагманська серія Mitsubishi, розроблена для інноваційних та складних виробничих систем. Контролери iQ-R забезпечують найвищу продуктивність і підтримують гаряче резервування для безперервної роботи в разі відмови компонентів. Вони інтегруються з мережами CC-Link IE та OPC UA і активно використовуються в проектах з високим ступенем автоматизації та кібербезпеки.

Mitsubishi Electric пропонує потужні програмні рішення для розробки, тестування та обслуговування своїх ПЛК. Ці інструменти підтримують стандарти IEC 61131-3 та забезпечують високу ефективність на всіх етапах автоматизації.

GX Works2 та GX Works3. Це основні середовища програмування для серій MELSEC FX, L, Q та iQ-R. Вони підтримують різні мови програмування, включаючи Ladder Logic (LD), Function Block Diagram (FBD) та Structured Text

(ST). GX Works2 використовується для старіших контролерів, тоді як GX Works3 призначений для нових серій, таких як iQ-F та iQ-R. Обидві програми надають зручні інструменти для симуляції та діагностики програм у реальному часі.

GT Designer3. Це програмне середовище для розробки інтерфейсів HMI (людина-машина) для панелей оператора, які інтегруються з ПЛК Mitsubishi. GT Designer3 дозволяє створювати інтуїтивно зрозумілі інтерфейси для моніторингу та управління виробничими процесами. Програма підтримує функції віддаленого доступу, що дозволяє операторам керувати обладнанням з будь-якого місця.

iQ Works. Це комплексний пакет програмного забезпечення для інтеграції всіх компонентів системи автоматизації. iQ Works об'єднує інструменти для програмування ПЛК, HMI та управління мережами в єдиному середовищі. Пакет дозволяє скоротити час на розробку проектів та спрощує обслуговування систем.

MELSOFT Navigator. Це інструмент для створення та управління архітектурою автоматизованої системи. MELSOFT Navigator надає можливість графічно проектувати виробничі лінії та налаштовувати зв'язки між компонентами, що спрощує конфігурування та діагностику системи.

MAPS (Mitsubishi Adroit Process Suite). Це програмний пакет для створення SCADA-систем, який забезпечує моніторинг та управління виробничими процесами в реальному часі. MAPS інтегрується з ПЛК Mitsubishi, надаючи користувачам аналітику даних та засоби для прийняття рішень.

MES Interface IT. Це рішення для інтеграції ПЛК з системами MES (Manufacturing Execution System). Воно дозволяє обмінюватися даними між ПЛК та інформаційними системами підприємства для підвищення ефективності виробничих процесів та автоматизації звітності [8].

Delta Electronics, ще один важливий гравець на ринку, спеціалізується на розробці рішень для енергоефективної автоматизації. Їхні ПЛК серії AS та

DVP відомі простотою інтеграції та оптимальним співвідношенням ціни та якості, що робить їх популярними серед підприємств малого та середнього бізнесу. ПЛК Delta підтримують протоколи Modbus, CANopen та Ethernet/IP, що сприяє зручній інтеграції в різні системи управління. Ці контролери часто застосовуються для керування насосами, вентиляційними системами, обладнанням для водопостачання та іншими технологічними процесами. Особливою перевагою Delta є підтримка енергоменеджменту, що допомагає підприємствам оптимізувати енергоспоживання та знижувати витрати. Компанія також пропонує програмне забезпечення для віддаленого моніторингу та діагностики, що підвищує надійність і зручність обслуговування.

DVP Series. Серія DVP – це компактні та високошвидкісні ПЛК, призначені для малих і середніх систем автоматизації. Вони підтримують різні протоколи, включаючи Modbus, CANopen та Ethernet/IP, що забезпечує зручну інтеграцію з іншими пристроями та системами. ПЛК серії DVP часто використовуються для керування конвеєрними лініями, вентиляційними системами та насосами. Завдяки своїй швидкодії та гнучкості ці контролери ідеально підходять для автоматизації HVAC та інших інфраструктурних об'єктів.

AS Series. AS – це більш продуктивна серія ПЛК, розроблена для складних виробничих систем. Вона підтримує такі протоколи, як Ethernet/IP та Modbus TCP/IP, що дозволяє використовувати її у складних мережах автоматизації. Контролери AS забезпечують високу швидкість обробки даних та підтримують управління рухом, що робить їх ефективним рішенням для виробничих ліній із сервоприводами та іншими високоточними механізмами.

AN Series. Це модульні ПЛК, призначені для великомасштабних промислових проектів, таких як водопостачання, управління енергетичними системами та важка промисловість. Серія AN підтримує гаряче резервування, що гарантує безперервну роботу системи навіть при відмові одного з

компонентів. Контролери цієї серії інтегруються з хмарними платформами для віддаленого моніторингу та управління.

MX Series. Контролери MX забезпечують управління сервоприводами та високоточними механізмами, що робить їх оптимальним вибором для автоматизації виробництва електроніки та точних механічних процесів. Вони підтримують швидкі інтерфейси для інтеграції з мережами EtherCAT та CC-Link, що забезпечує високу швидкість обміну даними.

Delta Electronics пропонує потужне програмне забезпечення, яке дозволяє здійснювати налаштування, моніторинг та управління ПЛК. Це забезпечує ефективну роботу систем автоматизації на всіх етапах – від розробки до введення в експлуатацію та обслуговування.

ISPSoft. Це головне середовище для програмування контролерів серій DVP та AS. ISPSoft підтримує мови програмування за стандартом IEC 61131-3 (Ladder Diagram, Function Block Diagram, Structured Text). Програмне забезпечення також надає інструменти для симуляції програм і налагодження в реальному часі, що скорочує час введення систем у експлуатацію.

DIASView SCADA. Це програмне рішення для створення систем моніторингу та управління (SCADA). DIASView дозволяє візуалізувати та контролювати роботу виробничих ліній, а також керувати системами автоматизації в реальному часі. Завдяки інтеграції з контролерами Delta забезпечується повний контроль над обладнанням та швидкий доступ до даних.

DIACloud. Це хмарна платформа для віддаленого моніторингу та управління обладнанням. Платформа DIACloud дозволяє збирати та аналізувати дані з ПЛК у реальному часі, що допомагає підприємствам виявляти проблеми та оптимізувати роботу систем. Це рішення особливо корисне для управління географічно розподіленими об'єктами, такими як системи водопостачання чи транспортні мережі.

DOPSoft. Це середовище для розробки інтерфейсів HMI, що інтегруються з контролерами Delta. Завдяки простому графічному інтерфейсу

програма дозволяє швидко створювати інтуїтивно зрозумілі панелі управління для операторів, що значно полегшує роботу з обладнанням.

Motion Control Studio. Це програмне забезпечення для розробки та управління сервоприводами та іншими високоточними механізмами. Воно дозволяє оптимізувати налаштування руху та синхронізувати дії різних компонентів у виробничих лініях [9].

Таким чином, різні виробники пропонують ПЛК із урахуванням потреб конкретних галузей і умов експлуатації. Розмаїття моделей та функціональних можливостей дозволяє підприємствам обирати оптимальні рішення для автоматизації своїх процесів. Інтеграція з сучасними комунікаційними протоколами та підтримка інноваційних технологій, таких як ІоТ і хмарні сервіси, роблять ці контролери ключовим елементом сучасних виробничих систем.

Висновок до розділу

У даному розділі було описано програмовані логічні контролери (ПЛК), їхню структуру та принципи роботи, зокрема цикл сканування та обробку даних. Наведено переваги використання ПЛК у промисловій автоматизації, включаючи гнучкість, надійність та можливість інтеграції з іншими системами. Окремо було описано програмне забезпечення для розробки, налагодження та управління ПЛК, а також мови програмування, які відповідають стандарту ІЕС 61131-3.

Проведено аналіз моделей і серій ПЛК різних виробників, таких як Siemens, Allen-Bradley, Schneider Electric, Mitsubishi Electric та Delta Electronics, з описом їх особливостей, функціональних можливостей та сфер застосування. Висвітлено ключові ринки для кожного виробника та специфіку використання ПЛК у різних галузях промисловості. Також розглянуто приклади проектів автоматизації на основі ПЛК, їх завдання та роль у підвищенні ефективності виробничих процесів.

2 ФОРМУЛЮВАННЯ ВИМОГ ДО РОЗРОБЛЯЄМОГО ЛАБОРАТОРНОГО СТЕНДУ

Розроблений лабораторний стенд має відповідати наступним критеріям: зручність експлуатації; наочність монтажу; можливість модернізації.

Лабораторний стенд має складатися з вхідних та вихідних сигналів та елементів що будуть їх оброблювати, а саме двох ПЛК.

Вхідні сигнали складаються з: перемикачів, потенціометрів. Вхідні сигнали оброблює ПЛК та створює керуючий сигнал, який подається на сигнальні лампи та проміжні реле.

Лабораторний стенд буде складатися з 14 двопозиційних вимикача, 24 сигнальних ламп та 24 проміжних реле. Для силового живлення застосовано автоматичний вимикач та блок живлення.

В залежності від завдання за допомогою перемикачів SW1-SW14 та потенціометрів R2, R4, R6, R8, R10, R12 подаються сигнали на ПЛК. Після обробки сигналів можна побачити роботу схеми та програм на стенді.

Розроблений стенд має дати змогу попрацювати з ПЛК серії Simatic S7-300, Simatic S7-1200, OPC UA, SCADA системою.

Висновок до розділу

Сформульовано вимоги до стенду, це необхідний етап задля здійснення вибору обладнання та подальшого монтажу лабораторного стенду.

3 РОЗРОБКА СХЕМИ ТА ВИБІР ОБЛАДНАННЯ

В лабораторному стенді використано Siemens Simatic S7-300 314-6CG03-0AB0 [10], технічні параметри якого наведено в таблиці 3.1.

Таблиця 3.1 – Технічні параметри Siemens Simatic S7-300 314-6CG03-0AB0

Параметр	Значення
Напруга живлення	24В DC
Захист від зворотної полярності напруги	Є
Номінальний споживаний струм	1 А
Кількість цифрових входів	24
Кількість цифрових виходів	16
Кількість аналогових входів	4
Кількість аналогових виходів	2
Тип виходів	Транзисторний
Розрядність АЦП	12 біт
Розрядність ЦАП	12 біт
Максимальний струм виходу	0.5А
Максимальний розмір блоку	<64 Кбайт
Протоколи	MPI, PROFIBUS DP
Кількість таймерів	256
Енергонезалежної пам'яті	256 байт
Високочастотні входи	4 по 60 кГц
Високочастотні виходи	4 по 2.5 кГц

Використано модуль розширення CP 343-1 Lean [11], технічні параметри якого наведено в таблиці 3.2.

Таблиця 3.2 – Технічні параметри CP 343-1 Lean

Параметр	Значення
Напруга живлення	24В DC
Протоколи	PROFINET, TCP/IP
Інтерфейси	2 порти RJ45

Використано модуль розширення 331-7KF02-0AB0 [12], технічні параметри якого наведено в таблиці 3.3.

Таблиця 3.3 – Технічні параметри 331-7KF02-0AB0

Параметр	Значення
Напруга живлення	24В DC
Кількість аналогових входів	8
Розрядність АЦП	9/12/14 біт

Використано модуль розширення 332-5HB01-0AB0 [13], технічні параметри якого наведено в таблиці 3.4.

Таблиця 3.4 – Технічні параметри 332-5HB01-0AB0

Параметр	Значення
Напруга живлення	24В DC
Кількість аналогових виходів	2
Розрядність ЦП	11/12 біт

Використано Siemens Simatic S7-1200 214-1AG40-0XB0 [14], технічні параметри якого наведено в таблиці 3.5.

Таблиця 3.5 – Технічні параметри 214-1AG40-0XB0

Параметр	Значення
Напруга живлення	24В DC
Захист від зворотної полярності напруги	Є
Номінальний споживаний струм	1 А
Кількість цифрових входів	14
Кількість цифрових виходів	10
Кількість аналогових входів	2
Тип виходів	Транзисторний
Розрядність АЦП	12 біт
Максимальний струм виходу	0.5А
Максимальний розмір блоку	<64 Кбайт
Протоколи	PROFINET
Кількість таймерів	256
Енергонезалежної пам'яті	256 байт
Високочастотні входи	14 по 100 кГц
Високочастотні виходи	4 по 100 кГц

Використано автоматичний вимикач Siemens Sirius 3RV2011-1FA10 [15], технічні параметри якого наведено в таблиці 3.6.

Таблиця 3.6 – Технічні параметри Siemens Sirius 3RV2011-1FA10

Параметр	Значення
Максимальна вхідна напруга	690 В
Кількість полюсів	3
Номінальний струм	4 А
Струм короткого замикання	100 кА

Використано блок живлення Siemens SITOP PSU8200 6EP3436-8SB00-0AY0 [16], технічні параметри якого наведено в таблиці 3.7.

Таблиця 3.7 – Технічні параметри Siemens SITOP PSU8200 6EP3436-8SB00-0AY0

Параметр	Значення
Діапазон вхідної напруги	400-575 В
Кількість полюсів	3
Максимальна потужність	480 Вт

Використано реле ETI MER2-24DC [17], технічні параметри якого наведено в таблиці 3.8.

Таблиця 3.8 – Технічні параметри ETI MER2-24DC

Параметр	Значення
Тип	Електромеханічне
Напруга живлення	24 В DC
Кількість СО контактів	2
Номінальний струм	8 А
Комутаційна напруга	250 В

Використано сигнальні лампи АСКО-УКРРЕМ AD16-16DS 24В [18], технічні параметри якого наведено в таблиці 3.9.

Таблиця 3.9 – Технічні параметри АСКО-УКРРЕМ AD16-16DS 24В

Параметр	Значення
Тип джерела світла	Світлодіодна матриця
Напруга живлення	24 В DC
Номінальний струм	6 мА

Використано резистор [19], технічні параметри якого наведено в таблиці 3.10.

Таблиця 3.10 – Технічні параметри резистора

Параметр	Значення
Опір	8.2 кОм
Номінальна потужність	0.25 Вт
Максимальний допуск опору	5%

Використано потенціометр [20], технічні параметри якого наведено в таблиці 3.10.

Таблиця 3.10 – Технічні параметри резистора

Параметр	Значення
Опір	5.6 кОм
Номінальна потужність	2 Вт
Максимальний допуск опору	5%

Розроблена схема електрична принципова показана на Додатку А.

Висновок до розділу

Обрано обладнання та розроблено схему електричну принципову, це необхідний етап для монтажу лабораторного стенду.

4 РОЗРОБКА ЗАВДАНЬ ТА МЕТОДОГІЧНИХ ВКАЗІВОК ДО ЛАБОРАТОРНОГО СТЕНДУ

4.1 Розробка загальних вимог до завдання

Загальне завдання до виконання студентам:

1. Розробка програми на ПЛК S7-300; підключення до контролеру до модульного OPC-серверу; підключення Simulink-моделі до OPC-серверу.
2. Розробка програми на ПЛК S7-1200.
3. Розробка зв'язку між контролерами та модернізація програм (додавання PID регулятора).
4. Розробка SCADA.
5. Підключення SCADA до контролерів за допомогою модульного OPC-серверу.

4.2 Розробка варіантів завдання

В дипломному проекті для виконання лабораторних робіт було розроблено шість завдань, які поділені між шістьма бригадами. В цілому на лабораторних роботах буде виконуватися постійна модернізація попередньо виконаної роботи. Номер бригади відповідає номеру завдання.

Для імітації вхідних дискретних сигналів використовуються двопозиційні перемикачі SW1-SW14, для вхідних аналогових сигналів використовуються потенціометри R2, R4, R6, R8, R10, R12, для вихідних дискретних сигналів використовуються світлові індикатори HL1-HL24.

Завдання №1

Технологічний процес: Автоматизована насосна станція. Має забезпечити наступне: підтримка заданого рівня води в резервуарі; контроль тиску в трубопроводах; автоматичне реагування на аварійні ситуації (перегрів насосів, переповнення резервуара); управління засувками для регулювання потоку води.

Опис сигналів

1. Вхідні сигнали

Дискретні (14 сигналів):

Контролер 1 (8 сигналів):

Low_Level – Датчик низького рівня води (BOOL).

Medium_Level – Датчик середнього рівня води (BOOL).

High_Level – Датчик високого рівня води (BOOL).

Critical_Level – Датчик критичного рівня води (BOOL).

Pump1_Status – Стан насоса 1 (робочий/непрацюючий, BOOL).

Pump2_Status – Стан насоса 2 (робочий/непрацюючий, BOOL).

Pump3_Status – Стан насоса 3 (робочий/непрацюючий, BOOL).

Pump4_Status – Стан насоса 4 (робочий/непрацюючий, BOOL).

Контролер 2 (6 сигналів):

Overheat – Перегрів двигуна насоса (BOOL).

Overflow – Переповнення резервуара (BOOL).

Valve1_State – Засувка 1 (відкрита/закрита, BOOL).

Valve2_State – Засувка 2 (відкрита/закрита, BOOL).

Valve3_State – Засувка 3 (відкрита/закрита, BOOL).

Valve4_State – Засувка 4 (відкрита/закрита, BOOL).

Аналогові (6 сигналів):

Контролер 1 (4 сигнали):

Water_Level – Поточний рівень води в резервуарі (REAL).

Pipeline_Pressure – Тиск у трубопроводі (REAL).

Flow1 – Витрата води на вході (REAL).

Flow2 – Витрата води на виході (REAL).

Контролер 2 (2 сигнали):

Motor_Temperature – Температура двигуна насоса (REAL).

Output_Pressure – Тиск на виході з насоса (REAL).

2. Вихідні сигнали

Контролер 1 (16 сигналів):

Pump1_Control – Увімкнення/вимкнення насоса 1 (BOOL).

Pump2_Control – Увімкнення/вимкнення насоса 2 (BOOL).

Pump3_Control – Увімкнення/вимкнення насоса 3 (BOOL).

Pump4_Control – Увімкнення/вимкнення насоса 4 (BOOL).

Valve1_Control – Відкрити/закрити засувку 1 (BOOL).

Valve2_Control – Відкрити/закрити засувку 2 (BOOL).

Valve3_Control – Відкрити/закрити засувку 3 (BOOL).

Valve4_Control – Відкрити/закрити засувку 4 (BOOL).

Alarm1 – Аварійний сигнал 1 (BOOL).

Alarm2 – Аварійний сигнал 2 (BOOL).

Alarm3 – Аварійний сигнал 3 (BOOL).

Alarm4 – Аварійний сигнал 4 (BOOL). 13–16. Резервні виходи.

Контролер 2 (8 сигналів):

Aux_Pump1_Control – Управління допоміжним насосом 1 (BOOL).

Aux_Pump2_Control – Управління допоміжним насосом 2 (BOOL).

Aux_Valve1_Control – Управління додатковою засувкою 1 (BOOL).

Aux_Valve2_Control – Управління додатковою засувкою 2 (BOOL).

Aux_Valve3_Control – Управління додатковою засувкою 3 (BOOL).

Aux_Valve4_Control – Управління додатковою засувкою 4 (BOOL).

Alarm5 – Аварійний сигнал 5 (BOOL).

Alarm6 – Аварійний сигнал 6 (BOOL).

Технологічний процес організований за такими кроками:

1. Моніторинг вхідних сигналів

Контролери постійно зчитують стан вхідних сигналів:

Дискретні сигнали: стан рівня води, стан насосів, стан засувки, аварійні сигнали.

Аналогові сигнали: поточний рівень води, тиск у трубопроводах, витрата води, температура двигуна.

2. Контролер 1: Основна логіка управління насосами

А. Управління насосами

Читається поточний рівень води за аналоговим датчиком `Water_Level`.

Порівнюється рівень води з заданим значенням (`SetPoint`):

Якщо рівень нижчий за встановлене значення мінус мертвої зони (наприклад, `SetPoint - DeadZone`), запускаються насоси.

Якщо рівень вищий за задане значення плюс мертва зона (`SetPoint + DeadZone`), насоси зупиняються.

Якщо критичний рівень (`Critical_Level`) активується, усі насоси аварійно зупиняються.

Б. Управління засувками

Засувки відкриваються або закриваються залежно від рівня води:

Якщо низький рівень (`Low_Level`), відкривається засувка для наповнення резервуара.

Якщо високий рівень (`High_Level`), засувка закривається.

В. Аварійна сигналізація

Якщо насос має бути увімкненим (згідно з рівнем води), але його стан показує, що він вимкнений (`Pump_Status = 0`), активується аварійний сигнал.

Якщо рівень води досягає критичної позначки (`Critical_Level`), активується аварійна сигналізація.

3. Контролер 2: Додаткові функції управління

А. Управління допоміжними насосами

Температура двигуна насосів порівнюється із пороговими значеннями:

Якщо температура нижча за 70°C , основний допоміжний насос (`Aux_Pump1_Control`) вмикається.

Якщо температура перевищує 75°C , насос зупиняється.

Б. Управління засувками

Стан засувки перевіряється (Valve_State):

Якщо засувка відкрита, вона закривається, і навпаки.

Логіка забезпечує синхронізацію роботи насосів і засувки.

В. Аварійна сигналізація

У разі спрацювання аварійних сигналів, таких як перегрів насоса або переповнення резервуара, активується відповідна сигналізація.

Дані аварійні сигнали також можуть передаватися на операторську панель для негайного реагування.

4. Узгодження роботи контролерів

Контролер 1 передає дані про поточний стан рівня води та стан основних насосів до контролера 2.

Контролер 2 може активувати додаткові насоси або управляти засувками в разі аварійного стану.

5. Робота ПІД-регулятора

ПІД-регулятор в контролері 1 розраховує сигнал для увімкнення/вимкнення насосів на основі:

Поточного рівня води.

Заданого значення рівня.

Поточних витрат води.

Мертва зона в регуляторі мінімізує часте перемикання станів насосів:

Поки рівень води залишається в межах мертвої зони, насос не перемикається.

6. Реагування на аварійні ситуації

Якщо спрацював сигнал аварії:

Насоси автоматично зупиняються.

Засувки переводяться у безпечний стан.

Активується звукова або візуальна сигналізація для оператора.

У разі усунення аварії:

Система автоматично повертається до нормального режиму роботи.

7. Робота операторської панелі

Оператор отримує дані про:

Рівень води.

Стан насосів і засувок.

Активність аварійних сигналів.

Оператор може вручну коригувати роботу системи в разі потреби, наприклад, примусово запустити або зупинити насос.

Завдання №2

Технологічний процес: Система управління автоматизованою лінією пакування продуктів. Має забезпечити: автоматична подача продуктів на лінію пакування; контроль роботи конвеєрів, пакувальних машин і транспортерів; реакція на аварійні ситуації, такі як застрягання продукту або переповнення накопичувача.

1. Вхідні сигнали

Дискретні (14 сигналів):

Контролер 1 (8 сигналів):

Start_Signal – Сигнал запуску системи (BOOL).

Stop_Signal – Сигнал зупинки системи (BOOL).

Product_In_Sensor – Датчик наявності продукту на вході (BOOL).

Package_Ready_Sensor – Датчик готовності пакувального матеріалу (BOOL).

Conveyor1_Jam_Sensor – Датчик застрягання на конвеєрі 1 (BOOL).

Machine_Failure_Sensor – Датчик несправності пакувальної машини (BOOL).

Conveyor2_Jam_Sensor – Датчик застрягання на конвеєрі 2 (BOOL).

E_Stop_Signal – Сигнал аварійної зупинки (BOOL).

Контролер 2 (6 сигналів):

Output_Full_Sensor – Датчик переповнення вихідного накопичувача (BOOL).

Conveyor3_Jam_Sensor – Датчик застрягання на конвеєрі 3 (BOOL).

Package_Seal_Sensor – Датчик завершення запайки упаковки (BOOL).

Transfer_Sensor – Датчик передачі упаковки на вихідний транспортер (BOOL).

Safety_Door_Sensor – Сигнал відкриття захисних дверей (BOOL).

Reset_Signal – Сигнал скидання аварійного стану (BOOL).

Аналогові (6 сигналів):

Контролер 1 (4 сигнали):

Conveyor_Speed1 – Поточна швидкість першого конвеєра (REAL).

Conveyor_Speed2 – Поточна швидкість другого конвеєра (REAL).

Product_Weight – Вага продукту (REAL).

Package_Length – Довжина пакування (REAL).

Контролер 2 (2 сигнали):

Seal_Temperature – Температура нагрівача пакувальної машини (REAL).

Transfer_Speed – Поточна швидкість транспортеру на виході (REAL).

2. Вихідні сигнали

Контролер 1 (16 сигналів):

Conveyor1_Control – Управління першим конвеєром (BOOL).

Conveyor2_Control – Управління другим конвеєром (BOOL).

Package_Machine_Control – Управління пакувальною машиною (BOOL).

Low_Speed1 – Низька швидкість для конвеєра 1 (BOOL).

Medium_Speed1 – Середня швидкість для конвеєра 1 (BOOL).

High_Speed1 – Висока швидкість для конвеєра 1 (BOOL).

Low_Speed2 – Низька швидкість для конвеєра 2 (BOOL).

Medium_Speed2 – Середня швидкість для конвеєра 2 (BOOL).

High_Speed2 – Висока швидкість для конвеєра 2 (BOOL). 10–16.

Резервні виходи.

Контролер 2 (8 сигналів):

Conveyor3_Control – Управління третім конвеєром (BOOL).

Transfer_Control – Управління вихідним транспортером (BOOL).

Seal_Heater_Control – Управління нагрівачем для запайки (BOOL).

Alarm_Signal – Сигнал аварії (BOOL).

Reset_Control – Сигнал скидання аварії (BOOL). 6–8. Резервні виходи.

Технологічний процес організований за такими кроками:

1. Моніторинг вхідних сигналів

Контролери постійно зчитують стан вхідних сигналів:

Дискретні сигнали:

Стан запуску та зупинки системи.

Стан датчиків наявності продукту, готовності пакувального матеріалу.

Стан застрягання продукту на конвеєрах.

Сигнали аварійної зупинки, відкриття захисних дверей, переповнення накопичувача.

Аналогові сигнали:

Поточна швидкість конвеєрів.

Температура нагрівача пакувальної машини.

Вага продукту та довжина пакування.

2. Контролер 1: Основна логіка управління лінією

A. Управління конвеєрами 1 і 2

Система зчитує сигнал Product_In_Sensor (наявність продукту) і Package_Ready_Sensor (готовність пакувального матеріалу).

У разі наявності продукту та готовності матеріалу:

Конвеєр 1 активується, сигнал Conveyor1_Control переходить у стан TRUE.

Конвеєр 2 активується, сигнал Conveyor2_Control переходить у стан TRUE.

У разі відсутності продукту або матеріалу:

Конвеєри 1 і 2 зупиняються.

Б. Управління швидкістю конвеєра 1

Поточна швидкість зчитується за допомогою аналогового сигналу Conveyor_Speed1.

ПІД-регулятор обчислює необхідну швидкість для підтримки заданого потоку продуктів.

Вихід регулятора порівнюється з межами мертвої зони:

Якщо швидкість менша за задане значення мінус мертва зона, активується Low_Speed1.

Якщо швидкість в межах мертвої зони, активується Medium_Speed1.

Якщо швидкість перевищує задане значення плюс мертва зона, активується High_Speed1.

В. Аварійна сигналізація

Якщо Conveyor1_Jam_Sensor або Conveyor2_Jam_Sensor вказує на застрягання:

Аварійний сигнал Alarm_Signal переходить у стан TRUE.

Відповідний конвеєр зупиняється.

У разі сигналу E_Stop_Signal (аварійна зупинка):

Усі конвеєри та пакувальна машина зупиняються.

3. Контролер 2: Додаткові функції управління

А. Управління конвеєром 3

Конвеєр 3 активується, якщо:

Немає переповнення вихідного накопичувача (`Output_Full_Sensor = FALSE`).

Немає застрягання продукту на конвеєрі 3 (`Conveyor3_Jam_Sensor = FALSE`).

Якщо спрацює будь-який з цих сигналів, конвеєр 3 зупиняється.

Б. Управління транспортером

Транспортер активується, якщо немає переповнення вихідного накопичувача:

`Transfer_Control = TRUE`.

У разі переповнення транспортер зупиняється.

В. Управління нагрівачем пакувальної машини

Сигнал `Seal_Heater_Control` активується, якщо температура нагрівача `Seal_Temperature` перевищує 150°C.

Якщо температура падає нижче 150°C, нагрівач вимикається.

Г. Аварійна сигналізація

Якщо `Conveyor3_Jam_Sensor` або `Output_Full_Sensor` активується:

Аварійний сигнал `Alarm_Signal` переходить у стан `TRUE`.

У разі надходження сигналу `Reset_Signal`, аварійний стан скидається.

4. Узгодження роботи контролерів

Контролер 1 відповідає за подачу продуктів і пакування.

Контролер 2 забезпечує транспортування готових упаковок і роботу нагрівача.

Дані про стан конвеєрів, транспортерів і аварійні сигнали можуть передаватися між контролерами для синхронізації.

5. Робота ПІД-регулятора

ПІД-регулятор у контролері 1 забезпечує стабільну швидкість першого конвеєра:

Вхідні параметри: поточна швидкість, заданий потік продуктів.

Вихід: сигнал вибору швидкості (низька, середня, висока).

Мертва зона мінімізує часті перемикання між рівнями швидкості:

Поки швидкість у межах мертвої зони, сигнал швидкості не змінюється.

6. Реагування на аварійні ситуації

Якщо спрацьовує сигнал аварії:

Відповідний конвеєр, транспортер або пакувальна машина зупиняється.

Активується аварійний сигнал для оператора.

У разі надходження сигналу Reset_Signal:

Аварійний стан скидається.

Система повертається до нормального режиму роботи.

7. Робота операторської панелі

Оператор отримує візуальну інформацію про:

Стан конвеєрів і транспортерів.

Поточну швидкість і температуру нагрівача.

Аварійні сигнали.

Оператор може вручну:

Запустити або зупинити систему.

Скинути аварійний стан.

Відрегулювати параметри пакувальної машини.

Завдання №3

Технологічний процес: Система управління автоматизованою сушаркою для зерна. Має реалізувати наступне: забезпечити контроль температури та вологості в сушарці для досягнення заданих параметрів; регулювати потік зерна через сушарку для досягнення оптимальної ефективності процесу; реагувати на аварійні ситуації, наприклад, перегрів сушарки або блокування зерна.

1. Вхідні сигнали

Дискретні (14 сигналів):

Контролер 1 (8 сигналів):

Start_Signal – Сигнал запуску системи (BOOL).

Stop_Signal – Сигнал зупинки системи (BOOL).

Grain_In_Sensor – Датчик наявності зерна на вході (BOOL).

Overheat_Sensor – Датчик перегріву сушарки (BOOL).

Fan1_Jam_Sensor – Датчик застрягання вентилятора 1 (BOOL).

Fan2_Jam_Sensor – Датчик застрягання вентилятора 2 (BOOL).

Emergency_Stop – Сигнал аварійної зупинки (BOOL).

Grain_Flow_Blocked – Сигнал блокування потоку зерна (BOOL).

Контролер 2 (6 сигналів):

Dryer_Full_Sensor – Датчик заповнення сушарки (BOOL).

Dryer_Empty_Sensor – Датчик спорожнення сушарки (BOOL).

Exhaust_Jam_Sensor – Датчик застрягання вихідного вентилятора (BOOL).

Temperature_Limit_Sensor – Сигнал перевищення температурного ліміту (BOOL).

Humidity_Sensor – Датчик підвищеної вологості (BOOL).

Reset_Signal – Сигнал скидання аварії (BOOL).

Аналогові (6 сигналів):

Контролер 1 (4 сигнали):

Input_Temperature – Температура на вході сушарки (REAL).

Output_Temperature – Температура на виході сушарки (REAL).

Input_Humidity – Вологість зерна на вході (REAL).

Output_Humidity – Вологість зерна на виході (REAL).

Контролер 2 (2 сигнали):

Fan1_Speed – Швидкість вентилятора 1 (REAL).

Fan2_Speed – Швидкість вентилятора 2 (REAL).

2. Вихідні сигнали

Контролер 1 (16 сигналів):

Dryer_Control – Управління сушаркою (BOOL).

Fan1_Control – Управління вентилятором 1 (BOOL).

Fan2_Control – Управління вентилятором 2 (BOOL).

Low_Fan1_Speed – Низька швидкість вентилятора 1 (BOOL).

Medium_Fan1_Speed – Середня швидкість вентилятора 1 (BOOL).

High_Fan1_Speed – Висока швидкість вентилятора 1 (BOOL).

Low_Fan2_Speed – Низька швидкість вентилятора 2 (BOOL).

Medium_Fan2_Speed – Середня швидкість вентилятора 2 (BOOL).

High_Fan2_Speed – Висока швидкість вентилятора 2 (BOOL). 10–16.

Резервні виходи.

Контролер 2 (8 сигналів):

Exhaust_Fan_Control – Управління вихідним вентилятором (BOOL).

Conveyor_Control – Управління транспортером зерна (BOOL).

Alarm_Signal – Сигнал аварії (BOOL).

Reset_Control – Сигнал скидання аварії (BOOL). 5–8. Резервні виходи.

Технологічний процес організований за такими кроками:

1. Моніторинг вхідних сигналів

Контролери постійно зчитують стан вхідних сигналів:

Дискретні сигнали:

Сигнали запуску та зупинки системи.

Сигнали наявності зерна, перегріву сушарки, блокування потоку зерна.

Стан заповнення або спорожнення сушарки.

Сигнали аварійної зупинки та скидання аварійного стану.

Аналогові сигнали:

Температура на вході та виході сушарки.

Вологість зерна на вході та виході.

Швидкість вентиляторів.

2. Контролер 1: Основна логіка управління сушаркою

А. Управління сушаркою та вентиляторами

Сушарка:

Читається сигнал Start_Signal для запуску системи.

Якщо система запущена та немає сигналу перегріву (Overheat_Sensor = FALSE), активується сушарка:

Dryer_Control = TRUE.

Якщо сигнал перегріву активний або надійшов сигнал зупинки (Stop_Signal = TRUE), сушарка вимикається.

Вентилятори:

Обидва вентилятори активуються разом із сушаркою:

Fan1_Control = TRUE.

Fan2_Control = TRUE.

У разі застрягання вентилятора (Fan1_Jam_Sensor або Fan2_Jam_Sensor) відповідний вентилятор зупиняється.

Б. Управління швидкістю вентилятора 1

ПІД-регулятор розраховує швидкість вентилятора 1 на основі:

Поточної температури на вході сушарки (Input_Temperature).

Заданого значення температури (SetPoint).

Результат порівнюється з межами мертвої зони:

Якщо значення нижче за SetPoint - DeadZone, активується Low_Fan1_Speed.

Якщо значення в межах SetPoint $\pm$ DeadZone, активується Medium_Fan1_Speed.

Якщо значення вище за SetPoint + DeadZone, активується High_Fan1_Speed.

В. Аварійна сигналізація

Якщо активується будь-який з таких сигналів:

Overheat_Sensor (перегрів).

Fan1_Jam_Sensor або Fan2_Jam_Sensor (застрягання вентилятора).

Grain_Flow_Blocked (блокування потоку зерна).

Система зупиняє сушарку та вентилятори, а Alarm_Signal переходить у стан TRUE.

3. Контролер 2: Додаткові функції управління

А. Управління транспортером

Читається стан датчиків заповнення та спорожнення сушарки:

Якщо сушарка не заповнена (Dryer_Full_Sensor = FALSE) і не порожня (Dryer_Empty_Sensor = FALSE), активується транспортер:

Conveyor_Control = TRUE.

Якщо сушарка заповнена або порожня, транспортер зупиняється.

Б. Управління вихідним вентилятором

Читається сигнал Exhaust_Jam_Sensor:

Якщо немає застрягання, вентилятор працює:

Exhaust_Fan_Control = TRUE.

У разі застрягання вентилятор зупиняється.

В. Аварійна сигналізація

Якщо активуються будь-які з таких сигналів:

Exhaust_Jam_Sensor (застрягання вихідного вентилятора).

Dryer_Full_Sensor (сушарка переповнена).

Temperature_Limit_Sensor (перевищення температурного ліміту).

Активується аварійний сигнал Alarm_Signal = TRUE.

Якщо надходить сигнал Reset_Signal, аварійний стан скидається.

4. Узгодження роботи контролерів

Контролер 1 управляє основними процесами: сушаркою, вентиляторами, температурою.

Контролер 2 відповідає за транспортування зерна та роботу вихідного вентилятора.

Дані про аварійні стани передаються між контролерами для узгодження роботи.

5. Робота ПД-регулятора

ПД-регулятор в контролері 1 обчислює необхідну швидкість вентилятора 1 для підтримки заданої температури.

Вихід ПД-регулятора обмежується трьома рівнями:

Низька швидкість.

Середня швидкість.

Висока швидкість.

Мертва зона мінімізує часті перемикання між рівнями.

6. Реагування на аварійні ситуації

У разі спрацювання аварійного сигналу:

Сушарка та вентилятори зупиняються.

Транспортер і вихідний вентилятор зупиняються.

Активується аварійний сигнал для оператора.

У разі надходження сигналу Reset_Signal, аварійний стан скидається, і система повертається до нормального режиму.

7. Робота операторської панелі

Оператор отримує інформацію про:

Температуру та вологість зерна.

Стан сушарки, транспортерів і вентиляторів.

Активність аварійних сигналів.

Оператор може вручну:

Запустити або зупинити систему.

Скинути аварійний стан.

Коригувати задану температуру.

Завдання №4

Технологічний процес: Система управління лінією змішування і фасування рідини

Мета:

Автоматизоване змішування рідких компонентів до заданих параметрів.

Контроль рівня і температури у змішувальній ємності.

Перекачування готової суміші до фасувальної лінії.

Реакція на аварійні ситуації, такі як переповнення ємності або перегрів.

1. Вхідні сигнали

Дискретні (14 сигналів):

Контролер 1 (8 сигналів):

Start_Signal – Сигнал запуску системи (BOOL).

Stop_Signal – Сигнал зупинки системи (BOOL).

Level_Low_Sensor – Датчик низького рівня у ємності (BOOL).

Level_High_Sensor – Датчик високого рівня у ємності (BOOL).

Overheat_Sensor – Датчик перегріву рідини (BOOL).

Pump_Failure_Sensor – Датчик несправності насоса (BOOL).

Valve1_State – Стан клапана 1 (відкрита/закрита, BOOL).

Emergency_Stop – Сигнал аварійної зупинки (BOOL).

Контролер 2 (6 сигналів):

Conveyor_Jam_Sensor – Датчик застрягання фасувальної лінії (BOOL).

Bottle_Full_Sensor – Датчик наповнення пляшки (BOOL).

Seal_Failure_Sensor – Датчик несправності запаювання (BOOL).

Conveyor_Empty_Sensor – Датчик порожньої лінії (BOOL).

Safety_Door_Sensor – Сигнал відкриття захисних дверей (BOOL).

Reset_Signal – Сигнал скидання аварійного стану (BOOL).

Аналогові (6 сигналів):

Контролер 1 (4 сигнали):

Tank_Temperature – Температура у змішувальній ємності (REAL).

Tank_Level – Поточний рівень рідини у ємності (REAL).

Input_Flow1 – Потік першого компонента (REAL).

Input_Flow2 – Потік другого компонента (REAL).

Контролер 2 (2 сигнали):

Conveyor_Speed – Швидкість транспортеру фасувальної лінії (REAL).

Output_Flow – Потік рідини на виході (REAL).

2. Вихідні сигнали

Контролер 1 (16 сигналів):

Mixer_Control – Управління мішалкою (BOOL).

Pump_Control – Управління насосом (BOOL).

Heater_Control – Управління нагрівачем (BOOL).

Low_Input_Flow – Низький потік компоненту 1 (BOOL).

Medium_Input_Flow – Середній потік компоненту 1 (BOOL).

High_Input_Flow – Високий потік компоненту 1 (BOOL).

Low_Input_Flow2 – Низький потік компоненту 2 (BOOL).

Medium_Input_Flow2 – Середній потік компоненту 2 (BOOL).

High_Input_Flow2 – Високий потік компоненту 2 (BOOL). 10–16.

Резервні виходи.

Контролер 2 (8 сигналів):

Conveyor_Control – Управління транспортером (BOOL).

Sealer_Control – Управління запаюванням (BOOL).

Alarm_Signal – Сигнал аварії (BOOL).

Reset_Control – Сигнал скидання аварії (BOOL). 5–8. Резервні виходи.

Технологічний процес організований за такими кроками:

1. Моніторинг вхідних сигналів

Контролери постійно зчитують стан вхідних сигналів:

Дискретні сигнали:

Стан запуску та зупинки системи.

Стан рівня рідини в змішувальній ємності.

Стан перегріву рідини та несправності насоса.

Стан фасувальної лінії (застрягання, наповнення пляшок, стан транспортеру).

Аналогові сигнали:

Поточна температура у змішувальній ємності.

Поточний рівень рідини.

Потік компонентів на вході.

Швидкість транспортеру.

2. Контролер 1: Основна логіка управління змішуванням

А. Управління змішувальною ємністю

Запуск системи:

Якщо сигнал Start_Signal активний, система розпочинає змішування.

Якщо сигнал Stop_Signal або Emergency_Stop активний, система зупиняється.

Контроль рівня:

Якщо рівень у ємності нижче заданого (аналоговий сигнал Tank_Level), активуються подачі компонентів через відповідні клапани:

Low_Input_Flow, Medium_Input_Flow, або High_Input_Flow залежно від рівня потоку.

Якщо рівень перевищує заданий, клапани закриваються.

Управління нагрівачем:

Нагрівач активується для підтримання заданої температури у ємності.

Температура контролюється ПІД-регулятором:

Якщо температура нижче SetPoint - DeadZone, активується Low_Input_Flow.

Якщо температура в межах $\text{SetPoint} \pm \text{DeadZone}$, активується `Medium_Input_Flow`.

Якщо температура перевищує $\text{SetPoint} + \text{DeadZone}$, активується `High_Input_Flow`.

Б. Управління мішалкою

Мішалка активується разом із запуском системи:

`Mixer_Control = TRUE`.

Якщо активується сигнал перегріву (`Overheat_Sensor`), мішалка зупиняється.

В. Аварійна сигналізація

Якщо активується один із сигналів:

`Overheat_Sensor` (перегрів рідини).

`Pump_Failure_Sensor` (несправність насоса).

`Level_High_Sensor` (переповнення ємності).

Система зупиняє всі процеси, активує сигнал `Alarm_Signal`.

3. Контролер 2: Управління фасувальною лінією

А. Управління транспортером

Читається сигнал стану лінії:

Якщо транспортер порожній (`Conveyor_Empty_Sensor`) і немає застрягань (`Conveyor_Jam_Sensor`), він активується:

`Conveyor_Control = TRUE`.

Якщо є застрягання, транспортер зупиняється.

Якщо пляшки на лінії заповнені (`Bottle_Full_Sensor`), транспортер переходить у режим очікування.

Б. Управління запаюванням

Якщо пляшки не заповнені, запаювання не активується:

`Sealer_Control = FALSE`.

Якщо пляшки заповнені, запаювання активується:

`Sealer_Control = TRUE`.

В. Аварійна сигналізація

Якщо активуються сигнали:

Conveyor_Jam_Sensor (застрягання на лінії).

Seal_Failure_Sensor (несправність запаювання).

Активується сигнал Alarm_Signal.

У разі сигналу Reset_Signal, аварійний стан скидається.

4. Узгодження роботи контролерів

Контролер 1 відповідає за основне змішування, контроль рівня та температури.

Контролер 2 забезпечує роботу фасувальної лінії.

Дані про стан рівня та температури передаються до контролера 2 для забезпечення узгодженої роботи системи.

5. Робота ПД-регулятора

ПД-регулятор у контролері 1 обчислює вихід для управління нагрівачем:

Вхід: Tank_Temperature.

Заданий параметр: SetPoint.

Вихід ПД-регулятора визначає, які сигнали активації подачі тепла використовуються:

Низький, середній, або високий рівень нагріву.

Мертва зона запобігає частому перемиканню між рівнями.

6. Реагування на аварійні ситуації

Якщо активується аварійний сигнал:

Система зупиняє всі процеси.

Активується сигнал Alarm_Signal.

У разі усунення причини аварії, оператор може скинути аварійний стан сигналом Reset_Signal, після чого система повертається до нормальної роботи.

7. Робота операторської панелі

Оператор отримує інформацію про:

Рівень і температуру у змішувальній ємності.

Стан транспортерів і фасувальної лінії.

Активність аварійних сигналів.

Оператор може вручну:

Запустити або зупинити систему.

Скинути аварійний стан.

Коригувати задану температуру.

4.3 Створення програми для S7-300

Перед написанням програми необхідно створити проєкт. Для цього у вкладці Project, натискаємо New (рис. 4.1). Після чого висвітлюється вікно Create a new project (рис. 4.2). Вказуємо назву проєкту, шлях для збереження, автора та коментар (за необхідністю).

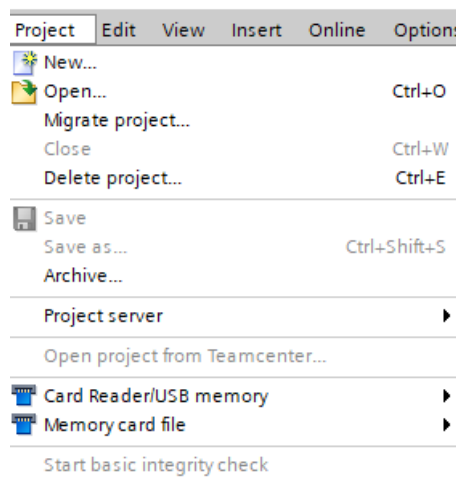


Рисунок 4.1 – Випадаючий список вкладки project

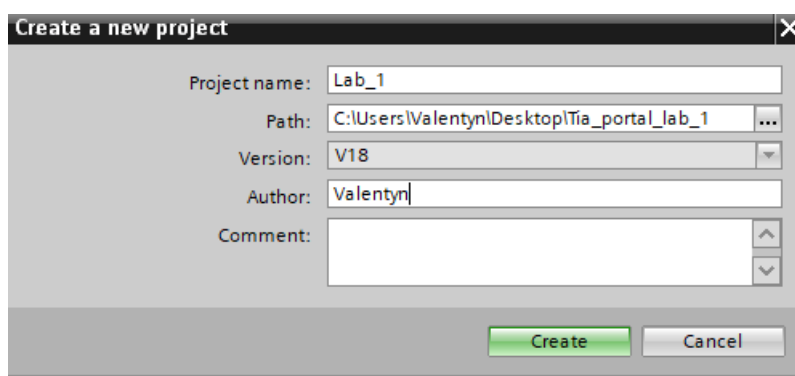


Рисунок 4.2 – Вікно створення нового проєкту

Додаємо новий пристрій, а саме у Project tree натискаємо Add new device, після чого з'являється вікно з відповідною назвою (рис. 4.3). Обираємо Controllers – SIMATIC S7-300 – CPU – CPU 314C-2 DP – 6ES7 314-6CG03-0AB0 v2.6, вказуємо назву PLC, наприклад PLC_1. Натискаємо ОК, внаслідок створюється нульова рейка з розміщеним контролером (рис 4.4).

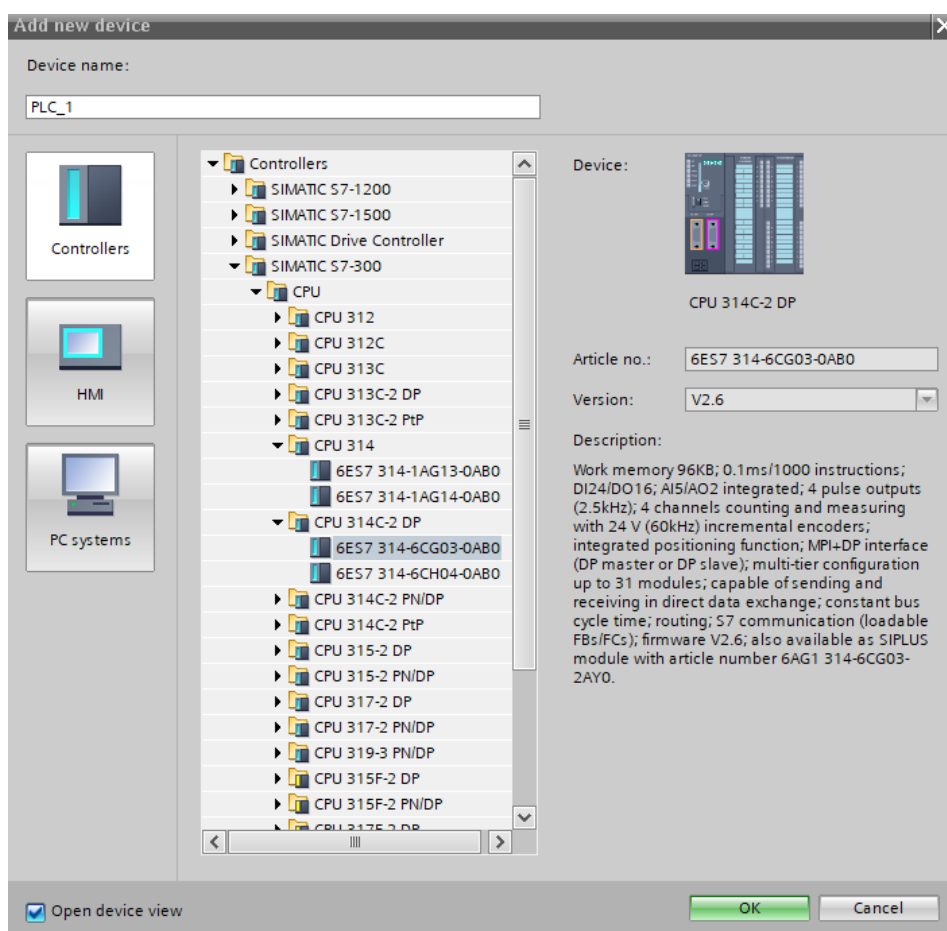


Рисунок 4.3 – Вікно додавання нового пристрою

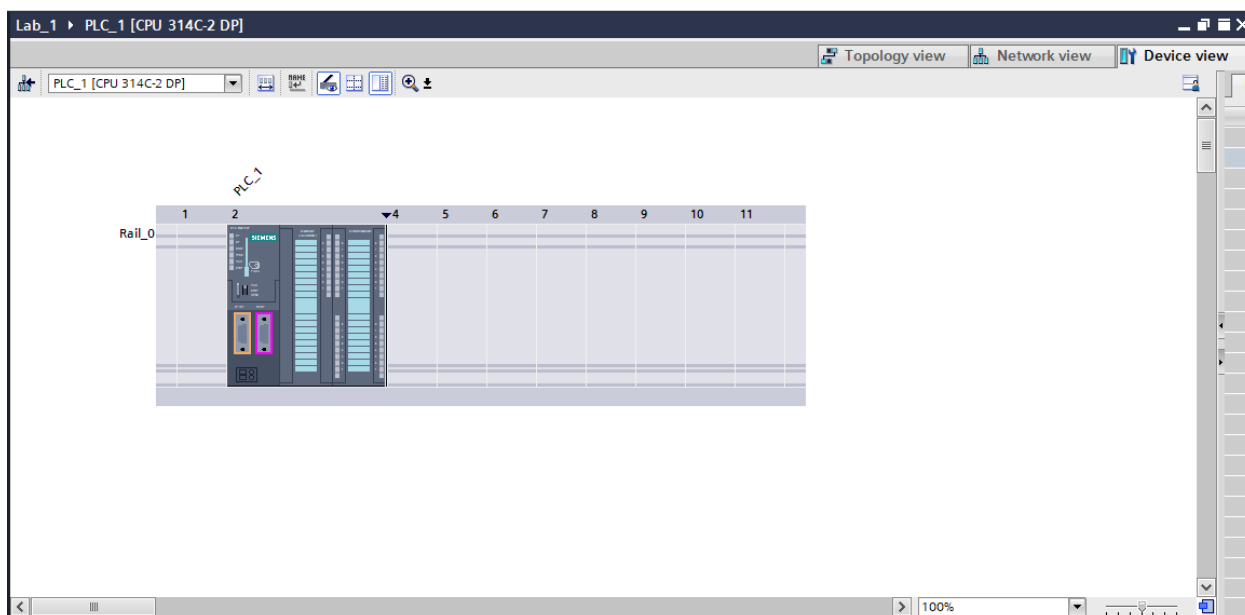


Рисунок 4.4 – Вікно огляду пристроїв

Додаємо модулі наявні модулі розширення на цю рейку до контролеру згідно схеми. В каталозі обираємо Communication modules – PROFINET/Ethernet – CP 343-1 LEAN - 6GK7 343-1CX10-0XE0; AI – AI 8x12BIT – 6ES7 331-7KF02-0AB0; AO – AO 2x12BIT – 6ES 332-5HB01 – 0AB0.

Щоб налаштувати необхідний модуль необхідно його обрати та перейти до вкладки Properties – General. Для виконання лабораторних робіт достатньо буде налаштувати наступні параметри контролеру S7-300:

AI 5/AO 2 – Inputs – Channel: для каналів 0 – 3 встановити тип вимірювання – напруга, границі вимірювання 0...10 V; канал 4 деактивувати.

Protection – Depending on the mode selector setting.

На вкладці IO tags видно тип даних для кожного входу та виходу, а також адресу комірки пам'яті, до якої необхідно буде звертатися в програмі. Ці адреси можна замінити за бажанням.

Для налаштування модуля CP 343-1 LEAN достатньо налаштувати наступні параметри:

PROFINET interface – Ethernet addresses – Internet protocol version 4 (IPv4) – Set IP address in the project – IP address 192.168.0.1, Subnet mask 255.255.255.0.

PROFINET interface – Operating mode – IO device on.

На цьому налаштування проєкту для виконання першої лабораторної роботи завершено. Для створення програми необхідно додати змінні, це можна зробити у вкладці PLC_1 – PLC tags – Default tag table. Адреси змінних починаються з наступних літер: «I» - входи; «M» - проміжні змінні; «Q» - виходи. Приклад наведено на рисунку 4.5.

Default tag table						
	Name	Data type	Address	Retain	Acces...	Visibl...
1	StartButton	Bool	%I0.0		☒	☒
2	StopButton	Bool	%I0.1		☒	☒
3	EmergencyStop	Bool	%I0.2		☒	☒
4	LevelSensorMin1	Bool	%I0.3		☒	☒
5	LeverSensorMin2	Bool	%I0.4		☒	☒
6	ValvesStatusA	Bool	%I0.5		☒	☒
7	ValvesStatusB	Bool	%I0.6		☒	☒
8	LevelAnalogReal	Real	%MD0		☒	☒
9	FlowRateAReal	Real	%MD4		☒	☒
10	FlowRateBReal	Real	%MD8		☒	☒
11	TemperatureReal	Real	%MD12		☒	☒
12	LevelAnalog	Int	%IW16		☒	☒
13	FlowRateA	Int	%IW18		☒	☒
14	FlowRateB	Int	%IW20		☒	☒
15	Temperature	Int	%IW22		☒	☒
16	ValveControlA	Bool	%Q24.0		☒	☒
17	ValveControlB	Bool	%Q24.1		☒	☒
18	PumpControl	Bool	%Q24.2		☒	☒
19	MixerControl	Bool	%Q24.3		☒	☒
20	DischargeValve	Bool	%Q24.4		☒	☒
21	StatusWorking	Bool	%Q24.5		☒	☒
22	StatusError	Bool	%Q24.6		☒	☒

Рисунок 4.5 – Приклад створення змінних

Після оголошення змінних необхідно перейти в program block та створити functional block або function з мовою SCL.

Функціональний блок має власну пам'ять для збереження стану даних між викликами, тобто може зберігати значення змінних після завершення виконання блоку. Функціональний блок підходить для реалізації таких алгоритмів: лічильники, тригери. Функція не має власної пам'яті. Усі змінні, що були оголошені всередині функції існують тільки під час виконання блоку та знищуються після завершення. Тобто функція може використовуватися для виконання обчислень або логіки, які не потребують збереження даних між викликами.. Функція отримує дані через параметри входу/виходу.

Для прикладу буде створено функціональний блок, що реалізує логіку роботи завдання №1, з наступним кодом:

```

FUNCTION_BLOCK "FB_MixingControl"
VAR_INPUT DB_SPECIFIC
  StartButton : Bool; // Кнопка "Пуск"
  StopButton : Bool; // Кнопка "Стоп"
  EmergencyStop : Bool; // Аварійна зупинка
  LevelSensorMin1 : Bool; // Датчик мінімального рівня у першому резервуарі
  LevelSensorMin2 : Bool; // Датчик мінімального рівня у другому резервуарі
  LevelAnalogReal : Real; // Рівень рідини у змішувальному резервуарі (0-10 В)
  FlowRateAReal : Real; // Витрата компонента А (0-10 В)
  FlowRateBReal : Real; // Витрата компонента В (0-10 В)
  TemperatureReal : Real; // Температура у резервуарі (0-10 В)
END_VAR
VAR_OUTPUT DB_SPECIFIC
  ValveControlA : Bool; // Керування клапаном компонента А
  ValveControlB : Bool; // Керування клапаном компонента В
  PumpControl : Bool; // Керування насосом
  MixerControl : Bool; // Керування міксером
  DischargeValve : Bool; // Клапан зливу суміші
  StatusWorking : Bool; // Система працює
  StatusError : Bool; // Аварія
END_VAR
VAR_IN_OUT DB_SPECIFIC
END_VAR
VAR DB_SPECIFIC
  ProcessActive : Bool := FALSE; // Активний процес змішування
END_VAR
VAR_TEMP
END_VAR
VAR CONSTANT
  SP_Level : Real := 5.0; // Заданий рівень рідини
  DeadZone : Real := 0.1; // Мертва зона для рівня
END_VAR

// Аварійна зупинка
IF #EmergencyStop THEN
  #ValveControlA := FALSE;
  #ValveControlB := FALSE;
  #PumpControl := FALSE;
  #MixerControl := FALSE;
  #DischargeValve := FALSE;
  #StatusWorking := FALSE;
  #StatusError := TRUE;
  RETURN;
END_IF;

// Запуск/зупинка процесу
IF #StartButton AND NOT #ProcessActive THEN
  #ProcessActive := TRUE;
  #StatusWorking := TRUE;
  #StatusError := FALSE;
END_IF;

IF #StopButton AND #ProcessActive THEN
  #ProcessActive := FALSE;
  #ValveControlA := FALSE;
  #ValveControlB := FALSE;
  #PumpControl := FALSE;
  #MixerControl := FALSE;
  #DischargeValve := FALSE;
  #StatusWorking := FALSE;

```

```

END_IF;
// Управління рівнем рідини
IF #ProcessActive THEN
  IF #LevelAnalogReal < #SP_Level - #DeadZone THEN
    #ValveControlA := TRUE;
    #ValveControlB := TRUE;
    #PumpControl := TRUE;
  ELSIF #LevelAnalogReal > #SP_Level + #DeadZone THEN
    #ValveControlA := FALSE;
    #ValveControlB := FALSE;
    #PumpControl := FALSE;
  END_IF;
  // Змішування
  IF #LevelSensorMin1 AND #LevelSensorMin2 THEN
    #MixerControl := TRUE;
  ELSE
    #MixerControl := FALSE;
  END_IF;
  // Злив готової суміші
  IF NOT #LevelSensorMin1 AND NOT #LevelSensorMin2 THEN
    #DischargeValve := TRUE;
  ELSE
    #DischargeValve := FALSE;
  END_IF;
END_IF;

```

Закінчивши написання коду функціонального блоку, необхідно скомпілювати його. Після чого можливо буде його використання в основній програмі (Main). Щоб використати функціональний блок, потрібно: створити в Main Network та перетягнути на нього з Program block у відповідне місце; дати ім'я базі даних, що створюється для цього функціонального блоку. Після цього кожному входу/виходу функціонального блоку, позначено false дати змінну, позначено %I0.0, приклад наведено на рисунку 4.6.

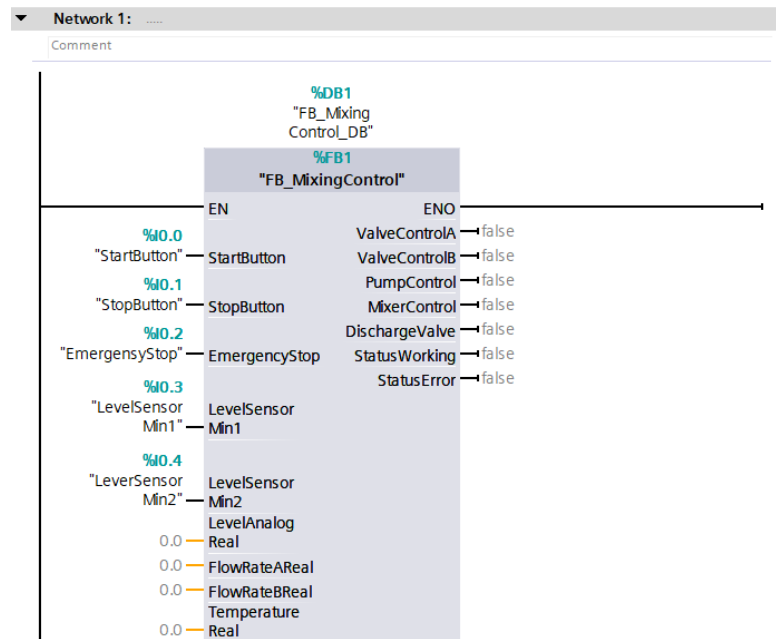


Рисунок 4.6 – Приклад використання функціонального блоку

На рисунку 4.7 зображено верхню панель. Червоним кольором обведено кнопку «Compile», зеленим кольором обведено кнопку «Download to device», синім кольором обведено кнопку «Upload from device», фіолетовим кольором обведено кнопку «Start simulation», блакитним кольором обведено кнопку «Monitoring on/off», рожевим кольором обведено кнопку «Accessible devices». Після кожної зміни в Main необхідно компілювати програму перед завантаженням в контролер.

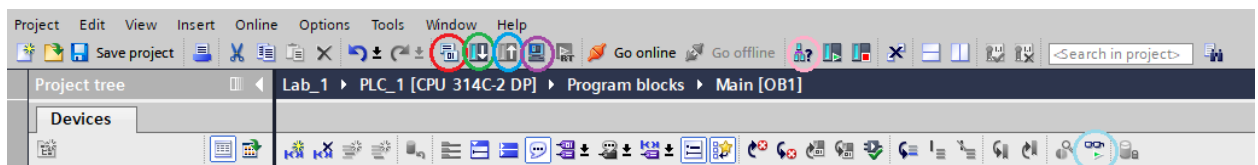


Рисунок 4.7 – Зображення верхньої панелі

Для завантаження в контролер необхідно натиснути Accessible devices (рис. 4.8), обрати тип інтерфейсу та контролер до якого бажаємо підключитися. Після чого можна натиснути «Go online» або «Download to device». При натисканні «Download to device» з'явиться вікно «Load preview»,

в якому будуть показані результати перевірок перед завантаженням. Якщо все добре, то буде активною кнопка «Load»

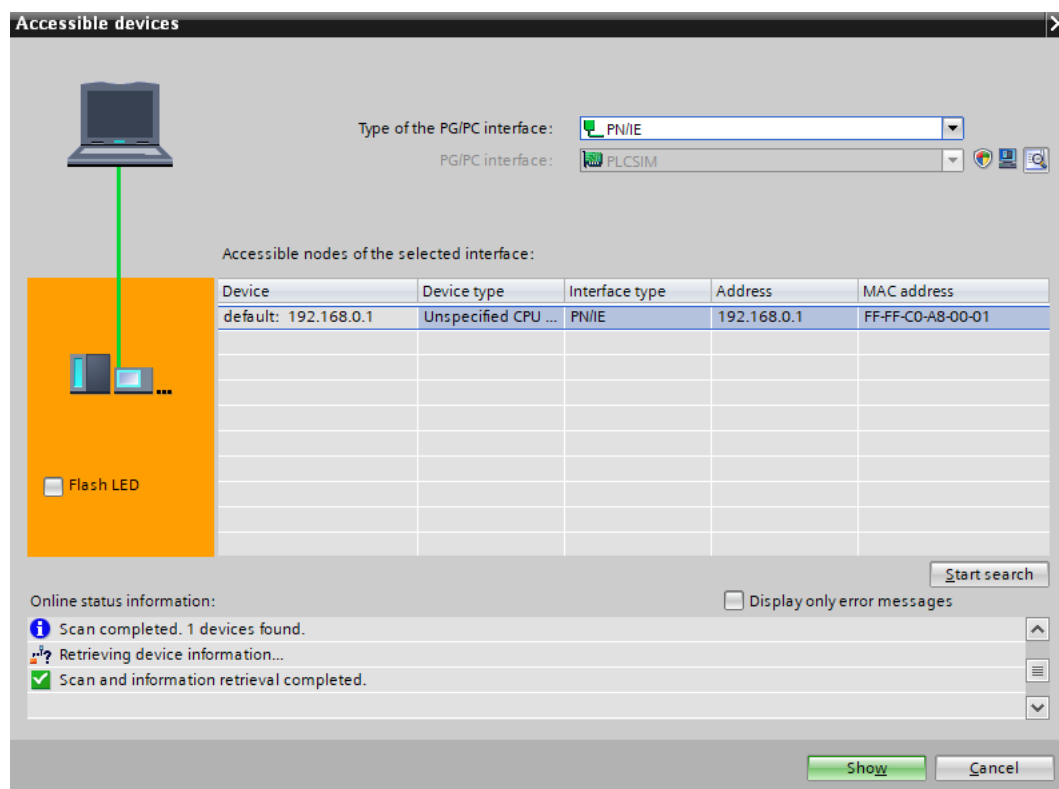


Рисунок 4.8 – Вікно Accessible devices

Після завантаження програми в контролер, можливо буде до нього підключитися та спостерігати зміну змінних з програмного забезпечення TIA Portal. За необхідності, з додатку можна змінювати значення змінних, ввімкнувши «Monitoring on/off», не перезавантажуючи програму (рис. 4.9).

При активному підключенні до контролера, якщо програма в проекті відповідає програмі, що завантажена в контролер, буде показано «ОК» навпроти кожного програмного блоку у вкладці «Program blocks».

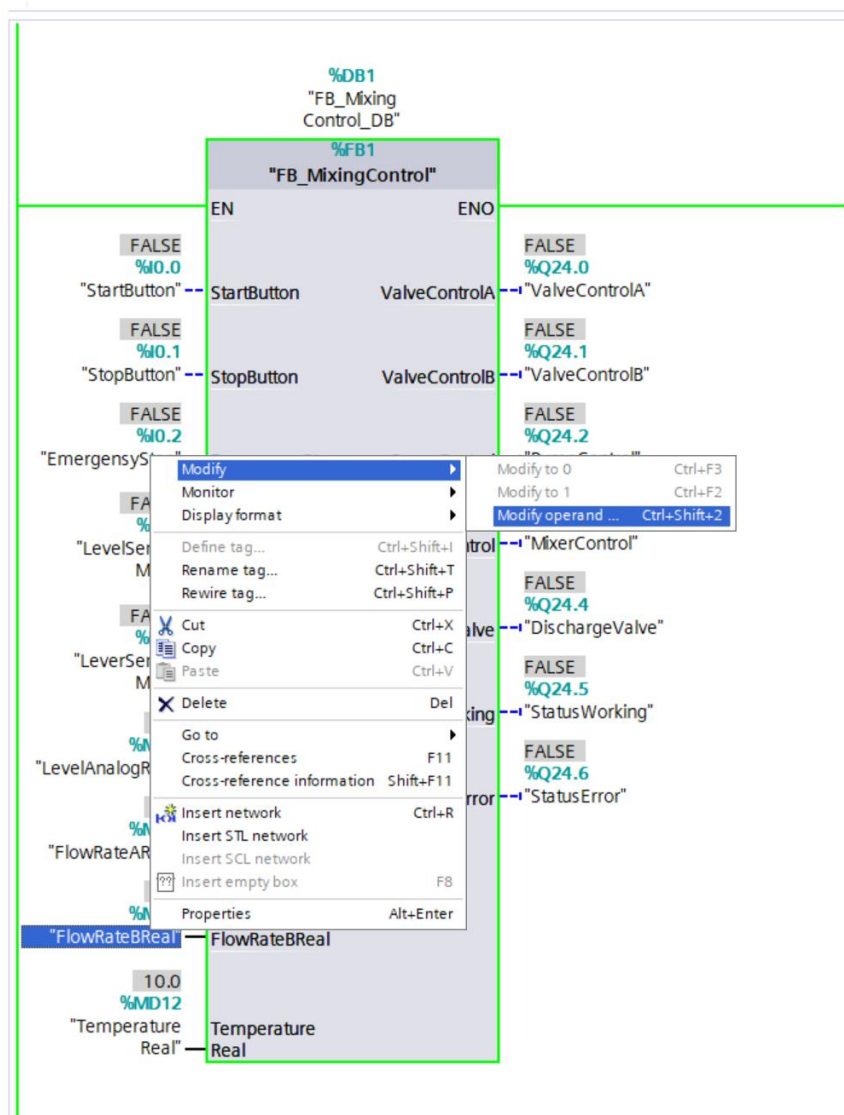


Рисунок 4.9 – Приклад зміни параметру під час онлайн підключення

4.4 Створення програми для KEPServerEX6

З серверів OPC-UA, буде використано KEPServerEX6. KEPServerEX 6 — це потужна платформа для промислової автоматизації, розроблена компанією Kerware Technologies. Вона забезпечує інтеграцію, збирання й обмін даними між різними промисловими пристроями, програмами та системами. Підтримує: понад 150 драйверів для різних пристроїв (PLC, RTU, HMI тощо); протоколи: Modbus, OPC UA, OPC DA, MQTT, BACnet, DNP3 та інші. Має інструменти для конфігурації, тестування та моніторингу в реальному часі.

Створимо підключення між KEPServerEX6 та контролером S7-300:

1. У вкладці Project – Connectivity додаємо новий канал.

2. Обираємо тип каналу: Siemens TCP/IP Ethernet.
3. Називаємо канал.
4. Обираємо мережевий адаптер, який знаходиться в одній підмережі з контролером.
5. В створеному каналі додаємо новий пристрій.
6. Записуємо IP-адресу контролеру. Всі інші параметри залишити стандартними.
7. Додаємо зміні для зчитування/запису. Приклад налаштування параметрів змінної наведено на рисунку 4.10.

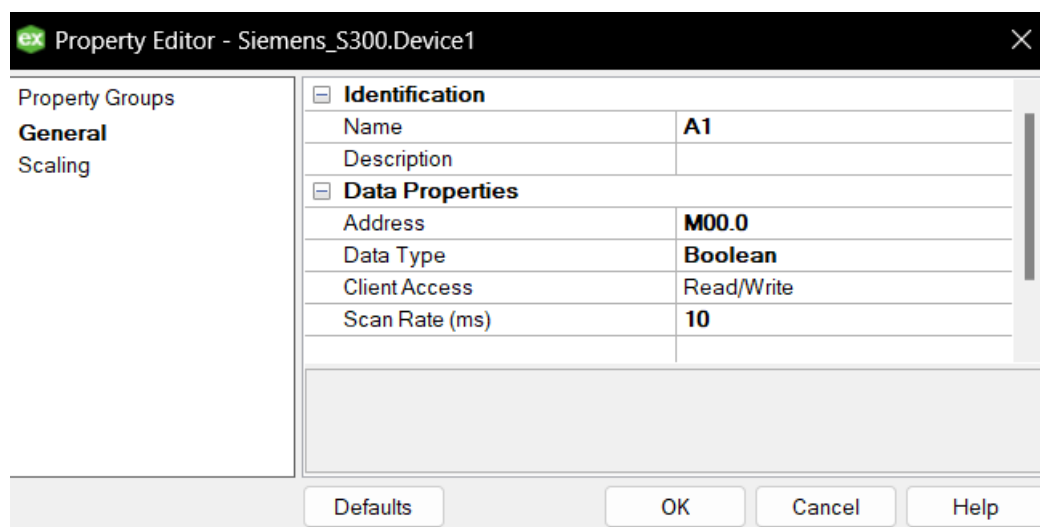


Рисунок 4.10 – Параметри налаштування для змінної A1

Створюємо підключення між KEPServerEX6 та моделю Simulink:

1. У вкладці Project – Connectivity додаємо новий канал.
2. Обираємо тип каналу: Simulator.
3. Називаємо канал.
4. Всі інші параметри залишити стандартними.
5. В створеному каналі додаємо новий пристрій. Даємо йому назву, всі інші параметри залишаємо стандартними.

Створюємо модель Simulink. Необхідними блоками є OPC Configuration, OPC Write, OPC Read. В блоці OPC Configuration натискаємо Configure OPC

Clients – Add – Select та обираємо створений канал з KEPServerEX6. Приклад успішного підключення наведено на рисунку 4.11.

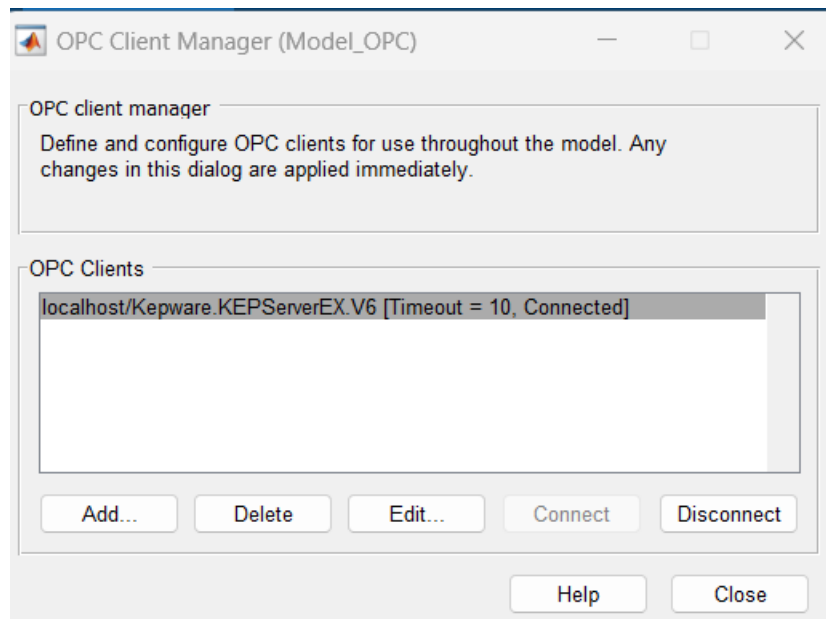


Рисунок 4.11 – Вікно OPC Client Manager

В блоках OPC Write та OPC Read натискаємо Add items, з’являється вікно Browse name space в якому знаходимо створені змінні, які хочемо зчитувати/записувати.

Для прикладу створено модель, в якій генерується синусоїдний сигнал. Створений сигнал передається в OPC сервер через блок OPC Write, після його обробки сервером, синусоїдний сигнал зчитується в моделі за допомогою блоку OPC Read. Створена схема зображена на рисунку 4.12. Результат порівняння двох сигналів зображено на рисунку 4.13.

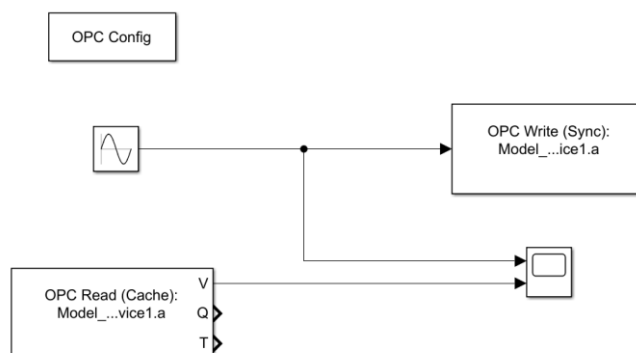


Рисунок 4.12 – Приклад моделі для запису та зчитування синусоїдного сигналу в OPC сервер

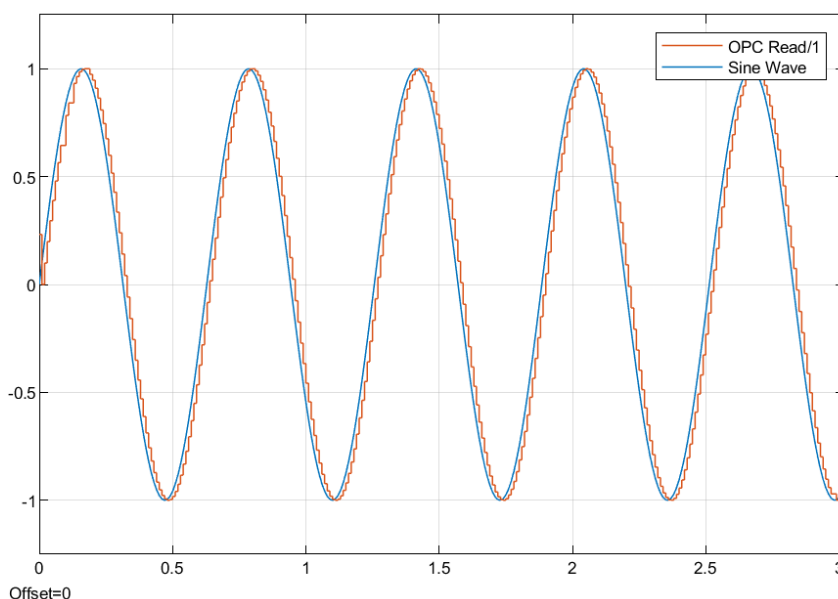


Рисунок 4.13 – Графіки згенерованого та записаного сигналу в OPC сервер та зчитаного з нього

З рис. 4.13 можна побачити що зчитаний синусоїдний сигнал є дискретизованим, на відміну від згенерованого аналогового сигналу. Крок дискретизації визначається блоком OPC Read, параметром Sample time. Однак в даному випадку для моделювання/обробки сигналів що використовуються в системах автоматизації похибка дискретизації не матиме важливої ролі.

4.5 Створення програми для S7-1200

В раніше створеному проєкті додаємо новий пристрій, а саме у Project tree натискаємо Add new device. Обираємо Controllers – SIMATIC S7-1200 – CPU – CPU 1214C DC/DC/DC – 6ES7 214-AG40-0XB0 v4.6, вказуємо назву PLC, наприклад PLC_2.

Налаштування параметрів подібне до S7-300, що описано було вище.

4.6 Розробка зв'язку між контролерами

Для налаштування зв'язку між контролерами необхідно у вкладці Network view додати зв'язок, потрібно малюнок роз'єму Ethernet PLC_1

затиснути та перетягнути на такий самий малюнок Ethernet PLC_2, тим самим створиться Profinet мережа, яка зображена на рис. 4.14.

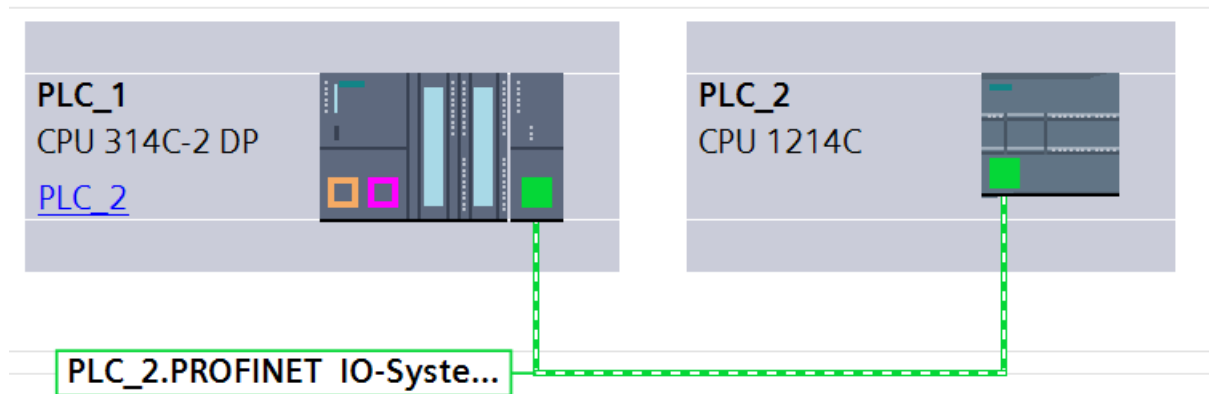


Рисунок 4.14 – Приклад створення мережі між контролерами

Обираємо модуль PLC_1 – CP 343-1 Lean, у вкладці General – PROFINET interface – Operation mode потрібно ввімкнути IO device та в випадаючому списку Assignet IO controller обрати створену мережу PLC_2.PROFINET interface_1.

В Network view натискаємо Connection – S7 connection та створюємо зв'язок між контролерами. Налаштування виконати як показано на рис. 4.15.

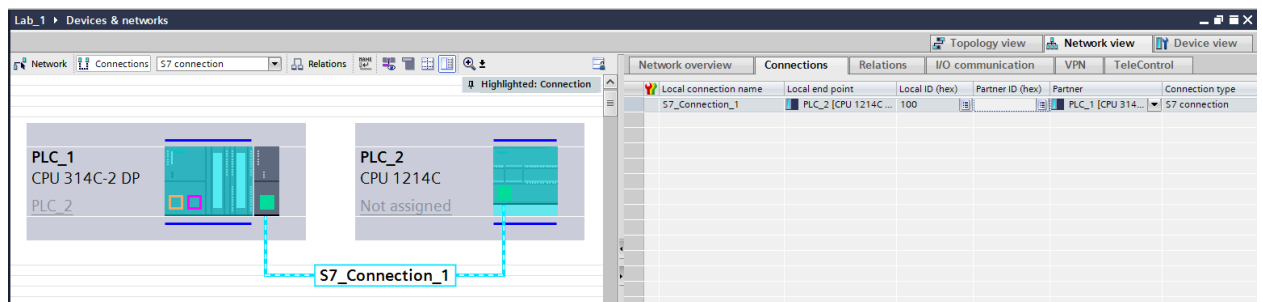


Рисунок 4.15 – Приклад налаштування зв'язку між контролерами

В програмних блоках для контролеру S7-1200 створюємо два блоки даних (DB) з масивом DATA Array [0...100] of Byte. Один блок буде використовуватися для зчитування даних, інший для передачі. Обов'язково в параметрах цих блоків даних необхідно вимкнути Attributes- Optimized block access.

Створюємо Organization block – Cyclic interrupt з Cyclic time 100 мс. В цьому блоці буде виконуватися зчитування/запис даних кожні 100 мс. В створеному блоці додаємо блоки PUT, для запису даних в віддалений контролер, та GET, для зчитування даних з віддаленого контролеру. Програма показана на рисунку 4.16.

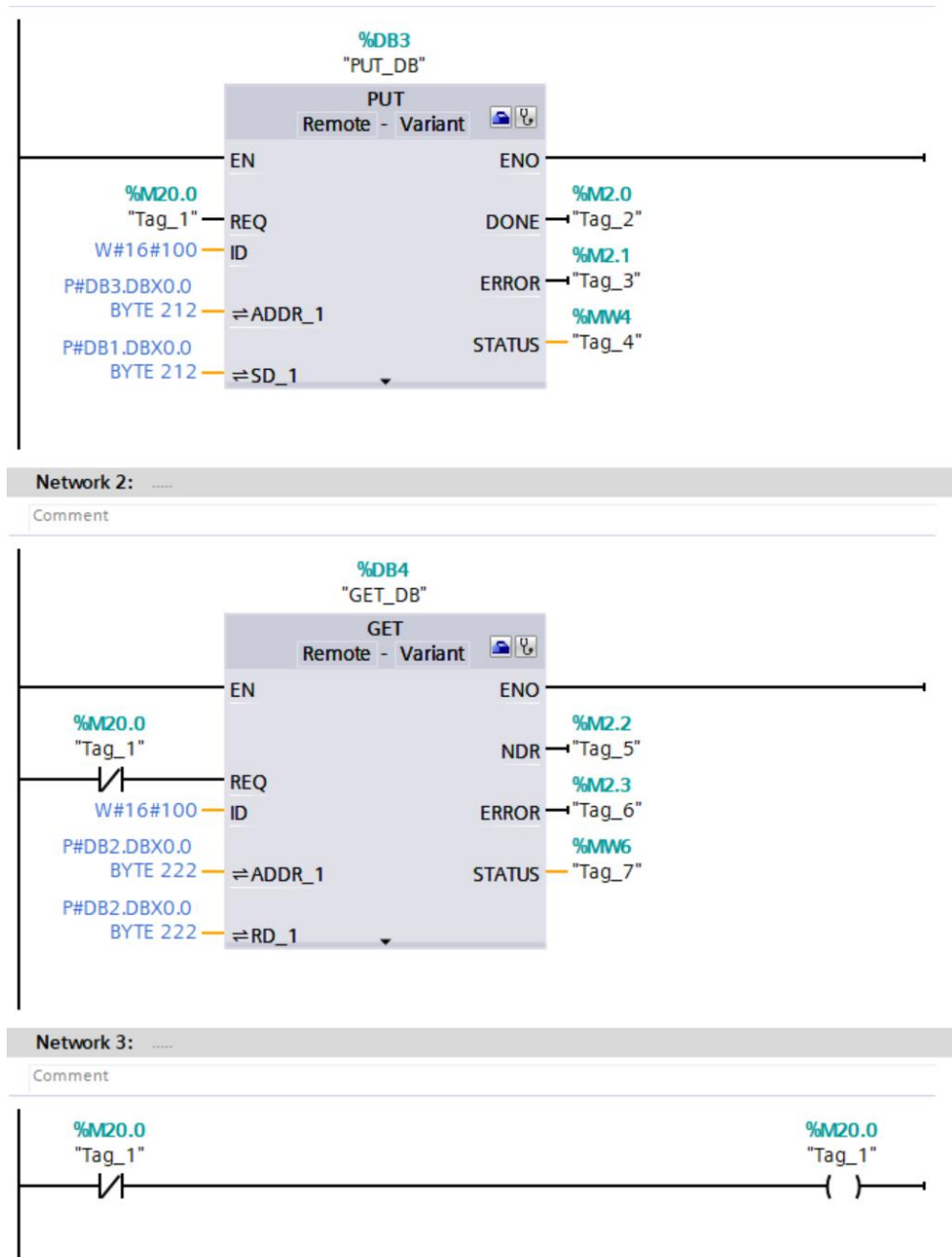


Рисунок 4.16 – Програма для запису/зчитування даних з віддаленого контролеру

Відкрити налаштування блоків можна натиснувши синю кнопку в правому верхньому куті кожного з них. Налаштування блоку PUT показано на рис. 4.17.

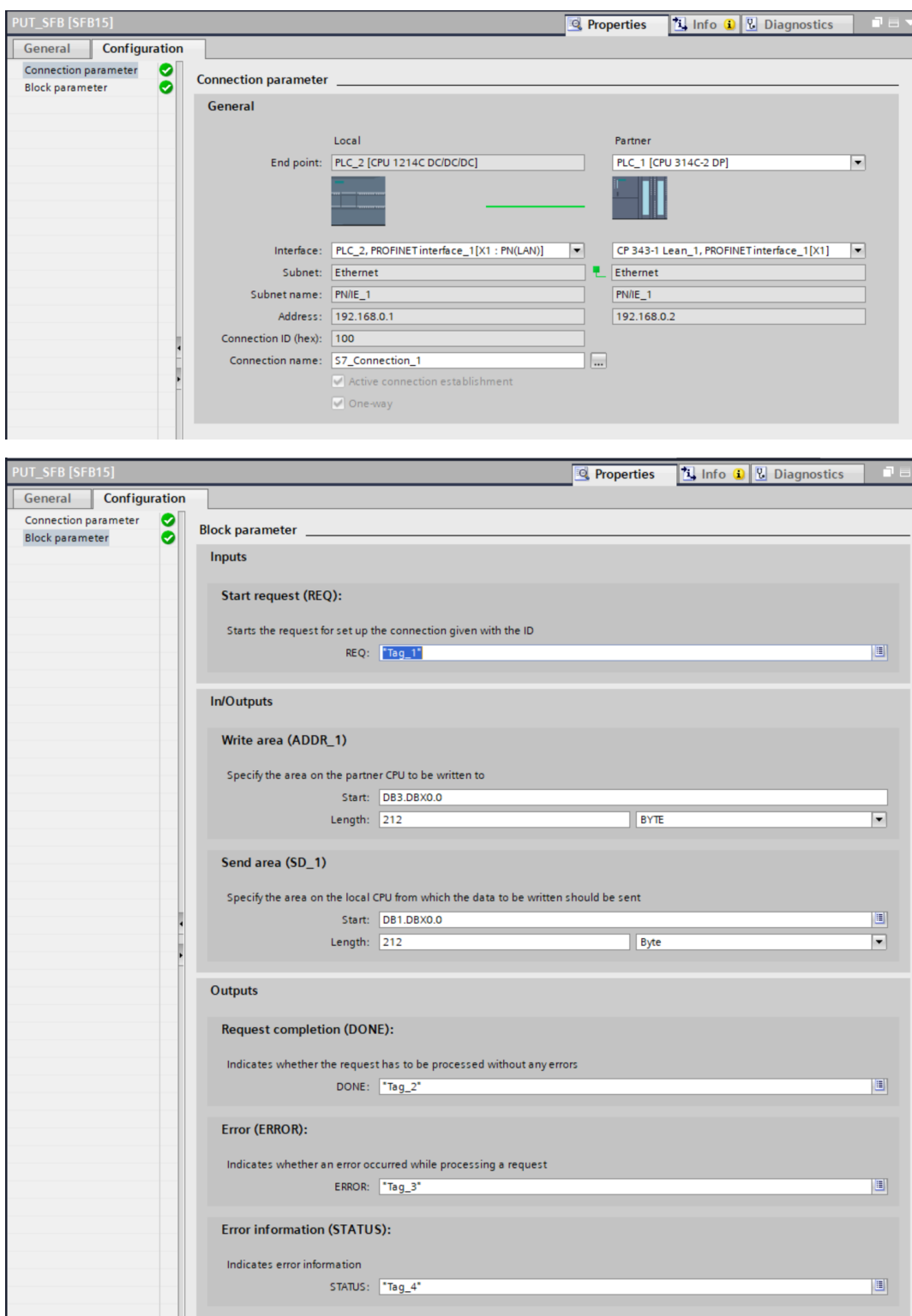


Рисунок 4.17 – Налаштування блоку PUT

В контролері S7-300 необхідно створити два блоки даних, в які будуть зчитуватися чи записуватися дані.

4.7 Розробка ПІД регулятора

Створюємо функціональний блок ПІД регулятора з наступним кодом:

```
FUNCTION_BLOCK "PID_regulator"

VAR_INPUT
    ProcessValue : Real;    // Виміряне значення (
    SetPoint : Real;        // Завдане значення
    EnablePID : Bool;       // Увімкнення ПІД-регулятора
END_VAR
VAR_OUTPUT
    ControlOutput : Real;   // Результат ПІД-регулятора
END_VAR
VAR_IN_OUT
END_VAR
VAR
    Error : Real;           // Відхилення (помилка) між SP і PV
    Integral : Real;        // Інтегральна складова
    Derivative : Real;      // Диференціальна складова
    PreviousError : Real;   // Помилка попереднього циклу
END_VAR
VAR_TEMP
END_VAR
VAR CONSTANT
    Kp : Real := 1.0;       // Пропорційний коефіцієнт
    Ki : Real := 0.1;       // Інтегральний коефіцієнт
    Kd : Real := 0.01;      // Диференційний коефіцієнт
    CycleTime : Real := 0.1; // Час циклу виконання у секундах
END_VAR

// Розрахунок відхилення
#Error := #SetPoint - #ProcessValue;

// Пропорційна складова
#ControlOutput := #Kp * #Error;

// Інтегральна складова
#Integral := #Integral + #Error * #CycleTime * #Ki;

// Диференціальна складова
#Derivative := (#Error - #PreviousError) / #CycleTime * #Kd;

// Загальний вихід PID-регулятора
#ControlOutput := #ControlOutput + #Integral + #Derivative;

// Обмеження вихідного сигналу в діапазоні [0..100]
IF #ControlOutput > 100.0 THEN
    #ControlOutput := 100.0;
ELSIF #ControlOutput < 0.0 THEN
    #ControlOutput := 0.0;
END_IF;

// Збереження помилки для наступного циклу
#PreviousError := #Error;
```

Створений блок необхідно застосовувати в організаційному блоку Cyclic interrupt, де він буде викликатися кожні 100 мс, якщо застосувати раніше створений організаційний блок для передачі даних.

4.8 Створення простої SCADA для перевірки підключення

Процес розробки SCADA в TIA Portal складається з наступних етапів:

1. Додавання нового пристрою: Add new device – PC systems – Simatic HMI application – WinCC RT Professional
2. Додавання Ethernet модуля до PC станції: hardware catalog – communication modules – PROFINET/Ethernet – IE general.
3. Додавання до мережі з контролерами, як показано в пункті 4.6.
4. Налаштування підключень SCADA з контролером та OPC сервером: Project tree – Simatic PC station – WinCC RT Professional – Connection. Додати підключення: в комунікаційному драйвері для підключення до S7-1200 обираємо відповідний драйвер та вводимо IP адресу PLC; в комунікаційному драйвері для підключення до OPC серверу обрати відповідний драйвер, вибрати створений раніше OPC сервер з локальних серверів.
5. Додавання змінних в HMI tags: додаємо нову змінну; даємо назву; обираємо тип підключення; та обираємо з випадаючого списку адресу, або прописуємо її вручну.
6. Створюємо екрани (Screen); додаємо текстові поля, кнопки, листи, бари і так далі, налаштування елементів можна знайти в мануалі [21].
7. Після закінчення створення необхідно скомпілювати файл та запустити симуляцію.

Висновки до розділу

У цьому розділі були розглянуті та реалізовані основні етапи створення лабораторного стенду для навчальних і дослідницьких потреб, зокрема:

1. Визначено загальні вимоги та завдання для студентів, які охоплюють створення програм для контролерів S7-300, S7-1200, розробку SCADA-

системи, використання OPC-сервера, а також інтеграцію та модернізацію систем із застосуванням PID-регуляторів.

2. Реалізовано проекти для контролерів Siemens Simatic S7-300 та S7-1200 у середовищі TIA Portal, включаючи налаштування апаратного та програмного забезпечення, конфігурацію мережевих з'єднань, створення функціональних блоків для керування системами, та інтеграцію PID-регуляторів.

3. Розроблено підключення між контролерами та середовищем моделювання Simulink через OPC-сервер KEPServerEX6, що забезпечує обмін даними між реальними пристроями та програмними моделями.

4. Створено базову SCADA-систему на базі WinCC RT Professional, яка дозволяє здійснювати моніторинг та управління технологічними процесами, включаючи зміну параметрів у реальному часі.

5. Реалізовано тестові моделі для перевірки роботи створених алгоритмів і контролерів, що забезпечило коректну взаємодію між компонентами системи.

У результаті виконаної роботи створено повнофункціональний лабораторний стенд, який дозволяє студентам здобувати практичні навички роботи з сучасними системами автоматизації, вивчати принципи програмування ПЛК, проєктування SCADA-систем, налаштування мережевих протоколів та використання сучасних технологій моделювання і симуляції.

5 РОЗРОБКА СИСТЕМИ ВЕКТОРНОГО КЕРУВАННЯ КУТОВОЮ ШВИДКІСТЮ АСИНХРОННОГО ДВИГУНА

5.1 Базові принципи проектування алгоритму векторного керування

Для моделювання систем керування АД використовується математична модель, яка записується в стаціонарній системі координат (a – b):

$$\begin{aligned}
 \dot{\omega} &= \frac{1}{J}(M - M_c) - v\omega \\
 M &= \mu(\psi_{2a}i_{1b} - \psi_{2b}i_{1a}) \\
 \dot{i}_{1a} &= -\gamma i_{1a} + \alpha\beta\psi_{2a} - \beta p_n \omega \psi_{2b} + \frac{1}{\sigma}u_{1a} \\
 \dot{i}_{1b} &= -\gamma i_{1b} + \alpha\beta\psi_{2b} - \beta p_n \omega \psi_{2a} + \frac{1}{\sigma}u_{1b} \\
 \dot{\psi}_{2a} &= -\alpha\psi_{2a} - p_n \omega \psi_{2b} + \alpha L_m i_{1a} \\
 \dot{\psi}_{2b} &= -\alpha\psi_{2b} - p_n \omega \psi_{2a} + \alpha L_m i_{1b}
 \end{aligned} \tag{5.1}$$

де ω – кутова швидкість ротора, $(i_{1a}, i_{1b})^T$ – компоненти струму статора і системи координат (a – b), $(\psi_{2a}, \psi_{2b})^T$ – компоненти вектора потокозчеплень ротора, $(u_{1a}, u_{1b})^T$ – компоненти вектора напруги статора, M_c – момент навантаження, v – коефіцієнт в'язкого тертя. Додатні константи, що відносяться до електричних і механічних параметрів АД, визначені в такий спосіб

$$\sigma = L_1(1 - \frac{L_m^2}{L_1 L_2}), \quad \beta = \frac{L_m}{L_2 \sigma}, \quad \mu = \frac{3}{2} p_n \frac{L_m}{J L_2}, \quad \alpha = \frac{R_2}{L_2}, \quad \gamma = \frac{R_1}{\sigma} + \alpha L_m \beta,$$

де J – повний момент інерції ротора, R_1 і R_2 – активні опори обмоток статора і ротора відповідно, L_1 і L_2 – індуктивні опори обмоток статора і ротора відповідно, L_m – індуктивний опір взаємоіндукції обмоток статора і ротора, p_n – число пар полюсів.

Алгоритми векторного керування АД, як правило, проектуються та записуються в синхронній системі координат $(d - q)$, яка обертається відносно стаціонарної системи координат з швидкістю ω_0 , зображено на рис. 5.1.

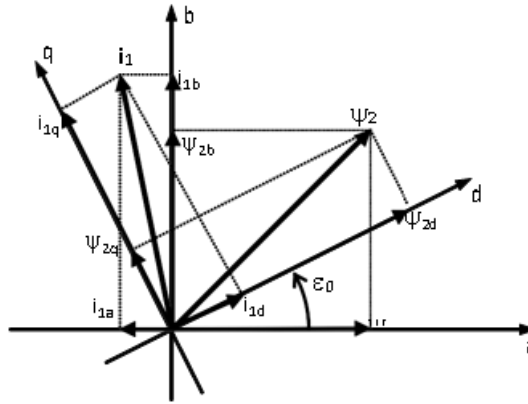


Рисунок 5.1 – Розташування векторів АД

Перетворення координат визначаються:

$$\begin{aligned} \mathbf{x}^{(d-q)} &= \mathbf{e}^{-J\varepsilon_0} \mathbf{x}^{(a-b)}; \\ \mathbf{x}^{(a-b)} &= \mathbf{e}^{J\varepsilon_0} \mathbf{x}^{(d-q)}; \end{aligned} \quad (5.2)$$

$$\mathbf{e}^{J\varepsilon_0} = \begin{bmatrix} \cos\varepsilon_0 & -\sin\varepsilon_0 \\ \sin\varepsilon_0 & \cos\varepsilon_0 \end{bmatrix}; \mathbf{e}^{-J\varepsilon_0} = \begin{bmatrix} \cos\varepsilon_0 & \sin\varepsilon_0 \\ -\sin\varepsilon_0 & \cos\varepsilon_0 \end{bmatrix}; \mathbf{J} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix},$$

де $\mathbf{x}_{yz}$ позначає двовимірний вектор напруги, потоку і струму статора, ε_0 – кутове положення системи координат $(d - q)$ відносно стаціонарної $(a - b)$.

Проблема відпрацювання заданих траєкторій кутової швидкості – потокозчеплення формується наступним чином. Припустимо, що для моделі АД, заданої (2.1), виконується наступне:

А.1. Струми статора та кутова швидкість ротора доступні для вимірювання.

А.2. Параметри АД відомі і незмінні.

А.3. Момент навантаження M_c невідомий, постійний та обмежений.

А.4. Задані траєкторії кутової швидкості ω^* і потокозчеплення $\psi^* > 0$ є обмеженими функціями з обмеженими першою та другою похідними по часу.

В умовах цих припущень відбувається проектування алгоритмів прямого на непрямого векторного керування які забезпечують

О1. Глобальне асимптотичне відпрацювання заданих траєкторій швидкості – потокозчеплення, тобто

$$\lim_{t \rightarrow \infty} \tilde{\omega} = 0, \lim_{t \rightarrow \infty} \tilde{\psi} = 0, \quad (5.3)$$

за умови обмеженості всіх внутрішніх змінних.

О2. Асимптотичне орієнтування по вектору потокозчеплення ротора

$$\lim_{t \rightarrow \infty} \psi_q = 0. \quad (5.4)$$

О3. Асимптотичну лінеаризацію підсистеми швидкості до лінійної повністю керованої форми.

О4. Асимптотичну розв'язку процесів керування електромеханічними та електромагнітними координатами АД.

5.2 Алгоритм непрямого векторного керування АД

Структура алгоритму непрямого векторного керування містить:

- розімкнутий регулятор модуля вектора потокозчеплення

$$i_{1d}^* = \frac{1}{\alpha L_m} (\alpha \psi^* + \dot{\psi}^*); \quad (5.5)$$

$$i_{1d}^* = \frac{1}{\alpha L_m} (\alpha \dot{\psi}^* + \ddot{\psi}^*); \quad (5.6)$$

$$\dot{\varepsilon}_0 = \omega_0 = \omega p_n + \alpha L_m \frac{i_{1q}^*}{\psi^*} + \phi, \quad (5.7)$$

де $\psi^*, \dot{\psi}^*, \ddot{\psi}^*$ задане значення модуля вектора потокозчеплення ротора та його перша і друга похідні відповідно, ϕ – робастифікуючий зв'язок, який буде визначено далі. В стандартному випадку $\phi = 0$.

- регулятор струму по осі d (польової складової струму статора)

$$\begin{aligned} u_{1d} &= \sigma(\gamma i_{1d}^* - \omega_0 i_{1q} - \alpha \beta \psi^* + \dot{i}_{1d} - k_i \tilde{i}_d - x_d); \\ \dot{x}_d &= k_{ii} \tilde{i}_d \end{aligned} \quad (5.8)$$

де $\tilde{i}_d = i_{1d} - i_{1d}^*$ – похибка відпрацювання струму по осі d, u_{1d} компонента вектора напруги статора по осі d, $(k_i, k_{ii}) > 0$ – коефіцієнти пропорційної та інтегральної складових регуляторів струму.

- регулятор кутової швидкості

$$\begin{aligned} i_{1q}^* &= \frac{1}{\mu \psi^*} (-k_\omega \tilde{\omega} + \hat{M}_c + \nu \omega^* + \dot{\omega}^*); \\ \hat{M}_c &= -k_{\omega i} \tilde{\omega} \end{aligned} \quad (5.9)$$

$$\dot{i}_{1q}^* = \frac{1}{\mu \psi^*} \left[-k_\omega (k_\omega \tilde{\omega} + \mu \psi^* \tilde{i}_q) + \hat{M}_c + \nu \dot{\omega}^* + \ddot{\omega}^* \right] - \frac{\dot{\psi}^*}{\psi^*} i_{1q}^*, \quad (5.10)$$

де $\omega^*, \dot{\omega}^*, \ddot{\omega}^*$ – задана кутова швидкість, перша і друга похідні відповідно, $\tilde{\omega} = \omega - \omega^*$ – похибка відпрацювання кутової швидкості, $(k_\omega, k_{\omega i}) > 0$ – коефіцієнти пропорційної та інтегральної складових регулятора кутової швидкості;

-регулятор струму по осі q (моментної складової струму статора)

$$\begin{aligned} u_{1q} &= \sigma(\gamma i_{1q}^* - \omega_0 i_{1d} - \omega p_n \beta \psi^* + \dot{i}_{1q} - k_{iq} \tilde{i}_q - x_q) \\ \dot{x}_q &= k_{ii} \tilde{i}_q \end{aligned} \quad (5.11)$$

де $\tilde{i}_q = i_{1q} - i_{1q}^*$ – похибка відпрацювання струму по осі q, u_{1q} – компонента вектора напруги статора по осі q.

Реальні напруги, що прикладаються до обмоток статора, дорівнюють:

$$\begin{pmatrix} u_a \\ u_b \end{pmatrix} = \begin{bmatrix} \cos \varepsilon_0 & -\sin \varepsilon_0 \\ \sin \varepsilon_0 & \cos \varepsilon_0 \end{bmatrix} \begin{pmatrix} u_d \\ u_q \end{pmatrix}. \quad (5.12)$$

5.3 Розрахунок параметрів номінального режиму роботи та схеми заміщення двигуна

Паспортні дані двигуна 4A250M8УЗ, при з'єднанні обмоток статора у зірку, наведені в табл. 5.1.

Таблиця 5.1 – Паспортні дані двигуна 4A250M8УЗ

Параметр	Значення
Номінальна потужність	$P_{2n} = 45 \text{ кВт}$
Номінальна лінійна напруга статора	$U_{1n} = 380 \text{ В}$
Число пар полюсів	$p_n = 4$
Момент інерції	$J_d = 1.4 \text{ кг} \cdot \text{м}^2$
Коефіцієнт корисної дії	$\eta = 0.91$
Коефіцієнт потужності	$\cos \varphi = 0.84$
Перевантажувальна здатність	$\lambda = 2$
Номінальне ковзання	$s_n = 0.014$
Критичне ковзання	$s_k = 0.09$
Номінальна частота напруги статора	$f = 50 \text{ Гц}$
Параметри Г-подібної схеми заміщення (відносні одиниці)	
Індуктивний опір розсіювання статора	$\overline{x_1}' = 0.99$
Активний опір статора	$\overline{R_1}' = 0.037$
Приведений індуктивний опір розсіювання ротора	$\overline{x_2}'' = 0.17$
Приведений активний опір ротора	$\overline{R_2}'' = 0.016$
Індуктивний опір намагнічуючого контуру	$\overline{x_\mu} = 2.6$

Кутова частота напруги статора:

$$\omega_{0n} = 2\pi f = 2 \cdot 3.14 \cdot 50 = 314 \frac{\text{рад}}{\text{с}}. \quad (5.13)$$

Швидкість ідеального холостого ходу двигуна:

$$\omega_{xx} = \frac{\omega_{0n}}{p_n} = \frac{314}{4} = 78.5 \frac{\text{рад}}{\text{с}}. \quad (5.14)$$

Номінальна швидкість двигуна:

$$\omega_n = \omega_{xx} (1 - s_n) = 314 \cdot (1 - 0.014) = 77.4 \frac{\text{рад}}{\text{с}}. \quad (5.15)$$

Номінальний момент двигуна:

$$M_n = \frac{P_{2n}}{\omega_n} = \frac{45000}{77.4} = 581.09 \text{ Н} \cdot \text{м}. \quad (5.16)$$

Критичний момент двигуна (по перевантажувальній здатності λ з каталогу):

$$M_k = \lambda M_n = 2 \cdot 581.09 = 1162.2 \text{ Н} \cdot \text{м}. \quad (5.17)$$

Номінальні діючі значення фазної напруги та струму статора:

$$U_n = U_{1n} / \sqrt{3} = 380 / \sqrt{3} = 220 \text{ В}; \quad (5.18)$$

$$I_n = \frac{P_{2n}}{3 U_n \eta \cos \varphi} = \frac{45000}{3 \cdot 220 \cdot 0.91 \cdot 0.84} = 89.4 \text{ А}. \quad (5.19)$$

Амплітудні значення фазної напруги і струму статора:

$$U_{na} = \sqrt{2} U_n = \sqrt{2} \cdot 220 = 310 \text{ В}; \quad (5.20)$$

$$I_{na} = \sqrt{2} I_n = \sqrt{2} \cdot 89.4 = 126.5 \text{ А}. \quad (5.21)$$

Амплітудне значення потокозчеплення статора в режимі холостого ходу при $R_1 = 0$:

$$\Psi_{1XX} = \frac{U_{na}}{\omega_{0n}} = \frac{310}{314} = 0.98 \text{ Вб} . \quad (5.22)$$

Математична модель АД отримана для Т-подібної схеми заміщення, що показана на рис. 5.2, в той час як приведені каталожні параметри відповідають Г-подібній схемі заміщення, рис. 5.3. Для перерахунку параметрів з Г-подібної схеми в Т-подібну використовується наступна методика:

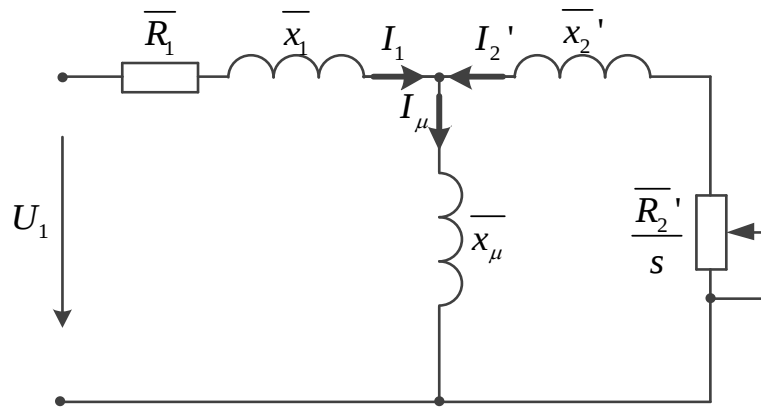


Рисунок 5.2 – Т-подібна схема заміщення асинхронного двигуна

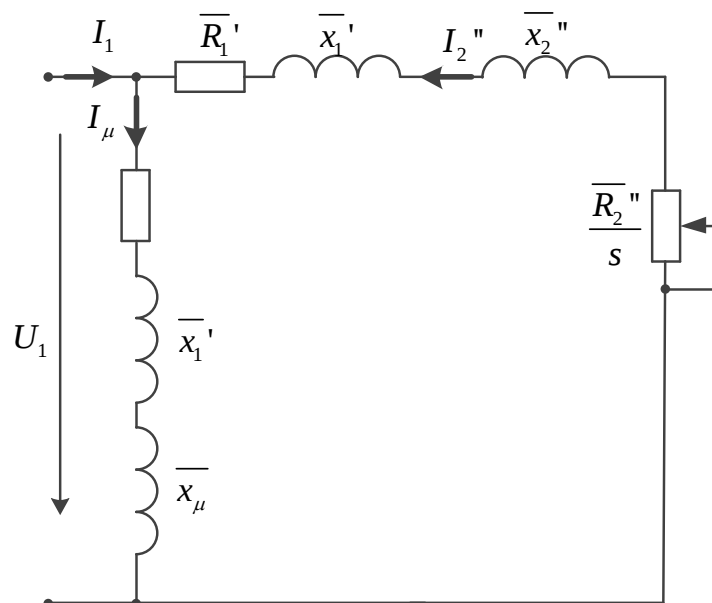


Рисунок 5.3 – Г-подібна схема заміщення асинхронного двигуна

Коефіцієнт перерахунку між Т-подібною та Г-подібною схемами заміщення:

$$c_1 = \frac{\overline{x}_\mu + \sqrt{\overline{x}_\mu^2 + 4 \cdot \overline{x}_1' \cdot \overline{x}_\mu}}{2\overline{x}_\mu} = \frac{2.6 + \sqrt{2.6^2 + 4 \cdot 0.99 \cdot 2.6}}{2 \cdot 2.6} = 1.29. \quad (5.23)$$

Параметри Т-подібної схеми заміщення у відносних одиницях:

$$\overline{x}_1 = \frac{\overline{x}_1'}{c_1} = \frac{0.99}{1.29} = 0.76; \quad (5.24)$$

$$\overline{x}_2' = \frac{\overline{x}_2''}{c^2} = \frac{0.17}{(1.29)^2} = 0.101; \quad (5.25)$$

$$\overline{R}_1 = \frac{\overline{R}_1'}{c_1} = \frac{0.037}{1.29} = 0.028; \quad (5.26)$$

$$\overline{R}_2' = \frac{\overline{R}_2''}{c^2} = \frac{0.016}{(1.29)^2} = 0.0096. \quad (5.27)$$

Параметри Т-подібної схеми заміщення в абсолютних одиницях запишуться:

$$x_1 = \overline{x}_1 \frac{U_n}{I_n} = 0.76 \cdot \frac{220}{89.4} = 1.87 \text{ Ом}; \quad (5.28)$$

$$R_1 = \overline{R}_1 \frac{U_n}{I_n} = 0.028 \cdot \frac{220}{89.4} = 0.07 \text{ Ом}; \quad (5.29)$$

$$x_2 = \overline{x}_2' \frac{U_n}{I_n} = 0.101 \cdot \frac{220}{89.4} = 0.25 \text{ Ом}; \quad (5.30)$$

$$R_2 = \overline{R}_2' \frac{U_n}{I_n} = 0.0096 \cdot \frac{220}{89.4} = 0.0234 \text{ Ом}; \quad (5.31)$$

$$x_{\mu} = \overline{x_{\mu}} \frac{U_n}{I_n} = 2.6 \cdot \frac{220}{89.4} = 6.37 \text{ Ом.} \quad (5.32)$$

Індуктивності розсіювання статора і ротора:

$$L_{1\sigma} = \frac{x_1}{\omega_0} = \frac{1.87}{314} = 0.0060 \text{ Гн}; \quad (5.33)$$

$$L_{2\sigma} = \frac{x_2'}{\omega_0} = \frac{0.25}{314} = 0.00079 \text{ Гн.} \quad (5.34)$$

Індуктивність намагнічуючого контуру:

$$L_m = \frac{x_{\mu}}{\omega_0} = \frac{6.37}{314} = 0.0203 \text{ Гн.} \quad (5.35)$$

Індуктивності статора і ротора:

$$L_1 = L_m + L_{1\sigma} = 0.0203 + 0.006 = 0.0263 \text{ Гн}; \quad (5.36)$$

$$L_2 = L_m + L_{2\sigma} = 0.0203 + 0.00079 = 0.0211 \text{ Гн.} \quad (5.37)$$

На основі проведених розрахунків обчислюємо значення параметрів $\alpha, \alpha_1, \beta, \gamma, \sigma, \mu_1$:

$$\alpha = \frac{R_2}{L_2} = \frac{0.0234}{0.0211} = 1.11 \frac{\text{Ом}}{\text{Гн}}; \quad (5.38)$$

$$\alpha_1 = \frac{R_1}{L_1} = \frac{0.0701}{0.0263} = 2.66 \frac{\text{Ом}}{\text{Гн}}; \quad (5.39)$$

$$\sigma = L_1 \left(1 - \frac{L_m^2}{L_1 L_2} \right) = 0.0263 \cdot \left(1 - \frac{0.0203^2}{0.0263 \cdot 0.0211} \right) = 0.0067 \text{ Гн}; \quad (5.40)$$

$$\beta = \frac{L_m}{L_2 \sigma} = \frac{0.0203}{0.0211 \cdot 0.0067} = 142.89 \frac{1}{\Gamma_H}; \quad (5.41)$$

$$\gamma = \frac{R_1}{\sigma} + \alpha L_m \beta = \frac{0.0701}{0.0067} + 1.11 \cdot 0.0203 \cdot 142.89 = 13.63 \frac{\text{Ом}}{\Gamma_H}; \quad (5.42)$$

$$\mu_1 = \frac{3 L_m}{2 L_2} = \frac{3}{2} \cdot \frac{0.0203}{0.0211} = 1.44. \quad (5.43)$$

Результати розрахунків параметрів двигуна 4A250M8Y3 зведені до табл.

5.2.

Таблиця 5.2 – Параметри двигуна 4A250M8Y3

$P_{2n}, \text{Вт}$	45000	$R_1, \text{Ом}$	0.07
$\omega_{xx}, \text{рад/с}$	78.5	$R_2, \text{Ом}$	0.0234
$\omega_n, \text{рад/с}$	77.4	L_1, Γ_H	0.0263
$M_n, \text{Н} \cdot \text{м}$	581.09	L_2, Γ_H	0.0211
$M_k, \text{Н} \cdot \text{м}$	1162.2	L_m, Γ_H	0.0203
λ	2	$\alpha, \text{Ом} / \Gamma_H$	1.11
$I_{na}, \text{А}$	136.5	$\alpha_1, \text{Ом} / \Gamma_H$	2.66
$\psi_{1xx}, \text{Вб}$	0.98	σ, Γ_H	0.0067
η	0.91	$\beta, 1 / \Gamma_H$	142.89
$\cos \varphi$	0.84	$\gamma, \text{Ом} / \Gamma_H$	13.63
$J_d, \text{кг} \cdot \text{м}^2$	1.4	μ_1	1.44

5.4 Дослідження динамічних характеристик в номінальному режимі

Для початку необхідно задати траєкторію потокозчеплення. Потокозчеплення досягає свого значення в 0.99 Вб за 1 с. Після цього в 1 с розпочинається завдання двигуна по швидкості, двигун розганяється за 1 с. Коли двигун працює на прямолінійній швидкості в час 2.5 с прикладається номінальне навантаження двигуна і в час 3.5 с воно прибирається. В час 4 с

двигун розпочинає гальмувати яке відбувається за 1 с. Результат моделювання роботи двигуна в номінальному режимі зображено на рис. 5.4 – 5.6.

На рис. 5.4 зображено графіки відпрацювання траєкторії швидкості, потокозчеплення та моменту двигуна.

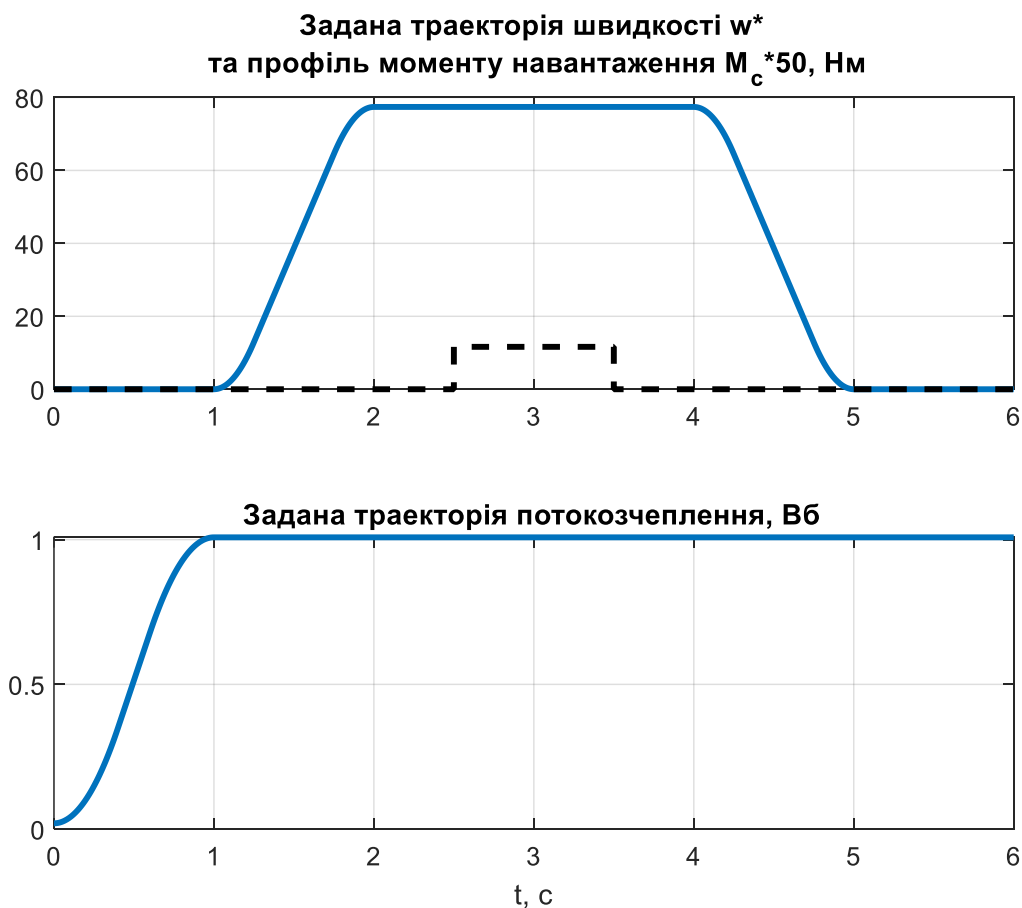


Рисунок 5.4 – Завдання швидкості, потокозчеплення та моменту двигуна

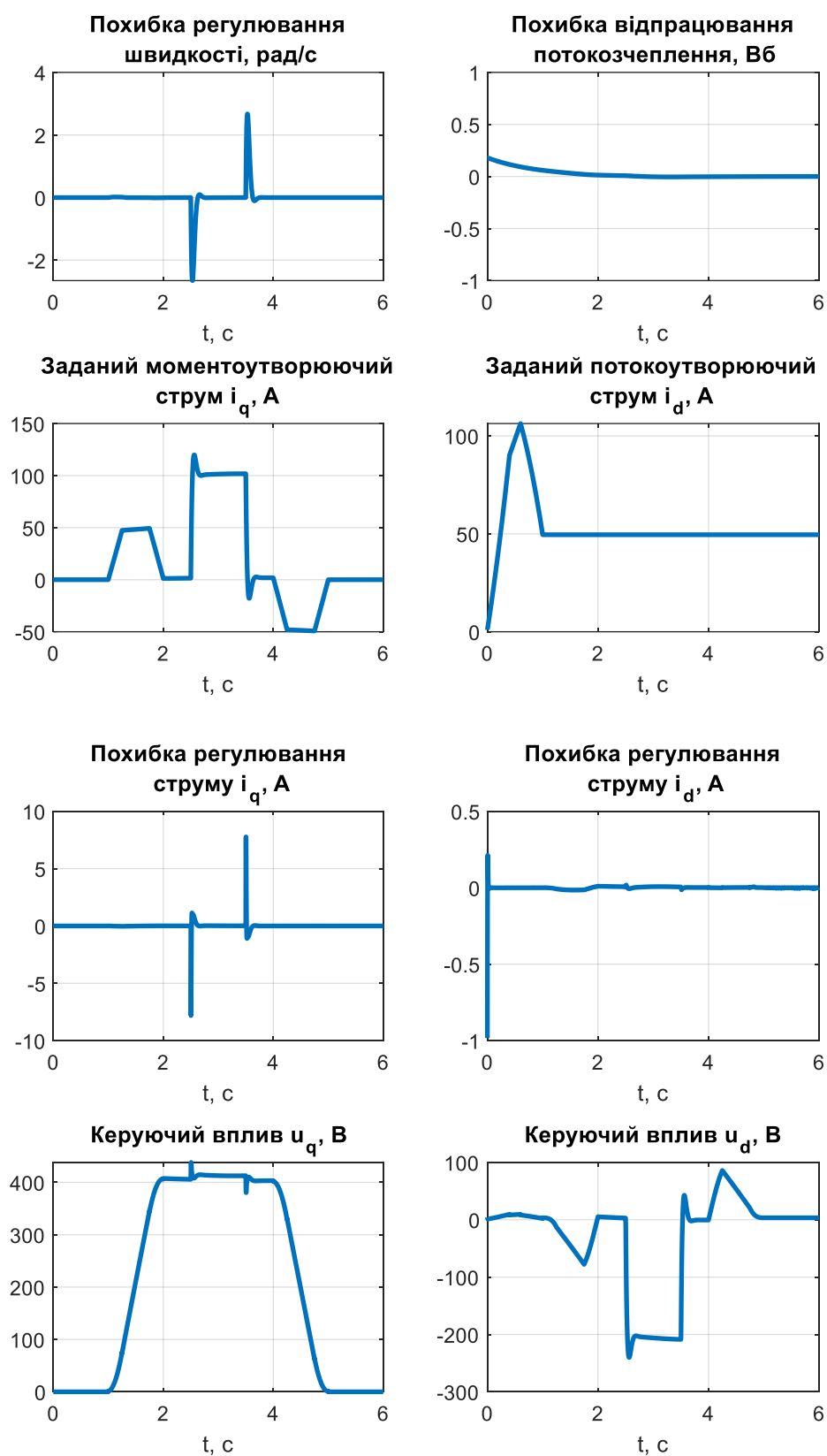


Рисунок 5.5 – Графіки перехідних процесів при непряму векторному керуванні швидкістю АД

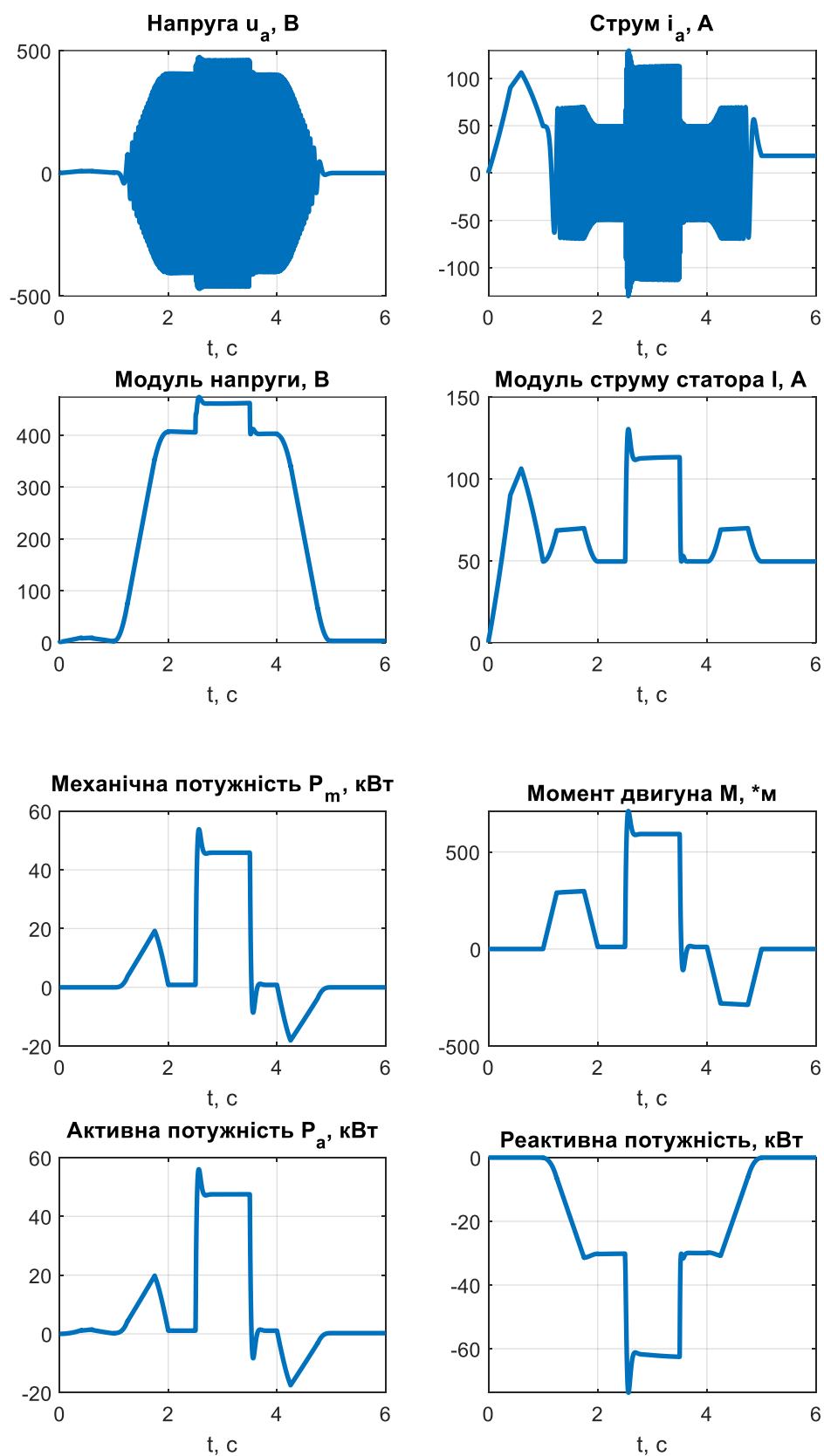


Рисунок 5.6 – Графіки перехідних процесів при непрямому векторному керуванні швидкістю АД

З рисунку 5.5, з графіків похибок швидкості можна побачити, що поставлене завдання керування швидкістю виконується, адже присутня динамічна похибка лише при накиданні навантаження. Похибка оціненого значення потокозчеплення нульова, проте вона досягається через досить тривалий час, оскільки початкове значення потокозчеплення рівне 0.2 Вб (потужний двигун). Звідси випливає, що в системі здійснюється асимптотичне відпрацювання швидкості і потокозчеплення.

З рисунку 5.5, з графіків струмів i_q^*, i_d^* видно, що струм i_d^* формує потокозчеплення двигуна, зокрема збільшення збудження, на що свідчить інтенсивне зростання струму до виходу на номінальне значення потокозчеплення. Нульова похибка струму i_d свідчить про підтримку полеорієнтування.

З рисунків 5.5 та 5.6, з графіків струмів i_q^*, i_d^* , модуля струму статора та напруг u_d, u_q видно, що при накиданні навантаження виникає зростання струму i , як наслідок, моменту двигуна. Крім цього зростає напруга u_q , що задає значення струму i_q .

Висновки до розділу

Розглянуто математичну модель асинхронного двигуна, представлену в стаціонарній та синхронній системах координат. Здійснено перетворення змінних з координат $a-b$ в $d-q$ для спрощення проектування системи керування. Розроблено алгоритм непрямого векторного керування.

Виконано перерахунок параметрів двигуна 4A250M8У3 з використанням методик між Γ -подібною та T -подібною схемами заміщення. Отримано необхідні параметри для складання моделі в Simulink.

Промодельовано двигун в Simulink та отримано динамічні характеристики при розгоні до номінальної швидкості та прикладенні номінального моменту.

6 РОЗРОБКА СТАРТАП ПРОЕКТУ

Розробка комплексного рішення для систем автоматизації викликає досить значні фінансові витрати через велику кількість механічних, електронних та софтових рішень. Створена електромеханічна система складається з ряду підсистем, які, зазвичай, розробляються окремо один від одного і їх процеси є розв'язаними в системі.

Компанії що розробляють складні системи автоматизації використовують метод прототипування. Тобто кожен підсистему створюють окремо та перевіряють в лабораторних умовах, які намагаються наблизити до реальних.

6.1 Опис ідеї

Лабораторний стенд для дослідження систем автоматизації на основі контролерів Siemens Simatic S7 призначений для навчання студентів, тестування програм і дослідження технологічних процесів. Система дозволяє користувачам моделювати, відлагоджувати та інтегрувати різноманітні процеси автоматизації з використанням сучасних промислових контролерів, таких як S7-1200 і S7-300. Стенд стане універсальним інструментом для підготовки фахівців у галузі автоматизації, мехатроніки та робототехніки. Напрямки застосування та вигоди для потенційних користувачів наведено в таблиці 6.1.

Таблиця 6.1 – Опис ідеї стартап проекту

Ціль ідеї	Напрямки використання	Вигода для користувача
Підвищення ефективності та доступності навчання в галузі автоматизації	Освітні заклади	Поліпшення якості навчання та практичних занять
	Наукові дослідження	Можливість тестування інноваційних підходів у системах автоматизації
	Промислові тренінги	Зниження витрат на навчання персонал
	Розробка прототипів	Прискорення процесу проектування та впровадження систем

Особливості ідеї:

1. Використання апаратних та програмних рішень Siemens для створення реалістичного середовища для навчання.
2. Можливість інтеграції з популярними платформами моделювання (MATLAB, Simulink, WinCC).
3. Підтримка протоколів OPC UA та PROFINET для сучасних підходів до обміну даними.

6.2 Технологічний аудит проекту

На сьогоднішній день лабораторні стенди для систем автоматизації широко використовуються у навчальних та дослідницьких процесах. Системи, що базуються на промислових контролерах, таких як Siemens Simatic S7, є стандартом у сучасній освіті й промисловості. Головними виробниками, які домінують на міжнародному ринку, є Siemens, Schneider Electric, Rockwell Automation, а також локальні виробники.

Особливості реалізації ідеї:

1. Використання готових апаратних і програмних рішень дозволяє зменшити витрати на розробку, зосереджуючи ресурси на інтеграції та налаштуванні.

2. Алгоритми управління розробляються у вигляді програмного забезпечення (прошивки) і не потребують значних капітальних затрат на модифікацію обладнання.

3. Система є масштабованою завдяки модульній структурі, що дозволяє інтегрувати додаткові функції (наприклад, тренди, звіти, аварійні сигнали).

Технологічна здійсненність проєкту представлена у таблиці 6.2.

Таблиця 6.2 – Технологічна здійсненність ідеї проєкту

№	Ідея проєкту	Технології реалізації	Наявність технологій	Доступність технологій
1	Створення лабораторного стенда для навчання	Siemens Simatic S7-1200/300, WinCC, PROFINET	Наявно	Повністю доступно
2	Інтеграція з симуляційними платформами	MATLAB, Simulink, OPC-UA	Наявно	Доступно за ліцензією
3	Реалізація аварійних сигналів та трендів	WinCC RT Professional	Наявно	Обмежено доступно через вартість
4	Підтримка віддаленого моніторингу	OPC-сервери, веб-доступ	Наявно	Доступно за наявності додаткових налаштувань

З таблиці 6.2 можна зробити висновок, що технічна реалізація даного проєкту є обмежено можливою.

6.3 Аналіз ринкових можливостей запуску стартап-проєкту

Потенційними клієнтами для впровадження стартапу на ринок є навчальні заклади, дослідницькі установи, а також промислові компанії, які використовують автоматизовані системи для тренінгів та тестування.

Аналіз можливих клієнтів, що представлені на ринку України, наведено в таблиці 6.3.

Таблиця 6.3 – Потенційні клієнти на ринку України

№	Виробник	Особливості
1	Siemens	Міжнародна компанія з широким спектром рішень для автоматизації
2	Schneider Electric	Французька багатонаціональна корпорація, спеціалізується на автоматизації
3	Phoenix Contact	Глобальний постачальник рішень для автоматизації та електротехніки

З аналізу представлених клієнтів можна зробити висновок: теоретично більший прибуток може бути отриманий у разі співпраці з міжнародними компаніями, але для даного стартапу доцільно зосередитись на локальному ринку України. Це пояснюється простішим доступом до ключових клієнтів, відділів розробок та спрощенням логістики.

На основі визначення потенційних клієнтів виконуємо аналіз ринкового середовища. Для цього складаємо таблиці факторів, що сприяють ринковому впровадженню проекту, та факторів, що йому перешкоджають.

Таблиця 6.4 – Фактори загроз проекту

Фактор	Зміст загрози	Можлива реакція компанії
Конкуренція	Наявність на ринку великих компаній	Пропонування унікальних функцій та адаптація під клієнта
Вартість	Збільшення вартості продукту через використання високоякісного обладнання	Пошук шляхів зниження вартості без втрати якості

Таблиця 6.5 – Фактори можливостей проєкту

Фактор	Зміст можливостей	Можлива реакція компаній
Зростання попиту	Потреба в сучасних навчальних стендах для автоматизації	Розширення модельного ряду та надання сервісного обслуговування
Інноваційність	Використання новітніх технологій в навчальних процесах	Розробка додаткових модулів для інтеграції з іншими платформами

6.4 Аналіз факторів конкурентоспроможності

На основі аналізу ринку, враховуючи унікальні характеристики стартап-проєкту, а також основні вимоги споживачів до навчальних лабораторних стендів, було визначено ключові фактори конкурентоспроможності. Результати представлено в таблиці 6.6.

Таблиця 6.6 – Обґрунтування факторів конкурентоспроможності

№	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проєктів значущими)
1	Модульність конструкції	Можливість додавання нових модулів та адаптація до різних навчальних програм та потреб замовників
2	Інтеграція з популярними платформами	Підтримка MATLAB, Simulink, OPC UA дозволяє використовувати стенд для широкого спектра завдань
3	Висока якість обладнання	Використання промислових компонентів Siemens забезпечує довговічність та надійність системи
4	Інтерактивність та віддалений доступ	Наявність веб-інтерфейсу та можливість дистанційного моніторингу відкривають нові способи навчання

Виконаємо порівняльний аналіз слабких та сильних сторін проєкту. Результати наведено в таблиці 6.7.

Таблиця 6.7 – Альтернативи ринкового впровадження стартап-проекту

№	Фактор конкурентоспроможності	Бали (1-20)	Рейтинг товарів-конкурентів у порівнянні з запропонованим (балів)
1	Модульність конструкції	18	15
2	Інтеграція з популярними платформами	16	12
3	Висока якість обладнання	15	14
4	Інтерактивність та віддалений доступ	14	10

6.5 Розроблення ринкової стратегії проекту

Створимо базову стратегію розвитку. Результати зображено в таблиці 6.8.

Таблиця 6.8 – Визначення базової стратегії розвитку

№	Альтернатива	Ймовірність отримання ресурсів	Термін реалізації
1	Розробка власних стендів, які за замовчуванням будуть використовуватись у навчальних програмах	70%	1 рік
2	Співпраця з виробниками обладнання для створення адаптивних рішень	90%	6 місяців

Визначаємо стратегію конкурентної поведінки. Результати наведено в таблиці 6.9.

Таблиця 6.9 – Визначення базової стратегії конкурентної поведінки

№	Чи є ідея першим ходом на ринку?	Чи буде компанія шукати нових споживачів або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Шукати нових клієнтів	Ні, соільки аналоги є комерційною таємницею	Стратегія зайняття конкурентної ніші

Розробимо стратегію позиціювання, яка полягає в формуванні ринкової позиції, за яким споживачі мають ідентифікувати цей стартап-проект. Результати показано в таблиці 6.10.

Таблиця 6.10 – Визначення стратегії позиціювання

№	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкуретноспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувану комплексну позицію власного проекту (три ключових)
1	Стенди повинні забезпечувати простоту у використанні та високу гнучкість	Стратегія спеціалізації	Інтеграція з популярними платформами, низька вартість та швидкість налаштування	Інтеграція з MATLAB, адаптивність до навчальних програм, підтримка трендів у SCADA

6.6 Розроблення маркетингової програми стартап-проекту

Першим етапом є формування маркетингової концепції товару, який отримає споживач. Для цього підсумовуємо результати попереднього аналізу конкурентоспроможності товару. Результати представлено в таблиці 5.11.

Таблиця 6.11 – Визначення ключових переваг концепції потенційного товару

№	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
1	Широкий діапазон ідеї	Простота інтеграції стендів у навчальні програми та промислові процеси	Інтеграція з сучасними платформами, такими як MATLAB, Simulink, OPC UA
2	Менші економічні витрати	Дешевизна реалізації та простота в освоєнні	Зниження витрат на навчання персоналу та швидкість впровадження

Трирівнева маркетингова модель товару наведена в таблиці 6.12.

Таблиця 6.12 – Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
1. Товар за задумом	Алгоритм тестування та моделювання з використання контролерів Siemens
	Властивості/характеристики
	1. Інтеграція з MATLAB/Simulink
	2. Простота навчання
2. Товар у реальному виконанні	Включає фізичний стенд, ПО, та інструкції для користувачів.
	Відсутність складного пакування.
3. Товар із підкріпленням	Включає технічну підтримку, онлайн-інструкції, а також подальші оновлення алгоритмів чи модулів.

Оскільки розроблений алгоритм представляє собою виключно інтелектуальну власність, його капіталізація проходитиме на основі

ліцензування та передачі прав. Відповідно до Закону України про авторське право, можливі наступні стратегії:

1. Передача прав на використання алгоритму за фіксовану вартість.
2. Передача прав на використання за відсоток від продажу обладнання (роялті).
3. Надання ліцензій на обмежений період із регулярним поновленням контрактів.

Ці стратегії дозволяють забезпечити постійний дохід та захист інтересів компанії на ринку, сприяючи довгостроковій співпраці з партнерами та клієнтами.

Висновки до розділу

В даному розділі представлено основні кроки щодо впровадження розробленого лабораторного стенду для дослідження систем автоматизації на ринок. Отримані наступні результати:

1. Визначені основні вигоди для користувача, а саме – спрощення процесу навчання, тестування та дослідження автоматизованих систем за рахунок інтеграції сучасних технологій та зниження витрат на впровадження обладнання.
2. Проведено аналіз ринку, в результаті якого було визначено ключові переваги впровадження лабораторного стенду для навчальних закладів, наукових установ та промислових компаній.
3. Розглянуто технологічні шляхи реалізації проєкту. На основі аналізу ринку України визначено основну цільову аудиторію – навчальні заклади, що спеціалізуються на автоматизації, а також інженерні центри, що займаються розробкою промислових рішень.
4. Для кожної групи клієнтів обрано оптимальні стратегії капіталізації. Вибір стратегії капіталізації базується на фінансових можливостях та потребах обраної цільової аудиторії.

Так, у разі співпраці з великими навчальними установами чи промисловими підприємствами, доцільно використовувати ліцензійну модель із регулярними виплатами за користування обладнанням. У випадку співпраці з малими компаніями чи стартапами вигідніше обирати разові продажі з можливістю технічної підтримки.

Крім того, слід враховувати динаміку зростання компаній-клієнтів. Наприклад, у разі стагнації великої компанії та стрімкого розвитку малої доцільно змінити стратегію з ліцензійної моделі на передачу прав за фіксовану плату. Це дозволить адаптуватися до ринкових умов та забезпечити стабільний дохід від реалізації проєкту.

ВИСНОВКИ

У магістерській дисертації розроблено лабораторний стенд для дослідження систем автоматизації на основі контролерів Siemens Simatic S7. Проведені дослідження дозволили досягти наступних результатів:

1. Розроблено концепцію лабораторного стенду для дослідження можливостей автоматизації на базі сучасних контролерів Siemens Simatic серій S7-300 та S7-1200. Було проведено огляд та порівняння технічних характеристик різних моделей ПЛК для обґрунтування вибору обладнання.

2. Створено електричну принципову схему стенду із застосуванням модулів розширення CP 343-1 Lean, 331-7KF02-0AB0, 332-5HB01-0AB0. Розроблено систему живлення на базі блоку Siemens SITOP PSU8200, що забезпечує стабільну роботу стенду.

3. Реалізовано програмування контролерів для виконання прикладних задач автоматизації у програмному середовищі TIA Portal. Створено варіанти програм для різних сценаріїв роботи лабораторного стенду, включаючи взаємодію з SCADA-системами.

4. Розроблено методологічні вказівки та завдання для студентів для ознайомлення з основними функціями ПЛК, організацією мережевих з'єднань (PROFINET, TCP/IP), а також налаштуванням зв'язку між різними контролерами.

5. Проведено тестування та моделювання роботи стенду у різних режимах. Розроблено рекомендації щодо використання стенду в навчальних цілях для вивчення основ програмування ПЛК, діагностики систем та організації автоматизації процесів.

6. Розроблено алгоритм непрямого векторного керування АД з короткозамкненим ротором. Розраховано параметри двигуна під Т-подібну схему. Проведено дослідження динамічних характеристик при номінальній швидкості та номінальному навантаженні.

7. Запропоновано старт-ап проект для інтеграції розробленого стенду у навчальні заклади та промислові підприємства, де є необхідність навчання та підготовки спеціалістів в галузі автоматизації. Визначено економічні переваги впровадження стенду та перспективи його подальшого вдосконалення.

Розроблений стенд забезпечує навчальні та практичні потреби, дозволяючи глибше ознайомитися з сучасними системами автоматизації, їхньою архітектурою та функціональними можливостями.

ПЕРЕЛІК ПОСИЛАНЬ

1. A Comparison between Microcontrollers and PLC. URL: <https://www.vemeko.com/blog/a-comparison-between-microcontrollers-and-plc.html>.
2. Microcontrollers Versus PLC's: A Detailed Comparison. URL: <https://www.c3controls.com/white-paper/microcontrollers-versus-plcs-detailed-comparison/>.
3. Seamless Connectivity: Unlocking the Power of PLC and MES Integration. URL: <https://praxie.com/plc-and-mes-integration/>.
4. PLC Hardware Components. URL: <https://www.plctable.com/plc-hardware-components/>.
5. Siemens. URL: <https://www.siemens.com/>.
6. Schneider Electric. URL: <https://www.se.com/ua/uk/>.
7. Allen-Bradley. URL: <https://www.rockwellautomation.com/>.
8. Mitsubishi. URL: <https://www.mitsubishielectric.com/fa/products/cnt/plc/>.
9. Delta. URL: <https://www.deltaww.com/>.
10. Siemens Simatic s7-300 6ES7314-6CG03-0AB0. URL: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6ES7314-6CG03-0AB0>
11. CP343-1 Lean. URL: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Products/10021181?ActiveTab=2>.
12. 6ES7331-7KF02-0AB0. URL: <https://mall.industry.siemens.com/mall/Catalog/Product/6ES7331-7KF02-0AB0>.
13. 6ES7332-5HB01-0AB0. URL: <https://mall.industry.siemens.com/mall/Catalog/Product/6ES7332-5HB01-0AB0>.
14. 6ES7214-1AG40-0XB0. URL: <https://mall.industry.siemens.com/mall/Catalog/Product/6ES7214-1AG40-0XB0>.
15. 3RV2011-1FA10. URL: <https://mall.industry.siemens.com/mall/Catalog/Product/3RV2011-1FA10>.

16. 6EP3436-8SB00-0AY0. URL: <https://mall.industry.siemens.com/mall/en/WW/Catalog/Product/6EP3436-8SB00-0AY0>.

17. ETI MER2-024DC. URL: <https://www.eti.ua/produksiya-ua/industrial-electromagnetic-relays-and-miniature-elctromagnetic-releys/002473032-mer2-miniature-electromagnetic-relays-24v-dc-8a>

18. AD16-16DS. URL: https://www.acko.ua/e-store/xml_catalog/svitlosignalna_ta_zvukova_armatura_serii_ad16_22/20030/.

19. Резистор. URL: <https://electronica.in.ua/ua/p1530394641-rezistor-82k-025w.html?>

20. Змінний резистор. URL: <https://somi.com.ua/ua/p1632030343-peremennyj-rezistor-potentsiometr.html?>

21. Посібник з HMI Siemens. URL: <https://www.gothightech.com/resources/pdf/Manual%20-%20WinCC%20v7.5%20Working%20with%20WinCC.pdf>