Національний університет кораблебудування
імені адмірала Макарова

Навчально-науковий інститут автоматики та електротехніки
(повна назва інституту/факультету)

Кафедра автоматики
(повна назва кафедри)

«До захисту допущено»
Завідувач кафедри

_____ І. С. Білюк
(підпис)

«16» 06 2025 р.

КЛАСИФІКАЦІЙНА РОБОТА

на здобуття ступеня бакалавра

зі спеціальності 141 Електроенергетика, електротехніка та електромеханіка
(код та назва спеціальності)

освітня програма Електромеханіка та електроенергетика
(назва освітньої програми)

на тему: Проєктування логічного контролера на базі мікроконтролера Microchip

Виконав: студент IV курсу, групи 4317ст3
Тимчук Ілля Сергійович _____
(прізвище, ім'я, по батькові) (підпис)

Керівник асистент каф. автоматики, к.т.н.,Козлов А.Ю. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Нормоконтроль доц. каф. автоматики, к.т.н., Васильєв О.Г. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент доц. каф. СЕЕС, к.т.н., доц., Новогрецький С.М. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Засвідчую, що у цій кваліфікаційній роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____
(підпис)

м. Миколаїв – 2025 р.

Національний університет кораблебудування імені адмірала Макарова

Інститут, факультет Навчально-науковий інститут автоматики та електротехніки
Кафедра автоматики
Освітній ступінь бакалавр
Галузь знань 14 «Електрична інженерія»
(шифр і назва)
Спеціальність 141 «Електроенергетика, електротехніка та електромеханіка»
(шифр і назва)
Освітня програма «Електротехніка та електротехнології»

ЗАТВЕРДЖУЮ
Завідувач кафедри, к.т.н. доцент
_____ І.С. Білюк
«7» 02 2025 року

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

гр.4317ст3 Тимчуку Іллі Сергійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи «Проєктування логічного контролера на базі мікроконтролера Microchip»
керівник роботи Козлов Андрій Юрійович, PhD.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від «23» квітня 2025 року № УЧ-343
2. Строк подання студентом роботи 13 червня 2025 р.
3. Вихідні дані до роботи літературні джерела за тематикою мікроконтролерів, характеристики мікроконтролерів, реалізація вихідних сигналів, обробка дискретних сигналів, програмні продукти автоматичного керування, методики створення алгоритмів програм.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1 Компоненти програмованих логічних контролерів, 2. Вибір мікроконтролеру для проєктування логічного контролера, 3. Програмне забезпечення логічного контролера, 4. Охорона праці

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
1 Мікроконтролер PIC16F877A. 2 Силові ключі для керування релейними виходами та реалізація зв'язку за протоколом RS485. 3 Принципова електрична схема програмованого логічного контролера на базі мікроконтролера PIC16F877A. 4. Функціональна схема проєктованого логічного контролера на базі мікроконтролера PIC16F877A. 5 Алгоритми керуючої програми. 6 Алгоритми підпрограм. 7 Результат роботи програми для контролю та діагностики системи по RS485

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Васильєв О.Г., к.т.н., доцент		

7. Дата видачі завдання 7 лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Програмовані логічні контролери та їх компоненти	22.04.25	
2	Вибір мікроконтролера	05.05.25	
3	Вибір силових ключів та реалізація зв'язку за протоколом RS485	07.05.25	
4	Розробка алгоритмів програми та написання		
	коду програми	05.06.25	
5	Охорона праці	09.06.25	

Студент _____ Тимчук І.С.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Козлов А.Ю.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

У сучасних умовах глобальної цифровізації промислових процесів зростає потреба у створенні гнучких, надійних та економічно ефективних систем автоматизації. Програмовані логічні контролери (ПЛК) відіграють ключову роль у таких системах, забезпечуючи швидке реагування на зміну технологічних параметрів, гнучке налаштування алгоритмів керування та можливість інтеграції з інформаційними системами підприємств.

Використання доступних апаратних рішень, зокрема мікроконтролерів сімейства PIC, дозволяє розробляти компактні та недорогі логічні контролери для широкого спектра застосувань. Одночасно впровадження інтерфейсу RS485 забезпечує надійний обмін даними між пристроями у промислових умовах, що критично важливо для сучасних розподілених систем автоматизації.

Таким чином, розроблення логічного контролера на базі мікроконтролера PIC16F877A з підтримкою RS485 є актуальним завданням, спрямованим на задоволення потреб сучасного ринку промислової автоматизації.

Мета роботи - розроблення програмованого логічного контролера на базі мікроконтролера PIC16F877A з підтримкою обміну даними по інтерфейсу RS485 для використання у системах промислової автоматизації.

Об’єкт дослідження - програмований логічний контролер на базі мікроконтролера PIC16F877A.

Предмет дослідження - методи та засоби проєктування апаратної та програмної частини логічного контролера для систем промислової автоматизації, організація обміну даними по інтерфейсу RS485.

Ключові слова: - програмований логічний контролер, ПЛК, PIC16F877A, мікроконтролер, RS485, MAX485, ULN2803, релейні виходи, інтерфейс зв’язку, індикація, LCD1602, алгоритм керування, автоматизована система.

ABSTRACT

In today's global digitalisation of industrial processes, there is a growing need for flexible, reliable and cost-effective automation systems. Programmable logic controllers (PLCs) play a key role in such systems, providing fast response to changes in process parameters, flexible configuration of control algorithms and the ability to integrate with enterprise information systems.

The use of affordable hardware solutions, such as PIC family microcontrollers, allows for the development of compact and inexpensive logic controllers for a wide range of applications. At the same time, the introduction of the RS485 interface ensures reliable data exchange between devices in industrial environments, which is critical for modern distributed automation systems.

Thus, the development of a logic controller based on the PIC16F877A microcontroller with RS485 support is an urgent task aimed at meeting the needs of the modern industrial automation market.

ЗМІСТ

ВСТУП 7
РОЗДІЛ 1 ПРОГРАМОВАНІ ЛОГІЧНІ КОНТРОЛЕРИ ТА ЇХ КОМПОНЕНТИ 9
1.1 Види програмованих логічних контролерів (ПЛК) та їх призначення 9
1.2 Входи та виходи ПЛК: типи, функціональне призначення, принципи підключення 20
Висновок за розділом 1 27
РОЗДІЛ 2 ВИБІР ТА ОБГРУНТУВАННЯ МІКРОКОНТРОЛЕРА ДЛЯ ПЛК 29
2.1 Вибір мікроконтролера 29
2.2 Характеристики мікроконтролера PIC16F877A 32
2.3 Вибір силових ключів для керування релейними виходами 38
2.4 Реалізація зв’язку за протоколом RS485. 43
2.5 Опис спроєктованої схеми 47
Висновки за розділом 2 52
РОЗДІЛ 3 ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЛОГІЧНОГО КОНТРОЛЕРА 53
3.1 Розробка функціональної схеми та алгоритму роботи контролера 53
3.2 Алгоритми роботи контролера 59
3.3 Програмна реалізація алгоритму логічного контролера 72
Висновок за розділом 3 82
РОЗДІЛ 4 ОХОРОНА ПРАЦІ 84
4.1. Аналіз шкідливих та небезпечних факторів при роботі з програмованим логічним контролером 84
4.2. Заходи з охорони праці та електробезпеки при експлуатації автоматизованої системи 87
Висновок за розділом 4 90
ВИСНОВОК 92
Список використаних джерел 94
ДОДАТОК А МУЛЬТИМЕДІЙНА ПРЕЗЕНТАЦІЯ 96

ВСТУП

Швидкий розвиток технологій автоматизації у промисловості, транспорті, енергетиці та інших галузях обумовлює постійне зростання потреби у використанні програмованих логічних контролерів (ПЛК). Ці пристрої забезпечують гнучке та надійне керування технологічними процесами, значно підвищуючи ефективність виробництва та якість продукції. Завдяки модульному підходу та широким можливостям конфігурації, ПЛК стали невід'ємною частиною сучасних автоматизованих систем керування.

Особливу актуальність має розроблення недорогих ПЛК на базі мікроконтролерів, що дозволяє створювати компактні та ефективні рішення для автоматизації як дрібних, так і середніх об'єктів. Одним із таких мікроконтролерів є PIC16F877A виробництва компанії Microchip, що поєднує у собі достатню обчислювальну потужність, наявність апаратної підтримки інтерфейсів зв'язку та розвинену периферію.

Важливою складовою сучасних систем автоматизації є забезпечення віддаленого обміну даними між контролерами та системами верхнього рівня. Для цього широко застосовується інтерфейс RS485, який забезпечує надійний обмін інформацією навіть у складних промислових умовах та дозволяє організовувати мережі з великою кількістю пристроїв.

Розроблення логічного контролера з використанням мікроконтролера PIC16F877A та підтримкою інтерфейсу RS485 дозволяє отримати універсальний інструмент для побудови гнучких систем автоматизації. У дипломній роботі виконано аналіз існуючих рішень, спроєктовано апаратну частину контролера, розроблено програмне забезпечення та здійснено тестування працездатності системи. Також приділено увагу питанням забезпечення охорони праці та безпеки при експлуатації автоматизованих систем.

Результати роботи демонструють практичну можливість створення ефективного логічного контролера на базі доступних апаратних рішень та

відкривають перспективи для його подальшого застосування у системах промислової автоматизації.

РОЗДІЛ 1

ПРОГРАМОВАНІ ЛОГІЧНІ КОНТРОЛЕРИ ТА ЇХ КОМПОНЕНТИ

1.1 Види програмованих логічних контролерів (ПЛК) та їх призначення

Програмовані логічні контролери, або ПЛК, є одним з найважливіших компонентів сучасних автоматизованих систем управління. Вони застосовуються у найрізноманітніших сферах промисловості для реалізації алгоритмів логічного керування технологічними процесами. Основною особливістю ПЛК є можливість виконання складних логічних та арифметичних операцій, керування виконавчими механізмами, обробки сигналів з різних типів датчиків, а також інтеграція з іншими інформаційними системами, зокрема SCADA та MES. Завдяки гнучкості програмування, високій надійності та адаптованості до роботи у жорстких умовах навколишнього середовища, ПЛК стали невід'ємною частиною цифрової трансформації виробництва в межах концепцій Індустрії 4.0 та Інтернету речей.

Історія створення та розвитку ПЛК почалася у 1960-х роках, коли зростаюча складність технологічних ліній на заводах, особливо в автомобільній промисловості, зумовила потребу у переході від релейно-контакторних систем до більш гнучких і зручних цифрових рішень. Перший ПЛК був розроблений для General Motors компанією Modicon, і з того часу ПЛК постійно розвиваються, набуваючи дедалі більшої функціональності. Сучасні ПЛК є високотехнологічними пристроями з мікропроцесорною архітектурою, які працюють у режимі реального часу, можуть містити інтегровані дисплеї, модулі зв'язку, спеціалізовані інтерфейси, а також підтримують різні мови програмування згідно зі стандартом IEC 61131-3 [1].

ПЛК класифікують за різними ознаками, зокрема за конструктивним виконанням, функціональними можливостями, масштабом застосування,

типом процесора, підтримуваними мережевими протоколами та інтерфейсами, типом програмного забезпечення та способом програмування. Найбільш поширеним критерієм є конструктивне виконання, відповідно до якого ПЛК поділяються на компактні (моноблочні), модульні та розподілені. Кожен з цих типів має свої особливості, переваги, недоліки і сфери застосування, що робить їх доречними у відповідних контекстах.

Компактні або моноблочні ПЛК (рис. 1.1) є найпростішим варіантом контролерів, у яких усі функціональні елементи — центральний процесор, модулі введення/виведення, блок живлення — об’єднані в одному корпусі. Такі ПЛК мають фіксовану кількість входів і виходів, не підлягають розширенню і зазвичай застосовуються для реалізації простих логічних задач, наприклад у системах освітлення, вентиляції, управління насосами чи обігрівачами. Їхньою головною перевагою є низька вартість, простота налаштування та експлуатації, а також компактність. До недоліків можна віднести обмежену гнучкість конфігурації та обмеження за кількістю сигналів, які вони можуть обробляти.

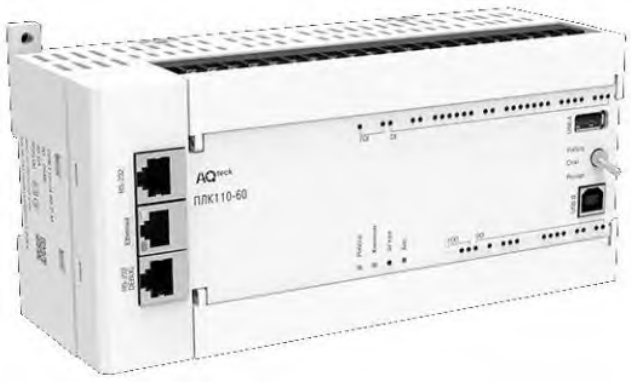


Рис. 1.1 – Моноблочний ПЛК

Модульні ПЛК (рис. 1.2), на відміну від компактних, побудовані за принципом модульності, що дозволяє гнучко формувати систему управління відповідно до конкретних вимог технологічного процесу. До складу таких ПЛК входять окремі модулі — процесорний, комунікаційний, модулі

введення/виведення, спеціалізовані модулі (наприклад, для аналогових сигналів або термопар), які монтуються на загальну шину або в спеціальне шасі. Це забезпечує високу адаптивність, можливість масштабування та зручність в обслуговуванні. Модульні ПЛК здатні працювати у великих системах керування з численними точками збору даних та складною логікою керування. Завдяки підтримці різноманітних протоколів зв’язку, таких як Modbus, Profibus, Ethernet/IP, Profinet, такі системи легко інтегруються в корпоративні інформаційні мережі, що дозволяє здійснювати моніторинг, збір статистики, діагностику та прогнозування у режимі реального часу. Їх активно застосовують у машинобудуванні, хімічній промисловості, енергетиці, виробництві напоїв, а також у великих логістичних комплексах і транспортних системах.

Рис. 1.2 – Модульний ПЛК

Розподілені ПЛК-системи є складнішими і реалізують концепцію децентралізованого управління. Вони складаються з кількох локальних контролерів або модулів введення/виведення, які встановлюються безпосередньо біля контрольованих об’єктів і з’єднуються між собою за

допомогою мережевих протоколів. Центральний ПЛК у такій системі може координувати роботу всіх підлеглих вузлів або ж функції управління можуть бути розподілені між контролерами за принципом горизонтального обміну. Цей підхід дає змогу скоротити довжину кабельних трас, зменшити кількість точок відмов, підвищити надійність системи і забезпечити збереження часткової працездатності навіть у разі збою окремого контролера. Розподілені системи широко використовуються у великих інфраструктурних об’єктах, таких як аеропорти, метрополітени, водоочисні споруди, електростанції, а також у складних технологічних лініях, де централізоване управління є складним або небажаним.

Окрім конструктивної класифікації, важливим критерієм для визначення типу ПЛК є їх функціональна складність. Залежно від того, які завдання повинен виконувати контролер, розрізняють базові, середнього рівня та високопродуктивні ПЛК. Базові моделі здатні виконувати лише прості логічні операції, працювати з дискретними сигналами та обслуговувати обмежену кількість точок вводу/виводу. Вони програмуються переважно за допомогою релейно-контактної мови програмування Ladder Diagram, яка є інтуїтивно зрозумілою навіть для інженерів без глибоких знань з ІТ. Середнього рівня ПЛК, крім логічних операцій, можуть виконувати базову арифметику, обробляти аналогові сигнали, підтримувати PID-регулювання, вести облік подій, зберігати дані у вбудованій пам’яті, а також обмінюватися інформацією з іншими пристроями по мережах. Високопродуктивні ПЛК призначені для виконання складних завдань з високими вимогами до швидкості обробки даних, точності, надійності та функціонального навантаження. Вони підтримують багатозадачність, логіку у реальному часі, інтеграцію з базами даних, обробку сигналів з високою частотою дискретизації, мають великі об’єми пам’яті та потужні процесори.

Поряд із конструктивною та функціональною класифікацією, не менш важливою є класифікація ПЛК за сферою їх застосування. Це дозволяє краще зрозуміти, як саме обираються ті чи інші типи контролерів у відповідності до

потреб конкретної галузі. У промисловості машинобудування, наприклад, ПЛК застосовуються для автоматизації верстатів, ліній складання, роботизованих маніпуляторів, транспортерів та іншого обладнання. Тут переважають модульні та розширені ПЛК, які забезпечують не лише високий рівень контролю, але й інтеграцію з системами візуалізації, управління приводами, вимірюванням точного положення та швидкості. У харчовій промисловості ПЛК виконують функції контролю температури, рівня, тиску, вагових дозувань, забезпечення стерильності, а також слідкують за станом пакувальних машин, етикетувальних станцій та сортувальних пристроїв. Тут важлива наявність можливостей для точного PID-регулювання, підтримка швидких аналогових сигналів, а також сумісність з вимогами гігієнічних стандартів.

В енергетичній сфері ПЛК використовуються для моніторингу та керування електричними мережами, трансформаторними підстанціями, генеруючими установками, вітровими і сонячними електростанціями. ПЛК у таких системах не тільки керують перемиканням електричних ліній і виконавчих пристроїв, а й виконують функції захисту, діагностики та передають дані до диспетчерських центрів. Вони повинні забезпечувати високу стійкість до збурень у мережі, можливість аварійного резервування та підтримку протоколів, що дозволяють інтегруватися у системи SCADA або диспетчерського управління. У транспортній сфері ПЛК відіграють ключову роль в управлінні сигналами світлофорів, системами автоматичних шлагбаумів, управлінням залізничним рухом, диспетчеризації електротранспорту, а також у системах контролю доступу на вокзалах або станціях метро. В таких випадках перевагу надають контролерам із високою надійністю, підтримкою резервування, широким температурним діапазоном роботи і здатністю до швидкої обробки сигналів.

У галузі будівельної інфраструктури ПЛК використовуються для реалізації так званого «розумного будинку» або «розумної будівлі», де вони забезпечують автоматичне керування освітленням, вентиляцією, опаленням,

кондиціонуванням повітря, охоронними та протипожежними системами, а також ліфтами. Тут контролери повинні бути сумісними з протоколами BACnet, KNX, DALI, мати зручний інтерфейс програмування і можливість віддаленого доступу через інтернет або локальні мережі. У водопровідно-каналізаційних господарствах ПЛК керують насосними станціями, станціями очищення води, резервуарами, гідрантами, забезпечують контроль рівня води, тиску, якості рідини, виконують діагностику систем і можуть автоматично регулювати подачу води в залежності від попиту. Таким чином, функціональність ПЛК у різних галузях суттєво відрізняється, але в основі залишається принцип керування на основі обробки вхідних сигналів за попередньо встановленим алгоритмом.

Ще одним важливим напрямком є використання ПЛК у системах реального часу, зокрема в системах точного позиціонування, обробки сигналів високої частоти, керування електроприводами та синхронізованими механізмами. Такі контролери, як правило, мають дуже потужні процесори, розвинену архітектуру з апаратною підтримкою багатозадачності, вбудовані модулі для обробки імпульсних сигналів, лічильників, енкодерів та інтерфейсів передачі даних з низькою затримкою. Їх застосування є критично важливим у виробництві мікроелектроніки, фармацевтиці, авіації, автомобілебудуванні, де навіть мілісекунди або мікросекунди можуть мати значення для точності та якості кінцевого продукту.

Особливу увагу останнім часом привертають так звані гібридні контролери, що поєднують у собі функції ПЛК, системи HMI (людино-машинного інтерфейсу), SCADA-модуля, а також шлюзів до промислових інтернет-мереж. Такі рішення дозволяють мінімізувати кількість обладнання на об’єкті, скоротити витрати на монтаж і обслуговування, спростити структуру керування. Часто вони поставляються з інтегрованими сенсорними панелями, потужними комунікаційними модулями, а також можуть працювати з хмарними сервісами, наприклад Amazon Web Services, Microsoft Azure, Siemens MindSphere. Завдяки цьому розширюються можливості щодо збору

даних, предиктивного аналізу, оптимізації енергоспоживання та дистанційного технічного обслуговування.

Тенденції останніх років свідчать про те, що розвиток ПЛК іде в напрямку збільшення швидкодії, енергоефективності, гнучкості конфігурації та взаємодії з хмарними платформами. Зростає популярність open-source рішень, таких як CODESYS, що дозволяють створювати крос-платформні програми, переносити їх між різними моделями контролерів, інтегрувати з системами вищого рівня. Також розвивається підтримка стандартів кібербезпеки, зокрема IEC 62443, оскільки з підключенням ПЛК до інтернету зростають ризики кіберзагроз. Більшість сучасних контролерів мають вбудовані засоби автентифікації, шифрування даних, журналювання подій і механізми віддаленої блокади.

Ще один важливий аспект — це програмне забезпечення, яке використовується для конфігурування та програмування ПЛК. Воно має вирішальне значення, оскільки від нього залежить зручність роботи інженера, час розробки алгоритмів, можливість інтеграції з іншими системами та подальшого обслуговування. У більшості випадків застосовується програмування відповідно до стандарту IEC 61131-3, який визначає п’ять мов: релейну схему (Ladder Diagram, LD), функціональні блоки (Function Block Diagram, FBD), структурований текст (Structured Text, ST), інструкційний список (Instruction List, IL) та послідовну функціональну схему (Sequential Function Chart, SFC). Кожна з цих мов має свої переваги в залежності від складності завдання та звичок інженера-програміста. Наприклад, LD добре підходить для простих логічних схем, SFC — для багатостадійних процесів, ST — для складних математичних обчислень.

Варто окремо зупинитися на впливі конструктивних і програмних особливостей ПЛК на їх експлуатацію у промисловому середовищі. Конструктивне виконання визначає рівень захисту корпусу, методи монтажу, тип охолодження, а також сумісність із розподільчими шафами або панелями керування. Наприклад, компактні ПЛК часто мають ступінь захисту IP20 або

IP40 та монтуються на DIN-рейку, тоді як розподілені контролери, які працюють безпосередньо на об’єктах, можуть мати ступінь захисту IP65 і вище, що дозволяє їм працювати на відкритому повітрі або у вологому середовищі без додаткових захисних кожухів. Деякі моделі мають пасивне охолодження, що забезпечує безшумну роботу, інші ж оснащені вентиляторами або навіть рідинним охолодженням у разі високих теплових навантажень. Високопродуктивні ПЛК іноді мають можливість гарячої заміни модулів (hot swap), що дозволяє проводити технічне обслуговування без зупинки технологічного процесу, а також модулі дублювання, які автоматично беруть на себе керування в разі відмови основного модуля.

Що стосується програмного забезпечення, то воно також має великий вплив на гнучкість системи автоматизації. Універсальне програмне середовище дозволяє зменшити час навчання персоналу, стандартизувати підходи до розробки, тестування та обслуговування системи. Наприклад, Siemens пропонує інженерну платформу TIA Portal, яка дозволяє працювати одночасно з ПЛК, панелями оператора, приводами та системами розподілу енергії. Аналогічно, Rockwell Automation реалізує платформу Studio 5000, яка поєднує конфігурування, програмування, моделювання та моніторинг. Багато сучасних систем підтримують функціональні блоки, які користувач може створювати самостійно і повторно використовувати в інших проектах. Це значно зменшує обсяг коду, прискорює відлагодження та підвищує надійність роботи системи.

Ще однією особливістю сучасних ПЛК є підтримка функціональної безпеки відповідно до стандартів SIL (Safety Integrity Level), зокрема IEC 61508 або ISO 13849. У цьому випадку контролери працюють у режимі підвищеної надійності, обробляючи сигнали від аварійних датчиків, кнопок екстреної зупинки, контролерів положення дверей або бар'єрів безпеки. Такі контролери мають подвійні процесори або модулі самодіагностики, що дозволяє виявляти внутрішні несправності і запобігати небезпечним станам системи. Впровадження систем функціональної безпеки є обов’язковим у

таких галузях, як хімічне виробництво, металургія, виробництво пального, атомна енергетика, а також у верстатах з високим ступенем потенційної небезпеки для персоналу.

Значну роль у функціонуванні ПЛК відіграють комунікаційні інтерфейси, які дозволяють здійснювати обмін даними між контролерами, датчиками, модулями введення/виведення, панелями оператора, системами SCADA та базами даних. До найпоширеніших фізичних інтерфейсів належать RS-232, RS-485, Ethernet, CAN, а також бездротові інтерфейси Wi-Fi, Bluetooth або GSM/3G/4G. Багато контролерів підтримують одночасно декілька інтерфейсів, що дозволяє інтегрувати їх у гетерогенні мережі без використання додаткових шлюзів. Поряд з фізичними інтерфейсами існують також протоколи обміну даними, серед яких виділяються Modbus RTU, Modbus TCP, Profibus DP, Profinet, Ethernet/IP, DeviceNet, OPC UA. Кожен з них має свої переваги і сферу застосування, наприклад, Modbus є простим і широко підтримується різними пристроями, тоді як OPC UA забезпечує високий рівень безпеки і масштабованості в мережах IIoT. Саме завдяки розвинутій системі комунікацій ПЛК можуть виступати не лише як виконавчі пристрої, а й як вузли збору, агрегації та аналітики даних.

Таким чином, сучасні ПЛК є багатофункціональними пристроями, які значно виходять за межі класичного уявлення про простий логічний контролер. Вони перетворилися на повноцінні обчислювальні платформи, здатні вирішувати широкий спектр завдань: від збору даних до аналітики, від простих логічних схем до адаптивного регулювання і взаємодії з хмарними сервісами. Вибір конкретного типу ПЛК залежить від багатьох факторів: складності технологічного процесу, кількості та типу сигналів, потреб у масштабованості, рівня безпеки, наявності мережевої інфраструктури, вимог до швидкодії та бюджету. У будь-якому випадку важливо розглядати ПЛК не просто як пристрій, а як частину комплексної системи автоматизації, яка повинна працювати ефективно, надійно, зручно для користувача та з можливістю подальшого розвитку.

Узагальнюючи вищезазначене, варто підкреслити, що вибір типу ПЛК є стратегічним кроком у процесі створення системи автоматизованого керування. Помилкове рішення на цьому етапі може зумовити серйозні наслідки: як технічні, так і економічні. Наприклад, використання надто простого контролера у складному процесі призведе до обмежень у функціональності, втрати стабільності роботи, а в гіршому випадку — до повної зупинки технологічного процесу. З іншого боку, придбання надто дорогого і складного обладнання для простої задачі спричиняє перевитрати, нераціональне використання ресурсів та ускладнення обслуговування. Тому важливо проводити детальний техніко-економічний аналіз перед прийняттям рішення про тип, модель і конфігурацію ПЛК [15].

Крім того, варто враховувати ще один фактор — кадровий. Успішна експлуатація ПЛК неможлива без належної кваліфікації персоналу, здатного розробляти, налаштовувати, обслуговувати й модернізувати програмне забезпечення. Незважаючи на те, що сучасні системи намагаються спростити процес програмування за допомогою графічних інтерфейсів, шаблонів і автоматичних генераторів коду, глибоке розуміння логіки процесу та принципів побудови керуючих систем залишається критично важливим. Підготовка інженерів-автоматиків має включати не лише знання програмування ПЛК, а й розуміння електротехніки, гідравліки, пневматики, сенсорики, комп’ютерних мереж, кібербезпеки та системного інжинірингу.

Певні перспективи має також розвиток стандартизації у сфері ПЛК. Сьогодні існує велика кількість виробників, які використовують власні апаратні платформи, протоколи, середовища програмування, що іноді створює труднощі при масштабуванні або інтеграції різнотипних систем. Наприклад, ПЛК Siemens працюють у середовищі TIA Portal, тоді як Rockwell Automation — у Studio 5000. Відсутність єдиної платформи часто потребує залучення фахівців з вузькою спеціалізацією, що обмежує мобільність персоналу та ускладнює експлуатацію змішаних систем. Саме тому рух до відкритих стандартів — таких як IEC 61131-3, OPC UA, MQTT, AutomationML — є

одним із головних векторів розвитку галузі, адже дозволяє створювати інтероперабельні рішення, які не залежать від конкретного постачальника [2].

Не менш важливою тенденцією є екологічність та енергоефективність. Сучасні ПЛК дедалі частіше мають вбудовані засоби моніторингу енергоспоживання, що дозволяє будувати системи енергоменеджменту. Наприклад, контролер може автоматично оптимізувати режим роботи насосів або кондиціонерів, знижувати частоту обертання електродвигунів у непікові години, вмикати або вимикати обладнання відповідно до розкладу або завантаженості. Такі функції не лише знижують витрати, а й сприяють зменшенню викидів CO_2, що відповідає сучасним екологічним стандартам.

У контексті цифровізації економіки та переходу до інтелектуального виробництва, роль ПЛК лише зростатиме. Вони залишатимуться основною ланкою між фізичним і цифровим світом, забезпечуючи надійний обмін даними між обладнанням і вищими рівнями управління. Їх використання дозволяє в реальному часі адаптувати процеси до змін середовища, потреб ринку, стану обладнання, забезпечуючи високу гнучкість і ефективність. Із розвитком цифрових двійників, предиктивної аналітики, хмарного керування та штучного інтелекту ПЛК еволюціонують у напрямку комплексних систем, які не лише керують, а й «думають», аналізують, приймають рішення.

Таким чином, програмовані логічні контролери є не лише технічними пристроями, а й ключовими елементами сучасної автоматизованої інфраструктури. Їх класифікація за конструктивними, функціональними, галузевими та програмними ознаками дозволяє забезпечити точне відповідність технічним вимогам і стратегічним цілям кожного окремого проекту. Завдяки постійному розвитку, підтримці нових технологій, інтеграції з інформаційними системами і відповідності глобальним стандартам ПЛК продовжують залишатися на передовій лінії інженерної еволюції в XXI столітті.

1.2 Входи та виходи ПЛК: типи, функціональне призначення, принципи підключення

У структурі будь-якого програмованого логічного контролера ключову роль відіграють входи та виходи, які забезпечують фізичну взаємодію пристрою з об’єктом автоматизації. Саме через ці канали ПЛК зчитує інформацію про стан технологічного процесу, приймає рішення відповідно до закладеної логіки та передає сигнали для виконання керувальних дій. Входи й виходи є основними інтерфейсами між цифровою логікою керування, реалізованою у мікропроцесорі, та реальним середовищем, яке складається з датчиків, виконавчих механізмів, сигналізаторів та інших компонентів. Розуміння їхнього функціонального призначення, типів, параметрів та принципів підключення є обов’язковим для будь-якого інженера, який займається проєктуванням, програмуванням чи обслуговуванням систем автоматизації.

Усі входи ПЛК поділяються на цифрові та аналогові залежно від типу сигналу, який вони обробляють. Цифрові входи приймають дискретну інформацію, яка має лише два можливі стани: логічний нуль (0) і логічну одиницю (1), що зазвичай відповідає низькому або високому рівню напруги. Такі входи використовуються для зчитування стану кнопок, перемикачів, кінцевих вимикачів, оптичних або індуктивних датчиків присутності, сигналів з реле, контакторів, а також з іншого цифрового обладнання. Напруга на цих входах може бути постійною або змінною, залежно від типу контролера. Найбільш поширеними є ПЛК з живленням 24 В постійного струму, оскільки це стандартне значення у більшості промислових систем через свою безпечність, низьку вартість джерел живлення і високу надійність. Однак існують моделі, здатні працювати з 120 або 230 В змінного струму, що дозволяє безпосередньо підключати побутові перемикачі або індустріальні пристрої без проміжних компонентів.

У свою чергу, аналогові входи дають змогу зчитувати змінні сигнали, значення яких можуть коливатися в межах певного діапазону. До таких сигналів належать струм 4–20 мА, напруга 0–10 В, 0–5 В, а також сигнали з термопар, терморезисторів типу PT100/PT1000, датчиків тиску, рівня, витрати, температури тощо. Аналогові сигнали обробляються за допомогою внутрішніх аналого-цифрових перетворювачів (АЦП), які перетворюють фізичну величину у цифрове представлення, що надалі використовується у програмі контролера. Роздільна здатність АЦП, як правило, становить 10, 12 або 16 біт, що визначає точність перетворення та діапазон значень. Наприклад, 12-бітний АЦП може представити 4096 унікальних рівнів сигналу, що забезпечує достатню точність у більшості індустріальних застосувань. Окрім цього, деякі входи мають можливість гальванічної розв’язки, що дозволяє уникати паразитних струмів між пристроями і підвищує електробезпеку системи.

Виходи ПЛК, як і входи, поділяються на цифрові та аналогові. Цифрові виходи використовуються для керування станом пристроїв: увімкнення або вимкнення виконавчих механізмів, таких як реле, магнітні пускачі, електромагнітні клапани, сигнальні лампи, зумери, соленоїди тощо. Цифрові виходи можуть бути транзисторного або релейного типу. Транзисторні виходи, у свою чергу, поділяються на відкритий колектор (NPN) і відкритий емітер (PNP) і є швидкодіючими, тому широко застосовуються для керування пристроями, що потребують високої частоти перемикання, зокрема в імпульсних системах, енкодерах або ПЧ. Релейні виходи більш універсальні, оскільки дозволяють комутувати як змінний, так і постійний струм, однак вони мають менший ресурс, повільнішу реакцію і схильні до зносу механічних контактів [17].

Аналогові виходи, у свою чергу, застосовуються для безперервного керування пристроями, де важлива точність і плавність регулювання. Типовими прикладами є частотні перетворювачі, серводвигуни, регулятори температури, дозувальні системи, пропорційні клапани. Як і у випадку з

аналоговими входами, сигнал на аналогових виходах може бути представлений у вигляді струму (4–20 мА) або напруги (0–10 В). Важливо, щоб виконавчий механізм був сумісним із вихідним сигналом контролера, інакше потрібні додаткові перетворювачі. Крім того, деякі ПЛК мають так звані PWM (широтно-імпульсні) виходи, які можуть емулювати аналоговий сигнал, модуляцією ширини імпульсу. Це дозволяє заощаджувати апаратні ресурси, особливо у випадках, коли не потрібна висока точність керування.

Принцип підключення входів і виходів ПЛК базується на правильній організації електричних ланцюгів, врахуванні полярності, ізоляції та електромагнітної сумісності. Для цифрових входів і виходів важливо забезпечити спільну точку заземлення або мінус живлення, особливо при використанні зовнішніх джерел живлення. При неправильному підключенні виникає ризик псування входів, фальшивих спрацювань, паразитних сигналів. У випадку релейних виходів рекомендується використовувати захисні діоди або RC-ланцюги для гасіння індуктивних струмів, що виникають при розриві кола. Для аналогових входів дуже важливо використовувати екраніровані кабелі, ізоляцію від завад та правильне масштабування сигналу. У багатьох випадках застосовуються сигнальні перетворювачі, що не тільки адаптують рівень сигналу, але й забезпечують гальванічну розв’язку, ізоляцію, компенсацію температурних дрейфів [8].

У промисловій практиці окреме значення мають спеціалізовані типи входів та виходів, які не завжди класифікуються як звичайні цифрові чи аналогові, проте виконують важливі функції у конкретних додатках. Одним із прикладів таких входів є імпульсні або швидкі входи, які дозволяють зчитувати високочастотні сигнали з енкодерів, тахометрів, сенсорів швидкості або імпульсних витратомірів. Вони використовуються для обчислення швидкості обертання валів, вимірювання довжини, обліку кількості продукції, визначення координат або кутових положень. Такі входи мають спеціальну апаратну обробку сигналів, іноді з використанням окремих таймерів або лічильників, що дозволяє фіксувати навіть короткочасні імпульси з високою

точністю. Іноді ці входи інтегровані в спеціальні модулі, які підтримують роботу в режимі квадратичного кодування (Quadrature Encoder Input) для двоканального зчитування позицій.

До спеціалізованих виходів належать також швидкодіючі імпульсні виходи, які дозволяють реалізовувати широтно-імпульсну модуляцію, генерацію високочастотних сигналів або навіть реалізацію елементарного управління сервоприводами. Такі виходи необхідні в системах позиціонування, де точність і швидкодія мають вирішальне значення. Вони забезпечують гнучке керування за допомогою параметрів частоти, шпаруватості та тривалості імпульсів, що дозволяє налаштовувати динаміку виконавчого пристрою. Важливо також відзначити, що деякі ПЛК можуть містити програмовані виходи, тобто такі, що можуть перемикатися між режимами — наприклад, з аналогового на цифровий, залежно від конфігурації в програмному середовищі.

Сучасні ПЛК також підтримують диференційовану обробку сигналів: це означає, що один і той самий вхід може трактуватися як позитивний або негативний імпульс, рахувати зростання або спадання фронтів сигналу, враховувати рівень або імпульсну тривалість. Це відкриває широкі можливості для реалізації прецизійних обчислень і точного контролю. Наприклад, у металургії або деревообробці навіть незначна затримка в обробці імпульсу може спричинити помилки в довжині розрізу матеріалу. Тому ПЛК, які мають функції обробки швидких імпульсів, значно підвищують точність і продуктивність технологічного процесу.

У контексті побудови надійної системи підключення важливу роль відіграє топологія підключення входів і виходів. Відомо, що великі відстані між ПЛК та датчиками можуть призводити до втрат сигналу, наведень і похибок. Тому інженери часто застосовують буферні модулі, активні конвертери, повторювачі, а також цифрові інтерфейси, які підтримують довгі лінії передачі. Наприклад, замість передавання аналогового сигналу 0–10 В на великій відстані краще використовувати струмову петлю 4–20 мА, оскільки

вона менш чутлива до електромагнітних завад, дозволяє легко виявити обриви ланцюга (якщо струм падає нижче 4 мА) і забезпечує стабільність вимірювання.

Ще один важливий аспект — це ізоляція. У багатьох ПЛК кожен вхід або група входів мають оптичну або трансформаторну гальванічну розв’язку. Це дозволяє уникнути взаємного впливу між каналами, підвищити електробезпеку та зменшити ризик пошкодження мікросхем при виникненні коротких замикань або високовольтних імпульсів. Подібна розв’язка особливо важлива у вибухонебезпечних середовищах, а також при роботі з потужними промисловими пристроями, які можуть створювати сильні завади у мережі живлення. У деяких системах із підвищеними вимогами до безпеки використовуються входи з подвоєною ізоляцією, що забезпечують ще вищий рівень захисту.

У проєктуванні та експлуатації ПЛК слід враховувати також логіку зчитування та обробки сигналів у циклі виконання програми. Контролер зчитує всі входи на початку циклу, зберігає їхні значення у внутрішній пам’яті, виконує програму на основі цих значень, після чого оновлює вихідні сигнали. Це означає, що реальний фізичний стан входу може змінитися під час виконання програми, але не буде врахований до наступного циклу. Для усунення цієї затримки іноді використовуються переривання, особливо для критичних подій, таких як аварійні зупинки, перевищення тиску або втручання оператора. Переривання дозволяють негайно обробити подію без очікування завершення циклу, однак при неправильному використанні можуть створити конфлікти ресурсів або непередбачувану поведінку системи.

З точки зору обслуговування важливо враховувати, що багато ПЛК мають вбудовані діагностичні функції для входів і виходів. Наприклад, індикація стану кожного входу або виходу за допомогою світлодіодів дозволяє миттєво визначити, чи сигнал надходить або чи виконується команда. У більш складних системах діагностика може включати фіксацію помилок підключення, обриву дроту, короткого замикання, перевантаження за

струмом, а також внутрішню самодіагностику каналів. Усі ці події можуть бути збережені в буфері пам’яті, передані до SCADA-системи або використані для автоматичного інформування обслуговуючого персоналу. Таким чином, правильна організація системи входів і виходів безпосередньо впливає не лише на функціональність, а й на надійність, безпеку та ефективність усієї автоматизованої установки.

Поряд з фізичними характеристиками сигналів важливу роль відіграють логічні рівні та сумісність входів і виходів ПЛК із зовнішнім обладнанням. Наприклад, у цифрових входах поширеним є застосування так званої транзисторної логіки, де рівень логічної одиниці відповідає напрузі від 15 до 30 В, а рівень нуля — 0 до 5 В. Проте існують також входи, що працюють з негативною логікою, коли навпаки — нуль означає активний стан. Такі особливості потрібно враховувати під час проєктування схеми підключення, щоб уникнути логічних помилок і помилкового трактування сигналів. Аналогічно, цифрові виходи можуть мати або так звану «джерельну» (sourcing), або «стічну» (sinking) топологію. У першому випадку вихід забезпечує позитивну напругу, у другому — замикає коло на землю. Неврахування цього може призвести до ситуації, коли виконавчий пристрій просто не спрацює, незважаючи на логічну активність виходу.

Особливу увагу слід приділяти також узгодженню електричних характеристик між ПЛК та підключеним обладнанням. Наприклад, якщо вихід контролера має обмеження на струм у 0,5 А, а підключене реле потребує 0,7 А, то рано чи пізно виникне перевантаження, що призведе до спрацювання захисту або навіть пошкодження вихідного транзистора. У такому разі рекомендується використовувати проміжні реле або зовнішні підсилювачі, які дозволяють зняти навантаження з самого контролера. Також потрібно пам’ятати про важливість захисних елементів, зокрема варисторів, діодів Шотткі, RC-фільтрів, які зменшують негативний вплив комутаційних процесів, електричних завад і паразитних імпульсів.

У контексті промислової безпеки важливою темою є реалізація функціонального резервування в системі входів і виходів. Деякі ПЛК дозволяють конфігурувати подвійні входи або дублювання каналів, що дає змогу порівнювати сигнали з двох джерел і виявляти розбіжності. У випадку критичних систем — наприклад, у хімічній промисловості або системах пожежогасіння — подібна надлишковість дозволяє уникнути хибних спрацьовувань і підвищити загальний рівень надійності. У таких системах також широко застосовуються «безпечні» входи, які реалізують постійне самотестування і повідомляють про збої або нестандартну поведінку.

Варто відзначити, що у великих системах із численними точками входу та виходу виникає потреба в організації так званих віддалених I/O станцій. Це модулі, що встановлюються безпосередньо на об’єкті — поруч із датчиками чи виконавчими механізмами — і з’єднуються з головним ПЛК через польову шину (наприклад, Profibus, Modbus, EtherCAT). Така структура зменшує витрати на прокладку кабелів, підвищує гнучкість у розташуванні обладнання та спрощує обслуговування. Більше того, багато таких станцій мають можливість гарячої заміни модулів, тобто відключення й підключення без зупинки системи, що особливо важливо у безперервних процесах.

У межах інтеграції з HMI- або SCADA-системами, входи та виходи ПЛК також відіграють роль джерела даних для візуалізації, моніторингу та аналітики. Оператору важливо знати не лише факт активації певного датчика або виконання дії, а й мати змогу відстежити часові характеристики, частоту спрацювань, а також виявляти тенденції або аномалії у функціонуванні обладнання. Для цього в ПЛК зазвичай передбачено функціональність архівування даних, використання флагів, лічильників подій, а також часових міток. Вся ця інформація передається в SCADA через OPC, Modbus або інші протоколи, після чого відображається у вигляді графіків, діаграм або алертів. Таким чином, роль входів і виходів ПЛК виходить за межі лише фізичного зчитування та керування — вони стають сенсорами системного аналізу, джерелами цифрових слідів технологічного процесу.

Підсумовуючи, можна зробити висновок, що структура, тип і організація входів та виходів програмованого логічного контролера визначають не лише його здатність до керування, а й рівень надійності, точності, швидкодії та інтеграційного потенціалу. Вони є тим каналом, через який логіка, запрограмована у пристрої, взаємодіє зі складним і динамічним середовищем реального виробництва. Без належного розуміння і грамотного проєктування вхідно-вихідної структури навіть найдосконаліший контролер не зможе забезпечити ефективне та безпечне функціонування автоматизованої системи. Сучасні тенденції — такі як модульність, розширюваність, діагностичні функції, програмована логіка та цифрова трансформація — лише підсилюють вимоги до точного інженерного підходу у роботі з входами та виходами. Тому питання вибору типів сигналів, їх електричних параметрів, методів підключення, ізоляції, резервування і діагностики повинні розглядатися з такою ж увагою, як і розробка програмного забезпечення або вибір архітектури всієї системи керування.

Висновок за розділом 1

У результаті проведеного аналізу у першому розділі було досліджено основні види програмованих логічних контролерів (ПЛК), їх класифікацію та сфери застосування. Сучасні ПЛК демонструють широкий спектр функціональних можливостей, які дозволяють використовувати їх у найрізноманітніших галузях — від простих локальних систем автоматизації до розподілених інтелектуальних мереж.

Розглянуто конструктивні особливості компактних, модульних та розподілених контролерів, а також тенденції розвитку систем автоматизації на базі ПЛК. Окрему увагу приділено питанням інтеграції ПЛК з мережевими технологіями, використанню комунікаційних протоколів, підвищенню рівня безпеки та надійності, а також впровадженню концепцій функціональної безпеки та енергоефективності.

Таким чином, ПЛК продовжують відігравати ключову роль у побудові сучасних систем автоматизації, забезпечуючи високу гнучкість, масштабованість та надійність керування технологічними процесами. Проведений огляд створює необхідне теоретичне підґрунтя для подальшого розроблення практичної апаратної частини ПЛК у наступному розділі.

РОЗДІЛ 2

ВИБІР ТА ОБГРУНТУВАННЯ МІКРОКОНТРОЛЕРА ДЛЯ ПЛК

2.1 Вибір мікроконтролера

Вибір мікроконтролера для побудови програмованого логічного контролера (ПЛК) є одним з ключових етапів проектування автоматизованої системи. Саме мікроконтролер визначає обчислювальні можливості ПЛК, його функціональність, наявність необхідних апаратних інтерфейсів, можливість гнучкого конфігурування та інтеграції у промислові мережі. Вибір апаратної платформи безпосередньо впливає на надійність, швидкодію та енергоефективність кінцевого пристрою.

При виборі мікроконтролера для конкретної задачі автоматизації необхідно враховувати кілька основних критеріїв. Насамперед слід визначити необхідну кількість входів/виходів, що забезпечують взаємодію ПЛК з об’єктом керування. Для більшості задач ПЛК потрібна підтримка як дискретних (цифрових), так і аналогових входів. Це дозволяє контролювати стан технологічного процесу, зчитуючи сигнали з датчиків різного типу — як цифрових, так і аналогових. У свою чергу, керуючі виходи повинні дозволяти формувати сигнали для приводу виконавчих механізмів, таких як електромагнітні реле, контактори, двигуни тощо.

Другим критично важливим фактором є наявність вбудованих інтерфейсів зв’язку. У сучасних ПЛК дедалі більшої ваги набуває можливість інтеграції у промислові мережі для організації розподілених систем автоматизації. Серед найбільш поширених інтерфейсів слід відзначити RS485, Ethernet, CAN та інші. Саме інтерфейс RS485, завдяки своїм характеристикам, широко застосовується у промислових умовах для організації стійкого до завад та відносно дешевої лінії зв’язку між контролерами та іншими пристроями.

Крім того, мікроконтролер повинен мати достатню обчислювальну потужність та об’єм пам’яті для реалізації алгоритмів керування та обробки даних. При виборі контролера для ПЛК важливо оцінити, наскільки ефективно він зможе виконувати задачі у реальному часі. Зазвичай ПЛК працює у циклічному режимі, опитуючи входи, обробляючи отриману інформацію, формуючи сигнали на виходи та організовуючи обмін з іншими вузлами системи. Затримки при виконанні цих операцій мають бути мінімальними, що обумовлює вибір мікроконтролера з відповідною тактовою частотою та оптимізованою архітектурою.

Не менш важливим є питання енергоспоживання мікроконтролера. В умовах промислової експлуатації ПЛК часто працює у цілодобовому режимі з тривалим терміном служби. Використання мікроконтролера з оптимальним споживанням енергії дозволяє знизити теплове навантаження на систему, спрощує конструкцію корпусу та систем охолодження, а також підвищує загальну надійність пристрою.

При розгляді можливих варіантів для розроблення ПЛК у рамках цієї дипломної роботи було проаналізовано кілька груп мікроконтролерів, які широко застосовуються у подібних задачах.

У першу чергу було розглянуто мікроконтролери сімейства AVR компанії Microchip (раніше Atmel), зокрема моделі ATmega328, ATmega16, ATmega32. Ці мікроконтролери мають гарно розвинену периферію, вбудовані модулі UART, ADC, достатню кількість цифрових входів/виходів. До їх переваг можна віднести зручну архітектуру та велику підтримку з боку спільноти розробників. Проте основним обмеженням для використання саме у ПЛК стало відносно обмежене число портів введення/виведення при необхідності реалізації значної кількості дискретних входів та виходів.

Другим кандидатом були мікроконтролери STM32 (ARM Cortex-M0/M3/M4). Це потужні 32-розрядні контролери з дуже широкими можливостями периферії, високою тактовою частотою та великим об’ємом пам’яті. Проте для даної задачі їх застосування було б надмірним як за

функціональністю, так і за собівартістю. Використання такого класу мікроконтролерів має сенс у складніших системах автоматизації з необхідністю обробки значних масивів даних або складної логіки. Для компактного ПЛК у межах дипломного проєкту це було б невиправдано.

Окремо було розглянуто мікроконтролери ESP32 та ESP8266, які мають інтегровані засоби бездротового зв'язку (Wi-Fi, Bluetooth). Вони активно застосовуються у проєктах IoT (Internet of Things). Проте наявність бездротового інтерфейсу не була критично важливою для цілей даної системи, а застосування таких мікроконтролерів передбачало б додаткові складності з організацією промислового інтерфейсу RS485 та підключенням значної кількості дискретних входів/виходів.

У результаті всебічного аналізу було прийнято рішення використовувати мікроконтролер сімейства PIC — а саме, модель PIC16F877A компанії Microchip. Це один із найпопулярніших представників серії PIC16, що забезпечує оптимальне поєднання функціональних можливостей, простоти програмування та надійності [3].

Серед основних аргументів на користь вибору PIC16F877A слід зазначити такі: по-перше, мікроконтролер має достатню кількість цифрових входів та виходів (33 I/O лінії), що дозволяє реалізувати необхідну для ПЛК конфігурацію системи керування. По-друге, він має вбудований модуль UART, який легко інтегрується з мікросхемою MAX485 для побудови інтерфейсу RS485. Це дозволяє реалізувати надійний обмін даними з іншими пристроями системи автоматизації.

Крім того, у PIC16F877A інтегровано модуль аналого-цифрового перетворювача (ADC) на 10 біт, що дає змогу зчитувати аналогові сигнали з точністю, достатньою для більшості промислових задач. Значним плюсом є також наявність достатнього обсягу пам'яті програми (Flash — 14 КБ) та оперативної пам'яті (368 байт RAM), що забезпечує комфортне розміщення усіх необхідних підпрограм та змінних.

Особливо важливим чинником при виборі PIC16F877A стало його широке використання у промисловій практиці. Завдяки великій кількості доступної документації, прикладів коду та розвинутій екосистемі засобів розроблення, впровадження цього мікроконтролера у проєкт було максимально спрощеним. При цьому зберігалася висока гнучкість програмної частини, що дозволяє легко модифікувати або розширювати функціонал ПЛК у разі потреби.

Слід також зазначити, що мікроконтролери сімейства PIC відомі своєю енергоефективністю та низьким рівнем споживання у режимах активної роботи та очікування. Це є суттєвою перевагою для пристроїв, які працюють у постійному циклічному режимі, зокрема для ПЛК, що встановлюються у розподілених системах з автономним або резервованим живленням.

Таким чином, вибір мікроконтролера PIC16F877A для розроблення програмованого логічного контролера у межах цієї дипломної роботи є цілком обґрунтованим. Він забезпечує необхідний функціонал для реалізації проектованої системи, підтримує всі необхідні апаратні інтерфейси, відрізняється високою надійністю та широко підтримується спільнотою розробників.

2.2 Характеристики мікроконтролера PIC16F877A

Мікроконтролер PIC16F877A (рис.2.1) компанії Microchip є одним із найбільш популярних і широко використовуваних представників 8-розрядних мікроконтролерів середнього класу. Завдяки оптимальному поєднанню функціональних можливостей, енергоефективності, простоти використання та широкої підтримки з боку розробницької спільноти, цей мікроконтролер став основою для багатьох проектів у галузі автоматизації, робототехніки, вбудованих систем керування та навчальних програм.

Його архітектура побудована за принципом RISC (Reduced Instruction Set Computer), що забезпечує високу ефективність виконання команд при мінімальній кількості тактів процесора. Завдяки цьому PIC16F877A демонструє добру швидкодію та низьке енергоспоживання, що робить його ідеальним вибором для розроблення програмованого логічного контролера (ПЛК) у складі компактної системи автоматизації.

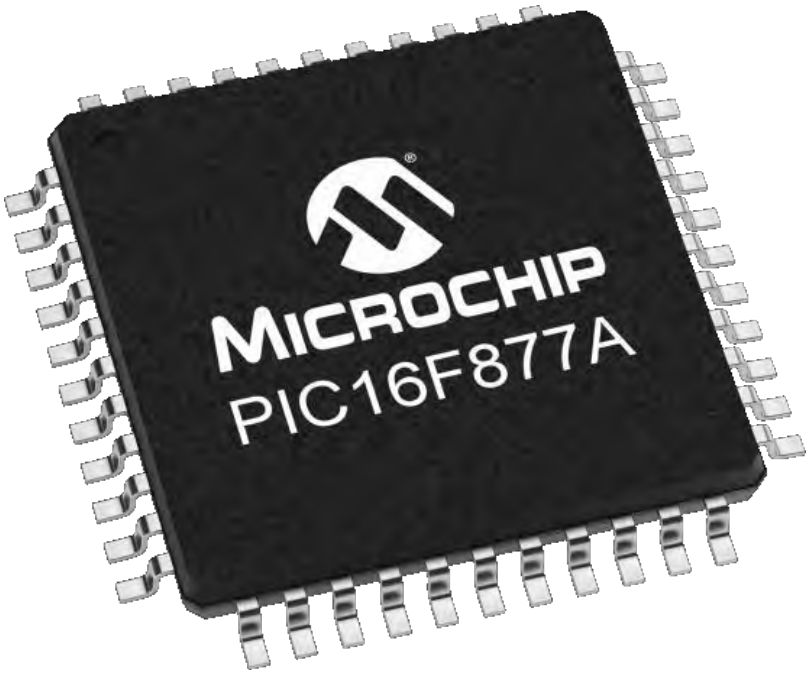


Рис. 2.1 – Мікроконтролер PIC16F877A

Однією з важливих переваг PIC16F877A є його гнучкість у конфігуруванні портів введення/виведення, наявність інтегрованих модулів обробки аналогових сигналів, підтримка широкого спектра інтерфейсів зв’язку та можливість гнучкого налаштування таймерів для реалізації алгоритмів керування у реальному часі [4].

Мікроконтролер PIC16F877A побудований на 14-бітовій командній архітектурі та має 35 базових інструкцій. Основні технічні характеристики пристрою наведено у таблиці 2.1.

Таблиця 2.1. Технічні характеристики мікроконтролеру PIC16F877A

Параметр	Значення
Архітектура	*RISC*, 8-розрядна
Тактова частота	до 20 МГц
Кількість команд	35
Кількість ліній вводу/виводу	до 40
Напруга живлення	2.0 В – 5.5 В
Тип корпуса	*DIP*40, *PLCC*44, *QFP*44
Інтерфейси зв'язку	*UART, SPI, I²C*
АЦП	10-бітовий, 8 каналів
Таймери	3 таймери (*Timer*0, *Timer*1, *Timer*2)
Споживання у активному режимі	~15 мА при 5 В і 4 МГц

PIC16F877A має до 40 універсальних ліній введення/виведення (I/O) (рис. 2.2), розділених на п'ять портів: *PORTA*, *PORTB*, *PORTC*, *PORTD* та *PORTE* [4].

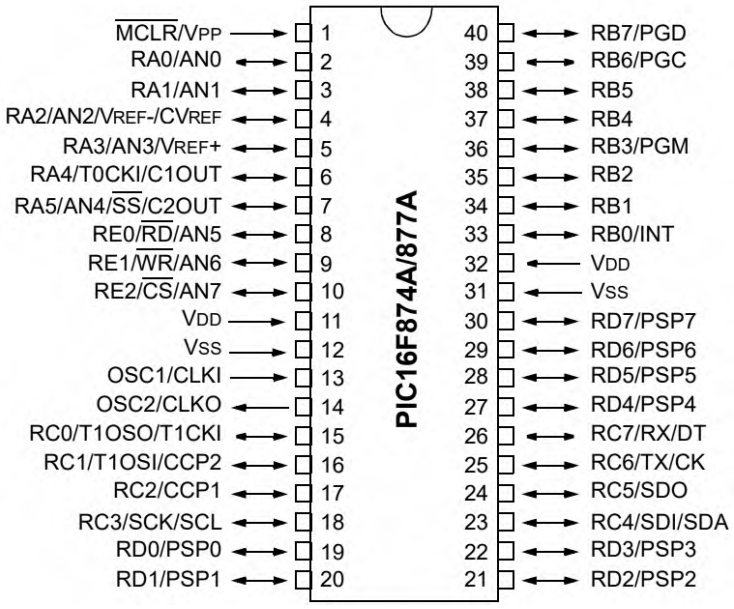


Рис. 2.2 - Розведення контактів 40-контактного мікроконтролера *PIC*16F877A

PORTA — це 6-бітовий порт, який може працювати як звичайний цифровий порт введення/виведення або виконувати функції аналогових входів для інтегрованого АЦП. Вхідні лінії RA0–RA5 можуть бути сконфігуровані для зчитування аналогових сигналів з датчиків або як цифрові входи/виходи.

PORTB — повноцінний 8-бітовий цифровий порт, який підтримує функцію внутрішніх підтягуючих резисторів на кожній лінії. Це дає змогу підключати кнопки, перемикачі та інші дискретні сенсори без необхідності додаткових зовнішніх компонентів. Також PORTB підтримує генерацію зовнішніх переривань по RB0/INT.

PORTC є 8-бітовим портом, основною особливістю якого є підтримка функцій спеціалізованих інтерфейсів зв'язку. Лінії RC6 та RC7 використовуються для роботи апаратного модуля UART (TX, RX), що дозволяє організувати надійний обмін даними, зокрема по інтерфейсу RS485 за допомогою мікросхеми MAX485. Інші лінії можуть бути використані для SPI або I^2C інтерфейсів.

PORTD — ще один 8-бітовий цифровий порт, який часто використовується для керування виконавчими механізмами. У рамках даної роботи PORTD застосовується для управління силовими ключами ULN2803, які у свою чергу керують релейними виходами. Це забезпечує чіткий та надійний контроль вихідних сигналів ПЛК.

PORTE — це 3-бітовий порт (RE0–RE2), що підтримує функції аналогових входів та управління мультиплексором АЦП. Може бути використаний для додаткових аналогових датчиків або для допоміжних функцій системи.

Модуль універсального асинхронного приймача-передавача (UART) мікроконтролера PIC16F877A є ключовим елементом для організації послідовного обміну даними. Його наявність дозволяє легко реалізувати фізичний рівень обміну по інтерфейсу RS485 за допомогою мікросхеми MAX485, що дуже важливо для промислових ПЛК.

UART підтримує програмовану швидкість передачі даних та забезпечує надійний обмін як у режимі full-duplex, так і half-duplex. Апаратна підтримка дозволяє здійснювати прийом і передавання з мінімальним навантаженням на центральний процесор.

Важливою особливістю є підтримка функцій управління перериваннями для UART. Це дозволяє реалізувати ефективні алгоритми обробки пакетів RS485 без втрати даних навіть при інтенсивному обміні.

PIC16F877A оснащено інтегрованим 10-бітовим АЦП з 8 аналоговими входами. Це дозволяє зчитувати аналогові сигнали від різноманітних датчиків (температури, тиску, освітленості тощо) та перетворювати їх у цифрову форму для подальшої обробки програмою ПЛК.

АЦП підтримує різні режими вибору джерела опорної напруги, що забезпечує гнучке налаштування системи під специфіку конкретного застосування. Інтерфейс доступу до АЦП реалізовано просто і зручно, що дозволяє легко інтегрувати модуль у загальну архітектуру програми.

Швидкодія АЦП достатня для забезпечення роботи ПЛК у реальному часі при циклічному опитуванні датчиків.

Мікроконтролер PIC16F877A має три апаратні таймери:

Timer0 — 8-бітовий таймер/лічильник з можливістю роботи від внутрішнього або зовнішнього тактового сигналу. Часто використовується для реалізації системного таймера або обробки тайм-аутів.

Timer1 — 16-бітовий таймер/лічильник, який може працювати у режимі генерації точних інтервалів часу або використовуватися для вимірювання тривалості зовнішніх подій.

Timer2 — 8-бітовий таймер, який зазвичай застосовується у поєднанні з модулями PWM для генерації широтно-імпульсних сигналів.

Гнучка система керування таймерами дозволяє ефективно організовувати різноманітні функції реального часу у складі ПЛК.

PIC16F877A також підтримує кілька додаткових функціональних модулів, що значно розширюють можливості системи автоматизації.

Модуль широтно-імпульсної модуляції (PWM) дозволяє формувати сигнали з регульованим робочим циклом. Це може бути корисно для керування швидкістю обертання двигунів, регулювання яскравості освітлення або інших аналогічних задач.

Сторожовий таймер (Watchdog Timer) є важливим елементом підвищення надійності ПЛК. У разі виникнення програмної помилки або зависання мікроконтролера WDT автоматично ініціює перезавантаження

Підтримка зовнішніх переривань на деяких лініях введення дає змогу реалізувати ефективну реакцію ПЛК на події, які вимагають негайного реагування, наприклад, спрацювання аварійного вимикача або надходження сигналу від високопріоритетного датчика.

PIC16F877A характеризується досить низьким енергоспоживанням. У активному режимі при тактовій частоті 4 МГц споживання становить близько 15 мА при напрузі живлення 5 В. У режимі "Sleep" споживання падає до мікроамперного рівня (~0.1 мкА), що дозволяє використовувати контролер у системах з обмеженим енергоспоживанням або у пристроях з живленням від батарей.

Можливість гнучкого керування режимами енергоспоживання дозволяє оптимізувати загальне енергоспоживання системи ПЛК, що є особливо важливим для автономних або енергоефективних промислових установок.

Поєднання функціональних можливостей PIC16F877A робить його ідеальною платформою для побудови програмованого логічного контролера. Велика кількість портів введення/виведення дозволяє без проблем реалізувати необхідну конфігурацію входів та виходів ПЛК. Інтегрований UART забезпечує надійний обмін по RS485, а модулі АЦП дозволяють підключати широкий спектр аналогових датчиків.

Гнучка система таймерів, підтримка зовнішніх переривань та модуль Watchdog Timer забезпечують високу надійність роботи контролера у реальному часі. Важливо також відзначити простоту розроблення

програмного забезпечення для PIC16F877A завдяки великій кількості доступних інструментів розробки та прикладів коду.

Таким чином, мікроконтролер PIC16F877A повністю відповідає вимогам до апаратної платформи для розроблення програмованого логічного контролера у межах даної дипломної роботи та забезпечує високу надійність, функціональність і енергоефективність проектованої системи.

2.3 Вибір силових ключів для керування релейними виходами

У системах автоматизації керування електромеханічними пристроями часто здійснюється за допомогою релейних виходів. Це дозволяє забезпечити електричну ізоляцію між керованими силовими ланцюгами та мікроконтролером, а також керувати навантаженнями з високим струмом або напругою, що перевищують можливості мікроконтролера. У даній дипломній роботі для реалізації релейних виходів у складі програмованого логічного контролера (ПЛК) було обрано використання зовнішніх силових ключів на основі драйвера ULN2803 (рис 2.3).

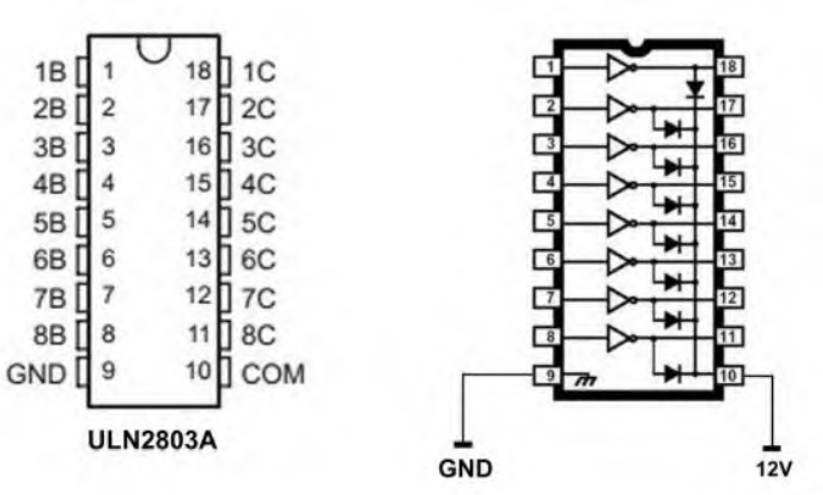


Рис. 2.3 - Структурна схема та призначення виводів драйвера силових ключів ULN2803A.

Релейний вихід у ПЛК виконує функцію керованого контакту, що замикається або розмикається за сигналом від логіки контролера. За своєю суттю реле є електромеханічним пристроєм, у якому при подачі керуючої напруги на обмотку спрацьовує магнітна система, приводячи у дію контакти силового ланцюга. Завдяки цьому ПЛК може керувати підключенням різноманітного обладнання: електродвигунів, освітлювальних пристроїв, електромагнітних клапанів тощо.

Однак безпосереднє підключення обмоток реле до мікроконтролера є неможливим або небажаним з кількох причин. По-перше, обмотка реле споживає значний струм (від десятків до сотень міліампер), що значно перевищує можливості вихідних ліній мікроконтролера. По-друге, при розриві струму в індуктивному навантаженні виникають імпульси зворотної ЕРС, які можуть пошкодити мікроконтролер або спричинити порушення його роботи.

Для усунення цих проблем у схемах керування реле застосовують силові ключі — проміжні елементи, що забезпечують узгодження логічного рівня сигналу з вимогами до керування реле за струмом і напругою. Типовим прикладом такого елемента є транзисторний ключ.

При проектуванні системи ПЛК було розглянуто кілька варіантів реалізації силових ключів:

1. Використання окремих дискретних транзисторів (наприклад, NPN біполярних транзисторів або N-канальних MOSFET) для кожного реле [19].

2. Використання інтегрованих драйверів силових виходів, що містять у собі декілька ключів з уніфікованим інтерфейсом.

Перший варіант хоча і є простим з точки зору теорії, на практиці створює ряд незручностей. При використанні окремих транзисторів для кожного реле необхідно підбирати для кожного елемента резистори для керування базою (для біполярних транзисторів) або затвором (для MOSFET), забезпечувати правильне підключення захисних діодів для індуктивного навантаження, а також враховувати теплові режими кожного ключа. Крім

того, розміщення такої кількості дискретних компонентів на друкованій платі ПЛК призводить до її значного ускладнення та збільшення розмірів.

Другий варіант передбачає використання готових мікросхем-драйверів, які поєднують у собі кілька силових ключів з усіма необхідними елементами для надійного керування індуктивними навантаженнями. Це дозволяє значно спростити схему, зменшити кількість компонентів на платі та підвищити загальну надійність конструкції.

Серед великої кількості доступних мікросхем-драйверів було обрано ULN2803 — класичний 8-канальний драйвер Дарлінгтона. Ця мікросхема давно зарекомендувала себе як надійне та зручне рішення для керування реле у ПЛК та інших системах автоматизації.

ULN2803 містить у собі вісім незалежних каналів, кожен з яких складається з пари транзисторів Дарлінгтона, що забезпечує високе підсилення струму. Завдяки такій структурі навіть при подачі невеликого струму на вхідний пін мікросхеми можна отримати на виході струм, достатній для надійного керування обмоткою реле.

Крім того, кожен канал ULN2803 має вбудований захисний діод для індуктивного навантаження. Це дозволяє без додаткових зовнішніх компонентів захистити ключі від імпульсів зворотної ЕРС, що виникають при вимиканні реле. Основні характеристики ULN2803 наведено у таблиці 2.2 [6].

Таблиця 2.2. Характеристики ULN2803

Параметр	Значення
Кількість каналів	8
Струм колектора	до 500 мА на канал
Загальний струм	до 2,5 А сумарно
Напруга колектора-емітера	до 50 В

Напруга логічного рівня входу	*TTL* / *CMOS*-сумісний
Захисні діоди	вбудовані на кожному каналі
Тип корпусу	*DIP*18, *SOIC*18, *SOP*18
Робочий діапазон температур	-40 °C до +85 °C

ULN2803 має дуже простий інтерфейс: кожен вхідний пін (IN1–IN8) підключається до вихідного порту мікроконтролера, а відповідний вихідний пін (OUT1–OUT8) — до одного з кінців обмотки реле. Другий кінець обмотки підключається до джерела живлення (+12 В або +24 В, залежно від типу реле). Загальний вивід захисних діодів (COM) підключається до позитивної шини живлення для замикання імпульсів зворотної ЕРС.

При подачі високого логічного рівня на вхід INx у транзисторній парі Дарлінгтона відкривається канал, і струм протікає через обмотку реле, активуючи його. При знятті керуючого сигналу канал закривається, реле вимикається.

Важливою перевагою ULN2803 є його сумісність з логічними рівнями мікроконтролерів, зокрема PIC16F877A. Вихідні порти мікроконтролера можуть безпосередньо керувати входами ULN2803 без потреби у проміжних буферних каскадах.

Ще однією перевагою є висока допустима напруга комутованого ланцюга (до 50 В), що дозволяє використовувати реле з широким спектром номіналів. Завдяки високому граничному струму колектора (до 500 мА на канал) ULN2803 підходить для керування практично будь-якими стандартними реле, що застосовуються у ПЛК.

При виборі ULN2803 було також враховано наявність у виробника Microchip (та інших постачальників) повного набору документації та

прикладів застосування цієї мікросхеми у системах керування реле. Це дозволило швидко інтегрувати її у розроблювану апаратну схему ПЛК.

У рамках проектованої системи програмованого логічного контролера ULN2803 використовується для керування вісьмома релейними виходами. Підключення мікросхеми до мікроконтролера здійснено таким чином, що кожному реле відповідає окремий вихідний пін PIC16F877A. Програмна логіка формує стани портів у відповідності до алгоритму роботи ПЛК, а ULN2803 забезпечує апаратний рівень комутації реле.

Таким чином, використання драйвера ULN2803 у складі проектованої системи забезпечує низку важливих переваг:

- значне спрощення апаратної частини керування реле;
- зменшення кількості необхідних дискретних компонентів;
- забезпечення електричного захисту ключів від імпульсів зворотної ЕРС;
- підвищення надійності та довговічності системи;
- сумісність з логічними рівнями мікроконтролера;
- зручність монтажу та обслуговування.

Вибір ULN2803 є цілком обґрунтованим з точки зору вимог до системи програмованого логічного контролера. Поєднання простоти інтеграції, достатніх технічних характеристик та високої надійності робить цю мікросхему оптимальним рішенням для побудови релейних виходів ПЛК.

У майбутньому, у разі необхідності, ця схема може бути легко масштабована або модифікована. Наприклад, при збільшенні кількості керованих реле можна просто додати ще одну мікросхему ULN2803 або перейти на аналогічні драйвери з розширеними можливостями. Також можливе використання модифікацій ULN2803 з підвищеними характеристиками для роботи з важкими індуктивними навантаженнями або з більшим робочим струмом.

Таким чином, застосування ULN2803 у складі розроблюваного ПЛК повністю відповідає сучасним підходам до побудови промислових систем автоматизації та забезпечує надійну і ефективну роботу релейних виходів.

2.4 Реалізація зв’язку за протоколом RS485.

У сучасних системах промислової автоматизації забезпечення надійного та ефективного обміну даними між різними пристроями є одним з ключових завдань. Особливо це стосується програмованих логічних контролерів (ПЛК), які, крім локального керування виконавчими механізмами, часто виконують функції елемента розподіленої системи автоматизації. У таких умовах важливою вимогою до організації обміну даними є забезпечення стійкості до промислових завад, можливість роботи на великих відстанях, простота реалізації та сумісність з широким спектром обладнання.

Саме з цих причин у проектованому ПЛК було обрано використання інтерфейсу RS485 для реалізації каналу зв’язку з іншими пристроями системи автоматизації. Інтерфейс RS485 є одним з найбільш поширених стандартів фізичного рівня для послідовної передачі даних у промислових мережах та володіє рядом переваг, що роблять його ідеальним для використання у системах типу ПЛК.

Загальні характеристики інтерфейсу RS485 визначаються стандартом EIA-485 (раніше RS-485), який описує методи передачі даних по симетричній диференціальній лінії. Основна концепція полягає у використанні диференціального сигналу, що передається по двох дротах (A і B), завдяки чому досягається висока стійкість до електромагнітних завад та можливість передачі даних на значні відстані — до 1200 метрів без втрати якості сигналу.

Ще однією важливою особливістю RS485 є можливість організації багатоточкової топології — на одній лінії можуть працювати до 32 передавачів і 32 приймачів (в залежності від параметрів застосованої мікросхеми). Це

дозволяє будувати гнучкі та розширювані системи, де кілька ПЛК або інших пристроїв взаємодіють між собою.

Однією з причин вибору саме RS485 для розроблюваного ПЛК стала необхідність інтеграції контролера у існуючі промислові мережі, де цей інтерфейс уже широко використовується. Крім того, RS485 має просту апаратну реалізацію, що дозволяє з мінімальними витратами додати підтримку цього інтерфейсу до мікроконтролерних систем.

Реалізація фізичного рівня RS485 у проектованому ПЛК здійснена за допомогою мікросхеми MAX485 (рис. 2.4). Це спеціалізований трансивер, призначений для організації повноцінного диференціального інтерфейсу відповідно до вимог стандарту EIA-485. Мікросхема MAX485 дозволяє реалізувати як передавання, так і прийом даних по одному двопровідному кабелю [5].

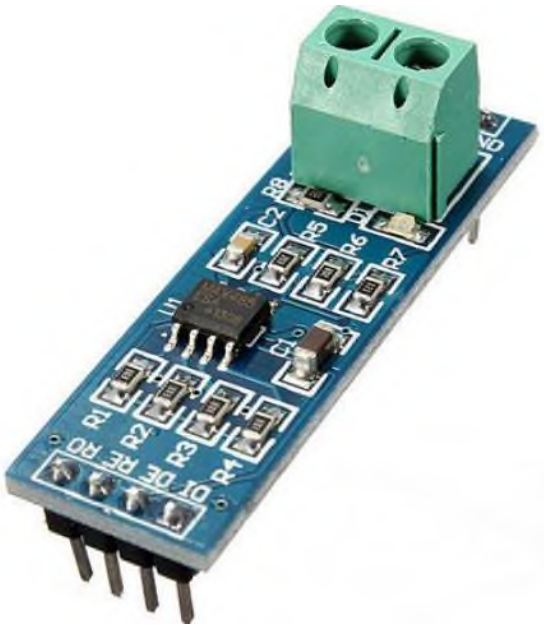


Рис. 2.4 - Модуль перетворювача RS485 (MAX485) для інтерфейсу Arduino

У роботі MAX485 застосовується Half-Duplex режим, що означає поперемінне використання лінії як для передавання, так і для приймання даних. Це найбільш економічно доцільний варіант для побудови систем на базі ПЛК, оскільки дозволяє зменшити кількість використовуваних дротів, спростити підключення та знизити вартість кабельної інфраструктури.

Для організації правильного режиму роботи мікросхеми MAX485 необхідно керувати її режимами прийому та передавання за допомогою відповідних вхідних сигналів DE (Driver Enable) та RE (Receiver Enable). У типовому режимі DE активується при необхідності передавання даних, а RE деактивується, і навпаки — при прийомі даних RE активується, а DE вимикається [7].

У розробленій апаратній схемі ці сигнали керуються програмно за допомогою портів мікроконтролера PIC16F877A. Це дає змогу забезпечити правильну синхронізацію режимів мікросхеми з програмними підпрограмами обміну даними у ПЛК.

Передавання та прийом даних по RS485 здійснюється через апаратний модуль UART мікроконтролера. Використання апаратного UART дозволяє досягти високої надійності обміну даними та знизити навантаження на центральний процесор контролера. Мікроконтролер передає та приймає послідовні дані у вигляді стандартного асинхронного протоколу (стартовий біт, 8 біт даних, стоповий біт), які мікросхема MAX485 перетворює у диференціальний сигнал для передавання по лінії.

Для забезпечення максимальної сумісності з існуючими промисловими мережами у проектованому ПЛК було прийнято рішення підтримувати такі стандартні параметри обміну:

- швидкість передачі — до 115200 бод;
- формат кадру — 8N1 (8 біт даних, відсутність парності, 1 стоповий біт);
- Half-Duplex режим роботи;
- використання контрольної суми (CRC) для забезпечення цілісності переданих даних [9].

У реалізованому програмному забезпеченні ПЛК для обробки обміну даними по RS485 передбачено кілька основних функціональних підпрограм. Насамперед це підпрограма ініціалізації інтерфейсу, яка налаштовує

апаратний UART мікроконтролера, встановлює необхідну швидкість обміну, конфігурує порти для керування режимами мікросхеми MAX485.

Далі реалізовано підпрограми передачі даних: перед початком передавання відповідний пін DE активується для переведення MAX485 у режим передавача, після чого послідовність байтів передається через UART. По завершенні передачі DE деактивується, а RE активується для переведення мікросхеми у режим прийому. Такий підхід гарантує правильну синхронізацію та запобігає конфліктам на лінії.

Прийом даних здійснюється через обробку переривань по UART або шляхом циклічного опитування буфера прийому. Для забезпечення цілісності даних кожен прийнятий пакет перевіряється за допомогою контрольної суми CRC. У разі виявлення помилки у пакеті програмне забезпечення ПЛК ініціює повторний запит даних або генерує повідомлення про помилку.

Важливим аспектом реалізації RS485 у ПЛК є також підтримка протоколу адресації. Оскільки по одній лінії може працювати кілька пристроїв, необхідно забезпечити, щоб лише адресований контролер реагував на відповідну команду. Для цього у протоколі передбачено включення у кожен переданий кадр поля з адресою одержувача. При прийомі пакета ПЛК перевіряє адресу, і лише у разі її збігу з власною конфігурованою адресою продовжує обробку пакета. Це дозволяє ефективно організувати роботу у багатоточковій мережі.

Використання RS485 також передбачає необхідність правильного узгодження лінії. Для цього на кінцях лінії встановлюються термінальні резистори (зазвичай 120 Ом), що запобігають відбиттю сигналів і покращують якість передавання даних. У розробленій схемі передбачено можливість встановлення термінальних резисторів за допомогою перемичок, що дозволяє адаптувати систему до різної конфігурації мережі.

У процесі проектування та тестування ПЛК було особливо ретельно опрацьовано питання стійкості до завад. Завдяки використанню диференціального інтерфейсу RS485 та правильної практики укладання

кабелів було досягнуто високу якість зв’язку навіть за значної довжини лінії (до 100 м у лабораторних умовах при 9600 бод). У реальних промислових умовах очікується, що система працюватиме стабільно і при більших довжинах кабелю.

Загалом, реалізація зв’язку за протоколом RS485 у проектованому ПЛК повністю відповідає сучасним вимогам до промислових систем автоматизації. Переваги, які забезпечує RS485 — стійкість до завад, можливість роботи на великих відстанях, підтримка багатоточкової топології, простота апаратної реалізації — дозволяють ефективно інтегрувати ПЛК у розподілені системи та забезпечити надійний обмін даними у широкому спектрі застосувань.

Вибір мікросхеми MAX485 як апаратної основи для реалізації інтерфейсу є цілком обґрунтованим завдяки її простоті інтеграції з мікроконтролером PIC16F877A, низькому споживанню енергії та підтримці всіх необхідних режимів роботи. Завдяки чітко спроєктованій взаємодії апаратної та програмної частин реалізовано ефективну та надійну систему обміну по RS485, що забезпечує високу якість роботи проектованого програмованого логічного контролера у складі системи промислової автоматизації.

2.5 Опис спроєктованої схеми

На основі проведеного аналізу та обґрунтованого вибору елементної бази було спроектовано повну електричну схему програмованого логічного контролера (рис. 2.5), що реалізує необхідний функціонал для автоматизації системи керування виконавчими механізмами. На рисунку представлено принципову схему проектованого пристрою.

Основу системи складає мікроконтролер PIC16F877A (на схемі позначений як IC1), що виконує функції центрального обчислювального ядра контролера. Саме він здійснює обробку сигналів з дискретних входів, прийом команд по інтерфейсу RS485, управління релейними виходами через драйвер ULN2803, а також забезпечує індикацію стану системи на LCD-дисплеї.

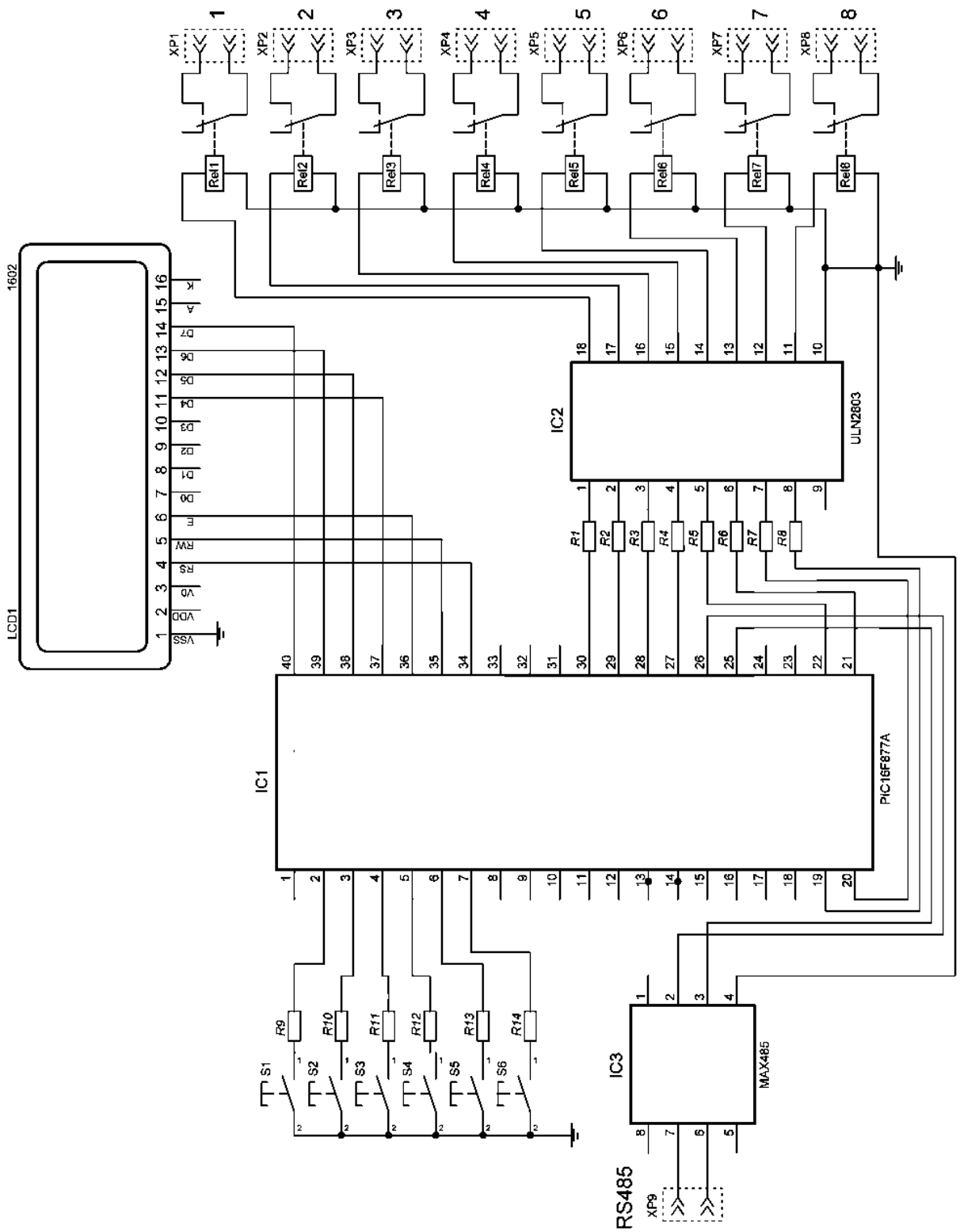


Рис. 2.5 - Принципова електрична схема програмованого логічного контролера на базі мікроконтролера PIC16F877A

У нижній частині схеми представлено блок дискретних входів. Для взаємодії оператора з системою передбачено шість кнопок керування (S1–S6). Кожна кнопка підключена до окремої лінії порту мікроконтролера через відповідний резистор (R9–R14), який виконує функцію підтягуючого резистора, забезпечуючи стабільний логічний рівень на вході при розімкненому контакті. При натисканні кнопки вхідний сигнал змінюється, що фіксується програмним забезпеченням ПЛК для реалізації відповідних функцій управління.

Цей підхід дозволяє ефективно реалізувати опитування дискретних входів у циклічному режимі, забезпечуючи коректну обробку станів кнопок та можливість реалізації різноманітних алгоритмів керування залежно від потреб конкретного технологічного процесу.

Для організації зв’язку ПЛК з іншими пристроями у складі промислової мережі у схемі передбачено використання інтерфейсу RS485. Реалізація фізичного рівня цього інтерфейсу здійснюється за допомогою спеціалізованої мікросхеми MAX485 (IC3), яка працює у режимі Half-Duplex. Передача та прийом даних здійснюється по двох проводах (A та B), підключених до роз’єму XP9.

Управління режимами прийому/передачі мікросхеми MAX485 здійснюється за допомогою ліній мікроконтролера, які підключені до відповідних виводів DE та RE мікросхеми. Це дозволяє програмно контролювати стан інтерфейсу, перемикаючи його у режим передавача при відправленні даних або у режим приймача при очікуванні вхідної інформації.

Передача та прийом послідовних даних реалізується через апаратний модуль UART мікроконтролера, що забезпечує високу надійність та стабільність обміну. У схемі передбачено пряме підключення ліній TX і RX мікроконтролера до відповідних виводів DI та RO мікросхеми MAX485.

Для управління виконавчими механізмами у системі передбачено вісім релейних виходів (Rel1–Rel8), які забезпечують можливість комутації зовнішніх навантажень. Для узгодження логічного рівня сигналів

мікроконтролера з вимогами до керування реле у схемі використано драйвер силових ключів ULN2803 (IC2).

Мікросхема ULN2803 включає у себе вісім транзисторних пар Дарлінгтона з вбудованими захисними діодами, що дозволяє ефективно керувати індуктивними навантаженнями без необхідності додаткових зовнішніх компонентів. Лінії керування (IN1–IN8) підключені безпосередньо до портів мікроконтролера (порти RD0–RD7), що дозволяє програмно встановлювати стан кожного реле.

Виходи ULN2803 (OUT1–OUT8) підключені до обмоток відповідних реле. При подачі логічної "1" на вхід драйвера відповідний транзисторний ключ відкривається, забезпечуючи протікання струму через обмотку реле, внаслідок чого його контакти замикаються або розмикаються. Це дозволяє керувати підключенням зовнішніх пристроїв, таких як двигуни, лампи, електромагнітні клапани тощо [14].

Важливою перевагою такої конфігурації є можливість реалізації гальванічної розв’язки між мікроконтролером та силовими ланцюгами, що значно підвищує безпеку та надійність роботи ПЛК. Крім того, використання ULN2803 дозволяє спростити схему та зменшити кількість дискретних компонентів.

Для індикації стану системи у схемі передбачено використання LCD-дисплея типу 1602 (LCD1), який має 2 рядки по 16 символів. Дисплей підключено до мікроконтролера за стандартною схемою з використанням 4-бітного режиму передачі даних, що дозволяє зменшити кількість задіяних портів.

До дисплея підключено лінії керування RS, RW, E, а також чотири лінії даних (D4–D7). Це забезпечує повноцінне керування відображенням інформації, включаючи виведення станів реле, показники аналогових входів, повідомлення про аварійні стани або службову інформацію для оператора.

Живлення усіх елементів схеми здійснюється від спільного джерела живлення +5 В для логічної частини та +12 В (або іншого відповідного

номіналу залежно від типу реле) для силової частини, що підключається до обмоток реле. На схемі передбачено відповідне розділення контурів живлення для забезпечення електричної безпеки та мінімізації перешкод.

Взаємозв’язок між усіма елементами схеми забезпечує реалізацію повного циклу роботи ПЛК:

- опитування стану дискретних входів (кнопок) — для формування команд керування;
- прийом команд по RS485 — для віддаленого управління ПЛК зі SCADA-системи або іншого пристрою;
- обробка отриманих команд у центральному процесорі мікроконтролера згідно з закладеним алгоритмом;
- управління станом релейних виходів через драйвер ULN2803;
- виведення актуальної інформації на LCD-дисплей;
- передача даних (стан системи, підтвердження виконання команд, аварійні повідомлення) по інтерфейсу RS485.

Завдяки такій архітектурі проектована схема забезпечує високу гнучкість та масштабованість. При необхідності її можна легко розширити — наприклад, додати нові входи або виходи, підключити додаткові модулі (датчики температури, струму тощо) або змінити алгоритм роботи ПЛК за допомогою відповідного програмного забезпечення.

Особливістю реалізованої схеми є її модульний підхід: кожен функціональний блок (входи, інтерфейс RS485, релейні виходи, індикація) реалізовано як окремий логічний модуль, що значно полегшує налаштування та обслуговування системи.

Таким чином, спроектована схема повністю відповідає вимогам до побудови програмованого логічного контролера для систем промислової автоматизації. Вона забезпечує необхідний набір функцій, високу надійність, гнучкість у використанні та можливість інтеграції у більш складні автоматизовані системи.

Висновки за розділом 2

У другому розділі було здійснено обґрунтований вибір елементної бази для побудови програмованого логічного контролера. Розглянуто різні варіанти апаратної реалізації та визначено доцільність використання мікроконтролера PIC16F877A, який забезпечує оптимальне співвідношення функціональних можливостей, простоти інтеграції та енергоефективності для задач автоматизації.

Вибір драйвера ULN2803 як силового ключа для керування релейними виходами дозволив значно спростити апаратну частину схеми та забезпечити надійне управління індуктивними навантаженнями. Реалізація інтерфейсу RS485 за допомогою мікросхеми MAX485 гарантує стійкий і надійний обмін даними з іншими пристроями у складі промислової мережі.

У результаті було розроблено повноцінну апаратну схему ПЛК, яка забезпечує інтеграцію усіх необхідних функціональних блоків: дискретних входів, релейних виходів, інтерфейсу зв’язку та системи індикації. Структура схеми дозволяє легко модифікувати або розширювати функціонал ПЛК залежно від вимог конкретного застосування.

Таким чином, сформовано апаратну платформу, що повністю відповідає поставленим вимогам та готова до подальшої розробки програмного забезпечення у наступному етапі проектування.

РОЗДІЛ 3

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ЛОГІЧНОГО КОНТРОЛЕРА

3.1 Розробка функціональної схеми та алгоритму роботи контролера

Функціональна схема проектованого логічного контролера (рис3.1) відображає структуру та логіку взаємодії основних апаратних і програмних компонентів системи. Вона створена з урахуванням вимог до гнучкої автоматизації технологічних процесів з можливістю подальшого розширення функціональності. У основі архітектури системи — мікроконтролер PIC16F877A, який виконує роль центрального обчислювального ядра і координатора роботи всіх підсистем.

Згідно з представленою функціональною схемою (Рисунок 3.1), система умовно складається з таких основних блоків: Блок введення дискретної інформації, Обчислювальний блок (PIC16F877A), Блок керування виконавчими механізмами, Блок індикації, Блок комунікації RS485, Блок периферійних модулів (ADC/Таймери), Блок живлення та Блоки підключених датчиків і виконавчих механізмів.

Кожен з цих блоків виконує специфічну функцію, а у взаємодії забезпечується реалізація повного циклу автоматизованого керування: зчитування інформації з об’єкта керування, прийняття рішень за закладеним алгоритмом, формування впливу на виконавчі механізми, а також забезпечення зворотного зв’язку та інформаційного обміну з оператором і системами верхнього рівня.

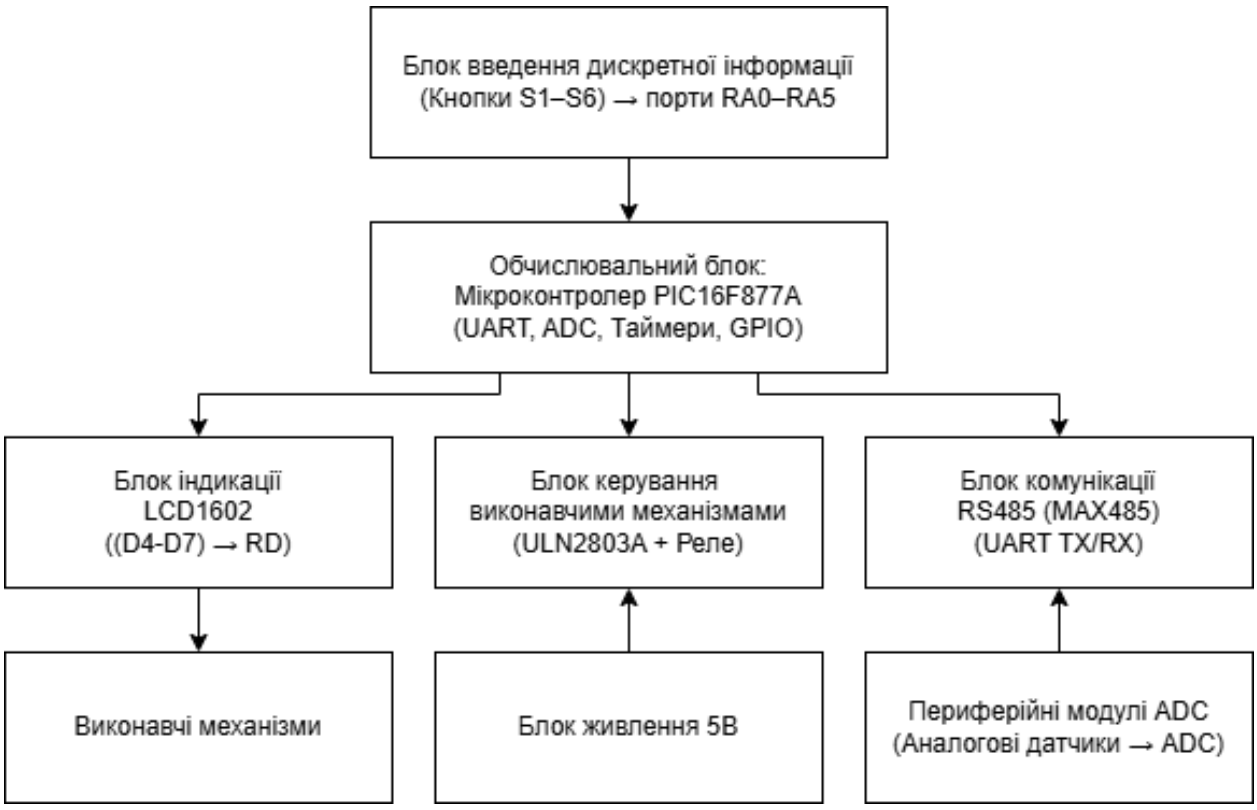


Рисунок 3.1 – Функціональна схема проєктованого логічного контролера на базі мікроконтролера PIC16F877A

Першим блоком у функціональній схемі є Блок введення дискретної інформації, що представлений шістьма кнопками (S1–S6), які підключені до відповідних ліній порту RA0–RA5 мікроконтролера. Кнопки виконують роль первинного елемента управління: через них оператор може подавати команди на запуск або зупинку технологічного процесу, вибір режимів роботи, підтвердження аварійних повідомлень, ініціалізацію обміну даними по RS485 тощо.

Принцип підключення кнопок передбачає використання резисторів підтяжки до логічної одиниці, завдяки чому у ненатиснутому стані на входах мікроконтролера фіксується стабільний високий рівень. При натисканні кнопки вхід "притягується" до землі, що розпізнається програмою як активний стан. Для усунення "дребезгу" контактів у програмі реалізується програмна фільтрація — стан кнопки фіксується тільки після стабільного утримання протягом визначеного інтервалу часу.

Важливим аспектом є програмна обробка кнопок у контексті роботи системи. Для запобігання повторному зчитуванню одного натискання використовується логіка edge detection (реєстрація фронту сигналу), що дозволяє інтерпретувати подію саме у момент переходу стану, а не під час утримання кнопки. Такий підхід підвищує зручність користування системою і дозволяє оператору точно керувати процесом.

Центральним елементом функціональної схеми є Обчислювальний блок, реалізований на базі мікроконтролера PIC16F877A. Саме цей блок забезпечує виконання всього програмного алгоритму, керування периферією та комунікаціями, формування вихідних сигналів.

Мікроконтролер отримує інформацію з входів, аналізує її у контексті заданої програми і на основі прийнятих рішень генерує відповідні керуючі сигнали. Архітектура PIC16F877A з апаратною підтримкою UART, ADC, таймерів, широким портом введення/виведення і достатнім обсягом пам’яті дозволяє ефективно реалізувати алгоритми як для простого дискретного керування, так і для більш складних систем зі зворотним зв’язком.

Програмний цикл роботи мікроконтролера передбачає постійний опитування стану дискретних входів, обробку інформації від ADC (за наявності аналогових датчиків), генерацію вихідних сигналів і підтримку комунікаційного протоколу RS485. При цьому обов’язково реалізується механізм реагування на події високого пріоритету (наприклад, аварійні стани), що гарантує високу надійність роботи системи у реальному часі.

Формування керуючих сигналів на фізичному рівні реалізується через Блок керування виконавчими механізмами. Він представлений мікросхемою ULN2803A, яка слугує драйвером для 8 реле, підключених до вихідного каскаду.

Через порт RD0–RD7 мікроконтролер подає сигнали логічного рівня, які активують відповідні канали ULN2803A. Завдяки вбудованим Дарлінгтонівським парам і захисним діодам, ULN2803 дозволяє

безпосередньо комутувати котушки реле без додаткових зовнішніх елементів. Це значно спрощує апаратну частину та підвищує надійність.

Підключені до реле виконавчі механізми можуть бути найрізноманітнішими: електромагнітні клапани, пускачі двигунів, виконавчі циліндри, освітлювальні пристрої тощо. Завдяки цьому ПЛК може управляти широким спектром об’єктів керування, як у дискретному (вкл/викл), так і у комбінованому режимі.

Важливо, що у програмному забезпеченні передбачено захист від "дріб’язкового перемикання" реле — за рахунок введення мінімального інтервалу між перемиканнями та контролю тривалості активації, що підвищує ресурс механічних елементів.

Важливою складовою функціональної схеми є Блок індикації, реалізований на базі символьного LCD-дисплея LCD1602. Цей модуль дозволяє виводити на екран службову інформацію, поточний стан системи, параметри роботи, аварійні повідомлення та інші діагностичні дані.

Дисплей підключено до порту RD мікроконтролера у чотирибітному режимі (D4–D7), що дозволяє зменшити кількість задіяних ліній I/O при збереженні повної функціональності. Управління дисплеєм здійснюється за допомогою стандартного набору команд HD44780: передача команд ініціалізації, управління курсором, очищення екрану, виведення символів.

Основні задачі, які реалізуються у блокові індикації:

- вивід поточного стану дискретних входів (наприклад, стан натиснутих кнопок).
- індикація стану виконавчих механізмів (активні реле, стан обладнання).
- індикація стану комунікаційного обміну (передача/прийом по RS485).
- вивід службових повідомлень (помилки, аварії, підтвердження дій оператора).

LCD1602 підтримує два рядки по 16 символів, що забезпечує достатній обсяг інформації для постійного моніторингу роботи системи. Вивід

інформації реалізовано за принципом циклічного оновлення з урахуванням оптимального балансу між частотою оновлення екрана та навантаженням на процесор.

Програмно передбачено буфер виводу, що дозволяє уникнути мерехтіння дисплея при оновленні лише змінних частин інтерфейсу. Завдяки цьому оператор завжди має актуальну інформацію про стан системи у зрозумілій та зручній формі.

Однією з ключових особливостей проектованого ПЛК є наявність блоку комунікації, який забезпечує обмін даними з іншими пристроями системи автоматизації. Для реалізації зв’язку використовується інтерфейс RS485, що працює через мікросхему MAX485.

Цей блок відіграє стратегічну роль у розширюваності системи: завдяки RS485 ПЛК може бути інтегрований у більші мережі (наприклад, SCADA-системи), а також обмінюватися даними з іншими ПЛК, сенсорними панелями тощо.

- особливості використання RS485:

- диференційний режим передачі забезпечує високу стійкість до завад, що критично для промислових об’єктів.

- можливість підключення до 32 вузлів на одну шину.

- велика довжина лінії зв’язку (до 1200 м).

У функціональній схемі лінії TX/RX мікроконтролера підключені до MAX485. Управління напрямком передачі реалізовано через спільну лінію DE/RE, керовану програмно. Таким чином, при передаванні даних мікроконтролер активує передавач MAX485, а після завершення — переходить у режим прийому.

Програмний стек підтримує побудову пакетного обміну: кожен пакет має структуру з адресою вузла, командою, полем даних та контрольною сумою. Це дозволяє організувати адресну мережу з керуванням за протоколом Master-Slave.

Мікроконтролер PIC16F877A має у своєму складі кілька додаткових периферійних модулів, які використовуються у проектованій системі:

ADC (аналого-цифровий перетворювач) — дозволяє зчитувати аналогові сигнали з підключених сенсорів (температура, тиск, рівень). Кожен канал ADC має роздільну здатність 10 біт, що забезпечує хорошу точність перетворення.

Таймери — використовуються для формування часових інтервалів, організації циклічного опитування, обробки "дребезгу" кнопок, генерації затримок для реле.

У функціональній схемі цей блок показаний окремо, оскільки він забезпечує критичну допоміжну функціональність для всього контролера. Таймери, наприклад, дозволяють синхронізувати операції з точністю до мілісекунд, що важливо при побудові систем реального часу.

Додатково ADC може бути використано для реалізації аналогового моніторингу параметрів об’єкта управління — що дає можливість подальшого функціонального розширення системи.

Надійна робота усіх компонентів контролера забезпечується блоком живлення. У даній системі використовується стабілізоване джерело живлення 5В, що подає енергію на мікроконтролер, дисплей, MAX485 та мікросхему ULN2803A.

Оскільки реле можуть створювати значні пускові навантаження, особливу увагу приділено розділенню ліній живлення:

- окрема лінія для живлення реле;

- окрема лінія для живлення логічної частини (МК + MAX485 + LCD).

Це дозволяє уникнути просідання напруги при перемиканні реле, що могло б призводити до збоїв у роботі ПЛК.

Також у блоці живлення передбачено:

- фільтрація живлення (конденсатори 100–470 мкФ);

- захист від перенапруги (TVS-діоди).

Таким чином, у підрозділі було розроблено та детально описано функціональну схему проектованого логічного контролера на базі мікроконтролера PIC16F877A. Було визначено склад та призначення основних апаратних блоків системи, а також їх взаємодію у процесі виконання керуючих алгоритмів. Запропонована структура забезпечує гнучкість, розширюваність та надійність роботи контролера, дозволяючи ефективно вирішувати широкий спектр задач автоматизації та забезпечувати інтеграцію у сучасні промислові мережі. Проведений аналіз створює необхідне підґрунтя для подальшого розроблення програмного алгоритму та коду програми, що буде реалізовано у наступних підрозділах.

3.2 Алгоритми роботи контролера

Після розроблення функціональної схеми наступним етапом проєктування логічного контролера стало створення основного алгоритму програми (рис. 3.2), що визначає логіку роботи всієї системи. Алгоритм побудовано з урахуванням структури функціональної схеми, технічних можливостей мікроконтролера PIC16F877A та загальних вимог до промислових систем автоматизації [18].

Основний алгоритм програми складається з кількох ключових етапів, які забезпечують повний цикл взаємодії між оператором, об’єктом керування, виконавчими механізмами та системами верхнього рівня. Розроблений алгоритм гарантує стабільну роботу контролера у реальному часі, своєчасну обробку зовнішніх подій і формування адекватної реакції на дії оператора або зміни стану технологічного процесу.

Після розроблення функціональної схеми наступним етапом проєктування логічного контролера стало створення основного алгоритму програми, що визначає логіку роботи всієї системи. Алгоритм побудовано з урахуванням структури функціональної схеми, технічних можливостей

мікроконтролера PIC16F877A та загальних вимог до промислових систем автоматизації.

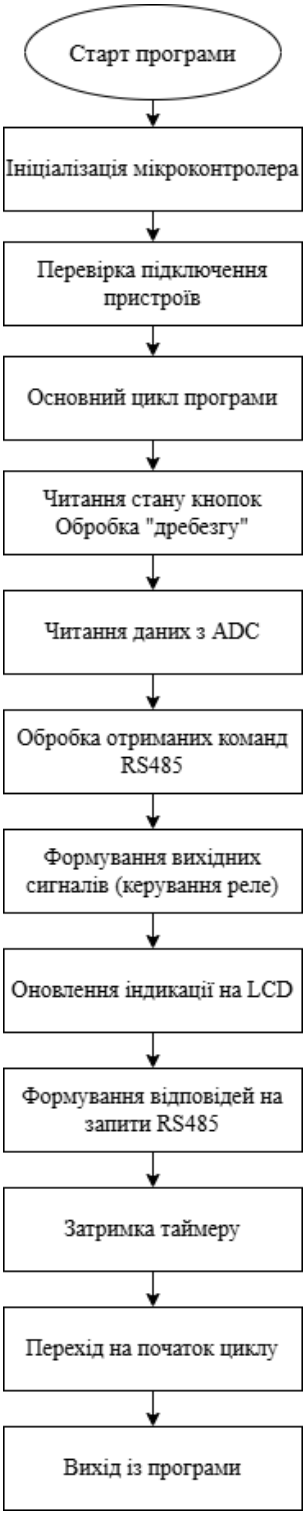


Рисунок 3.2 – Основний алгоритм програми

Після розроблення функціональної схеми наступним етапом проєктування логічного контролера стало створення основного алгоритму

програми, що визначає логіку роботи всієї системи. Алгоритм побудовано з урахуванням структури функціональної схеми, технічних можливостей мікроконтролера PIC16F877A та загальних вимог до промислових систем автоматизації.

Основний алгоритм програми складається з кількох ключових етапів, які забезпечують повний цикл взаємодії між оператором, об’єктом керування, виконавчими механізмами та системами верхнього рівня. Розроблений алгоритм гарантує стабільну роботу контролера у реальному часі, своєчасну обробку зовнішніх подій і формування адекватної реакції на дії оператора або зміни стану технологічного процесу.

Початковою фазою роботи контролера є етап старту системи. Після подачі живлення відбувається автоматичний запуск мікроконтролера, ініціалізується апаратна частина, завантажується програма з Flash-пам’яті та запускається основний цикл виконання. На цьому етапі надзвичайно важливо забезпечити коректне проходження процедури Power-On Reset, яка гарантує запуск мікроконтролера у відомому стані, без залишкових ефектів від попередніх запусків або коливань напруги живлення. У цей же момент активується Watchdog Timer, що слугує додатковим механізмом захисту від зависання програми у разі непередбачених збоїв [12].

Наступним етапом виконується ініціалізація всіх апаратних компонентів мікроконтролера та підключених периферійних пристроїв. Програма послідовно налаштовує порти введення/виведення згідно із закладеною конфігурацією: порти, до яких підключено кнопки, переводяться у режим цифрового входу; порти керування реле налаштовуються як цифрові виходи; порти, пов’язані з LCD-дисплеєм, переводяться у відповідний режим для роботи у чотирибітному інтерфейсі. Додатково ініціалізуються модулі UART, таймери та аналого-цифровий перетворювач. Ініціалізація UART включає встановлення швидкості обміну даними, конфігурацію формату кадру (8N1), активацію приймача та передавача, а також налаштування переривань для обробки подій прийому.

На етапі ініціалізації ADC відбувається конфігурація каналів, визначення опорної напруги, встановлення тривалості часу вибірки та запуск модуля у готовність до першого перетворення. Для таймерів встановлюється початковий режим роботи, визначаються коефіцієнти поділу тактової частоти, ініціалізуються регістри переривань. Особливу увагу приділено ініціалізації дисплея LCD1602: після включення живлення дисплею необхідно передати спеціальну послідовність команд для переводу у чотирибітний режим та активації режиму відображення символів.

Після завершення фази ініціалізації програма переходить до етапу перевірки підключених пристроїв. На цьому етапі здійснюється короткочасне тестування ланцюгів керування реле, дисплея та зв’язку RS485. Наприклад, проводиться імпульсне включення і виключення кожного реле з метою перевірки цілісності ланцюга та підтвердження працездатності вихідного каскаду. На дисплей виводиться повідомлення про проходження тесту. Якщо на будь-якому з етапів перевірки виявляється несправність, наприклад, неможливість активувати реле або відсутність відповіді з боку дисплея, система переходить у аварійний режим з відповідною індикацією, блокуючи подальший запуск основного циклу.

Основна частина програми реалізована у вигляді нескінченного циклу, у якому поетапно виконуються всі необхідні дії з обробки вхідної інформації та формування вихідних сигналів. Спочатку здійснюється читання стану кнопок, підключених до цифрових входів мікроконтролера. Особливість реалізації полягає у програмній фільтрації "дребезгу" контактів: кожен вхід опитується з певною періодичністю, і рішення про зміну стану кнопки приймається лише у випадку, якщо протягом визначеного часу стан залишається стабільним. Такий підхід дозволяє значно підвищити достовірність обробки натискань та уникнути хибних спрацьовувань.

Після обробки дискретних входів відбувається опитування ADC. Якщо у системі підключено аналогові датчики, на відповідні канали подаються аналогові сигнали. Програма ініціює цикл перетворення для кожного з каналів,

зчитує результат з регістру ADC та переводить його у цифровий еквівалент. Подальша обробка даних ADC залежить від конкретного застосування системи: у найпростішому випадку значення порівнюється з пороговими рівнями для прийняття рішень про включення або відключення виконавчих механізмів.

Наступним важливим етапом є обробка отриманих команд по інтерфейсу RS485. Для цього постійно контролюється стан регістрів прийому UART. При надходженні нового байта даних програма активує механізм зчитування пакета згідно із закладеною структурою протоколу. Після отримання повного пакета виконується перевірка контрольної суми. Якщо пакет коректний, відбувається аналіз команди: у разі отримання запиту на зміну стану реле відповідно модифікується керуючий сигнал; при отриманні запиту на зчитування стану входів або даних ADC формується відповідь з актуальною інформацією.

Формування вихідних сигналів для керування реле здійснюється шляхом встановлення відповідного стану ліній порту RD. Програма підтримує таблицю станів реле, яка оновлюється як на підставі натискань кнопок, так і на основі отриманих по RS485 команд. Завдяки цьому забезпечується гнучка логіка керування, коли реле можуть бути активовані як локально, так і віддалено. При зміні стану будь-якого реле програма автоматично забезпечує необхідну паузу для уникнення механічного дребезгу та захисту вихідного каскаду.

Після формування вихідних сигналів оновлюється індикація на LCD-дисплеї. Для цього використовується підготовлений буфер, у який попередньо записується актуальна інформація про стан системи: активні реле, натиснуті кнопки, аварійні повідомлення, результати вимірювань ADC або параметри обміну по RS485. Оновлення дисплея відбувається циклічно, з певною періодичністю, щоб уникнути мерехтіння.

Останнім етапом циклу є формування відповіді на отримані по RS485 запити. Якщо мікроконтролер отримав команду, яка передбачає відповідь

(наприклад, запит на зчитування стану входів), програма формує відповідний пакет згідно з правилами протоколу. Перед передачею активується передавач MAX485 шляхом встановлення відповідного стану ліній DE/RE. Після завершення передачі програма знову переводить MAX485 у режим прийому для продовження роботи у напівдуплексному режимі.

На завершення кожного циклу програма виконує невелику затримку за допомогою таймера, що дозволяє стабілізувати час виконання циклу та запобігає надмірному навантаженню процесора. Завдяки використанню точного таймера інтервал затримки може бути легко налаштований залежно від вимог до частоти оновлення системи.

Таким чином, розроблений алгоритм програми забезпечує повний цикл автоматизованої обробки вхідної інформації, формування вихідних впливів, індикацію поточного стану системи та підтримку надійного обміну даними по промисловому інтерфейсу RS485. Гнучка структура алгоритму дозволяє адаптувати контролер до широкого спектра задач автоматизації, забезпечуючи високу надійність, розширюваність та простоту інтеграції у складні промислові системи.

Підпрограма ініціалізації обладнання є одним з найважливіших етапів запуску логічного контролера, оскільки вона забезпечує підготовку всієї апаратної частини до коректної роботи у складі системи автоматизації. У процесі ініціалізації виконуються налаштування основних апаратних блоків мікроконтролера PIC16F877A, а також перевіряється працездатність підключених зовнішніх пристроїв.

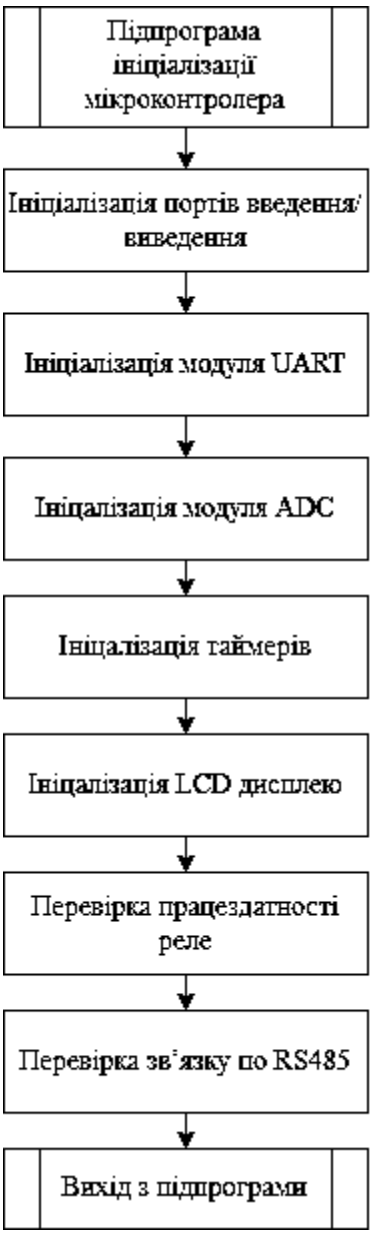


Рис 3.3 – Підпрограма ініціалізації обладнання

Підпрограма розпочинається одразу після завершення процедури скидання та початкового завантаження програми. На першому етапі здійснюється ініціалізація портів введення/виведення. Згідно з функціональною схемою контролера, частина портів налаштовується у режимі цифрового входу (для зчитування стану кнопок), інша частина — у режимі цифрового виходу (для керування реле та дисплеєм). Це забезпечує правильний обмін сигналами між мікроконтролером та периферією.

Далі виконується конфігурація апаратного модуля UART. На цьому етапі встановлюється швидкість обміну по RS485, визначається формат передавання даних (8 біт даних, без біта парності, один стоп-біт), активується приймач і передавач. Це дозволяє контролеру бути готовим до прийому і передавання даних по промисловій мережі RS485.

Наступним етапом є ініціалізація модуля ADC. Підпрограма налаштовує відповідні канали ADC, визначає джерело опорної напруги, встановлює необхідні параметри вибірки та запускає модуль у готовність до роботи. Це необхідно для коректного зчитування аналогових сигналів з підключених сенсорів.

Після цього відбувається конфігурація таймерів. Таймери налаштовуються для забезпечення необхідних часових інтервалів у роботі системи — обробка "дребезгу" кнопок, генерація затримок для реле, керування періодом оновлення індикації тощо.

Далі підпрограма переходить до ініціалізації LCD-дисплея. Відправляється необхідна послідовність команд для переведення дисплея у чотирибітний режим роботи, очищується екран, встановлюються параметри відображення.

На завершальному етапі підпрограма здійснює перевірку працездатності реле шляхом подачі тестових сигналів і перевіряє зв’язок по RS485, надсилаючи службовий запит у мережу. Лише після проходження усіх етапів ініціалізації управління передається у головний цикл програми.

Підпрограма обробки команд RS485 (рис 3.4) є ключовим елементом програмного забезпечення логічного контролера, оскільки вона забезпечує надійну взаємодію ПЛК з іншими пристроями промислової мережі. Завдяки використанню інтерфейсу RS485 та мікросхеми MAX485 досягається висока стійкість до завад і можливість організації мережі з багатьма вузлами.

Рис. 3.4 – Алгоритм підпрограми обробки команд RS485

Робота підпрограми починається з перевірки наявності прийнятого пакета у приймальному буфері UART мікроконтролера. Для цього постійно

контролюється стан флагів прийому UART. При наявності нових даних ініціюється процес зчитування пакета. Програма виконує зчитування усього пакета згідно із заданим протоколом. Після цього виконується перевірка контрольної суми (CRC), що дозволяє виявити спотворення даних під час передачі.

Наступним етапом є перевірка адреси пакета. Оскільки у мережі можуть працювати кілька пристроїв, кожен пакет містить поле адреси. Підпрограма порівнює отриману адресу з власною адресою контролера, збереженою у EEPROM. Якщо адреса збігається, пакет обробляється, інакше ігнорується.

Після цього виконується розбір коду команди. Програма аналізує тип отриманої команди — це може бути команда на керування реле, запит на зчитування стану входів, передавання даних ADC або інші спеціалізовані команди. Відповідно до типу команди викликається відповідна функція для її обробки.

Далі формується пакет відповіді. Після підготовки пакета програма переводить MAX485 у режим передавання, здійснює передачу даних по UART та повертає MAX485 у режим приймання. На цьому робота підпрограми завершується, і керування передається у головний цикл програми.

Підпрограма керування реле (рис. 3.5) забезпечує формування вихідних сигналів логічного контролера відповідно до поточного стану системи та отриманих команд. Вона виконує оновлення станів реле, до яких підключені виконавчі механізми, забезпечуючи таким чином фізичне керування об’єктом автоматизації.

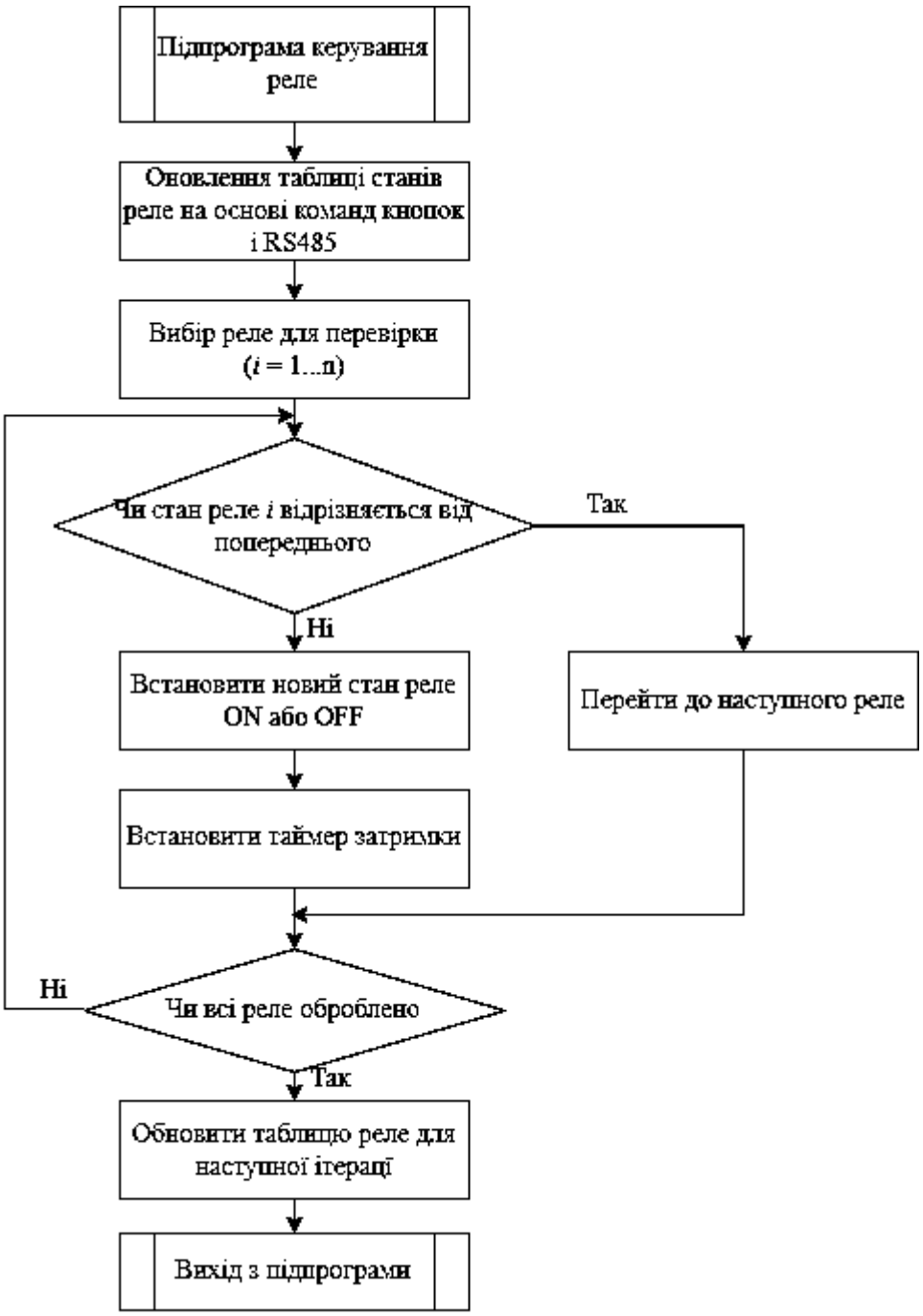


Рис. 3.5 – Алгоритм підпрограми керування реле

Робота підпрограми починається з оновлення таблиці станів реле на основі команд, що надходять від кнопок керування або по інтерфейсу RS485. Ця таблиця відображає бажаний стан кожного реле у поточному циклі. Далі у підпрограмі організовано цикл по усіх реле системи.

Для кожного реле проводиться перевірка: чи відрізняється бажаний стан реле від його поточного фізичного стану. У разі виявлення такої різниці

виконується оновлення фізичного стану реле шляхом подачі відповідного сигналу на порт виводу мікроконтролера. Це дозволяє або активувати, або деактивувати реле залежно від необхідної дії.

Після зміни стану реле активується програмний таймер затримки, який запобігає повторним швидким перемиканням. Це є важливим захистом від "дребезгу" і надмірного зношування механічних контактів реле. Якщо ж стан реле не змінився, програма переходить до обробки наступного реле.

По завершенні перевірки усіх реле таблиця актуальних станів оновлюється, що гарантує правильність подальшої роботи системи. Завершивши обробку, підпрограма передає управління у головний цикл програми, де продовжується робота контролера у нормальному режимі.

Підпрограма відповіді по RS485 є завершальним елементом циклу обробки команд у логічному контролері та виконує ключову роль у забезпеченні двостороннього обміну інформацією у промисловій мережі. Завдяки цій підпрограмі контролер може надавати відгук на запити Master-пристрою, передаючи актуальні дані про свій стан та параметри об’єкта керування.

Робота підпрограми починається з перевірки наявності запиту, що потребує відповіді. У структурі програми використовується спеціальний програмний флаг, який встановлюється під час обробки вхідних команд по RS485. Якщо флаг не активний, підпрограма миттєво завершується, повертаючи управління у головний цикл. Це дозволяє уникнути зайвого навантаження на UART та мінімізувати час обробки.

У разі наявності активного запиту підпрограма переходить до формування пакета відповіді. Структура пакета відповідає прийнятим стандартам: початковий байт STX, поле адреси, код команди, поле даних, контрольна сума (CRC) та кінцевий байт ETX. Формування структури пакета виконується у програмному буфері, що дозволяє підготувати повний блок даних для передавання.

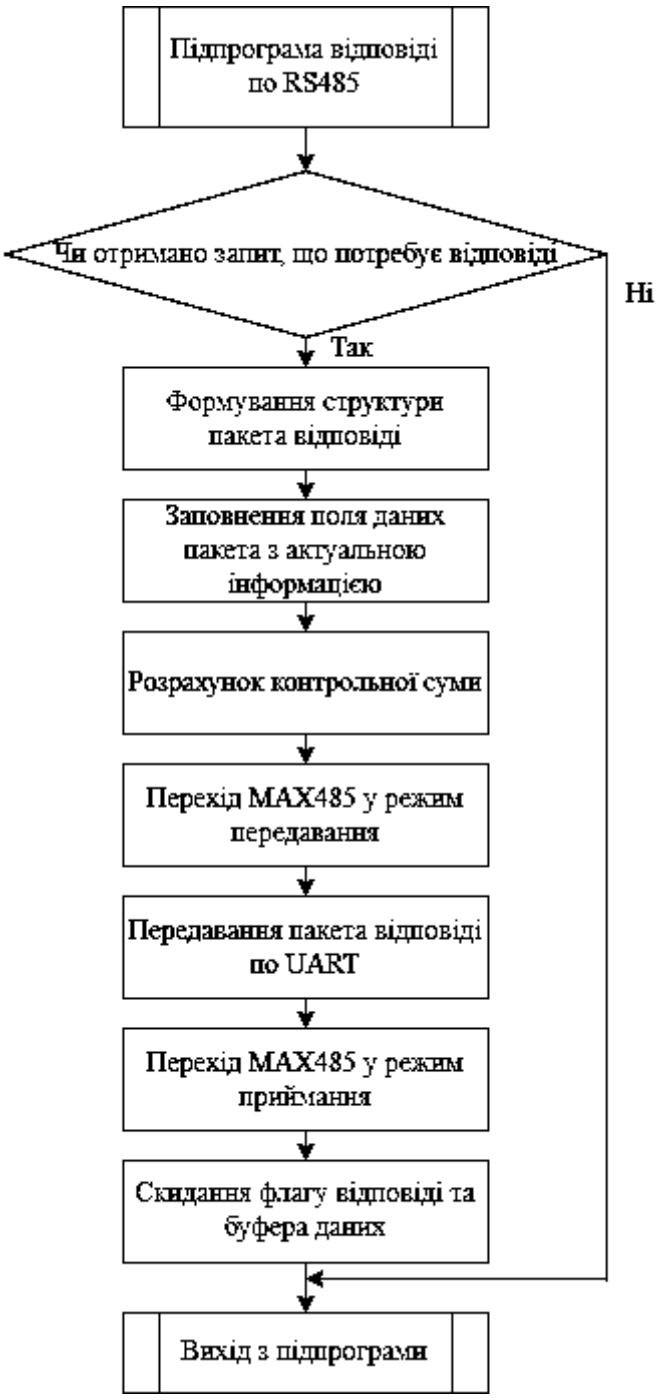


Рис. 3.6 – Алгоритм підпрограми відповіді по RS485

Поле даних заповнюється актуальною інформацією залежно від типу запиту. Це можуть бути стани дискретних входів, значення ADC, таблиця станів реле або службові повідомлення про стан контролера. Після заповнення поля даних виконується розрахунок CRC для забезпечення достовірності переданих даних.

Перед початком передавання підпрограма переводить мікросхему MAX485 у режим передавання. Це здійснюється за допомогою встановлення відповідного стану ліній DE/RE. Після цього весь підготовлений пакет передається по UART. По завершенні передачі MAX485 повертається у режим приймання. Завершальним етапом є скидання флага відповіді та очищення буфера даних, після чого підпрограма завершується.

Таким чином, у межах даного підрозділу було розроблено основний алгоритм роботи програмного забезпечення проектованого логічного контролера. Деталізовано структуру програми через побудову та опис підпрограм, що відповідають за ключові функціональні блоки системи: ініціалізацію обладнання, обробку команд по RS485, керування реле, формування відповідей у промисловій мережі.

Запропонована алгоритмічна структура забезпечує гнучку, надійну та масштабовану організацію програмного коду. Виділення окремих підпрограм дозволяє підвищити зрозумілість логіки роботи системи, спрощує тестування та подальший супровід ПЗ, а також забезпечує можливість модифікації контролера для нових задач без потреби значної переробки базового коду.

Побудовані алгоритми чітко відображають процеси, що відбуваються у контролері під час роботи у реальному часі, з урахуванням специфіки апаратної частини та промислових вимог до надійності та стабільності функціонування. Таким чином, підготовлена алгоритмічна основа є повністю готовою для переходу до практичної реалізації програмного забезпечення, що буде розглянуто у наступному підрозділі.

3.3 Програмна реалізація алгоритму логічного контролера

У даному підрозділі представлено програмну реалізацію алгоритму роботи проектованого логічного контролера у вигляді структурованого псевдокоду, виконаного у стилі LADDER-логіки. Такий підхід дозволяє наочно відобразити взаємозв’язки між окремими функціональними блоками

програми та полегшує сприйняття логіки роботи системи. Псевдокод відображає повний цикл виконання програми: від початкової ініціалізації апаратної частини до організації основного циклу керування з обробкою входів, формуванням вихідних сигналів та обміну даними по промисловій мережі RS485. Розроблена структура дозволяє легко адаптувати програму до змін вимог або конфігурації системи та забезпечує її модульність, надійність і простоту тестування.

| // Початкова ініціалізація системи
|----[NO] Power_On ----------(SET) Init_System
|----[NO] Init_System ----------(OUT) Init_Ports
|----[NO] Init_System ----------(OUT) Init_UART
|----[NO] Init_System ----------(OUT) Init_ADC
|----[NO] Init_System ----------(OUT) Init_LCD
|----[NO] Init_System ----------(OUT) Init_Timers
|----[NO] Init_System ----------(OUT) Init_Variables
|----[NO] Init_System ----------(SET) Ready_For_New_Cycle
|----[NO] Ready_For_New_Cycle ----------(SET) System_Reset
|----[NO] System_Reset ----------(OUT) Reset_Indicator
|----[NO] System_Reset ----------(OUT) Clear_Command_Buffer
|----[NO] System_Reset ----------(OUT) Clear_Button_States
|----[NO] System_Reset ----------(OUT) Clear_Relay_States
|----[NO] System_Reset ----------(OUT) Clear_LCD_Buffer
|----[NO] System_Reset ----------(SET) System_Ready
| // Основний цикл програми
|----[NO] System_Ready ----------(OUT) Main_Loop_Active
|----[NO] Main_Loop_Active ----------(OUT) Read_Buttons
|----[NO] Read_Buttons ----------(OUT) Debounce_Buttons
|----[NO] Debounce_Buttons ----------(OUT) Update_Button_States
|----[NO] Main_Loop_Active ----------(OUT) Start_ADC_Conversion

|----[NO] Start_ADC_Conversion ----------(OUT) Wait_ADC_Complete

|----[NO] Wait_ADC_Complete ---------- (OUT) Read_ADC_Values

|----[NO] Read_ADC_Values ----------(OUT) Update_ADC_Data

|----[NO] Main_Loop_Active ----------(OUT) Check_RS485_Buffer

|----[NO] Check_RS485_Buffer ----------(OUT) Validate_RS485_Packet

|----[NO] Validate_RS485_Packet ---------(OUT) Process_RS485_Command

|----[NO] Process_RS485_Command ---------(OUT) Update_Relay_Table

|----[NO] Process_RS485_Command ---------(OUT) Prepare_RS485_Response

|----[NO] Prepare_RS485_Response --------(OUT) Send_RS485_Response

|----[NO] Main_Loop_Active ----------(OUT) Process_Relay_Logic

|----[NO] Process_Relay_Logic ----------(OUT) Relay_1_Control

|----[NO] Relay_1_Control ----------(OUT) Relay_1_ON_Condition

|----[NO] Relay_1_Control ----------(OUT) Relay_1_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_2_Control

|----[NO] Relay_2_Control ----------(OUT) Relay_2_ON_Condition

|----[NO] Relay_2_Control ----------(OUT) Relay_2_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_3_Control

|----[NO] Relay_3_Control ----------(OUT) Relay_3_ON_Condition

|----[NO] Relay_3_Control ----------(OUT) Relay_3_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_4_Control

|----[NO] Relay_4_Control ----------(OUT) Relay_4_ON_Condition

|----[NO] Relay_4_Control ----------(OUT) Relay_4_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_5_Control

|----[NO] Relay_5_Control ----------(OUT) Relay_5_ON_Condition

|----[NO] Relay_5_Control ----------(OUT) Relay_5_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_6_Control

|----[NO] Relay_6_Control ----------(OUT) Relay_6_ON_Condition

|----[NO] Relay_6_Control ----------(OUT) Relay_6_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_7_Control

|----[NO] Relay_7_Control ----------(OUT) Relay_7_ON_Condition

|----[NO] Relay_7_Control ----------(OUT) Relay_7_OFF_Condition

|----[NO] Process_Relay_Logic ----------(OUT) Relay_8_Control

|----[NO] Relay_8_Control ----------(OUT) Relay_8_ON_Condition

|----[NO] Relay_8_Control ----------(OUT) Relay_8_OFF_Condition

|----[NO] Main_Loop_Active ----------(OUT) Update_LCD_Display

|----[NO] Update_LCD_Display ----------(OUT) LCD_Line_1_Update

|----[NO] LCD_Line_1_Update ---------- (OUT) Display_System_Status

|----[NO] LCD_Line_1_Update ---------- (OUT) Display_Active_Buttons

|----[NO] Update_LCD_Display ----------(OUT) LCD_Line_2_Update

|----[NO] LCD_Line_2_Update ---------- (OUT) Display_Active_Relays

|----[NO] LCD_Line_2_Update ---------- (OUT) Display_ADC_Data

| // Управління таймерами

|----[NO] Main_Loop_Active ----------(OUT) Process_Timers

|----[NO] Process_Timers ----------(OUT) Timer_Debounce_Update

|----[NO] Process_Timers ----------(OUT) Timer_Relay_Protection_Update

|----[NO] Process_Timers ----------(OUT) Timer_Display_Refresh_Update

| // Обробка аварійних станів

|----[NO] Main_Loop_Active ----------(OUT) Check_Emergency_Conditions

|----[NO] Check_Emergency_Conditions ----(OUT) Emergency_Stop_Detected

|----[NO] Emergency_Stop_Detected -------(OUT) Force_All_Relays_Off

|----[NO] Emergency_Stop_Detected -------(OUT) Display_Emergency_Message

|----[NO] Check_Emergency_Conditions ----(OUT) System_Temperature_Alarm

|----[NO] System_Temperature_Alarm ------(OUT) Disable_Non_Critical_Relays

|----[NO] System_Temperature_Alarm ------(OUT) Display_Temperature_Warning

|----[NO] Check_Emergency_Conditions ----(OUT) Voltage_Fault_Detected

|----[NO] Voltage_Fault_Detected --------(OUT) Disable_All_Outputs

|----[NO] Voltage_Fault_Detected --------(OUT) Display_Voltage_Warning

| // Перевірка Watchdog Reset

|----[NO] Main_Loop_Active --------------(OUT) Kick_Watchdog_Timer

| // Обробка користувацьких сценаріїв

|----[NO] Main_Loop_Active --------------(OUT) Process_User_Scripts

|----[NO] Process_User_Scripts ----------(OUT) Script_1_Check_Condition

|----[NO] Script_1_Check_Condition ------(OUT) Execute_Script_1_Actions

|----[NO] Process_User_Scripts ----------(OUT) Script_2_Check_Condition

|----[NO] Script_2_Check_Condition ------(OUT) Execute_Script_2_Actions

|----[NO] Process_User_Scripts ----------(OUT) Script_3_Check_Condition

|----[NO] Script_3_Check_Condition ------(OUT) Execute_Script_3_Actions

| // Підготовка до завершення циклу

|----[NO] Main_Loop_Active ----------(OUT) Save_System_State

|----[NO] Save_System_State ---------- (OUT) Save_Button_States

|----[NO] Save_System_State ---------- (OUT) Save_Relay_States

|----[NO] Save_System_State ---------- (OUT) Save_ADC_Data

|----[NO] Save_System_State ---------- (OUT) Save_Timers_State

|----[NO] Main_Loop_Active ----------(OUT) Clear_Temporary_Variables

|----[NO] Main_Loop_Active ----------(SET) Ready_For_New_Cycle

|----[NO] Ready_For_New_Cycle ----------(SET) System_Reset

| // Діагностика зв'язку RS485

|----[NO] Main_Loop_Active ----------(OUT) RS485_Link_Check

|----[NO] RS485_Link_Check ----------(OUT) Check_Last_Packet_Time

|----[NO] Check_Last_Packet_Time --------(OUT) RS485_Timeout_Detected

|----[NO] RS485_Timeout_Detected --------(OUT) Display_RS485_Error

|----[NO] RS485_Link_Check ----------(OUT) RS485_Valid_Link

|----[NO] RS485_Valid_Link ----------(OUT) Clear_RS485_Error_Indicator

| // Перевірка цілісності системи

|----[NO] Main_Loop_Active ----------(OUT) System_Integrity_Check

|----[NO] System_Integrity_Check --------(OUT) Check_Memory_Errors

|----[NO] Check_Memory_Errors ----------(OUT) Memory_Error_Detected

|----[NO] Memory_Error_Detected ---------(OUT) Display_Memory_Error

|----[NO] Memory_Error_Detected ---------(OUT) Force_System_Safe_Mode

|----[NO] System_Integrity_Check --------(OUT) Check_Stack_Overflow

|----[NO] Check_Stack_Overflow ----------(OUT) Stack_Overflow_Detected

|----[NO] Stack_Overflow_Detected -------(OUT) Display_Stack_Error

|----[NO] Stack_Overflow_Detected -------(OUT) Force_System_Safe_Mode

| // Самодіагностика системи

|----[NO] Main_Loop_Active --------------(OUT) Self_Test_Cycle

|----[NO] Self_Test_Cycle ---------------(OUT) Check_CPU_Temperature

|----[NO] Check_CPU_Temperature ---------(OUT) CPU_Overheat_Detected

|----[NO] CPU_Overheat_Detected ---------(OUT) Reduce_CPU_Load

|----[NO] CPU_Overheat_Detected ---------(OUT) Display_CPU_Temperature_Warning

|----[NO] Self_Test_Cycle ---------------(OUT) Check_Power_Supply_Stability

|----[NO] Check_Power_Supply_Stability --(OUT) Power_Supply_Fault_Detected

|----[NO] Power_Supply_Fault_Detected ---(OUT) Display_Power_Supply_Error

|----[NO] Power_Supply_Fault_Detected ---(OUT) Enter_Safe_Mode

| // Логування подій

|----[NO] Main_Loop_Active --------------(OUT) Event_Logger_Update

|----[NO] Event_Logger_Update -----------(OUT) Log_Button_Changes

|----[NO] Event_Logger_Update -----------(OUT) Log_Relay_Changes

|----[NO] Event_Logger_Update -----------(OUT) Log_RS485_Events

|----[NO] Event_Logger_Update -----------(OUT) Log_Emergency_Events

| // Завершення циклу

|----[NO] Main_Loop_Active --------------(OUT) Final_System_Check

|----[NO] Final_System_Check ------------(OUT) Update_Watchdog_Counter

|----[NO] Final_System_Check ----------(OUT) Verify_All_Relays_States

|----[NO] Final_System_Check ----------(OUT) Verify_LCD_Status

|----[NO] Main_Loop_Active ----------(OUT) Clear_Temporary_Flags

|----[NO] Main_Loop_Active ----------(OUT) Clear_Communication_Buffers

|----[NO] Main_Loop_Active ----------(OUT) Prepare_For_Next_Cycle

|----[NO] Prepare_For_Next_Cycle --------(SET) Ready_For_New_Cycle

|----[NO] Ready_For_New_Cycle ----------(SET) System_Reset

| // Скидання системи для нового циклу

|----[NO] System_Reset ----------(OUT) Reset_Indicator

|----[NO] System_Reset ----------(OUT) Clear_Command_Buffer

|----[NO] System_Reset ----------(OUT) Clear_Button_States

|----[NO] System_Reset ----------(OUT) Clear_Relay_States

|----[NO] System_Reset ----------(OUT) Clear_LCD_Buffer

|----[NO] System_Reset ----------(OUT) Clear_ADC_Buffer

|----[NO] System_Reset ----------(OUT) Reset_Watchdog_Counter

|----[NO] System_Reset ----------(OUT) Clear_Event_Log

|----[NO] System_Reset ----------(SET) System_Ready

У результаті реалізації програмного забезпечення логічного контролера та підпрограм взаємодії по інтерфейсу RS485 забезпечено стабільний та надійний обмін даними між контролером і операторською станцією.

На рисунку 3.7 наведено приклад діалогового вікна моніторингу обміну по RS485, що демонструє результат роботи розробленого коду у процесі реальної експлуатації системи.

У діалоговому вікні відображається відправлена команда, що містить адресу пристрою, код команди та передані дані. Відповідно до реалізованого алгоритму, контролер формує та передає структуровану відповідь у вигляді

пакета даних. Отримана відповідь відображає актуальний стан системи, зокрема стани вихідних реле, зчитані аналогові дані або службову інформацію.

Окрім цього, у вікні надається оперативна інформація про статус зв’язку (наприклад, "OK" або повідомлення про помилку), кількість виявлених помилок передачі по каналу RS485, а також час останнього успішного обміну. Наявність таких індикаторів дозволяє оператору швидко оцінити якість зв’язку та коректність роботи контролера у мережі.

Таким чином, наведене діалогове вікно є наочним підтвердженням коректної роботи підпрограми Process_RS485_Command та забезпечення повноцінного двостороннього обміну даними у складі проектованої системи.

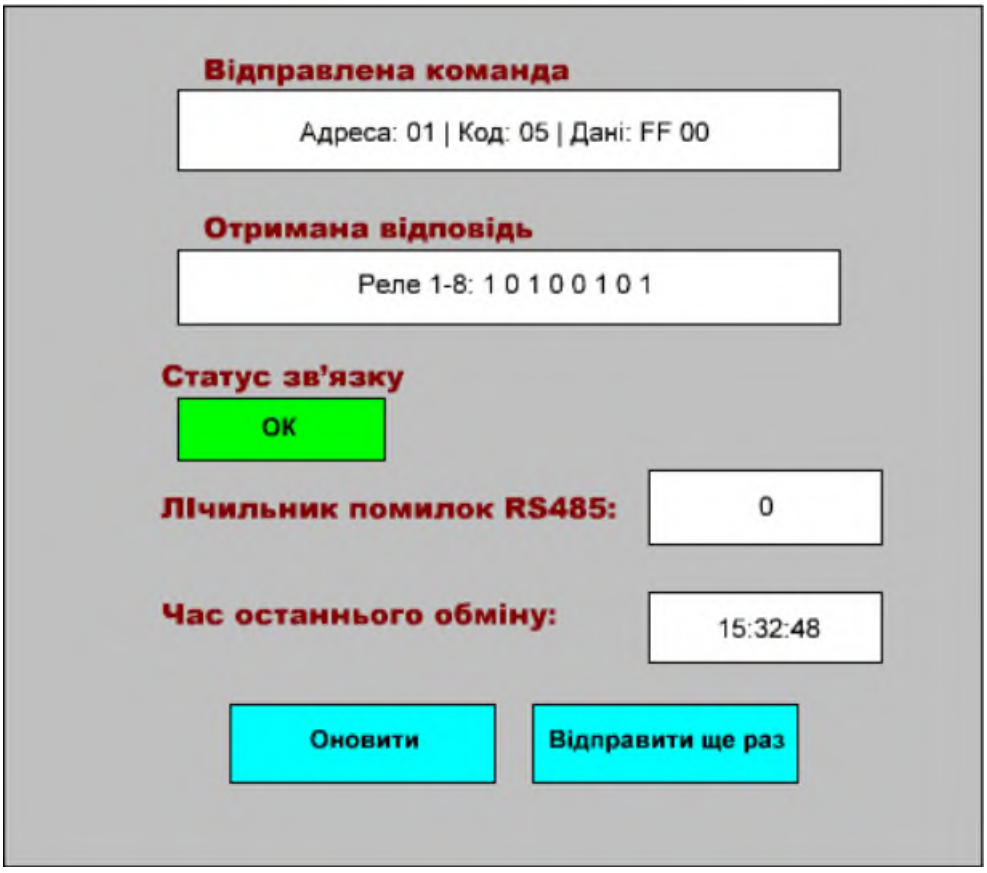


Рисунок 3.7 — Діалогове вікно відображення результатів обміну по інтерфейсу RS485 логічного контролера

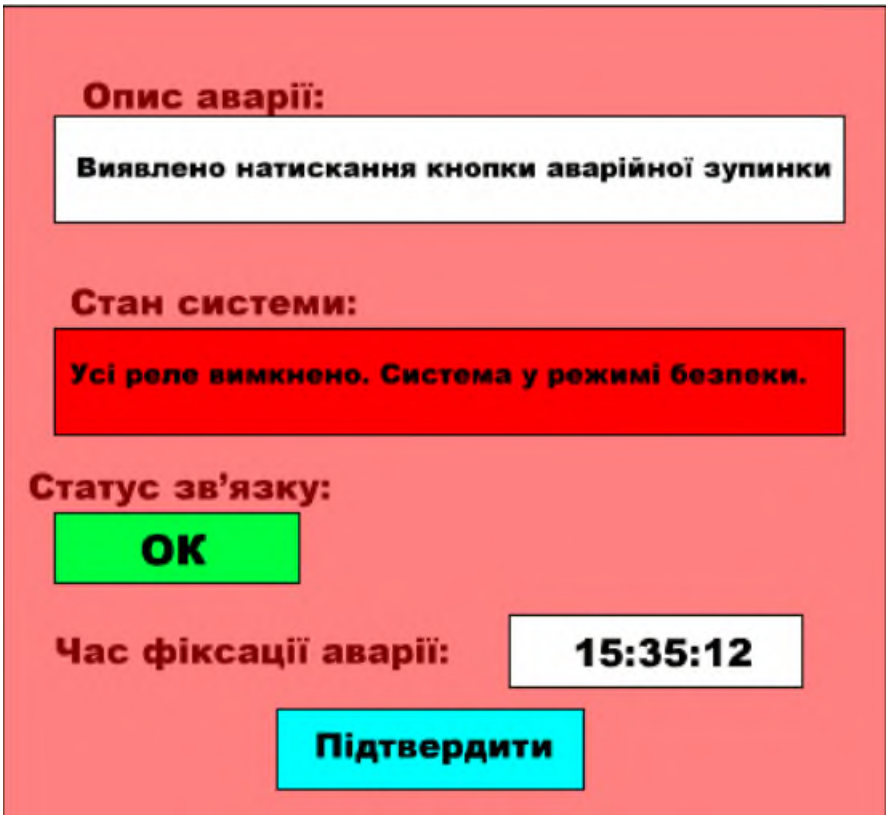


Рисунок 3.8 — Діалогове вікно відображення аварійного повідомлення Emergency STOP при роботі логічного контролера по RS485.

Особливу увагу при розробленні програмного забезпечення логічного контролера було приділено реалізації механізмів безпеки та аварійного захисту системи.

На рисунку 3.8 наведено приклад діалогового вікна, яке формується у разі активації аварійної ситуації — зокрема, натискання кнопки аварійної зупинки (Emergency STOP).

Реалізована у програмному коді підпрограма Check_Emergency_Conditions здійснює постійний моніторинг аварійних входів та інших критичних параметрів системи. У разі виявлення аварійної ситуації програмне забезпечення виконує примусове відключення усіх виконавчих механізмів (Relay_Force_OFF), переводить контролер у безпечний режим та ініціює формування аварійного повідомлення, яке передається по інтерфейсу RS485.

Діалогове вікно, яке демонструється оператору, містить текстове пояснення причини аварії, поточний стан системи (наприклад, "Усі реле вимкнено. Система у режимі безпеки."), статус зв’язку, а також час фіксації події. Наявність кнопки "Підтвердити" дозволяє оператору зафіксувати факт ознайомлення з аварійною ситуацією перед відновленням нормальної роботи.

Таким чином, представлене діалогове вікно є результатом роботи підпрограми аварійного захисту, що дозволяє гарантувати безпечну експлуатацію автоматизованої системи у складі розробленого логічного контролера.

Розроблений програмний код логічного контролера забезпечив повноцінну реалізацію функціональної схеми системи та передбачених алгоритмів керування. Представлені діалогові вікна наочно демонструють результати роботи підпрограм обробки команд по RS485 та механізмів аварійного захисту, підтверджуючи ефективність та надійність реалізованого програмного забезпечення. Отримані результати свідчать про готовність контролера до використання у реальних умовах промислової автоматизації

Висновок за розділом 3

У третьому розділі дипломної роботи було здійснено розроблення програмної частини логічного контролера на базі мікроконтролера PIC16F877A. На основі побудованої функціональної схеми та алгоритму роботи системи сформовано деталізовану структуру програмного забезпечення контролера.

Були розроблені та описані основний алгоритм роботи програми, а також окремі підпрограми, що відповідають за ключові функціональні блоки: ініціалізацію апаратної частини, обробку дискретних входів, перетворення аналогових сигналів, керування виконавчими механізмами (реле), організацію комунікаційного обміну по інтерфейсу RS485, індикацію стану системи на LCD-дисплеї, а також механізми самодіагностики і аварійного захисту.

Для наочного представлення логіки роботи програми було створено псевдокод у структурованому форматі LADDER-логіки, що дозволяє чітко відобразити взаємозв’язки між окремими функціональними блоками та спрощує сприйняття алгоритму роботи контролера. Псевдокод охоплює повний цикл функціонування системи, включаючи початкову ініціалізацію, основний цикл обробки подій у реальному часі та механізми захисту від збоїв.

Результатом виконаної роботи є сформована програмна структура, яка забезпечує стабільну та ефективну роботу логічного контролера у складі системи автоматизації. Розроблене програмне забезпечення дозволяє здійснювати як локальне керування процесами за допомогою дискретних входів, так і віддалене керування та моніторинг по промисловій мережі RS485. Забезпечено гнучкість та масштабованість системи, можливість розширення функціональності та адаптації до різних вимог об’єкта автоматизації.

Таким чином, третій розділ завершив комплексний процес проєктування логічного контролера, створивши основу для його практичної реалізації та впровадження у системи промислової автоматизації.

РОЗДІЛ 4
ОХОРОНА ПРАЦІ

4.1. Аналіз шкідливих та небезпечних факторів при роботі з програмованим логічним контролером

У процесі розроблення, налагодження, експлуатації та обслуговування програмованого логічного контролера (ПЛК), як і будь-якої електронної апаратури, існують певні потенційні небезпечні та шкідливі виробничі фактори, які можуть впливати як на працівників, так і на працездатність самої системи. Аналіз таких факторів є необхідною складовою забезпечення безпечних умов праці при роботі з автоматизованими системами.

У загальному випадку робота з ПЛК передбачає взаємодію персоналу з різними видами обладнання: електронними модулями, джерелами живлення, кабельними мережами, засобами індикації та керування, периферійними виконавчими механізмами, а також з програмним забезпеченням через комп’ютерні системи.

Враховуючи це, можна виділити кілька основних груп шкідливих та небезпечних факторів, які мають бути враховані при організації робочого місця та розробленні заходів з охорони праці.

Перш за все, значну небезпеку становить вплив електричного струму. Незважаючи на те, що безпосередньо логічна частина ПЛК працює від низьковольтного джерела живлення (зазвичай +5 В), система керує силовими елементами, такими як реле, котушки яких можуть працювати від напруги 12 В або 24 В, а комутовані силові ланцюги — від мережі змінного струму 220 В. Це створює реальну небезпеку ураження електричним струмом у разі порушення ізоляції, неправильного підключення або втручання у роботу обладнання при знятому захисті.

Особливо небезпечною є ситуація, коли монтаж або технічне обслуговування проводиться при підключеному живленні силової частини. У

такому випадку працівник може випадково торкнутися струмоведучих частин, що призведе до електротравми. Тому під час виконання будь-яких робіт з ПЛК повинна суворо дотримуватись вимога повного знеструмлення системи.

Наступним важливим фактором є можливий вплив електромагнітного випромінювання. Під час роботи ПЛК, особливо при інтенсивному обміні даними по інтерфейсу RS485, можливе виникнення електромагнітних полів у навколишньому середовищі. При проектуванні системи особливу увагу було приділено правильному укладанню кабелів та використанню екранованих з'єднань для мінімізації електромагнітних завад. Однак у процесі технічного обслуговування або діагностики слід враховувати можливість впливу таких полів на чутливі елементи електронного обладнання, розташованого поблизу.

Ще одним шкідливим фактором є можливий вплив іонізуючих випромінювань та статичної електрики. Хоча у стандартних умовах експлуатації систем автоматизації ці фактори є мінімальними, при роботі у промислових умовах, де присутні джерела статичної електрики або підвищений рівень радіаційного фону (наприклад, у хімічній промисловості або на енергетичних об'єктах), необхідно враховувати можливість накопичення електростатичних зарядів на елементах схеми ПЛК. Це може призвести до електростатичного пробою та виходу з ладу мікросхем. Тому при монтажі, налагодженні та обслуговуванні системи необхідно використовувати антистатичні засоби захисту (браслети, заземлені килимки тощо).

Важливим аспектом є також наявність механічних небезпек. При роботі з ПЛК персонал може взаємодіяти з релейними виходами, які керують виконавчими механізмами — електромагнітними клапанами, двигунами, пневматичними та гідравлічними приводами. При неправильному або необережному обслуговуванні ці механізми можуть почати рухатись у несподіваний момент, створюючи загрозу травмування. Особливо небезпечною є робота у системах, де відсутні міжопераційні блокування або де не передбачено блокування пуску при відкритих захисних кожухах. У таких випадках слід обов'язково впроваджувати програмні та апаратні заходи

захисту, що блокують можливість активації виконавчих механізмів при відкритих зонах обслуговування.

Також слід враховувати температурні фактори. При тривалій роботі або у разі неправильної організації системи вентиляції можливо локальне перегрівання елементів ПЛК, зокрема мікроконтролера *PIC*16*F*877A, драйвера *ULN*2803 та мікросхеми *MAX*485. Перегрівання може спричинити як зниження надійності, так і фізичне пошкодження компонентів. При виконанні монтажу системи необхідно забезпечити належний тепловідвід, а також передбачити температурний моніторинг, особливо у компактних закритих шафах керування.

Під час роботи з ПЛК оператор або інженер-технік значну частину часу проводить за комп'ютером під час програмування контролера або аналізу даних, що створює шкідливий ергономічний фактор — зорове напруження та статичне навантаження на опорно-руховий апарат. Недостатньо організоване робоче місце програміста або інженера автоматизації може призвести до розвитку професійних захворювань. Тому при обладнанні робочого місця слід дотримуватись стандартів з ергономіки: належне освітлення, правильне розташування монітора, зручне крісло, регулярні перерви у роботі.

Не можна ігнорувати також психофізіологічні фактори. Робота з програмованим логічним контролером передбачає високу відповідальність, необхідність постійної концентрації уваги, швидкого реагування на позаштатні ситуації та аналізу великої кількості інформації. Це може призводити до підвищеної нервово-психічної напруги у персоналу, особливо при роботі у режимі налагодження або ліквідації аварійних ситуацій. Для зниження негативного впливу слід регламентувати тривалість змін, забезпечувати можливість відпочинку та впроваджувати психологічну підтримку персоналу [13].

Крім зазначених факторів, слід враховувати також пожежну небезпеку. Незважаючи на те, що основна апаратна частина ПЛК працює від низьковольтного джерела живлення, у системі присутні силові ланцюги під

високою напругою (220 В), комутація яких здійснюється за допомогою реле. При неправильному виборі елементів, порушенні режимів експлуатації або несправності може виникнути перегрівання контактів, іскріння або коротке замикання, що створює ризик займання.

Для мінімізації пожежної небезпеки слід використовувати реле, розраховані на відповідний струм та напругу, забезпечити наявність захисних пристроїв (запобіжників, автоматичних вимикачів), а також дотримуватись вимог до пожежної безпеки при проектуванні шафи керування та розміщенні електронного обладнання.

Таким чином, при роботі з програмованим логічним контролером існує цілий спектр шкідливих та небезпечних виробничих факторів, які необхідно враховувати для забезпечення безпечних умов праці. Правильний аналіз цих факторів дозволяє розробити ефективні заходи захисту персоналу та підвищити надійність роботи автоматизованої системи у цілому.

4.2. Заходи з охорони праці та електробезпеки при експлуатації автоматизованої системи

Забезпечення охорони праці при експлуатації автоматизованої системи на базі програмованого логічного контролера (ПЛК) є одним із найважливіших аспектів організації безпечної роботи виробничого персоналу. Правильно розроблені заходи з охорони праці дозволяють мінімізувати ризики виникнення аварійних ситуацій, знизити ймовірність отримання травм, запобігти пошкодженню обладнання та забезпечити високу надійність функціонування системи у цілому.

Загальна концепція забезпечення охорони праці у системах автоматизації передбачає впровадження комплексу технічних, організаційних та програмних заходів. Розглянемо ключові напрямки, які було враховано при розробці та впровадженні проектованої системи ПЛК.

Перш за все, особливу увагу слід приділити заходам з електробезпеки. У системі присутні силові ланцюги з напругою до 220 В, керовані за допомогою релейних виходів. Для запобігання ураженню електричним струмом усі елементи силової частини розміщуються у металевій або пластиковій шафі керування, що забезпечує фізичний захист користувача від випадкового дотику до струмоведучих частин.

Корпус шафи підлягає обов’язковому заземленню відповідно до вимог нормативної документації. Усі кабелі силових ланцюгів мають бути прокладені у кабельних лотках або трубах, що виключає можливість їх механічного пошкодження. Використовуються лише сертифіковані кабелі, розраховані на відповідні струмові та температурні навантаження.

Важливим елементом системи електробезпеки є впровадження засобів автоматичного захисту. У проектованій системі передбачено застосування автоматичних вимикачів та запобіжників, які відключають живлення у разі короткого замикання або перевищення допустимого струму. Це дозволяє запобігти як ураженню працівників, так і пошкодженню обладнання у разі аварійної ситуації [20].

Особливу увагу слід приділити заходам щодо захисту від електростатичного розряду (*ESD*). При роботі з мікроконтролером *PIC*16*F*877A, мікросхемою *MAX*485 та іншими компонентами цифрової частини існує ризик пошкодження електроніки внаслідок накопичення електростатичних зарядів. Для захисту від *ESD* персонал зобов’язаний використовувати антистатичні браслети під час монтажу та ремонту електронних модулів. Робочі місця повинні бути обладнані заземленими антистатичними килимками, що запобігає накопиченню зарядів на поверхні столу.

Усі монтажні роботи проводяться за умови повного знеструмлення системи. Підключення живлення дозволяється лише після завершення всіх підключень та перевірки правильності монтажу. Персонал, який виконує монтажні та пусконалагоджувальні роботи, повинен пройти відповідне

навчання та мати допуск до роботи з електроустановками напругою понад 50В.

Ще одним важливим напрямком є забезпечення пожежної безпеки. Система автоматизації повинна бути спроектована таким чином, щоб мінімізувати ризики виникнення пожежі. Усі використовувані елементи повинні відповідати вимогам пожежної безпеки, а силові ланцюги мають бути оснащені відповідними захисними пристроями [10].

У приміщенні, де розміщується шафа керування, повинно бути передбачено наявність первинних засобів пожежогасіння — вогнегасників, придатних для гасіння електричних установок (вуглекислотних або порошкових). Працівники повинні бути проінструктовані щодо порядку дій у разі пожежі, вміти користуватись наявними засобами пожежогасіння та знати місцезнаходження аварійного вимикача електроживлення.

Важливе значення мають заходи з організації безпечної експлуатації автоматизованої системи. Для цього у програмному забезпеченні ПЛК передбачено механізми програмного блокування, що запобігають несанкціонованій активації виконавчих механізмів. Наприклад, активація релейних виходів можлива лише після проходження етапу ініціалізації системи та перевірки наявності усіх необхідних умов для безпечного запуску.

Також у системі реалізовано підтримку аварійного режиму (*Emergency STOP*). При активації відповідного вхідного сигналу усі релейні виходи примусово відключаються, що дозволяє швидко перевести технологічне обладнання у безпечний стан [11].

Усі елементи системи проходять регулярне технічне обслуговування та перевірку. Особливу увагу приділяють стану контактів реле, кабельних з’єднань та стану заземлення. Несправні елементи підлягають негайній заміні.

З метою забезпечення безпеки при роботі з програмованим логічним контролером важливим є також дотримання ергономічних вимог до організації робочого місця. Оператор або інженер-програміст, який взаємодіє із системою через комп’ютер та засоби індикації, повинен працювати у

комфортних умовах. Робоче місце повинно мати належне освітлення, зручне розташування монітора на рівні очей, регульоване крісло. Рекомендується робити регулярні перерви для зниження зорового та м’язового навантаження.

Також необхідно враховувати питання психофізіологічної безпеки персоналу. Працівники повинні мати чітко регламентований робочий графік, що забезпечує достатній час для відпочинку. Особливо важливо уникати надмірної нервово-психічної напруги при ліквідації аварійних ситуацій. Для цього необхідно чітко регламентувати порядок дій у нестандартних ситуаціях та забезпечити наявність відповідної документації.

Усі працівники, які обслуговують автоматизовану систему, проходять обов’язковий інструктаж з охорони праці перед початком роботи, а також періодичний інструктаж відповідно до вимог чинного законодавства. Особливу увагу приділяють знанням правил електробезпеки, дій у разі пожежі та правил роботи з комп’ютерною технікою [16].

Таким чином, впровадження комплексу технічних, організаційних та програмних заходів забезпечує високу безпеку при експлуатації автоматизованої системи на базі програмованого логічного контролера. Дотримання вимог охорони праці дозволяє мінімізувати ризики для персоналу, забезпечити надійність та довговічність роботи обладнання та створити умови для ефективної та безпечної роботи у виробничому середовищі.

Висновок за розділом 4

У четвертому розділі дипломної роботи було проведено аналіз шкідливих та небезпечних виробничих факторів, що можуть виникати при розробці та експлуатації програмованого логічного контролера, а також розроблено комплекс заходів з охорони праці та електробезпеки. Особливу увагу приділено питанням захисту від ураження електричним струмом,

забезпеченню пожежної безпеки, електромагнітної сумісності та захисту від електростатичних розрядів.

Запропоновані організаційні та технічні заходи спрямовані на створення безпечних умов праці для персоналу, підвищення надійності та довговічності роботи автоматизованої системи. Дотримання розроблених рекомендацій дозволить мінімізувати виробничі ризики та забезпечити ефективну експлуатацію ПЛК у складі систем промислової автоматизації.

ВИСНОВОК

У даній дипломній роботі було розроблено логічний контролер на базі мікроконтролера *PIC*16*F*877A для застосування у системах промислової автоматизації. Проведено комплексне дослідження та проєктування як апаратної, так і програмної частини системи з урахуванням вимог до надійності, функціональності та безпеки експлуатації.

У першому розділі розглянуто різновиди програмованих логічних контролерів та їх призначення у сучасних системах автоматизації, а також проведено аналіз типів входів і виходів ПЛК, їх функціонального призначення та принципів підключення. Це забезпечило базу для вибору оптимальної конфігурації системи.

У другому розділі було виконано розроблення апаратної частини логічного контролера. Описано проектовану схему на базі мікроконтролера *PIC*16*F*877A, проведено аналіз характеристик мікроконтролера, структури пам’яті, портів введення/виведення та реалізації обміну даними за допомогою інтерфейсу *RS*485 на основі мікросхеми *MAX*485.

У третьому розділі розроблено функціональну схему, алгоритм роботи системи та програмне забезпечення контролера. Створено псевдокод програми у форматі *LADDER*-логіки, розроблено підпрограми для обробки команд *RS*485, керування реле, обробки входів, індикації та аварійного захисту. Діалогові вікна, що було представлено як результат роботи коду, демонструють правильність реалізації логіки обміну та наявність механізмів безпечної роботи системи.

У четвертому розділі розглянуто питання охорони праці при експлуатації розробленого логічного контролера. Проведено аналіз можливих небезпечних факторів, визначено заходи щодо забезпечення електробезпеки та організації безпечних умов праці при роботі з автоматизованими системами.

В цілому, виконана дипломна робота підтвердила практичну можливість створення на базі недорогого мікроконтролера *PIC*16*F*877A

повноцінного логічного контролера з широким функціоналом, який може застосовуватись у різних галузях промислової автоматизації. Отримані результати дозволяють рекомендувати розроблену систему для подальшого впровадження та використання у складі промислових об’єктів.

Список використаних джерел

1. IEC 61131-3:2013. Programmable controllers – Part 3: Programming languages. International Electrotechnical Commission. — 2013.
2. Bolton W. Programmable Logic Controllers: Principles and Applications. 6th ed. Newnes, Elsevier, 2015.
3. Мікроконтролери Microchip. Офіційний сайт Microchip Technology Inc. [Електронний ресурс]. — Режим доступу: https://www.microchip.com
4. Microchip. PIC16F877A Datasheet. Microchip Technology Inc. — 2021.
5. Texas Instruments. MAX485 Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers. Datasheet. — 2020.
6. ULN2803A Datasheet. Texas Instruments / STMicroelectronics. — 2021.
7. National Instruments. Industrial Communication Protocols Overview. — National Instruments White Paper, 2020.
8. Hughes T. Programmable Logic Controllers. 5th ed. Newnes, 2011.
9. Documentation Modbus over Serial Line. Modbus Organization. — 2022.
10. Резнік О.І. Основи електробезпеки у виробничих умовах: навч. посіб. — Львів: Видавництво Львівської політехніки, 2020
11. ДСТУ EN 60204-1:2015. Безпека машин. Електрообладнання машин. Частина 1: Загальні вимоги. — Київ, 2015.
12. Allen-Bradley. PLC Programming Handbook. Rockwell Automation. — 2020.
13. Кисіль І.М. Охорона праці в галузі інформаційних технологій: навч. посіб. — К.: КНУБА, 2019.
14. Design guidelines for EMI/EMC in industrial automation. Siemens AG, 2021.

15. Siemens. S7-1200 Programmable controller system manual. Siemens AG, 2021.

16. Програма "Охорона праці": ДСТУ ISO 45001:2019. Системи управління охороною праці та безпекою здоров’я на роботі. — Київ, 2019.

17. Johnson C.D. Process Control Instrumentation Technology. 8th ed. Prentice Hall, 2013.

18. Modern Industrial Automation Software Design. — Springer, 2021.

19. Sedra A., Smith K. Microelectronic Circuits. 7th ed. Oxford University Press, 2015.

20. International Safety Standards for Electrical Equipment. IEC, 2020.

ДОДАТОК А

МУЛЬТИМЕДІЙНА ПРЕЗЕНТАЦІЯ

Кваліфікаційна робота для здобуття ступеню магістр на тему:

«Проєктування логічного контролеру на базі мікроконтролеру Microchip»

Виконав: студент групи 4317ст3
Тимчук Ілля Сергійович
Керівник: PhD, Козлов А.Ю.

Слайд 1: Титульний слайд

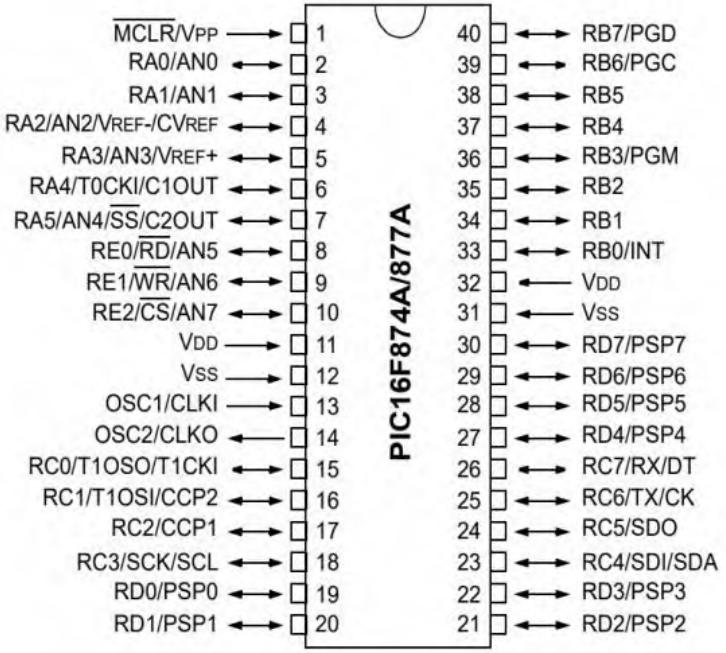
Мета: розроблення програмованого логічного контролера на базі мікроконтролера PIC16F877A з підтримкою обміну даними по інтерфейсу RS485 для використання у системах промислової автоматизації.

Об'єкт дослідження: програмований логічний контролер на базі мікроконтролера PIC16F877A.

Предмет дослідження: методи та засоби проєктування апаратної та програмної частини логічного контролера для систем промислової автоматизації, організація обміну даними по інтерфейсу RS485.

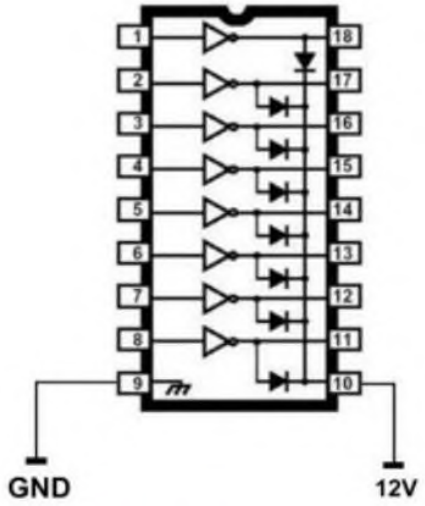
Слайд 2: Мета роботи, об'єкт та предмет дослідження

Мікроконтролер PIC16F877A

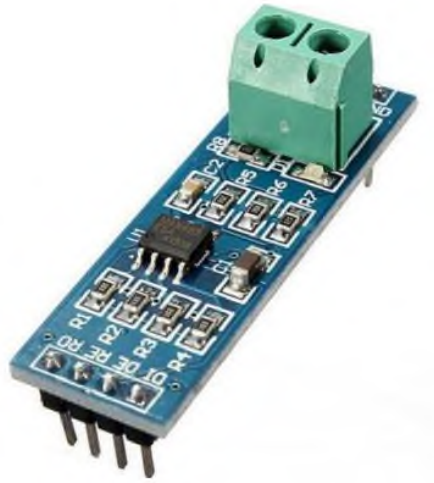


Слайд 3: Мікроконтролер PIC16F877A

Силові ключі для керування релейними виходами та реалізація зв’язку за протоколом RS485



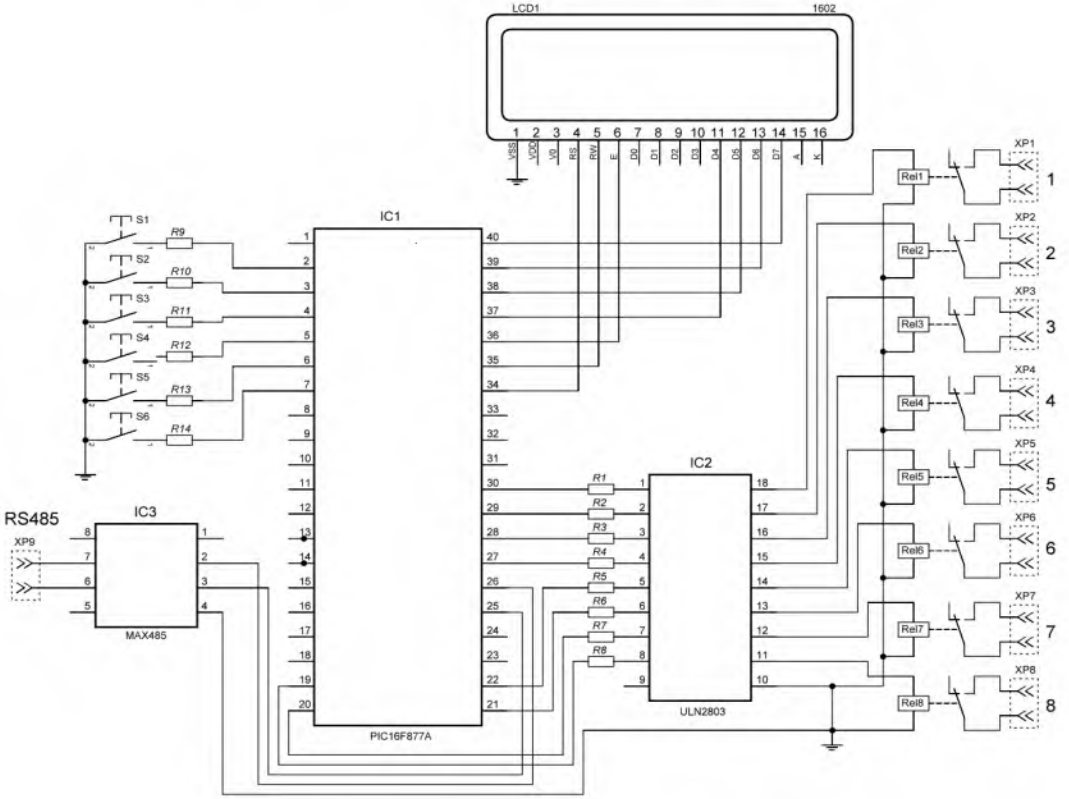
Структурна схема та призначення виводів драйвера силових ключів ULN2803A.



Модуль перетворювача RS485 (MAX485) для інтерфейсу Arduino

Слайд 4: Силові ключі для керування релейними виходами та реалізація зв’язку за протоколом RS485

Принципова електрична схема програмованого логічного контролера на базі мікроконтролера PIC16F877A



Слайд 5: Принципова електрична схема програмованого логічного контролера на базі мікроконтролера PIC16F877A

Функціональна схема проєктованого логічного контролера на базі мікроконтролера PIC16F877A

Слайд 6: Функціональна схема проєктованого логічного контролера на базі мікроконтролера PIC16F877A

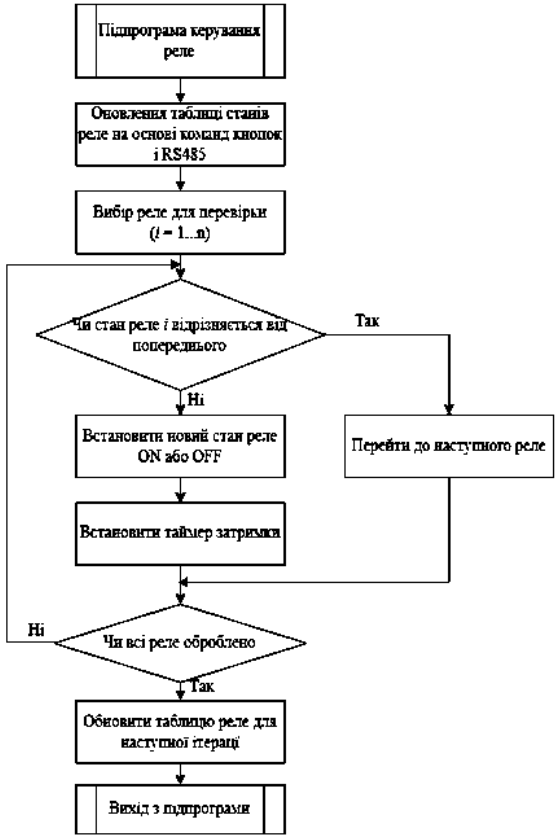
Алгоритми керуючої програми

Основний алгоритм програми

Алгоритм підпрограми Обробки команд по RS485

Слайд 7: Алгоритми керуючої програми

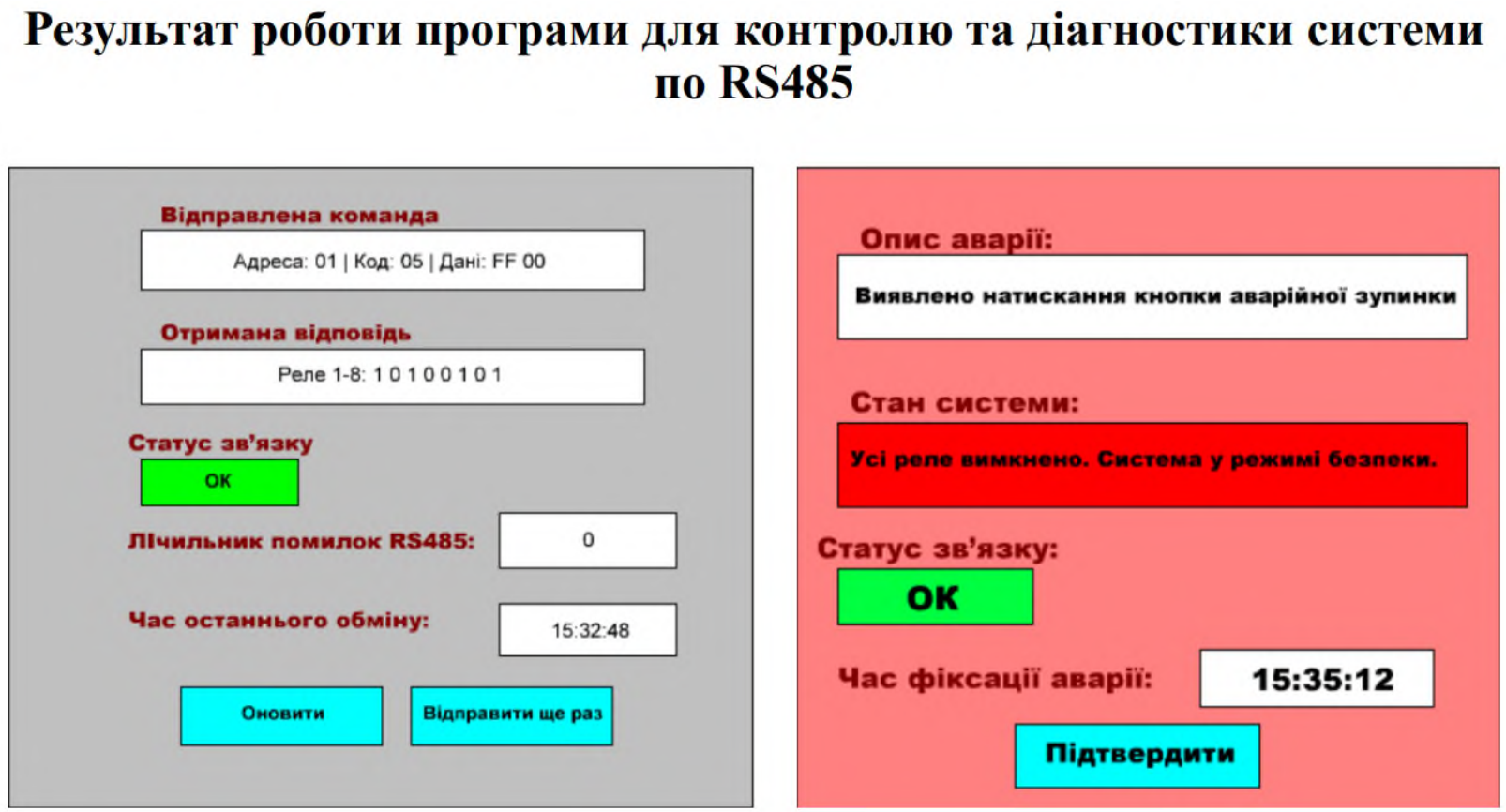
Алгоритми підпрограм



Алгоритм підпрограми керування реле

Алгоритм підпрограми відповіді по RS485

Слайд 7: Алгоритми підпрограм



Слайд 9: Результат роботи програми для контролю та діагностики системи по RS485

Висновки:

- розроблено логічний контролер на базі PIC16F877A для промислових систем автоматизації.

- реалізовано обмін даними через інтерфейс RS485 та алгоритми керування системою у форматі LADDER-логіки.

- забезпечено захист та безпечну експлуатацію контролера згідно з вимогами охорони праці.

- проведено тестування системи, що підтвердило її здатність забезпечувати безперервну та стабільну роботу.

Слайд 10: Висновки